Multi-contact Humanoid Posture Computation on Humans Point Cloud

Anastasia Bolotnikova, Sébastien Courtois, Abderrahmane Kheddar

Multi-contact Humanoid Posture Computation on Humans Point Cloud

Anastasia Bolotnikova[1,2], Sébastien Courtois[1], Abderrahmane Kheddar[2]

***Abstract*— For robots to interact with human in close proximity safely and efficiently, a specialized method to compute whole-body robot posture and plan contact locations is required. In this work we focus on physical interaction between a robot and a human with intent to assist the human to realize a motion task. We propose a method for posture generation on an arbitrary point cloud and describe in detail its implementation and application in human robot interaction. The proposed method allows to plan whole-body posture and contact locations on a parametric NURBS surface, fitted on RGBD camera data of human body, under robot equilibrium, friction cone, torque/joint limits, collision avoidance, and assistance task inherent constraints. The proposed framework is capable to deal in a uniform fashion with any arbitrary point cloud, of various shapes any sizes, for autonomously planing the contact locations and interaction forces on potentially moving, movable, and deformable surfaces, which occur in direct physical human robot interaction. We conclude the paper with examples of posture generation for physical human-robot interaction scenarios.**

I. Introduction

It is well-known that the world is facing an ageing population problem [1]. Our objective is to use humanoid technology to partly compensate for the lack of sufficient workforce in the caregiving sector, by enabling it to assist with daily-life motions of frail or ageing persons in well-identified situations to help support person's autonomy [2]. For example, help in walking safely, sit-to-stand transfer, getting out or in bed, etc. The assistance consists mainly in providing support by physical contacts through which extra-torque or guidance are provided to the human in achieving any such demanding task. Contrarily to some trend in this domain [3], [4], a robot, whatsoever is not expected to hold with full power a person that still has some capabilities of motion. We rather see the robot as a companion that could play the role of a reconfigurable static or moving hurdle to provide contact support to a person to help with motion.

A humanoid robot has a potential to be used as a reconfigurable and mobile multi-functional assistive support structure. Thus, one such platform can be used to assist in various tasks, unlike simpler robots that are designed for a specific task. Moreover, friendly anthropomorphic appearance of the robot can result in overall better usability due to higher likeability [5], more intuitive communication [6], [7] and easier gain of user trust and acceptance of the technology [8].

To reach our objective, several subtasks need to be addressed, including understanding of human motion and intent, adaptive control for assistance and positioning of the robot w.r.t. a human, which is the main focus of this work. We answer the question: how to compute a robot configuration in close contact with a human, in order to plan contact locations on an arbitrary point-cloud that represents moving, movable and deformable human body part, while accounting for human safety and comfort, as well as robot and assistance task inherent constraints?

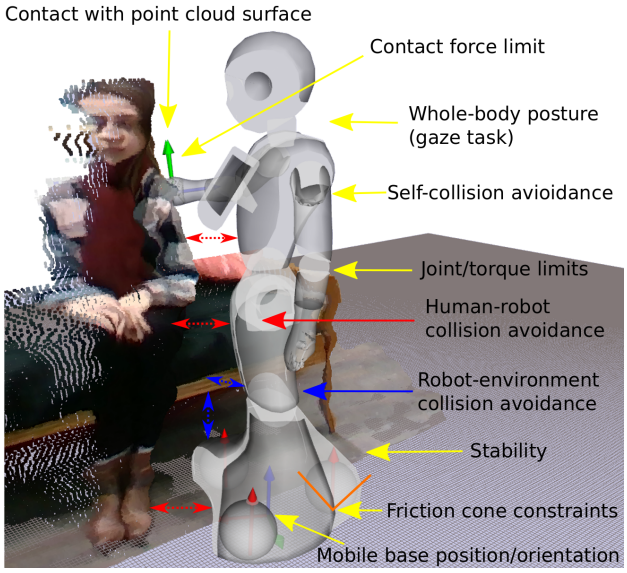

Fig. 1: Robot posture for HRI computed on point cloud.

Our approach builds upon previous work [9], which formulates posture generator (PG) as a non-linear optimization on non-euclidean manifolds. Our extension of this PG allows to plan robot posture as well as interaction forces and contact locations on a point cloud surface of human body part. We propose to fit a Non Uniform Rational B-Spline (NURBS) surface trimmed by NURBS curve on human data acquired from robot embedded RGBD camera. Additionally, we use constraints for the collision avoidance with human to ensure human-robot interaction (HRI) safety. In the same formulation, we include all the robot, task and eventually human inherent constraints, to compute feasible robot configurations that likely guarantee the execution of the assistance task. More explicitly, our contributions are the following:

1) A formulation of the optimization problem for multi-contact PG on an arbitrary point cloud (Sec. II-A);
2) A human point cloud processing framework that provides input to our proposed PG (Sec. II-B);
3) An evaluation of our implementation with various HRI tasks (Fig. 1), discussion and assessment of the computation time (Sec. III).

