



HAL
open science

Méthode de pré-conception robuste basée sur une optimisation multi-objectifs contrainte: Application au dimensionnement de suspensions de véhicule

David Mosnier, Frédéric Gillot, Mohamed Ichchou

► To cite this version:

David Mosnier, Frédéric Gillot, Mohamed Ichchou. Méthode de pré-conception robuste basée sur une optimisation multi-objectifs contrainte: Application au dimensionnement de suspensions de véhicule. CFM 2011 - 20ème Congrès Français de Mécanique, Aug 2011, Besançon, France. hal-03421325

HAL Id: hal-03421325

<https://hal.science/hal-03421325>

Submitted on 9 Nov 2021

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.

Méthode de pré-conception basée sur une optimisation multi-objectifs contrainte : Application au dimensionnement de suspensions de véhicule

D. Mosnier^{a,b}, F. Gillot^a, M. Ichchou^a

a. LTDS, Ecole Centrale de Lyon, 36 avenue Guy de Collongues, 69130 ECULLY

b. MFP Michelin, Centre de Technologies, ZI Ladoux, 63000 CLERMONT FD

La méthode proposée est ici appliquée directement sur un modèle classique de demi-véhicule sur lequel on cherche à optimiser un ensemble d'objectifs tels que le confort, la masse et la tenue de route. Les résultats de l'optimisation sont donnés sous la forme d'un nombre limité de typages, ce qui constitue un outil puissant d'aide à la décision pour le concepteur. Une seconde boucle d'optimisation contrainte permet de raffiner par la suite un typage choisi.

Mots clefs : Optimisation multiobjectifs contrainte ; Cartes auto-adaptatives

1 Introduction

Un problème de pré-conception de pièces mécaniques peut généralement être traduit sous la forme d'un problème d'optimisation multi-objectifs où chaque performance est représentée par une fonction objectif à optimiser. Parmi les différentes techniques existantes pour résoudre ce type de problème, les algorithmes évolutionnaires sont parmi les plus populaires car ils permettent d'obtenir un nombre important de solutions compromises (front de Pareto). L'autre avantage majeur de ces techniques réside dans le fait qu'il n'est pas nécessaire que le concepteur spécifie *a priori* l'importance à donner à chacun des objectifs.

