Enhancing in-hand dexterous micro-manipulation for real-time applications

Joël BAFUMBA LISELI[1], Redwan DAHMOUCHE[1], Pardeep KUMAR[1], Jean-Antoine SEON[2], Michaël GAUTHIER[1]

***Abstract*—This paper presents a new approach of planar trajectory generation for automated in-hand dexterous manipulation of miniaturized objects. The proposed method aims at improving the efficiency of the previous method [17] to be able to perform real-time in-hand manipulation trajectories generation. The main idea behind this new method is the representation of the configuration space as a set of stable rotations instead of stable grasps as it is usually done. The consequence of this representation is a more compact space that encapsulates more information. The developed algorithm that uses this approach is able to generate optimal trajectories to manipulate complex objects in less than 0.1 s, which represents a reduction in the processing time between 10^2 and 10^5 compared to the previous method.**

***Index Terms*—Micro-manipulation, Manipulation Planning, Dexterity, Grasping, In-hand manipulation.**

I. Introduction

Currently, systems integrate more and more functionalities into smaller volumes thanks to embedded micro-components. The assembly of those components requires accurate and precise micro-manipulation systems. Translation positioning already enables achieving nano-scale resolutions thanks to robotized systems. However, precise rotation in micro-scale is still a challenge since the center of rotation is difficult to control [1].

Dexterous micro-manipulation is a promising way to address this problem allowing the manipulation of a large variety of objects with a single hand. In macro-scale, anthropomorphic robotic hands have been used to manipulate object mimicking human hands [2]. At the micro-scale, it is not possible nor suitable to design anthropomorphic micro-hands. Instead, simple robotic micro-grippers are preferred. Indeed, micro-grippers have usually two fingers, each one having one or two Degrees of Freedom (DoF) at most [3], [4]. Over the three last decades, several researches have been done in this area aiming to use those micro-grippers to automate micro-assembly tasks. In [5], an automated pick and place task of micron blocks using two probes ("ortho-tweezers") to perform force feedback controlled assembly operations was presented. Similarly, Wason et al [6] developed an algorithm for the automated construction of 3D structures using only planar micro fabricated parts. Instead of using grippers, they chose a multiple sharp-tipped probes to coordinate the manipulation of the parts by using vision feedback.

[1]AS2M Department, FEMTO-ST Institute, Univ. Bourgogne Franche-Comté, Besançon, France. (firstname.lastname@femto-st.fr, [2]Department of Electrical Engineering, Aalto University. (jean-antoine.seon@aalto.fi)

Zhou et al. [7] presented a 6-DOF micro-gripper able to perform fully automated pick-and-place and handling operations of $300 \times 300 \times 100\mu m$ sized micro-components. In these methods, fingers rolling on the object during the rotation was neglected and the contact fingers/micro-object maintained during all the handling, which limits the rotation amplitude. To enlarge the rotation amplitude, Brazey et al. [8] introduced the design of a robot hand that achieves down to $120\mu m \times 120\mu m$ cubes rotations through rolling without sliding [9] and finger gaiting manipulation strategies [19]. Thanks to this design manipulation approach, it was possible to perform large rotations (over $\pm 180°$). A similar approach was developed by Seon et al [17] to automate in-hand manipulation of arbitrary shaped planar miniaturized-objects (Fig.1). Whereas various approaches try to minimize the adhesion effect to be able to use macro-scale manipulation principles [11], the alternative approach presented in [18] showed that exploiting these adhesion forces enhances micro-manipulation capabilities and effectiveness.

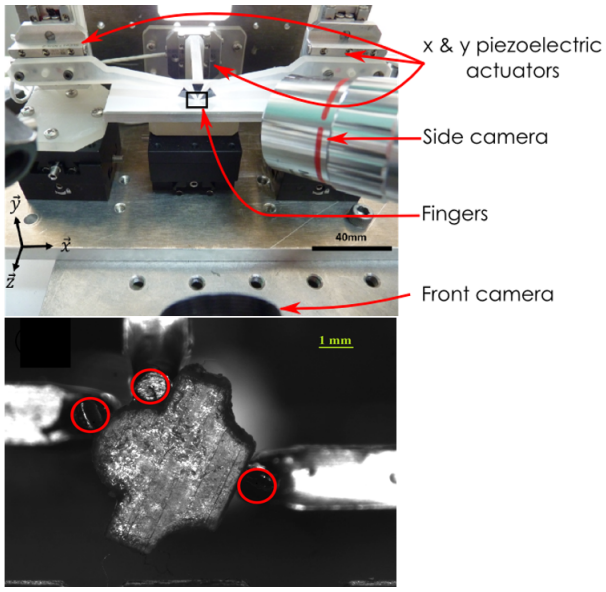

Fig. 1: A dexterous robotic micro-manipulation system using three translating fingers [8].

