



HAL
open science

CONTRIBUTION METHODOLOGIQUE A LA MODELISATION POUR LE DIMENSIONNEMENT DE MOTEURS A RELUCTANCE VARIABLE

Christophe Sauvey

► **To cite this version:**

Christophe Sauvey. CONTRIBUTION METHODOLOGIQUE A LA MODELISATION POUR LE DIMENSIONNEMENT DE MOTEURS A RELUCTANCE VARIABLE. Energie électrique. Institut National Polytechnique de Grenoble - INPG, 2000. Français. NNT: . tel-00689905

HAL Id: tel-00689905

<https://theses.hal.science/tel-00689905>

Submitted on 23 Apr 2012

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.

Remerciements

Je remercie les directeurs du Laboratoire d'Electrotechnique de Grenoble pour m'avoir accueilli au sein de leur univers.

Je tiens à remercier :

M. Albert Foggia, professeur à l'Institut National Polytechnique de Grenoble pour m'avoir fait l'honneur de présider le jury.

M. Claude Marchand, maître de conférences à l'université de Paris XI, et M. Jean-Pierre Masson, Professeur des Universités à l'INSA de Lyon, pour avoir accepté d'être rapporteurs de ce travail.

M. Frédéric Wurtz, chargé de recherches au CNRS, pour avoir encadré mes travaux de recherche pendant les trois ans.

M. Jean Bigeon, directeur de recherches au CNRS et chef de l'équipe CDI, pour avoir été mon directeur de thèse pendant trois ans et avoir, par conséquent, accompli l'ensemble des tâches qui lui incombent.

Cette thèse a été effectuée dans le cadre d'un contrat CIFRE avec la société Radio Energie. Je tiens donc à remercier ici M. Claude Poirson, son PDG, pour m'avoir recruté dans son effectif et Philippe Le Lan, responsable du service recherche et développement de la société. Je tiens aussi à remercier mes responsables directs successifs Philippe Siad Farah, Florent Binet, Jean-Louis Martin et Benali Bouallem, avec chacun desquels j'ai pu travailler librement et dans une ambiance agréable.

Je remercie également tous les membres de l'équipe CDI du LEG avec une mention particulière à Alain Bolopion qui me laissera un excellent souvenir, et pas seulement parce qu'il a participé intelligemment aux nombreuses corrections.

Je souhaite aussi remercier toute l'équipe informatique : E. Callegher, P. Eustache, P. Labie, M.T. Loubinoux, B. Ferrari qui ont été d'agréables secours dès que l'informatique me faisait des misères. Particulièrement Florence et Patrick (G.) qui sont devenus des amis.

Je voudrais aussi remercier Elise, Jacqueline, Monique et Danielle qui ont toujours été là lorsqu'il s'est agi de me remonter le moral après les moments difficiles (c'est la vie ...), au café ou ailleurs...

Je tiens maintenant à remercier toutes les personnes avec lesquelles je suis allé faire les pauses nécessaires au bon équilibre de tout travailleur : Coralie, Séverine, Nadège, Jean-François, Jean-Eric, Gérald, Nikola (footballeur bien connu de l'étoile rouge !), Bertrand et Charlotte (merci pour le pot !), et tous les autres thésards que j'ai oublié de citer, mais qui se reconnaîtront sans aucun doute.

Cette thèse m'a également permis d'élever mon niveau de jeu grâce au contact hebdomadaire de la grande équipe du Dynamo du LEG ! Orphée, Joël, Stéphane, Alain, Bruno, Patrice, Sami et Nikola, son vaillant capitaine !

Sans cette thèse, je n'aurais peut être jamais rencontré Jean-Louis, un grand professionnel plein de bon sens qui a compris, avant beaucoup, que faire confiance aux gens n'est pas forcément écorner son image de chef. Bien au contraire...

Enfin, je citerai les nombreuses personnes qui m'ont soutenu tout au long de ces 3 années et ont su m'encourager à persévérer malgré les difficultés : Papa et Maman, La Puce et Yohann, Nadège, Jeannine et les siens, Jean-Louis, tous mes amis extérieurs au labo et ceux qui, ici ou ailleurs, pensent à moi.

Je terminerai en disant un grand merci à Papa et Maman pour m'avoir toujours encouragé à aller plus haut, plus loin.

