6th World Congresses of Structural and Multidisciplinary Optimization
Rio de Janeiro, 30 May - 03 June 2005, Brazil

Optimal synthesis of mechanisms using time-varying dimensions and natural coordinates

J.-F. Collard[1] , P. Fisette[1], P. Duysinx[2]

[1]: Centre for Research in Mechatronics, Université Catholique de Louvain (UCL), Belgium
[2]: Department of Mechanics and Aerospace, Université de Liège (ULg), Belgium
e-mail: collard@prm.ucl.ac.be, fisette@prm.ucl.ac.be, P.Duysinx@ulg.ac.be

1. Abstract
This paper presents a simple approach to optimize the dimensions and the positions of 2D mechanisms for path or function-generator synthesis. The proposed method is particularly adapted to assembled mechanisms since time-varying dimensions always satisfy the assembly conditions which may represent a real difficulty when dealing with closed-loop mechanisms. The objective is to minimize the strain energy of the bars - considered as flexible - of the mechanism when this one follows perfectly the desired path. Two optimization strategies are developed and criticized. The first one is based on separated optimizations of design parameters and point coordinates. The second one is more global and is performed in two stages : multiple local synthesis are needed first to find the initial point coordinates, and then a global synthesis stage is undertaken to find both the best dimensions and coordinates. The use of natural coordinates is also particularly interesting since the only non linear functions to optimize are distance functions, and the objective function is rather well-conditioned for a gradient-based optimizer.

The question of finding the global optimum is addressed and discussed. Since a standard genetic algorithm may fail to find it, a different approach is proposed: exploring the design space to find several local optima among which the designer will choose the most relevant one taking other design constraints into account. A simple technique is applied which consists in running multiple optimization processes starting from uniformly-distributed initial dimensions (full-factorial design of experiments) across the parameter space. Three applications are presented: a simple four-bar path synthesis to illustrate the optimization strategies, a four-bar steering linkage synthesis for function generation – Ackermann relation – to highlight the limits of both strategies, and eventually a six-bar steering mechanism to explore the design space and find different local optima.
2. Keywords: optimal synthesis, natural coordinates, closed-loop mechanisms

3. Introduction
Optimization of complex multibody systems represents a real present interest along with the increasing development of computer resources. This is particularly true considering closed-loop mechanisms whose assembling constraints have to be satisfied before any analysis. They thus need a special attention when evolving the optimization process strategy. A few solutions have been proposed to deal with them. For example, the authors have suggested to penalize properly the objective function using the conditioning of the constraints Jacobian matrix [1]. This extends the parameter space and leads to a well-defined objective function but remains difficult to optimize. Another well-known approach in mechanisms path synthesis is to deform the mechanism subject to a perfect following of the desired path [2, 3, 4]. From this point of view, the path-following objective becomes a optimization constraint while the deformation energy is the actual objective to minimize. Therefore, the mechanism naturally assembles during the optimization process while the objective is fulfilled.

In optimal synthesis, a second issue is the choice of the formalism to describe the geometry of the mechanism. Among the different possibilities, one can mention the common use of relative coordinates in real [5, 6, 7, 8] or complex [9] form. This formalism has the advantages to limit the number of assembling constraints but it introduces trigonometric functions involving angular variables. This enhances the non-linearity of the problem and makes the optimization more tricky. The use of natural or point coordinates is also wide-spread [2, 4, 10, 11]. In comparison with relative coordinates, natural coordinates involve additional algebraic constraints. However, these equations only consist of linear or distance functions. This coordinates system is thus well suited to gradient-based optimization techniques such as least squares methods.

The proposed method tries to combine these two features: deformable mechanisms and natural coordinates. The first one enables to solve the problem of non-assembly while the second one greatly simplifies the type of objective function. The optimization strategy is based on the minimization of the deformation energy over the followed path, and a subsequent update of the mean lengths as suggested in [5]. This minimization-update sequence is repeated until convergence when the total deformation energy does not vary anymore or is sufficiently low. The method has a rather slow convergence rate but shows a monotonic decrease, and may sometimes fail to find the global optimum. To improve the convergence rate, the technique is extended to a global synthesis approach as proposed by [4]. Afterwards, this improvement has enabled to outline different local optima starting from different initial parameters. The choice of the optimal mechanism among these local optima relies on other design constraints which are difficult to compute and not taken into account in the original problem. By the way, it is also shown that the use of a standard genetic algorithm does not nexessarily enable to reach the global optimum. Since it is interesting for the designer to find all the best mechanisms, a mapping of the design space is proposed to explore most of the possible optima.

