On line Computation of Grasping Forces in Multi-fingered Hands

Jean-Philippe Saut*† , Constant Remond*, Véronique Perdereau* and Michel Drouin*
*Laboratoire des Instruments et Systèmes d'Île-de-France
Université Pierre et Marie Curie (Paris VI) boîte 252
3 rue Galilée, 94200 Ivry-sur-seine, France
† jean-philippe.saut@lisif.jussieu.fr

Abstract— This paper presents a new solution for solving the grasping force optimization problem, fundamental in dextrous manipulation by multifingered robotic hands. Several methods have been proposed in the literature, yielding optimal solutions, with either recursive or non linear programming techniques. However, most of them involve many computations and cannot be used online. Furthermore, they do not offer a smooth solution regarding to possible changes in the contact conditions due to finger rolling or gaiting, or in the desired resultant force to be exerted on the grasped object. The more recent ones are fast and smooth enough for real-time computation but the method we present here is faster, easier to implement and provide very satisfying results, even though the solution is sub-optimal. The method is based on the minimization of a cost function that gives an analytical solution but does not ensure by itself the satisfaction of the static frictional constraints. An associated iterative adjustment modifies this function until the internal forces enter the friction cone. The minimal solution is found within a few iterations. Force determination is therefore included in the simulation of an hybrid position/force controller to prove the effectiveness of such an approach for updating the force references during the grasped object motion.

I. INTRODUCTION

The study of the object manipulation by multi-fingered robotic hands has been the source of many works over the twenty past years. This research area involves the conception of efficient mechanical structures and control schemes, and implies issues like the kinematics and statics in the closed kinematic chain ([1],[2],[3],[4]), the optimization of the internal forces and the hybrid position/force control of the hand ([5],[6]). Other topics in robotic manipulation are the reconfiguration of the grasp ([7],[8]) and the determination of the optimal grasp ([9],[10],[11]), that is, the grasp that ensures the best robustness against slippage. The main concern of this paper is the optimization of the internal forces to be exerted on the manipulated object.

The determination of the optimal forces raises difficulty dealing with the nonlinearity of the contact friction models, usually chosen. Therefore, early works linearized the static frictional constraints to use linear programming methods like simplex or gradient search ([12],[13]). The drawbacks of such methods are the non respect of contact isotropy and the non smoothness of the solution as small perturbations in the grasp parameters may generate large changes in the solution. However, great advances have been done in the grasping force optimization problem with the work of Buss, Hashimoto and Moore (noted BHM thereafter). Indeed, [14] expressed the frictional constraints as the positive definiteness of a symmetric matrix, obeying also to linear constraints. The grasping force optimization was then written as an optimization problem on the manifold of linearly constrained and positive definite matrices and solved using gradient flow algorithms. This method was efficient but required a valid initial point to start the gradient algorithm. In reference [15], Han, Trinkle and Li proposed a variant (noted here HTL). The matrix whose positive definiteness represents the frictional constraints is written as a linear combination of matrices; the optimization problem becomes a linear matrix inequalities problem, using the determinant maximization (max-det). The chosen cost function is composed of one term minimizing the normal components of the contact forces and another term keeping the solution away from the frictional constraint boundaries. Finally, Liu and Li ([16]) proposed a solution for the initialization and a very good synthesis of the BHM's and HTL's works. They also implemented the algorithms and run them on a real experiment platform with the HK-UST three-fingered hand.

This paper presents an online method for the computation of optimal contact forces. In order to solve the redundancy involved by the multiple contacts between the fingers and the object, the inequality constraints such as the static frictional constraints are added to the set of the equilibrium equations. A new quadratic function to be minimized is built so that its positive definiteness is always guaranteed. Given the analytical expression of its minimum, the optimal solution is found without a lot of effort, allowing real-time implementation. The result of such a computation is therefore introduced in a hybrid position/force control structure for controlling the manipulation of an object. The grasping forces to be exerted by the fingers are adjusted during the motion according to the characteristics of the contacts, the desired resultant force and momentum, and the manipulated object mass properties. Although the presented method does not claim to offer better results than others, it provides reliable solutions, faster and is easier to implement.

