

Forward Dynamics Simulation of Human Figures on Assistive Devices using Geometric Skin Deformation Model

Yusuke Yoshiyasu¹ Ko Ayusawa¹ Eiichi Yoshida¹ Yoshio Matsumoto² Yui Endo¹

Abstract—We present a forward dynamics (FD) simulation technique for human figures when they are supported by assistive devices. By incorporating a geometric skin deformation model, called linear blend skinning (skinning), into rigid-body skeleton dynamics, we can model a time-varying geometry of body surface plausibly and efficiently. Based on the skinning model, we also derive a Jacobian (a linear mapping) that maps contact forces exerted on the skin to joint torques, which is the main technical contribution of this paper. This algorithm allows us to efficiently simulate dynamics of human body that interacts with assistive devices. Experimental results showed that the proposed approach can generate plausible motions and can estimate pressure distribution that is roughly comparable to the tactile sensor data.

I. INTRODUCTION

Recently, assistive devices have been receiving attentions and developed extensively in aging societies like Japan, in order to reduce the burden of caregivers, elderly and patients. However, the current development process for such devices is trial-and-error (i.e., iterations of designing, building prototypes and testing) and therefore takes a long time. To make this process more efficient, it is important to predict their usability and safety in advance. In this paper, we attempt to devise a simulation method for evaluating assistive devices, where the device is in contact with the patient's body. We extend the previous forward dynamics (FD) simulation techniques for human figures developed in biomechanics, robotics and computer graphics such that interactions between human body and assistive devices can be simulated. The following is the main properties that the simulation techniques for assistive devices should possess:

- Efficiency. The ultimate goal of our simulation technique is to provide feedbacks to design process. Thus, efficiency is important to test various candidates of device designs.
- Realism. Simulated motions should resemble human motions on a large scale. However, we believe that a small scale resemblance such as reproductions of secondary deformations, e.g., jiggling of belly, is not required for our purpose.

*This work was partly supported by JSPS KAKENHI grant number 25880033 and by METI project to promote the development and introduction of robotic devices for nursing care.

¹Yusuke Yoshiyasu, Ko Ayusawa, Eiichi Yoshida and Yui Endo are with Interactive Robotics Research Group and CNRS-AIST JRL (Joint Robotics Laboratory) UMI3218/RL, National Institute of Advanced Industrial Science and Technology (AIST), Japan. {yusuke-yoshiyasu, k.ayusawa, e.yoshida, y.endo}@aist.go.jp

²Yoshio Matsumoto is with Service Robotics Research Group, AIST, Japan. yoshio.matsumoto@aist.go.jp

Various simulation techniques have been proposed in robotics, biomechanics and computer graphics (CG). Previous FD simulation techniques in robotics and biomechanics focus mainly on generating active motions of articulated models, such as gaits, using rigid-body models [1], [9]. For the simulation of assistive devices that support the patient's body, the situation is slightly different in that the body moves almost passively according to device motions. The major challenge in devising such a simulation technique is the modeling of realistic human-device interactions. Contacts of articulated models and clothes have been well-studied in CG. Nevertheless, in many cases, articulated human models are animated by animators and only dynamics of clothes are considered. In the automotive industry, finite element methods (FEM) are used to simulate collisions of human body and a car during car crash. While this produces accurate results, this is not suitable for our aim i.e., testing a wide variety of design candidates and providing feedbacks for design process, because FEM is computationally expensive.

In this paper, we propose a forward dynamics simulation technique that can model human-device interactions efficiently in order to evaluate assistive devices. To this end, we incorporate a geometric skin deformation model for character animation, called linear blend skinning (skinning in short), into a rigid-body articulated dynamics model. The skinning method models skin deformations due to skeletal transformations in a plausible manner. Thus, we do not model deformation of inner structures of the body, such as soft tissues, muscles and organs. Instead, we develop a simple algorithm to transfer the contact forces to joints such that configurations of articulated models can be altered by such forces. This allows us to efficiently simulate dynamics of human body that interacts with assistive devices. The assistive devices that we tested in this paper are two types: a robotic and sling lift. The robotic lift supports transfer from a chair and consists of rigid-body parts. The sling lift supports transfer from a bed, which uses a deformable sheet that we model with a mass-spring model.