Different kinds of requirements may be encountered in dimensional mechanisms synthesis: path or function generation, body guidance, or mixed problems. Most applications concern path synthesis problems [5, 6, 7, 8, 9] and the four-bar mechanism will constitute a running example in the following. More realistic applications of function-generation synthesis will also be given based on the Ackerman steering linkage problem: a four-bar and then a six-bar synthesis. These last ones have been suggested by [12, 13, 14].

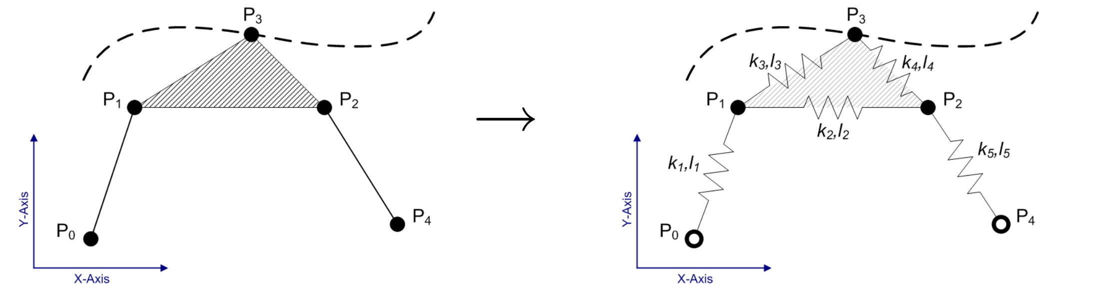
The paper is organized as follows: in Section 4, the general optimization problem is formulated in terms of objective function and sensitivity analysis. In Section 5, we develop the optimization strategy applying two methods to the path synthesis of a four-bar mechanism as running example. Section 6 presents a more realistic application of function generation synthesis for the four-bar Ackerman steering linkage. Section 7 deals with the mapping of the design space to find different local optima. Before some conclusions and prospects in Section 9, the application of the mapping technique is performed on a six-bar steering linkage in Section 8.

4. Problem formulation

The problem formulation is divided into two subsection: first, the objective function is evolved and then the corresponding sensitivity analysis is worked out.

4.1. Objective function

Let us consider the well-known 2D example of a four-bar mechanism which has to follow a desired path in dotted line (see Figure 1.a). In order to make the mechanism exactly follow the given path, the four-bar is modeled by a deformable mechanism replacing the bars and triangle by five springs whose stiffness are k_j and natural lengths l_j, $j = 1 \dots 5$ (see Figure 1.b).



a. Four-bar mechanism... b. ...modeled by a deformable mechanism

Figure 1: Path synthesis of a four-bar mechanism

The desired path is discretized in N points, leading to N different configurations of the mechanism. When it moves, the different points $P0 \dots P4$ composing the mechanism have different behaviors: $P0$ and $P4$ stay at the same place, $P3$ follows exactly the N points on the path and $P1$ and $P2$ are free. These points can thus be arranged in three groups: the *static* points $P0, P4$, the *moving* point $P3$ and

the *floating* points $P1, P2$. Their absolute coordinates are saved respectively in the following vectors: SP, MP and FP. As the moving point and the floating points may have different coordinates at each configuration, MP and FP are referenced by the index i: MP_i and FP_i, $i = 1 \dots N$.
Grouping the natural lengths l_j in the column vector L and the stiffness parameters k_j on the diagonal of the stiffness matrix K, we define the global strain energy as a scalar cost function:

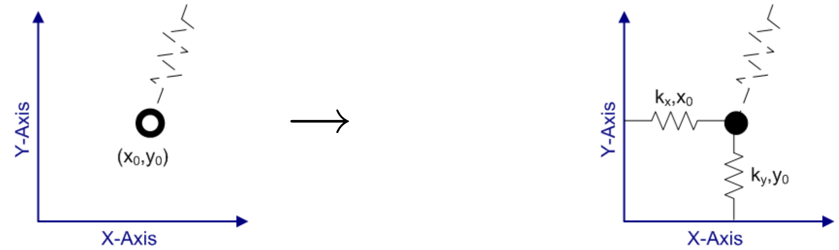
$$E\left(SP, MP_1, \dots, MP_N, FP_1, \dots, FP_N, L, K\right) = \frac{1}{2}\sum_{i=1}^{N}\left(D_i - L\right)^T K \left(D_i - L\right) \tag{1}$$

