CHAPTER VI

A HEURISTIC FOR 3D FRICTIONLESS FORCE CLOSURE GRASP

6.1 Introduction

In this chapter, we propose a grasp analysis approach that aims to improve actual running time rather than to improve asymptotic time complexity on the problem of computing all force closure grasps. The approach presented here, and also in the next chapter for the frictional case, differs significantly from the algorithms presented in the last two chapters. In particular, we propose a heuristic test for a frictionless seven finger force closure grasp. The test is based on a necessary condition of positively spanning of wrenches. As a necessary condition, it ensures that any grasp that does not satisfy the condition will not achieve force closure while the force closure property of the satisfying grasp is undetermined. The benefit of the condition is that it can be evaluated much faster than existing force closure tests. Its superior speed leads us to the main approach of this chapter: heuristic approach in force closure testing. The idea is to reject a grasp not satisfying the condition as quickly as possible while satisfying grasps are passed to a complete test to determine whether they achieve force closure. Figure 6.1 outlines our new approach in grasp analysis.

Apart from the benefit in computing all force closure grasps, the filtering approach also finds its use in many grasp synthesis methods. Some recently proposed grasp synthesis approaches

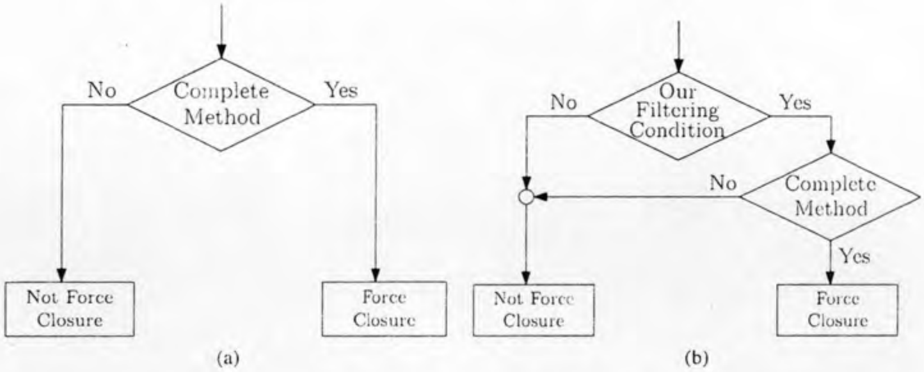

Figure 6.1: Flowcharts of the traditional approach compared with the proposed heuristic approach. (a) the traditional approach using only one complete test. (b) The filtered approach using a heuristic with a complete test.

are based on calling separable grasp analysis modules explicitly. We can find majority of such approach in grasp synthesis algorithms that adopt search based scheme. This scheme receive much attention especially in the case where input are discrete such as point could or discretized curved (see (Brost and Goldberg, 1996; Wang, 2000; Liu et al., 2004; Cornella and Suarez, 2006) for some examples). The underlying idea is to systematically search for an optimal grasping configuration in the finite space of such representation such that each candidate configuration is tested for desirability by a grasp analysis module. Different methods under this scheme vary by applying different search policy. A method based on hill climbing or branch and bound search is presented in (Zhu and Ding, 2006). Other optimizers are also adopted such as evolutionary computation as in (Hasegawa et al., 2000; Katada et al., 2001), or generate-and-test approach as in (Borst et al., 2003). Unlike traditional grasp synthesis framework, this search based scheme can conveniently take into account any grasp quality criteria. By developing a matching grasp analysis module, the user can compute a grasp that meets the requirement of their grasping task at hand without having to derive from scratch a new grasp synthesis method for the particular requirement. This advantage however arrives with the cost for assessing every candidate grasp by the grasp analysis module. By integrating our approach as a grasp analysis module, these synthesis scheme can take advantage from our approach.

The rest of this chapter is organized as follows. Section 6.2 introduces a necessary condition for force closure. Section 6.3 and Section 6.4 describe the implementation of the condition. Numerical example and comparison are given in Section 6.5. Finally, this chapter is summarized in Section 6.6.

6.2 Necessary Condition for Four Finger Force Closure Grasp

In this section, we present a necessary condition of frictionless seven finger force closure grasp. Our condition is based on the following lemma regarding the property of a positive span.

Lemma 6.1 *A necessary condition for a set of vectors to positively span $\mathbb{R}^n$ is that the projection of the vectors on any subspace $\mathbb{R}^{k<n}$ must positively span the subspace.*

It is clear that the lemma is necessary but not sufficient. Figure 6.2 illustrates some examples that satisfy Lemma 6.1. We apply Lemma 6.1 on wrench space. The wrench space consists of two distinct subspaces: the force space and the torque space, each is $\mathbb{R}^3$. We check whether the set of wrenches associated with the contact points positively span the force space and positively span the torque space. When the wrenches fail to positively span the force space or the torque space, the wrenches definitely fail to achieve force closure.