II. FORWARD DYNAMICS OF HUMAN-DEVICE INTERACTION

Figure 1 illustrates the steps that the proposed simulation technique go through during human-device interactions. Our model consists of three components: human body, device and contact models. The proposed human body model has a skeleton inside body and a skin surrounds the body. The pose of skeleton is computed by a multi joint rigid-body dynamics model developed in robotics. The resulting pose is

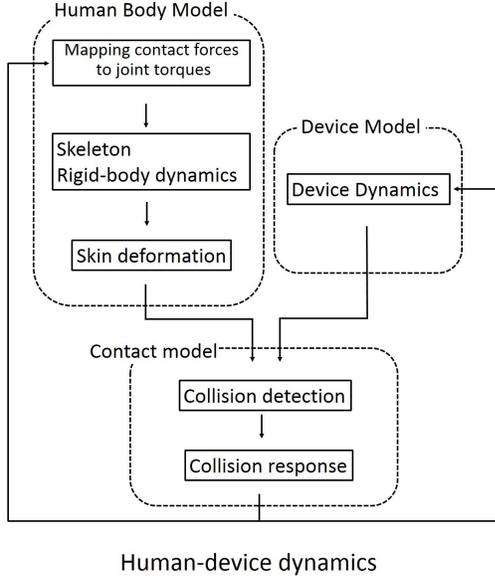


Fig. 1. Flow of forward dynamics during human-device interaction

then passed to the skin deformation model called skinning to produce a deformed skin surface geometrically without dynamics. Contact forces exerted on skin are directly mapped to the joints to produce joint torques. We do not include the dynamics of inner structures like fat, organs, muscles etc. For the device models, we tested two types of lifts: a robotic lift that supports transfer from a chair and a sling lift that supports transfer from a bed.

III. HUMAN BODY MODEL

A. Skeleton model

The skeleton is modeled as an open-loop tree structure with the root joint at the hip. We describe a pose of the skeleton using the generalized coordinates \mathbf{q} , which includes joint angles of the skeleton, the absolute position and orientation of the root joint. The equation of motion of the skeleton is then written as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\dot{\mathbf{q}}, \mathbf{q}) + g(\mathbf{q}) = \boldsymbol{\tau}^{\text{joint}} + \boldsymbol{\tau}^{\text{ext}} \quad (1)$$

where $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are joint velocity and acceleration, respectively. The quantities $\mathbf{M}(\mathbf{q})$, $\mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})$ and $g(\mathbf{q})$ are a inertia matrix, Coriolis/centrifugal and gravitational forces, respectively. The first term of the right hand side, $\boldsymbol{\tau}^{\text{joint}}$, is the joint torque and the second term $\boldsymbol{\tau}^{\text{ext}}$ is the moment due to contact. Equation (1) is solved and integrated using the algorithm proposed in [9] that uses the explicit 4-th order Runge-Kutta integration method.

To mimic joint torques due to the stiffness of joints and muscle contractions for keeping a posture, we use proportional-derivative (PD) controls using the initial pose as the reference pose. The joint torque generated by this control for joint p is:

$$\boldsymbol{\tau}_p^{\text{joint}} = k_p^{\text{P}}(\theta_p^{\text{ref}} - \theta_p) + k_p^{\text{D}}(\dot{\theta}_p) \quad (2)$$

