A new compliant mechanism design methodology based on flexible building blocks

Paul Bernardoni[a] Philippe Bidaud[b] Catherine Bidard[a] Florian Gosselin[a]

[a]Laboratoire d'Intégration des Systèmes et des Technologies - CEA/DRT/LIST
[b]Laboratoire de Robotique de Paris - Université Paris 6 - CNRS
18, Route du Panorama - 92265 Fontenay Aux Roses - France

ABSTRACT

The paper introduces a new design method for the synthesis of compliant mechanisms. This method is based on the optimization of the distribution of compliant building blocks within a given design domain. Building blocks are modeled by elementary frame ground structures. The topology, dimensions, material, contacts, fixed frame and actuators of the optimal compliant mechanism are generated automatically using a multi objective genetic algorithm such that the force/motion ratio is maximized. The set of optimal solutions is explored by using the notion of Pareto optimality. An application of this design method is tested on an actuated compliant mechanism with two-output degrees of freedom.

Keywords: design, compliant mechanism, genetic algorithm, building blocks

1. INTRODUCTION

The developments in MEMS (Micro Electronics and Mechanical Systems) for the manipulation in 3D space of micro-objects or smart miniaturized surgical tools motivate the research in compliant mechanisms design method. A compliant mechanism is a single-piece, flexible structure that delivers the desired motion and force by undergoing elastic deformation as opposed to rigid-body mechanisms. Compliant mechanisms eliminate backlash, friction, wear and effectively reduce the production and maintenance cost associated with the multiple piece assembly.[1] This paper focuses on the conceptual design method of compliant mechanisms.

The design methods can be categorized as kinematics-based approach and continuum-based approach.[2] The kinematics-based approach considers a compliant mechanism as an assembly of rigid links and flexible joints[3] such as notch hinges[4] or large displacement compliant joints.[5] This approach makes use of mechanism theories to design rigid links and flexible joints assembly (pseudo rigid-body model[6]). The continuum-based approach provides an optimal topology, shape and size of single piece compliant mechanisms for given deformation requirements. Optimal topology design methods such as homogenization methods,[7] levelset methods[8] and frame ground structure methods[9] refer to the appropriate material distribution in a fixed region. This optimal material distribution is usually obtained with combined finite element analysis (FEA) for objective evaluation and optimization procedures for improvement. These optimization procedures can lead to local optima, and recently genetic algorithms have been used in order to converge to a global optimum.[10]

These two basic approaches can be seen as complementary. Continuum-based methods can be considered as preliminary design methods since they can lead easily and rapidly to innovative solutions for multiple input and output requirements with respect to the boundary conditions (loading and support locations). However, in the design process, human intervention is limited, which can lead to technologically unrealistic structures.

Further author information paul.bernardoni@cea.fr, bidaud@robot.jussieu.fr

On the other hand, the kinematics-based approach is natural for engineers: it takes place within a more global design process and allows a more realistic design taking into account technological constraints. This approach is however limited to two-connection flexible elements[11] and often results in structures with lumped compliance. Moreover this approach is very time-consuming.

Our aim is to develop a new conceptual design method allowing easy and rapid design of compliant structures and taking into account technological constraints. Therefore, we propose a method that combines the advantages of both approaches. This method considers a compliant mechanism as an assembly of compliant building blocks. The use of blocks as elementary functional elements is a natural approach for engineers who can incorporate in their definition their own experience. For example the designer can define blocks with distributed compliance, or blocks with lumped compliance such as notch hinges. The use of blocks reduces the search space and leads to more tractable optimization problems. At present the blocks have four nodes and are defined with frame elements: thus they are not limited to two-connection flexible elements. Finally, a multi objective genetic algorithm is used for global optimization of compliant building blocks assemblies: this algorithm allows discrete variables such as building-blocks variables or boundary conditions variables.

An experimental toolbox, called FlexIn (Flexure Innovation) has been developed in Matlab^{TM}. In section 2, the principle of this conceptual design method is covered. Then, the multicriteria genetic algorithm, which searches solutions automatically, is presented in section 3. Section 4 illustrates the efficiency of the proposed method by studying a simple case with two-output degrees of freedom (dof).

2. FLEXIN: A DESIGN METHOD USING COMPLIANT BUILDING BLOCKS

2.1. Principle of the method

The purpose of FlexIn is to improve the design of realistic compliant structures together with an increase of the designer's knowledge. First, elementary compliant 2D building blocks have been defined. Their definition can take into account manufacturing constraints, as for instance different materials, thickness or realistic shapes with regard to the selected manufacturing process. These blocks constitute the elementary units that will be used for the compliant structure design method.