The proposed method consists in testing off-line all the possible configuration grasps and then exploring the configuration space on-line to generate the in-hand dexterous micro-manipulation trajectories. Rolling without sliding and finger-gaiting manipulation strategies were adopted to perform large motions.

However, since the grasp space is large (about 4×10^6 nodes), generating the optimal trajectories is time consuming which makes it not suitable for real-time applications.

The solution usually adopted in dexterous manipulation at the macro-scale is to sub-sample the configuration space to shrink it and use algorithms such as Rapidly Exploring Random Trees (RRT) [12], [13] or probabilistic approaches [14] to navigate within the sub-space. However, this sampling based approach has two main drawbacks. The first one is that existing solutions may not be found since not the whole configuration space is represented. The second one is that post-treatments are necessary to verify that the generated path is valid by checking if all the nodes are connected. This means that a trajectory really exists between one configuration to another [15].

In this article, we present a new approach that significantly reduces the time for trajectories generation without losing optimality (minimal cost trajectories) and completeness (if a solution exists then it will be found). The main idea behind this approach is to change the in-hand micro-manipulation paradigm from a succession of stable grasps to a sequencing of possible continuous trajectories. Formulating the problem on trajectories basis instead of stable grasps significantly reduces the configuration space without losing information. The gain in efficiency and processing time allows running and executing the planning algorithm in real-time.

II. Modeling and Background

In the industrial context, the manipulated micro-objects are usually planar since they are often manufactured from wafers. Thus, we focus in this paper on arbitrary shaped planar micro-objects manipulation. The micro-hand used for manipulations is made of three cylindrical fingers translating in a plane as it is shown in Fig.1. The objects are then quasi-statically manipulated in a plane considering rolling without sliding and finger gaiting manipulation strategies [19]. We also consider that the fingers displacement ranges are larger than the manipulated objects dimensions so the hands fingers can have access to any part of the object.

Contrary to macro-scale manipulation that usually focuses on generating stable grasps for unknown objects, the shapes of the objects to be manipulated in micro-scale are usually known through their CAD models. The prior knowledge of the objects geometries allows performing preprocessing tasks, such as finding stable grasps, that simplify the online dexterous motion planning such as in Seon et al [16].

A. Fingers/Micro-object Contact Modeling

First, let us consider the contact forces between the hand's fingers and the manipulated object. Since both elements are modeled as rigid bodies, each contact applies a pure force. Considering the Coulomb friction model, there is no slippage on the contact point since the applied force lays inside the friction cone (see Fig.2). The presence of adhesion force can simply be modeled as an additional force which is superposed to the force applied by the finger to the finger at the contact point. The non-slippage condition can then be formulated as follows:

$$f_t < \mu \cdot (f_n + f_{po}) \tag{1}$$

where f_t and f_n are respectively the tangential and the normal components of f, f_{po} is the pull-off force and μ is the friction coefficient.

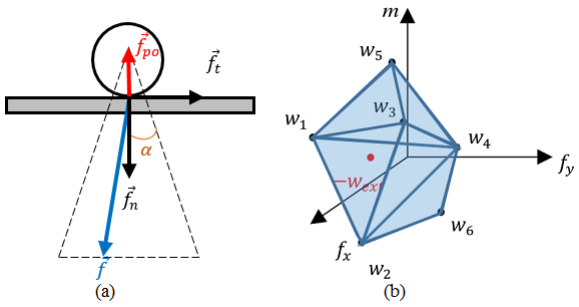

Fig. 2: (a) The modified Coulomb friction cone in micro-manipulation in which it is possible to observe the pull-off force f_{po} (b) Representation in the wrench space of a contact point finger/object related to the modified Coulomb friction cone.

The obvious consequence of this property is that the fingers can push but also pull the object as long as the pull force is less than the adhesion force (pull-off force). The other consequence is that origin of the friction cone is shifted as shown in Fig.2.

The applicable forces on the object can also be represented as a linear combination of three elementary forces as follows:

$$f = \alpha_{po} \cdot f_{po} + \alpha_l \cdot f_l + \alpha_r \cdot f_r \tag{2}$$

where f_l and f_r are the two limits of the applicable forces as represented in Fig.2(a). α_{po}, α_l and α_r are real coefficients. The Coulomb law imposes that all the coefficient must be positive and $\alpha_{po} < 1$.