[1]SoftBank Robotics Europe, Paris, France
[2]University of Montpellier–CNRS LIRMM, Interactive Digital Humans, Montpellier, France

II. Proposed method

For a given patient and assistance requirement, we assume the robot to be knowledgeable of the procedure steps. Indeed, assistance know-how and practices are instructed from the caregivers experience [10]. We also assume the robot able to locate the patient in the environment. Given these previous knowledge, a humanoid robot is asked to simultaneously plan its optimal floating base location (i.e. position and orientation) and its joint configuration (we call both a posture), together with the contact locations on the patient body to apply assistive motion forces. The computed postures must be such that it would enable the robot to supply or resist required contact forces in order to assist desired human motion task while ensuring human comfort and safety. We formulate the posture planing as an optimization problem. The eq. 1 shows the general form of the proposed method.

$$\min_{q,u,v,f} \quad \phi(q, q_h, \mathcal{S}) \tag{1a}$$

$$\text{subject to} \quad q_{\min} \leq q \leq q_{\max} \quad \tau_{\min} \leq \tau(q,f) \leq \tau_{\max} \tag{1b}$$

$$\epsilon_{ij} \leq d(r_i(q), r_j(q)) \quad \forall (i,j) \in \mathcal{I}_{\text{auto}} \tag{1c}$$

$$\epsilon_{ij} \leq d(r_i(q), O_k) \quad \forall (i,k) \in \mathcal{I}_{\text{obj}} \tag{1d}$$

$$\epsilon_{ij} \leq d(r_i(q), h_j(q_h)) \quad \forall (i,j) \in \mathcal{I}_{\text{hum}} \tag{1e}$$

$$s(q,f) = 0 \quad c(f_i) \geq 0 \quad \forall i \in [1,m] \tag{1f}$$

$$f_i^{\min} \leq f_i \leq f_i^{\max} \quad \forall i \in [1,m] \tag{1g}$$

$$P(u_i, v_i) \in \mathcal{N}_i \quad \forall i \in [1,m] \tag{1h}$$

$$u_i, v_i \in \Omega_i^c \quad \forall i \in [1,m] \tag{1i}$$

where, q is the vector of generalized robot coordinates, that includes joint angles and floating/mobile base position and orientation. The sets of collision avoidance link pairs, for robot, human and environment links $r_x(q), h_x(q_h)$ and O_k, are denoted $\mathcal{I}_{\text{auto}}, \mathcal{I}_{\text{hum}}$ and $\mathcal{I}_{\text{obj}}$ respectively. Human posture is denoted by q_h and serves as an input to the optimization problem. Contact locations are parametrized by $P(u_i, v_i)$, where u_i and v_i are decision variables. Robot contact forces (on human or environment) are denoted by f_i.

The assistance contact tasks known a priori are defined by a structure denoted as $\mathcal{S} = \{P_i^r, [\mathcal{N}_i, \Omega_i^c], f_i^{\min}, f_i^{\max}\}$, which includes m pairs of the robot end-effector P_i^r and human body or environment surface $\mathcal{N}_i$ area constrained to parametric space subset Ω_i^c, as well as specification of the bounds $f_i^{\min}$ and $f_i^{\max}$ for the force to be applied/resisted. Constraints eq. 1b ensure that computed posture is within robot joint and torque limits respectively. Constraints eq. 1c–1e avoid undesired self-, robot-environment and robot-human collisions. Constraints eq. 1f are to fulfil robot static balance, and prevent contact slippage by keeping contact forces inside friction cones. So far, the problem is similar to that in [9].

The eq. 1g allows commuting robot postures taking into account the forces that need to be resisted as specified by the assistance task structure $\mathcal{S}$. To enable the framework to plan a contact location on a surface of human body, specified in $\mathcal{S}$, we add eq. 1h to the problem, which restricts the contact to lie on the NURBS surface fitted to the point cloud. To allow our framework to equally handle any arbitrary point cloud, we use additional curve inclusion constraint, eq. 1i, that trims away areas in the parametric space of the surface that are not suitable for contact, e.g. not covered by point cloud. We detail the implementation of such constraints in the following section.

A. Contact constrained to a surface fitted on point cloud

In this section, we detail the implementation of the constraints that restricts the robot end-effector to contact an arbitrary point cloud segmented from the RGBD robot camera data (eq. 1h–1i).

