The Planning and Control of Robot Dextrous Manipulation [*]

Li Han [†]   Zexiang Li [‡]   Jeff C. Trinkle [§]   Zhiqiang Qin   Shilong Jiang [¶]

Abstract

Dextrous manipulation is a fundamental problem in the study of multifingered robotic hands. Given a robotic hand and an object to be manipulated by the hand in an environment filled with obstacles, the main objectives of dextrous manipulation are to have the hand grasp the object and transfer it from a start configuration to a goal configuration without collision. To fulfill such a task in general, we will need: (a) a manipulation planner to generate a feasible path for the hand; and (b) a controller to implement the planned path. In this overview paper, we define the manipulation planning problem and present a unified Control System Architecture for Multifingered Manipulation (CoSAM^2). By incorporating the various kinematic and static relationships of a multifingered robotic hand system with proper sensory data inputs at different stages, CoSAM^2 achieves the various objectives of dextrous manipulation. Theoretical background of the control system design along with real-time experimental results are described.

1 Introduction

Dextrous manipulation is a fundamental problem in the study of multifingered robotic hands. Given a robotic hand and an object to be manipulated by the hand in an environment filled with obstacles, the main objectives of dextrous manipulation are to have the hand grasp the object and transfer it from a start configuration to a goal configuration while simultaneously

[†]L. Han, Dept. of Computer Science, Texas A&M Univ., College Station, TX 77843-3112, USA. lihan@cs.tamu.edu.

[‡]Z. Li, Dept. of Electrical and Electronic Eng., HongKong Univ. of Science and Technology, ClearWater Bay, Kowloon, HongKong. eezxli@ee.ust.hk.

[§]J.C. Trinkle, Intellligent Systems Principle Dept., Sandia National Laboratory, Albuquerque, NM 87185-1004, USA. trink@cs.tamu.edu.

[¶]Z. Qin and S. Jiang, Dept. of Electrical and Electronic Eng., HongKong Univ. of Science and Technology, ClearWater Bay, Kowloon, HongKong. eeqin, eejiangl@ee.ust.hk.
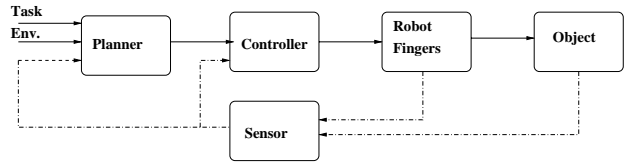
Figure 1: Dextrous Manipulation System

supplying the desired object wrench, avoiding collision, and respecting the position and actuator force limits of the joints. To fulfill such a task in general, we will need: (a) a manipulation planner to generate a feasible path for the hand; and (b) a controller to implement the planned path, based on the system model and sensor information, as shown in figure 1.

Planning a feasible path is an extremely difficult problem which involves complex systems of constraints imposed on a high-dimensional configuration space. In the system diagram shown in figure 1, it is the planner's job to generate a feasible path from a high-level specification of the task, a model of the environment, and possibly sensor measurement. The planner may also have to modify the path in the event that the sensors detect unmodeled obstacles or other unexpected, but important features of the environment.

The controller's major task is to provide the desired finger joint efforts and movements so as to make the object and contact move properly to comply with the physical law and achieve the desired motion. The main problem here is that only the robot fingers can be directly controlled by the robotic system while the manipulation objective and planned path are naturally specified in terms of object wrench, object and grasp configurations. Furthermore, the controller need to be robust to deal with the uncertainty arising from the system model, sensor noise and actuator inaccuracy.