where D_i is a column vector containing five distance functions $d_j(SP, MP_i, FP_i)$, $j = 1 \dots 5$, between each couple of points. For the moment, the only known parameters are the $2N$ coordinates of the moving point. The stiffness parameters k_j may be chosen by the user. A priori, they are all equal to unity but this may change afterwards making some bars stiffer than others if necessary. Therefore, the optimization problem is stated as follows:

$$\min_{SP, FP_1, \dots, FP_N, L} \frac{1}{2}\sum_{i=1}^{N}\left(D\left(SP, MP_i, FP_i\right) - L\right)^T K \left(D\left(SP, MP_i, FP_i\right) - L\right) \tag{2}$$

where the actual design parameters are SP and L. This constitutes an obvious non-linear least squares optimization problem. In the following, two propositions are given to improve the homogeneity of the problem and to avoid multiple closed-loop configurations.

Firstly, remark that the actual design parameters SP and L are located in different parts of the objective function. This makes the objective differently sensitive to both of them. We propose to transform each static point coordinates into the natural lengths of two springs (see Figure 2). In this way, a new floating point is inserted in FP, the vector SP is appended to the vector L and two new stiffness parameters are added to the diagonal of matrix K. Note that the corresponding distance functions d_j becomes actually the two coordinates which are not always positive: this introduces so-called *oriented* springs according to the sign of their natural lengths. Doing this transformation, all the design parameters are grouped in the same vector L.



a. Static point... b. ...transformed into a floating point and two springs

Figure 2: New model of static points

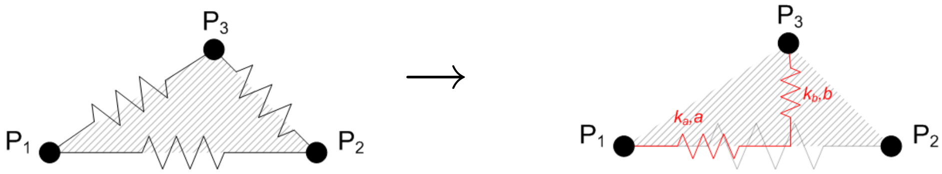
The second proposition relates to the three springs composing the triangle $P1, P2, P3$. Fixing the points $P1$ and $P2$, two stable positions remain for $P3$: above or below the $P1 - P2$ line. To remove the ambiguity, the use of *oriented* springs (see above) is proposed to locate univoquely $P3$ with respect to $P1$ and $P2$. Thus, the two springs $P1 - P3$ and $P2 - P3$ are replaced by two perpendicular *oriented* springs as shown in Figure 3.

Finally, taking both propositions into account, the objective function (see Eq. 1) becomes:

$$E\left(MP_1, \dots, MP_N, FP_1, \dots, FP_N, \tilde{L}, \tilde{K}\right) = \frac{1}{2}\sum_{i=1}^{N}\left(\tilde{D}_i - \tilde{L}\right)^T K \left(\tilde{D}_i - \tilde{L}\right) \tag{3}$$

leading to the following rearranged optimization problem:

$$\min_{FP_1, \dots, FP_N, \tilde{L}} \frac{1}{2}\sum_{i=1}^{N}\left(\tilde{D}\left(MP_i, FP_i\right) - \tilde{L}\right)^T \tilde{K} \left(\tilde{D}\left(MP_i, FP_i\right) - \tilde{L}\right) \tag{4}$$



a. Triangle element... b. ...transformed by replacing two springs

Figure 3: New model of triangle element

where the tilde symbol stands for both modifications described above.

4.2. Sensitivity analysis

Two kinds of optimization variables are now considered (see Eq. 4): $\tilde{L}$ and FP_i, $i = 1 \dots N$. The gradients of the global energy function with respect to these both vector are given by:

$$\frac{\partial E}{\partial FP_i} = \frac{\partial \tilde{D}^T (MP_i, FP_i)}{\partial FP_i} K \left(\tilde{D} (MP_i, FP_i) - \tilde{L} \right) \tag{5}$$

$$\frac{\partial E}{\partial \tilde{L}} = -\sum_{i=1}^{N} \tilde{K} \left(\tilde{D} (MP_i, FP_i) - \tilde{L} \right) \tag{6}$$

Remark that Eq. 5 only depends on the i^{th} configuration if the design parameters $\tilde{L}$ are fixed. This means that the sensitivity of the global energy function with respect to the i^{th} floating point coordinates vector is independent of the other configurations if the design parameters are known. This is the basis of the optimization strategy described in the next Section.

