Planning 3D regrasp operations with a polyarticulated mechanical hand

J.P. Gazeau, S. Zeghloul, A. Fernandez, M. Arsicault
Laboratoire de Mécanique des Solides - UMR 6610 CNRS - Université de Poitiers
SP2MI - BP 30179 – 86962 CHASSENEUIL FUTUROSCOPE CEDEX - FRANCE

Abstract : - This paper presents a method to solve the regrasping problem in the context of object manipulation with a mechanical hand. This problem is met during an object manipulation when one finger reaches the boundary of its workspace or when a collision between parts occurs. Our method synthesizes a new feasible grasp when this problem appears by a sequence of grasps involving the fourth free finger. The grasp synthesis problem is formulated as an optimization problem involving many parameters solved by using genetic algorithms.

In order to validate this approach, basic manipulation tasks are detailed in simulation. The efficiency of this method is also demonstrated with two examples : a rotation and a translation of an object inside the hand.

Key-Words : - Manipulation task - Mechanical hand - Regrasp - Genetic Algorithm - Path Planning - Fingertip

1 Introduction

The development of multifingered robot hands as multipurpose gripping devices in an automated system requires grasp synthesis and a global manipulation strategy including fingertip path planning and stability computation.

Research at the “Laboratoire de Mécanique des Solides” (LMS) join within the framework of these developments. The LMS carried out several studies in the theory of grasping and in the development of specialized grasping devices .

In 1996, a mechanical hand with 4 fingers and 16 degrees of freedom was carried out (cf. figure 1). The LMS hand based on an anthropomorphic design was built for the specific application : manipulation of objects with the fingertips.

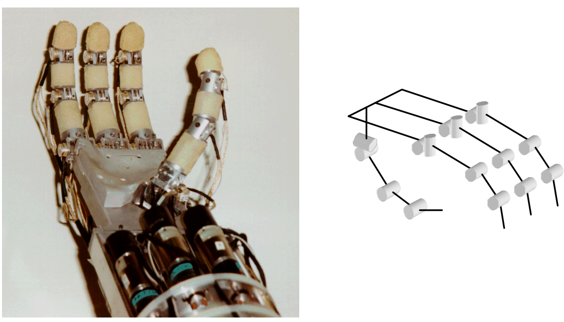


Fig. 1 : The LMS mechanical hand with 16 dof

The LMS developed a robust planning method of objects manipulation with fingertips. This approach is described in [2]. The strategy used for planning is based on a geometrical approach, and on model of contact with friction between the grasped object and the finger. The figure 2 illustrates the fingertips path planning results for an object rotation task.

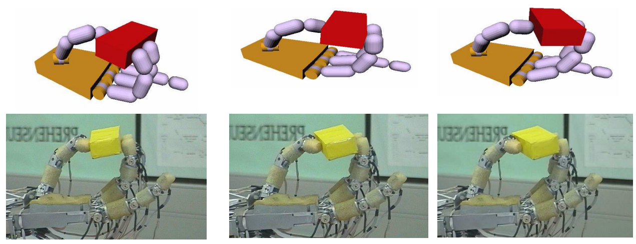


Fig. 2 : A manipulation task : prism rotation

A new efficient algorithm to compute finger forces involved in three-dimensional objects grasp was also developed as described in [3]. The proposed approach, based on quadratic programming and inequality constraints linearization, transforms the initial optimization problem into a minimal distance calculation. The main advantage of this method is its computational speed. It’s completely suited for real time applications and online manipulation tasks.

The objective now is to build a global strategy including path planning, object stability and grasp synthesis. The regrasping problem is part of the grasp synthesis problem.

This paper describes a method able to provide a solution to the regrasping problem in the context of manipulation with mechanical hands.

Regrasping an object can be achieved by placing the object on a table and picking it again with another grasp. This is not the way we want to use in our approach. With dextrous hands, which have more than three fingers, the regrasping operations can be done by repositioning the fingers on the object. This means, the fingers must change their contact points keeping the object in hand.

