Proceedings of the 2011 CPSC515
Probabilistic Robotics
Vancouver BC – December 2011

Dynamic Grasp Synthesis of Simple Convex Shapes

Daniel M. Troniak
Dept. of Computer Science
University of British Columbia
Vancouver, BC Canada
troniak@cs.ubc.ca

Abstract – The general grasp planning problem has been studied in the academic community for decades. Developing efficient and versatile grasping systems has great implications in automating tasks that require dexterous and compliant manipulation. Traditional approaches tend to favor optimality over efficiency, producing grasping systems that are slow. I propose a new strategy that combines successful probabilistic approaches, object model abstraction schemes and force-closure constraint approximations into a single unifying algorithm. It is my hypothesis that through combining successful techniques already published in the literature, a more efficient strategy can be developed. The proposed algorithm will be tested in the dynamic simulation environment GraspIt! and then tested on the physical Barrett Hand and WAM arm system.

I. Introduction

The planning of grasps for multi-fingered hand-arm systems is a complex problem. This complexity is largely due to the many degrees of freedom (DOF) of modern systems. A feasible and collision-free trajectory must be generated within this high-dimensional space such that the end-effector can stably grasp arbitrary objects.

There are three distinct steps to a successful grasp-and-lift strategy. First, a stable grasp must be generated for the hand-object pair. Since there are (potentially infinitely) many ways in which one may grasp an object (for example grasping a cup by the handle or by the base), it is necessary to calculate an appropriate grasp in a given context, within a reasonable timeframe. Second, the inverse kinematics of the robot arm must be solved in order to correctly place the end-effector in an appropriate state to perform the grasp. This is through direct manipulation of the various control joints that compose the arm. Finally, a feasible trajectory that solves the inverse kinematics problem must be selected such that there are no environmental collisions or self-collisions along this trajectory.

The contribution of this paper is to develop a grasp-and-lift strategy that executes in real time, approaching human speeds. In order to attain this speed requirement I propose the use of: (1) probabilistic path planning, (2) intelligent object model simplification techniques and (3) efficient force-closure constraint approximation. Performance of the developed grasping strategies will be implemented in the simulation environment GraspIt! [1] and then ported to the physical Barrett Hand and WAM arm system.

There are numerous applications to fast grasp-and-lift strategies. Faster object manipulation could be beneficial in time-sensitive environments such as manufacturing facilities, for example. Objects of varying size and shape could be continuously presented to the robot on a conveyor belt, to which the robot is then tasked to sort them based on size and shape estimates. As another example, faster, more compliant manipulation systems would clearly enhance the human-robot interaction experience within human-centered environments (for personal service, caregiving, etc.).

Following this introduction to the grasp planning problem, section II of this proposal presents the solutions already published in the literature, and how my solution is unique. Section III defines the concept of a stable grasp more formally. Section IV and V present details of my solution (including a tentative project timeline) and how I expect my system to compare to existing solutions. Finally, section VI presents a brief summary of the project followed by future research that I would like to pursue that is beyond the scope of this directed study.

II. Related Work

Grasp planning for robots with many-DOF is known to be P-Space hard [2]. Nevertheless, there exist examples in the literature that address the problem of planning grasps within reasonable timeframes. Miller et al in 2003, for example, recorded the generation of one to 44 candidate grasps of arbitrarily shaped objects between 11.4 and 478 seconds [3]. Harada et al, in 2008, published simulation results of 10 to 25 seconds to generate successful grasps by introducing heuristics and *force-closure* approximation strategies [4]. The probabilistic approach to grasp planning for high DOF systems has achieved much success over the past few years [5-8]. The most impressive results with respect to fast grasp planning published so far were in 2010 by Vahrenkamp et al. They were able to achieve successful grasps with a physical robot between 1.7 and 3.7 seconds. [6]

Constraining the system to the kinematics of a specific hand has proven to be fruitful in simplifying the problem further. In [9], the authors synthesize two and three-finger grasps by way of heuristic metrics for assessing grasp quality with the Barrett Hand. They also present rules on how to generalize grasping strategies developed specifically for the Barrett Hand to any arbitrary manipulator.

