A Design and Analysis Tool for Underactuated Compliant Hands

Matei Ciocarlie and Peter Allen

Abstract— Highly underactuated and passively adaptive robotic hands have shown great promise for robust performance in unstructured settings. In order to fully realize this potential, efficient tools are needed to analyze the execution of a grasp when using this class of devices. Along this line, this paper introduces a quasistatic analysis method for underactuated hands. First, we predict whether initial contacts between the fingers and the object are stable throughout the execution of a grasp, or the fingers will slip as the hand closes. Second, we compute the unbalanced forces applied to the object during the grasping process. Finally, once the grasp is complete, we analyze its stability as actuator forces are increased. These computations are performed in 3D, allow arbitrary kinematic structure of the fingers or geometry of the target object and take into account frictional constraints. We discuss applications of this method focusing on both on-line computation to execute a specific grasping task and off-line optimization to increase the range of grasps that can be performed using a given hand model.

I. INTRODUCTION AND RELATED WORK

Robots with the ability to operate in unstructured human environments are constantly evolving: the field has seen significant breakthroughs in areas such as localization, locomotion and navigation, visual servoing *etc*. However, before such robots can be successfully deployed for real-life applications, they must acquire the capability to interact with their environment by grasping (and eventually manipulating) surrounding objects. The ability to perform grasps of common objects (such as mugs, books, *etc*.) *in a robust and repeatable manner*, subject to realistic levels of sensing error and noise, is still considered an open problem, and a potential roadblock on the way to more complex applications.

A promising direction of research attempts to answer this problem through better robotic hand design, focusing on two key principles: underactuation and passive mechanical adaptation. The former is a well-established concept, traditionally implemented via rigid coupling between joints [1], [2]. More recent work has highlighted the advantages of combining underactuation with passive compliance, allowing the hand to adapt to the surface of the object at a mechanical, rather then computational level. There are multiple ways of achieving passive adaptation, such as breakaway transmission mechanisms [3], four-bar linkages [4] or tendon-driven compliant joints [5]; for a comprehensive review, we refer the reader to [6]. Hands built using these methods have been shown to be effective even in the presence of sensing errors, while

the highly reduced number of actuators can decrease the production cost and allow for faster design iterations.

While a complete optimization of the design parameters is often intractable in the case of complex hand designs, the apparent simplicity of highly underactuated hands suggests that optimization efforts can yield a high reward. However, it is crucial that any analysis methods take into account the particular nature of underactuated adaptive mechanisms. A number of such tools have been proposed. Birglen and Gosselin [7] have optimized the design of underactuated fingers to avoid the roll-back phenomenon where a finger recoils from the surface of the object. Dollar and Howe [6] have optimized the actuation and compliance forces of a tendon-driven design. These studies have led to the construction of remarkably efficient grippers. However, they share a number of limitations, such as two-dimensional frameworks, simplified friction models and highly simplified geometry of the target object. Generally, optimization of a highly underactuated, and thus deceptively simple hand, is a complex problem; in other words, *simple is hard*!

In this paper we propose a quasistatic force analysis tool for underactuated hands. We use a 3D framework with frictional constraints which poses no restrictions on the kinematics of the hand or the geometry of the grasped object. We discuss its applications in areas such as hand design optimization and grasp planning. It is important to note that no analysis method can be exhaustive, encompassing all the complex phenomena encountered during possible grasping tasks. We also discuss the limitations of our approach and possible ways to alleviate them. While robotic hand research can not yet produce a provably optimal hand design for a wide range of grasping tasks, it is our directional goal.

An important body of work has focused on the force generation capabilities of redundant or human-like tendon-driven mechanisms; relevant examples include [8]–[11]. Force generation has also been studied extensively in the context of fully actuated robotic hands, and a number of useful tools have been proposed; see [12]–[15] and references therein for details. However, highly underactuated and adaptive grasping present a number of additional challenges, some of which we attempt to tackle in our current work.

