Stable Simulation of Underactuated Compliant Hands

Alessio Rocchi[1,3], Barrett Ames[2], Zhi Li[3] and Kris Hauser[3]

***Abstract*— Despite increasing popularity of compliant and underactuated hands, few tools are available for modeling them. Thus, we propose a simulation technique to predict the success of a compliant gripper grasping irregular objects, which could be used in mechanism design as well as grasp planning. The simulator we propose integrates joint compliance simulation with a Boundary Layer Expanded Mesh (BLEM) technique to enhance the stability of contact estimation. We compare the proposed simulator with existing simulators via a set of stability and fidelity criteria, including *contact force variation*, *contact position variation*, and *contact normal variation*. Scores along these criteria are correlated with the simulator's accuracy of predicting the success/failure of a given grasp pose and preshape. A test set of 13 grasps, with two compliant underactuated hands were manually generated on 4 objects. Experiments suggest that our simulator leads to improvements in the stability criteria, predictability of grasp success, and reduction of simulation artifacts.**

Fig. 1: The experimental setup with the Baxter robot equipped with a Pisa/IIT Soft Hand (left) and Righthand Robotics Reflex Hand (right), shown with the set of grasped objects.

I. Introduction

Robot hand designers have been increasingly using compliance and underactuation in their designs, with recent examples being the iHY [23], Reflex Hand [25], Pisa/IIT SoftHand [7], Robotiq 3-Fingered Gripper [26], RBO [9] and RBO2 [10] and Yale Hands [21]. Underactuated hands provide many degrees of freedom of movement (DoFs) while being actuated by just a few degrees of actuation (DoAs), and clever passive mechanisms are used to adapt the hand shape to a wide range of grasped objects. This simplifies the otherwise complex hand control problem. Furthermore, compliant hands benefit from increased hardware robustness and reduced cost. Although compliant hands are more robust to sensing uncertainties than rigid ones, they introduce larger actuation uncertainties. As a result, successful grasping still requires deliberation, particularly for objects of irregular shape or mass distribution. It is also challenging to grasp objects in clutter because fingers may be disturbed by inadvertent contact with the environment or other objects. Hence, model-based predictions of motions and forces can offer powerful insights for hardware design and grasp planning.

Unfortunately, classical models for grasp analysis are poor at predicting the behavior of compliant hands. Classic kinetostatic analysis and kinematic simulation under the quasi-static assumption are inappropriate for studying the deformation of the hand under varying contact forces. Several researchers have turned to physics-based simulation analysis [17], [18], which improves grasp predictability by evaluating hundreds or thousands of trials, in which the simulator explores the range of nondeterministic effects, e.g., external perturbations, pose uncertainty, geometric uncertainty, and joint friction. Reliable simulation is essential for such studies. However, compliant hands pose a challenge for dynamic simulation tools, which often have trouble remaining stable in the presence of multiple frictional (and sometimes sliding or rolling) contacts and stiff internal mechanisms, like springs and tendons. When the object is not considered to be fixed to the environment, object-hand interactions need to be simulated taking dynamics into account, and a faithful representation of contact has to be available in order to detect the making/breaking of contacts, inadvertent contact, and sticking/slipping conditions.

This paper presents a new physics simulator for compliant hands interacting with rigid objects. It integrates models of compliant joints with recent contributions in robust mesh-mesh contact generation methods. Compliant elements are modeled as spring-dampers while stiffer elements like tendons are handled using constraints and Baumgarte stabilization. To handle mesh-mesh contact, we use the recent *Boundary Layer Expansion Mesh* (BLEM) method that generates continuously differentiable contact points and normals [15]. We explore several variations of BLEM against existing contact detection schemes in the Open Dynamics Engine [2] and Bullet [1] physics engines.

We establish a set of criteria for characterizing the stability of grasp simulations at the level of contact point and

*This work was partially supported by ERC Advanced Grant no. 291166 SoftHands

[1]Alessio Rocchi is with the Department of Advanced Robotics, Istituto Italiano di Tecnologia, Via Morego 30, 16163, Genova, Italy alessio.rocchi@iit.it

[2]Barrett Ames is with the Department of Computer Science, Duke University, Durham, NC 27708, USA cbames@cs.duke.edu

[3]Kris Hauser, Zhi Li and Alessio Rocchi are with the Department of Electrical & Computer Engineering, Duke University, Durham, NC 27708, USA kris.hauser@duke.edu, zhi.li2@duke.edu

force prediction. These criteria are validated via a series of experiments comparing grasps that are performed on the physical hardware against results from the simulation tools. We find that grasp predictability is found to be correlated with the contact stability criteria. Lastly, the new simulator is demonstrated to achieve improved fidelity regarding the predictability of grasp success compared to existing off-the-shelf simulators.