Another way to reduce complexity of the overall system is to assume objects are convex and therefore more easily graspable. Common everyday objects can be abstracted into one of seven categories of generic convex shapes, such as spheres or a tapered cylinders [10]. Another strategy is by way of decomposing complex 3D object models into a series of *superquadrics*. Once a feasible number of possible objects are determined, rules to generate grasp starting positions & pregrasp shapes can be easily formulated [11].

One novelty in the method proposed by Harada et al [4] is an approach to simplifying the force-closure detection policies for the robotic hand. By approximating the *friction cone* via an ellipsoid, quicker albeit rougher estimates of stable grasp acquisition can be attained.

Reducing complexity in the overall system is not the only path to developing faster manipulation strategies. Incorporating dynamics in the state space of the robotic system for the purposes of dynamic object regrasping is demonstrated in [12]. Their developed system executes extremely high-speed motions as a result.

Surprisingly, little work has been done in unifying the above approaches. Therefore, this paper proposes a novel approach to grasp planning based on successful probabilistic strategies and employing various heuristics and appropriate simplifying assumptions, such as simplified object models and relaxed force-closure constraints.

III. Problem Formulation

Let us now explore the grasping problem more formally. First we introduce the current state of the art in the kinematic and dynamic modeling of a grasp. We then use this notation to formally analyze what is called the force-closure constraint.

Kinematic Modeling

Before we derive formal definitions of grasp synthesis, I will introduce the notation commonly used in the grasp literature, as presented in [13]. For a quick summary of notation, please refer to Table 1. For a graphical representation of the kinematic model, see Figure 1.

Let {N} represent the global coordinate frame, located at some convenient and fixed location within the workspace. We then denote $c_i \in \mathbb{R}^3$ for each contact point i in {N} between the object and the hand. Then $\{C\}_i$ represents each of the coordinate frames with axes $\{ n_i, t_i, o_i\}$, where n_i is a unit vector normal to the contact tangent plane, directed toward the object. The unit vectors t_i, o_i are orthogonal and lie in the tangent plane of contact.

The robot hand comprises joints numbered from 1 to n_q, each with relative joint loads $\tau = [\tau_1, \cdots, \tau_{n_q}]$. For the Barrett hand, which we are concerned with in this study, $n_q = 3$ (one controllable joint per finger). Each of the joint loads are torques measured at the base of each finger during object manipulation tasks. Figure 2 shows the Barrett Hand grasping an object.

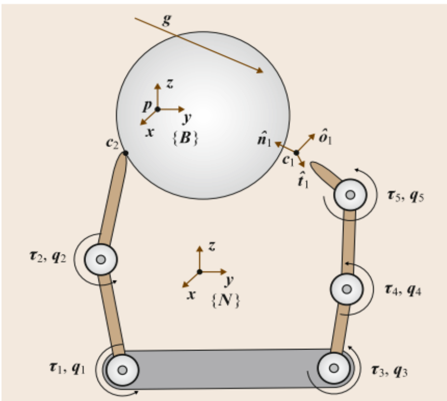

Figure 1: The general kinematic model of a grasp [13]

Let {B} represent a frame fixed to the object with its origin defined relative to {N} by the vector p $\in \mathbb{R}^3$(most conveniently positioned at the object's center of mass). Position and orientation of the object is then denoted as $u \in \mathbb{R}^6$, which is also the transformation of the object's coordinate frame {B} to match the world frame {N}. Velocities of the object are denoted as $v = [v^T \omega^T]^T \in \mathbb{R}^6$ and represent the positional velocity of p: $v \in \mathbb{R}^3$ and the object's angular velocity:$\omega \in \mathbb{R}^3$.