Given a point cloud $D = \{p_k | k = 0, ..., K\}$ and an initial guess of the NURBS surface $\mathcal{N}_i$ control point locations P_{ij}^{init}, a position update c_{ij} for the control points can be computed via quadratic optimization that minimizes the Euclidean distance between points p_k and corresponding closest points projected onto surface $\mathcal{N}(u_k, v_k)$

$$\min_{c_{ij}} \quad \sum_k \left(p_k - \frac{\sum_i \sum_j N_{i,p}(u_k) N_{j,q}(v_k) (P_{ij}^{\text{init}} + c_{ij})}{\sum_i \sum_j N_{i,p}(u_k) N_{j,q}(v_k)} \right)^2 \tag{2a}$$

$$0 \leq u_k, v_k \leq 1 \quad u, v \in \mathbb{R} \tag{2b}$$

where parameters p and q denote surface order, in directions U and V respectively, and $N_{x,y}(z)$ is a nonrational B-spline basis function. The number of control points in the surface can be either predefined or adjusted in the fitting process [11].

Example of a point cloud and fitted NURBS surface are shown in Fig. 2 (left). This figure also illustrates the next issue we need to address, the four-sided nature of the NURBS surface. Since NURBS parametric space is four sided, surface fitted to an arbitrary point cloud will likely need to be trimmed by fitting a constraining closed curve $c(t)$ that encloses the data points, thus defining a subspace of the surface parametric space $\Omega^c \subset UV$ that is suitable for making contact [12]. These curve control points are found by, first, projecting 3D points p_k into surface parameter space $UV \subset \mathbb{R}^2$, to get corresponding 2D points g_k. Given the initial guess of curve control points location P_i^{init}, so called, footpoint parameter t_k is computed for every g_k, so that point $c(t_k)$ on the curve is the closest point to g_k and n_{t_k} is curve normal at this point. The fitting process is done by solving the following optimization problem:

$$\min_{c_i} \quad \sum_k \left(g_k - \frac{\sum_i N_{i,p}(t_k)(P_i^{\text{init}} + c_i)}{\sum_u N_{i,p}(t_k)} \right)^2 \tag{3a}$$

$$(g_k - c(t_k))^T \cdot n_{t_k} \leq 0 \tag{3b}$$

Sample points g_k projected onto a parametric space of the NURBS surface UV along with fitted constraining curve $c(t)$ are shown in Fig. 2 (left). Note that if point cloud fits well under four-sided surface, this operation can be left undone, which saves computation time and simplifies the posture computation optimization formulation.

After surface and curve control point values have been adjusted to fit the input point cloud data, the goal of the PG is

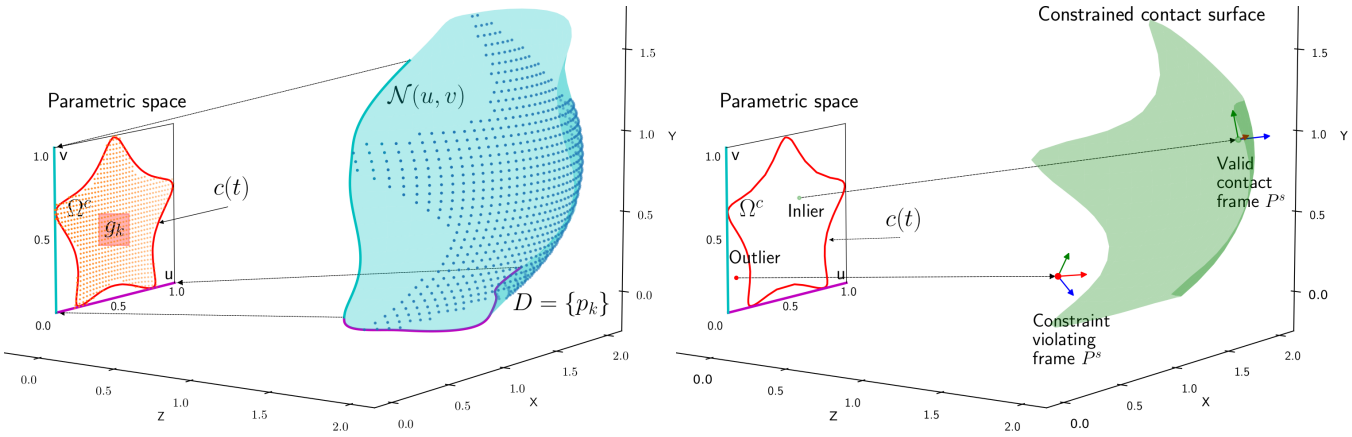

Fig. 2: NURBS surface and curve fitting to point cloud (left), trimmed contact surface constraint representation (right).

to find such parameter values for u and v so that the resulting point $\mathcal{N}(u,v)$ on the surface is inside the constraining curve, while satisfying all the other constraints eq. 1b–eq. 1g.