II. UNDERACTUATED GRASP ANALYSIS

A complete grasping process is commonly considered to have two phases: pre-grasp, where the initial shape of the hand as well as its location relative to the object are decided, and the execution of the grasp, when the fingers are closed until stable contact with the object is achieved. In this study we focus on highly underactuated hand models, where the

number of joints far exceeds the number of actuators. As mentioned above, recent studies have shown surprising levels of performance when using a single actuator for multiple fingers and up to 15 joints [16]. In such cases, there is limited flexibility in choosing the pre-grasp (and for single-actuator hands this phase is bypassed altogether). The finger closing stage assumes central importance, as the hand is expected to passively conform to the target object.

A key aspect of this process is that different fingers, as well as different links within a finger, make contact with the object at different times. With a fully actuated robot equipped with ideal sensors this phenomenon can be detected and the motor forces modulated so that the hand continues to close without applying any force at these contacts. In our case, the hand lacks the actuation mechanism needed to perform precise modulation of contact forces. As a result, while the hand continues to close, the links that have already made contact are applying some level of force to the object. We will refer to this as the *unbalanced force*, which has the following effects on the grasping process:

- if the unbalanced force exceeds the level that can be supported by friction between the target object and the surface that it's resting on, the object can be pushed away before the hand can fully enclose it.
- early contacts between the hand and the object might be unstable: the fingers can slip on the object surface. While in some cases this process can ultimately result in a stable grasp, it can also lead to the *rollback phenomenon* discussed in [7]. Our current analysis tags these cases as failures and only grasps where no slip occurs during the process are considered reliable enough for execution.
- after all the contacts have been established, actuator force is increased to the desired level for a stable grasp. However, when the object is lifted, any unbalanced forces in the hand-object system are no longer supported by friction between the object and the environment and will cause the hand to reconfigure itself. We thus seek grasps that minimize the amount of unbalanced force applied after all contacts have been established and the actuator force has reached the desired level for grasping.

It is important to note that the process presented above can prune out a number of configurations that, although unstable, can ultimately result in stable grasps (either through finger slip or hand re-configuration after object lift). In essence, our analysis method aims to identify underactuated grasps that are most likely to be stable throughout their execution. Intuitively, these grasps are the best candidates for reliable task completion. Furthermore, the ideal hand design should offer a wide range of stable postures over a variety of target objects. In the following section we will describe our formulation for performing this analysis.

III. QUASISTATIC COMPUTATION OF UNBALANCED FORCE

Consider an underactuated robotic hand with d actuators for m joints; in practice we focus on cases were $d \ll m$. The force generation characteristics of the hand will depend on

Parameter	Definition
m	number of joints
$\boldsymbol{\theta} \in \mathcal{R}^m$	joint angle values
$\boldsymbol{K} \in \mathcal{R}^{m \times m}$ diagonal	joint spring stiffnesses
$\boldsymbol{\tau} \in \mathcal{R}^m$	joint forces
p	number of contacts
$\boldsymbol{c} \in \mathcal{R}^{3p}$	column vector of contact forces
$\boldsymbol{J}_c(\theta)$	Jacobian of contact locations
d	number of actuators
$\boldsymbol{\alpha} \in \mathcal{R}^d$	actuator forces
$\boldsymbol{A}(\boldsymbol{\theta}) \in \mathcal{R}^{m \times d}$	matrix relating actuator to joint forces

TABLE I

NOMENCLATURE

the chosen underactuation method. Using the nomenclature presented in Table I, the applied joint forces are a function of the actuator forces, and possibly of the joint angle values, or $\boldsymbol{\tau} = f(\boldsymbol{\alpha}, \boldsymbol{\theta})$. In the rest of this paper, we focus on the case where force transmission is achieved through a tendon network using known insertion points, and passive adaptation is achieved through compliant joints modeled as linear springs of known stiffnesses. However, our formulation can be extended to other actuation mechanisms by changing the formulation of the function f used for computing $\boldsymbol{\tau}$.

Ignoring inertial effects, the system is in equilibrium if the joint forces due to actuation and joint springs are balanced by contact forces. Overall equilibrium is thus satisfied by:

$$\boldsymbol{J}_c^T \boldsymbol{c} = f(\boldsymbol{\alpha}, \boldsymbol{\theta}) = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{\alpha} + \boldsymbol{K}\theta \quad (1)$$

Our goals, as outlined in Section II, are to a) determine the existence of a set of legal contact forces that satisfy the equilibrium condition and b) compute the total force that the hand will apply on the object. In general, the system in (1) in undetermined. However, we must also add friction constraints on the contact forces, and, in case these constraints can be met, compute the solution that adds minimum energy to the system.