II. BASIC EQUATIONS

The cooperative manipulation of a rigid object by the fingers of a m-fingered mechanical hand is considered. Each finger is assumed to have a frictional point contact with the object. F_i is the force applied to the object by the ith finger (Fig. 1). This section quickly reviews the necessary mechanical background. The static equilibrium is expressed with respect to the object mass center as:

$$Q = WF \tag{1}$$

Where Q is the resultant generalized force applied to the object, $F = (F_1^T, F_2^T, ...F_m^T)^T$ represents the individual contact forces and W is the well known grasp matrix. Since Q is the generalized load required to move the object, one has to predict the individual finger contributions F for controlling the

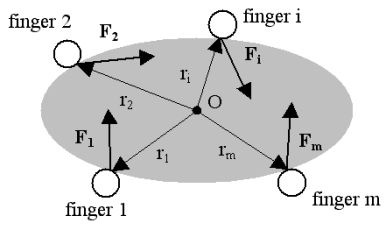

Fig. 1. Contact forces

articulated hand. The solution is clearly found in inverting the previous equation but the matrix W is not square; a general solution can be obtained with a Moore-Penrose generalized inversion leading to the decomposition of the force into two terms as:

$$F = W^{+}Q + (I - W^{+}W)y \tag{2}$$

Where $W^{+} = W^{T}(WW^{T})^{-1}$, is the identity matrix and y an arbitrary 3m-vector, with m the number of fingers (contacts). The first term in this equation is the so-called external force F_{ext} that balances the gravity force and inertial effects to move the object. The second term corresponds to the internal forces F_{int} that do not cause the object motion.

$$F = F_{ext} + F_{int} \tag{3}$$

Redundancy is inherently present in the multi-fingered system and the redundant degrees of freedom are used to adjust the internal forces. So, there is no unique solution for choosing the arbitrary vector y. Consequently, F_{int} is necessarily derived from an optimization process like the one described in the next section. This paper aims to give a new solution to this optimization problem, for computing on-line the internal forces during the manipulation.

III. GRASPING FORCE OPTIMIZATION

For any vector y, the vector computed from (2) satisfies (1) but grasp stability is not guaranteed for any set of contact forces. The contacts between the fingers and the grasped object are generally represented by one of the following friction models: point contact with friction (PCWF) or soft finger contact with elliptical approximation (SFCE). Let F_{n_i} and F_{t_i} be the normal and tangential components, respectively, of the ith contact force as represented in Fig. 2 and μ_i the Coulomb friction coefficient of this contact.

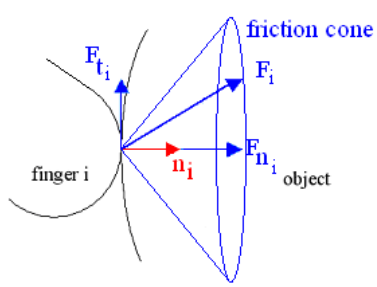

Fig. 2. Friction cone at contact point

In case of a PCWF model, no moment can be exerted through the contact, nor pulling force. The frictional constraint is expressed by:

$$\|F_{n_i}\| \geq \tfrac{1}{\mu_i}\|F_{t_i}\| \tag{4}$$

$$F_{n_i} \cdot n_i > 0 \tag{5}$$

On the contrary, in case of a SFCE model, a moment can be exerted around the normal direction n_i and the constraints become:

$$\|F_{n_i}\| \geq \sqrt{\tfrac{1}{\mu_i^2}\|F_{t_i}\|^2 + \tfrac{1}{\mu_{t_i}^2}\|m_i\|^2} \tag{6}$$