5. Optimization strategy

The optimization strategy has been developed from a first simple approach to a more global synthesis. The original idea relies on multiple local optimization followed by the update of the length parameters. This update strategy is then extended to make the method more global.

5.1. Mean values update

This strategy is inspired from Hansen [5] who proposed to minimize the deviation of each variable dimensions over a cycle and to update the mean value after each cycle. The main difference here is the use of natural coordinates instead of relative coordinates. The corresponding algorithm flowchart is summarized in Figure 5.a. Starting from given values of the design parameters $\tilde{L}$, the algorithm begins minimizing the total energy with respect to the FP_i. This is equivalent to solving N local optimization problems because the FP_i are independent and $\tilde{L}$ is constant:

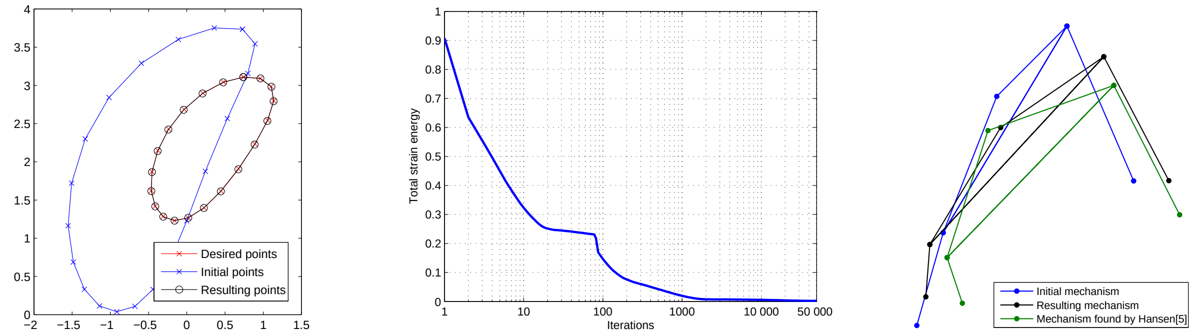
$$\min_{FP_1,\dots,FP_N} \frac{1}{2} \sum_{i=1}^{N} \left(\tilde{D}_i - \tilde{L} \right)^T \tilde{K} \left(\tilde{D}_i - \tilde{L} \right) \Leftrightarrow \frac{1}{2} \sum_{i=1}^{N} \min_{FP_i} \left(\tilde{D}_i - \tilde{L} \right)^T \tilde{K} \left(\tilde{D}_i - \tilde{L} \right) \tag{7}$$

These optimum distance functions $\tilde{D}_i$ enable to compute new design parameters $\tilde{L}$ by taking their mean values $\frac{\tilde{L}}{N}$. This is done in the second step of the algorithm (see Figure 5.a) and refers to the necessary condition for a local minimum (see Eq. 6):

$$\frac{\partial E}{\partial \tilde{L}} = 0 \Leftrightarrow \tilde{L} = \frac{1}{N} \sum_{i=1}^{N} \tilde{D} (MP_i, FP_i) \tag{8}$$

Note that a simple Levenberg-Marquardt algorithm is used for the local least squares optimization problems which only involve a few variables (e.g. 9 design parameters in the case or the four-bar).

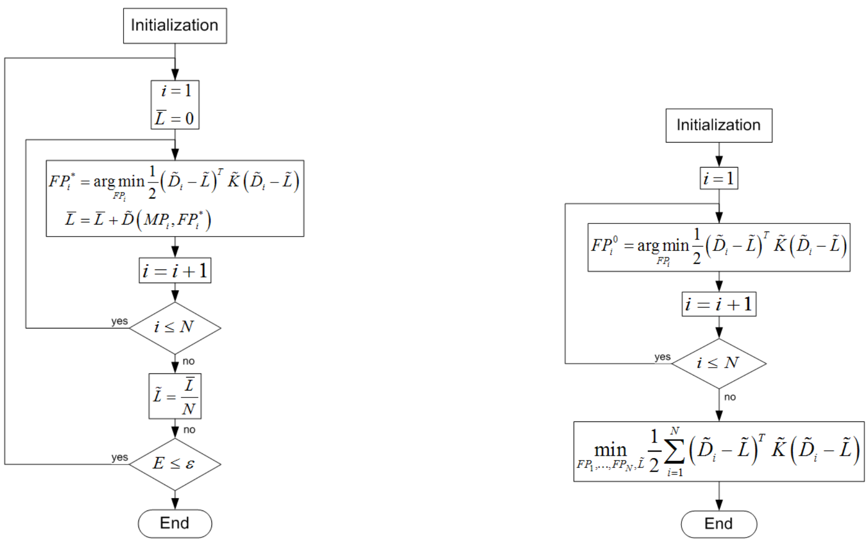
Figure 4 presents a four-bar path synthesis example excerpted from [5]. The initial, desired and resulting path are shown in Figure 4.a. In Figure 4.b, the global energy evolution is plotted. Figure 4.c presents the initial and resulting mechanisms as well as the one obtained in [5].