The fitted surface and corresponding trimming curve can now be incorporated into the posture computation optimization formulation in the following manner. Once the surface is fitted, we add its parameters as additional decision variables u and v, and use fitted parametric surface control points P_{ij} to add the following constraints to the optimization problem. Let's assume only one contact for clarity purpose ($\mathcal{S} = \{P^e, \mathcal{N}\{P_{ij}\}, f^{\min}, f^{\max}\}$, where $P^e = \{p^e, R^e = (x^e, y^e, z^e)\}$ specifies robot end-effector frame to be used for contact on $\mathcal{N}$), we can write

$$\min_{q,u,v,f} \quad \phi(q, q_h, \mathcal{S}) \tag{4a}$$

$$\text{s. t.} \quad \text{eq. 1b–eq. 1f} \tag{4b}$$

$$0 \leq u, v \leq 1 \quad u, v \in \mathbb{R} \tag{4c}$$

$$\mathcal{N}(u,v) = \frac{\sum_i \sum_j N_{i,p}(u) N_{j,q}(v) P_{ij}}{\sum_i \sum_j N_{i,p}(u) N_{j,q}(v)} \tag{4d}$$

$$([u\ v] - c(t_{uv}))^T \cdot n_{t_{uv}} \leq 0 \tag{4e}$$

$$p^s = \mathcal{N}(u,v) \tag{4f}$$

$$du = \frac{\partial \mathcal{N}(u,v)}{\partial u} \qquad dv = \frac{\partial \mathcal{N}(u,v)}{\partial v} \tag{4g}$$

$$z^s = du \times dv \qquad y^s = z^s \times du \tag{4h}$$

$$R^s = \left(\frac{du}{|du|}, \frac{y^s}{|y^s|}, \frac{z^s}{|z^s|} \right) \tag{4i}$$

$$\overrightarrow{p_s p_e} \cdot R^s = (0,0,0) \tag{4j}$$

$$z^e \cdot R^s_{xy} = (0,0) \qquad z^e \cdot R^s_z \geq 0 \tag{4k}$$

$$f^{\min} \leq f \leq f^{\max} \tag{4l}$$

Constraints eqs. 4c–4f ensure that the contact point lies on the authorized surface area. Constraints eqs. 4g–4i compute the normal at point on the surface p^s. Constraints eq. 4j and eq. 4k align the robot's end effector with the surface in 3 translational directions and 2 orientation axes, the robot is free to rotate around the surface contact normal. These constraints are illustrated on a simple case shown in Fig. 2 (right); specifically eq. 4e is illustrated in detail in Fig. 3. Here, the variable $t_{uv} \in [0\ 1]$, from the curve parametric space, is a footpoint parameter of $[u\ v]$ point, computed on previous optimization iteration, such that $c(t_{uv})$ is the closest point on the curve to $[u\ v]$ and $n_{t_{uv}}$ is the curve's normal at this point. Finally, eq. 4l, ensures that forces, applied by the robot at a contact point, satisfy predefined boundaries.

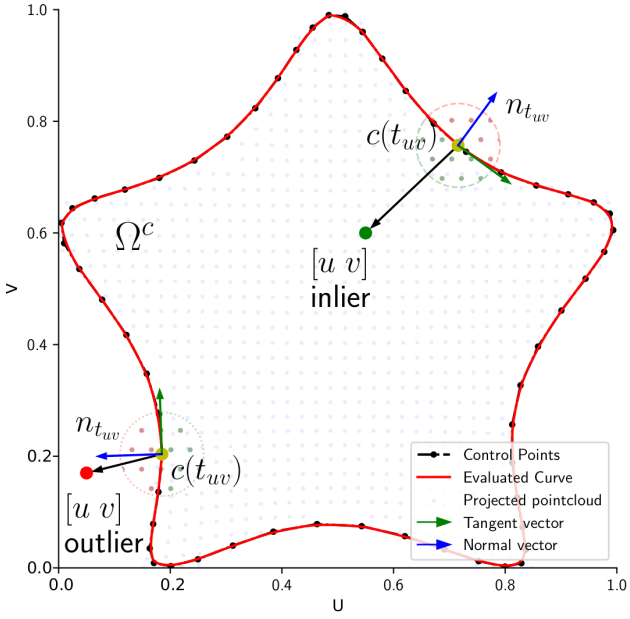

Fig. 3: Curve enclosure constraint on UV space.

