



HAL
open science

Outils avancés de CAO d'éléments de mécanismes et stratégies évolutives

Pascal Lafon, Laurence Moreau-Giraud

► **To cite this version:**

Pascal Lafon, Laurence Moreau-Giraud. Outils avancés de CAO d'éléments de mécanismes et stratégies évolutives. Revue Internationale de CAO et d'informatique graphique, 1999. hal-03620906

HAL Id: hal-03620906

<https://hal.science/hal-03620906v1>

Submitted on 29 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Outils avancés de CAO d'éléments de mécanismes et stratégies évolutionnaires.

Pascal Lafon — Laurence Moreau-Giraud

*Laboratoire des Systèmes Mécaniques et d'Ingénierie Simultanée (LASMIS)
Université de Technologie de Troyes
12 rue Marie Curie - BP 2060 - 10010 Troyes Cedex
{pascal.lafon, laurence.giraud}@univ-troyes.fr*

RÉSUMÉ : Dans cet article, nous introduisons une démarche d'optimisation d'éléments de mécanismes et proposons une méthode d'optimisation utilisant une stratégie évolutionnaire pour résoudre les problèmes de conception optimale. Nous commençons par présenter le contexte général des outils de calculs en conception mécanique et l'approche de certains logiciels prototypes basés sur le concept d'objet fonctionnel et de réalisation technologique des liaisons. Ensuite nous montrons comment formuler un problème d'optimisation en variables mixtes à partir de l'exemple d'une liaison pivot par roulement à billes. Puis les principes des stratégies évolutionnaires sont brièvement exposés suivis de l'application à l'exemple proposé. Enfin, les résultats obtenus montrent la bonne adéquation de ces algorithmes heuristiques à ces problèmes d'optimisation difficiles à résoudre avec des méthodes déterministes classiques.

ABSTRACT : In this paper, we introduce a methodology of mechanical components optimization using an evolution strategy in order to solve optimal design problems. We start by presenting the general context of computational tools in mechanical design and the approach of some prototype softwares based on the concept of functional object and technological realization of mechanical linkages. Then, we show how to formulate an optimization problem with mixed variables from the particular case of a ball bearing pivot link. The main aspect of the evolution strategies are briefly presented and are followed by the application to the proposed example. Numerical results emphasize the efficiency of these heuristic algorithm for mechanical components optimization problems which are difficult to solve with traditional deterministic methods.

*MOTS-CLÉS : CAO, optimisation en variables mixtes, algorithmes évolutionnaires, mécanisme.
KEY-WORDS : CAD, optimization with mixed variables, evolutionary algorithm, mechanical components.*

1. Introduction

Les logiciels de conception et de fabrication assistée par ordinateur sur le marché actuellement sont encore majoritairement centrés sur le modèle géométrique. Celui-ci reste le point de départ des principales tâches utilisant ces logiciels, et notamment les activités de calculs et de simulations. Le calcul par éléments finis est toujours considéré comme une tâche aval à celle de la modélisation géométrique, mais plus généralement le calcul en conception mécanique est caractérisé par un ensemble de tâches disséminées dans le processus de conception. Il est alors nécessaire d'effectuer certains calculs avant la modélisation géométrique puisque dans la démarche fonctionnelle d'un concepteur, la géométrie doit résulter d'un calcul [DEN 98] ; de sorte qu'à partir du moment où le modèle géométrique existe, une bonne partie de la conception est achevée.

Le modèle de produit utilisé dans tous ces logiciels repose pour l'essentiel sur l'aspect géométrique augmenté des concepts de forme fonctionnelle (features) et de structure de données diverses, notamment pour les assemblages et autres simulations cinématiques. Bien qu'indispensable, ce modèle ne permet pas de concevoir aisément un système mécanique et doit être largement complété [LAF 94].

2. Optimisation et conception d'éléments de machines

Le logiciel prototype F-CAD [DEN 98] permet la conception fonctionnelle de mécanisme à partir de primitives de conception. Le concepteur exprime sa vision conceptuelle de la solution en créant des instances de solutions technologiques et en spécifiant les relations entre ces instances. Cette approche s'appuie sur un modèle de produit structuré autour des trois concepts transversaux : ensemble, composant et liaison [DEN 98]. Les tâches de calculs sont assurées par des modules externes au logiciel et n'intègrent pas de processus d'optimisation. Le logiciel SICAM (Système Interactif de Conception Assistée de Mécanismes) [GUI 87] repose sur les notions d'objets fonctionnels et de décomposition fonctionnelle. Cette décomposition fonctionnelle, est définie comme la décomposition d'un objet technique de type connu, en sous-systèmes élémentaires (les objets fonctionnels) de plus en plus simples. Le dernier niveau de cette décomposition, le plus élémentaire est celui des liaisons mécaniques et notamment des réalisations technologiques de ces liaisons [GUI 87]. Le logiciel SICAM permet de concevoir des mécanismes composés d'engrenages montés sur des paliers à roulements. Le concepteur valide la structure du mécanisme et précise les données nécessaires au travers d'un ensemble de menus. Ce logiciel intégrant tous les modules de calculs nécessaires, optimise le mécanisme en minimisant la masse de chacun des objets fonctionnels. Le logiciel CII [HER 95] (conception intégrée intelligente), reprend les principes de SICAM et est architecturé autour d'un système expert autorisant plus de souplesse dans l'ordre et l'enchaînement des différentes étapes de la conception [HER 95]. Ces approches

ont en commun un concept de base : les solutions techniques d'une liaison ou de réalisations technologiques d'une liaison.

2.1. Réalisation technologique des liaisons

La réalisation technologique (solution technique) d'une liaison est constituée de l'ensemble des parties fonctionnelles (surfaces de contact, volumes de matières) appartenant aux différentes pièces et nécessaires pour assurer la transmission des efforts entre deux pièces principales définies dans une procédure de niveau supérieur. Cette notion de niveau fait référence ici à celle de décomposition fonctionnelle d'un mécanisme [GUI 87], accessible lorsque le type du mécanisme est parfaitement identifié.