II. Related Work

Robot Simulators. Simulation techniques for modern robot hardware could provide invaluable tools for design, research, and development for robot controllers. The DARPA Robotics Challenge (DRC), for example, fostered significant investment in reliable simulation tools for humanoid robots [16], which allowed teams to compete virtually before qualifying for expensive robot hardware. Gazebo [19] is one of the most popular general-purpose robot simulators, funded through DRC efforts, but there are several others including V-REP and Webots. A few robot simulators and toolkits are specialized in grasping, such as GraspIt! [22], OpenRave [11] and OpenGRASP [20], which have built-in functionality for grasp analysis. However, these prior methods assume fully and precisely actuated grippers.

In [6] a dynamic simulation of the Pisa/IIT Softhand is implemented in the multi-body dynamics simulator MSC Adams [8]. A method to generate pre-grasp palm configurations w.r.t. the object pose is provided. The simulator demonstrated moderate fidelity to an experimental scenario, but the authors acknowledge difficulties with hand-object penetrations and estimation of contact normals.

A recent contribution in the Klamp't simulator is the notion of boundary-layer expanded meshes (BLEM) which were demonstrated to enable robust mesh-to-mesh contacts, even in the presence of non-watertight meshes [15]. In this work we extend Klamp't to handle compliant mechanisms, and also we adapted the BLEM technique into Gazebo as a plugin.

Grasp analysis and planning. Classical grasp analysis typically studies the shape of the grasp wrench space (GWS), which is the convex hull of wrenches applicable by a unit force at each contact (e.g., [13], [24]). But recent studies have suggested weaknesses of classical methods, such as an inability to measure robustness to perturbations in contact locations or grasp / object pose [29]. Physics simulation in grasp quality assessment has been shown to yield improved prediction over classical criteria [17], [18]. In [18] the robustness of automatically generated grasp sets is assessed, and success of grasps in physical experiments is correlated with the predicted success using simulation. In [17] grasp stability criteria are correlated with grasp success as predicted by humans on a large database of objects. They concludes that physics-based metrics are more consistently predictive than a classical metric. The technique presented in this paper extends physics-based grasp assessment to the case of underactuated and compliant hands.

Simulation quality metrics. Generic criteria for evaluating the contact response fidelity of dynamic physics engines were established in [5], such as accuracy of the friction coefficient, and penetration error. [12] defines some application specific metrics, like a grasping metric, in which robustness to the contact handling is tested by increasing the integration step until the simulated grasp becomes unstable. In this work we are interested in relating micro-criteria of contact stability in the physics to the macroscopic fidelity of the simulator to the real world.

III. Stable Dynamic Grasp Simulation

A. Summary

An underactuated hand is modeled as n articulated rigid links with n_a degrees of actuation, with $n_a < n$. The state of the fingers is denoted as $q \in \mathbb{R}^n$. A control $u \in \mathbb{R}^{n_a}$ gives rise to a net torque on the joints $\tau \equiv \tau(q,u) \in \mathbb{R}^n$ which summarizes the sum of internal torques including gearing, stiffness, damping, joint stops, and friction.

For the three-fingered ReFlex, $n_a = 4$, with 1 DoA for each finger. An additional pregrasp mechanism changes the direction in which finger 2 and 3 close from power grasp to pinch grasp. Assuming the distal joints rotate along a fixed axis, $n = 7$, but in general the soft joint between the proximal and distal joints may flex and stretch. For the five-fingered SoftHand, $n_a = 1$, $n = 19$.

When in contact with external objects, the pressure distribution on a robot's link is a function $\rho_i(x) : \partial S_i(q) \to \mathbb{R}^3$ where $\partial S_i(q)$ denotes the surface of link i. Given such a distribution, the resultant torques on the robot's joints are

$$\tau_{contact} = \sum_i \int_{x \in S_i(q)} J_x^T(q) \rho_i(x) dx \qquad (1)$$

where $J_x(q)$ is the Jacobian matrix of point x.

Thus, the dynamics of the robot in contact are given by

$$B(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau(q,u) + \tau_{contact} + \tau_{ext} \qquad (2)$$

where $B(q)$ is the robot's mass matrix, $C(q,\dot{q})$ is the Coriolis force matrix, $G(q)$ is the generalized gravity vector, and τ_{ext} denotes torques resulting from external forces, such as joint friction.