Over the years, significant strides have been made in realizing features of multifingered manipulation. The kinematics and statics underlying a multifingered manipulation system have been identified and thoroughly
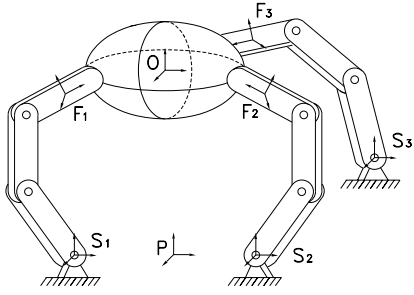
Figure 2: A multifingered hand manipulation system

analyzed [2, 4, 5, 9, 11, 13, 19, 22, 23, 24, 26, 28, 29, 30, 31, 32]. General framework and efficient algorithms [6, 14, 15, 21] for the collision-free path planning problem have been developed. Dextrous manipulation with rolling contacts and finger gaiting has been investigated in [10, 16, 23] along with several useful algorithms for finger motion planning. Coordinated control and compliance control algorithms for multifingered manipulation with either fixed point of contact or rolling contact have been extensively studied [7, 17, 25]. Tactile, force/torque and vision sensors have been developed or utilized to sense contact location and contact forces [1, 8, 27]. Furthermore, many articulated multifingered robotic hands have been developed as research tools to study dextrous manipulation.

Despite the enormous amount of research activity on multifingered robotic hands, many problems remain open and need to be solved before the robotic hands can reach the dexterity and functionality of the human hands. In this overview paper, we define the manipulation planning problem and present a unified Control System Architecture for Multifingered Manipulation (CoSAM^2). By incorporating the various kinematic and static relationships of a multifingered robotic hand system with proper sensory data inputs at different stages, CoSAM^2 achieves the various objectives of dextrous manipulation. Theoretical background of the control system design along with real-time experimental results are described.

2 Mathematic Preview

Follow the notation in[24], this section will briefly review the kinematics and statics underlying to multifingered manipulation.

2.1 Statics

Consider a k-fingered robotic hand manipulation system as shown in figure 2. let P be the palm frame, O the object frame, and F_i the frame of fingertip i. Denote the object wrench, contact forces and finger joint efforts (forces or torques, depending on the joint types) as $F \in \Re^6$, $x \in \Re^m$, and $\tau \in \Re^n$, where m is the total number of contact forces and n the total number of joint efforts. These forces are related by

$$
\begin{aligned}
F &= Gx \\
\tau &= J^T x + g_{ext}
\end{aligned} \tag{1}
$$

where $G \in \mathrm{R}^{6 \times m}$ is the grasp map, $J \in \mathrm{R}^{m \times n}$ is the hand Jacobian and $g_{ext} \in \mathrm{R}^n$ is the vector of generalized forces experienced by the joints due to external loads such as gravity (and Coriolis, centripetal, and inertial loads if dynamics need to considered).

In a practical robotic system, joint efforts are not unlimited. Assume that the elements of the joint effort vector are bounded by known constants: the lower bound $\tau^L \in \Re^n$ and the upper bound $\tau^U \in \Re^n$, then the joint limit constraints can be expressed as linear inequality constraints:

$$
\tau^L \leq \tau \leq \tau^U \tag{2}
$$

The friction constraints for contact forces are generally nonlinear, except for the frictionless contact. Take point contact with friction(PCWF) as an example. Its friction cone can be expressed as

$$
\mathcal{F}_i = \left\{ x_i \in \Re^3 \mid x_{i3} \geq 0, \quad \frac{1}{\mu_i^2}(x_{i1}^2 + x_{i2}^2) \leq x_{i3}^2 \right\} \tag{3}
$$

where x_{i3} is the normal force component at the point of contact, x_{i1}, x_{i2} are the tangential components and μ_i is the coefficient of Coulomb friction. In general, the contact force at each contact point i need to satisfy its own contact model $\mathcal{F}_i$, which may be frictionless contact, point contact with friction or soft-finger contact. In other words, the feasible contact force of a grasp must lie within the friction cone $\mathcal{F}$ of the grasp:

$$
x = [x_1^T \ldots x_i^T \ldots x_k^T]^T \in \mathcal{F} = \mathcal{F}_1 \times \cdots \times \mathcal{F}_i \times \cdots \times \mathcal{F}_k \subset \mathrm{R}^m \tag{4}
$$

Given a desired object wrench, we are interested in computing and optimizing grasp forces x and joint efforts τ which can generate the wrench, which are the standard grasp force feasibility and optimization problems.