2.2. Optimisation et conception de mécanisme

Pour un mécanisme de type connu, l'ensemble des solutions de conception possibles est l'ensemble des combinaisons des réalisations technologiques compatibles entre elles et dimensionnées de façon à satisfaire aux données du cahier des charges. Le nombre de ces combinaisons correspond au nombre de chemins de parcours d'un graphe dont les sommets sont les différentes réalisations technologiques et dont les arcs traduisent la compatibilité entre deux liaisons. Le meilleur mécanisme correspond bien au meilleur chemin de parcours du graphe au sens d'un critère déterminé, donc à la solution d'un problème d'optimisation. L'exploration de ce graphe pour un mécanisme complexe conduit à une combinatoire beaucoup trop grande pour être envisagée. De plus il est impossible de formuler mathématiquement le problème pour tout le graphe puisque l'évaluation d'un chemin est fonction des différentes réalisations technologiques (les sommets) et donc des modèles de dimensionnement utilisés. Ce problème d'optimisation ne peut être formulé que pour un chemin particulier, donc pour un ensemble de choix technologiques établis pour chacune des liaisons, et pour un objet fonctionnel comportant quelques liaisons élémentaires. La difficulté est que rien ne garantit qu'en ayant optimisé les objets fonctionnels à chaque niveau de décomposition on obtienne un mécanisme optimal [HER 95]. Ce problème d'optimisation formulé pour cet objet fonctionnel correspond alors à un problème de dimensionnement optimal, problème d'optimisation difficile à résoudre.

La démarche que nous proposons consiste à considérer que de nombreux mécanismes intègrent des objets fonctionnels communs et qu'il est possible de formuler pour ces objets fonctionnels un problème d'optimisation de manière à représenter un maximum de cas de conception. Le concepteur pourrait alors parcourir un ensemble de solutions important en assemblant des objets fonctionnels dimensionnés de manière à minimiser une fonction coût déterminée. On remarquera que cette démarche n'est applicable que dans le cas où l'on peut décomposer un

mécanisme en objets fonctionnels tel que toutes les fonctions remplies par ce mécanisme sont définies par ces objets fonctionnels et non par la structure d'agencement de ceux-ci [TOL 98].

Dans la suite de cet article nous montrons comment formuler ce problème d'optimisation pour un objet fonctionnel de type arbre + roue dentée en liaison pivot par roulement à billes, et nous proposons un algorithme d'optimisation capable de résoudre ce type de problème.

3. Exemple : liaison pivot par roulements à billes

Considérons les réalisations technologiques des liaisons d'un objet fonctionnel particulier, celle d'un arbre supportant une roue dentée en liaison pivot avec un bâti. Le couple imprimé à la roue dentée en O est transmis par cet arbre au récepteur en C par l'intermédiaire de cannelures. La liaison pivot est assurée par des roulements à une rangées de billes. On souhaite déterminer les 2 roulements ainsi que leur position sur l'arbre de façon à minimiser la masse de l'ensemble arbre + roulements. Les modèles utilisés pour la définition des différentes relations fonctionnelles sont des modèles classiques du bureau d'études [AUB 94]. Dans ce cas de figure, le roulement côté roue dentée est modélisé par une rotule, l'autre par une liaison linéaire annulaire, l'arbre est modélisée comme une poutre avec les théories classiques de RdM.

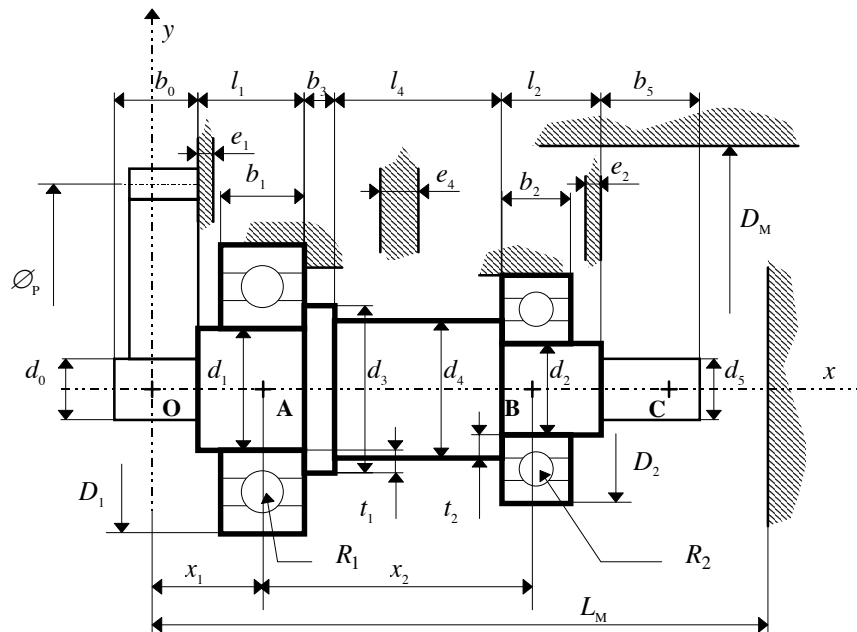


Figure1. Liaison pivot par roulements à billes .