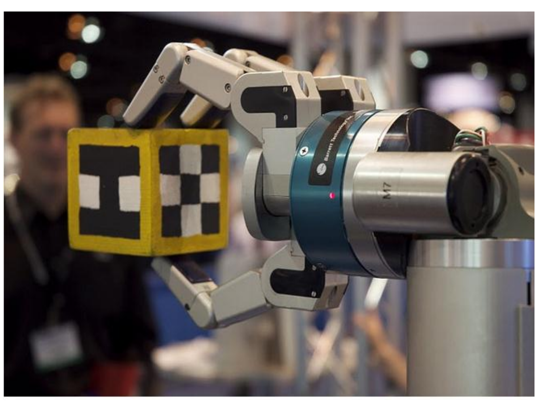

Figure 2: The Barrett Hand grasping an object

Finally, let $f \in \mathbb{R}^3$ represent the force applied to object at point p and moment (vector of motion) $m \in \mathbb{R}^3$. All external forces acting on the object are then combined into the *wrench* vector $g = [f^T m^T]^T \in \mathbb{R}^6$. Both f and m are expressed in {N} whereas g is expressed in any frame fixed to the object. The vector g is an example of a *noncontact* wrench. [13]

Table 1: Summary of Notation [13]

Notation	Definition
n_c	number of contacts
n_q	number of joints of hand
n_v	number of degrees of freedom of object
$\boldsymbol{q} \in \mathbb{R}^{n_q}$	joint displacements
$\dot{\boldsymbol{q}} \in \mathbb{R}^{n_q}$	joint velocities
$\boldsymbol{\tau} \in \mathbb{R}^{n_q}$	noncontact joint loads
$\boldsymbol{u} \in \mathbb{R}^{n_u}$	position and orientation of object
$\boldsymbol{v} \in \mathbb{R}^{n_v}$	twist of object
$\boldsymbol{g} \in \mathbb{R}^{n_v}$	noncontact object wrench
$\{\boldsymbol{B}\}$	frame fixed in object
$\{\boldsymbol{C}\}_i$	frame at contact i
$\{\boldsymbol{N}\}$	inertial frame

Dynamics and Equilibrium

Obtaining control over the dynamics of a system is desirable specifically if we are concerned with more than simply the kinematics of a stable grasp. The dynamics of our model can be summarized by the following equations:

$$M_{hnd}(q)\ddot{q} + b_{hnd}(q,\dot{q}) + J^T\lambda = \tau_{app} \qquad (1)$$
$$M_{obj}(u)\dot{v} + b_{obj}(u,v) - G\lambda = g_{app} \qquad (2)$$

Let $M_{obj}(\cdot)$ and $M_{\text{hand}}(\cdot)$ represent mass of the object and hand respectively and $b_{obj}(\cdot,\cdot)$ and $b_{hand}(\cdot,\cdot)$ be velocity product terms, which incorporate Coriolis and centrifugal forces into the system. Let λ be the wrench intensity vectors transmitted through all contact points, c. Specifically,

$$\lambda = \left[\lambda_1^T \ldots \lambda_{n_c}^T\right]^T | \; \lambda_i^T = \left[f_{in}\, f_{it}\, f_{io}\, m_{in}\right]^T.$$

where there exist one normal (n) and two tangential (t,o) components of each f and one normal component of each m.

Let $J^T\lambda$ represent the net joint load experienced at all finger joints and $G\lambda$ be the total wrench applied to the object by the hand. Then, g_{app} is the amount of force being applied to the object by external forces and τ_{app} is a vector of torques applied to the hand as a combination of external loads and actuator actions.

Incorporating Kinematic Constraints

So far, we have not considered any kinematic constraints in our dynamical model (Equations 1 & 2). Constraints on the system can be introduced in the following form:

$$\begin{pmatrix} J^T \\ -G \end{pmatrix} \lambda = \begin{pmatrix} \tau \\ g \end{pmatrix}$$

subject to $J\dot{q} = G^T v = v_{cc}$, where

$$\tau = \tau_{app} - M_{\text{hnd}}(q)\ddot{q} + b_{\text{hnd}}(q, \dot{q}), \qquad (3)$$

$$g = g_{app} - M_{obj}(u)\dot{v} + b_{obj}(u, v), \qquad (4)$$

and where v_{cc} denotes the twist of both object and hand (transmitted via contact points, c). [13]

Formal Grasp Analysis

The success of our system depends largely on its ability to obtain a stable grasp that is adequate to lift an object off the ground. We present in this section a formal definition of a stable grasp as well as the basic steps required to determine whether or not a given grasp has met these stability requirements.