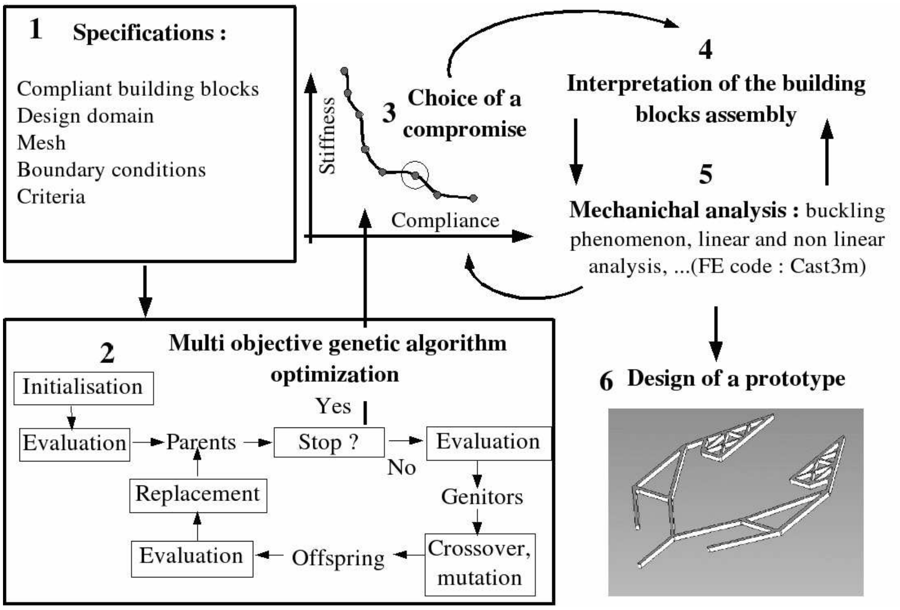

Figure 1. A compliant building blocks design method

The design method consists in searching for an optimal distribution of these building blocks with their associated parameters. A sub-set of blocks used in the optimization process can be selected. Selected blocks can also be 'manually' assembled in the design space with the help of the user interface (see **1** in fig.1). Specification of the output is then defined. Fixed frame location, contacts and loading conditions (or inputs dof defined by actuators) can also be imposed by the designer. Topology optimization algorithms generate a set of candidate solutions (see **2** in fig.1). The optimization solver is based on evolutionary algorithms (see section 3). In order to compute an optimal balance between stiffness and compliance, fitness functions can incorporate displacement and force at the output port, or strain energy[9] (see section 3.3). Then the designer chooses, interprets and analyses the obtained structures that best suit for his problem (see **3,4,5** in fig.1, Cast3m^{TM} is used for the FEA).

Yet, to start with the development of FlexIn, we have focused on bi-dimensional (2D) structures. Moreover, as conventional surface micromachining technology is adapted for thin structures, we have used beams to describe building blocks.

2.2. Compliant building blocks

A library of compliant elements is proposed in FlexIn. These blocks are mainly derived from the observation of existing compliant structures, such as cross flexure hinge, flexible beam, compliant translational joint (see figure 2). Two-dimensional quadrilateral blocks with four nodes linked by flexible beams appear as interesting. As shown in figure 3, different basic beam networks can be obtained with junctions between two nodes of the mesh. Others blocks can be obtained by linking three nodes or four nodes. A stiffness matrix characterizes each of these blocks. As an example, the 'square' block can act as a compliant sliding joint, whereas the blocks 'cross' and 'pivot' act more like flexure hinges. The block 'full', which links the four nodes together, is very stiff. The 'empty' block has a null stiffness matrix.

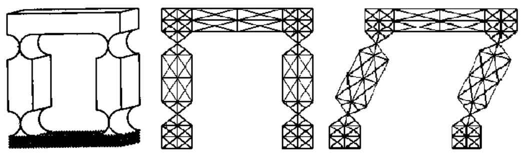

Figure 2. A compliant translational joint and its compliant building block representation

The blocks are composed of beams. Structural parameters of each rectangular blocks are height (a), width (b) and thickness (ep). Material characteristics are parameterized by Young modulus E_y, Poisson ratio ν and density ρ. In order to preserve their stiffness particularities, the ratio between width and height is kept in the range [0.25 4]. The complementary roles of the set of blocks have been studied by simulation in order to validate their relevance. Next subsection deals with the blocks model, the blocks assembly and its boundary conditions.