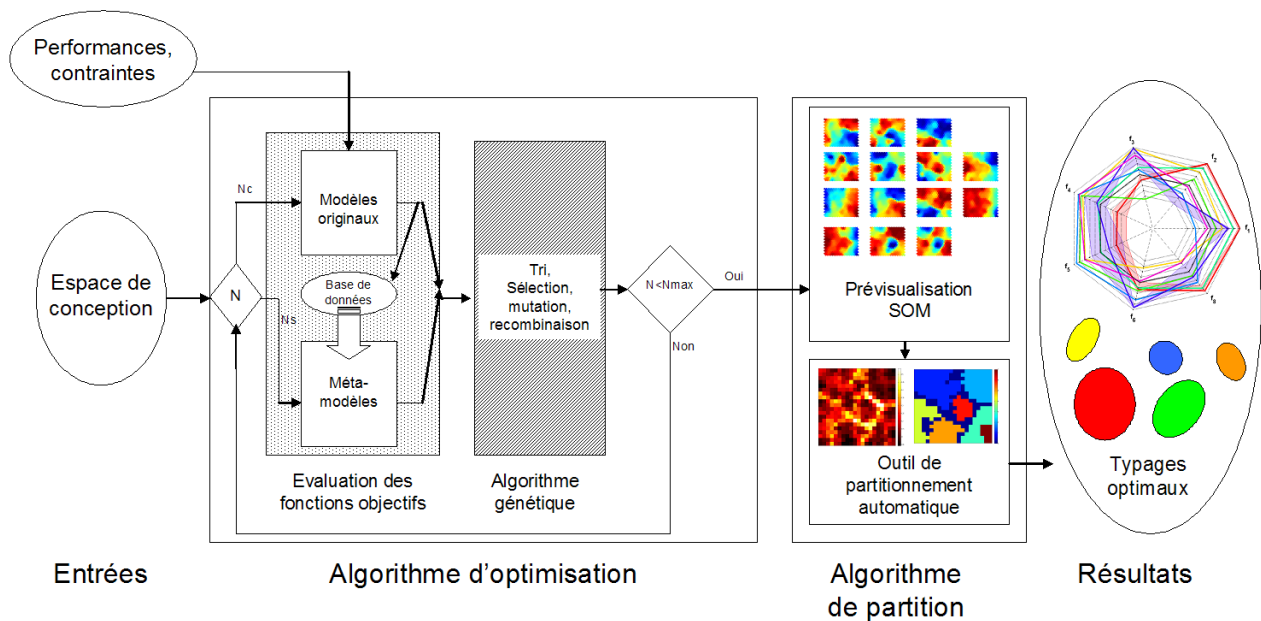


FIGURE 1 – Schéma global de la méthode proposée.

Dans cet article nous proposons une méthode de pré-conception intégrée dont le schéma de fonctionnement est présenté sur la figure 1. Nous présentons dans un premier temps le problème de pré-conception qui sera utilisé pour illustrer chaque étape du processus. La boucle d'optimisation s'appuyant sur un algorithme génétique est décrite dans la seconde partie. Une accélération des évaluations des différentes fonctions objectifs est proposée grâce à l'implémentation de métamodèles au sein du processus d'optimisation. Une troisième partie présente la méthode de partitionnement des solutions optimales obtenues en un nombre limité de typages. Enfin, une fois un typage sélectionné, nous présentons une méthode d'affinage de l'optimisation par rajout de contraintes avant de conclure.

2 Présentation du système

Nous proposons dans cette partie de décliner une démarche de pré-conception des différentes grandeurs qui composent la suspension d'un véhicule ainsi que sa monte pneumatique. Nous nous appuyons pour cela sur un schéma représentatif du système illustré sur la figure 2. L'ensemble des variables et paramètres du système sont donnés dans le tableau 1.

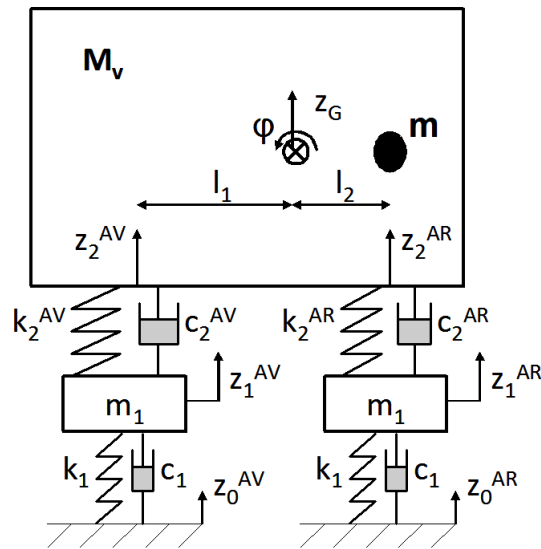


FIGURE 2 – Schéma du demi-véhicule simplifié.

Trouver : $c_1, c_2^{AV}, c_2^{AR}, k_1, k_2^{AV}, k_2^{AR}, m_1$

Minimisant : $f_1 = \text{RMS}(\ddot{z}_2^{AV})$
 $f_2 = \text{RMS}(F_1^{AV})$
 $f_3 = m_1$
 $f_4 = \max_{\omega \in [\omega_1, \omega_2]} (z_2^{AV} - z_0^{AV})$
 $f_5 = \max_{\omega \in [\omega_1, \omega_2]} (z_2^{AV} - z_1^{AV})$
 $f_6 = \max_{\omega \in [\omega_1, \omega_2]} (z_2^{AR*} - z_0^{AR})$
 $f_7 = \max_{\omega \in [\omega_1, \omega_2]} (z_2^{AR*} - z_1^{AR})$ (1)