Several studies concerning the regrasping problem have been developed in the litterature. Many developpements focused on assembly tasks as described in [4][5], however just a few studies deal with regrasping operations with mechanical hands and manipulation. Hong et al. [6] proposed the use of finger gaiting, i.e. a periodic movement of the fingers to form a new grasp with fingers lying within their workspace limits, to find a new grasp when one finger reaches a boundary limit,. Han et Trinkle [7] use the same approach to plan general manipulation tasks, i.e. first they move the object until one finger reaches its joint limit, then they use finger gaiting to form a new force closure grasp and they repeat the procedure if needed.

In this paper, we propose a new method to generate regrasping points on the object when a boundary limit appears. Thus the fourth finger is used to reach this new grasp without changing the object position and orientation inside the hand. With such a strategy, the object manipulation continuity is maintained and object motions with the desired amplitude can be achieved.

2 Problem Formulation

2.1 General overview

When a boundary limit is reached by finger or when a collision between finger and object appears during manipulation, it's necessary to regrasp the handled object in order to achieve the task.

Thus we have to determine a new feasible grasp. This new grasp is defined by locating 3 new contact points on the object. The object position and orientation inside the hand doesn't change. Once this grasp is defined, a sequence of grasps involving the fourth finger is done to reach the new grasp. This action is made by finger gaiting.

We propose to find a new optimal grasp. First of all, the grasp must be force-closure. We make the assumption that the contacts are hard-finger contacts with friction. The grasp will be force-closure if any external forces and torques exerted on the grasped object can be balanced by the fingers contact forces. The force closure grasp constraint is checked by the algorithm given by Li in [8]. Based on geometrical analysis, the algorithm is quite simple and needs a few algebraic calculations to compute force-closure grasps.

Three criteria for the evaluation of the quality of the grasp are used in the present approach :

- The minimization of the grasping forces;
- The maximization of the manipulability;
- The maximization of the distance from the joint limits.

These criteria are defined below.

2.1.1 The optimization criteria

We introduce the following criterion T_1 for the grasping forces minimization :

$$T_1 = \frac{1}{2} F^T \cdot F$$

where $F = (f_1, f_2, f_3)$ and f_i is the norm of the force vector of grasping finger i. This term will prevent the fingertip force for each finger from being excessive.

Another important factor is the manipulability as described by Yoshikawa in [9]. In this context, manipulability is the ease of arbitrarily changing the position and orientation of the fingertip in its workspace. The manipulability measure w is given by the following relation :

$$w = \sqrt{\det\left(JJ^T\right)}$$

where J(q) is the Jacobian matrix. J(q) is defined with the relation between the velocity vector v and the joint velocity $\dot{q}$:

$$v = J(q) \cdot \dot{q}$$

Thus we need to maximize the manipulability criterion T_2. The factor T_2 is given by :

$$T_2 = w_1 + w_2 + w_3$$

where w_i is the manipulability of finger i.

The third criterion is used to locate each finger joint as far as possible from its limit. With a such criterion, the joint displacement amplitude will be larger. This criterion is given by :

$$T_3 = \sum_{i=1..3,\, j=1..4} \left(q_{ij} - q_{avg\,ij}\right)^2$$

where q_{ij} is the joint value for finger i and joint j ; $q_{avg\ ij}$ is the average value for joint j of grasping finger i. We note that each finger has four degrees of freedom.

Thus we have to use these three criteria in the same objective function as described in section 3.1.

