Joint space and workspace analysis of a two-DOF closed-chain manipulator

Damien Chablat
Institut de Recherche en Communications et Cybernétique de Nantes,
UMR CBRS 6597, 1 rue de la Noë, Nantes, France
damien.chablat@irccyn.ec-nantes.fr

Abstract: The aim of this paper is to compute of the generalized aspects, i.e. the maximal singularity-free domains in the Cartesian product of the joint space and workspace, for a planar parallel mechanism in using quadtree model and interval analysis based method. The parallel mechanisms can admit several solutions to the inverses and the direct kinematic models. These singular configurations divide the joint space and the workspace in several not connected domains. To compute this domains, the quadtree model can be made by using a discretization of the space. Unfortunately, with this method, some singular configurations cannot be detected as a single point in the joint space. The interval analysis based method allow us to assure that all the singularities are found and to reduce the computing times. This approach is tested on a simple planar parallel mechanism with two degrees of freedom.

1 INTRODUCTION

The kinematic design of parallel mechanism has drawn the interest of several researchers. The workspace is usually considered as a relevant design criterion [1, 2, 3]. Parallel singularities [4] occur in the workspace where the moving platform cannot resist any effort. They are very undesirable and generally eliminated by design. Serial singularities [5] occur if the mechanism admit several solutions to the inverse kinematic model. To cope with the existence of multiple inverse kinematic solutions in *serial* manipulators, the notion of aspects was introduced in [6]. The aspects equal the maximal singularity-free domains in the joint space. For usual industrial serial manipulators, the aspects were found to be the maximal sets in the joint space where there is only one inverse kinematic solution.

A definition of the notion of aspect was given by [4] for parallel manipulators with only one inverse kinematic solution and was extended by [5] for parallel manipulators with several solutions to the inverse and direct kinematic problem. These aspects were defined as the maximal singularity-free domains in the Cartesian product of the joint space and the workspace. To compute the aspects, we can used discretization methods. However, we cannot find any singularity in particular if the singularity is a point. Interval based analysis method was implemented by [7, 8] in ALIAS to compute the workspace of parallel mechanism. However, the result is a set of boxes in which is not easy to separate the maximum singularity free regions of the workspace and the computational times is difficult to estimate.

This article introduces an algorithm based on the tree-like structure and the interval analysis based method which takes the advantages of the interval analysis based method and the simplicity of tree-like structures.

2 ALGORITHM

The aim of this section is to define an algorithm able to compute either the joint space or the workspace of parallel mechanism. This is done using the structure of a quadtree model and the interval based method. This algorithm will be illustrated by a planar parallel mechanism in section 3 but can easily be extended to mechanism with three degrees of freedom in using octree model. Unlike numerical computing methods, such a method allows to prove formally that there is no singular configuration in the final result.

2.1 Definition of the quadtree/octree model

The tree-like structure, called in this paper, quadtree or octree, are a hierarchical data structure based on a recursive subdivision of space [9]. It is particularly useful for representing complex $2-D$ or $3-D$ shapes, and is suitable for Boolean operations like union, difference and intersection. Since the quadtree/octree structure has an implicit adjacency graph, arcwise-connectivity analysis can be naturally achieved. The quadtree/octree model of a space $\mathcal{S}$ leads to a representation with cubes of various sizes. Basically, the smallest cubes lie near the boundary of the shape and their size determines the accuracy of the quadtree/octree representation. Quadtree/octrees have been used in several robotic applications [4], [10], [11].

The main advantages of the octree model are (i) very compact file (only B (Black), W (White) and G (Gray) letter) and (ii) accelerates display speed facilities by differentiating model into lower levels of resolution while being rotated and higher resolution while an orientation is temporarily set. Conversely, the computational times can be high due to the discretization. If we test only the center of each cube, some singularities can exist and are not detected. This feature is true even if we increase the level of resolution. The accuracy of the model is directly defined by the depth of the quadtree and the size of the initial box. For a quadtree model with a depth d and a initial box of lengths b, the accuracy is $b/2^d$.