A stable grasp can be described in terms of (1) *form closure* or (2) *force closure*. An object is considered form-closed if the object is constrained by the grasp so that its number of DOF becomes zero. As a subset of form closure grasps, *force closure grasps* allow the use of friction to keep the object in equilibrium. There is however usually an extra requirement that force closure grasps obtain control over all internal forces of the grasped object as well. Since we are interested in force closure grasps in our study, we define them in detail here.

Force Closure

Form closure grasps require that the number of contacts n_c with the object be in excess of the object's number of DOF (i.e. 7 contact points required for a 6 DOF rigid body). In contrast, *force closure* grasps require fewer contacts (only 2 for that same 6 DOF object assuming a soft-fingered manipulator). We must, however also ensure that an adequate amount of force is applied to the object (incurring appropriate frictional forces) to compensate for arbitrary external noncontact wrenches (such as gravity).

Considering Coulomb friction, frictional components of λ_i must lie within a circle of radius $\mu_i f_{in}$ where μ_i is the coefficient of friction at contact i. Then, the Coulomb *friction cone* is a subset of $\mathbb{R}^3$:

$$F_i = \left\{ (f_{in}, f_{it}, f_{io}) \middle| \sqrt{f_{it}^2 + f_{io}^2} \le \mu_i f_{in} \right\}$$

Definition: Frictional Form Closure

A grasp is said to have obtained frictional form closure if and only if:

$$\left.\begin{array}{c} G\lambda = -g \\ \lambda \in F \end{array}\right\} \forall \boldsymbol{g} \in \mathbb{R}^{n_v}$$

where F is the composite friction cone:

$$F = F_1 \times \cdots \times F_{n_c} = \{\lambda_i \in F_i; i = 1, \ldots, n_c\}$$

Murray et al [14] define *frictional form closure* to exist if and only if:

1. $Rank(\boldsymbol{G}) = n_v$
2. $\exists \lambda | \boldsymbol{G}\lambda = \boldsymbol{0}$ and $\lambda \in \boldsymbol{Int}(\boldsymbol{F})$

where $Int(F)$ denotes the interior of the composite friction cone.

Definition: Force Closure

Murray et al then define a stricter definition for full force closure, which also requires that the grasp obtain control over all internal object forces. A grasp has *force closure* if and only if:

1. $Rank(\boldsymbol{G}) = n_v$
2. $N(\boldsymbol{G}) \cap N(\boldsymbol{J}^T) = \boldsymbol{0}$
3. $\exists \lambda | \boldsymbol{G}\lambda = \boldsymbol{0}$ and $\lambda \in \boldsymbol{Int}(\boldsymbol{F})$

While there do exist exact nonlinear techniques for evaluating the friction cone constraint, it is more common for the friction cone to be approximated by some finitely sided pyramid. An example of such an approximation strategy is shown in Figure 3.

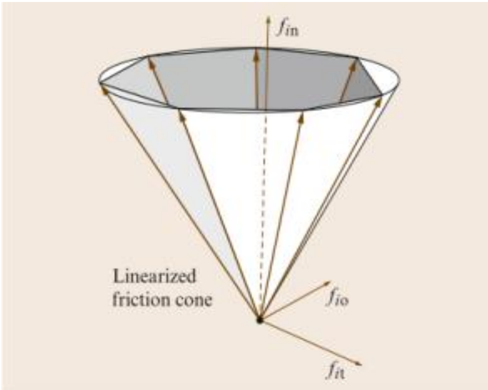

Figure 3. Quadratic cone approximated by seven-sided pyramid. [13]

We then represent wrenches at the object's i-th contact point:

$$G_i \lambda_i = S_i \sigma_i,\ \sigma \geq 0,$$

where $S_i = (s_{i1} \cdots s_{in_g})$ and σ_i is a vector of nonnegative generator weights. For soft-fingered manipulators (such as the Barrett Hand), we define:

$$\boldsymbol{S}_i = \left(\begin{array}{cccccc} \cdots & 1 & \cdots & 1 & 1 \\ \cdots & \mu_i \cos(2k\pi/n_g) & \cdots & 0 & 0 \\ \cdots & \mu_i \sin(2k\pi/n_g) & \cdots & 0 & 0 \\ \hline \cdots & 0 & \cdots & b\upsilon_i & -b\upsilon_i \end{array}\right),$$

where b is the characteristic length used to unify units. All face normal constraints can be combined into:

$$F\lambda \geq 0,$$

where $F = diag(F_1, \ldots, F_{n_c})$. Function *diag* here transforms a vector into a diagonal matrix. [13]