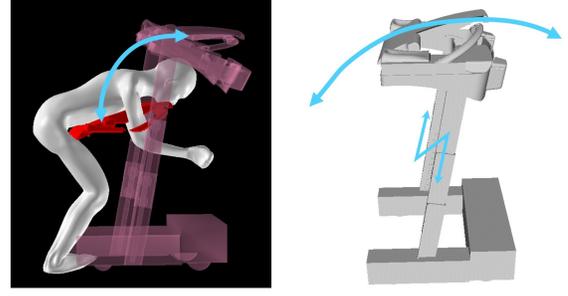


Fig. 2. Modeling of a robotic lift

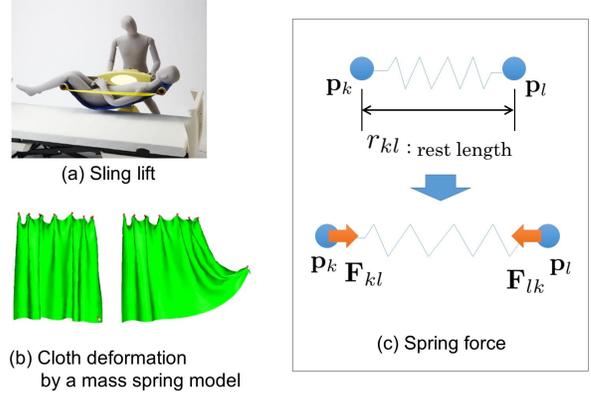


Fig. 3. Modeling of a sling lift

where k_p^{P} and k_p^{D} are PD spring constants and θ_p^{ref} is the reference joint angle.

B. Skin model

The skin surface is modeled as a triangle mesh that contains n vertices whose positions are described as a $n \times 3$ matrix, $\mathbf{v} = [\mathbf{v}_1 \dots \mathbf{v}_n]^T$. We employ linear blend skinning (cf. [5]), which is a popular geometric deformation method in character animation, in order to model skin deformations due to pose changes. To apply skinning, we convert pose \mathbf{q} into a matrix containing 3×3 absolute rotation matrices, $\mathbf{R} = [\mathbf{R}_1 \dots \mathbf{R}_m]^T$, and the one containing translations $\mathbf{t} = [\mathbf{t}_1 \dots \mathbf{t}_m]^T$. Given these transformations, skinning deforms the vertex in the rest state, \mathbf{v}_i^0 , and obtains the vertex in the deformed state, \mathbf{v}_i by:

$$\mathbf{v}_i = \sum_j \alpha_{ij} [\mathbf{R}_j(\mathbf{v}_i^0 - \mathbf{x}_j^0) + \mathbf{x}_j^0 + \mathbf{t}_j] \quad (3)$$

where \mathbf{x}_j^0 is the position of joint j in the rest pose. Here, α_{ij} is a weight that determines the contribution of bone j to vertex i . Weight α_{ij} can be automatically computed by using software such as Pinocchio [2].

IV. ASSISTIVE DEVICE

We modeled two types of assistive devices: a robotic lift (Fig. 2) and a sling lift (Fig. 3). The geometry of the device

is denoted as a triangle mesh that contains N vertices whose positions are described as an $N \times 3$ matrix, $\mathbf{p} = [\mathbf{p}_1 \dots \mathbf{p}_N]^T$. **Robotic lift** The robotic lift consists of rigid links and their movements are controlled by motors that generates translational and rotational movements of the links. As illustrated in Fig. 2, the device tested in this paper has 2 Dof, a single translation and rotation. In this paper, we do not model actuators and we directly apply translations and rotations to the link.