2.2 Notion of aspect for fully parallel manipulators

We recall here briefly the definition of the generalized octree defined in [5]:

Definition 1. The generalized aspects $\mathbf{A}_{ij}$ are defined as the maximal sets in $W \cdot Q$ so that $\mathbf{A}_{ij} \subset W \cdot Q$, $\mathbf{A}_{ij}$ is connected, and $\mathbf{A}_{ij} = \{(\mathbf{X}, \mathbf{q}) \in Mf_i \setminus det(\mathbf{A}) \neq 0\}$

In other words, the generalized aspects $\mathbf{A}_{ij}$ are the maximal singularity-free domains of the Cartesian product of the reachable workspace with the reachable joint space.

Definition 2. The projection of the generalized aspects in the workspace yields the parallel aspects $\mathbf{WA}_{ij}$ so that $\mathbf{WA}_{ij} \subset W$, and $\mathbf{WA}_{ij}$ is connected.

The parallel aspects are the maximal singularity-free domains in the workspace for one given working mode.

Definition 3. The projection of the generalized aspects in the joint space yields the serial aspects $\mathbf{QA}_{ij}$ so that $\mathbf{QA}_{ij} \subset Q$, and $\mathbf{QA}_{ij}$ is connected.

The serial aspects are the maximal singularity-free domains in the joint space for one given working mode.

The aim of this paper is to compute separately the parallel and serial aspects thanks to the properties of the quadtree models.

2.3 Introduction to ALIAS library

An algorithm for the definition of the joint space and workspace of parallel mechanism is described in the following sections. This algorithm uses the `ALIAS` library [7], which is a C++ library of algorithms based on interval analysis. These algorithms deal with systems of equations and inequalities of which expressions are an arbitrary combination of the most classical mathematical functions (algebraic terms, sine, cosine, log etc..) and of which coefficients are real numbers or, in some cases, intervals. Unfortunately, this library is not connected to the octree model and generates large data file to describe by a set of boxes the solution of the problem. Thus, the operations between the set of boxes is more difficult if we compare to the boolean operations that we can made with the octree models.

2.4 A first basic tool: Box verification

Our purpose is to determine the quadtree model associated with the joint space or the workspace, that we will call only the space $\mathcal{C}$. For a given box B defined by two intervals, we note *valid box* if it is included in $\mathcal{C}$ and *invalid box* otherwise. For that purpose we need to design first a procedure, called $\mathcal{M}(B)$, that takes as input a box B and returns:

- 1: if every point in B is valid,
- -1: if no point in B is valid,
- 0: if neither of the other two conditions could be verified.

To check if one point is valid, we can used several approaches. For example, in the Cartesian space, we have to compute the inverse kinematic model to test if at least one solution exists, which defines the workspace. Thus, for each solution, we can define completely the mechanism for each working mode and compute the determinant of the parallel Jacobian matrix [5]. This procedure is able to define the non-singular domains but their are not necessary connected. However, with a quadtree model, it is easy to perform a connectivity analysis to separate the quadtree model in connected domains. This is a main advantage of the tree-like structures in comparison with the method implemented in the `ALIAS` library. Some additional constraints can be to define dexterous domains in using a kinetostatic index as in [8].

The problem is now to implement an inverse and direct kinematic model able to take as input a box B and to return, if the box is valid, a box S which contains the solutions of the problem.

2.5 A second basic tool: Quadtree model definition

The definition of a quadtree is made recursively by calling several times the same procedure, call $\mathcal{Q}(B, d, P)$, that takes as input a box B, the local depth d in the tree and a pointer P on the data structure which contains the quadtree model.

- If $\mathcal{M}(B)$ returns -1, it is the end of the recursive search.
- If $\mathcal{M}(B)$ returns 1, a black node is created in the current depth.
- If $\mathcal{M}(B)$ returns 0, if the local depth is smallest than the maximal depth of the quadtree, we divide the box B into four new boxes B_1, B_2, B_3 and B_4, and we call

$$\mathcal{Q}(B_1, d+1, P) \quad \mathcal{Q}(B_2, d+1, P) \quad \mathcal{Q}(B_3, d+1, P) \quad \mathcal{Q}(B_4, d+1, P)$$

When $\mathcal{M}(B)$ returns -1, we can also create a another quadtree model, called *complementary space* in which we save the box B where at least one constraint is not valid. With this knowledge, we can compute the quadtree model with a higher definition without retesting the valid and non valid boxes. An example of this result will be given in the following section.

