To appear in Proceedings of the 1996 IEEE Conference on Robotics and Automation

Online Grasp Synthesis*

Jefferson A. Coelho Jr. Roderic A. Grupen

Laboratory for Perceptual Robotics †
Department of Computer Science
University of Massachusetts, Amherst, 01003

Abstract

This paper addresses the online grasp synthesis problem, in which statically stable grasp configurations are to be generated given instantaneous contact information. We use the preimage *of a given grasp controller to address object resolution, friction compensation, and resource allocation in a online fashion. The approach can be applied to piecewise smooth object geometries, and it can be extended to accommodate any number of contacts and 3D objects.*

1 Introduction

Dextrous manipulation and grasping tasks challenge conventional control methods, requiring the command of highly redundant systems operating in uncertain and unstructured settings. Multiple, conflicting goals must be attained in a broad range of control contexts. Typically, the computational burden associated with control becomes overwhelming, as the state space grows exponentially with the number of degrees of freedom (DOF) of the manipulator system.

This paper addresses the problem of online generation of statically stable multifingered grasp configurations. Our approach casts grasp synthesis as a robust control problem wherein variations in object geometry and contact friction properties are modeled as *disturbances*, and grasp configurations are adjusted in response to contact feedback. This formulation alleviates significantly the computational burden related to model building and grasp planning, and provides a mechanism for executing grasp plans when they exist. Continuous sensor feedback also addresses uncertainty and/or imprecision during grasp execution.

*Supported in part by NSF CDA-8922572, IRI-9116297, IRI-9208920, and by CNPq 202107/90.6.

†http://piglet.cs.umass.edu:4321

1.1 Related Work

In 1982, Salisbury employed the grip Jacobian, G, to transform contact forces to object frame wrenches[14]. Since then, the issue of constructing appropriate contact configurations (and therefore grip Jacobians) has received a great deal of attention. The *grasp synthesis* problem may be viewed as a search in contact configuration space for configurations that can transmit the right set of wrenches to the object. Most work in this regard concerns off-line grasp synthesis and may be categorized into two broad categories:

- Optimization techniques have been used to explore contact configuration space guided by an evaluation function, that measures the utility of the grasp configuration based on several possible attributes. In [5, 13], the set of configurations that satisfy the force closure condition are further sorted on the basis of precision requirements, and in [6], sorting is based on the total force applied to the object. Other evaluation functions incorporate manipulator kinematics and joint range limits, the ability of the grasp to resist external disturbances, the ratio of contact force magnitude to object wrench magnitude, and the number of contacts in addition to force closure[1, 9, 11]. In these broader contexts, the complexity of optimization-based approaches can be prohibitive.
- Taxonomy-based grasp synthesis observes that grasp geometry is strongly related to hand morphology. Taking cues from human grasping strategies, these techniques relate hand configuration (and thus contact configuration) to the geometry of the object in the context of the task [4, 10, 16]. This approach is a memory-intensive map from object geometry and task to appropriate hand configuration and pose.

These approaches can be distinguished on the basis of object geometries permitted (polygons, parametric surfaces, two and three dimensions), and on the basis of the type and number of contacts permitted. All depend on complete *a priori* information concerning object geometry and, in some cases, friction coefficients.

Elsewhere [2, 3], we have shown that statically stable grasps can be formed for some classes of objects without *a priori* models of the object's surface geometry. Specifically, two controllers were defined and composed according to a context-dependent rule, such as to minimize force and moment residuals over an object frame. We showed that equilibria in the composite controller correspond to optimal contact configurations for 2 and 3 contacts on regular, convex polygons. Furthermore, we have also presented a compensation mechanism for extending the composite controller to non-regular geometries.

This paper extends the grasp controller presented in [2, 3], introducing mechanisms for online friction compensation and contact (de)allocation. Grasp strategies are constructed which are guaranteed not to exceed a pre-specified limit on contact friction. We also show that it is possible to perform *object resolution* (the elimination of possible object interpretations from a initial set of objects) during the execution of a grasp, and adapt the grasp strategy as resolution is processed. In our formulation, object resolution is a secondary goal of the system – the grasp controller will remove objects from its set of candidate objects until it converges to a grasp solution. The remaining object interpretations in the candidate set are by construction compatible with the grasp solution found.

