On-Line Interactive Dexterous Grasping

Matei T. Ciocarlie and Peter K. Allen

Columbia University, New York, USA
{cmatei,allen}@cs.columbia.edu

Abstract. In this paper we describe a system that combines human input and automatic grasp planning for controlling an artificial hand, with applications in the area of hand neuroprosthetics. We consider the case where a user attempts to grasp an object using a robotic hand, but has no direct control over the hand posture. An automated grasp planner searches for stable grasps of the target object and shapes the hand accordingly, allowing the user to successfully complete the task. We rely on two methods for achieving the computational rates required for effective user interaction: first, grasp planning is performed in a hand posture subspace of highly reduced dimensionality; second, our system uses real-time input provided by the human user, further simplifying the search for stable grasps to the point where solutions can be found at interactive rates. We demonstrate our approach on a number of different hand models and target objects, in both real and virtual environments.

Keywords: dexterous grasping, human-machine interaction, hand prosthetics.

1 Introduction

Current research in the field of neural prosthetics is advancing rapidly, as it benefits from new insights into both human and robotic grasping. Prosthetic devices controlled by cortical output combine a degree of human control with artificial hardware and algorithms; while this direction of research appears to be very promising, it also poses new problems regarding the interface between the biological and artificial components.

In this paper we focus on the dexterous grasping problem. This can be formalized as the task of, given a robotic hand and a target object, determining an appropriate wrist position and hand posture for a stable grasp. We consider the case where the position of the hand relative to the target object can be determined directly by a human user; recent results [1] have shown that three-dimensional positioning of the tooltip of a robotic arm can be achieved under neural control, therefore this assumption can also hold in situations where the ability to actuate the arm directly is lost. However, controlling hand posture has proven to be significantly more difficult. A number of possible approaches are described in the literature, including electromyography (EMG) [2] and cortical implants [3]. These studies have shown success in decoding a limited number of information channels, therefore controlling a highly dexterous hand during

M. Ferre (Ed.): EuroHaptics 2008, LNCS 5024, pp. 104–113, 2008.

grasping remains an open and challenging problem. Current approaches, such as the one presented in [4], assume the user will be able to select from a small set of pre-grasp shapes, and rely on the passive mechanical adaptability of the hand to successfully complete the grasp.

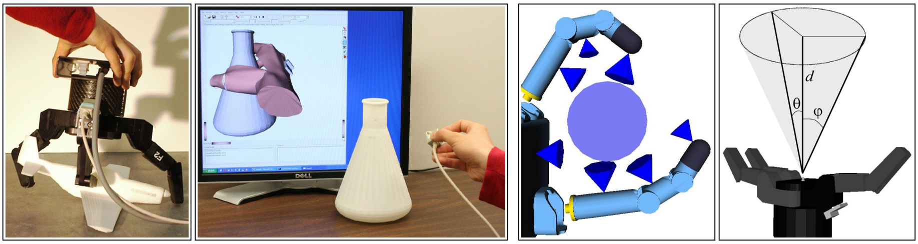
Our goal in this paper is to provide an automated grasp planning algorithm that compensates for the missing information that a human user cannot provide directly and enables the use of a wide range of hand postures. We describe a system that acts as an interface between a human operator and an artificial hand used for grasping tasks. This system enables the user to achieve stable grasps by controlling the position and orientation of the wrist, but with no direct control over finger posture. An important requirement regards computational efficiency: the system's response should be fast enough to allow for interactive operation. By satisfying this constraint we can effectively close the feedback loop, allowing the human user to not only set initial guidelines for the automated system, but also react to its behavior and assist in the successful completion of the task.

For comprehensive reviews concerning fully autonomous grasp synthesis for robotic hands we refer the reader to [5,6]. The difficulty in designing effective algorithms generally stems from the complexity of both the hand configuration space and the contact space of the grasped object [7]. It is important to note that our approach requires knowledge of the target object geometry, as well as its initial position relative to the hand. For application in unstructured environments, it therefore requires a complementary system for object recognition and localization such as [8]. However, the process of training an operator to use an artificial hand (such as a neural prosthetic) usually takes place in a controlled setting where this information is directly available. Our system can simplify this training process by enabling successful grasps even with limited control information, allowing the user to slowly develop better control abilities.