3 MECHANISM UNDER STUDY

For more legibility, a planar manipulator is used as illustrative example in this paper. This is a five-bar, revolute (R)-closed-loop linkage, as displayed in figure 1. The actuated joint variables are θ_1 and θ_2, while the Output values are the (x, y) coordinates of the revolute center P. The displacement of the passive joints will always be assumed unlimited in this study. Lengths L_0, L_1, L_2, L_3, and L_4 define the geometry of this manipulator entirely.

Two sets of dimensions are used to illustrate the algorithm. The first one, called $\mathcal{M}_1$, is defined in [5] and its dimensions are $L_0 = 9$, $L_1 = 8$, $L_2 = 5$, $L_3 = 5$ and $L_4 = 8$, in certain units of length that we need not to specify. And, the second one, called $\mathcal{M}_2$, is defined in [12] and its dimensions are $L_0 = 2.55$, $L_1 = 2.3$, $L_2 = 2.3$, $L_3 = 2.3$ and $L_4 = 2.3$.

3.1 Kinematic Relations

The velocity $\dot{\mathbf{p}}$ of point P, of position vector $\mathbf{p}$, can be obtained in two different forms, depending on the direction in which the loop is traversed, namely,

$$\dot{\mathbf{p}} = \dot{\mathbf{b_1}} + \dot{\theta}_3 \mathbf{E}(\mathbf{p} - \mathbf{b_1}) \quad \dot{\mathbf{p}} = \dot{\mathbf{b_2}} + \dot{\theta}_4 \mathbf{E}(\mathbf{p} - \mathbf{b_2}) \tag{1}$$

with matrix $\mathbf{E}$ defined as $\mathbf{E} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ and $\mathbf{b}_1$ and $\mathbf{b}_2$ denoting the position vectors, in the frame indicated in figure 1 of points B_1 and B_2, respectively.

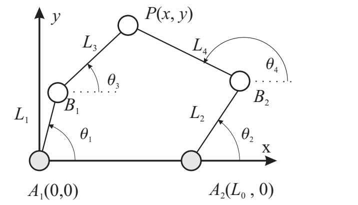

Figure 1: A two-dof closed-chain manipulator

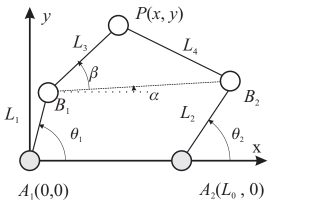

Figure 2: The angle α and β used to solve the DKP

Furthermore, note that $\dot{\mathbf{c}}$ and $\dot{\mathbf{d}}$ are given by

$$\dot{\mathbf{b_1}} = \dot{\theta}_1 \mathbf{E}(\mathbf{b}_1 - \mathbf{a}_2), \quad \dot{\mathbf{b_2}} = \dot{\theta}_2 \mathbf{E}(\mathbf{b}_2 - \mathbf{a}_2)$$

Two Jacobian matrix, $\mathbf{A}$ and $\mathbf{B}$, permit to study the serial and parallel singular configurations [4],

$$\mathbf{A} = \begin{bmatrix} (\mathbf{p} - \mathbf{b_1})^T \\ (\mathbf{p} - \mathbf{b_2})^T \end{bmatrix}, \ \mathbf{B}{=}\begin{bmatrix} L_1 L_2 \sin(\theta_3 - \theta_1) & 0 \\ 0 & L_3 L_4 \sin(\theta_4 - \theta_2) \end{bmatrix}{=}\begin{bmatrix} B_{11} & 0 \\ 0 & B_{22} \end{bmatrix} \quad (2)$$

Two assembly modes can be defined with the sign of $\det(\mathbf{A})$. To characterize the assembly mode, we can only compute $(\mathbf{p} - \mathbf{b_1}) \times (\mathbf{p} - \mathbf{b_2})$.

The parallel singular configurations are located at the boundary of the joint space. Such singularities occur whenever B_1, P and B_2 are aligned. In such configurations, the manipulator cannot resist an effort in the orthogonal direction of $B_1 B_2$. Besides, when B_1 and B_2 coincide, the position of P is no longer controllable since P can rotate freely around B_1 even if the actuated joints are locked. This singularity cannot be find by the discretization of the workspace but is detected by the interval analysis based method.

3.2 Direct kinematic problem

For planar mechanism, the computation of the direct kinematic problem (DKP) is very simple. However, if we want to return the good information to the function $\mathcal{C}$, we have to distinguish more cases. If there is no joint limits, they are zero or two solutions for the DKP.