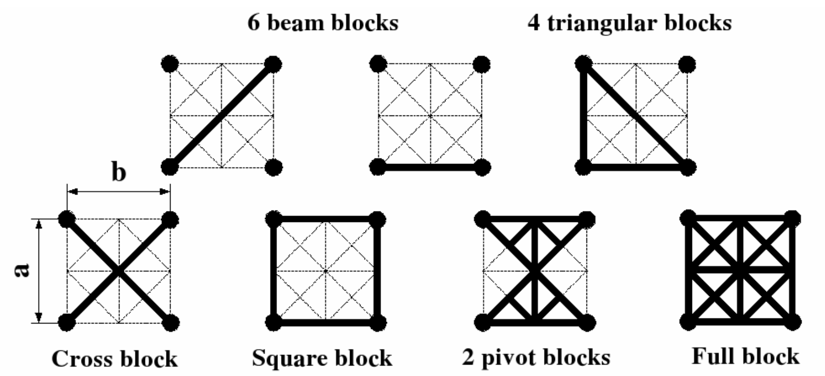

Figure 3. Compliant building blocks adapted to surface micromachining technology

2.3. Mechanical model

In FlexIn development, it is assumed that the two-dimensional planar compliant mechanism is composed of frame-like elements, where the structural deformation comes mainly from bending of the beams. Thus, we make the following assumptions, which are commonly used by continuum-based design methods[12]:
- static state calculation;
- small perturbations;
- homogeneous and linear elastic material;
- Navier-Bernoulli beams with rectangular section.

Under such hypotheses, FEA gives the stiffness matrix of a linear beam element:

$$K_{beam} = \begin{pmatrix} E_yS/L & 0 & 0 & -E_yS/L & 0 & 0 \\ 0 & 12E_yI/L^3 & 6E_yI/L^2 & 0 & -12E_yI/L^3 & 6E_yI/L^2 \\ 0 & 6E_yI/L^2 & 4E_yI/L & 0 & -6E_yI/L^2 & 2E_yI/L \\ -E_yS/L & 0 & 0 & E_yS/L & 0 & 0 \\ 0 & -12E_yI/L^3 & -6E_yI/L^2 & 0 & 12E_yI/L^3 & -6E_yI/L^2 \\ 0 & 6E_yI/L^2 & 2E_yI/L & 0 & -6E_yI/L^2 & 4E_yI/L \end{pmatrix} \tag{1}$$

The assembly of K_{beam} matrices of this type leads to the global stiffness matrix of a block. Let F_o and F_i be respectively the outer nodal forces and the inner nodal forces. Let U_o and U_i be respectively the outer nodal displacements and the inner nodal displacements. Then the global stiffness matrix of a block can be written as:

$$\begin{pmatrix} F_o \\ F_i \end{pmatrix} = \begin{pmatrix} K_{oo} & K_{oi} \\ K_{oi}^t & K_{ii} \end{pmatrix} \begin{pmatrix} U_o \\ U_i \end{pmatrix} \tag{2}$$

The block stiffness matrix can be simplified when assuming that the interaction between the blocks F_o (outer nodal force vector) is restricted to their four external nodes and that no force is applied inside the block (inner nodal force vector $F_i = 0$). Indeed, the model of each block is a meta-element with four nodes having three degrees of freedom each. This meta-element is characterized by a stiffness matrix K_{block}. Thanks to the condensation method (see Eq.3), blocks stiffness matrix size is reduced to only 12×12 instead of 33×33. Moreover, these matrices can be pre-processed.

$$F_o = (K_{oo} - K_{oi}^t \, K_{ii}^{-1} \, K_{oi}) \, U_o \tag{3}$$

$$F_o = K_{block} \, U_o \tag{4}$$

A compliant mechanism is considered as an assembly of flexible building blocks (an illustration is given in fig.2). Thus, the global stiffness matrix K_{global} of the compliant mechanism is an assembly of meta-elements. This assembly is represented by a mesh, which is defined by parameters such as the number and the size (a, b) of meta-elements. This mesh can be seen as the design domain in which the compliant structure has to fit.

The compliant mechanism is also defined by its boundary conditions. These boundary conditions represent:

- Fixed frame location: location and number of fixed points (grounded points) can be specified at nodes. The concerning nodes will have a null displacement, and the concerning dof will be removed.
- Input: several actuators can be defined from a variety of types. These actuators provide a nodal displacement or a nodal force and have respectively force or displacement limitations. Their stiffness and their location (the nodes where the actuators are applied) can also be specified. For example, we can assume that the effect of a piezoelectric actuator is a displacement (for example $1\mu m$), and that its maximum force and its stiffness are very high (for example: maximum force $10N$, and stiffness $10^6 N/m$).