a. Path synthesis b. Evolution of total energy c. Compared mechanisms

Figure 4: Optimization example with the "Mean update" algorithm

As it can be observed in the Figure 4.b, this algorithm works well but is slow convergent. Moreover, it may fail in some cases, e.g. when the minimum energy function draws a valley in the parameters space (see Section 6.2., Figure 8). This explains the new improvement developed in the next Section.



a. "Mean update" algorithm b. Local and global synthesis

Figure 5: Optimization algorithm flowcharts

5.2. Global synthesis

What is proposed here is to replace the natural lengths update by the global synthesis proposed by [4]. The new algorithm is represented in Figure 5.b. The beginning is the same as for the previous algorithm but the multiple local optimizations (see Eq. 7) are performed only once. They are used as hot starting points for the global optimization which involves all the floating points coordinates as well as all the design parameters. The number of optimization parameters may increase rapidly (e.g. $9 + 4N = 49$ for the four-bar mechanism and 10 synthesis points) if the mechanism and/or the path get more complex. As the parameters space is larger, a more robust optimization algorithm is needed: for instance, the so-called dog-leg algorithm described by Powell [15]. This trust-region method is well-known to solve systems of nonlinear equations. Applied to the four-bar steering synthesis, this method gives better results than the previous one as shown in Section 6.2.

6. Application to steering linkage synthesis

This Section presents an interesting application of function generation synthesis. The goal is to optimize steering linkage of vehicles. In the first subsection, the function to generate is established from the Ackermann condition. Secondly, both previous optimization strategies are applied to a four-bar steering linkage and then compared.

6.1. The Ackermann condition

One of the main requirements of the steering mechanism of a vehicle is to give to the steerable wheels a correlated turning, ensuring that the intersection point of their axis lies on the extension of the rear wheel axis (point P in Figure 6). The Ackermann relation of correct turning is:

$$\cot \delta_o - \cot \delta_i = \frac{l}{L} \tag{9}$$

where δ_o and δ_i are the outer and inner wheel angles respectively, l is the wheel track and L the wheelbase of the vehicle. Only the wheel track-wheelbase ratio influences this Ackermann steering relation.

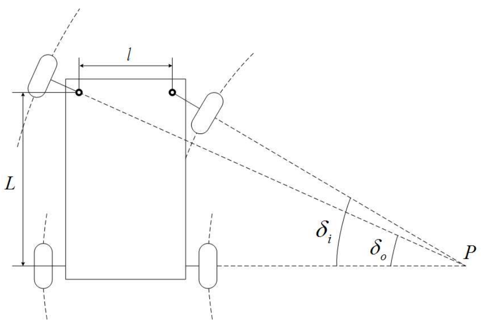
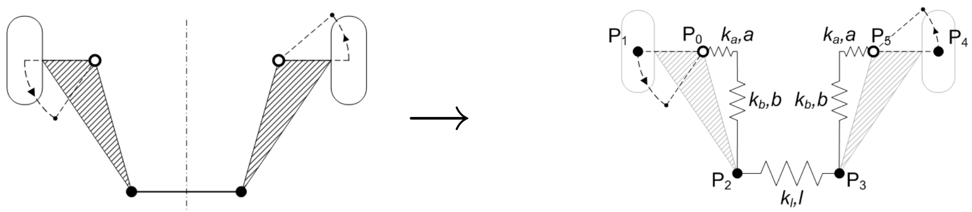