The procedure takes as input a box B defined by two intervals $\tilde{\theta_1} = [\underline{\theta_1}, \overline{\theta_1}]$ and $\tilde{\theta_2} = [\underline{\theta_2}, \overline{\theta_2}]$, the actuated joint variables. In this example, the values of the length are defined as a float but can be also defined by a interval to represent the tolerances of manufacturing.

The algorithm can be described by the following steps:

- Compute the position of $\tilde{\mathbf{b_1}}$ and $\tilde{\mathbf{b_2}}$ which is the location of B_1 and B_2 respectively.
- Compute the distance $\tilde{L}$ between B_1 and B_2:
  - $[\underline{L}, \overline{L}] = ||\tilde{\mathbf{b_1}} - \tilde{\mathbf{b_2}}||$

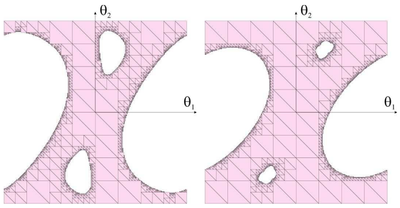

Figure 3: Joint space of $\mathcal{M}_1$ and $\mathcal{M}_2$

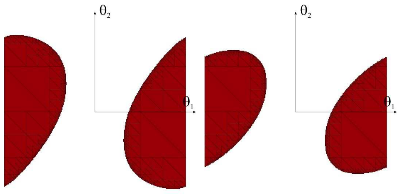

Figure 4: The complementary joint space of $\mathcal{M}_1$ and $\mathcal{M}_2$

  - If $\underline{L} > L_3 + L_4$ then the mechanism can be assembly and the function return -1.
  - If $\underline{L} = 0$ then the mechanism can be in a singular configuration.
- Compute the angle α and β of the triangle (B_1, B_2, P) (Figure 2).
  - $C(\alpha) = \sqrt{(\tilde{L}^2 + L_3^2 - L_4^2)/(2 L_3 \tilde{L})}$.
  - $\beta = \arctan\left(\frac{B_{2y} - B_{1y}}{B_{2x} - B_{1x}})\right)$
  - If $(\underline{C(\alpha)} \geq -1$ and $C(\alpha) \leq 1)$ then $\alpha = \arccos(C(\alpha))$.
- Compute the two solutions P_1 and P_2 of point P:
  - $P_{1x} = B_{1x} + L_3 \cos(\alpha + \beta)$
  - $P_{1y} = B_{1y} + L_3 \sin(\alpha + \beta)$
  - $P_{2x} = B_{1x} + L_3 \cos(\alpha - \beta)$
  - $P_{2y} = B_{1y} + L_3 \sin(\alpha - \beta)$.

To perform this procedure, all the trigonometric function come from the `ALIAS` library and accept as input an interval and return an interval. For each step, we have to check that the argument are not out of range.

For the two mechanism under study, we can plot with this function the joint space (Figure 3). The results are equivalent to the solution computed in [5] and [12]. However, conversely to a discretization method, we can detect the parallel singularity where B_1 and B_2 coincide. Normally, it is a point but with the interval analysis method, the algorithm cannot define valid boxec where $\det(\mathbf{A})$ is equal to zero. The set of box where at least one constraint is not valid is called the complementary joint space, figure 4.

3.3 Selection of the assembly mode

For a parallel mechanism with only two solutions to the direct kinematic problem, the assembly mode is characterize by the sign of $\det(\mathbf{A})$. To simplify, we can only compute

$$\tilde{\mathbf{t}} = (\tilde{\mathbf{b}_1} - \tilde{\mathbf{p}}) \times (\tilde{\mathbf{b}_2} - \tilde{\mathbf{p}}) \quad (3)$$

and to test if $\tilde{\mathbf{t}_z}$ is positive or negative.

3.4 Inverse kinematic problem

For the mechanism under study, there are four solutions for the inverse kinematic problem (IKP), two solutions for each leg. The procedure takes as input a box B defined by to two intervals $\tilde{x} = [\underline{x}, \overline{y}]$ and $\tilde{y} = [\underline{y}, \overline{y}]$, the position of $\tilde{P}$.