- Contacts[13]: intermittent contacts between different nodes of the elastic structure or between a node of the compliant structure and a rigid surface can be defined. Contact interactions give rise to interesting non-linear and non-smooth behavior. FlexIn allows the designer specifying the nodes that can be in contact, and the backlash between these nodes.
- Output: the output is specified in such a way that it reflects the number of output dof, their locations (nodes), their effects on the environment (such as a force or a stroke), and the environment's stiffness. As an example, a compliant microgripper for micromanipulation of cells is a one-output-dof mechanism. We can assume that the effect of its jaws is a $200\mu m$ stroke. We can assume that the range of the force it must deliver is low ($[0\mu N\ to\ 80\mu N]$), and that the stiffness of the objects grasped is also very low ($0.1N/m$).

Once the global stiffness matrix and the boundary conditions defined, the compliant mechanism can be simulated. The pre-processing of the meta-elements significantly reduces the computational time, and makes the modeling approach well adapted to semi-stochastic optimization of compliant mechanisms.

3. USING A MULTI CRITERIA GENETIC ALGORITHM

An optimization process has been developed to search for optimal topologies, dimensions and boundary conditions of compliant mechanisms. This process is based on genetic algorithms integrating a simulation of the task. In this context, genetic algorithms have demonstrated very interesting potentialities (flexibility, alternate optimal solutions, etc ...). They are semi-stochastic optimization techniques in which candidate solutions are considered as individuals in a population of solutions encoded by chromosomes made of genes. The whole population is then grown with simplified genetic laws and undergoes genetic operators such as selection, mutation and crossover (see **2** in fig.1). The mating pool is generally selected on evaluation criteria, which are gathered in a fitness function to reflect the degree to which the individuals solve the problem. This approach is inspired by the Darwin survival of fittest principle.

In section 3.1 the discrete parameterization of individuals is defined. There are two types of design variables: topological variables (blocks) and boundary conditions variables (fixed nodes, contacts and actuator input dof, see section 2.3). However, the designer must provide specifications, such as the mesh, the output requirements, and the design criteria, which are usually based on strain energy, strokes or forces at the output nodes (see section 3.3). Section 3.2 describes the stochastic operators that are used to modify blocks variables.

3.1. Discrete variable parameterization

The size of the design domain, the output requirements, and the number of blocks to be used (mesh size), are fixed parameters. The size of the design domain, and the output requirements are given by the specifications of the design problem. The size of the mesh requires to the designer's expertise: it must be large enough to describe complex solutions, but not too large in order to use the power of building blocks description.

Then the designer has to define the discrete variable parameterization that will define the search space. For this he has to answer the following questions:

- Discrete topology variables:
  - which types of building blocks will be used (blocks variable)?
  - what size the blocks can have (see a, b in fig. 3) ? The designer gives discrete values of blocks lengths and blocks widths.
  - which materials can be used (E_y, ν, ρ) ? The designer gives discrete values of Young modulus, Poisson ratios and densities.
  - what thickness range is possible for these materials (ep) ? The designer gives discrete values of blocks thickness.
- Discrete boundary-conditions variables, which are steady for all output dof:
  - what are the minimum and maximum numbers of fixed nodes ?
  - where the fixed nodes are allowed ?
  - what are the minimum and maximum numbers of contacts ?

- where does the designer allow contacts ?
- what is the backlash in the contacts ?

- Discrete boundary-conditions variables, which change for each output dof:
  - what are the minimum and maximum numbers of actuators ?
  - where does the designer allow the actuators ?
  - what are the forces provided by actuators and the maximum strokes ?
  or what are the strokes provided by actuators and the maximum forces ?
  - what is the stiffness of each actuator ?

To explore such a discrete search space, a genetic algorithm has been developed. This algorithm is based on NSGA2,[14] a fast and elitist multi-objective genetic algorithm. While exploring the search space, the genetic algorithm keeps the most adequate solutions with regard to the design criteria. Next subsection focuses on the stochastic operators for the building block variables.

3.2. Stochastic operators for blocks variables

The stochastic operators of a genetic algorithm (see **2** in fig.1) are the selection of the initial population, the mutation and the crossover. These operators are representation dependent. The building blocks assembly's genotype is coded by a matrix of integers. Each integer represents one building block (example: 1 for 'empty', 2 for 'cross', 3 for 'first triangle', 4 for 'second triangle'... see fig. 4).