1.1 System Overview

The central component of our system is the grasp planning algorithm, which runs on the platform provided by the *GraspIt!* simulator for robotic grasping [9]. The simulator receives user input and sends it to the grasp planner which processes it and outputs potential grasps, which are in turn used to generate commands for the robotic hand. Even though the grasp planner runs in a simulated environment, the results can be applied to a real robotic hand, allowing the user to interact with the hand directly and use it to pick up surrounding objects. Later in this paper we will present our results using the Barrett robotic hand. We have also tested our method on a range of more complex hand designs (including the DLR hand, the Robonaut hand as well as human hand model) using the virtual environment in *GraspIt!*. In order to successfully complete a grasping task, our method proceeds through the following three stages:

- human input recording: when a Barrett robotic hand is used for grasping tasks, the user can hold the hand and approach the target object (Fig. 1a). The position of the hand relative to the target is tracked using a Flock of Birds



a) b) c) d)

Fig. 1. a) real Barrett hand used to pick up an object; **b)** dexterous grasping using a complex hand model in a simulated environment; **c)** potential contact forces considered for a given hand posture. Each cone shows the magnitude of the potential forces applied by a contact, which is scaled depending on the distance to the object. **d)** by re-parameterizing the spatial component of the search using variables ϕ, θ and d, we constrain it to a conical region around the approach direction specified by the user.

(Ascension Corp., VA) magnetic tracker. When operating in a simulated environment, the user can change the position of the virtual wrist by directly manipulating the magnetic tracker (Fig. 1b). In both cases, the user has no control over finger posture except for a binary "close all fingers" command.

- **grasp planning:** the automated grasp planner attempts to find a stable grasp, considering the current approach direction given by the user. The details of the planning algorithm are presented in section 2, while user interaction is discussed in section 3. Once at least one acceptable grasp is found, the system can advance to the next stage.
- **grasp execution:** the system positions the fingers of the hand according to the best grasp currently available. This allows the user to react, by either continuing to approach the object along the same direction or moving the hand to a new position in order to search for a better grasp. Once the user is satisfied with the hand posture, he can issue the "close all fingers" command and complete the grasp. An extensive set of grasping tasks examples is presented in section 4.

2 Low-Dimensional Grasp Planning

In general, automatic grasp synthesis can be thought of the task of finding the combination of hand posture (intrinsic degrees of freedom, or DOF's) and position (extrinsic DOF's) that produces a stable grasp, according to a given grasp quality metric. From this perspective, it can be approached as an optimization problem, seeking to maximize the value of the grasp quality Q expressed as a function over a high-dimensional domain:

$$Q = f(\boldsymbol{p}, \boldsymbol{w}) \tag{1}$$

If d is the number of intrinsic hand DOF's, $\boldsymbol{p} \in \mathcal{R}^d$ represents the hand posture and $\boldsymbol{w} \in \mathcal{R}^6$ contains the position and orientation of the wrist. In this section, we address the intrinsic component $\boldsymbol{p}$ of the quality function, which is entirely controlled by the automated grasp planner; the wrist component $\boldsymbol{w}$, which is directly affected by the human user, is discussed in the following section.

The traditional form for specifying a hand posture is to set a value for each individual DOF of the hand. For complex hands, such as the human hand modeled in our study with $d = 20$, the high dimensionality of the posture space makes direct searches for good grasps intractable. In order to meet the interactivity constraint of our system, we perform the grasp planning task in a subspace of highly reduced dimensionality. Our approach is based on the results of Santello *et al.* [10], who have shown that the range of postures that humans use in everyday grasping exhibits significant clustering in the d-dimensional DOF space. However, while human grasping subspaces can be determined through user studies, defining similar subspaces for non-anthropomorphic robotic hands is an interesting open problem. In previous work [11], we have applied this concept to five hand models, using the results of Santello *et al.* for the anthropomorphic models (such as the DLR and Robonaut hands) and empirically derived subspaces for the non-anthropomorphic ones (such as the Barrett hand). Brown and Asada [12] also present an anthropomorphic robotic hand with a low-dimensional control system along directions similar to those presented in [10]. A discussion on different dimensionality reduction techniques for hand control is presented by Peters and Jenkins [13], applied to teleoperation data on a Robonaut hand.