The solution to eq. 4 is a whole-body robot posture and contact location with force plan, which allows the robot to make contact with the point cloud approximated by trimmed NURBS surface while satisfying joint and torque limits, maintaining balance and avoid collisions.

In the following subsection, we detail the process of point cloud segmentation that supplies the input D for a proposed PG framework. Then, in the next section, we present several examples of pHRI task plans computed using our proposed method and discuss its advantages and limitations.

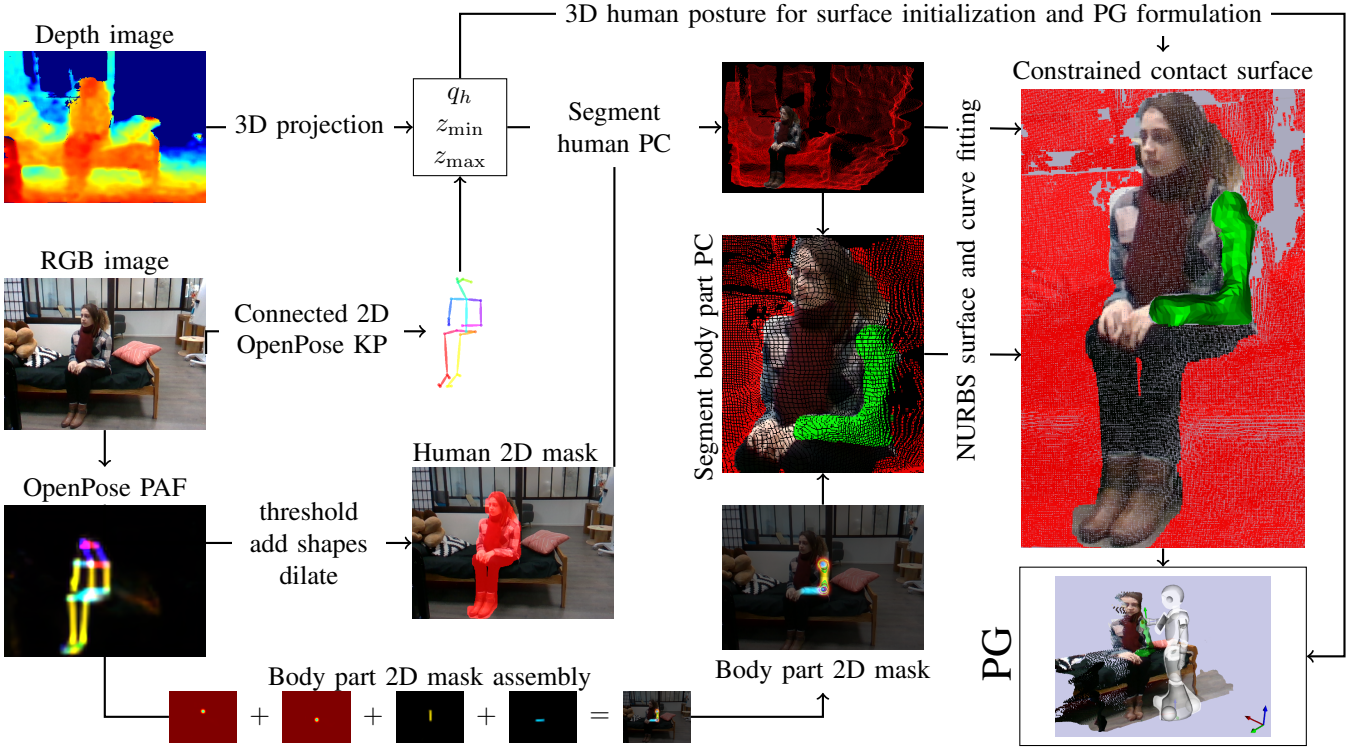

Fig. 4: Point cloud processing pipeline to supply input for the posture optimization problem formulation in pHRI.

B. *Processing of human point cloud*

Our proposed posture planning methodology can be used for specific human assistance pHRI tasks, as long as, the point cloud of interest (D) (e.g. human shoulder, human back) is properly segmented out from an entire scene observed by the robot embedded RGBD camera. In this subsection, we describe the RGBD data processing and point cloud segmentation pipeline that supplies input to our PG. The overview of the entire pipeline is shown in Fig. 4, in the following we describe its main components.