A simulator is given an initial state $(q_0, \dot{q}_0)$, a control trajectory $u(t)$, and a final time T. The goal is then to generate a trajectory of the robot $q(t) : [0,T] \to \mathbb{R}^n$ as well as the motions of other objects $O_1, \ldots, O_m$. The most challenging portions of simulating underactuated hands are 1) calibrating accurate models, particularly of τ, 2) calculating contact force distributions ρ to prevent interpenetration and to simulate friction, and 3) maintaining stability under often stiff dynamics, particularly in τ.

The linear complementarity problem (LCP) [3] method is a popular method for calculating contact force distributions in rigid body physics simulators because it solves for forces that prevent interpenetration (to some degree; slight interpenetration does occur due to numerical errors). All the simulators compared in this paper use the LCP method, although other

techniques such as penalty forces and impulse-based methods have also been developed.

B. Modeling underactuation and compliance

Underactuated and compliant hands are linked with transmission mechanisms, e.g., tendons or mechanical linkages, that distribute actuator effort across multiple joints. They also include spring mechanisms that restore the hand to a consistent rest state once gripping effort is removed. Simulations must allow for actuators to drive multiple links forward, but also to allow for forces on one link to affect the distribution of effort across other links. We use a fairly general model for these effects [14].

First, we use a general constraint model that relates actuator displacements s to configuration displacements q:

$$s = Rq \tag{3}$$

where the reference configuration is chosen so the zero actuator and joint correspond. The $n_A \times n$ matrix R determines how joint movements pull on each actuator, and can be constant, as in the case of the Pisa/IIT Softhand, or configuration dependent in the case of the ReFlex Hand [4]. In practice the tendons may be slightly elastic, but we assume they are sufficiently stiff so that direct simulation is unstable.

We assume the drive mechanism generates resultant torques on individual joints in order to maintain these constraints. Denote the force generated by the actuator as $\sigma \in \mathbb{R}^n$, and the torques generated by the drive mechanism be denoted $\tau_d \in \mathbb{R}^n$. By the principle of virtual work, we have $\tau_d = R^T\sigma$. Let us also define the joint torques produced by spring mechanisms as $\tau_s = -Eq$ where E is a $n \times n$ joint stiffness matrix (which is usually diagonal). Neglecting friction effects, the resultant vector of joint torques is

$$\tau = R^T\sigma - Eq. \tag{4}$$

Rather than simulate these torques directly, we derive a rest state of q that attains quasistatic equilibrium by equating τ with the sum of inertial effects minus contact forces in (2). First, we determine the desired value τ determined by the l.h.s. of (2) minus external forces:

$$\tau \equiv B(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) - \tau_{contact} - \tau_{ext} \tag{5}$$

We then solve for σ and q that satisfy the constraints (3) and (4) by solving a system of linear equations, which has a closed form solution in terms of s, R, E, and τ [14].

Given the solved q and σ, we then simulate joints using critically- or over-damped PID controllers with setpoint q. We may also simulate the effect of the actuator force σ on the actuator state s.

We note this procedure needs to iterate over multiple time steps to achieve equilibrium between mechanism torques and contact forces, which may cause chattering especially if contact forces are nonsmooth. Future work may achieve better results by simultaneously choosing mechanism torques and forces via a constraint in the LCP solver.

C. Contact generation

Contact generation is an important stage in simulation when the region of contact between two bodies is approximated by a finite number of contact points and normals. In LCP solvers, contact forces at these points are restricted to be positive in the normal direction with tangential component limited by the friction coefficient. Moreover, residual penetrations caused by numerical error are corrected for via penetration depth estimates computed in this stage.

It is difficult to compute a stable, accurate representation of the penetration region between nonconvex bodies, and hence existing engines ODE and Bullet use the approximate GIMPACT and OPCODE methods for calculating mesh-mesh contact points. The experience of many users as well as recent work suggested this method often leads to unstable simulations [15]. The same work presented the Boundary Layer Expanded Mesh (*BLEM*) representation that approximates a mesh with a slightly expanded version, which allows calculating accurate penetration distances and normals [15].