In our work, we consider a hand posture subspace defined by a number of d-dimensional basis vectors called **eigengrasps**; the implication is that these vectors can be linearly combined to closely approximate most common grasping positions. By choosing a basis comprising b eigengrasps, a hand posture $\boldsymbol{p}$ placed in the subspace defined by this basis is uniquely defined by the vector $\boldsymbol{a} \in \mathcal{R}^b$ containing the amplitudes along each subspace axis. In previous work [14], we have discussed the feasibility of finding good grasps for dexterous hands by searching a subspace defined by two eigengrasps. This implies a significant dimensionality reduction of the grasp quality function domain, which can be expressed as

$$Q = f(\boldsymbol{a}, \boldsymbol{w}), \quad \boldsymbol{a} \in \mathcal{R}^2 \tag{2}$$

However, this low-dimensional subspace is only useful as long as it contains the hand postures needed for stable grasps of a large variety of objects. The results presented in [14] show that, in general, postures where the hand conforms *perfectly* to the surface of the target can not be found in eigengrasp space. However, by searching this subspace we can usually find a posture that is *very close* to a desired grasp. The eigengrasp space can therefore be thought of as a pre-grasp, or planning space: the best pre-grasps found in this subspace have a good chance of producing stable grasps by simply closing each finger until motion is stopped by contact with the object. This suggests a two-stage grasp planning algorithm, detailed in the remainder of this section: the first stage searches the low-dimensional eigengrasp subspace, while the second stage tests the resulting pre-grasps and outputs the best solutions.

2.1 Grasp Quality Function and Optimization Algorithm

Most grasp quality metrics that have been proposed in the literature are based on the locations of the contacts between the hand and the target object. Our context is somewhat different: we need a metric that can asses the quality of a *pre-grasp*, where the hand is very close, but not in contact with the target. One possibility would be to simply close the fingers and compute the resulting contacts; however, this is an expensive process that can only be applied at interactive rates to a few pre-grasp postures. We therefore propose an alternative quality function which is fast to compute, but only provides an approximation of the final grasp quality. Our formulation is a modified version of the Ferrari-Canny metric [15] intended to assess the *potential* quality of a posture.

For a given hand shape, we assume that the hand can apply potential forces to the target object through a number of pre-defined contact locations. The potential forces are scaled according to the distance between each finger and the target (Fig. 1c). We then compute the total wrench space that can be applied by the scaled potential forces, and apply the Ferrari-Canny metric [15] on the resulting wrench space. The intuition behind this approach is that this formulation rewards postures that bring the fingers as close as possible to the object, but are also likely to create enveloping grasps with stable wrench spaces and no weak points. This quality function is therefore used to guide the search towards promising grasping postures and produce candidate pre-grasps.

Even when using a two-dimensional subspace to constrain the posture of the hand, the additional variables needed to specify the position of the wrist make it intractable to perform an exhaustive search under the time constraints imposed by our application. We therefore perform the search using the simulated annealing [16] algorithm; the reduced dimensionality of the search domain allows this algorithm to find effective hand postures at interactive rates (computation times will be presented in section 4). We have implemented this algorithm using the *GraspIt!* simulation engine. For each state generated during the annealing schedule, *GraspIt!* uses forward kinematics to place the hand model in the appropriate posture and checks for collisions against the object to be grasped as well as other obstacles. If the state is found to be legal, we proceed to compute its associated quality function value and continue the simulated annealing optimization process. The hand postures with the highest quality values found during the search are buffered and sent to the second stage of the planning process.