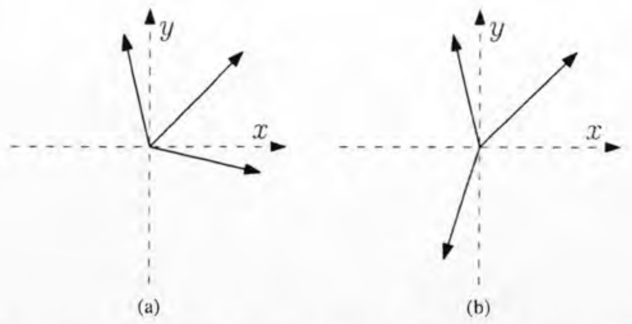

Figure 6.2: Example of vectors that satisfied Lemma 6.1. (a) the vectors do not positively span the space. (b) the vectors positively span the space.

It should be noted that any subspace can be used in Lemma 6.1. However, a larger subspace usually has a higher chance to catch more non-force closure grasps. This fact emphasizes preference to high dimensional subspace. Nevertheless, for the condition to be useful, it must also be computationally inexpensive. The force space and the torque space each describes an entity whose structure can be exploited. This allows us to derive an efficient computational implementation for both spaces.

Asserting whether the force space is positively spanned is straightforward. It amounts to testing whether the corresponding force vectors of the contact points positively span the force space. Positively spanning of the torque space is, however, more complicated to test. A torque varies according to the choice of the origin, although the force closure property as a whole is invariant to the relocation of the origin. It is possible that for some choice of the origin, a non force closure grasp may have its wrenches positively span the torque space. See for example the planar case in Figure 6.3. In the figure, it is clear that the contact points generate only counterclockwise torques when the origin is located at A. However, when the origin is located at B, the contact points generates both clockwise and counterclockwise torques, thus they positively span the torque space. The effect of this situation can be reduced by considering whether the torque space is positively spanned with respect to various choices of the origin of the workspace. Our condition for seven finger grasps considers seven choices of the origin, each of which is located at each contact point. Using a contact point as the origin has a benefit that the contact point being used will produce no torque. Hence, for each choice of the origin, only six torques have to be considered instead of seven torques when a general choice of origin were made.

In summary, our condition consists of eight tests: one considers the force space and the other seven consider the torque space. When any test fails, it is guaranteed that the grasp being tested does not achieve force closure. Let the contact points of the grasp be located at $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_7$ and let $\boldsymbol{f}_i$ and τ_i be the forces and the torques associated with the contact point at $\boldsymbol{p}_i$. Let $\mathcal{T}$ be

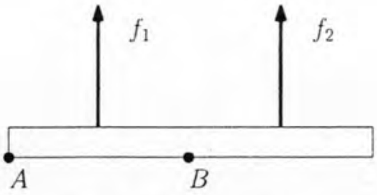

Figure 6.3: Example of forces.

the set of all torques. Our eight tests are listed as follows.

1. $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_7$ must positively span the force space.
2. Let the origin be located at $\boldsymbol{p}_1$; $\mathcal{T} - \{\tau_1\}$ must positively span the torque space.
3. Let the origin be located at $\boldsymbol{p}_2$; $\mathcal{T} - \{\tau_2\}$ must positively span the torque space.
4. Let the origin be located at $\boldsymbol{p}_3$; $\mathcal{T} - \{\tau_3\}$ must positively span the torque space.
5. Let the origin be located at $\boldsymbol{p}_4$; $\mathcal{T} - \{\tau_4\}$ must positively span the torque space.
6. Let the origin be located at $\boldsymbol{p}_5$; $\mathcal{T} - \{\tau_5\}$ must positively span the torque space.
7. Let the origin be located at $\boldsymbol{p}_6$; $\mathcal{T} - \{\tau_6\}$ must positively span the torque space.
8. Let the origin be located at $\boldsymbol{p}_7$; $\mathcal{T} - \{\tau_7\}$ must positively span the torque space.

6.3 $\mathbb{R}^3$-Positive Span of Force Components

This section introduces a method for testing whether seven 3D force vectors positively span the force space. Since the positively span problem is extensively studied in the literature, there are several existing algorithm for the task. For example, several methods in Section 3.3.1 can directly be used.

For this particular situation, our modified version of the method of Brost and Mason, which is proposed in Section 3.3.1.1 is used. From our preliminary experiment, this method is the fastest method for the task.

6.4 $\mathbb{R}^3$-Positive Span of Torque Components

In this section, a method for testing whether six 3D torque vectors positively span the torque space is presented. Similar to the case of force components, a generic positively spanning assertion method can directly be used. Our modified version of Brost and Mason is also chosen as our choice.