Problem 1 [Grasp Force Feasibility Problem]
Given an object wrench F, joint external loads g_{ext}, a grasp map G, and a hand Jacobian J, determine if $\exists\, x \in \Re^m$ and $\tau \in \Re^n$, satisfying the force equilibrium constraints (1), joint effort limit constraints (2) and friction cone constraints (4).

Problem 2 [Grasp Force Optimization Problem]
Given an object wrench F, joint external loads g_{ext}, a grasp map G, and a hand Jacobian J, find "optimal" grasp force $x \in \Re^m$ and joint effort $\tau \in \Re^n$ satisfying the force equilibrium constraints (1), joint effort limit constraints (2) and friction cone constraints (4).

Although these problems involve nonlinear friction cone constraints, they can be efficiently solved [2, 4, 5, 9, 20]. More specifically, the feasible domains of these problems are convex, and physically meaningful convex objective functions can be formulated. Thus, the problems can be solved by convex programming algorithms.

2.2 Kinematics

Denote the forward kinematics of finger i by

$$g_{pf_i} \quad = \quad g_{pf_i}(\theta_i) \tag{5}$$
$$V_{pf_i} \quad = \quad J_{pf_i}(\theta_i)\dot{\theta}_i \tag{6}$$

where $g_{pf_i} \in SE(3)$ and $V_{pf_i} \in se(3)$ are the configuration and velocity of fingertip i with respect to the palm, respectively; $\theta_i = (\theta_{i1}, \cdots \theta_{in_i})$ is the joint variable vector of finger i, and $J_{pf_i}(\theta_i)$ is the Jacobian of the forward kinematic map $g_{pf_i}(\theta_i)$.

Let the contact configuration between the object and finger i be $\eta_i = (\alpha_{o_i}, \psi_i, \alpha_{f_i})$, where $\alpha_{f_i} \in \Re^2$ and $\alpha_{o_i} \in \Re^2$ are the coordinates of contact points relative to the fingertip and the object frames, respectively, and ψ_i is the contact angle. Various contact models, such as rolling contacts or sliding contacts, pose different constraints on feasible contact movements [23]. In the following discussion, we will use $\tilde{\eta}_i$ to denote a subset of contact parameters whose velocities can determine the velocities of other contact parameters. Let $q_i = (\theta_i, \eta_i) \in \Re^{n_i+5}$ be the extended joint coordinates of finger i. First note that the admissible values and velocities of joint variables and contact parameters are limited. Similar to the joint effort limit constraints (2), the extended joint coordinates q_i need to satisfy the following linear inequalities:

$$q_i^L \quad \le \quad q_i \le q_i^U$$
$$\dot{q}_i^L \quad \le \quad \dot{q}_i \le \dot{q}_i^U \tag{7}$$

Expressing the object configuration relative to the palm frame through finger i yields

$$g_{po} = g_{pf_i}(\theta_i) \cdot g_{f_i o}(\eta_i) \tag{8}$$

Equation (8) need to be satisfied by all finger links in order to maintain a kinematically valid grasp, and
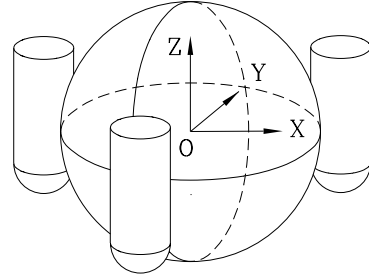

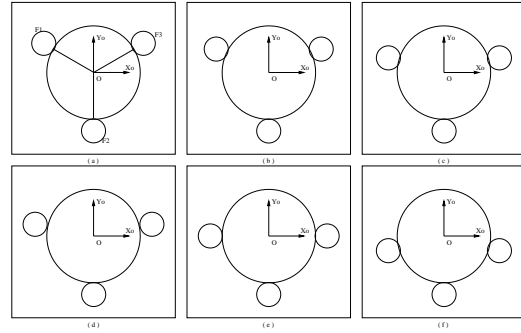Figure 3: A three-fingered robotic hand manipulating a ball.