Table des Matières

INTRODUCTION	1
1 PROBLÉMATIQUE DE DIMENSIONNEMENT DES MOTEURS À RÉLUCTANCE VARIABLE.	5
1.A NOTRE APPROCHE DU DIMENSIONNEMENT.	7
1.A.1 <i>Définition du problème de dimensionnement.</i>	7
1.A.1.1 Etape de modélisation	8
1.A.1.2 Etape de dimensionnement	9
1.A.2 <i>Justification de nos choix de modèles et de méthodes de dimensionnement.</i>	10
1.A.2.1 Besoin de modèles dérivables.	10
1.A.2.2 Choix des algorithmes de type gradient.	13
1.A.3 <i>Définition d'un problème d'optimisation sous contraintes.</i>	16
1.A.4 <i>Formulation du problème de dimensionnement en un problème d'optimisation sous contraintes.</i>	17
1.B DÉFINITION DES TROIS GRANDS TYPES DE MODÈLES.	17
1.B.1 <i>Modèle Analytique.</i>	18
1.B.2 <i>Modèle Semi Analytique.</i>	19
1.B.3 <i>Modèle Numérique.</i>	20
1.B.4 <i>Analyse des caractéristiques des différents modèles en vue de leur utilisation pour le dimensionnement.</i>	20
1.B.4.1 Facilité de calcul des dérivées.	21
1.B.4.2 Connaissances consommables par le concepteur.	21
1.B.4.3 Eléments de choix de modèle.	22
1.C DIFFÉRENTES APPROCHES POUR LE DIMENSIONNEMENT DE MOTEURS À RÉLUCTANCE VARIABLE.	23
1.C.1 <i>Difficulté de dimensionnement d'un moteur à réluctance variable.</i>	23
1.C.2 <i>Avec un modèle éléments finis et une méthode d'interpolation adaptée à l'optimisation.</i>	25
1.C.2.1 Présentation de la méthodologie.	25
1.C.2.2 Positionnement par rapport à cette approche.	27
1.C.3 <i>Avec un modèle semi analytique et une méthodologie d'optimisation de type gradient.</i>	28
1.C.4 <i>Avec deux modèles complémentaires.</i>	28
1.D CONCLUSION	29
2 LES BOÎTES DE CALCUL : UNE MÉTHODOLOGIE POUR INTÉGRER DES MODÈLES SEMI ANALYTIQUES DANS UN PROCESSUS D'OPTIMISATION DE TYPE GRADIENT.	31
2.A MÉTHODOLOGIE PASCOSMA ET LIMITES DANS LA PRISE EN COMPTE DES MODÈLES SEMI ANALYTIQUES.	34
2.A.1 <i>Présentation de la méthodologie PASCOSMA.</i>	34
2.A.2 <i>Limites de PASCOSMA dans la prise en compte des modèles semi analytiques.</i>	39
2.A.2.1 Les paramètres implicites.	40
2.A.2.2 Calcul des paramètres de sortie.	42
2.A.2.3 Calcul des dérivées des paramètres de sortie.	42
2.A.2.4 Conclusion.	43

2.A.3	<i>Encapsulation du problème.</i>	44
2.A.4	<i>Outil générique</i>	45
2.A.4.1	Données nécessaires à la génération d'une boîte de calcul	45
2.A.4.2	Fichiers générés par l'outil	46
2.A.4.3	Fonctionnement général	48
2.B	GUIDE MÉTHODOLOGIQUE PROPOSÉ PAR LES BOÎTES DE CALCUL POUR GÉRER DES MODÈLES SEMI ANALYTIQUES.	49
2.B.1	<i>Définition d'une boîte de calcul.</i>	50
2.B.1.1	Le concept de boîte noire.	50
2.B.1.2	Définition des interfaces.	51
2.B.1.3	Pourquoi ce formalisme ?	53
2.B.1.4	Intérêts pour le dimensionnement.	53
2.B.2	<i>Comment créer une boîte de calcul avec un modèle semi analytique ?</i>	57
2.B.2.1	Problèmes identifiés.	57
2.B.2.2	Solution sur les paramètres de sortie.	58
2.B.2.3	Solution sur la dérivée.	60
2.B.2.4	Implantation.	61
2.B.3	<i>Exemples de boîtes de calcul réalisées.</i>	62
2.B.3.1	Fonctions continues affines par morceaux.	62
2.B.3.2	Méthode des moindres carrés.	68
2.B.3.3	Utilisation de sommes finies	73
2.B.3.4	Calcul des racines d'un polynôme en fonction de ses coefficients.	75
2.B.4	<i>Boîtes de calcul qu'il pourrait être intéressant de réaliser.</i>	78
2.B.4.1	Calcul d'intégrales.	79
2.B.4.2	Equations différentielles.	80
2.C	EXEMPLE D'APPLICATION DE LA MÉTHODOLOGIE DES BOÎTES DE CALCUL AU DIMENSIONNEMENT D'UN MOTEUR À RÉLUCTANCE VARIABLE DE TYPE CLASSIQUE.	81
2.C.1	<i>Structure du modèle de MRV Classique.</i>	81
2.C.1.1	Structure du moteur.	81
2.C.1.2	Description du modèle semi analytique.	82
2.C.2	<i>Dimensionnement d'un MRV Classique avec PASCOSMA et les boîtes de calcul nécessaires.</i>	86
2.C.2.1	Utilisation d'une fonction affine par morceaux.	86
2.C.2.2	Méthode des moindres carrés.	87
2.C.2.3	Résultats de dimensionnement.	88
2.D	CONCLUSION.	93
3	MODÉLISATION EN VUE DU DIMENSIONNEMENT D'UN MOTEUR À RÉLUCTANCE VARIABLE À MOTIFS.	95
3.A	PRÉSENTATION DE LA STRUCTURE DU MOTEUR À RÉLUCTANCE VARIABLE À MOTIFS À FLUX RADIAL.	97
3.A.1	<i>Structure du moteur.</i>	97
3.A.1.1	Noyaux magnétiques.	98
3.A.1.2	Création d'une couronne par assemblage de noyaux.	98
3.A.1.3	Création d'un moteur par assemblage de couronnes.	99