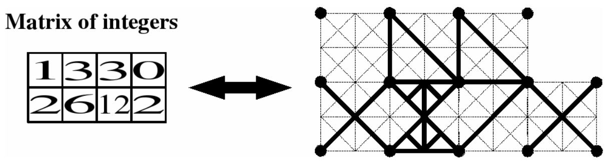

Figure 4. Genotype of a blocks assembly

Let *popsize* be the size of the population. At the initialization, a set of *popsize* matrices is randomly generated. An experimented designer can also add several solutions into this first generation. The initial population is evaluated, and if objectives are reached, the algorithm stops (see **2** in fig.1). Else a tournament selects *popsize*/2 couples of genitors among the parents. The genitors reproduce: a random matrix of binary is used to generate two offsprings' genotypes with two genitors' genotypes (see fig. 5). Then *pmut* % of generated genotypes mutate: *pmutb* % of the blocks are randomly changed. The offspring are evaluated: the Pareto rank is also computed for multi objective optimization. Then among the parents and the offspring, *popsize* solutions are selected and replace the old generation: a variation of NSGA2 method[14] has been used for replacement. The algorithm continues until a stopping criterion is reached or when the number of generations reaches *nbgenemax*, *nbgenemax* being the maximum number of generations given by the designer.

Next subsection is about the evaluation of the individuals, which is the most time consuming step of the genetic algorithm.

3.3. Evaluation of individuals

Every generation, the criteria of *popsize* individuals are evaluated. First a Matlab^{TM} function verifies that boundary conditions are correct: fixed nodes, input nodes (actuators) and output nodes must be linked by a single piece assembly of blocks. A wrong individual is automatically rejected at the last rank for selection without evaluation. The correct individuals' global matrices (K_{global}) are inverted, from which several criteria can be evaluated:

- the displacement *maxds* at the output nodes.

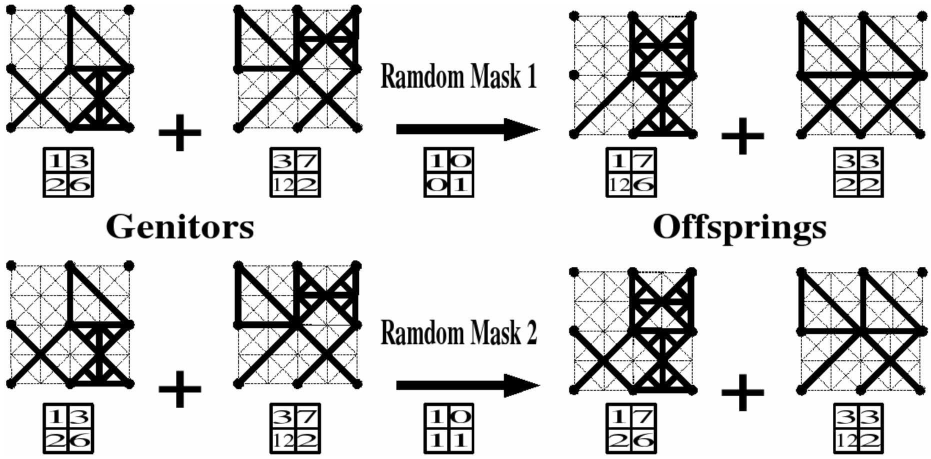

Figure 5. Crossover operator for the blocks

- the ratio *ampde* between the displacement at the output node and the displacement at the input node provided by the actuator. If there are several outputs and actuators, *ampde* is the ratio between a linear combination of outputs displacements and inputs displacements. This criterion is similar to the geometric advantage.[15]
- the ratio $ampfb$ between the force at the output node and the force provided by the actuator. There are several variations of this criterion depending on the conditions at the output node: a spring (a stiffness is added at the output dof), a fixed frame (the output dof is removed) or a contact (a backlash between the output node and a fixed surface).
- the ratio *msese* between the displacement at the output port, and the strain energy.[9]
- a criterion *sqdet* inspired by the well-known manipulability criterion and used to quantify the isotropy in motion transmission of robotic mechanisms.[16] The condensation method reduces the compliant mechanism stiffness matrix K_{global} to inputs (F_{red}) and outputs (U_{red}) dof:

$$\begin{aligned} K_{global} &= \begin{pmatrix} K_{output} & K_{oiput} \\ K_{oiput}^t & K_{input} \end{pmatrix} \\ K_{red} &= (K_{output} - K_{oiput}^t\, K_{input}^{-1}\, K_{oiput}) \text{ with } F_{red} = K_{red}\, U_{red} \end{aligned}$$

We propose the 'manipulability' criterion for compliant structures: $sqdet = \sqrt{det(K_{red}^{-t}\, K_{red}^{-1})}$.