2 Control Basis for Grasping

This section reviews the control basis for grasping introduced in [2]. The proposed control basis consists of two controllers: the force residual ($\mathcal{C}_f$) controller, and the moment residual ($\mathcal{C}_m$) controller. Their definition assumes unit magnitude contact forces, transmitted to the object through frictionless point contacts. Both controllers attempt to minimize ϵ, the feedback grasp error defined as the squared magnitude of the net object frame wrench:

$$\epsilon \;=\; \vec{\rho}^{\,T}\vec{\rho} \;=\; \left(\frac{1}{n}\sum_{i=1}^{n}\vec{\omega}_i\right)^T \left(\frac{1}{n}\sum_{i=1}^{n}\vec{\omega}_i\right), \quad (1)$$

where $\vec{\rho}$ is the normalized wrench residual and expresses the net wrench over n contacts, and $\vec{\omega}_i$ is the wrench resulting from the i^{th} contact force.

Note that $\vec{\rho} = \vec{0}$ is a necessary condition for stable grasp configurations. The squared residual, ϵ, can be minimized by following its negative gradient with respect to the contact configuration. $\frac{\partial\epsilon}{\partial\theta_i}$ is given by:

$$\frac{\partial\epsilon}{\partial\theta_i} \;=\; 2\vec{\rho}^{\,T}\frac{\partial\left(\frac{1}{n}\sum_i\vec{\omega}_i\right)}{\partial\theta_i} \;=\; \frac{2}{n}\left(\frac{\partial\vec{\omega}_i}{\partial\theta_i}\right)^T\vec{\rho}, \quad (2)$$

where $\frac{\partial\vec{\omega}_i}{\partial\theta_i}$ is the gradient of the i^{th} contact wrench with respect to the coordinate θ_i. In vector notation

$$\frac{\partial\epsilon}{\partial\vec{\theta}} \;=\; \frac{2}{n}\begin{bmatrix} (\frac{\partial\vec{\omega}_1}{\partial\theta_1})^T \\ (\frac{\partial\vec{\omega}_2}{\partial\theta_2})^T \\ \vdots \\ (\frac{\partial\vec{\omega}_n}{\partial\theta_n})^T \end{bmatrix}\vec{\rho} \;=\; \frac{2}{n}W^T\vec{\rho}. \quad (3)$$

Both W and $\vec{\rho}$ in Eq. 3 require an expression for the contact wrench as a function of contact coordinates. This function depends directly on the geometry of the object. To represent a range of object geometries, we have developed two artificial potentials from which control may be derived, as described below.

2.1 Force Residual Control

The force residual potential models contact wrenches as if the contacts were interacting with a spherical object geometry. If θ and ϕ are the angular coordinates on the sphere, then the corresponding object frame wrench, $\vec{w}_f(\theta,\phi)$, can be written:

$$\begin{array}{lcllcl} w_{fx} &=& -cos(\theta)cos(\phi) & w_{mx} &=& 0 \\ w_{fy} &=& -sin(\theta)cos(\phi) & w_{my} &=& 0 \\ w_{fz} &=& -sin(\phi) & w_{mz} &=& 0. \end{array}$$

The force residual error ϵ_f is defined by inserting $\vec{w}_f(\theta,\phi)$ into Eq. 1. Likewise, the force residual controller $\mathcal{C}_f$ is defined by inserting $\vec{w}_f(\theta,\phi)$ into Eq. 3.

2.2 Moment Residual Control

The moment residual potential models contact wrenches as if each contact were interacting with a planar object facet. The resulting model $\vec{w}_m(\theta,\phi)$ is:

$$\begin{array}{lcl} w_{fx} &=& -cos(\theta_n)cos(\phi_n) \\ w_{fy} &=& -sin(\theta_n)cos(\phi_n) \\ w_{fz} &=& -sin(\phi_n) \\ w_{mx} &=& -r_\theta sin(\phi_n)cos(\theta_n) + r_\phi sin(\theta_n) \\ w_{my} &=& -r_\theta sin(\phi_n)sin(\theta_n) - r_\phi cos(\theta_n) \\ w_{mz} &=& r_\theta cos(\phi_n), \end{array}$$