Systematically Determining Force Closure

The following function can be formulated to solve for frictional form closure:

$$\begin{array}{llr} \textit{LP2:} & \text{maximize :} & d \\ & \text{subject to:} & \boldsymbol{G\lambda} = \boldsymbol{0} \\ & & \boldsymbol{F\lambda} - \boldsymbol{1}d \geq \boldsymbol{0} \\ & & d \geq 0 \\ & & \boldsymbol{e\lambda} \leq n_c, \end{array}$$

where d measures distance of contact forces from boundaries of the friction cone and $e\lambda$ represents the sum of magnitudes of the normal components of all contact forces. It follows that if d = 0, frictional form closure does not exist, if d > 0 then it does. [13]

Finally, to determine full force closure, we must verify the condition:

$$N(G) \cap N(J^T) = 0,$$

if this holds, then the grasp is said to have force closure. The above condition can be determined by way of another function:

$$\begin{array}{llr} \textit{LP3:} & \text{maximize :} & d \\ & \text{subject to:} & \boldsymbol{G\lambda} = \boldsymbol{0} \\ & & \boldsymbol{J}^\top \boldsymbol{\lambda} = \boldsymbol{0} \\ & & \boldsymbol{E\lambda} - \boldsymbol{1}d \geq \boldsymbol{0} \\ & & d \geq 0 \\ & & \boldsymbol{e\lambda} \leq n_c, \end{array}$$

where $E = diag(e)$. [13]

We conclude this section with a summary of how force closure is determined via three high-level steps:

1. **Compute rank(G)**
   a. If rank$(G) \neq n_v$ then force closure does not exist. Break.
   b. If rank$(G) = n_v$, continue.
2. **Solve LP2**: Test frictional form closure
   a. If d = 0, then frictional form closure does not exist. Break.
   b. If d > 0, then frictional form closure exists, continue.
3. **Solve LP3**: Test control of internal forces.
   a. If d > 0, then force closure does not exist. Break.
   b. If d = 0, then force closure exists. Return true.

IV. Solution Methodology

The criteria for a good solution to the general grasp planning problem shall be defined as follows:

1. Generate a stable grasp of a convex polygonal object for the Barrett Hand.

2. Solve the Inverse Kinematics of the WAM arm, bringing the attached Barrett Hand to the object.
3. Ensure the trajectory (calculated in 2.) and stable grasp (calculated in 1.) are found and executed reliably within a strict time constraint (less than 5 seconds).

As was discovered through a preliminary literature review, the relative speed of obtaining feasible grasps can be increased through complexity reduction techniques. Table 2 summarizes the simplifying techniques I am concerned with, suggests how they might help speed up my overall algorithm and explains the main reason as to why I chose these techniques over alternatives.

Note that a simplified workspace {N} is considered first, with obstacles not being introduced until a later time. This subsequent introduction of obstacles into the configuration space of the robot is reasonable since the probabilistic approach to grasp planning in my solution also works in environments with obstacles [6] (albeit most likely at a lower level of efficiency in the absence of further optimization).

In order to verify the accuracy of my solution, I will test various grasps generated by my planner within GraspIt!: a physics-based simulator for robotic arms and hands. [1] The most successful strategies developed in simulation will then be tested on the physical Barrett Hand and WAM Arm system.

Table 2. Summary of complexity reduction techniques including my reasons for studying them

Technique	***Complexity Reduction***	***Ref.***	***Selection Reasoning***
Probabilistic Grasp Planning	*Reduces configuration search space*	[6]	*Impressive published results*
Simplify Object models	*Reduces feasible grasp search space*	[10]	*Versatile approximation scheme*
Approximate Friction Cone with Ellipsoid	*Reduces force-closure determination complexity*	[4]	*Most efficient force-closure approximation strategy*

The following sections will discuss each of the above techniques in greater detail.