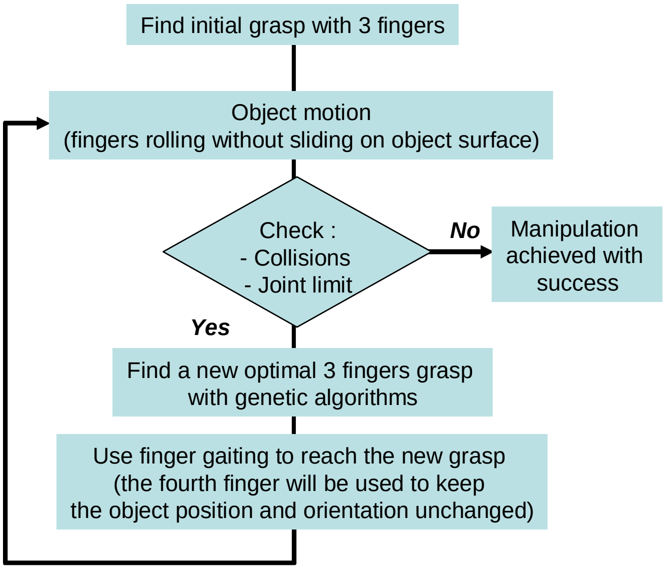
So we choose an objective function F_{fit} with the following form :

$$F_{\text{fit}} = A_1 \cdot T_{1\text{norm}} + A_2 \cdot T_{2\text{norm}} + A_3 \cdot T_{3\text{norm}}$$

where A_1, A_2, A_3 are weighting coefficients and $T_{i\text{norm}}$ is normalized value of T_i .These parameters are defined later (see section 3.1). The optimization of objective function will also lead to the desired grasp.

2.1.2 Resolution

The searched grasp must optimize these three criteria. In order to solve this problem with different parameters, we use genetic algorithms (GAs). Then the global manipulation task with a mechanical hand will be planned as follows :



2.2 GAs

The optimization problem with GAs needs first to define a population of chromosomes of individuals. This population has to describe the grasps that will be candidate solutions to the optimization problem. The evolution starts from a population of randomly generated grasps. In each generation, the fitness of every individual in the population is evaluated. Multiple individuals are selected from the current population (based on their fitness), and modified (mutated or recombined) to form a new population. The new population is then used in the next iteration of the algorithm.

So we need to define :

- a genetic representation of the solution domain,
- a fitness function to evaluate the solution domain; this fitness function in GAs approach corresponds to the objective function (section 2.1.1) in a classical optimization scheme.

Thus once we define the genetic representation and the fitness function, GAs initialize a population of solutions randomly, then improve it through repetitive application of evolution operators.

2.3 Genetic representation

The objective is to determine a 3 fingers grasp by using GAs, making the object manipulation possible out of the fingers limits. We have to define first the genetic representation of a grasp.

We make the assumption that we manipulate objects with 3 fingers; the fourth finger is used only for finger gaiting. So it's necessary to define the parameters that characterize the grasp.

The grasp is defined by locating the three contact points on the object with the points P1, P2, P3. With these three points, a plan called grasp plan is defined.

The point P1 (thumb contact with object) is opposed to other fingers P2 and P3, as described by Cutkosky in [10]. When a grasp is randomly generated, we first generate a point I_{AL} inside the object. We name this point, the center of grasp. The next step consists in defining a grasp plan including the point I_{AL}.

The orientation of this plan is defined by pitch, roll and yaw angles φ_1, φ_2, φ_3 in the object frame R_{OB}. The frame R_P (X_P,Y_P, Z_P) attached to the grasp plan *P* is computed with these angles corresponding respectively to rotations around axis X_{OB}, Y_{OB} and Z_{OB} as shown on figure 3.

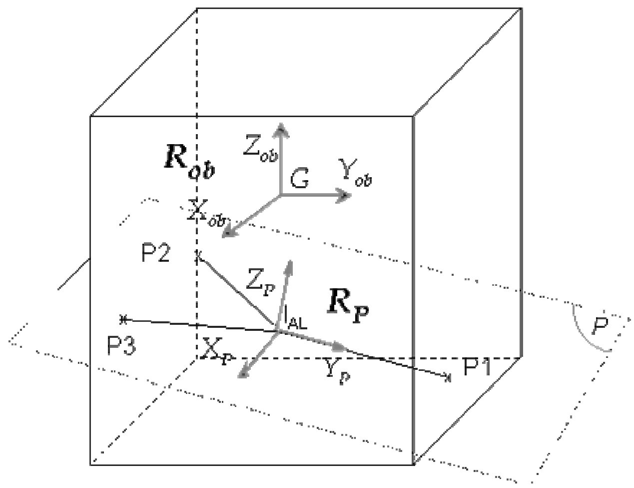


Fig. 3 : Grasp parameters

Practically the normal to the plan *n* (equal to Z_P) is also randomly generated. Then we choose a line with point I_{AL} on this plan. The vector Y_P is placed on this line. The vector X_P is then given by : $X_P = Y_P \wedge Z_P$.