Finally, several constraints such as the maximum force or stroke of specific actuators are taken into account. As an example, let us assume that the algorithm is computing the criterion of a compliant structure whose actuator delivers a force of $1600\mu N$ over a limited stroke of $100\mu m$. In this example, the constraint is that the actuator node displacement must be less than $100\mu m$. If the $1600\mu N$ force induces the actuator node's displacement to be larger than $100\mu m$, then the compliant structure is too flexible, and its criterion is penalized.

4. A TWO-OUTPUT DOF EXAMPLE

4.1. Specification

This section illustrates the use of FlexIn on a simple example. The objective is to design a two-output dof compliant mechanism:
- that can fit in a $2mm \times 2mm$ square,
- actuated by two piezoelectric actuators that can provide $1\mu m$ displacement each,

- allowing to amplify the stroke of both actuators by about 10, in order to have a $10\mu m \times 10\mu m$ workspace. The stroke ratio of each output dof is an adequate criterion for this problem. At this size, we assume that the compliant mechanism will be made by surface micromachining technology, and that the sixteen building blocks described in subsection 2.2 are adequate. Mainly two materials can be used, and we impose the thickness to be constant:
- polysilicon (Young modulus in megapascals $E_y = 192000MPa$, and Poisson Ratio $\nu = 0.3$), constant thickness ($ep = 2\mu m$ or $ep = 4\mu m$);
- SU8 resin ($E_y = 4020MPa$, $\nu = 0.22$), constant thickness ($ep = 20\mu m$ or $ep = 40\mu m$).

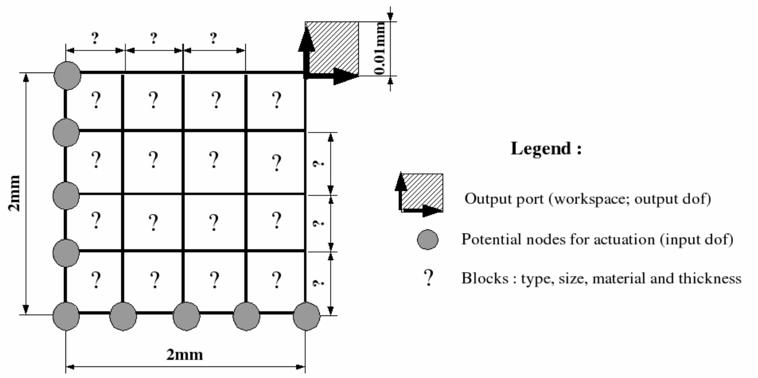

Figure 6. Specifications with FlexIn

We assume that a sixteen building blocks assembly (4×4 mesh) is large enough to describe complex solutions, but not too large in order to use the power of building blocks description. The solutions will have two fixed nodes, but we do not make any assumption about their location. No contact is wanted. For each output dof, we impose one piezoelectric actuator, which has a $1\mu m$ stroke, a maximum force of $10N$, and whose stiffness is $10^6 N/m$ in actuated dof (and $0N/m$ in orthogonal direction). We assume that these actuators should be located on outer nodes (see fig. 6).

Using a Matlab^{TM} interface, the designer can easily enter these specifications into FlexIn. The solutions found by FlexIn's genetic algorithm are shown in the next section.

4.2. Optimization

When the genetic algorithm stops, the best compromises are kept. The criteria and the material of three of these optimal solutions are shown in table 1. All these solutions are made of polysilicon. SU8 resin is too soft for this application with piezoelectric actuators. As specified, all these solutions have two fixed nodes, and two actuators.

Table 1. Criteria, material and thickness of solutions found by the genetic algorithm

Solution	Stroke ratio of DOF 1	Stroke ratio of DOF 2	Material and Thickness	Fixed nodes and Actuators
A	5.4	17.2	polysilicon, $4\mu m$	2, 2
B	10.4	10.3	polysilicon, $2\mu m$	2, 2
C	16.7	-5	polysilicon, $2\mu m$	2, 2

The criteria of these solutions are plotted on a Pareto graph (see fig.7). The designer can choose among these solutions. The solutions A, B and C are also displayed on figure 7. The solution B seems to be the best compromise (dof 1 $ampde = 10.4$ and dof 2 $ampde = 10.3$). The FEA of solution B has been done with Cast3m^{TM} code. The maximum Von Mises stress is $68MPa$ for the first output dof and $7MPa$ for the second one : it is far below the yield stress of polysilicon ($1200MPa$). However, the critical load of buckling is nearly reached: the security factor is 1.05 for the first dof, and 1.35 for the second dof. In the next section, interpretation, FEA and improvement of these solutions are presented.