where θ_n, ϕ_n are the angle coordinates for the normal to the contact plane that intersects the origin of the object frame, and r_θ, r_ϕ are the coordinates of the contact on the facet. The point in which the normal defined by θ_n, ϕ_n intersects the contact plane defines the origin of the r_θ, r_ϕ coordinate frame. The insertion of $\vec{w}_m(\theta,\phi)$ into Eqs. 1 and 3 defines ϵ_m and $\mathcal{C}_m$, respectively. The models $\vec{w}_f$ and $\vec{w}_m$ describe 3D objects; 2D object models can be derived by making ϕ constant. See [2, 3] for further details.

2.3 Control Composition

The composition of $\mathcal{C}_f$ and $\mathcal{C}_m$ defines the grasp controller $\mathcal{C}^*$, as follows:

$$\mathcal{C}^* \quad : \quad \text{if} \quad \epsilon_f > \tau \quad \text{then} \quad \mathcal{C}_f \quad \text{else} \quad \mathcal{C}_m, \tag{4}$$

where τ is a small positive force residual threshold. By construction, $\mathcal{C}_f$ yields a unique $\epsilon_f = 0$ grasp configuration in the plane for $k = 2$ and 3 consisting of evenly spaced contacts separated by $2\pi/k$ radians. Then, $\mathcal{C}_m$ is activated to minimize the moment residual ϵ_m. We have shown that the $\mathcal{C}^*$ is correct for the class of regular convex polygons, grasped by 2 or 3 contacts[2].

3 Control Preimage

Each grasp configuration $\vec{\theta}$ on a given object can be associated with a point $\vec{\epsilon} = [\epsilon_f \ \epsilon_m]^T$ in the error plane – the *state* of the grasp. As the $\mathcal{C}^*$ controller modifies the current contact configuration, a succession of states $\{\vec{\epsilon}^{\,0}, \vec{\epsilon}^{\,1}, \ldots, \vec{\epsilon}^{\,t}\}$ is constructed. Upon convergence, the minimum friction coefficient μ_0 required for static equilibrium is evaluated, and used to label the equilibrium state and its precursor states:

$$\mu^{k+1}(\epsilon_f, \epsilon_m) = \max(\mu_0, \mu^k(\epsilon_f, \epsilon_m)),$$

where k denotes the previous grasping trial. We refer to $\mu(\epsilon_f, \epsilon_m)$ as the *preimage* of controller $\mathcal{C}^*$; it predicts the minimum required friction coefficient on convergence, in the worst case. The pattern of accessible states in μ encodes a wrench domain signature of objects from the perspective of the $\mathcal{C}_f$ and $\mathcal{C}_m$ controllers.

The $\mathcal{C}^*$ preimage for a given object is constructed over a number of grasping trials; in each trial, n contacts are placed randomly over the object surface, and $\mathcal{C}^*$ runs until convergence. After convergence, μ_0 is calculated and used to label all visited states.

The minimum required friction coefficient μ_0 is determined by performing a binary search in the interval $[0, 1]$. The estimated value of μ_0 is used to specify constraints in a mathematical programming problem that determines whether an appropriate contact force distribution exists. The search algorithm follows:

***Compute*-μ_0** (μ_{min}, μ_{max})
  if $(\mu_{max} - \mu_{min} < \delta)$
    return (μ_{max});
  $\mu = 0.5(\mu_{max} + \mu_{min})$;
  if (***Equilibrium-Possible*** $(\mu, \vec{\Theta})$)
    return ***Compute*-μ_0** (μ_{min}, μ);
  else
    return ***Compute*-μ_0** (μ, μ_{max}).