Les données du problème, fixées par le concepteur et définies à un niveau supérieur lors de la décomposition fonctionnelle du mécanisme sont les suivantes :

b_0	30 mm	Largeur de la roue dentée.	L_M	220 mm	Encombrement maximal en longueur.
b_5	42 mm	Longueur des cannelures + joint.	ρ	7800 kg/m ³	Masse volumique de l'acier.
d_0 d_5	28 mm	Diamètre extérieur des extrémités de l'arbre.	ω	970 tr/min	Vitesse de rotation de l'arbre.
b_3	6 mm	Épaisseur de l'épaule.	C	233.31 N.m	Couple transmis.
e_1	8 mm	Distance minimale entre les dentures du pignon et le roulement R_1 .	L_V	1800 h	Durée de vie.
e_2	1 mm	Distance minimale entre R_2 et le joint d'étanchéité.	φ	20°	Angle de pression des dentures.
e_4	7 mm	Valeur minimale de l_4	Φ_p	80 mm	Diamètre primitif de la roue dentée.
D_M	100 mm	Encombrement maximal en diamètre.	K_e , K_s	0.83,0.96	Coefficient de facteur d'échelle et d'état de surface pour les calculs de fatigue.
R_m	917 MPa	Limite à la rupture du matériau de l'arbre (35CD4 traité).	δ_{max}	0.1 mm	Valeur maximale de la flèche admissible en bout d'arbre
σ_D	425 MPa	Limite de fatigue du matériau de l'arbre.	α_S α_F	1.5	Coefficient de sécurité statique et en fatigue.
R_e	845 MPa	Limite élastique du matériau de l'arbre.			

Tableau 1. Données fixées par le concepteur.

3.1. Relations fonctionnelles de définition

Il s'agit ici de lister l'ensemble des relations fonctionnelles (égalités ou inégalités) qui seront utilisées pour modéliser ce mécanisme. Il est clair que cette étape dans la formulation du problème mathématique d'optimisation fait appel à des connaissances d'experts de la part du concepteur. Le modèle formulé doit être suffisamment complet et précis afin de représenter un maximum de cas de conception.

Les efforts supportés par les roulements se déterminent simplement et permettent d'exprimer les conditions limites sur les valeurs de capacité de charge dynamique (C_1 et C_2) de chacun d'eux, paramètre de détermination des roulements :

$$C_1 \geq (60\omega L_V)^{1/3} F(1 + x_1/x_2)$$

$$C_2 \geq (60\omega L_V)^{1/3} F(x_1/x_2)$$

$$\text{avec } F = \frac{2C}{\phi_P} \sqrt{1 + \tan^2 \varphi}$$

Les conditions de résistance statique, en tenant compte des coefficients de concentration de contraintes K_{ti} en torsion associée à la section en A ($i=1$) et en B ($i=2$) donnent :

$$d_i^3 \geq \frac{16\sqrt{3}K_{ti}C\alpha_S}{\pi R_e}$$

La section en A est également soumise à de la flexion alternée. La condition de résistance en fatigue, en tenant compte des coefficients de concentration de contraintes en flexion K_{tff1} et en torsion, s'exprime :

$$d_1^3 \geq \alpha_F \left(\frac{32K_{tff1}F}{\pi K_e K_s \sigma_D} x_1 + \frac{16\sqrt{3}K_{t1}C}{\pi(2R_m - \sigma_D)} \right)$$

Les expressions des différents coefficients de concentration de contraintes sont tirés de [FAU 92] et tiennent compte des rayons de raccordement entre les sections de l'arbre. Afin de prendre en compte les déformations de l'ensemble nous avons utilisé pour déterminer les déformations de chaque roulement la relation [HER 98] :

$$\delta_r = \left(\frac{P}{21490Z\sqrt{D_w}} \right)^{2/3}$$

avec P : effort radial (N), δ , déplacement radial (mm),
 D_w diamètre des billes (mm) et Z nombre de billes

La déformation due à la flexion de l'arbre est calculée en simplifiant la forme de l'arbre (2 tronçons de diamètre d_1 et d_4 et de longueur x_1 et x_2). La flèche totale sur laquelle on impose une valeur limite, s'exprime alors de la manière suivante :

$$\delta = \left(\frac{F}{21490} \right)^{2/3} \left[\frac{(1+x_1/x_2)^{5/3}}{(Z_1 \sqrt{D_{w1}})^{2/3}} + \frac{(x_1/x_2)^{5/3}}{(Z_2 \sqrt{D_{w2}})^{2/3}} \right] + \frac{64F}{3\pi E} x_1^2 \left(\frac{x_2}{d_4^4} + \frac{x_1}{d_1^4} \right) \text{ et } \delta \leq \delta_{\max}$$

Cette simplification conduit à surestimer la flèche de l'arbre en O d'environ 40%, ce qui va évidemment dans le sens de la sécurité. Néanmoins il est tout à fait possible de prendre en compte un modèle plus précis, mais cela alourdit la formulation du problème d'optimisation.

A ces relations fonctionnelles exprimant des conditions sur les efforts, les contraintes et les déformations, il est nécessaire d'ajouter des conditions de montage [1],[2],[3] et de fonctionnement [4],[5],[6] ainsi que des conditions d'encombrement géométrique [7],[8].

$d_1 \geq d_0$: nécessaire pour le montage du roulement R_1 par la gauche. [1]

$d_2 \geq d_5$: nécessaire pour le montage du roulement R_2 par la droite. [2]

$D_1 \geq D_2$: montage de l'ensemble arbre + roulements dans le boîtier par la gauche. [3]

$l_1 \geq b_1 + e_1$: espace minimal entre le roulement R_1 et la roue dentée en bout d'arbre. [4]

$l_4 \geq e_4$: espace minimal entre les roulements R_1 et R_2 . [5]

$l_2 \geq b_2 + e_2$: espace minimal entre le roulement R_2 et l'épaulement des cannelures. [6]

$D_1 \leq D_M$: encombrement diamétral maximal. [7]

$x_1 + x_2 - b_2/2 + l_2 + b_5 \leq L_M$: encombrement maximal en longueur. [8]

D'autre part les différents diamètres et les côtes de longueur sur l'arbre sont liés par les relations :

$$d_3 = d_1 + 2t_1 \quad [9]$$

$$d_4 = d_2 + 2t_2 \quad [10]$$

$$x_1 + b_1/2 = b_0/2 + l_1 \quad [11]$$

$$x_2 = b_1/2 + b_3 + l_4 + b_2/2 \quad [12]$$

Des paramètres liés aux roulements interviennent dans les relations exprimées ci-dessus. Ces paramètres sont issus des tableaux de données des fournisseurs de roulements à billes. Dans notre cas nous avons considéré une série standard de 32 roulements. Pour chaque roulement on a :

$C_i = \phi_1(R_i)$	Capacité de charge dynamique.	$b_i = \phi_6(R_i)$	Largeur.
$D_i = \phi_2(R_i)$	Diamètre extérieur.	$r_i = \phi_7(R_i)$	Rayon minimal de raccordement.
$d_i = \phi_3(R_i)$	Diamètre intérieur.	$D_{wi} = \phi_8(R_i)$	Diamètre des billes.
$m_i = \phi_4(R_i)$	Masse.	$Z_i = \phi_9(R_i)$	Nombre de billes.
$t_i = \phi_5(R_i)$	Epaisseur minimale de l'épaulement.		