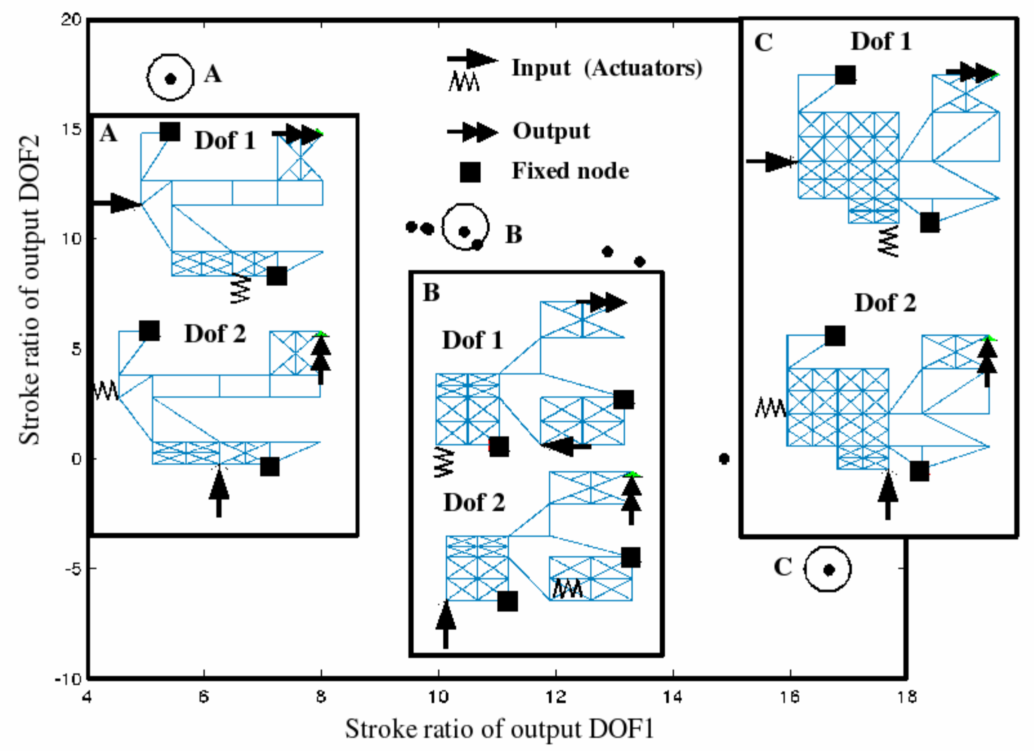

Figure 7. Pareto front of solutions, only blocks assembly of solutions A, B and C are displayed

4.3. Discussion of the results

A trial and error procedure (see **3,4,5** in figure 1) quickly leads to a simplified solution from solution B (see fig.8). During this procedure, a more detailed analysis than during the optimization phase is made. The same hypotheses as made in section 2.3 have been formulated, but the simulation has been done on a linear beam network, which has about 400 nodes (1200 dof), whereas the model used for the genetic algorithm evaluation has only 25 nodes (75 dof). Moreover buckling phenomenon and Von Mises stress are computed.

First, the output port has been reinforced (see fig. 8): a full block is used and a beam is added to support it. Then the actuator of the first dof is moved to the right bottom corner: this modification allows an easy fixation of actuator without significant change in the performances of the compliant structure. The last modification is the removal of beams on the left up corner. Neither the thickness of the structure, nor the width of each beam has been modified. During this trial an error procedure, a FEA has shown that performances of the solution have been globally improved:

- the stroke ratio of dof 1 is increased: $ampde = 12.1$ instead of 10.4, whereas the stroke ratio of dof 2 is quite the same: $ampde = 10.25$ instead of 10.3.
- the maximum Von Mises Stress for dof 1 is reduced to $56MPa$ instead of $68MPa$, whereas it is increased to $21MPa$ instead of $7MPa$ for the second dof. This computed stress remains far below the yield stress of polysilicon ($1200MPa$).

- the critical load of buckling is increased for the first dof, and decreased for the second one, leading respectively to security factors of 1.25 and 1.15

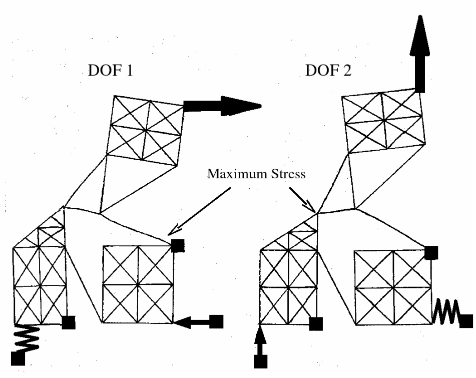

Figure 8. Conceptual design of a 2 dof compliant mechanism

As a conclusion, FlexIn conceptual design method has quickly led to a compliant solution verifying the design requirements. A prototype could be designed from these results in order to validate the performances of the solution (see **6** in fig.1). Further applications of FlexIn have been performed[17].[18]