$$F_{n_i} \cdot n_i > 0 \tag{7}$$

Where m_i is the moment exerted along the contact normal and μ_{t_i} an additional friction coefficient. In both cases, these inequalities form together a set of m linear constraints and m quadratic constraints for a m-fingered hand. However, they do not allow us to discriminate a single solution from all possible. A criterion is necessarily optimized to find the right grasping force configuration. As large internal forces are not appropriate for breakable objects and may give rise to an unexpected moment at the center of mass, resulting in object slipping, optimal forces are accordingly the minimal forces that generate the specified resultant force Q under the static frictional constraints. The optimization problem is then stated as:

$$\begin{array}{ll} \textit{minimize } F^T F & \\ \textit{with respect to} & \left\{ \begin{array}{l} K_i > 0 \\ F_{n_i} \cdot n_i > 0 \end{array} \right. \end{array} \tag{8}$$

Where K_i depends on the chosen contact model.

$$K_i = \|F_{n_i}\| - \tfrac{1}{\mu_i}\|F_{t_i}\| \qquad \text{PCWF model}$$

$$or \quad K_i = \|F_{n_i}\| - \sqrt{\tfrac{1}{\mu_i^2}\|F_{t_i}\|^2 + \tfrac{1}{\mu_{t_i}^2}\|m_i\|^2} \qquad \text{SFCE model}$$

Two main approaches are chiefly used to solve this optimization problem: an iterative approach and an analytical approach. The iterative approach consists in a numerical procedure. Many algorithms have been proposed, either based or not on the gradient knowledge. However, methods based on the gradient may derive solutions that correspond to local minima. In such cases, a slight modification in the contact positions when the fingers roll or when the desired resultant force changes may generate an important modification in the solution of the algorithm. Such discontinuities can affect the stability of the hand.

Some recent techniques like the one developed by Han et al. ([15],[16]), offer excellent results, allowing real-time but their complexity makes their implementation fastidious and they still need a powerful computing system to run fast. The analytical approach is based on the Lagrange non linear programming method. A solution is easily obtained when the Lagrange function is quadratic with respect to the parameter of the cost function. Here, this condition is verified since the force magnitude criterion is quadratic and the $2m$ inequality constraints are linear or quadratic. This procedure has been developed by Nakamura ([6]). Unfortunately, the calculation must be repeated a large number of times and the advantage of having an analytical solution is lost. For example, 3 fingers leading to 6 inequality constraints, 64 parameter combinations have to be examined, each involving a 3-square matrix inversion. Such

a solution cannot be utilized for an on-line implementation. The original method proposed in this paper first transforms the constrained problem into an unconstrained one, then finds an analytical solution and, finally, uses an iterative tuning for the fine adjustment of the internal forces.

IV. PROPOSED METHOD

The main idea of the proposed method consists in transforming the constrained optimization problem to be solved into an unconstrained quadratic one in such a way that an analytical solution is easily obtained. A classical solution could have been to introduce the $2m$ inequality constraints (with m the number of fingers) into the criterion thanks to a penalty approach. In this case, however, the m quadratic inequalities cannot directly provide an analytical optimal solution because the positive definiteness of the second-order derivative is not true in all cases. Hence, the term corresponding to these constraints to be introduced in the criterion must be linear. A new criterion is therefore chosen as:

$$J = F^T F - \sum_i \sigma_i f_{n_i} \tag{9}$$

where $f_{n_i} = F_{n_i} \cdot n_i$.
The second term aims to increase the normal components and, consequently, draws up the forces F_i within the friction cone. The larger the weight parameter σ_i is, the better the force belongs to the friction cone but the larger the force magnitude is.
The new criterion J is equivalently written as follows:

$$J = F^T F - (N\sigma)^T F \tag{10}$$

with:

$$N = \begin{pmatrix} n_1 & 0_{3\times1} & \cdots & \cdots & 0_{3\times1} \\ 0_{3\times1} & n_2 & \ddots & 0_{3\times1} & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & 0_{3\times1} & \ddots & \ddots & 0_{3\times1} \\ 0_{3\times1} & \cdots & \cdots & 0_{3\times1} & n_m \end{pmatrix} \text{ for PCWF model} \tag{11}$$

$$N = \begin{pmatrix} n_1^{'} & 0_{4\times1} & \cdots & \cdots & 0_{4\times1} \\ 0_{4\times1} & n_2^{'} & \ddots & 0_{4\times1} & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & 0_{4\times1} & \ddots & \ddots & 0_{4\times1} \\ 0_{4\times1} & \cdots & \cdots & 0_{4\times1} & n_m^{'} \end{pmatrix} \text{ for SFCE model} \tag{12}$$

where $n_1, n_2, \ldots, n_m$ are the contact normals,

$$n_i^{'} = \begin{bmatrix} n_i \\ 0 \end{bmatrix} \text{ and } \sigma = \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_m \end{bmatrix} \tag{13}$$

We now want to find the vector F that minimizes J under the condition $WF - Q = 0$.
For that purpose, we apply a non-linear programming method based on the Lagrange multipliers. The Lagrange function is written as follows:

$$L(F, \lambda) = F^T F - (N\sigma)^T F + \lambda^T (WF - Q) \tag{14}$$

It is quadratic with respect to F and its second order derivative in F is positive definite. The necessary and sufficient condition for (F_0, λ_0) to be the minimum of $L(F, \lambda)$ is:

$$\left.\frac{\partial L}{\partial F}\right|_{F=F_0;\lambda=\lambda_0} = 0 \quad and \quad \left.\frac{\partial L}{\partial \lambda}\right|_{F=F_0;\lambda=\lambda_0} = 0 \tag{15}$$

Resolving (15) yields the analytical solution:

$$F = W^+ Q + \frac{1}{2}(I - W^+ W) N\sigma \tag{16}$$

Note that this solution looks close to the general solution (2) of $WF = Q$ but contains only m unknowns, i.e. one unknown per contact (versus $3 \times m$ for the general one with PCWF model and $4 \times m$ with SFCE model, i.e. three and four unknowns per contact, respectively).
However, this expression in which only the vector σ is variable, does not ensure that this solution is suitable for grasp stability. Indeed, it does not ensure that each contact force vector belongs to its friction cone. The additional term introduced in the criterion plays a similar role to penalty function. Consequently, an iterative procedure is now necessary to adjust the weight parameters in order to satisfy the inequality constraints and a sub-optimal solution is finally obtained by carrying the two following steps.
In the first step, a rough estimation of the parameters σ_i is found by noting that increasing the coefficient σ_i associated with the ith contact increases the corresponding contact force normal component f_{n_i} and that this component must be all the more big as the object weight Mg is big, the finger number m is small and the friction coefficient μ small. So we propose the initialization of σ_i as follows:

$$\sigma_i = \frac{Mg}{\mu m} \tag{17}$$

The resulting force F and parameters K_i are computed undertaking (9) and (16).
This estimation does not automatically ensure that the static frictional constraints are all satisfied, but is a good initialization for the next step. In the second step, the weighting factors are progressively increased to make each force vector F_i belong to the friction cone, according to the following procedure where c is the iteration index and $\alpha < 1$, an incrementing factor:

1) For each i,
   IF $K_i < 0$ or $f_{ni} < 0$
   THEN $\sigma_i^{c+1} = \frac{\sigma_i^c}{\alpha}$
   ELSE
   { IF $K_i > 0$ and $f_{ni} > 0$
   THEN $\sigma_i^{c+1} = \alpha\ \sigma_i^c$ }
2) Compute F with the new parameter vector σ^{c+1}
3) IF $\exists i, F_i \notin$ friction cone
   THEN return to step 1.
   ELSE exit the program.
   A condition can be added to this third step to ensure that the contact forces are close to the friction cone boundaries like "do not leave the loop if $K_i < bK$" (with b a number less than but close to 1) that will decrease the norm of the contact forces.