Probabilistic Grasp Planning

As was discussed previously, the most successful solutions to the grasp planning problem have taken a probabilistic approach. Success is largely due to the intractable nature of exhaustively traversing through the search-space of many-DOF robotic systems. Published results for the *Grasp-RRT* algorithm by Vahrenkamp et al have been the most impressive. I will therefore implement the *Grasp-RRT* algorithm presented by the authors. For an introduction to the basic RRT algorithm on which *Grasp-RRT* is based, readers are directed to LaValle [15].

The *Grasp-RRT* algorithm comprises the following high level steps:

1. Grow an RRT from a starting location and expand it randomly.
2. Enter the Approach Trajectory Generation (ATG) step with some significant probability at each RRT Growing step.
3. Score the feasibility for each candidate grasp generated by each step of ATG.
4. If a feasible grasp is located by ATG, then the corresponding grasping trajectory is executed.

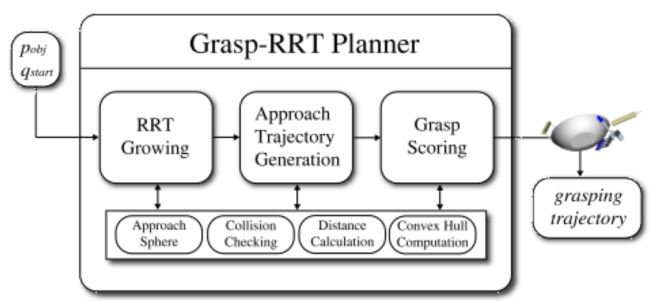

Figure 4. High level modules of the *Grasp-RRT* algorithm [6]

What makes *Grasp-RRT* interesting is that the search for a feasible grasp is done online and without the use of a database of potential grasps. This reduces the complexity of the algorithm to only test for feasible grasps that are approachable by the current potential trajectory plan generated via the RRT. In addition, the Inverse Kinematics problem and the generation of collision-free grasps are handled by the focused RRT-growing within the ATG step, eliminating the need to perform these necessary grasp-planning steps independently.

Algorithm 1: $GraspRRT(q_{start}, p_{obj})$

```
RRT.AddConfiguration(q_start);
while (!TimeOut()) do
    ExtendRandomly(RRT);
    if (rand() < p_SearchGraspPose) then
        n_grasp ← ApproachTrajectory(RRT, p_obj);
        if (ScoreGrasp(n_grasp) > score_min) then
            return BuildSolution(Grasp);
    end
end
```

Figure 5. The *Grasp-RRT* algorithm. Details of the Approach Trajectory Generation step not shown here for brevity. Readers are directed to [6] **for a full description.**

What Vahrenkamp et al neglected to include in their solution was the potential of performing bi-directional growth of the RRT, where search trees are grown simultaneously from both start and goal states. It is my hypothesis that incorporating bi-directional elements (as is presented in [7]) into the *Grasp-RRT* algorithm will increase the algorithm's execution efficiency.

There exists no shortage of grasp planning algorithms based on RRTs. I will also consider other RRT-based grasp planning schemes if it turns out that results obtained by Vahrenkamp et al are not reproducible. Solutions presented in [5-8] are some of the most promising I have discovered.

Convex Shape Assumption

In [10] the authors present guidelines for abstracting arbitrary objects into one of seven types of primitive shapes. Assuming simple convex models, the grasp planning procedure can be simplified by restricting grasp configurations in advance of quality evaluation. Figure 6 demonstrates this effect.

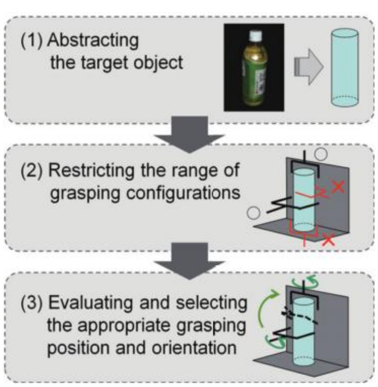

Figure 6. Once objects are abstracted into simple convex shapes, the resulting search for feasible grasps is greatly simplified. [10]

Therefore, it is safe to make the assumption that all objects to be picked up are convex (since the vast majority of everyday objects can be safely abstracted to convex polygons [10]). This assumption would speed up the algorithm by reducing the number of candidate grasps needed to find a stable grasp.