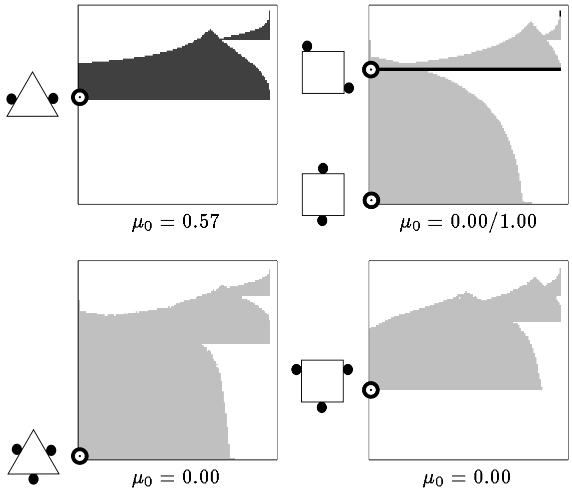

Figure 1: $\mathcal{C}^*$ preimage for the triangle and square objects, for 2 and 3 contacts.

where δ specifies the desired precision of μ_0, and $\vec{\Theta}$ is the vector of contact coordinates and normals. More efficient algorithms for identifying μ_0 are available [15]; the approach reported was chosen for its simplicity.

The function ***Equilibrium-Possible*** $(\mu, \vec{\Theta})$ returns `TRUE` if a contact force solution exists given μ and the contact configuration. The problem is expressed as a mathematical programming problem [12, 8] subject to constraints for each contact:

1. $|\vec{f}_n| \geq +1$ — normal forces are compressive with magnitude greater than or equal to 1,
2. $|\vec{f}_t| \leq \mu|\vec{f}_n|$ — the net contact force must remain within the friction cone, and
3. $\sum \vec{f} = \sum \vec{r} \times \vec{f} = \vec{0}$.

The objective function for the resulting linear programming problem is immaterial, since we are only concerned with whether the set of constraints yields a non-empty solution space. The binary search finds the minimum friction coefficient for which this is true.

Figure 1 depicts the $\mathcal{C}^*$ preimages for 2 and 3 contact grasps on a regular triangle and square. The error plane is represented by a 128×128 grid. Vertical (ϵ_m) and horizontal axes (ϵ_f) are on a logarithmic scale, $log(max(\epsilon, 0.01)/0.01)$. The value for μ_0 at each cell is proportional to the grey level and is also recorded numerically under each plot; white cells are inaccessible for the given object. The circles indicate attractor states for $\mathcal{C}^*$, or the states to which $\mathcal{C}^*$ converges; the corresponding grasp configurations are shown in the figure. Notice that the square has one spurious attractor (top circle); for numerical reasons, τ (the minimum force residual in Eq. 4) must satisfy $\tau > 0$, what allows for "bad" solutions whenever a regular polygon with an even number of edges is grasped by pairs of contacts.

4 Friction Compensation

It is possible to combine the preimage of many objects into a composite preimage, defined as:

$$\mu_0^*(\epsilon_f, \epsilon_m) = \max(\mu_0^{obj}(\epsilon_f, \epsilon_m))$$

where *obj* is an element of an object database Ω. μ_0^* predicts the worst-case friction coefficient required over all objects in Ω. The top row of Figure 3 depicts instances of composite preimages.

The composite preimage μ_0^* can be used to implement friction-dependent grasp strategies, by making the control actions of $\mathcal{C}^*$ dependent on the friction coefficient μ_{task}, supplied by the system user. The idea is to build a conservative grasp controller, that is either driving the contacts towards a grasp solution, or searching for a state (ϵ_f, ϵ_m) from which a favorable outcome is guaranteed, that is $\mu_0^*(\epsilon_f, \epsilon_m) \leq \mu_{task}$.

Depending on the value of μ_{task}, it is not always possible to achieve a stable grasp configuration given the object and the number of contacts; for example, grasping a regular triangle with two fingers requires $\mu \geq 0.57$; if μ_{task} is smaller than 0.57, the $\mathcal{C}^*$ controller will keep on searching for a solution, until an upper limit on the number of control steps is reached. A simple but effective search mechanism involves superimposing a small, random control action onto the $\mathcal{C}^*$ controller whenever the $\mu_0^*(\epsilon_f, \epsilon_m) > \mu_{task}$. As soon as the controller enters the preimage of a suitable attractor ($\mu_0^*(\epsilon_f, \epsilon_m) \leq \mu_{task}$), the random control action is deactivated and $\mathcal{C}^*$ converges. This procedure is described by the following algorithm:

Controller $\mathcal{C}^*$ $(\vec{\Theta}, \mu_{task}, \Omega)$
  ϵ_f = force-residual $(\vec{\Theta})$*;*
  ϵ_m = moment-residual $(\vec{\Theta})$*;*
  if $(\epsilon_f > \tau)$
    $\Delta\theta = -k\frac{\partial \epsilon_f(\vec{\Theta})}{\partial \theta}$*;*
  else
    $\Delta\theta = -k\frac{\partial \epsilon_m(\vec{\Theta})}{\partial \theta}$*;*
  if $(\mu_0^*(\epsilon_f, \epsilon_m) > \mu_{task})$
    $\Delta\theta = \Delta\theta + \nu.$

where $\nu \sim \mathcal{N}(0, 0.0001)$[1].

[1] ν is a random number drawn from a normal distribution with mean zero and variance 0.0001.

4.1 Contact (De)Allocation

In many cases, unsatisfactory grasps can be repaired by allocating additional contacts. The addition of a contact changes the composite preimage μ_0^*, and consequently the control policy. An additional contact is allocated whenever $\mu_0^*(\epsilon_f, \epsilon_m) > \mu_{task}$ for all objects in Ω. The location of the extra contact will in practice be determined by kinematic constraints on the multifingered hand system; we adopted the policy of inserting the extra contact at a fixed distance from the last contact.

Conversely, the system can remove contacts whenever the resulting $n-1$ contact configuration is such that $\mu_0^*(\epsilon_f, \epsilon_m) \leq \mu_{task}$ for all candidate objects. The decision of which contact to remove can be made so as to improve the predicted outcome at the resulting grasp state.

4.2 Object Resolution

Object recognition tasks involve identifying which element from a database of objects is being probed by either visual or tactile sensors. Grimson [7] presents a comprehensive review of the latest object recognition research efforts. Typically, the initial candidate object set includes the complete object database; gradually, sensor information from the unknown object(s) becomes available, and those objects in the candidate set that do not conform with these measurements are then removed from the set.

If the preimage $\mu(\epsilon_f, \epsilon_m)$ for each element of the object database is complete, i.e. every accessible state has received its labeling, then it is possible to use the composite preimage μ_0^* to remove objects from the candidate set. In this framework, the accessible states in the composite preimage μ_0 are a rotationally-invariant representation of the object, given a certain number of contacts. The mechanism for eliminating objects from the candidate set is very simple: if $\mu_0^{obj}(\epsilon_f, \epsilon_m)$ is not accessible, then remove *obj* from the set of candidate objects, and update the composite map μ_0^*. The construction of a complete preimage $\mu_0(\epsilon_f, \epsilon_m)$ for a given object may require a large number of grasping trials; alternatively, it could be done off-line, if a geometric model of the object is available.

5 Results

In order to assess the performance of the $\mathcal{C}^*$ controller, we defined a set of eight objects: four regular polygons, two irregular polygons, and two piecewise smooth objects – see Figure 2.

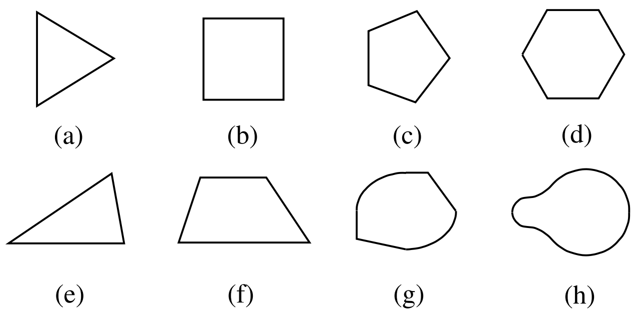

Figure 2: Set of candidate objects: (a) Triangle, (b) Square, (c) Pentagon, (d) Hexagon, (e) Irregular Triangle, (f) Trapezoid, (g) Leaf, and (h) Bulb.