<u>Remarks:</u>

1) As it will be shown by numerical examples, a few iterations are involved and, as a matter of fact, the computation time is largely lower than the computation

time of other methods (no matrix inversion is involved, except in the computation of W^+).

2) If the coefficient α is increased, the number of iterations decreases but the force magnitude increases.
3) In an on-line object manipulation, the value of the vector σ can be initialized, at each step of the control loop, by taking the value computed at the previous step. Consequently, the conditions (values of Q, W or N) being not very different, only a small iteration number will be needed. Moreover, this initialization offers a smooth solution during the manipulation; small changes in Q, W or N will not generate a great change of computed internal forces.
4) Algorithm convergence: the form of the proposed resolution method –a procedure using IF instruction in the solution computation –makes hard to prove wether or not and at which conditions it finds a solution. Nevertheless, doing lots of tests, we observed that it always finds a solution excepted when the conditions are close to the problem feasibility limit. However methods like BHM and HTL ones need a valid initial condition and methods to find this valid initial condition,like the one of Liu and Li[16], even if they should theoretically succeed, practically need too much time, for realtime applications, to find a solution in these cases.

V. NUMERICAL EXAMPLES

The method was implemented in C++ language. Several examples have been tested to verify its effectiveness. In this section, we give the results obtained for two of them. The computation times correspond to experiments conducted on a simple desktop PC (Intel Pentium IV, 1.8GHz, 480 Mo RAM). For simplicity reasons, the dimensions were chosen as round values since they have no influence on the computation convergence.

The first example corresponds to the case of a 4-fingers grasp of a cube object with PCWF contact model and non-planar contact positions (Fig. 3):

$$\begin{array}{llll} r_1 = [\ -0.5 & -0.5 & -1\ ]^T & n_1 = [\ 0 \quad 0 \quad 1\ ]^T \\ r_2 = [\ 0 & -1 & -0.5\ ]^T & n_2 = [\ 0 \quad 1 \quad 0\ ]^T \\ r_3 = [\ 1 & -0.5 & 0\ ]^T & n_3 = [\ -1 \quad 0 \quad 0\ ]^T \\ r_4 = [\ -1 & 0.5 & 0.5\ ]^T & n_4 = [\ 1 \quad 0 \quad 0\ ]^T \end{array}$$

We want a resultant force that compensates for the gravity for a 1kg-object and submits the object to a moment around the z-axis:

$$Q = [\ 0 \quad 0 \quad 9.81 \quad 0 \quad 0 \quad 2\ ]^T$$

The friction coefficient of all the contacts is 0.6 and the vectors are initialized as in (17). α is arbitrarily chosen as 0.9.

The results are summarized as follows:

	F_x	F_y	F_z	K_i
finger 1	-0.0189	-0.9005	1.5392	0.5852
finger 2	0.7673	3.4715	1.2974	0.4342
finger 3	-12.0835	3.9651	5.5229	0.5627
finger 4	11.3351	-6.5361	1.4505	0.5906
$\sqrt{F^T F}$	19.57			

Computing time	0.140ms=140 μs
Matlab language	0.20s

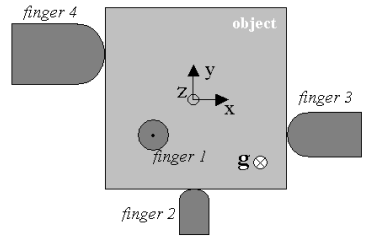

Fig. 3. Top view of the grasp.

Remark :

As shown in the following table, the solution convergence is improved when the coefficient α increases but the force magnitude increases too. The adjustment of this coefficient is the result of a compromise.