The algorithm can be described by the following steps:

- Compute the distance $\tilde{M}_1$ between A_1 and P, and $\tilde{M}_2$ between A_2 and $\tilde{P}$.
- If ($\underline{M_1} > L_1 + L_3$ or $\underline{M_2} > L_2 + L_4$) then P is outside the workspace and the function returns -1.
- If ($\overline{M_1} < ||L_1 - L_3||$ or $\overline{M_2} < ||L_2 - L_4||$) then P is located in a hole of the workspace and the function returns -1.
- If ($\underline{M_1} > 0$) and ($\underline{M_2} > 0$) and ($\underline{M_1} > ||L_1 - L3||$) and ($\underline{M_2} > ||L_2 - L4||$) and ($\overline{M_1} < L_1 + L_3$) and ($\overline{M_2} < L_2 + L_4$) then compute the angles $\beta_1 = \widehat{B_1 A_1 P}$ and $\beta_2 = \widehat{B_2 A_2 P}$ noted in Figure 5:
  - $\tilde{C}(\beta_1) = \frac{L_1^2 + \tilde{M}_1^2 - L_3^2}{2\tilde{M}_1 L_1}$ $\quad \tilde{C}(\beta_2) = \frac{L_2^2 + \tilde{M}_2^2 - L_4^2}{2\tilde{M}_2 L_2}$
  - If ($||\tilde{C}(\beta_1)|| \leq 1$) and ($||\tilde{C}(\beta_2)|| \leq 1$) then
    - $\alpha_1 = \arctan(P_y/P_x)$ $\quad \alpha_2 = \arctan(P_y/(C_2 - P_x))$
    - $\beta_1 = \arccos(\tilde{C}(\beta_1))$ $\quad \beta_2 = \arccos(\tilde{C}(\beta_2))$
    - $\alpha_{11} = \alpha_1 + \beta_1$ $\quad \alpha_{12} = \alpha_1 - \beta_1$ $\quad \alpha_{21} = \pi - \alpha_2 + \beta_2$ $\quad \alpha_{22} = \pi - \alpha_2 - \beta_2$
    - Returns 1;
  - Else, returns 0;

For the two mechanism under study, we can plot with this function the joint space (Figure 6). For the mechanism $\mathcal{M}_\in$, there is a hole in A_1 and A_2 which is only a point. On the picture, we can only notice that there exists a sub-division of the space because there are small boxes. The size of the hole is equal to the size of the smallest box.

3.5 Selection of the working mode

For a parallel mechanism with only four solutions to the inverse kinematic problem, the working mode is characterized by the sign of B_{11} and B_{22}. This can be done simply

$$\tilde{\mathbf{u}} = (\tilde{\mathbf{b_1}} - \tilde{\mathbf{a_1}}) \times (\tilde{\mathbf{p}} - \tilde{\mathbf{b_1}}) \quad \tilde{\mathbf{v}} = (\tilde{\mathbf{b_2}} - \tilde{\mathbf{a_2}}) \times (\tilde{\mathbf{p}} - \tilde{\mathbf{b_2}}) \tag{4}$$

and to test the sign of $\tilde{\mathbf{u_z}}$ or $\tilde{\mathbf{v}_z}$.

3.6 Computation of the generalized aspects

The number of aspects is the same for the both mechanisms. We can compute separately the serial and parallel aspects for all the working modes. For a give sign of $\det(\mathbf{A})$, B_{11} and B_{22}, we have define a quadtree model. To obtain the generalized aspects, a connectivity analysis is made to separate the connected regions. We need to test only one point in the workspace and its projection on the joint space to associate a serial aspect to its parallel aspect counterpart.

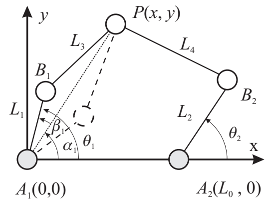

Figure 5: The angle α and β used to solve the IKP

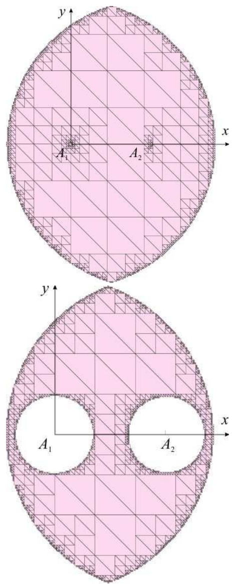

Figure 6: Workspace of $\mathcal{M}_1$ and $\mathcal{M}_2$

Table 1: Comparison between classical quadtree computation and the new algorithm to compute the joint space and the workspace