Figure 4: Uncoordinated motion of a three-fingered hand leading to failure of force-closure condition for small friction coefficients.

will be referred as *closure constraints* in the following discussion. Differentiating equation (8) yields velocity closure constraints:

$$V_{po} = Ad_{g_{f_i o}^{-1}} V_{pf_i} + T_i(\eta_i)\dot{\tilde{\eta}}_i \tag{9}$$

Contact kinematics[23] can be used to simplify equation (9) to

$$G^T V_{po} = J\dot{\theta} \tag{10}$$

where G and J are, same as in equation (1), the grasp map and the hand Jacobian of the manipulation system. Since it relates object velocity to finger joint velocity directly, equation (10) has been widely used to generate finger joint velocities to accommodate the object velocity by utilizing generalized inverse method. However, this strategy doesn't take the contact movement into consideration and may cause degradation of the grasp quality as shown in the following example.

Example 1 *Consider a three-fingered robotic hand manipulating a ball as shown in figure 3. The fingertips are all cylindrically-shaped with a hemispheric top. At the initial grasp configuration the three fingers are located along an equator with 120 degrees apart from*

each other. This is considered an optimal grasp for a three-fingered hand. [1] *Let the desired trajectory of the object be a translation along the y-axis, i.e., away from Finger Two. Computing the fingertip velocities using equation (10) and a PCWF friction model, we found that Finger Two would simply translate upward and the other two fingers roll down toward Finger Two. Eventually, the three fingers will cluster around Finger Two, leading to degradation of the grasp quality. A simulated sequence of intermediate configurations projected from the top is shown in figure 4.* □

The preceding example shows that the generalized inverse method with respect to equation (10) has no regard to the quality of the resulting grasp. As manipulation proceeds, grasp quality would degrade, leading eventually to dropping of the grasped object. Therefore, in general, we need to include all kinematic variables in the manipulation planning and control.

3 Dextrous Manipulation Planner

Define a configuration, or a state, of a manipulation system to be

$$c = (g_{po}, \eta, \theta, F) \in SE(3) \times \Re^{5k} \times \Re^n \times \Re^6$$

Here we assume that the system moves slowly and satisfies quasi-static conditions. Thus we don't need to include the velocities as state variables.

Note that the kinematic variables (g_{po}, η, θ) are constrained by the closure constraints (8), which define a manifold [3] and can be used to determine the values of a subset of the kinematic variables based on those of other kinematic variables that can serve as *parameterization variables*. Since different patches of a manifold may have different parameterization variables, we include all of the kinematic variables (g_{po}, η, θ) in the states with the understanding that they satisfy the closure constraints (8).

Definition 1 [Feasible Configuration] *A configuration $c = (g_{po}, \eta, \theta, F)$ of a manipulation system is feasible if the system is collision free, (g_{po}, η, θ) satisfy the kinematic closure constraints (8) as well as limit constraints (7), and $\exists x \in \Re^m, \tau \in \Re^n$ that satisfy the force equilibrium constraints (1), joint effort limit constraints (2) and friction cone constraints (4).*

Definition 2 [Feasible Paths] *A path $c(t), t \in [0, T]$ is feasible if each configuration on the path is feasible.*

---

[1] The grasp quality in this particular example and in the experiment described in section 5 is defined as the area of the triangle formed by the three contact points, which is motivated by the concept of force closure.

Problem 3 [Dextrous Manipulation Planning Problem]
Given an initial configuration c^0 and a goal configuration c^1, find a feasible path $c(t), t \in [0, T]$ such that $c(0) = c^0$ and $c(T) = c^1$.