α	0.7	0.8	0.9	0.99
iteration number	12	18	32	298
computing time (ms)	0.062	0.078	0.140	1.234
$\sqrt{F^T F}$	23.93	21.76	19.57	19.02

The HTL method using the maxdet Matlab module, found in http://www.stanford.edu/ boyd/MAXDET.html gives the minimal finger force normal components sum when choosing the appropriate weight vector:

	F_x	F_y	F_z	K_i
finger 1	1.6778	0.1394	2.8060	0.6000
finger 2	0.0431	0.0748	-0.0124	0.5997
finger 3	-10.0850	4.0990	4.4511	0.6000
finger 4	8.3642	-4.3133	2.5653	0.6000
$\sqrt{F^T F}$	15.62			

As shown in this table, the results are close from what we found above. However, the computation time is four times the computation time of the proposed method (0.82s versus 0.20s with a Matlab program). For this example, we used the LMI formulation of the problem and Matlab LMI toolbox (lmilab) to determinate –if it exists– the friction coefficient value under which the problem becomes unfeasible. We found $\mu = 0.50429$. Our method can not find a solution for μ values under 0.525. For others methods, the only difficulty is to find a valid initial condition. Using the Liu and Li algorithm ([16]) to find this initial condition leads to an unacceptable computing time (complete optimal force computation using HTL method under Matlab takes more than 50s) while using Matlab LMI toolbox leads to a computation time of $1.2s$.

The second example illustrates the case of a SFCE contact model (Fig. 4) so that a moment around each contact normal is created.

$$\begin{array}{ll} r_1 = [\ -1 \quad 2 \quad 0\ ]^T & n_1 = [\ 1 \quad 0 \quad 0\ ]^T \\ r_2 = [\ -1 \quad -2 \quad 0\ ]^T & n_2 = [\ 1 \quad 0 \quad 0\ ]^T \\ r_3 = [\ 1 \quad 1 \quad 0\ ]^T & n_3 = [\ -1 \quad 0 \quad 0\ ]^T \\ r_4 = [\ 1 \quad -1 \quad 0\ ]^T & n_4 = [\ -1 \quad 0 \quad 0\ ]^T \end{array}$$

We want a resultant force that compensates for the gravity for a 1kg-object, pushes it along x-axis and exerts a moment around the x-axis:

$$Q = [\ 5 \quad 0 \quad 9.81 \quad 2 \quad 0 \quad 0\ ]^T$$

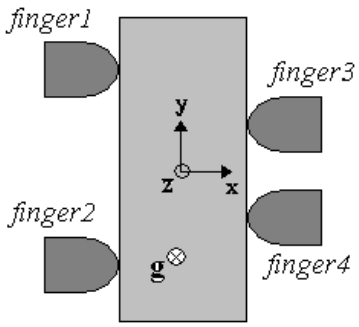

Fig. 4. Top view of the grasp.

The friction coefficients μ_i and μ_{t_i} are both 0.5. The results are given in the following table:

	F_x	F_y	F_z	m_i
finger 1	7.672	-0.097	2.738	0.143
finger 2	7.215	-0.097	2.167	0.143
finger 3	-5.206	0.097	2.595	-0.143
finger 4	-4.680	0.097	2.309	-0.143
$\sqrt{F^T F}$	13.57			

Computing time (ms)	0.844ms=844μs
Iteration number	201

With HTL method:

	F_x	F_y	F_z	m_i
finger 1	7.581	0.000	2.734	0.235
finger 2	7.229	0.000	2.170	0.284
finger 3	-5.257	0.000	2.263	0.000
finger 4	-4.552	0.000	2.276	0.000
$\sqrt{F^T F}$	13.50			

The computing times for Matlab language programs are still very different: 0.22s for the proposed method and 0.75s for the HTL method, respectively. In this example, the problem is always feasible, whatever are the friction coefficient values. The proposed method converges however friction coefficient are small.