Sling lift The sling sheet is modeled using a mass-spring model proposed by Liu et al. [6]. The mass spring system comprises point masses defined at vertices and springs defined at edges that connect point masses. The equation of motion for this system is:

$$\mathbf{M}\ddot{\mathbf{p}} + \frac{\partial E}{\partial \mathbf{p}} + g(\mathbf{p}) = \mathbf{f} \quad (4)$$

where \mathbf{M} is a diagonal mass matrix, E is the internal energy, $g(\mathbf{p})$ is the gravitational force and \mathbf{f} is the compound of contact and friction forces. The force generated at vertex k can be computed from the partial derivative of the internal energy as $\frac{\partial E}{\partial \mathbf{p}_k} = \sum_l \mathbf{F}_{kl}$, where \mathbf{F}_{kl} denotes the force acting at the edge between k and l . The spring is modeled as a Hookean spring and \mathbf{F}_{kl} is computed as:

$$\mathbf{F}_{kl} = k^{\text{sheet}} (\|\mathbf{p}_k - \mathbf{p}_l\| - r_{kl}) \frac{\mathbf{p}_l - \mathbf{p}_k}{\|\mathbf{p}_k - \mathbf{p}_l\|} \quad (5)$$

where k^{sheet} is the spring constant for a deformable sheet. This method employs a variant of the implicit backward Euler integration.

V. CONTACT AND FRICTION

A. Contact force

To compute contact forces, we use the method similar to the one proposed by Guan et al. [3]. We first find a vertex on the skin that is closest to vertex k of the assistive device, which we denote its index as $\text{idx}(k)$. Let $\mathbf{n}_{\text{idx}(k)}$ be the surface normal of the skin at vertex $\text{idx}(k)$. The penetration depth is computed by: $d_k = \mathbf{n}_{\text{idx}(k)} \cdot (\mathbf{v}_{\text{idx}(k)} - \mathbf{p}_k)$. If the penetration depth is $d_k > 0$, then we apply a contact force to vertex $\text{idx}(k)$:

$$\mathbf{f}_{\text{idx}(k)}^{\text{contact}} = k^{\text{contact}} \cdot d_k \cdot \mathbf{n}_{\text{idx}(k)} \quad (6)$$

Here k^{contact} is the spring constant where we used the same value for all springs.

B. Friction

We employ the friction model proposed by Martins and Oden [8]. This model approximates Coulomb frictions with viscosity frictions when the relative velocity is around zero. Let $\mathbf{V}_{\text{idx}(k)} = \dot{\mathbf{v}}_{\text{idx}(k)} - \dot{\mathbf{p}}_k$ be the relative velocity of the skin with respect to the device. The magnitude of friction force is thus computed as:

$$\mathbf{f}_{\text{idx}(k)}^{\text{fric}} = \begin{cases} -F_c \cdot \mathbf{V}_{\text{idx}(k)} / \epsilon & \text{if } \|\mathbf{V}_{\text{idx}(k)}\| \leq \epsilon \\ -F_c \cdot \mathbf{V}_{\text{idx}(k)} / \|\mathbf{V}_{\text{idx}(k)}\| & \text{otherwise} \end{cases} \quad (7)$$

where F_c and ϵ are the Coulomb force and the velocity threshold, respectively.

C. Calculating joint moments from contact forces

In order to map contact forces from skin to joints, we use skinning equation Eq. (3). Since a skinning equation relates transformations of joints with skin vertices, we can calculate joint moments from external forces by inverse mapping. The velocity of the skin vertex can be derived as:

$$\dot{\mathbf{v}}_i = \sum_j \alpha_{ij} (\dot{\mathbf{x}}_j - [\mathbf{R}_j (\mathbf{v}_i^0 - \mathbf{x}_j^0) \times \boldsymbol{\omega}_j]) \quad (8)$$

where $\dot{\mathbf{x}}_j$ $\boldsymbol{\omega}_j$ are the end-effector velocity. This can be written in the matrix form as:

$$\dot{\mathbf{v}} = \mathbf{J}^S \begin{pmatrix} \dot{\mathbf{x}} \\ \boldsymbol{\omega} \end{pmatrix} \quad (9)$$

The end-effector velocities and the joint angle velocities can be related with the basic Jacobian \mathbf{J}^B [4]:

$$\begin{pmatrix} \dot{\mathbf{x}} \\ \boldsymbol{\omega} \end{pmatrix} = \mathbf{J}^B \dot{\mathbf{q}} \quad (10)$$