2.2 Final Grasp Computation

For each candidate pre-grasp resulting from the previous stage, we use the contact detection engine within *GraspIt!* to compute the final grasp that results by closing the fingers on the object. Once the contacts between the hand and the object have been determined, we compute the exact quality value of the final grasp by applying the Ferrari-Canny metric in its original form presented in [15]. If the grasp is found to have form-closure, it is presented as a potential solution, along with its associated quality value, to the next component of the system, which is responsible for interaction with the human user.

When computing the final grasping posture resulting from a candidate pre-grasp, we take into account specific mechanical properties of the hand, such as passive adaptation to the shape of the target and soft fingertips. A number of robotic hands, such as the Barrett Hand, the SDM Hand [17] and the Cyber-Hand [18] rely on passive mechanical adaptation, as it significantly increases grasp stability without increasing the complexity of the control mechanisms. All of the results involving the Barrett hand presented in this paper take into account its adaptive actuation mechanism which allows distal joints to close even when proximal joints controlled by the same motor have been stopped due to contact. The ability to create stable, encompassing grasps with subsets of fingers is also increased by using soft fingertips that deform during contact and apply a larger space of frictional forces and moments than their rigid counterparts. In order to take into account such effects, we use a fast analytical model for soft finger contacts that we have introduced in previous work [19]. This enables the use of rubber-coated fingertips for our robotic hands, without compromising the accuracy of the grasp quality computations.

In our implementation, the two planning phases described in this section (simulated annealing search for pre-grasps and final grasp testing) run in separate threads. As soon as a candidate pre-grasp is found, it is queued for testing, but the search for new candidates continues independently of the testing phase. Also, candidate pre-grasps are independent of each other, and can be tested simultaneously. This parallelism allows us to take advantage of the current evolution in multi-core architectures, largely available on standard desktop PCs.

3 On-Line Human Interaction

In order to uniquely define a grasp, a given finger posture has to be associated with a hand position relative to the target object. In general, 3 variables are needed to specify the position of the palm, and 3 more for its orientation. In the context of our application, we expect the user to specify a desired approach direction to the target; however, this does not fully eliminate the spatial component of the grasp planning search. First, it is not practical to wait until the user has brought the palm into a final grasping position before starting the search for an appropriate finger posture, as this behavior would decrease the interactivity of the system. Rather, it is preferable to start the search early, and attempt to predict where the user intends to place the palm. Second, this prediction allows the system to offer feedback to the user: as soon as an anticipated grasp is found, the grasp planner can shape the fingers accordingly. This allows the users to decide if the grasp is satisfactory and continue to approach the target, or choose another approach direction if the system is unable to shape the fingers.

This behavior can be implemented efficiently by re-parameterizing the spatial component of the grasp planner to a total of 3 variables, comprising an approach distance and two angles, as shown in Fig. 1d. By adding these 3 variables to the 2 eigengrasp amplitudes that define hand posture, the automated grasp planner will cover a conical region around the current approach direction. We can now provide a complete step-by-step walkthrough of a grasping task, as it proceeds through the following stages:

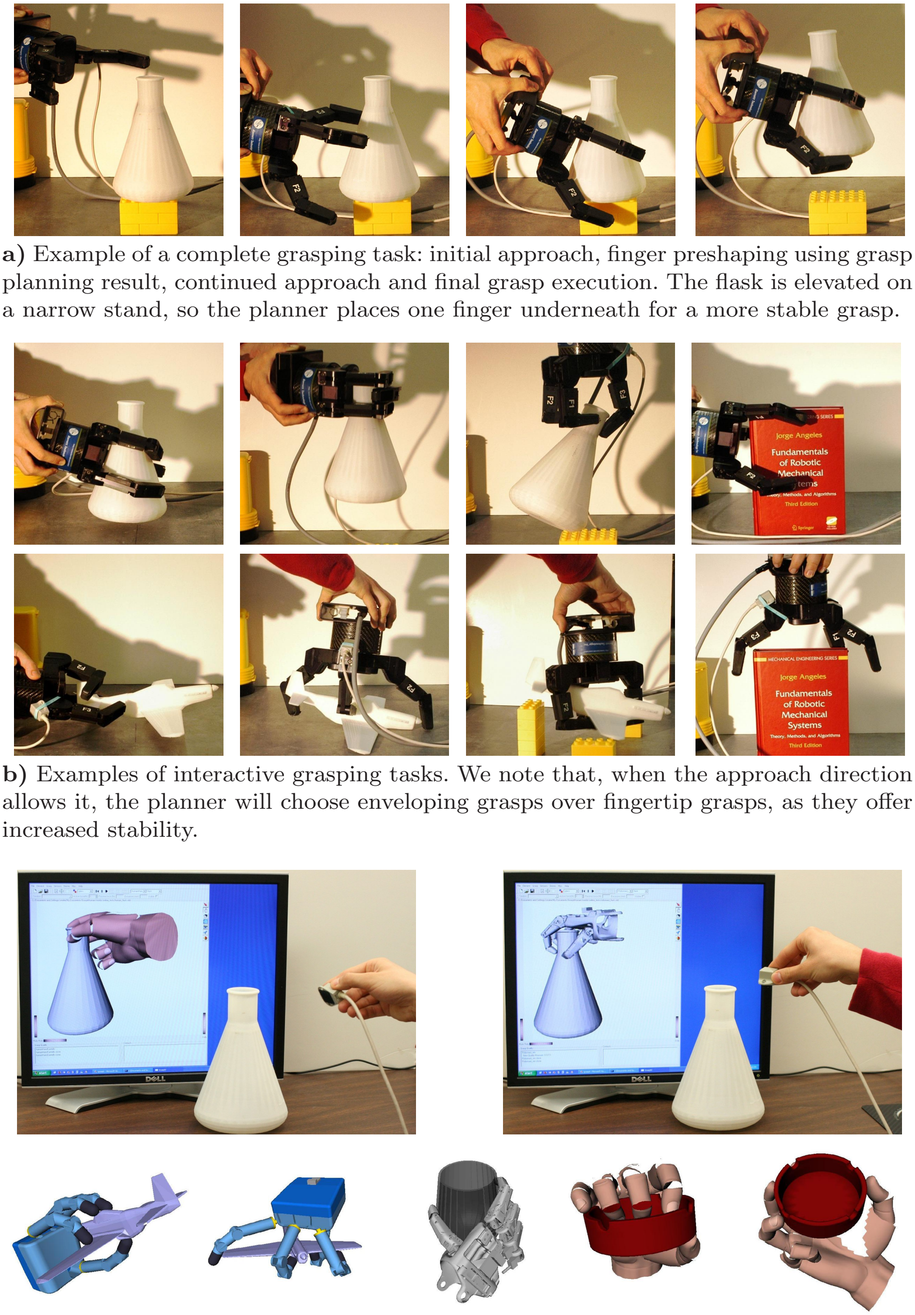
- as the user approaches the target object, the grasp planner searches for a good grasp in a cone-shaped area around the given approach direction; when a solution is found, it is used to set the hand posture, allowing the user to react.
- the planner continuously attempts to improve the current result, by finding new grasps that either have a better quality metric or are closer to the current position established by the user.
- if the planner is unable to find a grasp in the current search area, or the user is not satisfied with the resulting hand posture, the user can reposition the hand and attempt to grasp a different part of the target object.
- if the user is satisfied with the hand posture, he continues along the current approach direction. As the real hand position approaches the target grasp, the fingers are gradually closed around the object. This allows the user to closely predict where the object will be touched. Finally, the user issues a "close all fingers" command and completes the grasping task.

4 Dexterous Grasping Examples

Fig. 2 presents the application of our method using the Barrett Hand in a real environment, as well as the DLR hand, the Robonaut hand and a human hand model in a simulated environment. In most cases, the images show only the final grasp applied by the user; due to space constraints we are unable to include images showing the evolution of the grasping task from approach direction, pre-grasp and final grasp. In order to better evaluate the interactive nature of our application, a movie showing a number of complete examples can be found at `http://www.cs.columbia.edu/~cmatei/eh08`.