The *BLEM* method treats a mesh as having a small margin r around it, and when two BLEMs are overlapping by a distance less than the sum of their margins, the penetration depth and direction can be computed by distance queries on the underlying meshes. More precisely, a BLEM (M, r) is the Minkowski sum (or dilation)

$$M \oplus S = \{v + s | v \in M, s \in S\} \tag{6}$$

with $S = B(0, r)$ being a sphere centered in the origin, with radius r.

Contact generation between two BLEMs (M_1, r_1) and (M_2, r_2) is handled by detecting all pairs of primitive triangles in M_1 and M_2 whose distance d is less than $r_1 + r_2$. A point is then generated in the overlapping region between the two closest points p_1 and p_2 on the respective triangles. The penetration distance is $r_1 + r_2 - d$ and the normal is a unit vector proportional to $p_2 - p_1$.

D. Contact clustering

During close mesh-mesh contact, the contact generator may produce a large number of contact points, which leads to slow computation of contact response. This is a particular problem for LCP-based solvers due to their superlinear running time. As a result, researchers have taken an interest in contact clustering methods that combine multiple contacts into fewer contacts, which still yield similar post-contact response. Also, since having nearby contacts lead to ill-conditioned wrench matrices, clustering may in fact improve numerical stability.

Our experiments indicate that a deterministic version of the k-means algorithm, applied to the 7-D space of positions, normals, and friction coefficients, provides more stable clusters than hierarchical clustering, or the simple contact sorting used in Gazebo (Fig. 2).

IV. Contact stability metrics

Here we present the metrics used in this paper for evaluating the stability of contact points and contact forces at a fine

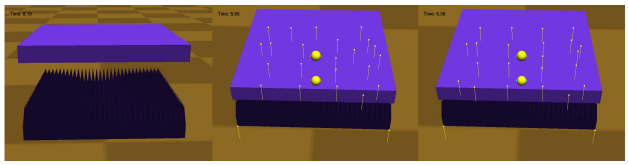

Fig. 2: Illustrating two contact clustering methods on a 2,000-spike "spiky block". Left: hierarchical clustering. Right: k-means clustering. Centers of pressure are draw

granularity during the simulation of grasping. These metrics correspond to the position, normal, and force dimensions of point contact models used in classical grasp analysis, which forms the basis of most grasp planning techniques.

A. Contact Force Variation

At equilibrium, we expect the contact forces exerted on the objects to be as close to constant as possible. Our first measurement penalizes fluctuations and spikes in force at an (ostensible) equilibrium state.

The resultant contact force exerted on body i is $f_i \equiv \int_{x \in S_i(q)} \rho_i(x) dx$, which the simulator approximates by a finite set of contact forces. The contact force smoothness index measures the normalized standard deviation of resultant forces. (It may be possible to also include resultant torques $m_i \equiv \int_{x \in S_i(q)} (x - o_i) \times \rho_i(x) dx$ but these are highly correlated to forces for most grasps so we ignore them.)

Over a span of time t where the object should be held in equilibrium by the physical hand, we measure the quantities $\sqrt{\|Var[f_i(t)]\|_F} / E[\|f_i(t)\|]$ for all bodies i in the set C of bodies touching the object. Here $\|\cdot\|_F$ denotes the matrix Frobenius norm. We then report the final score (0 being best):

$$S_{cf} \equiv \frac{1}{|C|} \sum_{|i \in C|} \sqrt{\|Var[f_i(t)]\|_F} / E[\|f_i(t)\|]. \quad (7)$$

B. Contact position and normal variation

The second and third scores measure the stability of contact positions and normals during an equilibrium grasp. A simulator generates a set of point contacts at location $x_1, \ldots, x_k$ and their corresponding normals $n_1, \ldots, n_k$ (k varying by time step). The scores measure the fluctuations of the average contact position $\hat{x}_i$ on body i and average contact normal $\hat{n}_i$ over time (0 being best).

The equations are as follows:

$$S_{cp} \equiv \frac{1}{C} \sum_{|i \in C|} \sqrt{\|Var[\hat{x}_i(t)]\|_F} \quad (8)$$

$$S_{cn} \equiv \frac{1}{C} \sum_{|i \in C|} \sqrt{\|Var[\hat{n}_i(t)]\|_F} \quad (9)$$

V. Experiments

Experiments are performed both with a RightHand Robotics' ReFlex Hand and Pisa/IIT SoftHand. A set of grasps with varying degrees of robustness are designed by human inspection and tested on a set of physical objects and their simulated counterparts.