3.A.1.4 Conclusion.	101
3.A.2 <i>Electronique associée.</i>	101
3.A.2.1 Electronique de puissance.	101
3.A.2.2 Electronique de commande.	103
3.B MODÈLE ANALYTIQUE.	104
3.B.1 <i>Construction modulaire du modèle.</i>	105
3.B.1.1 Raison structurelle de ce choix.	105
3.B.1.2 Méthode de création du modèle.	105
3.B.1.3 Intérêts pour le modèle.	105
3.B.1.4 Intérêts pour le processus de dimensionnement.	106
3.B.2 <i>Modélisation du comportement statique.</i>	107
3.B.2.1 Hypothèses.	107
3.B.2.2 Description du modèle.	108
3.B.3 <i>Modélisation du comportement dynamique.</i>	111
3.B.3.1 Hypothèses.	112
3.B.3.2 Description du modèle.	113
3.C MODÈLE NUMÉRIQUE.	116
3.C.1 <i>Modélisation par éléments finis.</i>	116
3.C.1.1 Hypothèses	116
3.C.1.2 Difficulté d'obtenir la courbe $\Phi(i,\theta)$	117
3.C.1.3 Réflexion sur les différents modèles possibles.	117
3.C.1.4 Approche modulaire.	120
3.C.2 <i>Résolution de l'équation différentielle de circuit avec la méthode de Runge Kutta.</i>	121
3.C.2.1 Hypothèses.	121
3.C.2.2 Description du modèle.	122
3.D PROPOSITION D'UNE MÉTHODE DE DIMENSIONNEMENT.	123
3.D.1 <i>Comparaison des deux modèles</i>	124
3.D.2 <i>Plusieurs sources d'incertitudes dans les modèles.</i>	125
3.D.3 <i>Méthode de dimensionnement proposée.</i>	126
3.D.3.1 Rapprochement des deux modèles.	127
3.D.3.2 Application de dimensionnement avec le premier modèle.	127
3.D.3.3 Prise de décision et construction du prototype.	128
3.D.3.4 Essais et correction des modèles.	128
3.E CONCLUSION.	129
CONCLUSION	131
BIBLIOGRAPHIE	135

Introduction

Les concepteurs en génie électrique se trouvent confrontés en permanence à la difficulté de devoir dimensionner des dispositifs à la géométrie souvent complexe dans lesquels interviennent des phénomènes non linéaires et fortement couplés (électromagnétiques, mécaniques, thermiques, ...). Ils doivent répondre à des cahiers des charges induisant des contraintes sur ces paramètres.

Dans notre vie quotidienne, on fait de plus en plus appel aux petits actionneurs électriques. Ils sont généralement fabriqués en très grande séries de sorte à ce qu'ils soient moins chers. Parmi eux, le moteur à réluctance variable connaît un développement important. Rappelons rapidement que le moteur à réluctance variable fonctionne grâce au principe de conversion électromécanique par variation de réluctance, établi et mis en œuvre dès les tous premiers débuts de l'électrotechnique. L'intérêt que présente ce moteur est qu'il n'est pas cher à l'achat (il ne contient pas de cuivre au rotor, donc il contient globalement moins de cuivre). De plus, il est robuste et donc moins cher à l'usage (simple empilement de tôles au rotor, pas besoin de changer les systèmes balais/collecteurs).