For any given grasping task, the exact computational effort required to find a stable grasp depends on the complexity of the hand and target object, as well as the approach direction chosen by the user. On average, the first stage of the grasp planning algorithm processes approx. 1000 hand postures per second, while the second testing phase, running in parallel, can evaluate approx. 20 candidate pre-grasps per second. In most cases, solution grasps are found at interactive rates: in the example presented in Fig. 2a, the grasp planner found 8 stable grasps in 13.6 seconds of computation on a commodity 2.13GHz Intel Core2 CPU. These are representative numbers for our system, which generally requires less than 2 seconds to find a solution grasp for a new approach direction.

The ability of the system to allow for successful task completion in a short time is more difficult to quantify, as it also depends on how well the user reacts to the behavior of the automated components. All the results presented in Fig. 2 were obtained at interactive rates, usually requiring between 5 and 15 seconds from first approach to final grasp execution. For the more difficult tasks, taking up to 30 seconds to complete, we found two main reasons that led to the increased execution time: either the planner repeatedly failed to find solution grasps for


F2
F1
F3
Jorge Angeles
Fundamentals of Robotic Mechanical Systems
Theory, Methods, and Algorithms
Third Edition
MECHANICAL ENGINEERING SERIES
Springer
DELL

a) Example of a complete grasping task: initial approach, finger preshaping using grasp planning result, continued approach and final grasp execution. The flask is elevated on a narrow stand, so the planner places one finger underneath for a more stable grasp.

b) Examples of interactive grasping tasks. We note that, when the approach direction allows it, the planner will choose enveloping grasps over fingertip grasps, as they offer increased stability.

c) Form-closure grasps using dexterous hands in a simulated environment. Top row images also show the user providing the approach direction via a magnetic tracker.

Fig. 2. Examples of grasping tasks in both real and simulated environments

selected approach directions, or the human user could not interpret some of the finger postures selected by the planner and had to attempt different grasps. These cases represent a small minority of our tests and examples; however, the tests were performed by well-trained users familiar with the inner workings of the planning algorithm. As a next development step, we intend to test our system in user studies with untrained subjects; the results will allow for a more precise quantification of both the robustness and the efficiency of the system.

5 Conclusions

In this paper we have presented a novel method for human-robot interaction applied to dexterous grasping. The demand for such interfaces stems from the area of hand neuroprosthetics, where a human user must interact with an artificial limb using limited control methods. Our main contributions include:

- an on-line grasp planner that can find stable grasps of a target object fast enough for interaction with a human user. This automated component has exclusive control over the hand posture and continuously searches for solution grasps. In order to achieve interactive rates, this search is separated into two processes: the first one finds a small number of optimized pre-grasps in a hand posture sub-space of highly reduced dimensionality, while the second one processes these results and computes the quality of the final grasping positions.
- an interaction method that allows the user to provide input to the grasp planner, effectively guiding it towards grasping the intended area of the target. This input simplifies the hand position and orientation components of the automated planner, which only searches for good grasps in a small region around the approach direction desired by the user.
- a complete system that allows a human user to perform dexterous grasping tasks with an artificial hand without having direct control over the posture of the hand. We have tested this system in both real and simulated environments and have presented results involving a number of different hand models and objects.

In future work, we intend to further explore the interplay between human and automatic control for dexterous grasping. In the implementation presented here we assume that the user has complete control over hand position, but none over finger posture. However, our method has a native ability to adapt to changes to these premises. As the automated planner searches a neighborhood around the hand position given by the user, it can succeed in finding stable grasps even if human position input is incomplete or unreliable (as in the case where the arm is also under indirect neural control). We will also consider the case where limited hand posture input is available from the user: since our search algorithm operates in a low-dimensional subspace rather than on individual joint values, the final hand posture can be significantly influenced using only one (or few) channels of information. Furthermore, the user-specified posture does not need to match the grasped object perfectly, as the planner will automatically refine it as much as possible. We believe that this approach will prove a valuable component for building robust, yet versatile and effective hand prostheses.