Figure 6: The Ackermann condition

6.2. Four-bar steering linkage synthesis

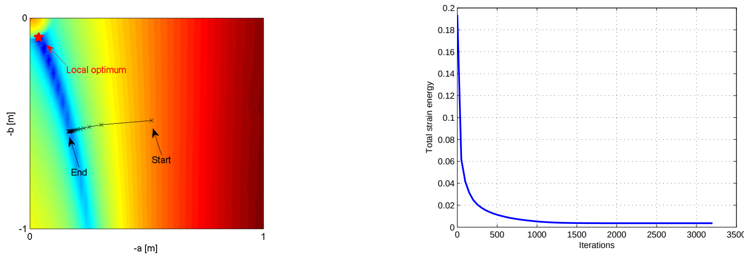
The modeling of the four-bar steering linkage is worked out according to the rules depicted in Section 4.1. To satisfy the Ackermann condition of Eq. , the correlated path-following of the wheel centers are imposed while the inner wheel angle takes 20 different values between 0 and the maximum (See Figure 7). Also observe in the Figure that the static points are not transformed into floating points because they do not belong to the design parameters . These parameters are made of three natural lengths a priori: a, b and l. However, the problem symmetry reduces their number to only two − a and b − because: $l = \left\| \overrightarrow{P0P5} \right\| - 2a$. Note that this is particularly interesting to visualize the objective function in two dimensions once minimized with respect to the FP_i.



a. Four-bar steering linkage... b. ...modeled by a deformable steering linkage

Figure 7: Function generation with a four-bar mechanism

As announced before, this nice example can show some drawbacks of both previous strategies. As the minimum energy function (see Eq. 7) draws a valley in the design parameters space, the evolution of the mean update algorithm may not converge to the local optimum (see Figure 8.a). Starting from $a = -0.5m$ and $b = 0.5m$, the process stops after 3250 iterations because the total energy does not change anymore (see Figure 8.b). Remark that the function is penalized around the origin to avoid singular configurations of the mechanism.



a. Evolution of the design parameters b. Evolution of the total energy

Figure 8: Wrong convergence of the "mean values update" strategy

As for the global synthesis algorithm, it is observed that the optimization process may reach one of both local optima [13]. Starting from different initial parameters, it is sometimes hard to guess where it converges. The Figure 9 shows that running the algorithm from initial points located on a uniform 7-by-7 grid, these processes lead to 47 relevant optimization results. Among these results – symbolized by non-bold "o" and "x" –, 11 of them converge to one local optimum – symbolized by bold "**x**" – while the 36 others reach another one – symbolized by bold "**o**" – which is actually the global one. The optimization method is thus not global.

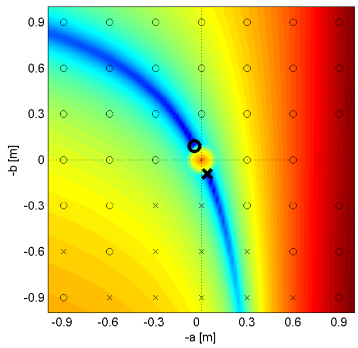


Figure 9: Optimization of the steering linkage from different starting points

The two best linkages are drawn in Figure 10: a trailing one and a leading one. Their steering error functions are plotted in Figure 11 which represents the deviation of the outer wheel angle with respect to the Ackermann condition when the inner wheel turns from 0° to 40°. The small difference between both mechanisms performances do not justify the selection of one instead of the other. The last decision comes to the designer who will perhaps choose the worse trailing mechanism because of its smaller dimensions.

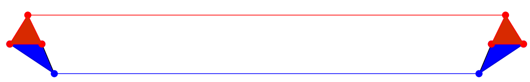


Figure 10: Two local optima found for the 3-bar steering linkage

7. Exploration of the design space

Finding the unique global optimum is not easy even with genetic algorithms as shown in the next Subsection. However, as explained in the previous example, it may be interesting to propose several

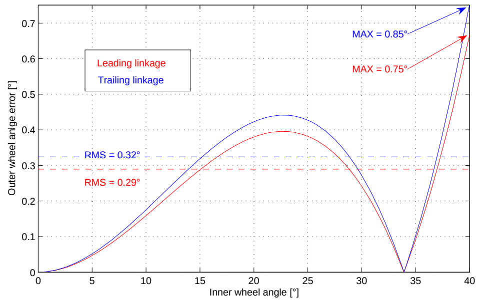


Figure 11: Steering error of both optimum linkages

local optima to the designer. The second Subsection describes a simple method to explore the entire parameters space in order to do that. This method is applied to a more complex six-bar steering linkage.

7.1. Global optimization may fail with genetic algorithms

How to find the global optimum ? Always a tricky question. One may propose to use a standard genetic algorithm to explore the design space at best. But this is not as reliable as it could seem to be. The cost function is the resulting total strain energy after the minimization with respect to the FP_i (see Eq.7). Two subsequent run of the G.A. may give different local optima for the four-bar linkage as shown in Figure 12 to compare with Figure 9.