Object	Weight	Size
Hammer	480 g	290mmX35mmX95mm
Spray Bottle (no cap)	703.5 g	220mmX120mmX40mm
Spray Bottle (with cap)	726 g	270mmX120mmX40mm
Ketchup	948.6 g	195mmX100mmX70mm
Pasta Box	487g	120mmX19mmX72mm

TABLE I: Weight of all the objects and approximate dimensions

For the two hands, a series of grasps are selected for each object in the set: *hammer, ketchup bottle, spray bottle* and a *pasta box*. The *hammer* and the *pasta box* are modeled by hand. The rest of the objects have been scanned using a *Makerbot Digitizer*. The results and details of the mesh reconstruction are shown and detailed in Fig. 3. Dimensions are given in Tab. I. We also assume an approximate mass distribution for the *hammer*, and for the mass of the the rest of the objects. In all the experiments, the objects are laying on a table.

Human-chosen grasps are selected in order to have a selection of robust grasps, non-robust grasps, and failed grasps as described in Tab. II. Robust grasps tolerate small variations in the object pose and disturbance forces. Non-robust grasps succeed some of the time, but do not always tolerate such variations. Failed grasps fail consistently.

The hands and the table are marked with fiducial markers, and their position and orientation are recorded for each grasp via a Microsoft Kinect camera. The objects are placed on the table in a known pose, and the grasps illustrated in Fig. 5, 6 and 7 are performed. For every nominal grasp pose, the hand has been positioned manually on the desired configuration via the gravity compensation control of the *Baxter* robot. A set of 6 trials is performed for each grasp, and grasp scores are averaged over the attempts over three indices:

Object	Pose	Effort	Grasp Position	Preshape
Hammer	P1	0.3	Handle	0
Hammer	P2	0.3	CoG	0
Hammer	P3	0.3	Head	0.5
Spray Bottle (no cap)	P1	0.2	Low (spray facing outside)	0
Spray Bottle (no cap)	P2	0.2	High (spray facing outside)	0
Spray Bottle (no cap)	P3	0.2	Low (spray facing inside)	0
Spray Bottle (no cap)	P4	0.2	High (spray facing inside)	0
Spray Bottle (with cap)	P1	0.2	High	0
Ketchup	P1	0.3	Cap	1.5
Ketchup	P2	0.3	Under cap ridge	1.3
Ketchup	P3	0.2	Lateral ridge	0
Ketchup	P4	0.2	Above lateral ridge	0
Pasta Box	P1	N/A	Top, parallel	N/A
Pasta Box	P2	N/A	Top, tilt along yaw	N/A
Pasta Box	P3	N/A	Side	N/A

TABLE II: Grasp configuration

Fig. 3: The 3d laser range scans are reconstructed using Poisson surface reconstruction, and the number of faces of the mesh is then reduced using Quadric Edge Collapse Decimation. All the meshes have a number of faces between 8000 and 12000. As a last step, the bottom of the object is manually cropped to a flat surface.

- $S1$ **object pose deviation**. This is designed in order to be performed easily by eye without complicated measurements attempts. As it can be seen in 4 it is practically useful only in the *hammer* set of grasps, because of the advantageous lever the CoM has on the grasp. In the bidimensional case, the index uniformly discretizes the possible swing angle of the hammer in 3 regions, with scores 1, 0.6 and 0.3. The score will be 0 in case of an unsuccessful grasp.
- $S2$ **the normalized number of fingers in contact**. For the Reflex Hand, the index can assume values 1 for a three-fingers grasp and 0.5 for a two fingers grasp. For the Soft Hand, the index can assume values 1,0.75,0.5 and 0.25.
- $S3$ **ability to resist perturbation**. By applying a shaking motion on the hand of the robot, we check if one of the previous indices changes in value as a consequence of the perturbation, in which case the score will be 0.6. In case of a dropped object as a consequence of the perturbation, the score will be 0.3.

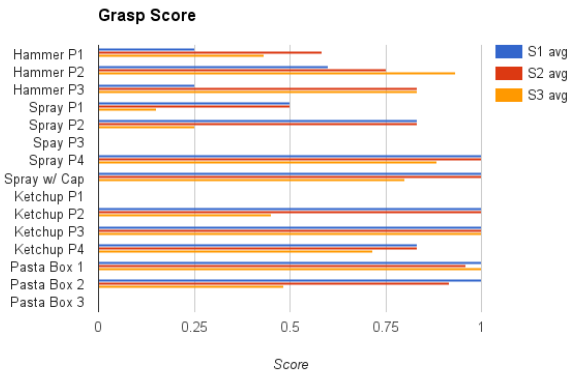

Fig. 4: Grasp scores for physical experiments. On the x axis, $1-3$ are for *hammer-P1* to *hammer-P3*, $4-7$ are for *spray-P1* to *spray-P4*, while score 8 is for *spray-full-P1*, $9-13$ are for *Ketchup-P1* to *Ketchup-P5*