Tableau 2. Paramètres discrets des roulements à billes.

Dans le tableau 2, R_i désigne le numéro de chacun des roulements. La formulation de ce problème d'optimisation utilise donc des paramètres discrets dépendant d'une variable entière. Les fonctions ϕ_i expriment cette dépendance sans la définir explicitement puisque ces paramètres sont issus d'un tableau de valeurs.

3.2. Formulation du problème d'optimisation

La fonction objectif à minimiser, la masse de l'ensemble arbre + roulement s'écrit :

$$M = m_1 + m_2 + \rho \frac{\pi}{4} \left[d_1^2 l_1 + (d_1 + 2t_1)^2 b_3 + (d_2 + 2t_2)^2 l_4 + d_2^2 l_2 \right]$$

De manière générale l'ensemble des relations fonctionnelles écrites lors de l'étape précédente se divise en m_e équations fonctionnelles et m_i inégalités fonctionnelles faisant intervenir n_p paramètres descriptifs (non compris les paramètres considérés comme données du problème et donc fixés par le concepteur). Le « degré de liberté » du problème égal à $(n_p - m_e)$ est nécessairement positif, dans le cas contraire cela signifierait que l'on a oublié des

paramètres descriptifs ou que certaines équations fonctionnelles sont redondantes. Théoriquement ces m_e équations fonctionnelles doivent permettre d'éliminer par substitution m_e paramètres descriptifs. En pratique ces éliminations ne sont possibles que pour des expressions analytiques suffisamment simples. Si n_{pe} est le nombre de paramètres éliminés, il existe autant de formulations du problème d'optimisation que de combinaisons de $(n_p - n_{pe})$ paramètres libres pris parmi les n_p paramètres. Ces $(n_p - n_{pe})$ paramètres libres seront donc les variables du problème d'optimisation, les m_i inégalités fonctionnelles définiront le domaine des solutions réalisables auxquelles il conviendra d'ajouter les $(m_e - n_{pe})$ égalités fonctionnelles restantes. Parmi ces expressions possibles du problème d'optimisation certaines sont irréalisables parce qu'elles utiliseront comme variables des paramètres qui auraient pu être éliminés aisément ou parce que aucun des paramètres choisis n'interviendront dans l'expression de la fonction objectif.

Dans le cas de ce problème, on dénombre 25 paramètres descriptifs (18 pour les deux roulements ajoutés à : $d_3, d_4, x_1, x_2, l_1, l_2, l_4$) et 20 égalités (16 pour les deux roulements plus les égalités [9],[10],[11],[12]). Il reste donc 5 paramètres libres. En remarquant que l'inégalité $l_2 \geq b_2 + e_2$ peut se ramener à une égalité puisque la masse de l'arbre sera minimale pour $l_2 = b_2 + e_2$, cela réduit à 4 le nombre de paramètres libres. On obtient finalement la formulation finale d'un problème d'optimisation non linéaire de 4 variables comportant 11 inégalités fonctionnelles ou fonctions contraintes.

$$F(R_1, R_2, x_1, x_2) = m_1 + m_2 + \rho \frac{\pi}{4} \left[0.5 \left(d_1^2 (b_1 - b_0) - (d_2 + 2t_2)^2 (b_1 + b_2) \right) \right] \\ + \rho \frac{\pi}{4} \left[(d_1 + 2t_1)^2 b_3 + d_2^2 l_2 + x_1 d_1^2 + d_2^2 (x_2 - b_3) \right]$$

$$g_1(R_1, R_2, x_1, x_2) = 0.5(b_0 + b_1) - x_1 + e_1 \leq 0$$

$$g_2(R_1, R_2, x_1, x_2) = 0.5(b_1 + b_2) - x_2 + e_4 + b_3 \leq 0$$

$$g_3(R_1, R_2, x_1, x_2) = D_2 - D_1 \leq 0$$

$$g_4(R_1, R_2, x_1, x_2) = d_{\min 1} - d_1 \leq 0$$

$$g_5(R_1, R_2, x_1, x_2) = d_{\min 2} - d_2 \leq 0$$

$$g_6(R_1, R_2, x_1, x_2) = D_1 - D_M \leq 0$$

$$g_7(R_1, R_2, x_1, x_2) = x_2 + x_1 + 0.5b_2 - L_M + e_2 + b_5 \leq 0$$

$$g_8(R_1, R_2, x_1, x_2) = \left(\frac{32K_{\text{uff1}}F}{\pi K_e K_s \sigma_D} x_1 + \frac{16\sqrt{3}K_{\text{ul}}C}{\pi(2R_m - \sigma_D)} \right) - \frac{d_1^3}{\alpha_F} \leq 0$$

$$g_9(R_1, R_2, x_1, x_2) = (60\omega L_V)^{1/3} F(1 + x_1/x_2) - C_1 \leq 0$$

$$g_{10}(R_1, R_2, x_1, x_2) = (60\omega L_V)^{1/3} F(x_1/x_2) - C_2 \leq 0$$

$$g_{11}(R_1, R_2, x_1, x_2) = \left(\frac{F}{21490} \right)^{2/3} \left[\frac{(1 + x_1/x_2)^{5/3}}{(Z_1 \sqrt{D_{w1}})^{2/3}} + \frac{(x_1/x_2)^{5/3}}{(Z_2 \sqrt{D_{w2}})^{2/3}} \right]$$

$$+ \frac{64F}{3\pi E} x_1^2 \left(\frac{x_2}{d_4^4} + \frac{x_1}{d_1^4} \right) - \delta_{\max} \leq 0$$

$$\text{Avec : } d_{\min 1} = \text{Max} \left\{ d_0, \left(\frac{16\sqrt{3}K_{tt1}C\alpha_S}{\pi R_e} \right)^{1/3} \right\}$$

$$d_{\min 2} = \text{Max} \left\{ d_5, \left(\frac{16\sqrt{3}K_{tt2}C\alpha_S}{\pi R_e} \right)^{1/3} \right\}$$