En revanche, et malgré l'intérêt qu'il suscite, on a moins d'expérience sur le MRV que sur les autres moteurs dans la mesure où il est plus jeune et encore moins utilisé que les autres. Notre objectif est de fournir au concepteur une aide au dimensionnement de moteurs à réluctance variable.

En effet, pour construire un nouveau produit, il faut le dimensionner. Et pour dimensionner quelque chose, il faut se baser sur un modèle. C'est ce que nous commencerons à présenter dans le premier chapitre. Nous y présenterons notre approche du dimensionnement principalement basée sur deux étapes : la modélisation puis le dimensionnement en lui même. Nous y définirons aussi les trois grands types de modèles qu'un concepteur est susceptible de rencontrer. Enfin, nous présenterons différentes approches qui ont été adoptées pour concevoir des moteurs à réluctance variable.

Nous avons travaillé sur deux grands types de moteurs à réluctance variable (MRV) que nous présenterons en détails : celui de type classique dont la structure sera rappelée dans le deuxième chapitre et le MRV à motifs pour lequel elle est vraiment différente et sera présentée au début du troisième chapitre.

Pour le travail que nous avons effectué sur le MRV classique, nous disposions déjà d'un modèle. Nous avons donc travaillé sur le dimensionnement « à modèle imposé ». Cela nous a permis de mettre en évidence ce que nous avons appelé les « boîtes de calcul » qui permettent de réaliser un dimensionnement avec des techniques d'optimisation sur un modèle semi analytique. Dans le deuxième chapitre, nous proposons un guide méthodologique permettant de refaire le travail qui a été le notre pour d'autres parties de modèles posant problème. Nous illustrons notre propos avec le dimensionnement d'un moteur à réluctance variable de type classique à l'aide des boîtes de calcul que nous avons développées.

Pour ce qui concerne le MRV à motifs, le problème qui nous a été posé était différent dans la mesure où nous n'avions pas de modèle. Nous avons donc travaillé ici plutôt sur ce que l'on pourrait appeler la modélisation pour le dimensionnement. Nous avons créé deux modèles que nous estimons complémentaires. Dans le troisième chapitre, nous commençons par présenter la structure peu commune de ce moteur à réluctance variable à motifs, puis nous présentons successivement les deux modèles que nous avons créés. L'un est plutôt analytique et l'autre plutôt numérique. Sur cette complémentarité et après avoir comparé les deux modèles, nous proposons enfin un processus de dimensionnement dont les principales étapes sont le rapprochement des deux modèles, la création d'une application de dimensionnement avec le premier modèle (plus analytique), la construction d'un prototype et enfin les essais qui débouchent sur la correction des modèles.

1 Problématique de dimensionnement des moteurs à réluctance variable.

L'objectif de ce chapitre est de présenter les problèmes auxquels nous nous sommes confrontés au cours de ce travail. Notre objectif général est la contribution à la modélisation pour le dimensionnement de moteurs à réluctance variable (MRV).

Nous commencerons par en décrire notre approche, d'abord en donnant notre définition du problème, puis en justifiant nos choix de modèles et de méthodes. Nous définirons ce qu'est un problème d'optimisation. Enfin, nous nous attacherons à montrer l'équivalence entre un problème de dimensionnement sous contraintes et un problème d'optimisation sous contraintes.

Ensuite, nous définirons les trois grands types de modèles que l'on peut rencontrer pour faire du dimensionnement avec des techniques d'optimisation : les modèles analytiques, semi analytiques et numériques. Nous procéderons aussi à une comparaison en insistant sur les différences fondamentales entre analytique et numérique, puis nous en dégagerons les éléments principaux de choix d'un modèle à des fins de conception.

Enfin, nous présenterons différentes approches appliquées aux cas de moteurs à réluctance variable. Nous insisterons particulièrement sur les deux approches que nous avons suivies au cours de ce travail de sorte à introduire les chapitres qui suivront.

1.A Notre approche du dimensionnement.

Plusieurs méthodes existent. En effet, le concepteur n'utilise pas forcément toujours les *mêmes modèles* (il n'a pas forcément toujours le même produit à concevoir), ni n'utilise pas forcément non plus toujours les *mêmes méthodes* pour arriver à dimensionner son dispositif (il n'en sera pas toujours au même niveau d'avancement dans son travail).

1.A.1 Définition du problème de dimensionnement.

Selon l'analyse de [COUTEL-99], un problème de dimensionnement comporte trois éléments :

- une représentation paramétrée du système à concevoir qui correspond à la description de la structure à dimensionner.

- un modèle mathématique, qui permet d'évaluer les autres paramètres du système.
- une logique de dimensionnement, qui décrit la démarche d'obtention d'un ensemble de valeurs de paramètres respectant le cahier des charges spécifié.