In practice, a manipulation task may be specified only with respect to a subset of the states. Consider a manipulation task which is to move the object from one displacement g_{po}^0 to another displacement g_{po}^1. Although the task is only related to the object displacement, we still need to plan a path for all state variables, $(g_{po}, \eta, \theta, F)$ with the freedom to choose initial and final (η, θ, F) as long as they make feasible configurations with g_{po}^0 and g_{po}^1, respectively.

4 Dextrous Manipulation Controller

Given a desired path $(g_{po}^d, \eta^d, \theta^d, F^d)(t), t \in [0, T]$, generated by a planner, a controller need to incorporate sensor measurement, system kinematics and statics to generate proper control commands in order to achieve desired path. This section briefly introduces a unified Control System Architecture for Multifingered Manipulation (CoSAM[2]) as shown in figure 5.

In this section, the superscripts d and m will be used to denote desired values and measured values. Let T_0 be the sampling period. Then the discretized desired trajectory is

$$(g_{po}^d, \eta^d, \theta^d, F^d)(k) = (g_{po}^d, \eta^d, \theta^d, F^d)(kT_0) \quad (11)$$
$$k = 0, \cdots, [T/T_0]$$

where $[T/T_0]$ is the smallest integer not less than T/T_0. If T/T_0 is not a integer, we need to take special care of the last time interval. However, this can be done easily and the notation(11) will be used for simplicity. At step k, the tasks performed by the controller components are the following.

- Object Motion Generator

  let $g_{po}^m(k) \in SE(3)$ be the position and orientation of the object sensed by say vision sensors or tactile sensors. Then, compute $V_{po}^d(k) \in se(3)$ by solving

  $$\exp(V_{po}^d(k)T_0) = (g_{po}^m(k))^{-1} \cdot g_{po}^d(k+1) \quad (12)$$

  Note that we can use screw motions [12] or piecewise screw motions to specify desired object trajectories. Geometrically, a screw motion is simple to visualize and also easy to generate.
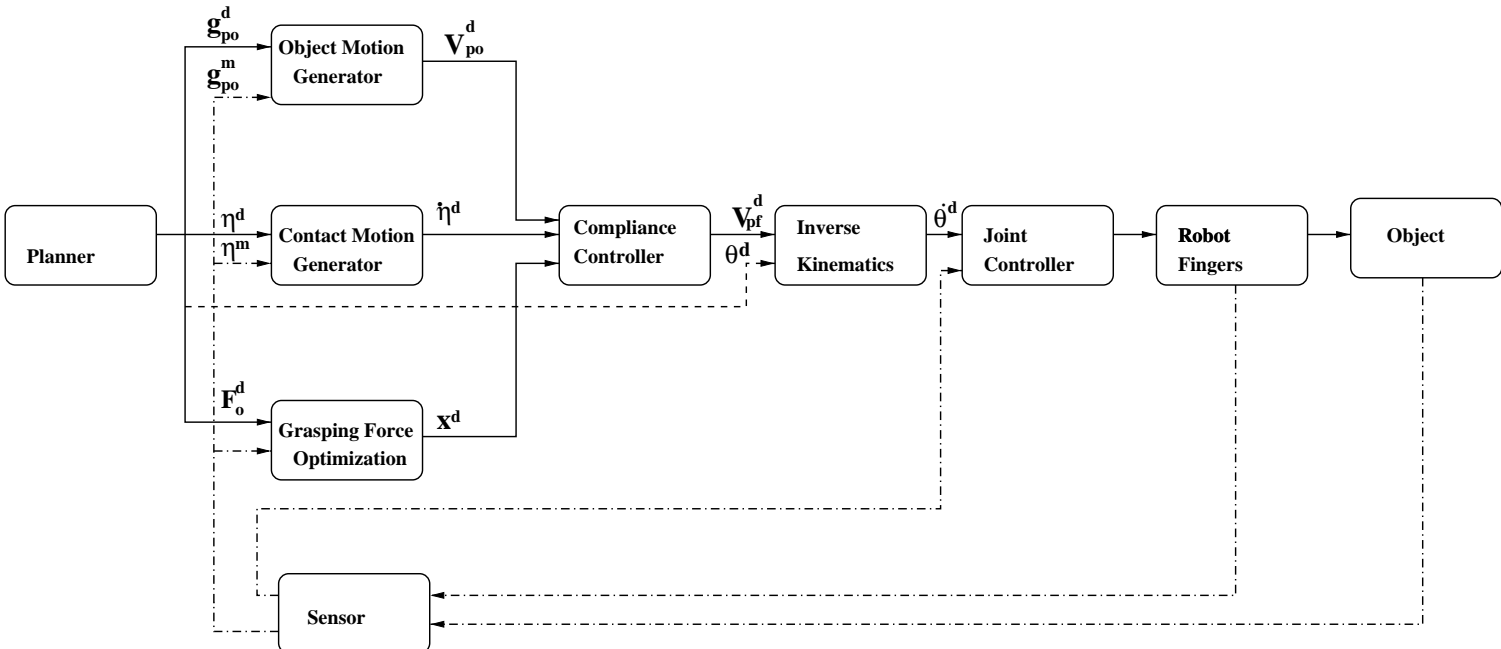
- Contact Motion Generator