Où $\text{RMS}(g) = \frac{1}{\omega_2 - \omega_1} \sqrt{\int_{\omega_1}^{\omega_2} g^2(\omega) d\omega}$
 et $F_1^{AV} = k_1(z_1^{AV} - z_0^{AV}) + c_1(\dot{z}_1^{AV} - \dot{z}_0^{AV})$
 et $\omega_1 = 0.06 \text{rad/s}, \omega_2 = 125 \text{rad/s}$

La fonction f_1 correspond à une moyenne fréquentielle d'accélération au niveau de la caisse sur l'essieu avant, ce qui correspond à critère d'inconfort à minimiser. f_2 désigne la moyenne fréquentielle des efforts vus par les pneumatiques sur l'essieu avant. La minimisation de ce critère permet d'augmenter l'endurance du pneumatique sur le véhicule. Sur ces deux critères, la moyenne fréquentielle est calculée à partir de l'opérateur de moyenne géométrique RMS sur une bande de fréquence comme défini dans l'équation (1).

Paramètre	Description	borne inf	borne sup	Paramètre	Description
c_1	Amortissement des pneus	10 Ns/m	120 Ns/m	z_2^{AV}	Déplacement z suspension AV
c_2^{AV}	Amortissement suspension AV ¹	1 kNs/m	8kNs/m	z_2^{AR}	Déplacement z suspension AR
c_2^{AR}	Amortissement suspension AR ¹	1 kNs/m	8kNs/m	z_0^{AV}	Déplacement z sol AV
k_1	Raideur des pneus	120 kN/m	580kN/m	z_0^{AR}	Déplacement z sol AR
k_2^{AV}	Raideur suspension AV	10 kN/m	80kN/m	z_1^{AV}	Déplacement z pneu AV
k_2^{AR}	Raideur suspension AR	10 kN/m	80kN/m	z_1^{AR}	Déplacement z pneu AR
m_1	Masse non suspendue	25 kg	55kg	z_G	Déplacement z CG ³
M_v	Masse suspendue	1000 kg		x^*	Paramètre x calculé avec m
m	Masse additionnelle	200 kg		φ	angle de tangage
l_1	Longueur entre l'AV et le CG du véhicule	1 m			
l_2	Longueur entre l'AR et le CG du véhicule	1.4 m			

TABLE 1 – Table de synthèse des paramètres et variables du problème de pré-conception multi-objectifs. Nous redonnons également dans cette table la plage de variation des paramètres de conception. Ces derniers sont laissés indépendants dans le processus d'optimisation.

La fonction f_3 désigne la masse non suspendue. La minimisation de ce critère va nous permettre de réduire la consommation de carburant du véhicule par des effets de réduction d'inertie des ensembles tournant notamment. La minimisation de la fonction f_4 permet de conserver une altitude de l'essieu avant stable par rapport au sol. Ce critère est donc à relier avec une performance de tenue de route ; on s'assure grâce à lui que le pneumatique reste le plus possible en contact avec le sol. La fonction f_5 correspond à la course maximale des suspensions. La minimisation de ce critère va apporter une réduction de l'encombrement de la suspension sur le véhicule. Enfin, f_6 et f_7 sont des fonctions objectifs construites de manière symétrique à f_4 et f_5 pour l'essieu arrière. Pour le calcul de ces critères, on vient en plus modifier la condition de charge du véhicule en plaçant une masse ponctuelle supplémentaire m dans le coffre du véhicule.