Il faut d'abord procéder à une phase de **modélisation du système**. On veut en effet pouvoir évaluer quantitativement, en fonction des grandeurs descriptives du système, l'expression des variables qui caractérisent le fonctionnement du système, et particulièrement celles qui figurent dans le cahier des charges. Le concepteur devra choisir parmi les nombreuses possibilités de modélisation qui s'offrent à lui.

Ensuite, il devra dimensionner avec le modèle qu'il aura choisi. Le **dimensionnement** consiste à réaliser l'opération inverse du calcul d'analyse avec un modèle. En effet, alors qu'un modèle exprime les performances d'un système en fonction de ses paramètres de construction, le dimensionnement est la recherche des paramètres de construction, à partir d'un fonctionnement à réaliser spécifié dans le cahier des charges. C'est ce qu'on appelle généralement la résolution du problème inverse.

1.A.1.1 Etape de modélisation

On doit toujours s'appuyer sur un modèle pour dimensionner un produit, quel qu'il soit. La modélisation est la première étape de tout processus de dimensionnement. Ce que nous désignons par modèle est simplement une manière, souvent mathématique et géométrique, de décrire le produit à dimensionner. Cette description est un moyen de calculer des paramètres de sortie en fonction des paramètres d'entrée.

Les **modèles analytiques**, dont nous donnerons une définition détaillée plus loin dans ce chapitre, sont incontournables dans la phase de préconception de tout dispositif électromagnétique [BOULES-90]. On peut citer le cas des machines asynchrones où pratiquement tous les livres de dimensionnement de ces machines s'appuient sur des modèles analytiques [ALGER-70], [COCHRAN-89]. Des modèles analytiques simplifiés existent même sous forme d'abaques ou de règles d'expertise.

Cette catégorie de modèles est donc l'outil de base du concepteur, au moins pour les phases initiales de son travail de dimensionnement.

D'une manière générale, lorsque l'on parle de modélisation, on lui oppose souvent les **modèles numériques**. En effet, ces derniers sont plus lourds à manipuler et ne permettent pas au concepteur de voir les dépendances entre les différents paramètres dès que leur nombre augmente. La connaissance physique du système à concevoir n'y est pas formalisée, elle y existe de manière implicite. De plus, ces méthodes sont souvent coûteuses en temps de calcul, ce qui peut devenir rédhibitoire dans le choix du modèle pour certaines applications. Ce genre de modèles est souvent utilisé pour le dimensionnement fin d'une partie de l'ensemble à concevoir, ce qu'on appelle généralement « optimisation de forme » [KADDED-93]. On peut, dans cette optique, les utiliser en fin de dimensionnement pour affiner une structure dimensionnée par d'autres méthodes.

Nous donnerons nos définitions de modèles analytique et numérique ainsi que quelques éléments de choix de modèles pour un processus de dimensionnement plus loin dans ce chapitre. Cependant, on peut déjà constater que, quelle que soit la définition que l'on donne à ces deux qualificatifs (analytique et numérique), on trouve toujours des modèles « inclassables », ce qui nous donne donc toujours une troisième catégorie de modèles. Dans celle-ci, on peut trouver, par exemple, des réseaux de réluctances saturables (dont les valeurs des réluctances qui les composent varient en fonction du flux qui les traverse). Le nom que nous avons choisi pour cette catégorie de modèles est semi analytique.

1.A.1.2 Etape de dimensionnement

Une fois que la structure est modélisée, il faut ensuite passer à la phase de dimensionnement proprement dite. Le concepteur doit trouver une solution au problème posé par le cahier des charges. Plusieurs possibilités s'offrent à lui allant de la plus simple à la plus compliquée :

- Il peut itérer simplement sur son modèle en appliquant une boucle simple de type essai/erreur.

- Il peut suivre une approche procédurale qui consiste à fixer certains paramètres et d'en déduire les autres au fur et à mesure, en revenant en arrière en cas de violation de contrainte et en effectuant des itérations pour améliorer [BELOT].
- Il peut utiliser des systèmes experts. En effet, en plus de servir au choix de la structure, ils permettent également d'optimiser le dimensionnement de cette structure [GENTILHOMME-91].
- Il peut utiliser des méthodes d'optimisation. Celles ci parcourent l'espace des solutions en cherchant la solution optimale au regard du cahier des charges imposé, en respectant les contraintes posées [COUTEL-99].

1.A.2 Justification de nos choix de modèles et de méthodes de dimensionnement.