Consequently, we arrive at the relationship between the velocities of skin vertices and joint angles:

$$\dot{\mathbf{v}} = \mathbf{J} \dot{\mathbf{q}} \quad (11)$$

where $\mathbf{J} = \mathbf{J}_S \mathbf{J}_B$. Finally, according to the duality of differential kinematics and statics, the joint moments due to contact and friction forces exerted on skin vertices, $\mathbf{f} = \mathbf{f}^{\text{contact}} + \mathbf{f}^{\text{fric}}$, is computed as:

$$\boldsymbol{\tau}^{\text{ext}} = \mathbf{J}^T \mathbf{f} \quad (12)$$

Once joint moments are calculated as above, they are plugged in to the equation of motion Eq. (1) to update the pose of the skeletal model.

VI. RESULTS

A. Simulation model and parameters

We used the simplified version of the skeletal model proposed in [7], which has 47 degrees of freedom, $\mathbf{q} \in \mathbf{R}^{47}$, where the body segments are divided into hip (6 Dof), thigh (3 Dof), shank (1 Dof), foot (3 Dof), toe (1 Dof), lumber (3 Dof), thoracic (3 Dof), head (1 Dof), upper arm (3 Dof), forearm (3 Dof) and hand (3 Dof). The numbers of vertices are 14k, 20k and 4k for the skin, robotic lift and sling sheet, respectively. We set the mass properties of the skeleton model according to [7]. Other parameters for our model is summarized in Table I. We set the joint stiffness of the vertebrae high in order to prevent from collapsing. Because the sling sheet is not stretchable, we set its spring constant k^{sheet} to high value. The stiffness of the contact spring is set to the same value as k^{sheet} . Initial poses are given manually with forward kinematics and we start simulation after the pose gets close to an equilibrium state.

TABLE I
PARAMETERS USED FOR SIMULATION.

k_{shoulder}^P	1 [Nm/rad]	$\sum_k M(k, k)$	10
k_{thoracic}^P	100 [Nm/rad]	k_{sheet}	1000
k_{lumber}^P	100 [Nm/rad]	k_{contact}	1000
k_{hip}^P	1 [Nm/rad]	ϵ	0.1
k_{knee}^P	1 [Nm/rad]	F_c	0.5
k_{ankle}^P	1 [Nm/rad]		

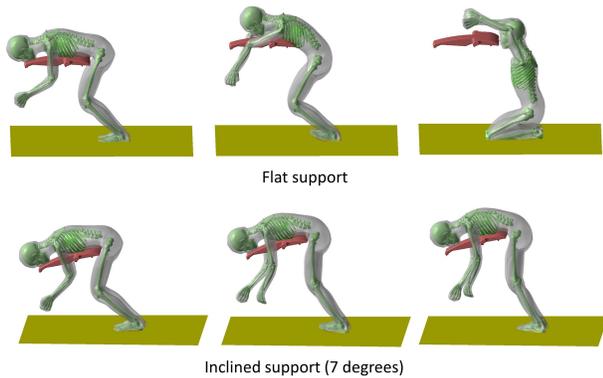


Fig. 4. Simulation result of robotic lift

B. Simulation results

The robotic device is tested for a relatively wide range of support angles. Figure 4 shows the two representative cases. For the result with a flat support, the body fell from the support. In contrast, the successful result was generated with the inclined (7 degrees) support.

We also tested the sling lift and generated the pose when the body is on the sling sheet. We estimated the pressure distribution of the back from the contact response. To evaluate the estimated result, we measured the pressure distribution with a tactile sensor. Note that for this experiment we selected the subject that has similar body type as the simulation model. Figure 5 (b) shows that the patterns of the estimated pressure distribution resembles with the measured data where they show highest pressure areas at the knees and slightly low pressures at the upper back and the hip.