The grasp procedure for the *Reflex Hand* consists in simultaneously closing each finger until the motor's effort threshold is met, at which point the finger is stopped. For the *Soft Hand*, the fingers close simultaneously until the nominal torque limit of the single actuator is reached. To check if the grasp is successful, the object is then raised. If successfully raised, the robustness of the grasp is checked by shaking the gripper by hand.

The grasp is then performed in simulation. We simulate the *Reflex Hand* using *Klamp't* v0.6.2, and the *Soft Hand* using *Gazebo* v4.0 with a preliminary version of the *Soft Hand plugin* [28]. The *Gazebo* simulator is tested against our patch [27] where *BLEM* is used. In Gazebo, the simple contact sorting algorithm is used for contact filtering. The *Klamp't* simulator is tested with *BLEM* against the default in Open Dynamics Engine *OPCODE*. In Klamp't examples, we use k-means clustering with a maximum of 20 contacts.

The robot's motor parameters, tendon Baumgarte stability constants, and friction parameters were tuned by hand so that the hands would move qualitatively similarly to the real hands, and to be stable both in free space motion and while grasping on a spherical object. No quantitative calibration procedure was performed.

The simulation procedure is as follows:

- The object and hands are spawned in the pregrasp and preshape configuration,
- As in the physical hardware, the hands are closed until an actuator force threshold is reached.
- The hand is lifted by 10cm,
- The grasp is perturbed by commanding jerking movements of the wrist,
- Finally the hand opens to drop the object.

Fig. 9 illustrates the results of the contact stability indices between BLEM and OPCODE contact generation. It can be observed BLEM leads to higher stability in general, most strikingly for the contact normal metric.

Grasp	Grasped	Dropped	Grasped	Dropped
ReFlex	**ODE / BLEM**		**ODE / OPCODE**	
Hammer 1	Y	Y	Y	**N**
Hammer 2	Y	Y	Y	Y
Hammer 3	Y	Y	Y	**N**
Spray 1+	Y	Y	Y	**N**
Spray 2+	Y	Y	Y	**N**
Spray 3*	Y	Y	Y	Y
Spray 4	Y	Y	Y	**N**
Spray Full 1	Y	Y	Y	Y
Ketchup 1*	Y	Y	**N**	–
Ketchup 2	Y	Y	**N**	–
Ketchup 3	Y	Y	Y	**N**
Ketchup 4	Y	Y	Y	**N**
SoftHand	**Gazebo / BLEM**		**Gazebo / OPCODE**	
Pasta Box 1	Y	Y	X	Y
Pasta Box 2+	Y	Y	X	Y
Pasta Box 3*	N	–	N	–

TABLE III: Success rates of ReFlex grasps (top) and SoftHand grasps (bottom) in simulation. Asterisks indicate failed grasps on physical experiments. The plus sign indicates non robust grasps on physical experiments. An entry of X indicates variability for the same initial conditions.

Tab. III illustrates the results of simulations. 12/12 suc-

Fig. 5: Comparison between simulation and experiment: *Hammer-P1*, *Hammer-P2*, *Hammer-P3*

Fig. 6: Comparison between simulation and experiment: *Ketchup-P1*, *Ketchup-P2*, *Ketchup-P3*, *Ketchup-P4*

Fig. 7: Comparison between simulation and experiment: *Spray Bottle (with cap)*, *Spray Bottle (no cap) - P1*, *Spray Bottle (no cap) - P2*, *Spray Bottle (no cap) - P3*, *Spray Bottle (no cap) - P4*

Fig. 8: Comparison between simulation and experiment: *Pasta Box - P1*, *Pasta Box - P2*, and *Pasta Box - P3*.

cessful grasps were reliably grasped by BLEM and 9/12 were grasped by OPCODE. In all successful grasps, the object was securely held through perturbations. In the two ODE/OPCODE failure cases, the hand penetrated completely through the object without contact. In the two Gazebo/OPCODE unreliable grasp cases, we observed non-determinism in the simulator which caused the grasp to fail in approximately 1/3 of simulation trials, even with the exact same initial conditions.