Afin de justifier nos choix de modèles et de méthodes de dimensionnement, il faut commencer par signaler que nous nous plaçons dans les premières phases du travail du concepteur. Nous souhaitons contribuer à lui offrir un premier jeu de valeurs optimal à peu de frais, aussi bien pour le modèle que pour la méthode. Le terme « à peu de frais » signifie que nous souhaitons que ce premier processus de dimensionnement soit particulièrement peu coûteux en temps. Ainsi, cette rapidité permettra au concepteur, s'il le souhaite, de mettre une maquette en fabrication pour valider ses idées.

Nous allons expliquer pourquoi, afin de répondre aux exigences de rapidité et de précision définies ci-dessus, nous allons avoir besoin de modèles dérivables ainsi que d'algorithmes d'optimisation de type gradient.

1.A.2.1 Besoin de modèles dérivables.

Nous entendons par modèle dérivable un modèle pour lequel on est capable de calculer les dérivées partielles de toutes les sorties en fonction de toutes les entrées. Si nous avons les expressions des s_i en fonction des e_j , on veut pouvoir calculer de manière formellement exacte toutes les expressions $\delta s_i / \delta e_j$. Parmi tous les types de

modèles avec lesquels nous avons la possibilité de travailler, nous souhaitons utiliser des modèles dérivables.

Deux raisons principales nous mènent à ce choix :

- **Ils donnent des informations qualitatives au concepteur.**
 - sur l'interaction entre les paramètres. Il peut parfois lui être bien utile de connaître quels sont les paramètres qui influent sur une contrainte qu'il a du mal à satisfaire. Cette information lui est donnée par la dérivée : si celle ci est égale à zéro, le paramètre de sortie ne dépend pas du paramètre d'entrée considéré (en tout cas pour ce point particulier de calcul).
On peut obtenir cette information même si le modèle n'est pas dérivable mais simplement explicite. Pour les deux points suivants, la dérivabilité du modèle est une condition nécessaire.
 - sur le sens d'influence des paramètres d'entrée sur une sortie. Cette information est donnée par le signe de la dérivée. Si elle est positive, c'est qu'une augmentation du paramètre d'entrée entraîne une augmentation du paramètre de sortie considéré (et inversement). On peut ainsi d'un coup d'œil voir les actions à mener sur les paramètres d'entrée pour faire entrer un paramètre de sortie dans un intervalle (contrainte) donné(e).
 - sur le degré d'influence de ces paramètres sur une sortie. Cette information est donnée par la valeur absolue de la dérivée. Cette valeur est d'autant plus grande que l'influence du paramètre d'entrée est forte sur le paramètre de sortie considéré. Sur un problème peu contraint, le concepteur voit ainsi rapidement quels sont les paramètres sur lesquels il est intéressant de jouer pour obtenir rapidement la solution.
- **Ils permettent de converger plus rapidement et plus précisément vers une solution.** On peut illustrer cela avec un exemple simple. Prenons une fonction f qui exprime un paramètre de sortie s_k en fonction de n paramètres d'entrée (e_1, \dots, e_n) . On a : $s_k = f(e_1, \dots, e_n)$.

- Rapidité offerte par la dérivée : si on veut faire varier le paramètre s_k de Δs_k parce qu'il sort de la contrainte sur s_k de cette valeur, uniquement en jouant sur e_1 , on a besoin de la dérivée partielle $\frac{\partial f}{\partial e_1}$ pour savoir de quel

Δe_1 on devra faire varier le paramètre e_1 .

- Si on l'a parce que l'on dispose des gradients, un appel à cette fonction suffira.
- Si on ne dispose pas des gradients, il faudra le calculer ainsi :

$$\frac{\partial f}{\partial e_1} = \frac{f(e_1 + h, e_2, \dots, e_n) - f(e_1, e_2, \dots, e_n)}{h}$$

Cela implique deux appels à la fonction f , alors que dans le premier cas, un seul appel à la dérivée suffit.

- Précision offerte par le calcul exact de la dérivée : Le problème qui existe lorsque l'on procède à un calcul numérique de la dérivée du type $\frac{f(x+h) - f(x)}{h}$ est que l'on introduit des erreurs de précision sur la valeur de cette dérivée [RAM-73]. Cette erreur peut avoir des conséquences sérieuses, pouvant aller jusqu'à un échec de convergence.

Les modèles analytiques sont facilement dérivables. Nous venons de voir les deux principaux avantages qu'apportent des modèles dérivables (informations pour le concepteur et bonnes propriétés de convergence). Nous choisirons donc d'utiliser, à chaque fois que cela nous sera possible, des modèles analytiques. De plus, ces modèles présentent l'atout d'être rapides en temps de calcul. Ceci est parfaitement compatible avec notre volonté de proposer au concepteur un processus de dimensionnement (ou de pré-dimensionnement selon l'usage qu'il souhaite en faire) rapide.