The point P1, corresponding to the thumb's position on the object, can be computed now; it's the intersection from the line defined by the point I_{AL} and by the vector Y_P with the object.

The figure 4 presents the configuration of the points P2 and P3. The points P2 and P3 are opposite to point P1. So the line I_{AL}P1 is defined as the median line between points P1 and P2. The angle α is introduced to determine the configuration of those points as shown on figure 4.

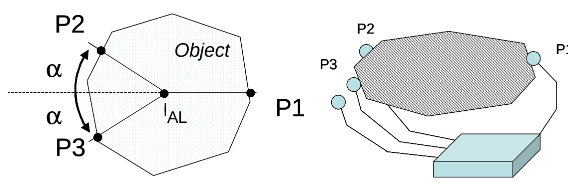


Fig. 4 : Grasp parameters : angle α

We note on figure 4 that if we change this parameter α, the points P1, P2 corresponding to both opposite fingers will be more or less distant. The limit value for α depends on the amplitude of the finger's abduction-adduction movement.

A grasp is now completely defined with the position of the center of grasp and with the orientation of the grasp plan. By using this formulation, we have the seven following parameters that describe the grasp :

- X, Y, Z : position of the center of grasp in the object frame;
- φ_1, φ_2, φ_3 : the orientation of the grasp plan defined by pitch, roll and yaw angles in the object frame;
- α : angle between the lines of action.

Figure 5 illustrates this coding of the individuals. The chromosomes of individuals are represented by these seven real-valued numbers. For the generation of the initial population, we thus generate randomly points inside the object, as well as an orientation and an angle α. Thus the objective is to determine with GAs the best individual, or the best grasp, after a given number of population generations.

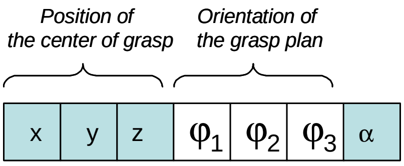


Fig. 5 : Coding of individual

3. Problem Resolution

3.1 Fitness function

For each generation of the GAs algorithm, the fitness of every individual in the population is evaluated, multiple individuals are selected from the current population (based on their fitness). the fitness function used in this paper uses the three criteria described in section 2.1.1. This function is given by :

$$F_{fit} = A_1 \cdot T_{1norm} + A_2 \cdot T_{2\,norm} + A_3 \cdot T_{3\,norm}$$

where A_1, A_2 and A_3 are the weighting coefficients of the fitness function, and T_{1norm}, T_{2norm}, T_{3norm} are the normalized factors relating to the 3 criteria. The criterion T_i are normalized by using the following relations :

$$T_{inorm} = \frac{T_{i\max} - T_i}{T_{i\max} - T_{i\min}} \text{ for i=1,3 and } T_{2norm} = \frac{T_2 - T_{2\min}}{T_{2\max} - T_{2\min}}$$

where T_{imax} and T_{imin} are limit values for each criterion.

3.2 GA parameters

GA evolution parameters are the following ones : the population size N_{Pop}, crossover probability P_C, mutation probability P_M, the maximum number of generations G_{max}. These parameters were defined by testing several population sizes (4,8,12,24,36,72 individuals) and different probabilities with GA and two classical objects (cf. figure 6). Finally, we chose these values :

N_{Pop} = 24 individuals ; P_C = 1; P_M = 25%; G_{max} =1000

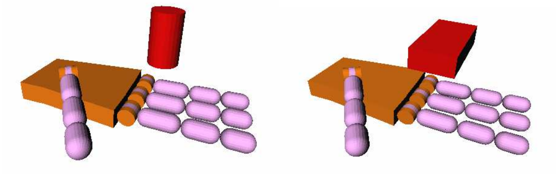


Fig. 6 : Objects used for GAs parameters adjustment.