3 Algorithme d'optimisation

Nous nous appuyons sur un algorithme génétique de type NSGA-II [1] pour réaliser la boucle d'optimisation. Ce type d'algorithme est facile à implémenter et ne nécessite pas de formuler de quelconque hypothèse sur la forme des fonctions à traiter, ni sur une éventuelle corrélation entre elles, et est donc robuste quel que soit le problème posé. De plus, il n'est pas nécessaire de hiérarchiser les performances *a priori*. D'autre part, les modèles utilisés pour évaluer les différentes fonctions objectifs présentent souvent une complexité rédhibitoire pour qu'ils soient implémentés dans un algorithme d'optimisation. Le recours à des méthodes de métamodélisation tel que le Krigeage permet de surmonter cette difficulté [3]. En effet, ce dernier est capable de représenter des modèles ayant des non linéarités en un temps très court et sa construction ne nécessite qu'un nombre fini de points. Nous proposons ainsi de faire appel tantôt aux modèles approchés, tantôt aux modèles originaux afin d'accélérer le temps de calcul tout en gardant une précision de modélisation correcte. Nous avons appliqué cette démarche d'optimisation sur le problème de pré-conception décrit dans la partie 2 et nous avons obtenu un ensemble de cinq cent solutions optimales, au sens de Pareto. Néanmoins, étant donné que chacune des solutions se composent de sept paramètres de conception et d'autant d'objectifs, il est très difficile pour le concepteur de les visualiser et donc d'interpréter les résultats pour venir choisir la solution finale.

4 Algorithme de partition automatique

Un outil d'aide à la décision devient nécessaire afin que le concepteur puisse visualiser et trier les solutions obtenues en sortie d'optimisation. Les cartes auto-adaptatives (Self Organized Map, SOM [4]) peuvent être utilisées comme un outil de visualisation des solutions, après une phase de normalisa-

1. AV se rapporte à l'essieu avant, AR se rapporte à l'essieu arrière
2. CG signifie Centre de Gravité

tion des différentes composantes afin de les rendre comparables. Ces cartes permettent de mettre en évidence un certain nombre de relations entre les différentes composantes et donc entre les différents compromis existants au sein de la population obtenue. Si d'éventuelles corrélations existent entre les variables de conception ou entre les objectifs sur l'ensemble des solutions optimales proposées, l'emploi de SOM permettra de les mettre en évidence mais l'étude de ces corrélations n'a pas été abordée dans ce travail en tant que tel. Néanmoins, les cartes SOM ne permettent pas d'effectuer un tri de cette population.

Nous avons développé un algorithme automatique qui tire parti de la phase d'apprentissage des SOM pour déterminer des groupes de solutions équivalentes, c'est à dire qui offrent des performances proches. Nous partons du constat qu'une partie du travail de regroupement de solutions similaires a été effectuée au cours de la phase d'apprentissage de la construction des cartes auto-adaptatives où des vecteurs similaires se retrouvent dans une même zone de la carte. On considère alors la matrice des distances (D-matrice) représentant sur une grille la distance d'un individu à son voisin sur chaque vecteur. La recherche de groupe d'individus ayant des performances proches revient alors à rechercher les zones iso-chromes sombres sur la représentation 3. Les zones plus claires forment alors des frontières entre les différents groupes. Nous proposons ainsi un algorithme de reconnaissance d'image qui permet de déterminer des groupes à partir de la visualisation des distances projetées sur la grille d'une carte auto-adaptative. Le résultat de l'algorithme de partition des vecteurs prototypes est représenté sur la grille de droite sur la figure 3.