B. Equilibrium Grasps

The well-known static equilibrium condition of a rigid body is that the resultant of all the forces and all the moments is null. Considering the equilibrium in the plane, the resulting wrench for each contact are composed of two forces and one moment:

$$W = \begin{bmatrix} f_x & f_y & m_z \end{bmatrix}^t \tag{3}$$

The static equilibrium condition can then be written as :

$$\sum_{i=1}^{n} W_i + W_{ext} = 0 \tag{4}$$

where W_i represents the wrench caused by each finger and W_{ext} is an external wrench applied on the object.

Since each contact force can be decomposed into several components as shown in Eq. 2, a wrench can be associated

with each component of the force applied by each finger (Fig.2(b)). The geometrical condition of equilibrium is that the convex hull of the set of wrenches contains $-W_{ext}$ as illustrated in Fig.2(b).

III. Stable Grasp Space Generation

First, one can note that translating the object in the plane can simply be done by translating all the fingers. The contact positions on the object are not modified so the grasp stability is not affected. Since translating the object has no effect on the grasp stability it will not be considered in the following formalism.

To obtain the stable grasp space which depends on the objects shape, the geometry of the object is sampled from the CAD model for instance.

One of the drawbacks of this approach is that the configuration space could be excessively large. To leverage this effect, we chose to attach the working frame to the object and use the curvilinear abscissa on the object's contour to identify the position of the fingers on the object as illustrated in Fig.3. The dimensionality of the problem is then $l^3 \times m$ where l is the number of samples, 3 represents the number of fingers and m the number of samples defining the object orientation.

Note that even if the space is much smaller than it would be by choosing Cartesian coordinates ($l^6 \times m$), it is still a relatively large space. For instance, having 100 samples for each dimension would result in a space having 10^7 possible configurations. Note also that the situation where a finger is not in contact with the object is taken into account by adding one sample to the possible contact positions on the object for each finger.

The stability of each grasp is then checked using the stability criteria defined in the previous section. This defines all the grasps that can be adopted to manipulate the object. However, not all the configurations are checked since some contact points corresponding to vertices and double contact configurations between the fingers and the object are avoided. In addition, collision configurations are also considered to restrict stable grasp space. The stable grasp space is restricted to realistic and feasible configurations as shown in Fig.3. This space can be represented as a map as shown in Fig.4.

To pick-up a micro-object from a substrate, the contact positions on the object that are not accessible (typically between the object and the substrate) are inhibited. In addition, to detach the object from the substrate a force has to be applied in order to overcome the pull-off force between object and substrate.

Similarly, in finger gaiting sequence, detaching a finger applies a perturbation force that the remaining fingers must resist to (see Fig.5). This subset of grasps is also identified so any grasp reconfiguration that requires removing a finger has to belong to this subset.

IV. In-Hand Manipulation Planning

For three fingers in-hand manipulation in the plane (Fig.6), four parameters are sufficient to describe the grasp configuration: the object orientation, and the three curvilinear abscissas $N_i = [\ \theta\ d_1\ d_2\ d_3\ ]$ of the contact positions on the object. To plan a trajectory, we proposed in [17] to construct a graph by making connections between nodes (stable grasps). Each connection means that it is possible to pass from one stable grasp to another.

As stated in the previous section, the possible elementary actions are, rotating the object, placing a finger, and removing a finger from it. Given these elementary operations, all the admissible connections between admissible grasps are created. These elementary operations define all the possible connections between the nodes.

In our previous method [17], a cost was associated to each elementary operation and A^* algorithm was used to find the optimal trajectory. In the new method described in this paper,

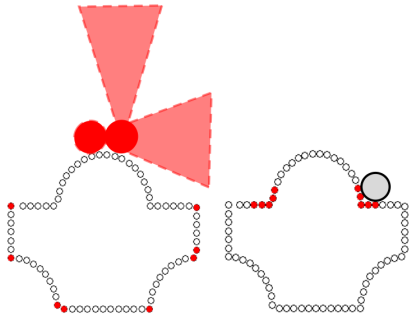

Fig. 3: Non admissible configurations because of (a) collision between fingers/contacts on corners and (b) double contacts configurations.