A. Friction Constraints

In order to model friction constraints, we start from the linearized formulation introduced in [17] and further discussed in [18], which we briefly review here. Consider first a single contact i where the total contact force is $\boldsymbol{c}_i \in \mathcal{R}^3$, the surface normal is $\hat{\boldsymbol{n}}_i$ and the magnitude of the normal force is $n_i \geq 0$. In the case of Coulomb friction, the tangential forces at the contact are limited by n_i and the friction coefficient μ_i. From a geometric standpoint, the tangential friction component of the contact force has to lie inside a "friction circle" of radius $\mu_i n_i$. This constraint can be linearly approximated by expressing the frictional component of the contact force as a weighted linear combination of k discrete vectors on the boundary of the friction circle:

$$\boldsymbol{c}_i = [\hat{\boldsymbol{n}}_i \ \boldsymbol{D}_i] [n_i \ \boldsymbol{\beta}_i]^T \quad (2)$$

Here the columns of the matrix $\boldsymbol{D}_i \in \mathcal{R}^{3 \times k}$ are the k vectors that sample the friction circle and $\boldsymbol{\beta}_i = \left[\beta_i^1, \ \beta_i^2, \ ..., \ \beta_i^k\right]$

is the vector of weights (in practice we use $k = 8$). Additionally, all the weights must be positive, and their sum is bounded by the magnitude of normal force:

$$[\mu_i \ -e]\,[n_i \ \boldsymbol{\beta}_i]^T \ \geq \ 0 \tag{3}$$

$$n_i, \boldsymbol{\beta}_i \ \geq \ 0 \tag{4}$$

where $\boldsymbol{e} = [1, \ 1, ..., 1] \in \mathcal{R}^k$.

Constraints (2) through (4) refer to a single contact i. We now assemble them in matrix form for the complete system:

$$\boldsymbol{c} \ = \ \boldsymbol{D\beta} \tag{5}$$

$$\boldsymbol{\beta}, \ \boldsymbol{F\beta} \ \geq \ 0 \tag{6}$$

where the vector of unknowns $\boldsymbol{\beta}$ contains the entries $[n_i \ \boldsymbol{\beta}_i]^T$ in block column form, the matrix $\boldsymbol{D}$ contains the entries $[\hat{\boldsymbol{n}}_i \ \boldsymbol{D}_i]$ in block diagonal form and the matrix $\boldsymbol{F}$ contains the entries $[\mu_i \ -e]$ also in block diagonal form.

We note that this model can be directly extended to consider complex frictional phenomena, such as soft finger contacts. In addition to tangential friction, such contacts can also apply frictional torque. Instead of a contact force we therefore have a contact wrench, whose frictional component is bounded to lie inside a "friction ellipsoid" [19]. This effect can be captured by linearizing the friction ellipsoid as shown in [20], and using the appropriate entries in the $\boldsymbol{D}_i$ matrices.

B. Optimization Problem

Even with contact constraints, the system can still allow an indeterminate number of solutions and we must choose the one that adds minimum energy to the system. We therefore compute the solution that minimizes the magnitude of the resultant wrench applied to the external object. For a contact i, we consider the matrix $\boldsymbol{R}_i \in \mathcal{R}^{6 \times 3}$ which transforms contact forces into 6D object wrenches (considering both the force and the torque produced on the object). The sum of the resulting object wrenches gives us the total object wrench $\boldsymbol{o}$:

$$\boldsymbol{o} = \boldsymbol{SRD\beta} \tag{7}$$

where $\boldsymbol{R}$ contains the entries $\boldsymbol{R}_i$ in block diagonal form and $\boldsymbol{S} \in \mathcal{R}^{6 \times 6p}$ is a summation matrix of the form $[\boldsymbol{I}_6, \ \boldsymbol{I}_6, ..., \boldsymbol{I}_6]$. The complete transform from the unknown contact force magnitudes $\boldsymbol{\beta}$ to the resultant object wrench $\boldsymbol{o}$ can be encapsulated in the grasp map matrix $\boldsymbol{G} = \boldsymbol{SRD}$.