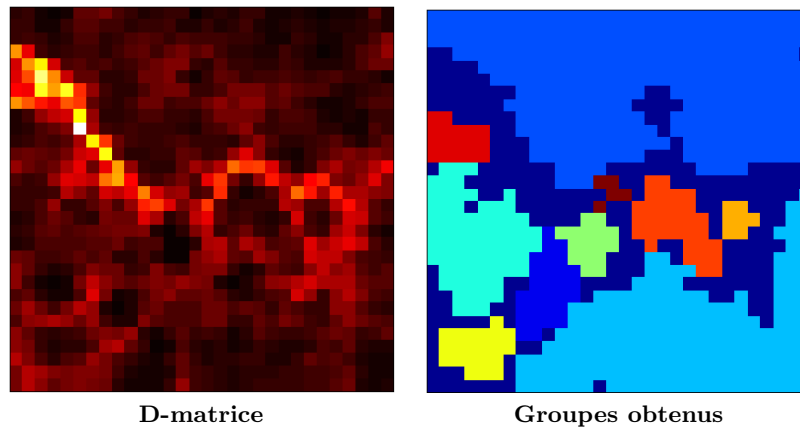


FIGURE 3 – La figure de gauche correspond à la projection des distances de chaque individu à ses voisins sur une grille de carte auto adaptative. La figure de droite représente le résultat de l'algorithme de partitionnement sur l'image de gauche. Les zones de couleurs correspondent à des groupes de solutions équivalentes quand le bleu marine correspond à des zones "frontières" dans lesquelles les individus sont très dissemblables des individus avoisinants

On peut associer chaque individu de la population à un groupe en recherchant le vecteur prototype de la carte qui le représente le mieux. On associera ensuite l'individu au groupe auquel appartient le vecteur prototype correspondant. Ceci permet de construire une partition des solutions en différentes classes d'équivalence et ce, sans spécifier *a priori* le nombre de groupes à déterminer. De plus, l'autre avantage important de cet algorithme tient dans le fait que son résultat ne dépend pas de son initialisation et ne nécessite aucune phase de convergence. La partition optimale ainsi obtenue, nous disposons de groupes de solutions similaires. A partir de là, il est plus facile de représenter les différents groupes de solutions sur des supports de visualisation plus conventionnels, à partir de représentants de ces groupes. La figure 4 représente ainsi chaque groupe par une couleur et par 3 représentants, ce qui permet de faire une synthèse simple de chaque groupe. Ces 3 individus sont l'individu médian et les individus extrêmes du groupe respectivement représentés par un trait en gras et 2 lignes délimitant une surface colorée. C'est un moyen de donner une représentation de notions de moyenne et d'écart type adaptées aux typages. Ces individus sont mathématiquement définis par (3) et (4) :

$$\text{Centre du groupe } C : \quad f^* = \left[\frac{1}{N_C} \sum_{x \in C} f_1, \dots, \frac{1}{N_C} \sum_{x \in C} f_M \right] \quad (2)$$

$$\text{Individu médian : } x^m = x \in C \quad / d(x^m, f^*) = \min_{x \in C} d(x, f^*) \quad (3)$$

$$\text{Individus extrêmes : } x^A, x^B \in C \quad / d(x^A, x^B) = \max_{x, y \in C} d(x, y) \quad (4)$$