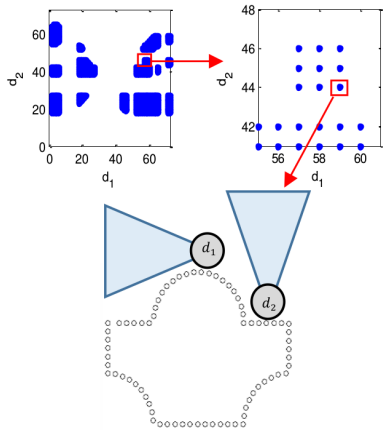

Fig. 4: Stable 2-fingers grasps represented as a function of the contacts positions on the object.

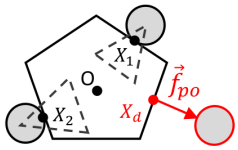

Fig. 5: Detachment of a finger applies a pull-off force that the remaining fingers have to withstand.

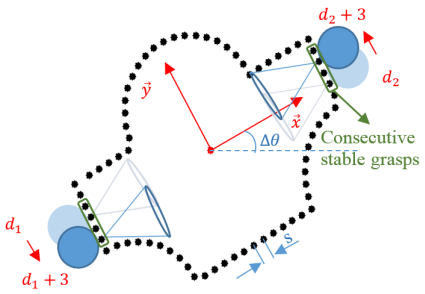

Fig. 6: Rolling without sliding in-hand manipulation kinematics. When rotating the object, the contact points positions move along the object. The limits of the stable rotations can then be determined in both directions.

we propose to integrate the manipulation strategy constraints (rolling contact) to reformulate the problem and simplify its resolution.

By considering the rolling without sliding constraint, there are only two adjacent nodes for each stable grasp, each one corresponds to each rotation direction. Thus, the connection of successive stable grasps represents a possible stable rotation. The stable rotations are bounded by the first unstable configurations encountered in each direction (clockwise and anticlockwise).

A. Stable Rotation Nodes

To improve the efficiency of the dexterous in-hand motion planning algorithm, we propose to integrate the rolling without sliding kinematic constraint into the configuration space representation. Thus, instead of representing the stable grasps we represent stable rotations. Indeed, each stable rotation can be represented by the two grasp configurations at the limit of the rotation. The stable rotations represent the new node of a more optimized configuration space since a large set of successive stable nodes can be replaced by only two coordinates as illustrated in Fig.7.

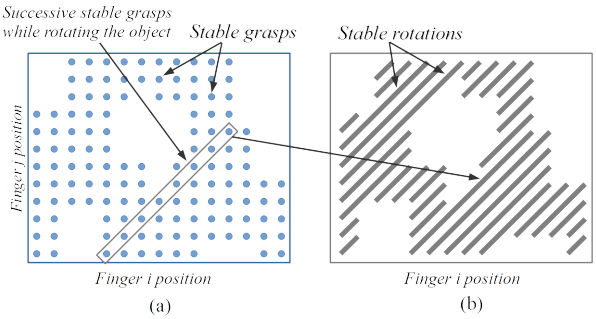

Fig. 7: From stable grasps to stable rotations representation. Rotating the object in rolling without sliding mode imposes a certain trajectory in the configuration space (a). All successive stable grasps can be gathered in a new configuration space: stable rotations which contains much less elements (b).

In addition, this representation improves the information density contained in this configuration space. Indeed, in the previous representation where the nodes were represented by stable grasps, reducing the sampling period between grasps leads to significantly increase the dimension of the configuration space. Indeed, we saw in the previous section that the dimensionality of the problem is $l^3 \times m$, where l is the number of sampled contact points on the object. Thus, just multiplying the number of samples by two would increase the configuration space by eight times. Instead, a node that represents a continuous rotation contains an infinite number of stable grasps without any increase in the configuration space dimension. Obtaining the rotations limits with a better accuracy than the sampling period can also be obtained with algorithms such as dichotomy.

To be able to generate trajectories from this new configuration space representation, a new graph is generated. To do so, the connections between the new nodes have to be defined.

B. Graph Construction

To construct the manipulation graph, all the connections between the nodes have to be defined. Actually, the connections represent the possible elementary actions other than rotating the object which is already contained in the node (pick-up from the substrate, adding a finger and removing a finger). Note also that the connections are not bi-directional. Indeed, going from 2-fingers stable configuration to 3-fingers configuration is always possible; but going from 3-fingers stable configuration to 2-fingers configuration requires that the removal of 3^{rd} finger does not disturb the grasp (because of pull-off force).

Thus, two nodes are connected if they share same contact coordinates of two fingers for a given object orientation as depicted in Fig.8. If the connection corresponds to removing a finger, the other condition to be satisfied is that the grasp has to withstand the detaching pull-off force.