We used the experimental design method [11] for the determination of the weighting coefficients of the fitness function. We obtain the values from table I. We note that the coefficients are nearly the same for a parallelepiped and for a cylinder.

	Cylinder	*Parallelepiped*
A_1	0.467	0.449
A_2	0.411	0.456
A_3	0.475	0.401

Table I : Weighting coefficients determination

So we choose the following expression for fitness function :

$$F_{fit} = 0.467 \cdot T_{1norm} + 0.411 \cdot T_{2\,norm} + 0.475 \cdot T_{3\,norm}$$

Thus for the best grasp the fitness function value corresponds to the smallest value.

3.3 The algorithm for optimization process

The objective of the optimization process based on GAs is to find a new grasp when a collision or a joint limits occurs during manipulation.

During the grasp optimization process, different conditions are checked. These conditions are the following ones :

- The grasp is force-closure ;
- The grasp is out of the joint limits;
- There's no collision between parts; we verify this condition by using a fast algorithm for distance calculation between convex objects developed by LMS in [12];
- Avoid contacts with forbidden object facets : for example for a cup, it's forbidden to put the finger on top of the cup as shown on figure 7. We characterize these facets in the manipulation definition.

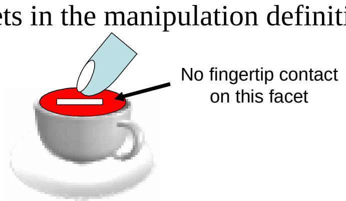


Fig. 7 : Non authorized facets : example of a cup

The algorithm using GAs is described by this pseudo-code :

```
Generate randomly 24 individuals (Initial Population)
 For each individual Generate ramdomly 7 parameters
     If individual i verifies the different conditions Endif
     Else restart Generation
  /* Optimization process with GAs */
  While number of generations < Gmax Or Ffit not stable
     Evaluate Individual
      Sort Individuals according to the fitness value
      Select 12 individuals
     /* Crossover */
          While New population size < 24 Do
               Crossover between 2 individuals
               If the 2 new individuals verify the conditions
               Then  Increment New population size
         End While
      Select the 6 best individuals (elitism)
     /* Mutation  */ (with 18 remaining individuals)
          if the individual verifies the conditions after
           mutation then the individual is mutated
                        else the  individual remains unchanged
End While
```

The application of this approach for a cylindrical object's manipulation planning is described in following section. The studied tasks are the translation and the rotation of a cylindrical object.

4. Results

4.1 Grasp synthesis examples

The object is a cylinder; its diameter is 30mm and its weight is 20g. Object position and orientation with respect to the hand's frame are given. If we choose to minimize only the grasping forces in the optimization process, the fitness function is given by:

$$F_{fit}= T_{1norm} \text{ (A2=A3=0 and A1=1)}$$

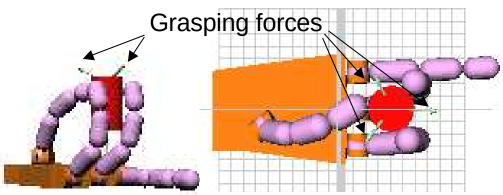


Fig. 8 : Cylinder grasp synthesis using GAs and forces minimization

Thus GAs compute the grasp illustrated in figure 8. We note that the contact points define an equilateral triangle as shown on figure 9, and the three fingertip forces have the same norm equal to 0,2N.

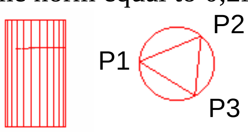


Fig. 9 : Grasp plan and contact points on cylinder – GAs optimization with $F_{fit}= T_{1norm}$ (A2=A3=0 and A1=1)

If we choose to maximize only the manipulability in the optimization process, the fitness function is given by:

$$F_{fit}= T_{2norm} \text{ (A1=A3=0 and A2=1)}$$

Thus we obtain a grasp configuration far from the singularities as shown on figure 10.