Figure 5 (block diagram), nodes and signal labels:

Planner → g_{po}^d, g_{po}^m → **Object Motion Generator** → V_{po}^d

η^d, η^m → **Contact Motion Generator** → $\dot{\eta}^d$

F_o^d → **Grasping Force Optimization** → x^d

→ **Compliance Controller** → V_{pf}^d, θ^d → **Inverse Kinematics** → $\dot{\theta}^d$ → **Joint Controller** → **Robot Fingers** → **Object**

Sensor

Figure 5: CoSAM²: A unified Control System architecture for Multifingered Manipulation

The desired contact velocity is simply the linear velocity to move the contact from current configuration $\tilde{\eta}^m(k)$ to next desired configuration $\tilde{\eta}^d(k+1)$

$$\dot{\tilde{\eta}}^d(k) = (\tilde{\eta}^d(k+1) - \tilde{\eta}^m(k))/T_0 \qquad (13)$$

- Grasp Force Optimization

For a desired object wrench $F^d(k+1)$, find a feasible grasp force $x^d(k+1)$ and then optimize it. As discussed in section 2.1, these problems can be formulated as convex programming problems and can be solved efficiently.

- Compliant Motion Control and Inverse Kinematics

Our current control system uses a compliance control strategy and treats each finger as a position-controlled device. We choose this architecture mainly because our experimental platform, HKUST Hand, is equipped with PID position controller for each finger. However, it would be possible to use other control strategies and would be interesting to compare the performances of different control laws. Assume the compliance matrix of finger i is $K_{ci} \in \Re^{6\times6}$. Then the desired grasp force $x^d(k+1)$ can be converted into equivalent fingertip displacement

$$K_{ci}(F_i^d(k+1) - F_i^m(k)) \qquad (14)$$

where $F_i^d(k+1)$ and $F_i^m(k)$ are the desired and measured fingertip wrenches of finger i, computed from the desired and measured contact forces $x_i^d(k+1)$ and $x_i^m(k)$.

The kinematic equation (9) determines a kinematically desired fingertip velocity, $V_{pf_i}^K(k)$, based on the desired object and contact velocity (12, 13). (The superscript in $V_{pf_i}^K(k)$, K, denotes the kinematically desired fingertip velocity.) Combining this term with the compliance displacement (14) yields the desired fingertip velocity

$$\boxed{V_{pf_i}^d(k) = V_{pf_i}^K(k) + K_{ci}(F_i^d(k+1) - F_i^m(k))} \qquad (15)$$

which will be sent to the Inverse Kinematics module to compute the required joint displacement of the fingers.