Figure 3 illustrates a typical grasp on a trapezoid for which $\mu_{task} = 0.1$. The top row depicts the composite pre-image for the objects in the bottom row, given the current number of contacts. The initial grasp configuration has two contacts ($\theta = (0, \pi)$). At $t = 0$, the regular and the irregular triangle (depicted in grey in Figure 3) are eliminated. All other objects have plausible grasp interpretations for the (ϵ_f, ϵ_m) state derived from the trapezoid. At $t = 1$, the controller inserts a third contact in the grasp, at $\theta_2 \approx \pi$. At $t = 21$, the controller removes contact 0 ($\theta_0 \approx 0$), and subsequently rejects the pentagon, the hexagon, the leaf, and the bulb objects. At this point, the set of candidate objects contains only the square and the trapezoid; the controller converges at $t = 87$ to a suitable grasp configuration, requiring a friction coefficient $\mu \leq \mu_{task}$.

The performance of the proposed grasp controller was evaluated by subjecting it to 10,000 grasping trials. In each trial, the object to be grasped was selected at random from the set of eight objects depicted in Figure 2. Next, the object database was completed through the random selection of other objects, for database sizes varying from 2 to 8 objects. The resulting database was then inputed to the grasp control algorithm described previously. For each trial, the value of μ_{task} was set to the minimum possible value of μ_0, given the object identity, and the initial number of contacts (2 contacts). For instance, the trapezoid object is in equilibrium if it is grasped by two contacts on the top and bottom edges ($\mu_0 = 0$), or if the contacts are on the lateral edges and $\mu = 0.44$; in this case, $\mu_{task} = 0$.

Three parameters were used to evaluate system performance with respect to changing database size (Table 1): percentage of random actions, percentage of timeouts (percentage of trials in which the predicate $\mu_0 \leq \mu_{task}$ has not been satisfied after 1000 control steps), and the average number of objects in the object database after convergence.

Table 1: System behavior with changing database size.

Database size	Perc. random actions	Percentage timeouts	# objects after conv.
2	20.07	1.74	1.24
3	27.21	1.68	1.54
4	32.13	1.29	1.81
5	34.90	1.48	2.07
6	36.66	1.19	2.36
7	40.01	1.78	2.58
8	40.84	1.98	2.89

In Table 1, the percentage of random control actions increases monotonically with the size of the database, but in an apparently sub-linear fashion. Increased randomness does not seem to affect the number of timeouts – in fact, the overall average for timeouts is 1.59%. Table 1 also shows an almost linear increase on the number of objects in the database after convergence. This suggests that the grasp controller eliminates a larger percentage of objects from the database as its size increases. Overall, 54% of the convergent grasps required 3 contacts, the remaining 46% of the grasps were two-contact grasps.

6 Conclusion

We have extended the grasp synthesis framework introduced in [3], by storing in the control preimage a worst-case estimate on the friction coefficient required for static equilibrium, after convergence of the grasp controller. This information incorporates the friction coefficient between manipulator and object into the grasp synthesis process, allowing for online friction compensation. In addition, the approach described provides mechanisms for online object resolution and contact allocation. Yet, the whole approach is economical in terms of sensory information: it requires only local contact position and normals.

The proposed framework can be improved upon in many ways; for instance, it would be possible to enhance the object resolution capabilities of the system by keeping track of the history of interaction between object and contacts – the current approach is essentially memoryless. Likewise, contact addition and removal could be explicitly done so as improve the ongoing resolution task.