First, an RGB image is processed by two-branch multi-stage convolutional neural network (CNN), a framework known as OpenPose [13]. This CNN predicts confidence maps (CM) for 25 main human body keypoints (KP), by assigning the likelihood of the presence of a particular human body part to every image pixel. Simultaneously, OpenPose predicts, so called, Part Affinity Fields (PAF), which encode the location and orientation of human body parts in the image. Those two types of CNN output are jointly processed to compute connected 2D human skeleton. In our work, we also use the output of the CNN to compute 2D masks for point cloud segmentation. By combining PAFs of all body parts, we obtain a 'heat' map, which indicates which pixels of 2D image are more likely to belong to human body picture. We threshold combined PAF image to consider only high likelihood pixels ($\geq \sigma = 40\%$), which are assigned likelihood 100% after thresholding. The pixels with likelihood below σ, are assigned 0%, yielding a black and white image mask. We use 2D torso and head KP to set human body part likelihood of all pixels within torso polygon and head ellipse to 100%. We dilate the resulting mask to remove small holes and expand the borders. Similar process is performed to create specific body part 2D mask, by considering CM and PAFs of requested pHRI task relevant body parts.

Secondly, a depth image and camera intrinsic parameters are used to compute 3D point cloud of the entire scene, and to get 3D locations of human body's KP detected on 2D image. As a result, we obtain an approximate human body posture in 3D space q_h as well as depth range for human point cloud segmentation, Z axis values of the closest/farthest 3D KP $z_{\min}$ and $z_{\max}$. With those values at hand, and with previously computed human 2D mask, we can successfully segment only those points from the entire scene point cloud that are likely to belong to human body. The points in the human subcloud are further processed by applying task related body part 2D mask to segment the point cloud D that will contain human body part suitable for contact.

Finally, with D segmented out, we perform the NURBS surface $\mathcal{N}(u, v)$ and constraining curve $c(t)$ fitting, as described in details in the previous subsection, using NURBS algorithms implemented in Point Cloud Library [11], [14]. Previously computed nominal human 3D posture q_h can be used, at this point, to compute the initial control points values P_{ij}^{init} and P_i^{init} for the surface and curve fitting process. Once the trimmed surface is computed, it is passed to the PG to build the optimization problem eq. 4. The points in the human (and environment) subcloud, that do not participate in $\mathcal{N}$ fitting, must be used to build strict collision avoidance constraints in PG, for instance by fitting basic minimum volume convex shape to these parts of the point cloud [15].

III. Experimental Results

We present three HRI scenarios in the context of human care and assistance. We demonstrate the resulting whole-body postures computed using proposed method.

A. Attracting human's attention

We consider a situation where robot is required to attract human attention by lightly touching the human's left arm, as a human carer would do. A human is observed by the robot RGBD camera sitting on the bed in the environment. The goal is to plan whole-body posture, including the position of a mobile base in the environment, that would allow robot to make the contact while satisfying all the constraints imposed by the environment and the robot structure itself. The left arm surface is computed as described in the previous section and is passed to PG to build the optimization problem. The solution of PG is demonstrated with annotations of all tasks and constraints in Fig. 1. Another view of the same scene, that better illustrates result of contact point location planning on a trimmed NURBS surface, is shown in Fig. 5.

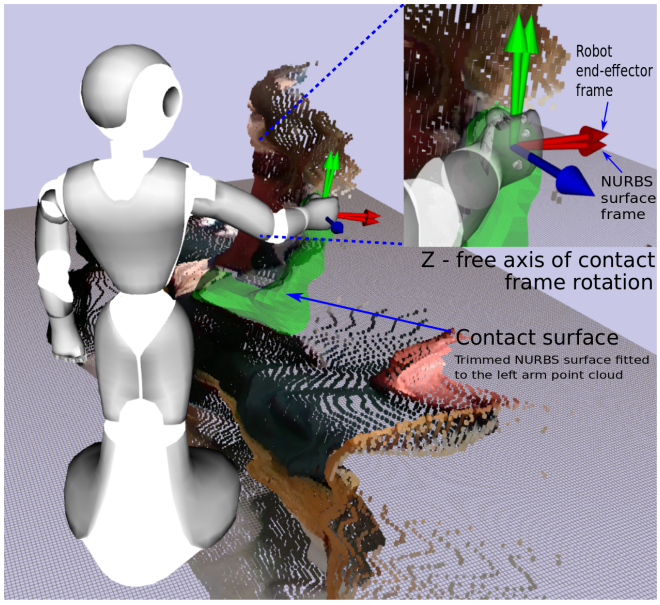

Fig. 5: Single contact with a trimmed NURBS surface.