Perhaps more worrisome is the result that ODE/OPCODE did not successfully release the grasped object in 7/9 successful grasps. This is explained by major interpenetration artifacts, in which the object gets "stuck" on the robot's finger and cannot be let go (Fig. 10.

BLEM did not successfully predict 2/3 failed grasps. This likely indicates an improperly calibrated coefficient of friction or absolute grasp force. This suggests that although stable simulation is crucial for obtaining reasonable predictions, it is still challenging for simulation to accurately discern the boundary of feasible/infeasible grasps without considerable tuning. An interesting area for future work would be to automatically tune the robot model to obtain a match with physical experiments.

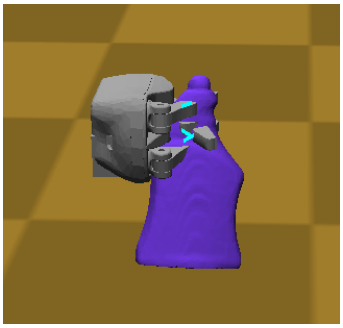

Fig. 10: An artifact exhibited in OPCODE causing the object to penetrate and then "stick" to the robot's finger.

Lastly, computation times during grasp for the *Klamp't* case were similar between BLEM, with 2.7ms and OPCODE, with $\sim$ 2.9ms for each millisecond of simulated time. For Gazebo, the experimental BLEM patch resulted in computation times of $\sim$ 2.4ms against $\sim$ 1.6ms of OPCODE for each millisecond of simulated time. All measurements were taken on an Intel Core i5-4460 CPU @3.2HGz.

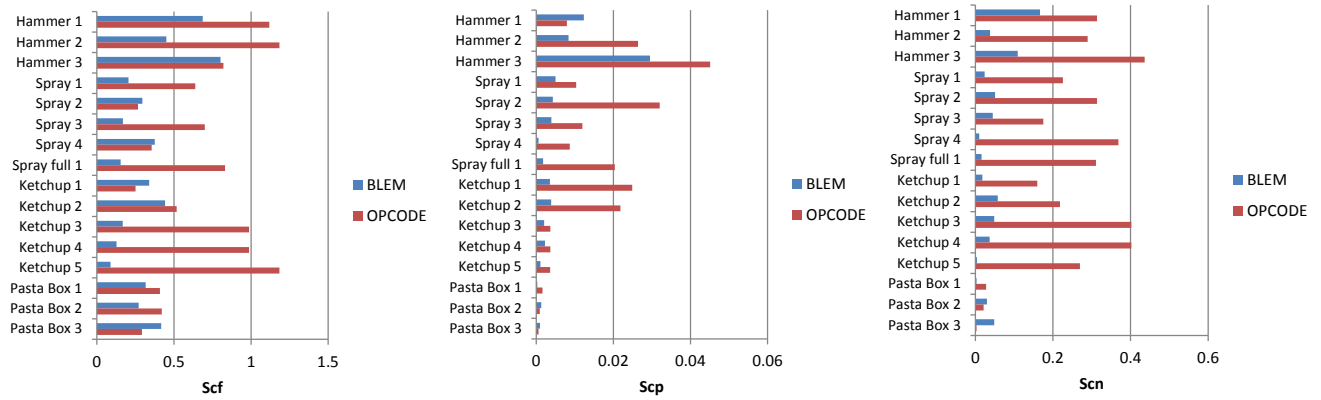

Fig. 9: Metrics S_{cf}, S_{cp}, and S_{cn} over the simulated grasp set. Lower numbers are better.

VI. Conclusion

A novel simulation technique for compliant and underactuated hands has been tested on two different hardware platforms, the *RightHand Robotics' Reflex Hand* and the *Pisa/IIT SoftHand*. A set of grasps was performed on the physical hardware using 4 different objects of different shapes and characteristics, and then validated in simulation. Experimental results show that simulations performed with the *BLEM* contact generation algorithm have better stability indices, exhibit fewer artifacts, and yield better fidelity to real-world experiments without extensive tuning.

References

[1] Bullet physics engine. http://bulletphysics.org/wordpress/. Accessed: 2016-1-20.

[2] Open dynamics engine. http://ode.org/. Accessed: 2016-1-20.

[3] M. Anitescu and F. Potra. Formulating dynamic Multi-Rigid-Body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics*, 14(3):231–247, 1997.

[4] L. Birglen. From flapping wings to underactuated fingers and beyond: a broad look to self-adaptive mechanisms. *Mechanical Sciences*, 2(1):5–10, 2011.