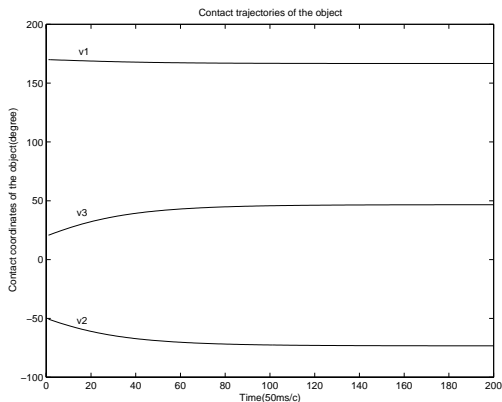Figure 6: Three cylinder fingertips manipulating a ball

Figure 7: Experimental Contact trajectories

5   Experiment Results

In this section, we will use a simple manipulation example conducted on HKUST hand system to show that the proposed control architecture CoSAM² (figure 5) works correctly when the desired object wrench, object, *and* contact trajectories are planned properly. The manipulation task is relatively simple, which enables us to bypass the planner and to let the controller generate a feasible path. All control components perform the same computation as described in the previous section, except the *contact motion generator*, which will generate desired contact velocity to optimize grasp quality. More information and experimental results can be found in [18].

The task here is to manipulate a ball with three cylindrical fingertips as shown in Figure 6. The initial grasp configuration is $\eta_1 = (0, 170°, 180°, 0, 0)$, $\eta_2 = (0, -50°, 180°, 0, 0)$ and $\eta_3 = (0, 20°, 180°, 0, 0)$. The desired object motion is a screw motion along the z- axis, i.e $\xi = [\ 0\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ ]^T$ and motion magnitude is $\theta = 100mm$. Recall that the manipulation task in the simulation example (figure 4) is similar to this one. In the previous case, the con-
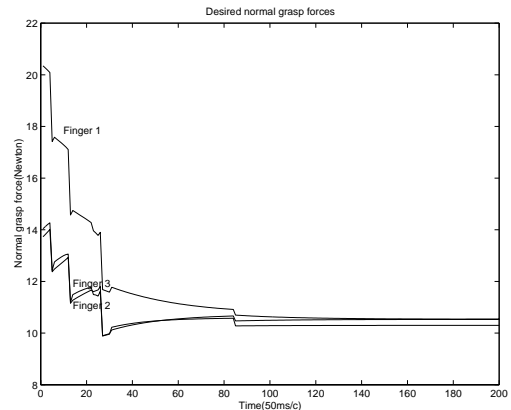

Figure 8: Experimental Grasp Forces

tact movement is not controlled by the robotic system and the manipulation task fails because of degradation of the grasp quality . On the other hand, our experimental results (figures 7 and 8) show that the grasp quality is improved (the final contacts form an equilateral triangle on the equator of the ball, which is optimal) and the grasp forces are optimized during manipulation process.

6   Conclusion

Dextrous manipulation is a fundamental problem in the study of multifingered robotic hands. In this overview paper, we defined the manipulation planning problem and presented a unified Control System Architecture for Multifingered Manipulation (CoSAM²). By incorporating the various kinematic and static relationships of a multifingered robotic hand system with proper sensory data inputs at different stages, CoSAM² can achieve the various objectives of dextrous manipulation. Theoretical background of the system design along with real-time experimental results were described.

In future work, we plan to pursue several research directions to help us further understand the dextrous manipulation problem. These include the development of efficient planning algorithms, the design of controllers robust to the significant sources of uncertainty, and the incorporation of dynamics into the system model.

References

[1] P. Allen, A. Miller, P. Oh, and B. Leibowitz. Using tactile and visual sensing with a robotic hand. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, pages 676–681, 1997.

[2] A. Bicchi. On the closure properties of robotic grasping. *International Journal of Robotics Research*, 14(4):319–334, August 1995.

[3] W. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry*. Academic Press, 1975.

[4] M. Buss, L. Faybusovich, and J. Moore. Dikin-type algorithms for dextrous grasping force optimization. *International Journal of Robotics Research*, 17(8), Aug. 1998.