It should be noted that the problem determines positively spanning of six 3D wrenches. This problem is equivalent to the problem of 2D three finger force closure grasp discussed in Chapter 3. However, the method presented therein cannot be used directly because we cannot move the origin to ensure that two torques are on $\tau = 0$ plane. Even though Proposition 2.12 allows us to rotate all six torques such that two of of them are on $\tau = 0$, the cost of doing so is relatively high. Employing the BM algorithm is more efficient in this case.

6.5 Numerical Comparison

The presented condition is introduced as a filtering criteria. The criteria guarantees that an unsatisfied grasp does not achieve force closure while the force closure property of a satisfying grasp is undetermined. An additional complete method is therefore needed to test these satisfying grasps. We will refer to a non force closure grasp that satisfies the proposed condition as a false positive. Our condition sacrifices completeness in favor of efficiency in rejection. To benefit from the condition, the time additionally taken by our condition in the case of false positive must be offset by the time saved from the reduction of the number of complete method queries needed to be computed.

To help justify the benefit of our approach, we compare two grasp analysis frameworks: a canonical framework that uses a complete method to assert force closure and our filtering approach that uses the same complete method together with the presented condition as a filtering criteria. The two frameworks represent the diagrams in Figure 6.1.

An obviously preferable property of a filtering criteria is a large difference in the computational effort between that is used by the criteria and that is used by the complete method. The difference is the time to be saved by the approach when the criteria detects a non force closure grasp.

Another preferable property is high specificity of the test. A specificity is the ratio between the number of grasps not satisfying the condition and the total number non force closure grasps. A high specificity indicates that a large fraction of non force closure grasps are correctly identified by the filtering criteria, and the computational effort is saved by the difference between that of the criteria and that of the complete method.

Aforementioned speedup is then amplified by the number of actual non force closure grasps being tested. This number varies according to the situation that the condition is integrated into. We provide an empirical comparison in the scenario of computing all force closure grasps.

The comparison consists of several set of grasps. Each set contains $C_{25,7}$ grasps generated from each object shown in Figure 6.4. For the canonical framework, several complete force closure tests are chosen for comparison. The selected methods include ray shooting method of Liu (presented in Section 3.3.1.4), GJK algorithm (presented in Section 3.3.1.3), Q Distance method (presented in Section 3.3.1.5), and Quick Hull algorithm (presented in Section 3.3.1.6). For the filtering framework, we select Quick Hull algorithm as a complete test. The Quick Hull algorithm is selected because it yields most accuracy, according to our preliminary experiment. All implementations use double precision floating points. Additionally, we provide a reference method that employs arbitrary precision arithmetic using CGAL (Board, 2006). The reference method uses incremental convex hull algorithm presented in (Clarkson et al., 1993; Burnikel et al., 1994).

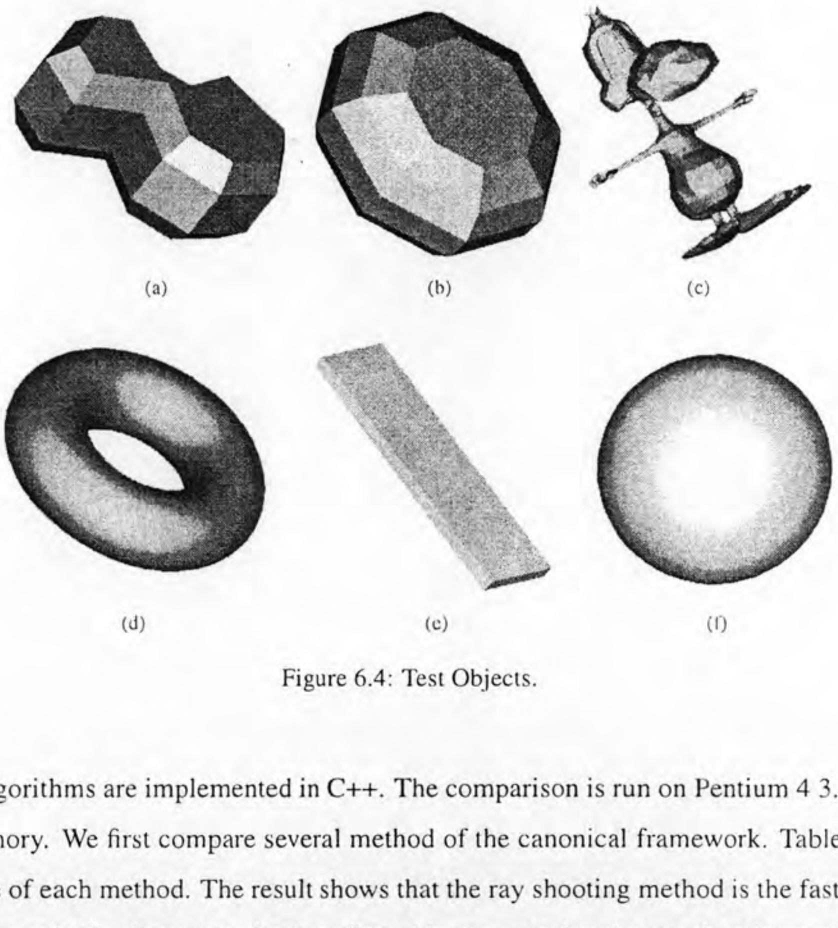

Figure 6.4: Test Objects.