[5] A. Boeing and T. Bräunl. Evaluation of real-time physics simulation systems. In *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, pages 281–288. ACM, 2007.

[6] M. Bonilla, E. Farnioli, C. Piazza, M. Catalano, G. Grioli, M. Garabini, M. Gabiccini, and A. Bicchi. Grasping with soft hands. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pages 581–587. IEEE, 2014.

[7] M. G. Catalano, G. Grioli, E. Farnioli, A. Serio, C. Piazza, and A. Bicchi. Adaptive synergies for the design and control of the pisa/iit softhand. *The International Journal of Robotics Research*, 33(5):768–782, 2014.

[8] M. S. Corp. Adams. http://www.mscsoftware.com/product/adams. Accessed: 2015-08-26.

[9] R. Deimel and O. Brock. A compliant hand based on a novel pneumatic actuator. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2047–2053, May 2013.

[10] R. Deimel and O. Brock. A novel type of compliant, underactuated robotic hand for dexterous grasping. *Robotics: Science and Systems, Berkeley, CA*, pages 1687–1692, 2014.

[11] R. Diankov and J. Kuffner. OpenRAVE: A planning architecture for autonomous robotics. *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-34*, 79, 2008.

[12] T. Erez, Y. Tassa, and E. Todorov. Simulation tools for model-based robotics: Comparison of bullet, havok, mujoco, ode and physx. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 4397–4404, May 2015.

[13] C. Ferrari and J. Canny. Planning optimal grasps. In *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, pages 2290–2295. IEEE, 1992.

[14] G. Grioli, M. Catalano, E. Silvestro, S. Tono, and A. Bicchi. Adaptive synergies: an approach to the design of under-actuated robotic hands. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1251–1256. IEEE, 2012.

[15] K. Hauser. Robust contact generation for robot simulation with unstructured meshes. In *International Symposium on Robotics Research, Singapore*, 2013.

[16] J. M. Hsu and S. C. Peters. Extending open dynamics engine for the darpa virtual robotics challenge. In *Simulation, Modeling, and Programming for Autonomous Robots*, pages 37–48. Springer, 2014.

[17] D. Kappler, J. Bohg, and S. Schaal. Leveraging big data for grasp planning. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 4304–4311. IEEE, 2015.

[18] J. Kim, K. Iwamoto, J. Kuffner, Y. Ota, and N. Pollard. Physically based grasp quality evaluation under pose uncertainty. *Robotics, IEEE Transactions on*, 29(6):1424–1439, Dec 2013.

[19] N. Koenig and A. Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2149–2154 vol.3, Sept 2004.

[20] B. León, S. Ulbrich, R. Diankov, G. Puche, M. Przybylski, A. Morales, T. Asfour, S. Moisio, J. Bohg, J. Kuffner, et al. Opengrasp: a toolkit for robot grasping simulation. In *Simulation, Modeling, and Programming for Autonomous Robots*, pages 109–120. Springer, 2010.

[21] R. Ma, L. Odhner, and A. Dollar. A modular, open-source 3d printed underactuated hand. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2737–2743, May 2013.

[22] A. Miller and P. Allen. Graspit! a versatile simulator for robotic grasping. *Robotics Automation Magazine, IEEE*, 11(4):110–122, Dec 2004.

[23] L. U. Odhner, L. P. Jentoft, M. R. Claffee, N. Corson, Y. Tenzer, R. R. Ma, M. Buehler, R. Kohout, R. D. Howe, and A. M. Dollar. A compliant, underactuated hand for robust manipulation. *The International Journal of Robotics Research*, 2014.

[24] F. T. Pokorny and D. Kragic. Classical grasp quality evaluation: New theory and algorithms. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Tokyo, Japan, 2013.

[25] RightHand Robotics. Reflex sf spec sheet. http://www.righthandrobotics.com/main:reflex. Accessed: 2015-08-26.

[26] Robotiq. 3-finger adaptive robot gripper spec sheet. http://robotiq.com/products/industrial-robot-hand/. Accessed: 2015-08-26.

[27] A. Rocchi. Gazebo blem patch. https://bitbucket.org/arocchi/gazebo. Accessed: 2015-08-26.

[28] C. J. Rosales. Pisa/IIT soft hand. https://github.com/CentroEPiaggio/pisa-iit-soft-hand. Accessed: 2015-08-26.

[29] J. Weisz and P. Allen. Pose error robust grasping from contact wrench space metrics. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 557–562, May 2012.