Ce type de problème est tout à fait représentatif des difficultés inhérentes aux problèmes de conception optimale. En effet, on obtient généralement un problème non linéaire comportant des variables mixtes (continues, entières et discrètes). Des paramètres de descriptions d'éléments standards ou normalisés interviennent dans cette formulation, paramètres qui dépendent des variables discrètes. Cette particularité ne permet pas de traiter ces variables discrètes comme des variables continues dans l'algorithme d'optimisation. De plus ces problèmes de conception optimale doivent être résolus pour n'importe quelles valeurs des données, ce qui entraîne des changements de topologie dans le domaine des solutions réalisables défini par les fonctions contraintes. On peut donc avoir à traiter des cas où le domaine des solutions est vide (on considère alors que les données sont incompatibles) ou convexes pour certaines valeurs des données, ou encore non convexes pour d'autres. Il est donc nécessaire de disposer de méthodes d'optimisation particulièrement robustes pour résoudre ce type de problème.

4. Approche heuristique : les stratégies évolutionnaires

La stratégie évolutionnaire (ES) fait partie de la classe des algorithmes évolutionnaires [BÄC 97], méthodes de recherche heuristiques, basées sur les concepts de l'évolution naturelle et guidées par le modèle de Darwin: survie des meilleurs et transmission des caractéristiques des meilleurs aux populations suivantes. Ces méthodes imitent le processus de l'évolution naturelle où une population d'individus subit des transformations de croisement, de mutation et de

sélection. Chaque individu représente un point de recherche pour le problème donné. Grâce à l'opérateur de sélection, favorisant les individus de meilleure qualité à se reproduire aux générations suivantes, la population évolue vers des régions de plus en plus prometteuses de l'espace des solutions. Ces algorithmes d'ordre 0 présentent l'avantage de ne nécessiter aucune information sur les gradients des fonctions objectifs et contraintes. Ils permettent généralement de résoudre des problèmes difficiles pour lesquels les méthodes classiques d'optimisation, basées sur le calcul des gradients, ne peuvent pas être employées.

ES est une méthode récente, les premières applications datent des années 60 (voir [REC 4] et [SCH 81]). Cette méthode est généralement utilisée pour résoudre des problèmes d'optimisation numériques, en variables continues ou mixtes. Elle se présente comme une méthode intéressante pour résoudre nos problèmes de conception optimale en variables mixtes, pour lesquels le calcul des gradients de fonctions est difficile.

La structure de ES est la suivante:

```
t:=0
initialisation de la population P(t)
évaluation de P(t)
tant que le critère d'arrêt n'est pas satisfait:
    P'(t) = croisement (P(t))
    P''(t) = mutation (P'(t))
    évaluation (P''(t))
    P(t+1) = sélection (P''(t) ∪ P(t))
    t = t+1
```

4.1. Représentation:

Dans cet algorithme, $P(t)$ correspond à une population de μ individus. Dans la version standard de ES ([BÄC 96], [BÄC 93]), chaque individu (a) est représenté par un couple de vecteurs en valeurs réelles $a = (x_C, \sigma)$. Le vecteur $x_C \in \mathbb{R}^{nc}$ contient les variables continues et le vecteur $\sigma \in \mathbb{R}_+^{nc}$ regroupe les écarts types appelés paramètres de stratégie. Ces paramètres de stratégie, incorporés dans la structure de l'individu, subissent le processus d'évolution comme les variables continues x_C et ont pour objectif de contrôler l'application de l'opérateur de mutation sur les variables continues. Cette version de ES peut s'appliquer uniquement à des problèmes en variables continues. Afin de traiter des problèmes en variables mixtes, plusieurs extensions de ES ont été proposées ([HER 91], [BÄC 95]). Dans cet article, nous avons choisi la représentation proposée par Bäck et Schütz [BÄC 97] dans laquelle chaque individu est représenté par quatre vecteurs: $a = (x_C, x_D, \sigma, p)$ où $x_D \in \mathbb{R}^{nd}$ est un vecteur contenant les variables discrètes. Des paramètres de stratégie p_j ($p_j \in [0,1], j = 1, \dots, n_d$), appelés probabilités de

mutation, ont été ajoutés à chaque individu afin de contrôler l'application de l'opérateur de mutation sur les variables discrètes x_D .

4.2. Evaluation

Pour des problèmes d'optimisation sans contrainte, l'évaluation de chaque individu est effectuée en calculant la fonction objectif au point associé à chaque individu. Afin de traiter les problèmes d'optimisation avec contraintes, plusieurs méthodes ont été développées [MIC 96]. Parmi celles-ci, nous avons choisi une méthode de pénalité dynamique avec un coefficient de pénalité croissant au cours du processus d'optimisation. Les méthodes de pénalité sont généralement des méthodes efficaces nécessitant le réglage de peu de paramètres.

4.3. Croisement

L'opérateur de croisement génère une nouvelle population $P'(t)$ de λ individus ($\lambda > \mu$). Cet opérateur permet le brassage d'informations appartenant aux parents et les transmet à leurs descendants (les enfants). Plusieurs mécanismes de croisement existent (cf. [BÄC 96], [BÄC 95], [MDO 95]). Ils peuvent être utilisés sous forme locale (deux parents choisis de manière aléatoire forment un individu) ou sous forme globale (un premier parent est sélectionné de manière aléatoire et pour chaque variable, on choisit un deuxième parent différent de manière aléatoire). L'opérateur de croisement est appliqué sur les variables continues et discrètes des individus mais aussi sur les paramètres de stratégie. Des mécanismes de croisement différents peuvent être choisis pour chaque type de variable.