Friction Constraint Approximation

An interesting variation to the approximation of the composite friction cone is presented by Harada et al. [4] Instead of modeling the cone by a series of span vectors (as presented in section III), we can approximate the friction cone constraint by the intersection of an ellipsoid with a sphere (see Figure 7).

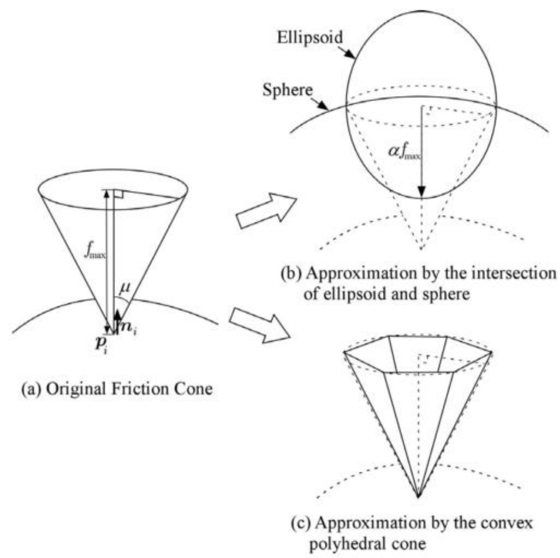

Figure 7. Harada et al present a different approximation to the friction cone: (a) is the original nonlinear representation; (b) is the approximation presented by Harada et al; and (c) represents the standard approximation of the friction cone. [4]

This is a rough approximation since very small forces cannot be modeled correctly. In addition, contact forces outside the actual friction cone may result from very large contact forces. It, nonetheless, does succeed at reducing the complexity in determining feasible force-closure grasps. It will be interesting to explore how this approximation scheme would affect the overall speed of my grasping system.

Tentative Timeline

The anticipated timeframe to bring this study to completion will be four months in length. Table 3 presents a rough timeline estimate that I will follow.

Table 3. Proposed Project Timeline

#	Details	Weeks
1	Prepare and test robotic system	two
2	Install and prepare GraspIt! simulator	one
3	Implement *Grasp-RRT* in GraspIt!	four
4	Integrate object abstraction step	one
5	Integrate friction cone approximation	one
6	Test on robot and compile results	three
	Total	**16**

V. Anticipated Results

In order to deem my solution a success, I will be comparing results obtained by the fastest known strategy in the grasp planning literature, published by Vahrenkamp et al (see Figure 8). They further partition the total amount of time to achieve a grasp into three distinct phases:

1. **Grasp Scoring**: time required to calculate a feasible grasp.
2. **Approach Movements**: speed at which the arm performed its required movements to bring the hand to the object.
3. **RRT Buildup**: amount of time the algorithm spent searching for a feasible path from initial to grasping pose of the end effector.

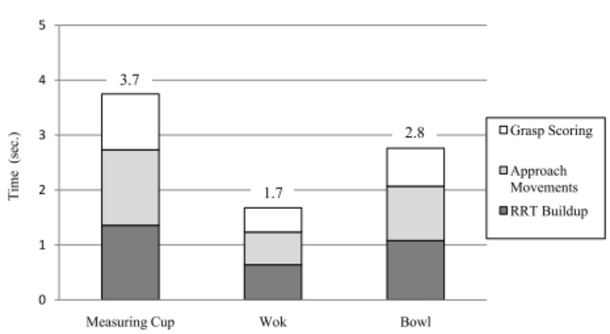

Figure 8. Profile of the Grasp-RRT algorithm searching for feasible grasps of three objects. [6]

Assuming their solution is reproducible, I should be able to match the results shown in Figure 8. It is then my hypothesis that improvements are possible through appropriate assumptions. For example, assuming simple convex shapes and efficient approximation of the friction cone would directly decrease the amount of time spent within the *Grasp Scoring* phase. Building the RRT simultaneously from both initial and goal positions would increase the efficiency of the *RRT Buildup* phase. Finally, modeling and attempting to control some of the higher order dynamics (e.g. velocity of the arm) has the potential of speeding up the *Approach Movements* phase.