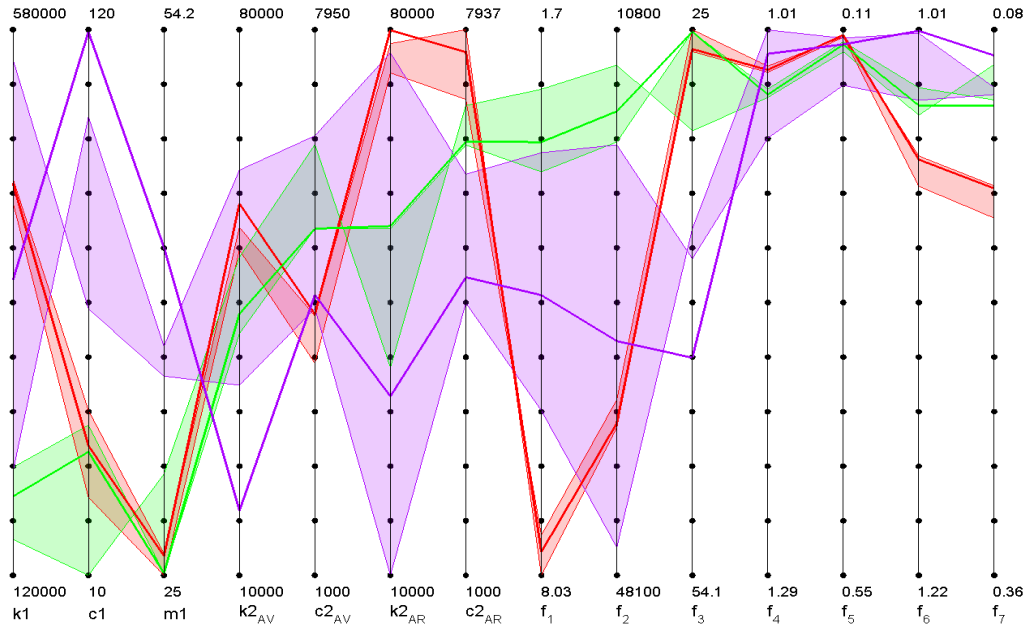


FIGURE 4 – Diagramme parallèle des principaux typages. Pour des questions de facilité de lecture, seuls trois typages sont affichés avec leurs trois représentants

5 Raffinement de l'optimisation par contraintes

A ce stade de la démarche de conception, le concepteur a à sa disposition un ensemble de solutions optimales réorganisées en typages. Néanmoins, le choix de la solution finale reste encore à sa charge à partir des solutions proposées. Dans le cas où le nombre d'objectifs est important, le front de Pareto idéal est très vaste et ne saurait être pleinement décrit avec un nombre discret de points. Nous proposons ici de définir un jeu de contraintes permettant de ne conserver que les solutions dans une zone de l'espace des performances qui intéresse le concepteur [2]. L'utilisation des bornes maximales (dans le cas où le problème est un problème de minimisation) des typages obtenus lors d'une première optimisation sans contraintes peut constituer une base pertinente pour la définition de ces contraintes. Les contraintes correspondants au trois typages représentés précédemment sont données dans le tableau 2. Cette démarche revient à approfondir la recherche de solutions optimales sur chacun des typages.

Jeu de contraintes	$\eta_{f_1}^{\max}$	$\eta_{f_2}^{\max}$	$\eta_{f_3}^{\max}$	$\eta_{f_4}^{\max}$	$\eta_{f_5}^{\max}$	$\eta_{f_6}^{\max}$	$\eta_{f_7}^{\max}$
C_1	0.00	0.27	0.96	0.92	0.98	0.71	0.65
C_2	0.74	0.79	0.81	0.87	0.96	0.84	0.86
C_3	0.21	0.00	0.00	0.80	0.85	0.81	0.73

TABLE 2 – Tableau des différents seuils servant à définir les contraintes sur chacun des objectifs pour trois cas de calculs différents.

Les résultats de calcul d'optimisation, pour chacun de ces trois cas, sont donnés dans le tableaux 3. Dans ce tableau, nous redonnons pour chaque cas de calcul une synthèse de la population avec les individus les plus dissemblables obtenus ainsi que l'individu médian. Nous redonnons aussi les valeurs extrêmes obtenues sur chaque composante de la population optimale, pour chaque calcul.