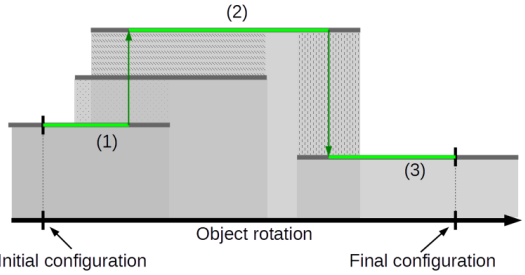

Fig. 8: Representation of connection between nodes via shaded area and patterns, and Example of planning result defining the succession of stable rotations from an initial configuration to a final one through transitions.

C. Motion Planning

Trajectory generation consists in navigating within the generated graph. The A^* algorithm was chosen because it is a complete heuristic graph search algorithm that provides an

optimal path between an initial and a goal node. In addition, it has been successfully used in micro-manipulation [23], micro-assembly [24], and for planning in-hand dexterous manipulations [10].

The A^* algorithm uses a heuristic to guide its search while ensuring that it computes a path with a minimal cost through the nodes (n). Thus, for the determination of the optimal sequence of the nodes, the A^* optimizes a function $f(n)$ which depends on a cost function $g(n)$ and a heuristic function $h(n)$. To be admissible, the heuristic function must never overestimate the cost function $g(n)$.

$$f(n) = h(n) + g(n) \tag{5}$$

The algorithm's objective is to optimize the distances traveled by the fingers during the manipulation which induces an optimized execution time if we suppose a constant velocity displacement of the fingers. The most efficient way to reach the desired configuration is to rotate the object without any finger gaiting. If there is a node that is large enough to reach the desired configuration it should be selected in priority. Privileging rotations instead of finger gaiting can be obtained by affecting a null cost to rotations in the right directions. Since the heuristic must not overestimate the cost function, the latter has also to be null while rotating the object ($h(n) = 0$).

During finger gaiting, the cost function $g(n)$ is defined as the distance traveled by a finger when it is not in contact with the manipulated object and two cases are considered: attaching and detaching a finger. When detaching a finger, the coordinates of the finger are calculated as a fixed distance on the normal to the object surface from the last contact point p_c. This distance is chosen to guarantee that the finger is detached (pull-off forces may induce the finger to bend and still attached) and to ensure that no collision between the removed finger and the object occurs while continuing the manipulation using the other fingers. The cost function is considered set to this fixed distance C.

$$g(n) = C \tag{6}$$

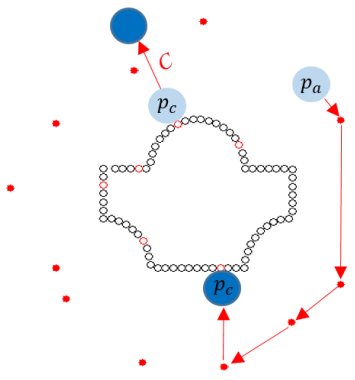

Fig. 9: Finger gaiting actions and their corresponding cost functions (distances).

In the case of adding a finger, the cost function calculates the distance from the current position of the finger p_a to the contact point p_c avoiding all possible collisions between the fingers and the manipulated object (Fig.9).

$$g(n) = dist(p_a, p_c) \tag{7}$$

To guarantee that no collision occurs during off contact fingers trajectories, a surrounding polygon is defined to serve as a path the fingers have to follow to get from their current position to a desired one. This approach prevents sampling the Cartesian space surrounding the object to define the off-contact trajectory, which would be less efficient from the memory size and processing time.

As stated, the A^* algorithm requires defining an admissible heuristic that estimates the cost to the desired orientation. We chose heuristic as the remaining angle from the maximum rotation (θ_n) a node can perform (limit of the stable rotation) to the desired orientation (θ_d).

$$h(n) = \Delta\theta = \theta_d - \theta_n \tag{8}$$

V. Implementation and Results

The method presented in the previous section was implemented to generate trajectories for three planar objects (Fig.10) using three cylindrical fingers moving in a plane. Each object's contour is sampled to create candidate contact points for the grasp stability test (see Fig.3 corresponding to Fig.10.b). The sampling distance may be chosen to meet the level of precision of the fingers positioning. We chose to have 72 samples on the object Fig.10.b on the object's contour.

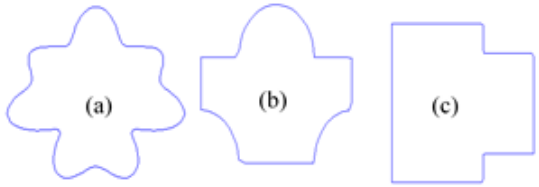

Fig. 10: Shapes of the objects used for in-hand dexterous manipulation in the plane.