5. CONCLUSION

A new conceptual design method of compliant mechanisms has been presented. This method considers a compliant mechanism as a basic assembly of compliant building blocks. The use of blocks is a natural approach for engineers who can incorporate design experience. In FlexIn toolbox, the blocks have four nodes and are defined with frame elements, which lead to short computation time. Moreover the use of blocks gives a discrete variable parameterization which reduces the search space and leads to more tractable optimization problems. The efficiency of calculation and the discrete variable parameterization allow the use of a genetic algorithm to optimize the assembly of compliant building blocks. Thus different discrete possibilities may be explored such as materials possibilities, fixed frames locations, actuators locations, and contacts locations. Moreover, the use of a genetic algorithm allows a multi criteria optimization with non-smooth criteria, and also provides several solutions plotted on a Pareto graph. Of course these solutions could be further optimized after experimental validation of the concept.

One of the perspective of this work is the refinement of blocks models thanks to experimental knowledge acquired with first prototypes. Another perspective is the extension of this conceptual method to spatial mechanisms.

REFERENCES

1. G. K. Ananthasuresh and S. Kota, “Designing compliant mechanisms,” *Mechanical Engineering* , pp. 93–96, November 1995.
2. M. P. Bendsoe and O. Sigmund, *Topology Optimization: Theory, Methods and Applications*, Springer, 2003.
3. S. T. Smith, *Flexure*, Gordon and Breach Science Publishers, 2000.
4. S. Henein, *Conception de guidages flexibles*, Presses polytechniques et universitaires romandes, 2001.
5. Y. M. Moon, B. P. Trease, and S. Kota, “Design of large displacement compliant joints,” in *ASME Design Engineering and Technical Conference*, pp. MECH–34207, (Montreal), 2002.
6. L. L. Howell, *Compliant mechanisms*, John Wiley and Sons Inc., 2001.
7. G. Allaire, *Shape optimization by the homogeneization method*, Springer, 2002.

8. G. Allaire, F. Jouve, and A. M. Toader, "Structural optimization using sensitivity analysis and a level-set method," in *Internal report 508*, C. E. Polytechnique, ed., 2003.
9. M. I. Frecker, G. K. Ananthasuresh, S. Nishikawi, N. Kikuchi, and S. Kota, "Topological synthesis of compliant mechanisms using multi-criteria optimization," *Journal of Mechanical design* **119**, pp. 238–245, 1997.
10. A. Saxena, "On multiple material optimal compliant topologies : discrete variable parametrization using genetic algorithm," in *ASME Design Engineering and Technical Conference*, pp. MECH–34209, (Montreal), 2002.
11. A. Mettlach and A. Midha, "Multiple segment pseudo rigid body model concept in compliant mechanism and analysis," in *ASME Design Engineering and Technical Conference*, pp. MECH–14148, (Baltimore, Maryland), 2000.
12. A. Saxena, X. Wang, and G. K. Ananthasuresh, "Pennsyn : a topology synthesis software for compliant mechanisms," in *ASME Design Engineering and Technical Conference*, pp. MECH–14139, (Baltimore), 2000.
13. N. D. Mankame and G. K. Ananthasuresh, "Contact aided compliant mechanisms : concept and preliminaries," in *ASME Design Engineering and Technical Conference*, pp. MECH–34211, (Montreal), 2002.
14. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: Nsga-2," in *Parallel Problem Solving from Nature VI Conference*, 2002.
15. S. Kota, J. Joo, Z. Li, S. M. Rodgers, and J. Sniegowski, "Design of compliant mechanisms : applications to mems," *Analog Integrated Circuits and Signal Processing* **29**, pp. 7–15, 2001.
16. T. Yoshikawa, "Manipulability of robot mechanisms," *The International Journal of Robotics Research* **4**, pp. 3–9, 1985.
17. C. Bolzmacher, M. Hafez, M. B. Khoudja, P. Bernardoni, and S. Dubowsky, "Polymer based actuators for virtual reality devices and rehabilitation applications," in *Smart Structure 2004*, SPIE, ed., *Fabrication and Characterization Techniques*, (San Diego), 2004.
18. P. Bernardoni, A. Riwan, H. Tsitsiris, P. Bidaud, O. Millet, and L. Buchaillot, "From the mechanical analysis of a polyarticulated micro gripper to the design of a compliant micro gripper," in *Smart Structure 2004*, SPIE, ed., *Modeling, Signal Processing, and Control*, (San Diego), 2004.