4.4. Mutation

La mutation est conçue pour apporter une certaine diversité dans la population. Dans le cas des variables continues, elle consiste à ajouter un vecteur aléatoire, distribué selon une loi normale au vecteur de variables continues. L'opérateur de mutation modifie les écarts types σ_i puis modifie les variables continues [BÄC 96]:

$$\begin{aligned}\sigma'_i &= \sigma_i \cdot \exp(\tau' \cdot N(0,1) + \tau \cdot N_i(0,1)) \\ x'_{Ci} &= x_{Ci} + \sigma'_i N_i(0,1) \\ \text{avec } \tau' &= (\sqrt{2n_c})^{-1} \text{ et } \tau = (\sqrt{2\sqrt{n_c}})^{-1} \\ n_c &: \text{ nombre de variables continues}\end{aligned}$$

$N(0,1)$ est une variable aléatoire distribuée selon une loi normale de moyenne 0 et d'écart moyen 1. L'indice i dans $N_i(0,1)$ indique que l'on génère une nouvelle valeur aléatoire pour chaque variable i . Dans le cas des variables discrètes, l'opérateur de mutation modifie la probabilité de mutation p_j . La nouvelle probabilité de mutation permet ensuite de décider si on modifie les variables discrètes [BÄC 95].

$$p'_i = \left(1 + \frac{1 - p_i}{p_i} \cdot \exp(-\gamma \cdot N_i(0,1)) \right)^{-1}$$

$$x'_{Di} = \begin{cases} x_{Di}, u_i > p'_i \\ x_i, u_i < p'_i \end{cases} \quad u_i \in [0,1], \text{ nombre aléatoire}$$

$$\text{avec } \gamma = \left(\sqrt{2\sqrt{n_d}} \right)^{-1}$$

n_d : nombre de variables discrètes

x_i : nouvelle valeur discrète

L'évolution des paramètres de stratégie au cours du processus d'optimisation par l'intermédiaire du croisement et de la mutation est appelé mécanisme d'auto-adaptation. Il permet à l'algorithme d'améliorer ses propres paramètres de stratégie durant la recherche.

4.5 Sélection

ES utilise un schéma de sélection déterministe permettant de diriger la recherche vers les régions les plus intéressantes. Les meilleurs μ individus parmi les λ enfants (stratégie (μ, λ)) ou parmi l'union des μ parents et λ enfants (stratégie $(\mu+\lambda)$) sont sélectionnés comme nouveaux parents.

5. Algorithme adapté aux problèmes de conception optimale

Lors de la présentation de ES, nous avons pu voir qu'il existait deux opérateurs de sélection différents (stratégies (μ, λ) et $(\mu+\lambda)$) et plusieurs opérateurs de croisement possibles. Dans le cadre des problèmes de conception optimale, aucuns travaux n'existent concernant le choix de ces opérateurs. Nous avons donc testé plusieurs opérateurs (2 opérateurs de sélection et 48 combinaisons d'opérateurs de croisement) sur différents problèmes de conception optimale afin de connaître la meilleure combinaison d'opérateurs pour cette classe de problèmes.

Les résultats de cette étude, non présentée dans cet article ont montré que la sélection $(\mu+\lambda)$ était plus intéressante du point de vue fiabilité et rapidité que la sélection (μ, λ) .

Concernant le croisement, les meilleurs résultats ont été obtenus avec un croisement discret sous forme globale (x_i^S ou x_i^{Ti}) pour les variables discrètes x_D et un croisement intermédiaire sous forme globale [$x_i^S + 0.5(x_i^{Ti} - x_i^S)$] pour les paramètres de stratégie p associés aux variables discrètes. Dans les notations utilisées ci-dessus, S et T indiquent deux parents choisis de manière aléatoire. $u \in [0, 1]$ est une variable aléatoire choisie selon une loi uniforme. L'indice i dans T_i ou u_i indique que T ou u sont sélectionnés à nouveau pour chaque valeur de i .

Pour les variables continues, x_C , et les paramètres de stratégie associés à ces variables, σ , c'est l'opérateur de croisement intermédiaire généralisé [$x_i^S + u_i(x_i^{Ti} - x_i^S)$] avec $u_i \in [-0.5, 1.5]$ qui a été le plus performant.

Afin de résoudre ce problème de conception optimale, nous avons utilisé avec ES une population de 7 parents et 50 enfants. Notre algorithme est basé sur la sélection (7+50), les opérateurs de croisements optimaux pour la classe de problèmes de conception optimale et l'opérateur de mutation décrit dans la section précédente. ES reposant sur des règles probabilistes, nous réalisons 15 exécutions pour ce problème afin d'obtenir des résultats significatifs. Un nombre fixe de 50 générations est ici effectué pour chaque exécution.

6. Applications et résultats

L'algorithme présenté ci-dessus a été appliqué au problème de liaison pivot par roulements à billes. Afin de résoudre ce problème avec ES, nous avons numéroté les 32 roulements à billes susceptibles d'être choisis, de 1 à 32, dans le même ordre que celui du tableau normalisé où sont sélectionnés ces roulements. Les variables de notre problème d'optimisation avec ES sont : x_1, x_2 , deux variables continues et R_1, R_2 deux variables entières, considérées comme discrètes. Les paramètres discrets ($C_i, D_i, d_i, m_i, t_i, b_i, r_i, D_{wi}, Z_i$) dépendent directement du choix des 2 roulements et leurs valeurs sont prises dans les tableaux normalisés lors de l'évaluation des individus.