We can finally formulate the complete contact problem:

$$\text{minimize} \ \ \boldsymbol{o}^T\boldsymbol{o} \ = \ \boldsymbol{\beta}^T\boldsymbol{G}^T\boldsymbol{G}\boldsymbol{\beta} \ \ \text{subject to :}$$

$$\boldsymbol{J}_c^T\boldsymbol{D\beta} \ = \ \boldsymbol{J}_t^T\boldsymbol{A}\boldsymbol{\alpha} + \boldsymbol{K}\theta$$

$$\boldsymbol{\beta}, \ \boldsymbol{F\beta} \ \geq \ 0$$

This is a standard Quadratic Program, with linear constraints. The matrix that defines the quadratic (and only) component of the objective function is positive semidefinite by definition, as it is the product of the matrix $\boldsymbol{G}$ and its transpose. Therefore, the optimization problem is convex, so whenever the conditions are feasible, a global minimum can be determined. In this study, we used the Mosek [21] package to solve all the optimization problems of this form.

Algorithm 1 Grasp analysis algorithm.

---
maxUnbal = 0
repeat
    close fingers incrementally until a new contact is made
    formulate contact force quadratic program
    unbal = Optimize(quadratic program)
    if program unfeasible **or** unbal > unbalThreshold **then**
       return unstableGrasp
    maxUnbal = MAX(unbal, maxUnbal)
until all fingers stopped
set desired level of actuation forces
formulate contact force quadratic program
finalUnbal = Optimize(quadratic program)
if program unfeasible **then**
    return unstableGrasp
graspQuality = f(maxUnbal, finalUnbal)
return graspQuality

There are three possible results to the optimization problem presented above:

- the problem is unfeasible; this indicates that no legal contact forces exist that can balance the system. The fingers will slip on the surface of the object.
- the problem is feasible and a non-zero global optimum is found; the contacts are stable but some level of unbalanced force is applied to the object. If this force is not balanced externally (*i.e.* by interactions between the target object and another surface in the environment), the hand will have to reconfigure itself.
- the problem is feasible and the global optimum is zero; the contacts are stable and contact forces balance each other on the object producing a null resulting wrench. The hand-object system is stable in its current configuration.

Before proceeding to applications of this method, we must note that alternative formulations of contact friction constraints are also possible. Anitescu and Potra [17] use the same linearized model applied here, but cast it as a Linear Complementarity Problem solvable using Lemke's algorithm. Buss *et al.* [13] use an exact quadratic friction model formulated as positive semi-definiteness constraints. Han *et al.* [15] extend this approach and solve it using Linear Matrix Inequations. The formulation presented here enables straightforward control over the minimized objective function, but using different approaches, such as the ones above, might also be feasible. We intend to explore these possibilities as part of our future studies.

C. Grasp Execution Analysis

We can now present the analysis method applied to a complete grasp execution. The goal is, for a given starting position and finger closing direction, to predict if the outcome is a successful grasp and, if so, to also assign it a numerical quality metric. A step-by-step execution is presented in Algorithm 1.

We note that, in the presented form, our algorithm does not provide an exact formulation of the grasp quality metric,

but rather suggests that it can be computed as a function of both the maximum level of unbalanced force created while closing the fingers and the unbalanced force of the final grasp, after actuator forces have been set to the desired levels. An ideal grasp will minimize both of these values. However, the weight placed on each of these goals can be adapted to the particular characteristics of the hand and the environment. In the results presented in the following sections, we chose to use only the final unbalanced force as the returned quality metric; other choices are also possible.

The final grasps can also be pruned according to other quality metrics, such as the widely used Grasp Wrench Space (GWS) ϵ metric proposed by Ferrari and Canny [22]. In our implementation, we prune all final grasps that have an ϵ value below 0.05. This metric builds the GWS by considering only contact frictional constraints. The presence of additional underactuation constraints would require further pruning of the GWS; as such, the ϵ value we currently use can be considered an upper bound for the true GWS quality of an underactuated grasp.

IV. APPLICATIONS AND RESULTS

To showcase the analysis method introduced above, we propose two applications. The first one focuses on existing hand designs: given an underactuated hand and an object to be grasped, it can be used to quickly prune a very large number of possible grasps and keep only the best candidates for stable execution. Our current results are presented in a simulated environment using a full dynamics engine with inertial effects to provide baseline performance and ground truth. The second application aims at an earlier stage of the hand design process: our method can be used to efficiently quantify the capabilities of a hand design over a wide range of objects and grasping scenarios. This enables the analysis of design choices and optimization parameters before the final design is set and the hand is constructed.