The potential contact points on the object's surface are then used to generate the set of stable grasps and create two graphs. The first one uses connections between stable grasps (see [20] for more details) and the second one uses stable rotations as presented in section IV. Finally, an implementation of the A^* algorithm is used to find optimal trajectories in the two graphs.

Both methods produce similar manipulation trajectories that that have been experimentally validated [17], [18] (see Fig.11). First, the object is gripped with two fingers. Note that the initial grasp is not intuitive since both fingers are not opposite each other regarding the manipulated object. What improves the stability of this initial grasp is the adhesion force between the fingers and the object which allows applying a sufficient force to overcome the pull-off force necessary to detach the object from the substrate.

After picking up the object with a pure translation, a third finger is then attached to be able to perform a large rotation of the object. This means that the algorithm moved from

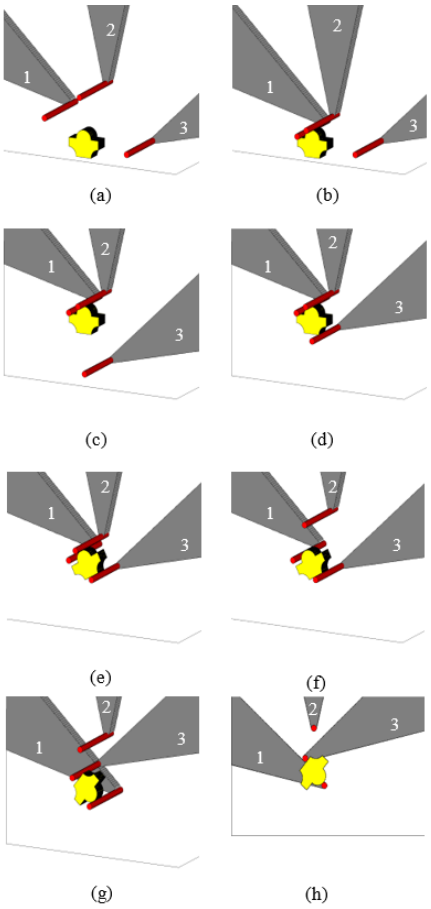

Fig. 11: Sequence of operations for a rotation of 229.18° of the object 10.b. (a) to (c): Initial grasp of the object; (d) and (f): Reconfiguration of the robotic hand; (e) and (g): Rotation without reconfiguration; (h) Front view after rotation of the object.

one node to another in the graph and a non-null cost was associated to this step. The rotation is continued until a risk of collision occurs between finger one and finger two. This configuration also corresponds to the limit of the current node. Since the grasp ensured by fingers one and three can resist the detachment of finger two, the latter is removed which allows continuing the rotation until the desired one.

One can notice that the whole manipulation process required only few nodes. Indeed, only three nodes were necessary to perform the whole rotation (see Fig.11) namely: the initial grasp (b and c), the three fingers rotation (d and e), and the two fingers rotation (f and g). This represents the main advantage of this in-hand manipulation planning method compared to existing ones that use planning between stable grasps instead of stable rotations [2], [13], [17].

Fig.12 shows the calculation time needed to generate the fingers trajectories of the robotic hand as a function of the desired orientation for both the previous method [10] and the current one. It can be noticed from the figure that the current method performs much better than the previous one in all cases (different objects and rotation amplitudes). For instance, the gain in time for 257° rotations varies between 1.7×10^2 for object (a) and 3.7×10^5 for the object (c) (see Fig.10).

This performance can be explained by the complexity of the algorithm. Indeed, the time complexity of the A* algorithm given by $O(b^d)$ where d is the depth of the solution (the number of nodes to reach the solution) and b is the branching factor which represents the average number of branches per node. In the presented approach, both parameters are reduced. For instance, even if the contour of the object represented in Fig.10.b contains only 72 samples, the number of nodes in the new graph (8.6×10^5 nodes) is divided by more than 10 compared to the previous one (9.6×10^6 nodes), moreover Table. I provides the details of nodes for different objects of Fig. 10

Another significant difference is that increasing the number of samples on the object exponentially increases the number of nodes on the first method but does not affect the second one since stable rotations do not change. Instead, this number depends on the object complexity. Indeed, if the shape of the manipulated object is highly non-regular, the amplitude of the stables grasps is lower which increases the number of nodes.