VI. Conclusion and Future Work

An efficient strategy to the grasp-planning problem has been proposed in this paper. It is by a probabilistic approach, combined with object abstraction rules and simplified force-closure constraints that this will be accomplished. In future work, I hope to retain the same degree of efficiency within more complex environments containing obstacles.

Solutions that exist in the literature will be combined into a single unifying algorithm and tested within the grasping simulation environment GraspIt! and then tested on the physical Barrett Hand and WAM arm system.

There exist many more areas to explore that would not only increase the speed of execution of the grasping system, but allow the system to perform and appear more dexterous and human-like as well. For example, by introducing dynamics in our system model, we can gain greater control over the velocities and accelerations carried out by the arm. Preliminary work has already been started along this path with encouraging results. [12]

It has been suggested that dynamical models do not work well with probabilistic planning strategies such as RRTs. [16] However there has been some work in partitioning dynamic trajectories into a series of distinct movements (called Dynamic Motion Primitives, or DMPs) [17], which could allow for the use of dynamically modeled approach movements within probabilistic frameworks.

Developing fast and compliant hand-arm systems is a necessary step towards widespread adoption of robotic systems working in dynamic and human-centered environments.

References

[1] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *Robotics & Automation Magazine, IEEE*, vol. 11, no. 4, pp. 110–122, 2004.

[2] J. Reif, "Complexity of the mover's problem and generalizations," in *Proceedings of the 20th Annual Symposium on Foundations of Computer Science*, 1979, pp. 421-427.

[3] a T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, pp. 1824-1829.

[4] K. Harada, K. Kaneko, and F. Kanehiro, "Fast grasp planning for hand/arm systems based on convex model," *2008 IEEE International Conference on Robotics and Automation*, pp. 1162-1168, May. 2008.

[5] D. Berenson, R. Diankov, and J. Kuffner, "Grasp planning in complex scenes," *2007 7th IEEE-RAS International Conference on Humanoid Robots*, pp. 42-48, Nov. 2007.

[6] N. Vahrenkamp, M. Do, T. Asfour, and R. Dillmann, "Integrated grasp and motion planning," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 2883–2888.

[7] R. Diankov, N. Ratliff, D. Ferguson, S. Srinivasa, and J. Kuffner, "Bispace planning: Concurrent multi-space exploration," *Proceedings of Robotics: Science and Systems IV*, 2008.

[8] D. Bertram, J. Kuffner, R. Dillmann, and T. Asfour, "An integrated approach to inverse kinematics and path planning for redundant manipulators," *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pp. 1874-1879, 2006.

[9] A. Morales, P. Sanz, and A. del Pobil, "Vision-based three-finger grasp synthesis constrained by hand geometry," in *Robotics and Autonomous Systems*, 2006, pp. 1-32.

[10] N. Yamanobe and K. Nagata, "Grasp planning for everyday objects based on primitive shape representation for parallel jaw grippers," in *Robotics and Biomimetics (ROBIO), 2010 IEEE International Conference on*, 2010, pp. 1565–1570.

[11] C. Goldfeder, P. Allen, and C. Lackner, "Grasp planning via decomposition trees," in *IEEE Int. Conf. on Robotics and Automation*, 2007.

[12] N. Furukawa, a Namiki, S. Taku, and M. Ishikawa, "Dynamic regrasping using a high-speed multifingered hand and a high-speed vision system," *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, no. May, pp. 181-187, 2006.

[13] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer Handbook of Robotics*, vol. 4, no. 5, 2008, pp. 671-700.

[14] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC, 1994.

[15] S. LaValle, "Randomized kinodynamic planning," *The International Journal of Robotics*, vol. 20, no. 5, pp. 278-400, 2001.

[16] D. Berenson, "Constrained Manipulation Planning," 2011.

[17] S. Schaal, P. Mohajerian, and A. Ijspeert, "Dynamics systems vs. optimal control--a unifying view.," *Progress in brain research*, vol. 165, no. 1, pp. 425-45, Jan. 2007.