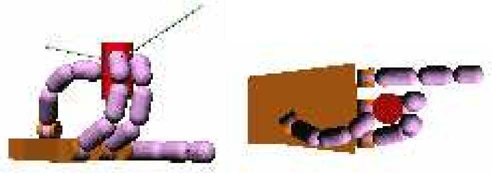


Fig. 10 : Cylinder grasp synthesis using GAs and manipulability maximization

Figure 11 shows the corresponding contact points position on the grasped cylinder.

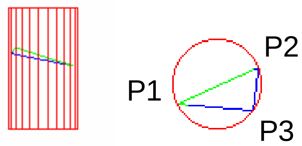


Fig. 11 : Grasp plan and contact points on cylinder – GAs optimization with $F_{fit}= T_{2norm}$ (A1=A3=0 and A2=1)

An interesting result is the convergence of fitness function, After 30 generations, the fitness function is stabilized, so optimization process is achieved.

4.2 Objects manipulation with regrasping

In this section, we present complete manipulation tasks with regrasping and finger gaiting using GAs optimization process. The fitness function used for the next examples is the following one :

$$F_{fit} = 0.467 \cdot T_{1norm} + 0.411 \cdot T_{2\,norm} + 0.475 \cdot T_{3\,norm}$$

4.2.1 Cylinder rotation

The first example shows a cylinder rotation around vertical axis. This manipulation is illustrated with figure 13. A first joint limit occurs after a cylinder rotation of 24°. Thus finger gaiting starts to reach the grasp computed by GAs optimization process (the new free grasp on figure 13). The figure 13 shows the sequence of fingers motion used to reach the grasp computed by GAs optimization process. During finger gaiting, we verify that the intermediate grasp are force-closure grasps.

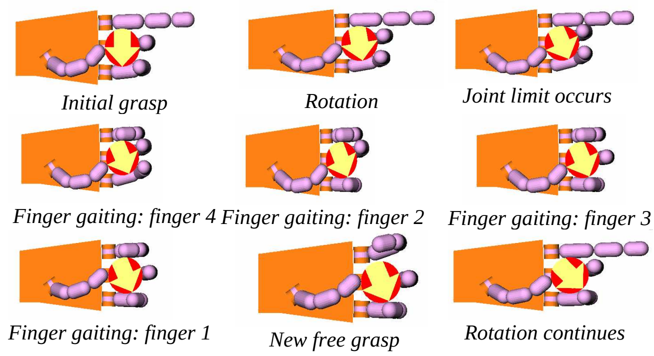
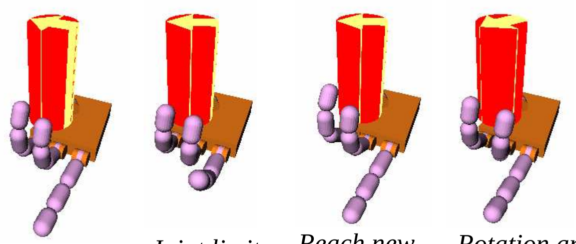


Fig. 13 : Rotation of a cylinder with regrasp

The finger gaiting described on figure 13, is based on Han and Trinkle approach [7]. It's called *finger rewind* in [6]. It relocates the limiting fingers back to their workspace. The figure 14 shows a different view for this manipulation with main steps. If a new joint limit occurs, we repeat new grasp synthesis with GAs and new finger gaiting; with such a strategy the desired amplitude for the rotation can be done.

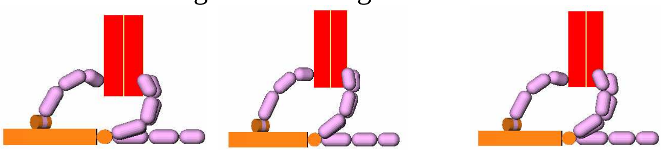


Initial grasp *Joint limit occurs* *Reach new free grasp* *Rotation and new joint limit*

Fig 14 : Main steps in the rotation of a cylinder

4.2.2 Cylinder translation

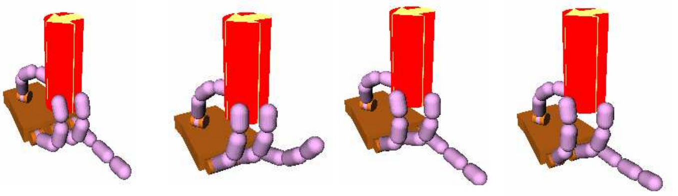
The second example concerns the vertical translation of a cylinder. The main steps of the manipulation are illustrated on figure 15 and figure 16.