Finally, notice that the generated optimal trajectories are similar to the ones produces by the previous method, which have already been tested and validated experimentally [17].

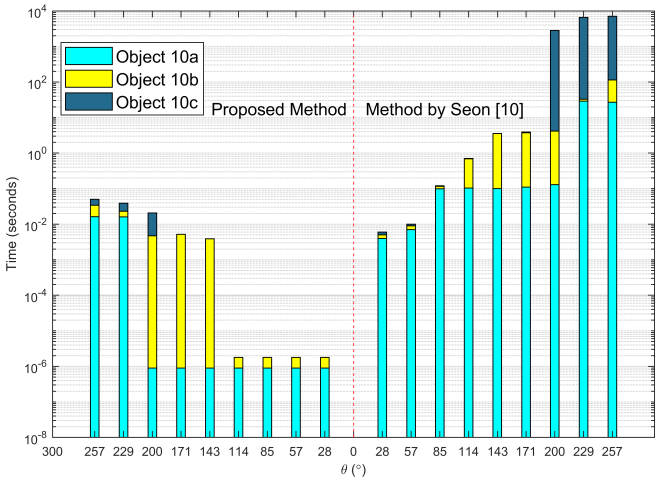

Fig. 12: Comparison between the new Stable Rotations approach (left side) and the previous Stable Grasps (right side) approach in terms of processing time required to generate the in-hand manipulation trajectories for each object represented in Fig.10 as a function of the rotation amplitude. Each bar represents the sum of the computation time required to generate trajectories for each object (segments in the bars). Note that the time axis is logarithmic which means that the computation times are compared in orders of magnitudes.

TABLE I: Comparison of Nodes Generated

	Object 10a	Object 10b	Object 10c
no. of samples	148	72	172
nodes gen. by prop. method	3647016	866681	16120204
nodes gen. by Seon [10]	14588064	9629798	48849103

VI. Conclusion

In this paper, a new in-hand manipulation planning method of planar objects is proposed. The method uses rolling without sliding and finger gaiting manipulation strategies to reach the desired configuration. The planning algorithm also takes into account the adhesion forces that may exist when manipulating micro-objects. The originality of this method is that it uses stable rotations instead of the classical stable grasps. This is achieved by integrating the rolling without sliding kinematic constraint during the rotation into the configuration space representation. The latter encapsulates more information related to the manipulated object, the manipulating system, and the interaction of both into a more compact space. Consequently, the generated graph used to generate the trajectories contains fewer nodes (stable rotations) and connections (actions) compared to the traditional representation. While producing complete and optimal trajectories using the A^* algorithm, the execution time is reduced by a factor variating from 100 to more than 10^5 times. In addition, all the trajectories are generated in less than 0.1 s, even large rotations. Finally, even if the method has been developed for miniaturized objects manipulation, this method is applicable to dexterous manipulation at the macro-scale.

The performance obtained with this method will allow implementing real time trajectory generation integrated in a control loop of the manipulating system. In addition, optimizing the planar manipulation algorithm is a pre-requisite for the extension of this method to the manipulation of arbitrary shaped 3D object in the 3D space.

Acknowledgment

This work was supported by ACTION, the French ANR Labex no. "ANR-11-LABX-01-01" (`http://www.labex-action.fr`) and by the Conseil Régional de Bourgogne Franche-Comté.

References

[1] D. O. Popa, R. Murthy, and A. N. Das, M3-Deterministic, multiscale, multirobot Platform for microsystems packaging: Design and quasi-static precision evaluation, IEEE Trans. Autom. Sci. Eng., vol. 6, no. 2, pp. 345361, Apr. 2009.

[2] A. Sahbani, S. El-Khoury, and P. Bidaud, An overview of 3D object grasp synthesis algorithms, Rob. Auton. Syst., vol. 60, no. 3, pp. 326336, Mar. 2012.

[3] A. Albut, Q. Zhou, C. del Corral and H. N. Koivo,"Development of Flexible Force-Controlled Piezo-Bimorph Microgripping System," Proceedings of 2nd VDE World Microtechnologies Congress, MICRO.tec, pp. 507-512, 2003.

[4] E. Shimada, J. A. Thompson, J. Yan, R. J. Wood and R. S. Fearing, "Prototyping Millirobots using Dextrous Microassembly and Folding," Symposium on Microrobotics, ASME int.Mechanical Engineering Cong. and Exp., vol. 69, no. 2, pp. 933-940, 2000.

[5] J. Thompson and R. S. Fearing, "Automating Microassembly with Ortho-tweezers and Force Sensing," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, vol. 3, p. 13271334, 2001.