Joint space	Depth #	5	6	7	8	9	10
	$\mathcal{M}_1$	99%	73%	40%	22%	13 %	9 %
	$\mathcal{M}_2$	111%	83%	40%	18%	8 %	4 %
Workspace	Depth #	5	6	7	8	9	10
	$\mathcal{M}_1$	72%	45%	25%	13%	7 %	3 %
	$\mathcal{M}_2$	65%	37%	19%	10%	5 %	2 %

The study of the joint space allow us to know if a trajectory between two generalized aspects exists by passing through a serial singularity. Figures 9 and 10 permit us to conclude that, for $\mathcal{M}_1$ and $\mathcal{M}_2$, no trajectory exists in which we change only one time the working modes, between the aspects 1 and 4 and between the aspects 2, 3 and 5.

3.7 Comparison between classical quadtree computation and the new algorithm

The aim of this section is to compare the number of times where the inverse or direct kinematic function is called to build the quadtree for the both mechanisms and for several depths of the tree which is equivalent to the accuracy of the model. For the joint space, the initial box B is defined by two intervals equal to $[-\pi, \pi]$ and for the workspace, the initial box B is defined by two intervals equal to $[-(L_1 + L_3), (L_1 + L_3)]$.

With the discretization method, the number of times where the inverse or direct kinematic problem is used is $n_{discretization} = 2^{2d}$ with d is the depth of the tree. We call $n_{quadtree}$ the number of times where where the inverse or direct kinematic problem is used to build the quadtree model. To compare the computing cost between the both method, we define the following criteria:

$$\mathcal{K} = n_{quadtree}/n_{discretization} \tag{5}$$

Table 1 compares the computing times for the joint space and the workspace for the mechanisms $\mathcal{M}_1$ and $\mathcal{M}_2$. When the depth of the quadtree is small, there is no advantage to used the interval analysis based method. But, when the depth increase, the advantages can be very important. For example, to compute the workspace of $\mathcal{M}_1$ with $d = 10$, we call 36893 times the IKP while the discretization method calls $2^{20} = 1048576$ times the IKP.

4 CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented an algorithm able to compute the joint space and workspace of parallel mechanism by using the octree model and the interval based method. We have the maximum singularity regions, called *aspects*, for two planar mechanisms and we have compared the number of times where the inverse and direct kinematic problem is called according to the accuracy used. Thanks