C. Performance

The proposed simulation algorithm is implemented with C++ and Matlab on an Intel Core i7 3.4GHz 64-bit workstation. We used the C++ library for musculo-skeletal simulation developed in [7]. It takes approximately 5 min for the proposed technique to generate 1 sec of simulation, which is significantly faster than FEM.

VII. CONCLUSION

We presented a forward dynamics simulation technique to evaluate assistive devices. Our primary focus was to develop a efficient method so that we can generate simulation results for a variety of conditions. Preliminary experiments presented in this paper showed that the proposed simulation technique can generate plausible motions and pressure

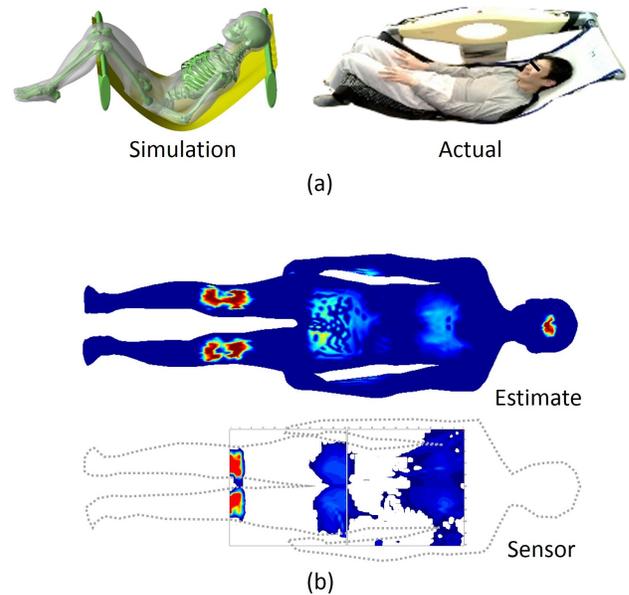


Fig. 5. Sling sheet simulation. (a) Overall pose. (b) Pressure distribution.

distributions, which implies the potential of our method to be used in evaluation of assistive devices. In future work, we will incorporate parameter identification techniques to estimate the physical parameters of body as well as devices. In addition, we will devise a way to help designing assistive devices by extracting meaningful feedbacks from simulation results.

ACKNOWLEDGMENTS

We thank Satoshi Iwai for helping us capture the tactile sensor data.

REFERENCES

- [1] "Opensim," <https://simtk.org>, accessed: 2015.
- [2] I. Baran and J. Popovic, "Automatic rigging and animation of 3D characters," *ACM Trans. Graph.*, vol. 26, no. 3, p. 72, 2007.
- [3] P. Guan, D. Reiss, L. and Hirshberg, A. Weiss, and M. J. Black, "Drape: Dressing any person," *ACM Trans. Graphics (Proc. SIGGRAPH)*, vol. 31, no. 4, jul 2012.
- [4] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 1, pp. 43–53, Feb. 1987.
- [5] J. Lewis, M. Cordner, and N. Fong, "Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-driven Deformation," *Proc. SIGGRAPH*, pp. 165–172, 2000.
- [6] T. Liu, A. W. Bargteil, J. F. O'Brien, and L. Kavan, "Fast simulation of mass-spring systems," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 214:1–214:7, Nov. 2013.
- [7] Y. Nakamura, K. Yamane, Y. Fujita, and I. Suzuki, "Somatosensory computation for man-machine interface from motion-capture data and musculoskeletal human model," *Robotics, IEEE Transactions on*, vol. 21, no. 1, pp. 58–66, Feb 2005.
- [8] J. M. J. Oden, "A numerical analysis of a class of problems in elastodynamics with friction," *Computer Methods in Applied Mechanics and Engineering*, vol. 40, no. 3, pp. 327–360, 1983.
- [9] K. Yamane and Y. Nakamura, "Efficient parallel dynamics computation of human figures," in *Proceedings of the 2002 IEEE International Conference on Robotics and Automation, ICRA 2002, May 11-15, 2002, Washington, DC, USA, 2002*, pp. 530–537.