A. Grasp Planning

Grasp planning is a common problem in robotic hand research (for review see [23], [24]), usually formulated as follows: given a hand and a target object, find a hand configuration and location relative to the object in order to ensure a stable grasp. As we discussed in Section II, in the case of underactuated hands, the number of starting configurations is limited, but the process of passive adaptation during execution is complex. Here we apply our analysis method to evaluate these configurations and decide if they are likely to result in a stable grasp.

In our implementation we used a model of the SDM Hand introduced by Dollar and Howe [5]. This hand uses a single actuator to drive eight joints that articulate four fingers, relying almost exclusively on passive adaptation for grasping a wide range of objects. Our planning method, implemented using the publicly available *GraspIt!* simulation engine [25], goes through the following stages:

- create a large number (between 150 and 1000) of possible grasping positions for each object in our test set. Figure 1
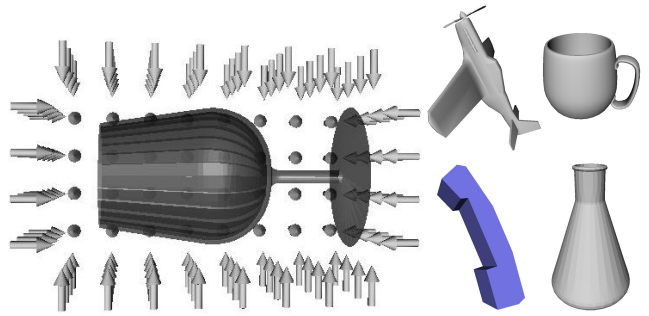

Fig. 1. Set of five objects (glass, flask, mug, phone and toy airplane) used in our tests. The glass also shows a number of possible grasps generated by aligning the hand with its bounding box. For each possible grasp, the approach direction (shown by the arrows) was parallel to one axis of the bounding box. The rotation of the hand around the approach direction (not shown here) was set by aligning the hand with the other axes of the box.

shows the objects in our set, and exemplifies the sampling process for creating candidate grasps. This was done by aligning the hand with the bounding box of each object and advancing towards the object until first contact is made.
- analyze each possible grasping position using the quasistatic analysis algorithm presented in Section III-C. Sort the grasps in order of the quality metric.
- use *GraspIt!*'s dynamics engine to simulate the execution of a grasp candidate in order to provide ground truth and a computational performance baseline. This engine, presented in [18], uses a time step integration method to compute body velocities and accelerations in response to actuator and contact forces. If the dynamic execution of a grasp results in the object firmly held in the hand against gravity, the grasp is deemed to be successful.

Our first test was intended to provide a baseline performance measure: for each object, we tested all candidate grasps using the dynamic engine. The percentage of successful grasps over the entire set of objects was 17% (the complete results for each object are shown in Table II, third column). The result shows that this hand is indeed an effective grasping device, but a random choice of approach direction yields an unsatisfactory success rate. Furthermore, the average time for complete dynamic analysis of all the grasp candidates was 20 minutes per object, restricting its applicability in on-line grasping scenarios.

In contrast, the quasistatic analysis algorithm is significantly faster. In our experiments, totaling more than 1500 grasps over 5 objects, the time required for quasistatic analysis of a stable grasp ranged between 100 and 200 milliseconds. Bad grasps are evaluated even quicker, as either finger slip or a high level of unbalanced force lead to an early exit with an *unstableGrasp* label. In general, the time required for analyzing a complete set of candidate grasps densely sampled along the object bounding box ranged between 16 and 75 seconds. All of our experiments were performed on a commodity desktop workstation equipped with a 2.13GHz Intel Core2 CPU.