Initial grasp *Joint limit occurs* *Regrasp*

Fig. 15 : Translation of a cylinder with regrasp

Once the free new grasp is reached by using finger gaiting, the translation can continue until a new joint limit appears.



Initial grasp *Joint limit occurs* *Reach new free grasp* *Translation and new joint limit*

Fig 16 : Main steps in the translation of a cylinder

5. Conclusion

In this paper, we show the general regrasping strategy for object manipulation with mechanical hands. This strategy is based on finger gaiting and grasp synthesis using GAs. This method determines the regrasping points and working fingers during manipulation according to the object shape and size. The new grasp is obtained by optimizing an objective function including three critiria. We show effectiveness of the proposed generation method for regrasping motion with good simulation results. The proposed approach is completely suited for realistic manipulation with mechanical hands. This approach is currently developed for experiments with the LMS mechanical hand. Experimental results with the LMS hand will be detailed in a future paper.

References:

[1] J.P. Gazeau, S. Zeghloul, M. Arsicault, J.P. Lallemand, The L.M.S. Hand : force and position controls in the aim of the fine manipulation of objects , *IEEE ICRA 2001*, 2001, Séoul, Corée.

[2] J.P. Gazeau, S. Zeghloul, M. Arsicault, J.P. Lallemand, Manipulation with the LMS mechanical hand – Strategy for fingertip manipulation task , *Journal Européen des Systèmes automatisés*, Vol. 36, N°9/2002, pp.1205-1219.

[3] J.P. Gazeau, S. Zeghloul, G. Ramirez, « Manipulation with a polyarticulated mechanical hand : a new efficient real-time method for computing fingertip forces for a global manipulation strategy», Robotica, Volume 23, pp. 479-490, 2005.

[4] S.A. Stoeter, S. Voss, P. Papanikolopoulos, H. Mosemann, "Planning of regrasp operations", *Proc. of the 1999 IEEE ICRA*, May 1999, Detroit.

[5] Rapela D.R., Rembold U., Kuchen B., "Planning regrasping operations for dextrous hand assembly task", *Journal of intelligent and Robotic Systems*, Vol. 33, N°3, 2002, pp.231-266.

[6] Hong J.W., Lafferriere G., Mishra B., Tang X.L.,"Fine manipulation with multifinger hand", IEEE Proc. Of Int. Conf. On Robotics And Automation, pages 1568-1573, 1990.

[7] Han L., Trinkle J.C., "Dextrous manipulation by rolling and finger gaiting", *Proc. Of the IEEE Int. Conf. On Robotics And Automation*, 1998.

[8] [Li 2003] J.W. LI, H. LIU , "On Computing Three-Finger Force Closure grasps of 2-D and 3-D objects ", *IEEE Transactions on robotics and Automation*, Vol. 19, No. 1, February 2003.

[9] T. Yoshikawa, "Manipulability of robotics mechanisms", International Journal of Robotics Research, 4(2), pp. 3-9, 1985.

[10] Cutkosky M.R., "On grasp choice, grasp models, and the design of hands for manufacturing tasks", *IEEE Transactions On Robotics And* Automation, Vol. 5.

[11] Goupy J., « Pratiquer les plans d'expériences », Dunod / *L'Usine Nouvelle - Collection Technique et Ingénierie*, 568 pages – 2005, ISBN : 2100042173.

[12] S. Zeghloul S., P. Rambeaud, « A fast algorithm for distance calculation between convex objects using the optimization approach », Robotica 14, pp. 355-363, 1996.