Comme nous l'avons vu dans la présentation de ES, chaque individu est représenté par 4 vecteurs: $a = (x_C, x_D, \sigma, p)$. Ici, pour notre problème: $x_C = (x_1, x_2)$, $x_D = (R_1, R_2)$, $\sigma = (\sigma_1, \sigma_2)$ et $p = (p_1, p_2)$. A chaque variable continue ou discrète (x_1, x_2, R_1, R_2) correspond respectivement un paramètre de stratégie ($\sigma_1, \sigma_2, p_1, p_2$). Dans le cas des variables discrètes, nous avons utilisé $p_1 = p_2$ comme le conseillent Bäck et Schütz [BÄC 95].

Nous avons résolu le problème de liaison pivot par roulements à billes pour différents jeux de données. En utilisant les données par défaut, présentées dans le tableau 1, nous obtenons la solution suivante:

$$(R_1=14, R_2=5, x_1=35.5, x_2=68.84) \\ f^* = 728213.81$$

Cette solution correspond au meilleur individu de la population finale de la meilleure exécution parmi les 15 exécutions effectuées. Afin de savoir si ce point correspondait à l'optimum global du problème, nous avons utilisé une méthode énumérative de tous les points solutions possibles.

Nous avons pu vérifier que ES nous avait bien fourni l'optimum global du problème. En prenant les variables discrètes R_1 et R_2 fixées, égales à celles du meilleur individu c.a.d. 14 et 5, on obtient la configuration du domaine des solutions présentée figure 2, à la fin de la première exécution. Il faut noter que tous les parents et les enfants sont regroupés sur un même point, correspondant au meilleur individu de coordonnées (35.5, 68.84). Nous pouvons observer que les contraintes actives à l'optimum sont les contraintes g_1, g_{11} et que toutes les fonctions contraintes ne participent pas à la définition du domaine des solutions (domaine hachuré sur la figure 2). Ce qui montre clairement que la topologie du domaine dépend des données fixées par le concepteur.

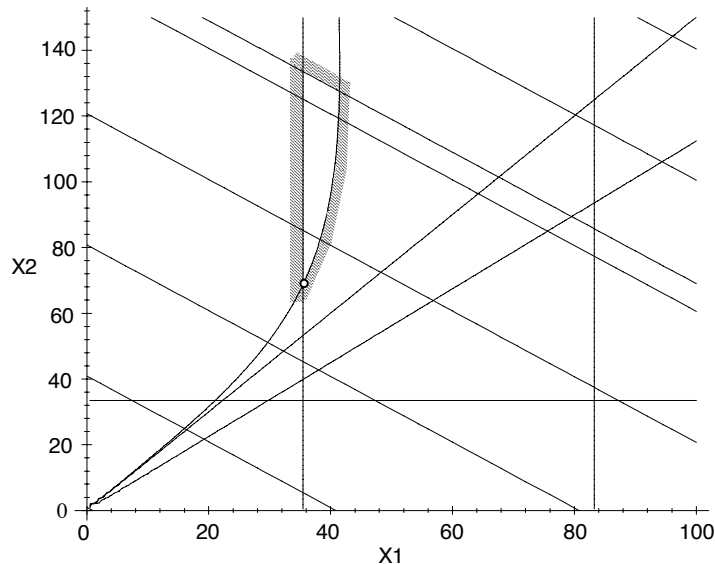


Figure 2. *Domaine des solutions avec les données par défaut.*

Le tableau 3 présente les solutions optimales obtenues avec ES pour différents jeux de données. Nous avons pu vérifier avec la méthode d'énumération que ces différentes solutions correspondaient bien à l'optimum global du problème pour chaque jeu de données. L'erreur moyenne, donnée dans le tableau correspond à la moyenne de l'erreur finale des 15 exécutions par rapport à la solution de la méthode énumérative. Ces calculs ont nécessité 37605 évaluations des fonctions objectifs et contraintes.

	(x_1, x_2, R_1, R_2)	f^*	Erreur moyenne (%)
Données par défaut	(14, 5, 35.5, 68.84)	728213.81	2.35
$\delta_{\max} = 0.4 \text{ mm}$	(14, 6, 35.5, 39.89)	524261.28	6.31
$\delta_{\max} = 0.07 \text{ mm}$	(14, 24, 35.5, 106.37)	1672142.12	3.59
$L_v = 1000 \text{ h}$	(14, 3, 35.5, 76.24)	705803.40	3.6
$\omega = 800 \text{ tr/min}$	(14, 7, 35.5, 82.83)	917014.96	3.47
$C = 49.22 \text{ N.m}$	(4, 2, 29.5, 25.55)	269777.08	2.35

Tableau 3. Résultats pour différents jeux de données.

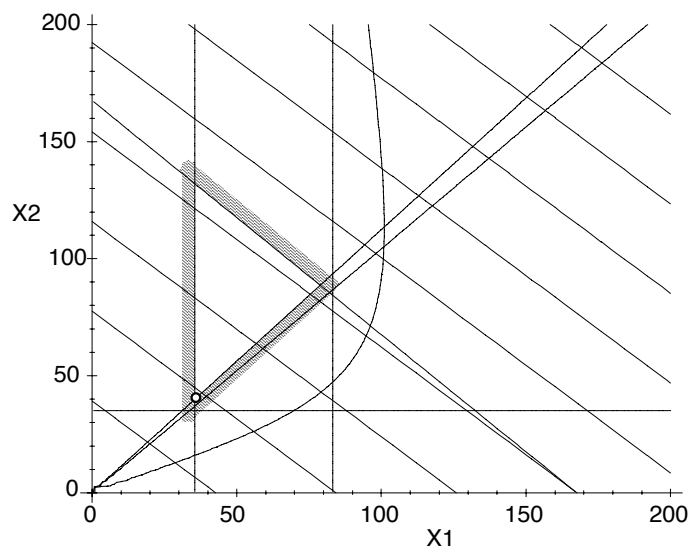


Figure 3. Domaine des solutions avec $\delta_{\max} = 0.4$.