VI. REAL TIME APPLICATION

This section presents the numerical results obtained from the complete simulation of a manipulation task. It was made using the Open Dynamics Engine (ODE) C++ library [17] that allows to simulate articulated bodies and rigid bodies contact dynamics. The simulated hand is constituted of four 3-DOF fingers (Fig. 5) that are controlled by a force-position control scheme already described in [18]-[19]. At each sampling period of the control loop, during the object motion, new contact force commands are generated according to the evolution of the positions and normals of the contacts, due to finger rolling or gaiting, and to the changes of the desired resultant force of the body. For that purpose, the vector computed at the previous cycle time is used as an initialization for the internal force computation. The manipulation task shown here is a screwing movement of a box-shaped object: it must be lift up and turned around the vertical axis. In order to show that the method can deal with difficult situations, a contact is broken at an arbitrary chosen time. The object characteristics and simulation parameters are the followings:

	initial	desired
Position(cm)	4.7	8
Orientation (rad)	[0 0 0] 0	[0 0 1] $\pi/8$
Dimensions (cm)	$3 \times 3 \times 6$	
Mass (kg)	0.1	
Contact model	point contact with friction	
Friction coefficient	1	
Time step of dynamics layer of the simulation (ms)	1	
Sampling period of the simulated control loop (ms)	10	
coefficient α used by the weights tuning procedure	0.95	

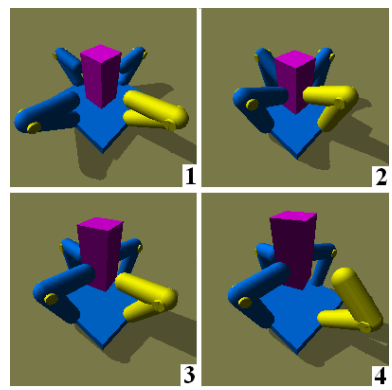

Fig. 5. TimesNewRomanPSMTpsyro Four steps of the screwing movement with breaking of a contact.

During the manipulation, the desired resultant force changes smoothly since it is provided by the controllers that are in charge of the servoing of the object position and orientation. As shown in Fig. 6, the desired contact forces evolve as smoothly thanks to the chosen iterative method.

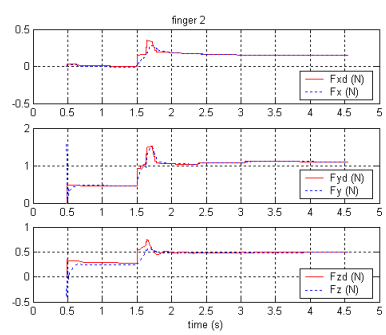

Fig. 6. Desired contact force of one of the fingers -calculated with our method- and the real contact force. Contact is made at 0.5s and broken at 1.5s after the beginning of the simulation.

Fig. 7 represents the evolution of the iteration number needed to perform the contact force computation. As expected, it is

high immediately after the beginning of the grasp and the breaking of the contact, and decreases quickly after. It always remains low enough to ensure computation times less than 0.1ms whereas the computation times of the other methods are about several ms.

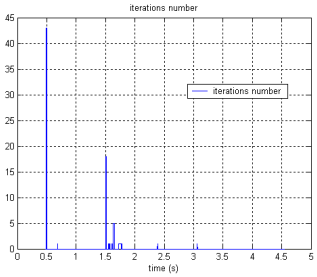

Fig. 7. Evolution of the iterations number.

VII. CONCLUSION

This paper presented an efficient method for computing the internal forces in a multi-fingered hand system. Grasping force optimization leads basically to the minimization of a quadratic function with respect to linear and quadratic constraints. Instead of solving this primal optimization problem, a new unconstrained problem was stated first. The inequality constraints were taken into account by introducing an adequate quadratic term in the objective function. Consequently, the solution was easily obtained by a m-matrix inversion and forces were strained within the friction cone by tuning up weighting parameters. Simulation results showed the effectiveness of the approach for solving on line the grasping force optimization problem during a manipulation task.