[5] M. Buss, H. Hashimoto, and J. Moore. Dextrous hand grasping force optimization. *IEEE Transactions on Robotics and Automation*, 12(3):406–418, 1996.

[6] J.F. Canny. *The complexity of robot motion planning*. MIT Press, 1988.

[7] A. Cole, J. Hauser, and S. Sastry. Kinematics and control of a multifingered robot hand with rolling contact. *IEEE Transaction on Automatic Control*, 34(4), 1989.

[8] R. Fearing. Tactile sensing for shape interpretation. In S.T. Venkatarman and T. Iberall, editors, *Dextrous robot hands*. Springer-Verlag, 1989.

[9] L. Han, J.C. Trinkle, and Z.X.Li. Grasp analysis as linear matrix inequality problems. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, 1999.

[10] J.W. Hong, G. Lafferriere, B. Mishra, and X.L. Tang. Fine manipulation with multifinger hand. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, pages 1568–1573, 1990.

[11] R. Howe, I. Kao, and M. Cutkosky. The sliding of robot fingers under combined torsion and shear loading. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, 1988.

[12] K. Hunt. *Kinematic Geometry of Mechanisms*. Oxford University Press, 1978.

[13] J. Kerr and B. Roth. Analysis of multifingered hands. *International Journal of Robotics Research*, 4(4):3–17, 1986.

[14] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1), 1986.

[15] J.C. Latombe. *Robot Motion Planning*. Kluwer Publishers, 1990.

[16] Z.X. Li and J. Canny. Motion of two rigid bodies with rolling constraint. *IEEE Transactions on Robotics and Automation*, RA2-06:62–72, 1990.

[17] Z.X. Li, P. Hsu, and S. Sastry. On grasping and coordinated manipulation by a multifingered robot hand. *International Journal of Robotics Research*, 8:4, 1989.

[18] Z.X. Li, Z.Q. Qin, and S.L. Jiang. Cosam^2: A unified control system architecture for multifingered manipulation. Technical report, EEE Dept., Hong Kong Univ. of Science & Technology., 1997.

[19] Y.H. Liu. Qualitative test and force optimization of 3d frictional form-closure grasps using linear programming. *IEEE Transactions on Robotics and Automation*, 15(2), 1999.

[20] M. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and Applications, special issue on linear algebra in control, signals and image processing*, 284:193–228, November 1998.

[21] T. Lozano-Perez. Spatial planning: A configuration space approach. *IEEE Trans. on Computers*, C-32(2):108–120, 1983.

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

[23] D. Montana. The kinematics of multi-fingered manipulation. *IEEE Transactions on Robotics and Automation*, 11(4):491–503, 1995.

[24] R. Murray, Z.X. Li, and S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[25] Y. Nakamura, K. Nagai, and T. Yoshikawa. Mechanics of coordinative manipulation by multiple robotic mechanisms. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, pages 991–998, 1987.

[26] V.-D. Nguyen. Constructing force-closure grasps. *International Journal of Robotics Research*, 7(3), 1988.

[27] A. Okamura, M. Turner, and M. Cutkosky. Haptic exploration of objects with rolling and sliding. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, 1997.

[28] J.S. Pang and J.C. Trinkle. Complementarity formulations and existence of solutions of dynamic multi-rigid-body contact problems with coulomb friction. *Methematical Programming*, pages 73:199–226, 1996.

[29] J.S. Pang, J.C. Trinkle, and G. Lo. A complementarity approach to a quasistatic rigid body motion problem. *Journal of Computational Optimization and Applications*, 5(2):139–154, March 1996.

[30] Jean Ponce and Bernard Faverjon. On computing three-finger force-closure grasps of polygonal objects. *IEEE Transactions on Robotics and Automation*, 11(6), 12/1995.

[31] E. Rimon and J. Burdick. On force and form closure for multiple finger grasps. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, 1996.

[32] M. Zefran, V. Kumar, and C. Croke. Metrics and connections for rigid body kinematics. In *Proc. of IEEE Intl. Conf. on Robotics and Automation*, 1996.