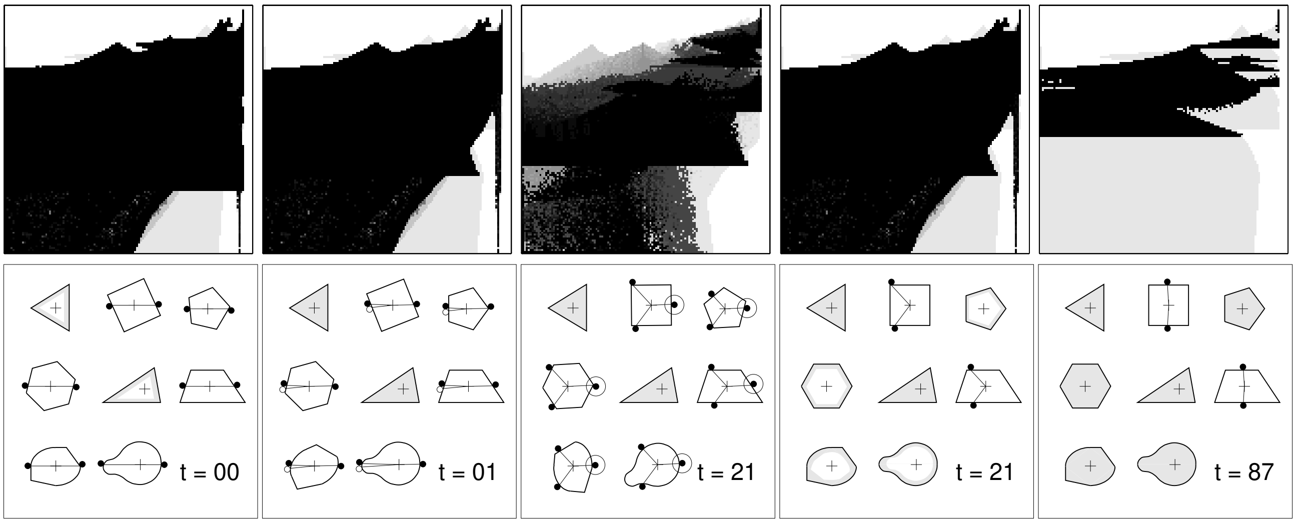

Figure 3: Grasping a trapezoid with two contacts, with $\mu_{task} = 0.1$.

References

[1] Chen, I., and Burdick, J. Finding antipodal point grasps on irregularly shaped objects. In *Proc. IEEE Int. Conf. Robotics Automat.* (Nice, France, May 1992), pp. 2278–2283.

[2] Coelho Jr., J. Effective multifingered grasp synthesis. Master's thesis, UMass, Department of Computer Science, Amherst, MA, Sept. 1993.

[3] Coelho Jr., J., and Grupen, R. Control preimaging for multifingered grasp synthesis. In *Proc. IEEE Int. Conf. Robotics Automat.* (San Diego, CA, May 1994), pp. 3117–3122.

[4] Cutkosky, M. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Trans. Robotics Automat. 5*, 3 (June 1989).

[5] Faverjon, B., and Ponce, J. On computing two-finger force-closure graps of curved 2d objects. In *Proc. IEEE Int. Conf. Robotics Automat.* (Sacramento, CA, May 1991), pp. 424–429.

[6] Ferrari, C., and Canny, J. Planning optimal grasps. In *Proc. IEEE Int. Conf. Robotics Automat.* (Nice, France, May 1992), pp. 2290–2295.

[7] Grimson, W. *Object recognition by computer : the role of geometric constraints.* The MIT Press, Cambridge, MA, 1990.

[8] Grupen, R., Biggers, K., Henderson, T., and Meek, S. Task defined internal grasp wrenches. Tech. Rep. UUCS-88-001, Department of Computer Science, University of Utah, 1988.

[9] Guo, G., Gruver, W., and Jin, K. Grasp planning for multifingered robot hands. In *Proc. 1992 IEEE Int. Conf. Robotics Automat.* (Nice, France, May 1992), vol. 3, pp. 2284–2289.

[10] Iberall, T. Grasp planning for human prehension. In *Proc. IJCAI* (1987), pp. 1153–1156.

[11] Jameson, J., and Leifer, L. Automatic grasping: An optimization approach. *IEEE Trans. Syst., Man, Cybern. SMC-17*, 5 (Sept./Oct. 1987).

[12] Kerr, J., and Roth, B. Analysis of multifingered hands. *Int. J. Robotics Res. 4*, 4 (1986), 3–17.

[13] Nguyen, V. Constructing stable grasps. *Int. J. Robotics Res. 8*, 1 (1989), 26–37.

[14] Salisbury, J. K. *Kinematic and Force Analysis of Articulated Hands.* PhD thesis, Stanford University, Stanford, CA, May 1982.

[15] Schwartz, J., and Sharir, M. Finding effective "force targets" for two-dimensional multifinger frictional grips. Tech. rep., Robotics and Computer Science Vision Laboratory, Courant Institute of Mathematical Science, 1986.

[16] Tomovic, R., Bekey, G., and Karplus, W. A strategy for grasp synthesis with multifingered robot hands. In *Proc. IEEE Int. Conf. Robotics Automat.* (Raleigh, NC, April 1987), pp. 83–89.