References

[1] L. Han and J. Trinkle, "The instantaneous kinematics of manipulation," Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1944–1949, May 1998.

[2] D. Montana, "The kinematics of multi-fingered manipulation," IEEE Transactions on Robotics and Automation, vol. 11, no. 4, pp. 491–503, Aug. 1995.

[3] Y. Nakamura, K. Nagai, and T. Yoshikawa, "Mechanics of coordinative manipulation by multiple robotic mechanisms," Proceedings of the IEEE International Conference on Robotics and Automation, pp. 991–998, Apr. 1987.

[4] T. Yoshikawa and K. Nagai, "Manipulating and grasping forces in manipulation by multifingered robot hands," IEEE Transactions on Robotics and Automation, vol. 7, no. 1, pp. 67–77, Feb. 1991.

[5] V. Kumar and K. Waldron, "Force distribution in closed kinematic chains," Proceedings of the IEEE International Conference on Robotics and Automation, pp. 114–118, 1988.

[6] Y. Nakamura, K. Nagai, and T. Yoshikawa, "Dynamics and stability in coordination of multiple robotic mechanisms," The International Journal of Robotics Research, vol. 8, no. 2, pp. 44–61, Apr. 1989.

[7] M. Cherif and K. Gupta, "Planning quasi-static motions for re-configuring objects with a multi-fingered robotic hand," Proceedings of the IEEE International Conference on Robotics and Automation, pp. 986–991, Apr. 1997.

[8] L. Han and J. Trinkle, "Dextrous manipulation by rolling and finger gaiting," Proceedings of the IEEE International Conference on Robotics and Automation, pp. 730–735, May 1998.

[9] A. Bicchi, "On the closure properties of robotic grasping," The International Journal of Robotics Research, vol. 14, no. 4, pp. 319–33, Aug. 1995.

[10] N. Brook, M. Shoham, and J. Dayan, "Controllability of grasps and manipulations in multi-fingered hands," IEEE Transactions on Robotics and Automation, vol. 14, no. 1, pp. 185–192, Feb. 1998.

[11] M. Horat and S. Wu, "On the grasping stability and optimality under external perturbations," Journal of Robotic Systems, vol. 16, no. 9, pp. 509–526, 1999.

[12] F. Cheng and D. Orin, "Efficient algorithm for optimal force distribution - the compact-dual lp method," IEEE Transactions on Robotics and Automation, vol. 6, no. 2, pp. 178–187, Apr. 1990.

[13] W. Chung and K. Waldron, "Force distribution by optimizing friction angles for multifinger system," IEEE International Conference on Robotics and Automation, pp. 717–722, 1993.

[14] M. Buss, H. Hashimoto, and J. Moore, "Dextrous hand grasping force optimization," IEEE Transactions on Robotics and Automation, vol. 12, no. 3, June 1996.

[15] L. Han, J. Trinkle, and Z. Li, "Grasp analysis as linear matrix inequality problems," IEEE Transactions on Robotics and Automation, vol. 16, no. 6, Dec. 2000.

[16] G. Liu and Z. Li, "Real-time grasping force optimization for multifingered manipulation: Theory and experiments," IEEE/ASME Transactions On Mechatronics, vol. 9, no. 1, Mar. 2004.

[17] R. Smith, "Open dynamics engine," http://ode.org, 2003.

[18] C. Remond, V. Perdereau, and M. Drouin, "Design of a hybrid external structure for a multi-fingered hand," 6th IFAC Symposium on Robot Control, Sept. 2000.

[19] V. Perdereau and M. Drouin, "Hybrid position/force control for multi-fingered robot hand," Proceedings of the IEEE-SMC IMACS multiconference on Computational Engineering in Systems Applications, Apr. 1998.