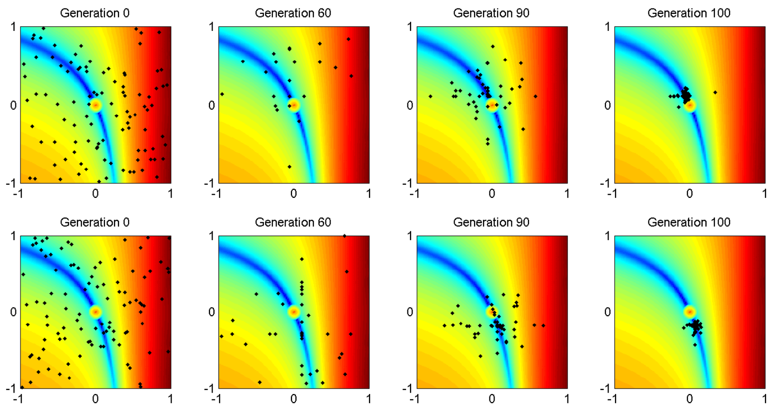


Figure 12: Two evolutions of the population distribution along the generations

7.2. Full-factorial design of experiments

The actual most relevant issue is to find not the global optimum but the one that satisfy all design constraints not necessarily taken into account in the optimization computing. If it may be easy for the four-bar linkage, this is not the case for the six-bar described in Figure 13. This model is composed of five design parameters $- a, b, l_1, l_2, y -$ which are reduced to four because of the symmetry [14]. The idea is to perform multiple optimization processes starting from points located on a full-factorial design of experiments in the parameters space. This design has firstly 2 levels leading to 2^P processes where P is the number of parameters. The level number is then increased to 3, 5, or 9 to refine the results. All these designs reuse the results of the previous levels. For example, in the case of the six-bar linkage, this gives successively 16, 65 (=81-16), 544 (=625-81) and 5936 (=6561-625) optimization processes at each of these levels. It is hoped that the number of local optima stabilizes with the increasing levels.

Regarding the six-bar steering example, the number of local optima is reported in Table 1. The grouping of all the optimization results is undertaken on the basis of the parameters vector norm, after the rejection of the non-convergent processes. When the starting grid is refined, the number of different

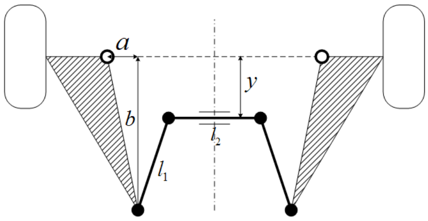


Figure 13: Six-bar steering linkage

local optima increases much slower than the number of effective processes. It could have been interesting to continue until the number of different local optima tends to a constant but it is too time-consuming. Obviously, this promising approach has to be improved in the future.

Table 1: Numerical results of the mapping

Level	number of optimizations	number of convergent optimization	number of different local optima	number of local optima per convergent optimizations
2	16	15	6	0.4
3	81	78	11	0.141
5	625	545	15	0.0275
9	6561	5591	26	0.0047

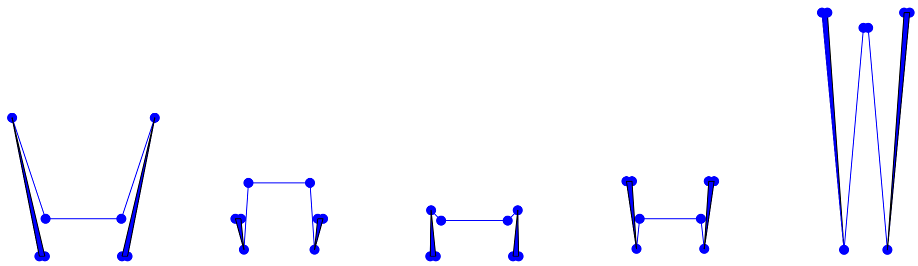


Figure 14: Five examples of local optima for the six-bar linkage

In Figure 14, five local optimum linkages are shown among the 26 of the level 9 (see Table 1). As previously explained, the selection of the most relevant one is not straightforward and depends on other design constraints. For example, one may choose the fourth mechanism because of its compactness.