[6] J. Wason, J. T. Wen, J. Gorman and N. G. Dagalakis, "Automated Multiprobe Microassembly Using Vision Feedback," IEEE Transactions on Robotics, vol. 28, no. 5, p. 10901103, 2012.

[7] Q. Zhou, P. Korhonen, J. Laitinen and S. Sjovall, "Automatic dextrous microhandling based on a 6 DOF microgripper," Journal of Micromechatronics, vol. 3, no. 3, p. 359387, 2006.

[8] B. Brazey, R. Dahmouche, J. A. Seon and M. Gauthier, "Experimental validation of in-hand planar orientation and translation in microscale," Intelligent Service Robotics, Special Issue on Multi-scale Manipulation Toward Robotic Manifacturing Technologies, vol. 9, no. 2, pp. 101-112, 2016.

[9] A. Bicchi and R. Sorrentino, "Dexterous Manipulation Through Rolling," Int. Conf. on Robotics and Automation, vol. 1, p. 452457, 1995.

[10] J. A. Seon, R. Dahmouche, B. Brazey and M. Gauthier, "Finger Trajectory Generation for Planar Dexterous Micro-Manipulation," IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 392-398, 2016.

[11] J. Dejeu, M. Bechelany, P. Rougeot, L. Philippe and M. Gauthier, "Adhesion control for micro-and nanomanipulation," ACS nano, vol. 5, no. 6, pp. 4648-4657, 2011.

[12] D. J. Cappelleri, J. Fink, B. Mukundakrishnan, V. Kumar, and J. C. Trinkle, Designing open-loop plans for planar micro-manipulation, in Proceedings - IEEE International Conference on Robotics and Automation, 2006, vol. 2006, pp. 637642.

[13] T. Ju, S. Liu, J. Yang, and D. Sun, Apply RRT-based path planning to robotic manipulation of biological cells with optical tweezer, 2011 IEEE Int. Conf. Mechatronics Autom. ICMA 2011, pp. 221226, 2011.

[14] A. G. Banerjee, A. Pomerance, W. Losert, and S. K. Gupta, Developing a stochastic dynamic programming framework for optical tweezer-based automated particle transport operations, IEEE Trans. Autom. Sci. Eng., vol. 7, no. 2, pp. 218227, 2010.

[15] J. P. Saut and D. Sidobre, Efficient models for grasp planning with a multi-fingered hand, Rob. Auton. Syst., vol. 60, no. 3, pp. 347357, 2012.

[16] J.-A. Seon, R. Dahmouche, B. Brazey, and M. Gauthier, Trajectory Planning for Dexterous in-Hand Micromanipulation in Presence of Adhesion Forces, in International Conference on Intelligent Robots and Systems, 2016.

[17] J.-A. Seon, R. Dahmouche, and M. Gauthier, Enhance in - Hand Dexterous Micro - Manipulation by Exploiting Adhesion Forces, IEEE Trans. Robot., vol. 34, no. 1, pp. 113125, 2017.

[18] J.-A. Seon, R. Dahmouche, and M. Gauthier, On the contribution of adhesion and friction in planning dexterous in-hand micromanipulation, J. Micro-Bio Robot., vol. 12, no. 14, pp. 3344, Jun. 2017.

[19] R. R. Ma and A. M. Dollar, On dexterity and dexterous manipulation, in 15th International Conference on Advanced Robotics (ICAR), 2011, pp. 17.

[20] J. Ponce and B. Faverjon, "On computing Three-Finger Force-Closure Grasps of Polygonal Objects," IEEE Trans. on Robotics and Atutomation, vol. 11, no. 6, pp. 868-881, 1995.

[21] "Khanacademy/ Representing graphs," 2006. [Online]. Available: https://www.khanacademy.org/computing/computer-science/algorithms/graph-representation/a/representing-graphs.

[22] D. Maquin, "Elments de Thorie des Graphes," Institut National Polytechnique de Lorraine - Ecole Nationale Suprieure d'Electricit et de Mcanique, Lorraine, 2003.

[23] Y. Wu, D. Sun and W. Huang, "Force and motion analysis for automated cell transportation with optical tweezers," 9th World Congress on Intelligent Control and Autmation (WCICA). IEEE, pp. 839-843, 2011.

[24] D. J. Cappeleri, M. Fatovic and U. Shah, "Caging micromanipulation for automated microassembly," Int. Conf. on Robotics and Automation (ICRA). IEEE, pp. 3145- 3150, 2011.