Acknowledgments. This work was funded in part by NIH BRP grant 1RO1 NS 050256-01A2.

References

1. Taylor, D., Tillery, S., Schwartz, A.: Direct cortical control of 3D neuroprosthetic devices. Science 296, 1829–1832 (2002)
2. Zecca, M., Micera, S., Carrozza, M.C., Dario, P.: Control of multifunctional prosthetic hands by processing the electromyographic signal. Critical Reviews in Biomedical Engineering 30, 459–485 (2002)
3. Taylor, D., Tillery, S., Schwartz, A.: Information conveyed through brain control: Cursor versus robot. IEEE Trans. Neural Syst. Rehab Eng. 1, 195–199 (2003)
4. Cipriani, C., Zaccone, F., Stellin, G., Beccai, L., Cappiello, G., Carrozza, M., Dario, P.: Closed-loop controller for a bio-inspired multi-fingered underactuated prosthesis. In: IEEE Intl. Conf. on Robotics and Automation, pp. 2111–2116 (2006)
5. Shimoga, K.B.: Robot grasp synthesis algorithms: a survey. Intl. J. of Robotics Research 15, 230–266 (1996)
6. Bicchi, A., Kumar, V.: Robotic grasping and contact: A review. In: IEEE Intl. Conf. on Robotics and Automation, pp. 348–353 (2000)
7. Roa, M., Suarez, R.: Geometrical approach for grasp synthesis on discretized 3d objects. In: IEEE-RSJ Intl. Conf. on Intelligent Robots and Systems (2007)
8. Kragic, D., Miller, A., Allen, P.: Real-time tracking meets online planning. In: IEEE Intl. Conf. on Robotics and Automation (2001)
9. Miller, A., Allen, P.K.: GraspIt!: A versatile simulator for robotic grasping. IEEE Robotics and Automation Magazine 11, 110–122 (2004)
10. Santello, M., Flanders, M., Soechting, J.F.: Postural hand synergies for tool use. Journal of Neuroscience 18(23), 10 105–10 110 (1998)
11. Ciocarlie, M., Goldfeder, C., Allen, P.: Dexterous grasping via eigengrasps: A low-dimensional approach to a high-complexity problem. In: Robotics: Science and Systems Manipulation Workshop - Sensing and Adapting to the Real World (2007)
12. Brown, C., Asada, H.: Inter-finger coordination and postural synergies in robot hands via mechanical implementation of principal components analysis. In: IEEE-RAS Intl. Conf. on Intelligent Robots and Systems, pp. 2877–2882 (2007)
13. Peters, R.A., Jenkins, O.C.: Uncovering manifold structures in Robonaut's sensory-data state space. In: IEEE-RAS Intl. Conf. on Humanoid Robots (2005)
14. Ciocarlie, M., Coldfeder, C., Allen, P.: Dimensionality reduction for hand-independent dexterous robotic grasping. In: IEEE-RSJ Intl. Conf. on Intelligent Robots and Systems (2007)
15. Ferrari, C., Canny, J.: Planning optimal grasps. In: IEEE Intl. Conf. on Robotics and Automation, pp. 2290–2295 (1992)
16. Ingber, L.: Very fast simulated re-annealing. J. Mathl. Comput. Modelling 12(8), 967–973 (1989)
17. Dollar, A., Howe, R.: Simple, robust autonomous grasping in unstructured environments. In: IEEE Intl. Conf. on Robotics and Automation (2007)
18. Carrozza, M.C., Cappiello, G., Micera, S., Edin, B.B., Beccai, L., Cipriani, C.: Design of a cybernetic hand for perception and action. Biol. Cybern. 95(6), 629–644 (2006)
19. Ciocarlie, M., Lackner, C., Allen, P.: Soft finger model with adaptive contact geometry for grasping and manipulation tasks. In: Joint Eurohaptics Conf. and IEEE Symp. on Haptic Interfaces (2007)