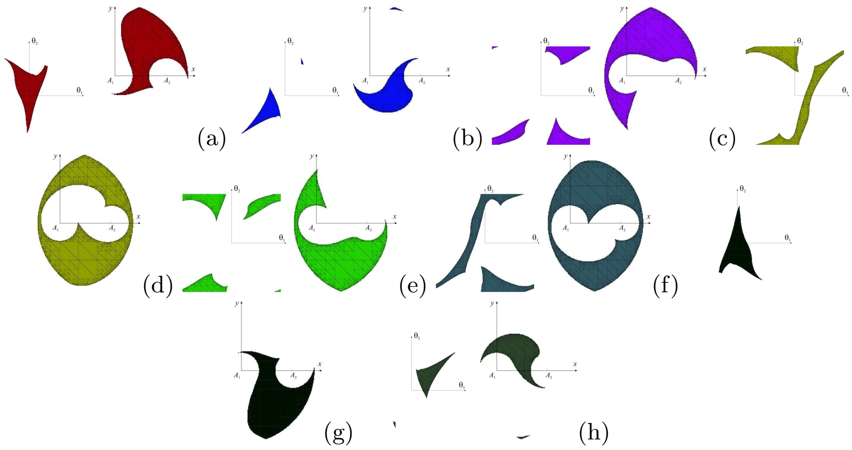

Figure 7: Aspects of $\mathcal{M}_1$

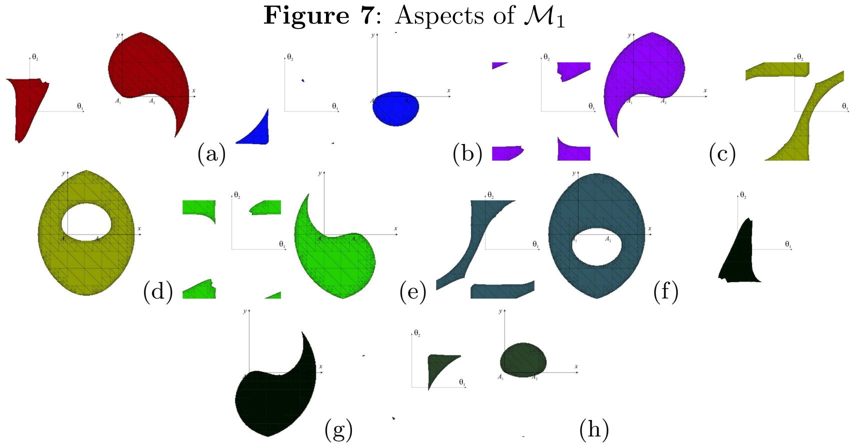

Figure 8: Aspects of $\mathcal{M}_2$

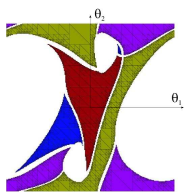

Figure 9: The serial aspects of $\mathcal{M}_1$ for $\det(\mathbf{A}) > 0$

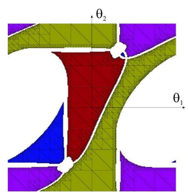

Figure 10: The serial aspects of $\mathcal{M}_2$ for $\det(\mathbf{A}) > 0$

to the use of the interval analysis based method, the result is guaranteed for all the depth used, i.e. we are sure to detect all the singular configurations. The quadtree model can be saved in text file and its size is very small. Plugin exist now to visualize the model in a 3D viewer in Web pages (as www.octree.com). Future works will be made to define as interval the lengths of the legs and to find the influence of the manufacturing error on the joint space and workspace.

5 ACKNOWLEDGMENTS

This work was supported partly by the French Research Agency A.N.R. (Agence Nationale pour la Recherche).

REFERENCES

[1] Merlet, J-P., “Workspace-0riented Methodology for Designing a Parallel Manipulator,” Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3726–3731, 1996.

[2] Clavel, R., “DELTA, a Fast Robot with Parallel Geometry,” Proc. 18th Int. Symposium of Industrial Robots, pp. 91–100, 1988.

[3] Gosselin, C. and Angeles, J., “A Global Performance Index for the Kinematic Optimization of Robotic Manipulators,” Journal of Mechanical Design, vol. 113, pp. 220–226, 1991.

[4] Wenger Ph., and Chablat D., “Definition Sets for the Direct Kinematics of Parallel Manipulators,” Proc.8th International Conference in Advanced Robotics, pp. 859–864, 1997.

[5] Chablat D., and Wenger Ph., “Working Modes and Aspects in Fully-Parallel Manipulator,” Proc. IEEE International Conference on Robotics and Automation, pp. 1964–1969, May 1998.

[6] Borrel, P., “A study of manipulator inverse kinematic solutions with application to trajectory planning and workspace determination,” Proc. IEEE International Conference on Robotic And Automation, pp. 1180–1185, 1986.

[7] Merlet, J-P., “ALIAS: an interval analysis based library for solving and analyzing system of equations,” Pro. Séminaire Systèmes et équations algébriques, SEA, June, pp. 1964–1969, 2000.

[8] Chablat D., Wenger P., Majou F., and Merlet J.P., “An Interval Analysis Based Study for the Design and the Comparison of 3-DOF Parallel Kinematic Machine”, International Journal of Robotics Research, pp. 615–624, vol. 23(6), June 2004.

[9] Meagher, D., “Geometric Modelling using Octree Encoding”, Technical Report IPL-TR-81-005, Image Processing Laboratory, Rensselaer Polytechnic Institute, Troy, 1981, New York 12181.

[10] Faverjon B., “Obstacle avoidance using an octree in the configuration space of a manipulator”, Proc. IEEE International Conference on Robotics And Autaumation, pp. 504–510, 1984.

[11] Garcia G., Wenger P., Chedmail P., “Computing moveability areas of a robot among obstacles using octrees”, Proc. International Conference on Advanced Robotics, Columbus, Ohio, USA, June 1989.

[12] Figielski, A., Bonev, I.A., and Bigras, P., "Towards Development of a 2-DOF Planar Parallel Robot with Optimal Workspace Use", IEEE International Conference on Systems, Man, and Cybernetics (SMC 2007), Montréal, QC, Canada, 7–10 October 2007.