On peut remarquer que lorsque l'on modifie les données du problème, les roulements R_1 et R_2 de la solution optimale ne sont plus les mêmes et la configuration du domaine de solution varie (figure 3).

Prenons l'exemple de la flèche maximum au point O, δ_{\max} . En choisissant au départ $\delta_{\max} = 0.4$ au lieu de 0.1, la solution du problème est alors: ($R_1 = 14$, $R_2 = 6$, $x_1 = 35.5$, $x_2 = 39.89$) avec $f^* = 524261.28$. Les contraintes actives à l'optimum ne sont plus g_1 et g_{11} mais g_1 et g_9 . La figure 3 présente la configuration du nouveau domaine des solutions à la fin de l'exécution avec ce nouveau jeu de données, les variables R_1 et R_2 étant égales à celles du meilleur individu (14, 6).

Il est intéressant de comparer le coût de ces calculs au coût d'une méthode énumérative qui consisterait à résoudre pour tous les couples de roulements ($32 \times 32 = 1024$ couples) le problème d'optimisation non linéaire comportant 2 variables continues et 7 fonctions contraintes. Ce problème s'obtient à partir du précédent en considérant tous les paramètres des roulements comme données. Pour être compétitive par rapport à notre méthode de stratégies évolutionnaires il faudrait résoudre ce problème de 2 variables et 7 fonctions contraintes en moins de 40 évaluations de fonctions, performance très difficile à atteindre.

7. Conclusions

Le problème du calcul de dimensionnement d'objets fonctionnels se formule comme un problème d'optimisation. Nous avons exposé une démarche de formulation de ce problème d'optimisation sur un exemple. Un algorithme de résolution utilisant les stratégies évolutionnaires a été proposé. Les opérateurs de croisement, de mutation et de sélection de cet algorithme ont été choisis de manière à obtenir le meilleur compromis performance / précision. Cet algorithme testé sur d'autres problèmes a également donné de bons résultats. On peut néanmoins améliorer ces résultats en traitant les variables continues avec des méthodes déterministes. Cela nécessite de coupler au niveau de l'itération de base d'une stratégie évolutionnaire (croisement, mutation, sélection) pour les variables discrètes et un algorithme de type lagrangien augmenté pour les variables continues.

Le concepteur dispose de modules de dimensionnement optimaux au sens d'un critère préétabli. Ces éléments ne constituent que des briques élémentaires d'un système plus complet de conception assistée par ordinateur intégrant des outils de calcul et de dimensionnement.

Bibliographie

- [AUB 92] AUBLIN M., *Systèmes mécaniques : théorie et dimensionnement*, Edition Dunod, Paris 1992
- [BÄC 93] BÄCK T., SCHWEFEL H.P., « An overview of evolutionary algorithms for parameter optimization », *Evolutionary computation*, vol. 1, pp. 1-23, 1993.
- [BÄC 95] BÄCK T., SCHÜTZ M., « Evolution strategies for mixed-integer optimization of optical multilayer systems », *Proceedings of the Fourth Annual Conference on Evolutionary Programming*, J.R. McDonnell, R.G. Reynolds, and D.B. Fogel, editors, MIT Press, Cambridge, MA, pp 33-51, 1995.
- [BÄC 96] BÄCK T., *Evolutionary algorithms in theory and practice*, Oxford University Press, New York, 1996.

- [BÄC 97] BÄCK T., HAMMEL U., SCHWEFEL H.P., « Evolutionary computation: comments on the history and current state », *IEEE Transactions on evolutionary computation*, vol. 1, pp. 1-15, 1997.
- [DEN 98] DENEUX D., « Conception fonctionnelle intégrant le calcul mécanique : le prototype F-CAD », *Revue de CFAO et d'informatique graphique*, Vol. 13, n°2, p 191-204.
- [FAU 92] FAURIE J. P., MONNIER P., NIKU-LARI A., *Guide du dessinateur : les contraintes*, Edition CETIM, 1992
- [GUI 87] GUILLOT J., « Méthodologie de définition des ensembles mécaniques en conception assistée par ordinateur , recherche des solutions optimales », Thèse d'état de l'Université Paul Sabatier Toulouse III, N° d'ordre 1343, 30 sep 1987
- [HER 91] HERDY M., « Application of the evolution strategy to discrete optimization problems », *Proceedings of the first international conference on parallel problem solving from nature (PPSN)*, vol. 496, Lecture note in Computer Science, Springer-Verlag, 1991.
- [HER 95] HERNOT X., DAIDIE A., SILBERBERG Y., GUILLOT J. « Conception intégrées de mécanismes », *Revue internationale de CFAO et d'informatique graphique*, Vol. 10, n°1-2, p57-70.
- [HER 98] HERNOT X., « Modélisation des arbres et des paliers à roulements en de l'optimisation de l'ensemble arbre-paliers », Thèse de doctorat de l'Institut National des Sciences Appliquées de Toulouse, N° d'ordre 465, 5 fév 1998.
- [LAF 94] LAFON P., « Conception optimale de systèmes mécaniques : optimisation en variables mixtes », Thèse de doctorat de l'Institut National des Sciences Appliquées de Toulouse, N° d'ordre 273, 15 fév 1994.
- [MDO 95] MAC DONNELL J.R., WAAGEN D.E., « An empirical study of recombination in evolutionary search », *Proceedings of the Fourth Annual Conference on Evolutionary programming*, J.R. Mac Donnel, R.G. Reynolds and D.B. Fogel (Eds), pp. 465-478, 1995.
- [MIC 96] MICHALEWICZ Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, Berlin, 1996.
- [REC 94] RECHENBERG I., « Evolutionstrategie: optimierung technischer System nach prinzipien der biologischen Evolution », Frommann-Holzboog, Stuttgart, 1994.
- [SCH 81] SCHWEFEL H.P., *Numerical optimization of computer models*, Chichester: Wiley, 1981.
- [TOL 98] TOLLENAERE M., *Conception de produits mécaniques, méthodes, modèles et outils*, Edition Hermès, Paris 1998.