After this analysis was done, we selected the 10 most stable grasps from the ordered list of candidates. One exception was the flask, for which only 8 stable grasps were found

Object	#G	%G-S	#QS	%QS-S	QS-time
Flask	542	13%	8	100%	45.18s
Plane	849	9%	10	90%	75.14s
Mug	337	14%	10	90%	29.56s
Phone	177	44%	10	100%	16.17s
Glass	220	39%	10	80%	36.9s

#G:	total number of grasp candidates generated
%G-S:	percentage of candidates from this list that result in a stable grasp
#QS:	number of best candidates taken from the list ordered through quasistatic analysis
%QS-S:	percentage of those that result in a stable grasp
QS-time:	time required to perform the quasistatic analysis and return the best candidates

TABLE II

QUASISTATIC ANALYSIS FOR GRASP PLANNING.

(this is intuitively explained by the conic shape of this object which makes it difficult for this hand model to hold against gravity). This subset was then tested using the dynamic engine. Our complete results are presented in Table II. We note that the quasistatic approach provides an efficient and reliable method of pruning down a very large number of possible grasps to a small number of reliable candidates.

For application in real life environments, this method places a number of requirements on the sensing capabilities of the system. One possibility is to acquire a model of, or recognize, the object to be grasped. The method presented above can then be applied to find reliable grasps for execution. Another possible option would use tactile sensors and proprioception to analyze the grasp currently being executed. Both of these options require extensive sensing, which runs against the stated motto of simplicity and low-cost designs. An interesting alternative is to optimize the hand off-line so that a wider range of grasps can be executed with increased robustness. We believe that replacing grasp-specific on-line computation with off-line hand design optimization can prove a fruitful direction for improving robotic grasp performance. We explore this option in the following section.

B. Hand Design Optimization

We used the quasistatic analysis presented here to investigate how grasping performance can be improved by changing hand design parameters. Using a model of the SDM Hand, we focused on two such parameters: the actuator torque ratio between the proximal and distal joints of each finger and the spring stiffness ratio between the same joints. These parameters are determinant for the behavior of the hand, as they affect both the posture of the hand before touching an object and the forces transmitted after contact is made. In particular, we investigated all possible combinations ranging from 0.2 to 1.0 (in steps of 0.2) for the torque ratio and from 0.1 to 1.0 (in steps of 0.1) for the stiffness ratio.

The benchmark test consisted of 2000 possible grasps distributed evenly across the 5 models in our test set. For each torque and stiffness combination, we tested all the candidate grasps and reported the number of them that are stable throughout their execution. To enable direct comparison
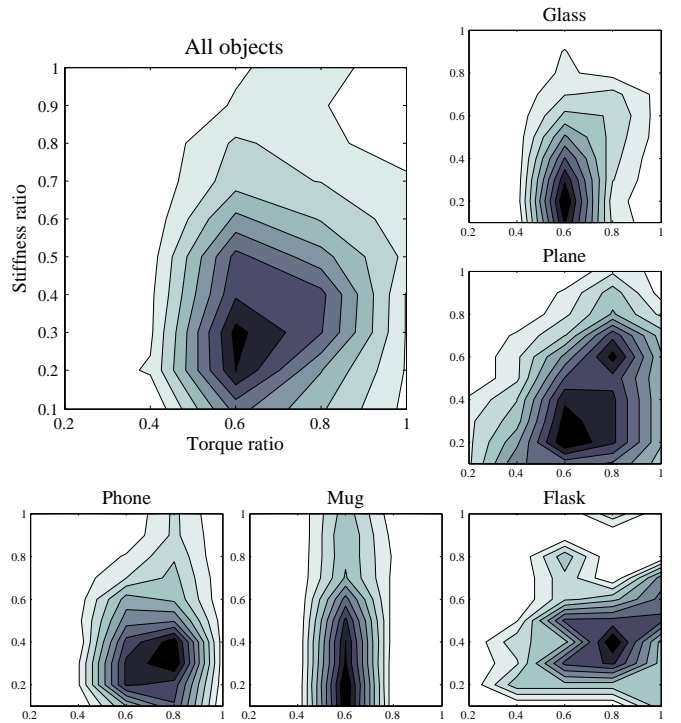

Fig. 2. The effect of hand design parameters on the likelihood of obtaining a stable grasp. For each combination of joint torque and stiffness ratios, the color indicates the number of stable grasps obtained from a fixed set of candidate grasps; a darker color means a higher number of stable grasps. For each object, the results were normalised to a scale of 0 to 1 by dividing by the maximum number of grasps found for that object.

across different objects, each set of results was normalized to a scale of 0 to 1 through division by the maximum number of grasps found for that object. Figure 2 shows these results for each of the five objects, as well as their average over the entire set. The contour maps reveal which areas of the optimization range offer the best performance; in particular they suggest a torque ratio of 0.6 and a stiffness ratio of 0.3. The overall resemblance between the patterns suggests that the global optimum of the average profile is a good compromise, likely to work well on all objects. However, the patterns exhibit enough variation to illustrate the importance of performing this analysis over a large set of models, spanning a wide range of shapes and grasping scenarios. We also note that our torque ratio is in agreement with the optimal value found in the optimization studies that led to the construction of the SDM Hand [6].