Ensuite, si le concepteur peut, en plus d'avoir un modèle dérivable (c'est à dire lui donnant des informations qualitatives et convergeant rapidement vers une solution), avoir une solution optimale, il ne s'en plaindra pas. C'est pourquoi nous allons voir dans le paragraphe suivant que nous choisissons préférentiellement les algorithmes de type gradient qui permettent d'utiliser les informations données par les dérivées.

1.A.2.2 Choix des algorithmes de type gradient.

Le choix de modélisation qui est fait induit le choix de la méthode de dimensionnement que l'on va lui adjoindre. En effet, comme nous allons dériver le modèle pour les raisons expliquées plus haut, il serait dommage de ne pas utiliser cette information pour dimensionner.

Nous allons donc nous intéresser dans un premier temps à la manière dont prennent en compte les contraintes les méthodes de dimensionnement proposées plus haut, puis nous regarderons ensuite leurs capacités à prendre en compte de manière explicite les contraintes.

1.A.2.2.1 *Prise en compte des dérivées.*

Nous allons regarder successivement les méthodes de dimensionnement qui s'offrent à nous et regarder quelles sont celles qui prennent le mieux en compte les dérivées, en particulier celles qui sont le moins coûteuses en temps et offrent un certain confort au concepteur, c'est à dire celles qui lui demandent le moins de travail.

- Les méthodes du type essai/erreur permettent la prise en compte des dérivées mais demandent un gros travail au concepteur pour converger. Avec de telles méthodes, c'est le concepteur lui même qui se charge de la convergence. La difficulté de la tâche croît avec le nombre de paramètres
- Les approches procédurales permettent elles aussi la prise en compte des contraintes. Elles demandent un peu moins de travail au concepteur dans la mesure où elles le guident avec un cahier des charges fixé. Elles restent néanmoins assez coûteuses en temps.
- Les systèmes experts sont les moins bien adaptés à la prise en compte des dérivées pour l'optimisation. Ils ont aussi l'inconvénient d'être assez coûteux en temps. Cependant, ils offrent une très bonne facilité d'utilisation en raison de leur généricité.
- Les méthodes d'optimisation sont sans doute celles qui prendront le mieux en compte les dérivées. De plus, elles offrent l'avantage d'être les moins coûteuses en temps de calcul et demeurent faciles à mettre en œuvre pour utiliser l'information fournie par les dérivées.

C'est pour cette raison que nous avons choisi d'utiliser des méthodes d'optimisation de type gradient pour le dimensionnement.

1.A.2.2.2 Prise en compte des contraintes.

En plus de notre volonté d'utiliser les dérivées pour dimensionner, nous souhaitons prendre en compte les contraintes de manière explicite. Nous devons donc choisir une méthode adaptée à notre volonté. Si nous reprenons les quatre méthodes de dimensionnement décrites précédemment, nous allons nous rendre compte que les méthodes d'optimisation sont, là encore, les mieux adaptées pour relever ce défi :

- Les méthodes de type essai/erreur ainsi que les approches procédurales demandent trop de travail au concepteur. S'il est trop occupé à faire converger les résultats et à veiller au respect des contraintes, il risque de perdre du recul vis à vis de ceux-ci pour les analyser (ce qui constitue l'objet de son travail).
- Les systèmes experts savent mal prendre en considération les contraintes lorsqu'elles sont exprimées sous forme mathématique, de même qu'ils ne sont pas bien adaptés pour prendre en compte les dérivées pour l'optimisation.
- Les méthodes d'optimisation, elles, savent parfaitement gérer les contraintes, ce qui leur donne encore un atout supplémentaire. Elles doivent cette aptitude particulière à leur mécanisme d'exploration de l'espace des solutions [WURTZ-96].

Sur ce point de la prise en compte des contraintes, les méthodes d'optimisation sont encore les mieux adaptées. Il nous reste maintenant à montrer pourquoi, outre le fait qu'elles utilisent les dérivées et qu'il serait dommage de s'en passer, **les méthodes d'optimisation utilisant le gradient sont plus performantes** que celles ne l'utilisant pas **pour la gestion des contraintes**.