All algorithms are implemented in C++. The comparison is run on Pentium 4 3.0GHz with 1GB of memory. We first compare several method of the canonical framework. Table 6.1 shows running time of each method. The result shows that the ray shooting method is the fastest method while Q distance is the slowest method. At first, this suggests that the ray shooting method should be used as a complete method in the filtering framework. However, the accuracy comparison, shown in Table 6.2, indicates that the ray shooting method exhibits noticeable number of errors. The best method in terms of accuracy is the Quick Hull method. This vindicates the preference of Quick Hull over other method in the filtering framework.

Table 6.1: Actual running time of the canonical framework.

Test Objs.	Running Time (s.)			
	Liu	GJK	QHull	ZW
(a)	9.34	19.55	53.55	106.09
(b)	9.54	12.45	55.05	73.98
(c)	9.53	11.33	55.32	76.73
(d)	9.49	23.75	56.93	87.80
(e)	8.49	31.25	43.24	179.03
(f)	10.27	79.56	55.92	88.43
Avg	9.44	29.65	53.33	102.01

Table 6.2: Accuracy comparison of complete methods

Test Objs.	#Force Closure Grasp	#Fault			
		Liu	GJK	QHull	ZW
(a)	2,606	1,202	48,192	-	424
(b)	2,867	256	224	-	-
(c)	4,913	-	80	-	-
(d)	3,554	21,208	24,566	1,102	2,810
(e)	-	45,416	178,185	-	363
(f)	12,066	2,663	230,212	2,494	2,494
Avg	4,334	11,791	80,243	599	1,015

The comparison between the filtering framework and the canonical framework is shown in Table 6.3. The second and the third column show the actual running time of the canonical framework and the filtering framework. Speedup factor is given in the forth column. The fifth column shows the number of grasps that satisfy our condition but are rejected by the Quick Hull algorithm, i.e., the number of false positive with respect to the Quick Hull algorithm. The specificity of the test, also with respect to the Quick Hull algorithm, is shown in the sixth column.

Table 6.3 shows that both methods yield the identical solution. The filtering approach provides the solution provides approximately 10 times faster. A further investigate shows that

Table 6.3: Comparison between the canonical framework and the filtering framework.

Objects	Time (seconds)		Speedup Factor	#False Positive	Specificity
	Unfiltered	Filtered			
(a)	53.55	4.52	11.84	14,860	0.970
(b)	55.05	3.99	13.81	13,699	0.972
(c)	55.32	4.08	13.57	25,802	0.949
(d)	56.93	4.40	12.94	40,177	0.922
(e)	43.24	1.50	28.81	-	1.000
(f)	55.92	5.55	10.08	58,938	0.888
Avg	53.33	4.01	15.18	25,579.33	0.950

the time used per query of our condition and that of the Quick Hull algorithm is approximately 0.008ms and 0.111ms, respectively. This indicates that, for each true negative solution, a running time is reduced to approximately 7.51%. In the case of false positive, the running time is increased to 107.51%. From the average specificity shown in Table 7.1, approximately 95.4% of the negative solutions is correctly identified by our condition.

It is clear from the result that our condition exhibits the favorable property of a good filtering criteria. This benefit is more visible when the set of queries contains a large number of non force closure grasps. For example, the box object (e) has no force closure grasp and our condition can detect all non force closure grasp. The speedup in this case is maximum. The object (f) has a lots of force closure grasps. Thus, the speedup in this case is least visible. Nevertheless, it is clear that the filtering approach provides significant benefit.

6.6 Summary

In this chapter we propose a necessary condition for force closure property of a hard frictionless contact points. We also provide an implementation of the condition which utilizes the structure of force and torque of a hard contact. Specifically, we employ the BM algorithm in determining whether seven or six 3D vectors positively span $\mathbb{R}^3$.

The benefit of the method is that it can be computed using relatively low computational effort. This allows the condition to be used as a filtering criteria which a grasp must be satisfied before a complete force closure assertion method is applied. From the numerical example. This approach could greatly speeds up the running time in the case of multiple force closure queries which proven to be useful in many situations.