The focus of the present study is the analysis method itself rather than a particular design choice or optimization task. We therefore chose only one of the many aspects of a hand model that can be analyzed in similar fashion. These include kinematic chain design, link lengths and shapes, number of fingers, *etc*. In this light, the computational performance of the analysis method becomes a key aspect: a more efficient algorithm will allow for more design iterations, investigating more parameters over a larger domain. The analysis presented here consisted of a total of 20,000 grasps for each object (400 candidates for each of the 50 combinations of force and torque ratios); the typical time spent per object was

15 minutes. This performance level suggests the possibility of scaling up to significantly larger test sets.

In recent work, we have introduced and validated a large corpus ($n = 1,814$) of 3D models annotated with pre-computed grasp information for a number of dexterous hand models [26]. These results have underlined the advantage of using a large knowledge base for on-line grasp planning; we conjecture that this approach can also prove useful in the case of hand design. Using a computationally efficient analysis method, such as the one presented here, a given hand model can be benchmarked across this entire set of models, quantifying its overall performance and identifying the range of shapes that are particularly difficult to grasp.

V. CONCLUSION AND FUTURE WORK

In this paper we have introduced a quasistatic analysis tool for highly underactuated hands performing grasping tasks. At its core, our method attempts to compute the contact forces that satisfy the static equilibrium conditions, given a hand configuration and a set of contacts with the grasped object. If such forces do not exist, the configuration is reported to be unstable as the fingers will slip on the surface of the object. If the equilibrium constraints are feasible, we compute the resultant wrench applied to the target object. The magnitude of this wrench provides a measure of the stability of the hand-object system, and implicitly of the grasp.

The equilibrium conditions are solved through formulation as a convex Quadratic Program with linear constraints. This enables us to consider arbitrary hand kinematics, friction constraints and complex 3D object geometry. In addition, this method is computationally efficient and can be used to analyze hundreds of grasps in less then a minute. Therefore, it is well suited for two applications that we have presented. The first one is grasp planning, where a large number of possible grasps must be evaluated quickly to select the best candidates for reliable execution. The second application is hand design, where the effect of changing parameter values can be quantified efficiently by evaluating hand performance over a large number of simulated grasp scenarios.

Due to the complex interactions between multiple bodies under various constraints, modeling grasp execution in general, and underactuated grasping in particular, are challenging tasks. Current analysis tools can not claim to take into account all of its aspects in a computationally efficient way, and our approach is no exception. In particular, our algorithm discards a number of configurations that it detects are unstable, but which could ultimately result in stable grasps. The planning results we have presented show that, while our method reliably identifies a number of viable grasp candidates, it also eliminates a significant number of other grasps that could also be executed successfully. In future work, we would like to address these cases, beginning with quasistatic simulation of finger slip conditions.

Complementing improvements to the analysis tool itself, we would like to expand on the two applications that we have presented initial results for. Our planning algorithm can be extended to identify grasps that not only have a high chance of successful execution, but are also robust to hand positioning errors. Hand design applications are potentially limitless, as a large number of design parameters can be improved through optimization methods analogous to the one presented here. Our ultimate goal is to realize the hand design equivalent of a constructive proof in mathematics: to show that a hand exists that can provably perform a given set of grasping tasks, and in the process to also show how to perform these tasks, in a repeatable and robust manner.

REFERENCES

[1] J. Butterfass, M. Grebenstein, H. Liu, and G. Hirzinger, "DLR-Hand II: Next generation of a dextrous robot hand," in *IEEE Intl. Conf. on Robotics and Automation*, 2001, pp. 109–114.