B. Initiating assistance for sit to stand transfer

The second scenario consists in initiating a process of assistance for sit to stand transfer. Note that a suitable strategy for an assistance in such scenario may vary from patient to patient. In our scenario, we assume that a suitable strategy is to initiate two contacts. The first contact is closer to the patient's shoulder, which is intended to allow robot to apply pushing force forward and upward. The second contact is closer to the hand of the patient, which would allow to control human's forward movement by resisting force applied on the robot end-effector by the patient during the sit to stand transfer. The same trimmed NURBS surface is used to formulate both PG contact constraints. However, different initial points in the surface parametric space are used for

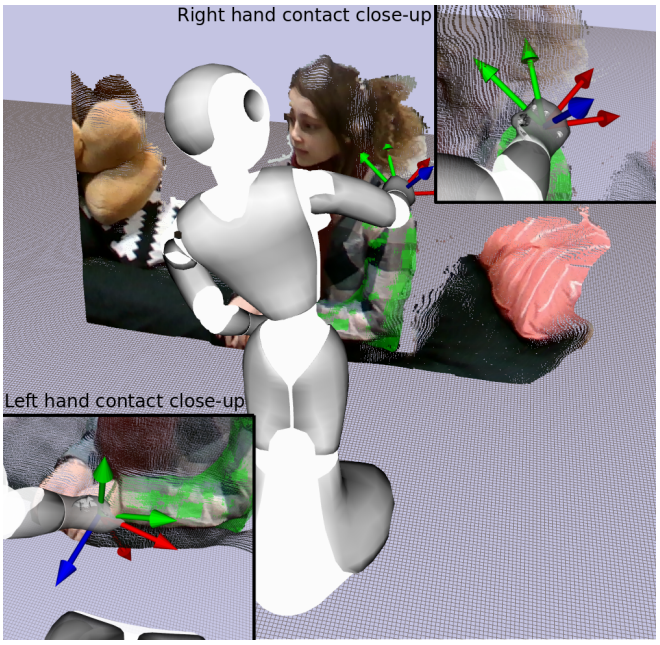

Fig. 6: Two contacts with a trimmed NURBS surface.

decision variables initial values, one closer to the shoulder and other closer to the hand, to facilitate PG convergence to the desirable assistance posture shown in Fig. 6.

C. Checking for responsiveness

In our last presented scenario, robot is required to check if a person lying on a bed is responsive. The robot observes the human in the environment and selects a human link suitable and appropriate for contact, in this case the human's right arm. The selected link's point cloud is segmented out and used for surface and curve fitting. The goal is then to plan a whole-body posture that would allow robot to lightly touch human's right arm surface while respecting all other PG constraints. This case can also be regarded as "waking-up" HRI scenario. The resulting whole-body posture and optimal contact location are shown in Fig. 7.

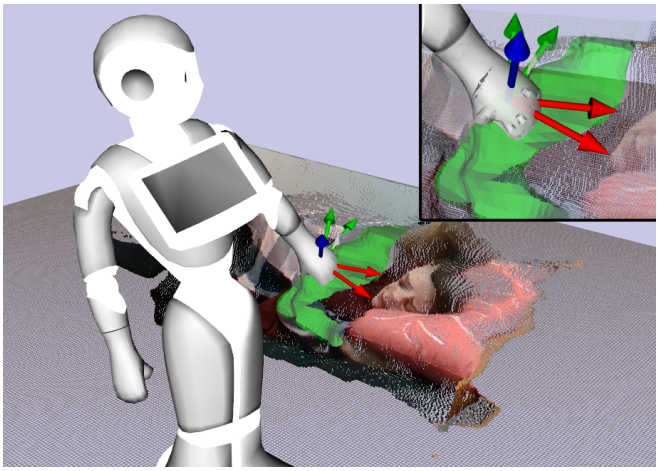

Fig. 7: Contact between robot left arm and human right arm.

D. Discussion and future work

Besides the task specific contact constraints, each scenario includes (self-)collision avoidance, joint and torque limits, stability, friction cone and end-effector force limit constraints. Additionally, a low weight posture task is added to the objective function of PG to indicate user preference in robot overall posture and, for example, gaze direction (looking at a human or end-effector contact location) which is an important aspect in HRI. The average time and number of iteration for convergence is presented in Tab. I. All computations are performed on the GeForce GTX 1050 Ti GPU.

Avg time PC processing	Avg time PG	Avg PG iterations
5.75s	7.14s	138

TABLE I: Average point cloud (PC) processing time, PG solver computation time and the number of solver iterations.

In our future work, the proposed method will be optimized to be used in online (re)planning of the robot motion in HRI scenarios. Following research phases also include using the output of proposed PG to control the robot in experiments where robot engages in physical contact with people for the purpose of assistance in motion [16]. The user/patient reaction to such interaction will be analyzed and accounted for in our following steps of development of motion planning methods and a 'robotiquette' for HRI in close proximity [17].