		k_1	k_2^{AV}	k_2^{AR}	c_1	c_2^{AV}	c_2^{AR}	m_1	f_1	f_2	f_3	f_4	f_5	f_6	f_7
C_1	Extrême 1	241.60	29.83	45.57	0.06	2.90	4.64	25.05	0.67	0.73	1.00	0.94	0.99	0.72	0.66
	Extrême 2	389.14	47.67	10.04	0.12	6.64	7.99	26.11	0.32	0.39	0.96	0.94	1.00	0.94	0.90
	Médian	354.81	48.76	39.37	0.09	5.94	7.97	25.23	0.36	0.47	0.99	0.93	1.00	0.92	0.88
	min	154.85	13.34	10.00	0.01	2.68	3.73	26.20	0.15	0.27	0.96	0.92	0.98	0.72	0.66
	max	455.43	65.43	69.21	0.12	8.00	8.00	25.00	0.89	0.92	1.00	1.00	1.00	1.00	0.98
C_2	Extrême 1	207.21	11.58	11.99	0.09	2.72	2.98	28.36	0.84	0.80	0.88	0.99	0.99	1.00	0.93
	Extrême 2	202.00	66.91	42.80	0.05	8.00	8.00	25.01	0.77	0.81	1.00	0.87	0.96	0.86	0.96
	Médian	176.08	30.93	27.76	0.07	5.35	7.12	28.68	0.80	0.87	0.87	0.89	1.00	0.89	0.88
	min	141.38	10.23	10.00	0.01	2.34	2.83	30.41	0.74	0.80	0.81	0.87	0.96	0.84	0.86
	max	211.19	68.10	53.84	0.12	8.00	8.00	25.00	0.93	0.95	1.00	0.99	1.00	1.00	1.00
C_3	Extrême 1	135.25	13.02	32.50	0.01	6.06	5.70	53.97	0.95	0.97	0.01	0.81	0.93	0.92	0.99
	Extrême 2	523.55	62.45	66.74	0.08	4.66	5.59	25.24	0.24	0.12	0.99	0.97	0.94	0.89	0.81
	Médian	391.35	53.45	24.17	0.05	6.39	7.46	37.72	0.35	0.39	0.56	0.94	1.00	0.93	0.90
	min	120.20	10.00	10.00	0.01	2.57	2.42	54.19	0.21	0.00	0.00	0.80	0.85	0.81	0.73
	max	580.00	79.47	80.00	0.12	8.00	8.00	25.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00

TABLE 3 – Tableau synthétisant les paramètres de conception et les performances des résultats obtenus pour les trois cas de calculs avec contraintes précédemment décrits. Dans ce tableau, l'ensemble des objectifs a été normalisé entre 0 et 1 afin de pouvoir comparer les différentes composantes entre elles dans le processus de partitionnement.

On observe que les différents typages sont alors mieux décrits car l'ensemble des solutions optimales proposées sont bornées par les contraintes et décrivent une partie de l'espace des performances beaucoup plus précise. Il est possible de faire là encore appel à un algorithme de partitionnement dans le cas où la diversité de solutions proposées par l'optimisation contrainte est encore trop élevée pour être appréhendée facilement. On peut alors itérer et relancer des optimisations avec un jeu de contraintes plus sévères dictées par les valeurs extrêmes des nouveaux typages obtenus.

Nous remarquons par ailleurs que la diversité de solutions capables de répondre à un jeu de contraintes strict, comme celui correspondant au calcul C_2 , est moins élevée. Il n'y a donc pas un nombre infini de solutions capables de répondre à un cahier des charges précis. Toutefois, la difficulté consiste à définir les bonnes contraintes permettant de réduire suffisamment la diversité mais sans devenir trop sévère au risque de ne plus avoir de solutions viables.

6 Conclusion

Nous avons présenté, dans cet article, une méthode de pré-conception intégrée capable de délivrer au concepteur un nombre fini de typages de solutions optimales. Le choix d'un de ces typages permet de définir un jeu de contraintes pertinent afin de procéder à une optimisation plus fine sur une zone de l'espace des objectifs identifiée comme intéressante. De plus, l'implémentation de métamodèles au sein du processus d'optimisation permet de surmonter les difficultés engendrées par l'utilisation de modèles complexes. Cette méthode a été appliquée avec succès à un problème de préconception des caractéristiques des pneumatiques et des suspensions d'un véhicule comportant des objectifs variés tels que le confort, l'endurance, la réduction de masse ou encore la tenue de route.

Références

- [1] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. 2000 A fast and elitist multi-objective genetic algorithm : NSGA-II *Kanpur Genetic Algorithms Laboratory*
- [2] Deb, K. 2000 An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering* **186(2-4)** 311-338
- [3] Simpson, T., Peplinski, J., Koch, P., Allen, J. 2001 Metamodels for computer-based engineering design : Survey and recommendations. *Engineering with computers* **17** 129-150
- [4] Vesanto, J., Himberg, J., Alhoniemi, E., Parhankangas, J. 2000 Som toolbox for matlab 5. *Helsinki University of Technology Report A57*