[2] C. Lovchik and M. Diftler, "The Robonaut hand: a dextrous robot hand for space," in *IEEE Intl. Conf. on Robotics and Automation*, 1999.

[3] W. T. Townsend, "The BarrettHand grasper - programmably flexible part handling and assembly," *Industrial Robot*, vol. 10, no. 3, 2000.

[4] L. Birglen, T. Laliberte, and C. Gosselin, *Underactuated Robotic Hands*. Springer Tracts in Advanced Robotics, 2008.

[5] A. Dollar and R. Howe, "Simple, robust autonomous grasping in unstructured environments," in *IEEE Intl. Conf. on Robotics and Automation*, 2007.

[6] A. Dollar and R. Howe, "Joint coupling design of underactuated grippers," in *30th Annual Mechanisms and Robotics Conference*, September 2006.

[7] L. Birglen and C. Gosselin, "Kinetostatic analysis of underactuated fingers," *IEEE Trans. on Robotics and Automation*, vol. 20, 2004.

[8] R. Kurtz and V. Hayward, "Dexterity measure for tendon actuated parallel mechanisms," in *IEEE Intl. Conf. on Advanced Robotics*, 1991.

[9] A. Bicchi and D. Prattichizzo, "Analysis and optimization of tendinous actuation for biomorphically designed robotic systems," *Robotica*, vol. 18, pp. 23–31, 2000.

[10] N. Pollard and R. Gilbert, "Tendon arrangement and muscle force requirements for humanlike force capabilities in a robotic finger," *IEEE Intl. Conf. on Robotics and Automation*, pp. 3755–3762, 2002.

[11] J. Fu and N. Pollard, "On the importance of asymmetries in grasp quality metrics for tendon driven hands," in *IEEE-RAS Intl. Conf. on Intelligent Robots and Systems*, 2006.

[12] M. Mason and K. Salisbury, *Robot hands and the mechanics of manipulation*. MIT Press, 1985.

[13] M. Buss, H. Hashimoto, and J. Moore, "Dextrous hand grasping force optimization," *IEEE Trans. on Robotics and Automation*, vol. 12, 1996.

[14] L.-W. Tsai, *Robot Analysis*. John Wiley & Sons, 1999.

[15] L. Han, J. Trinkle, and Z. Li, "Grasp analysis as linear matrix inequality problems," *IEEE Trans. on Robotics and Automation*, vol. 16, 2000.

[16] C. Gosselin, F. Pelletier, and T. Laliberte, "An anthropomorphic underactuated robotic hand with 15 Dofs and a single actuator," *IEEE Intl. Conf. on Robotics and Automation*, 2008.

[17] M. Anitescu and F. A. Potra, "Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems," *Nonlinear Dynamics*, vol. 14, pp. 231–247, 1997.

[18] A. Miller and H. Christensen, "Implementation of multi-rigid-body dynamics within a robotic grasping simulator," in *IEEE Intl. Conf. on Robotics and Automation*, 2003, pp. 2262–2268.

[19] R. Howe and M. Cutkosky, "Practical force-motion models for sliding manipulation," *Intl. J. of Robotics Research*, vol. 15, no. 6, 1996.

[20] M. Ciocarlie, C. Lackner, and P. Allen, "Soft finger model with adaptive contact geometry for grasping and manipulation tasks," in *Joint Eurohaptics Conf. and IEEE Symp. on Haptic Interfaces*, 2007.

[21] "Mosek ApS Denmark," http://www.mosek.com.

[22] C. Ferrari and J. Canny, "Planning optimal grasps," in *IEEE Intl. Conf. on Robotics and Automation*, 1992, pp. 2290–2295.

[23] K. B. Shimoga, "Robot grasp synthesis algorithms: a survey," *Intl. J. of Robotics Research*, vol. 15, pp. 230–266, 1996.

[24] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," *IEEE Intl. Conf. on Robotics and Automation*, pp. 348–353, 2000.

[25] A. Miller and P. K. Allen, "GraspIt!: a versatile simulator for robotic grasping," *IEEE Robotics and Automation Mag.*, vol. 11, no. 4, 2004.

[26] C. Goldfeder, M. Ciocarlie, H. Dang, and P. Allen, "The Columbia grasp database," *IEEE Intl. Conf. on Robotics and Automation*, 2009.