Lastly, the proposed posture computation method will be incorporated into the motion synthesis framework to compute full robot trajectories taking into account the type of motion that human and robot must undergo while maintaining or switching contacts [18], [19]. This must be done in order to guarantee that the postures outputted by the planner are indeed suitable for a particular known a priori assistance in motion task all along the predefined motion path.

IV. Conclusion

We have presented a novel approach for planning contacts on human point cloud. In our framework, a human skeleton is first detected on a 2D image. The skeleton detection results are used to segment out a human point cloud from the entire scene observed by RGBD camera. The segmented human point cloud is further processed to segment a task specific human link point cloud. The link point cloud is used to fit a NURBS surface and a constraining curve. The surface and the curve are used to formulate geometric contact constraints in a nonlinear optimization PG framework for HRI scenarios.

The method and implementation details have been throughly described. We have presented several scenarios of HRI and the results of posture generation computed using proposed method. the results demonstrate robot postures and optimal contact locations as well as interaction forces, determined autonomously, that result in a safe and feasible HRI while satisfying collision avoidance, joint/torque/force limits, friction cone and stability constraints.

V. Acknowledgment

Authors thank ProtoLab Team members at SoftBank Robotics Europe for help in collecting data for this project. We also thank Michel Besombes and Alban Laflaquière for their useful suggested edits. Last but not least, authors thank Adrien Escande for introducing the PG framework to us and Pierre Gergondet for his help with the code of PG.

References

[1] "World population ageing," report ST/ESA/SER.A/390, United Nations, Department of Economic and Social Affairs, Population Division, 2015.

[2] M. Niemelä and H. Melkas, "Robots as social and physical assistants in elderly care," in *Human-Centered Digitalization and Services*, vol. 19, pp. 177–197, 2019.

[3] M. Onishi, Z. Luo, T. Odashima, S. Hirano, K. Tahara, and T. Mukai, "Generation of human care behaviors by human-interactive robot riman," in *IEEE International Conference on Robotics and Automation*, pp. 3128–3129, 2007.

[4] T. Mukai, S. Hirano, H. Nakashima, Y. Kato, Y. Sakaida, S. Guo, and S. Hosoe, "Development of a nursing-care assistant robot riba that can lift a human in its arms," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5996–6001, 2010.

[5] M. Staffa and S. Rossi, "Recommender interfaces: the more human-like, the more humans like," in *International Conference on Social Robotics*, pp. 200–210, 2016.

[6] 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.

[7] E. Torta, F. Werner, D. O. Johnson, J. F. Juola, R. H. Cuijpers, M. Bazzani, J. Oberzaucher, J. Lemberger, H. Lewy, and J. Bregman, "Evaluation of a small socially-assistive humanoid robot in intelligent homes for the care of the elderly," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 1, pp. 57–71, 2014.

[8] D. Li, P. P. Rau, and Y. Li, "A cross-cultural study: Effect of robot appearance and task," *International Journal of Social Robotics*, vol. 2, no. 2, pp. 175–186, 2010.

[9] S. Brossette, A. Escande, and A. Kheddar, "Multicontact postures computation on manifolds," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1252–1265, 2018.

[10] O. Khatib, K. Yokoi, O. Brock, K. Chang, and A. Casal, "Robots in human environments: Basic autonomous capabilities," *The International Journal of Robotics Research*, vol. 18, no. 7, pp. 684–696, 1999.

[11] L. Piegl and W. Tiller, *The NURBS book*. 2012.

[12] S. Flöry and M. Hofer, "Constrained curve fitting on manifolds," *Computer-Aided Design*, vol. 40, no. 1, pp. 25–34, 2008.

[13] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7291–7299, 2017.

[14] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *IEEE International Conference on Robotics and Automation*, pp. 1–4, 2011.

[15] A. Escande, S. Miossec, M. Benallegue, and A. Kheddar, "A strictly convex hull for computing proximity distances with continuous gradients," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 666–678, 2014.

[16] A. M. López, J. Vaillant, F. Keith, P. Fraisse, and A. Kheddar, "Compliant control of a humanoid robot helping a person stand up from a seated position," in *2014 IEEE-RAS International Conference on Humanoid Robots*, pp. 817–822, 2014.

[17] K. Dautenhahn, "Socially intelligent robots: dimensions of human–robot interaction," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 362, no. 1480, pp. 679–704, 2007.

[18] D. Rakita, B. Mutlu, and M. Gleicher, "Stampede: A discrete-optimization method for solving pathwise-inverse kinematics," in *IEEE International Conference on Robotics and Automation (ICRA). IEEE*, 2019.

[19] A. Escande, A. Kheddar, and S. Miossec, "Planning contact points for humanoid robots," *Robotics and Autonomous Systems*, vol. 61, no. 5, pp. 428–442, 2013.