8. Conclusion and prospects

Based on a strain energy approach of deformable mechanisms coupled with the use of natural coordinates, a nice optimization method has been developed to solve path synthesis and function generation problems. Divided into two stage, this method tries to minimize first the total deformation energy with respect to the point coordinates and updates thereafter the natural lengths of the springs. This first method has then been extended to cope with some limitations due to the second stage. This stage has been replaced by a global synthesis approach. It has seemed to be more robust but still not global regarding the four-bar steering linkage application. Moreover, genetic algorithms have been shown to also miss the global optimum. The interest of finding this global optimum has been reviewed and replaced by exploring the design space to find the most of local optima. A simple but time-consuming method has thus been proposed based on full-factorial design of experiments whose level increases progressively reusing previous results.

In terms of the prospects, our effort will concentrate on improving the exploring strategy of the design space. Other type of design of experiments could be used or the refinement strategy could be developed further. To cope with the problem of time consuming, a better use of the multiple optimization processes could be made, saving and exploiting intermediate informations like the function evaluations during the processes. Extending the application field is also an interesting prospect: other kinematic objectives instead of path or function-generator synthesis or even dynamical ones could be investigated. Three-dimensional mechanisms could be modeled. The more challenging issue of topology optimization of mechanisms could be tackled on the basis of this energy formulation, taking the stiffness parameters as topological parameters.

9. Acknowledgments

This research has been sponsored by the Belgian Program on Interuniversity Attraction Poles, *Advanced Mechatronic Systems - AMS* (IUAP5/06), initiated by the Belgian State — Prime Minister's Office — Science Policy Programming (IUAP V/6). The scientific responsibility is assumed by its authors.

10. References

[1] J.-F. Collard, P. Fisette, and P. Duysinx. Contribution to the optimization of closed-loop multibody systems: Application to parallel manipulators. *Multibody System Dynamics*, 2005, 13, 69–84.

[2] J. M. Jiménez, G. Alvarez, J. Cardenal, and J. Cuadrado. A simple and general method for kinematic synthesis of spatial mechanisms. *Mechanism and Machine Theory*, 1997, 32(3), 323–341.

[3] Javier Vallejo. Nonlinear optimization of planar linkages for kinematic syntheses. *Mechanism and Machine Theory*, 1995, 30(4), 501–518.

[4] J.A. Alba, M. Doblaré, and L. Gracia. A simple method for the synthesis of 2D and 3D mechanisms with kinematic constraints. *Mechanism and Machine Theory*, 2000, 35, 645–674.

[5] John M. Hansen. Synthesis of mechanisms using time-varying dimensions. *Multibody System Dynamics*, 2002, 7, 127–144.

[6] J.A. Cabrera, A. Simon, and M. Prado. Optimal synthesis of mechanisms with genetic algorithms. *Mechanism and Machine Theory*, 2002, 37, 1165–1177.

[7] M.A. Laribi, A. Mlika, L. Romdhane, and S. Zeghloul. A combined genetic algorithm-fuzzy logic method GA-FL in mechanisms synthesis. *Mechanism and Machine Theory*, 2004, 39, 717–735.

[8] Jawaharlal Mariappan and Sundar Krishnamurty. A generalized exact gradient method for mechanism synthesis. *Mechanism and Machine Theory*, 1996, 31(2), 413–421.

[9] R. Sancibrian, F. Viadero, P. García, and A. Fernández. Gradient-based optimization of path synthesis problems in planar mechanisms. *Mechanism and Machine Theory*, 2004, 39, 839-856.

[10] Mauro Da Lio, Vittore Cossalter, and Roberto Lot. On the use of natural coordinates in optimal synthesis of mechanisms. *Mechanism and Machine Theory*, 2000, 35, 1367–1389.

[11] Ole Fjendbo Jensen and John M. Hansen. Treating the problem of non-assembly for dimensional synthesis of spatial mechanisms. *Proc. of WCSMO6*, 2003, Venice, 431–432.

[12] S. Pramanik. Kinematic synthesis of a six-member mechanism for automotive steering. *ASME Journal of Mechanical Design*, dec 2002, 124, 642–645.

[13] P.A. Simionescu and D. Beale. Optimum synthesis of the four-bar function generator in its symmetric embodiment: the ackermann steering linkage. *Mechanism and Machine Theory*, 2002, 37, 1487–1504.

[14] P.A. Simionescu, M.R. Smith, and I. Tempea. Synthesis and analysis of the two loop translational input steering mechanism. *Mechanism and Machine Theory*, 2000, 35, 927–943.

[15] M.J.D. Powell. *Nonlinear Programming*, chapter A new algorithm for unconstrained optimization. Academic Press, 1970.