1.A.2.2.3 *Justification du choix des algorithmes.*

Nous voulons pouvoir gérer efficacement (c'est à dire en offrant un temps de réponse convenable) beaucoup de paramètres et de contraintes. Nous allons exprimer successivement nos volontés pour mettre en évidence la méthode d'optimisation la mieux adaptée à notre problème. Pour résumer nos volontés, on peut dire que nous voulons une méthode qui sache gérer beaucoup de contraintes, qui puisse gérer beaucoup de paramètres et, si possible, qui ne se laisse pas tromper par les optima locaux. Cependant, comme nous savons qu'il sera difficile de trouver une méthode qui satisfait absolument tous ces critères, nous sommes prêts à faire des concessions sur le troisième (les optima locaux).

- Nous voulons une méthode qui utilise explicitement les contraintes, parce que **nous pensons qu'elle sera en mesure d'en gérer un plus grand nombre**. Cela nous permet d'éliminer parmi les méthodes d'optimisation celles qui utilisent implicitement les contraintes. Ce sont des méthodes de minimisation dans lesquelles les contraintes sont introduites sous forme de pénalités. On peut citer parmi celles ci les algorithmes stochastiques comme les algorithmes génétiques ou le recuit simulé [DE VASC-94]. Ces algorithmes sont assez coûteux en temps de calcul. On peut également citer les algorithmes de type gradient avec méthode de transformation qui sont moins coûteux en temps de calcul grâce à l'utilisation des gradients, mais avec lesquels on résout le même problème (les contraintes sont toujours prises en compte sous forme de fonctions de pénalités) [SALD-88] [KADDED-93].
- Nous voulons aussi une méthode qui prenne en compte beaucoup de paramètres. Malheureusement, les méthodes qui ne se laissent pas tromper par des optima locaux sont incapables de prendre en compte beaucoup de paramètres pour des temps de réponse raisonnables. On peut citer parmi celles ci les méthodes de Branch and Bound. **Notre algorithme devra donc être choisi parmi ceux qui se laissent tromper par des optima locaux**. Nous acceptons ce risque en sachant que la solution que nous donnera l'algorithme sera sensible au point de départ.

La méthode d'optimisation que nous avons choisie est la méthode de **Minimisation Séquentielle Quadratique (MSQ)** [POWELL-85]. Cette méthode d'optimisation résout, à chaque itération (*séquentielle*), un problème simplifié dans lequel on suppose que la fonction objectif est *quadratique* et où toutes les contraintes sont ramenées à des droites. Il est important de préciser que la création de recherche est choisie en prenant explicitement en compte les contraintes. La méthode renvoie le jeu de paramètres pour lequel la fonction objectif est *minimale* avec ces hypothèses. On recalcule les dérivées du problème en ce nouveau point et on a un nouveau problème à soumettre à l'algorithme. Lorsque le problème est bien posé, on peut espérer converger vers la solution au fil des itérations, chaque point de calcul privilégiant une partie de l'espace de recherche à l'autre grâce aux dérivées.

Après avoir expliqué les deux principales raisons de notre choix d'utiliser les méthodes d'optimisation utilisant des algorithmes de type gradient, nous allons maintenant expliquer plus en détails à quoi se rapporte un problème d'optimisation sous contraintes.

1.A.3 Définition d'un problème d'optimisation sous contraintes.

La formulation du problème d'optimisation sous contraintes se présente ainsi. On veut parcourir l'espace des solutions, en cherchant à résoudre le problème suivant [DE VASC-94] :

$$\begin{cases} \min f(p) \\ g_i(p) \leq 0, \forall i \in \{1, \dots, l\} \\ g_i(p) = 0, \forall i \in \{l+1, \dots, m\} \\ p_j \min \leq p_j \leq p_j \max, \forall j \in \{1, \dots, k\} \end{cases} \quad (1)$$

où :

- p est un vecteur de dimension k , contenant les paramètres de l'espace à explorer, qui sont eux mêmes soumis à des contraintes d'inégalité ($p_{j\min} \leq p_j \leq p_{j\max}$),
- $fob(p)$ est la fonction objectif dépendant de ces paramètres,
- $g_i(p)$ sont des fonctions dépendant aussi de ces paramètres et sont soumises à des contraintes d'inégalité (pour $i=1, l$) et d'égalité (pour $i=l+1, m$).

1.A.4 Formulation du problème de dimensionnement en un problème d'optimisation sous contraintes.

Plusieurs auteurs ont montré l'équivalence du problème de dimensionnement sous contraintes avec un problème d'optimisation sous contraintes [KONE-93], [SALD-88]. Ces contraintes imposées par le cahier des charges doivent être prises en compte par l'algorithme d'optimisation utilisé.

La démarche de dimensionnement envisagée, qui vise à rechercher une solution satisfaisant aux critères du cahier des charges, sous les contraintes définissant les spécifications de celui-ci et la structure considérée se présente sous la forme d'un problème d'optimisation [KONE-93].

En effet, le modèle mathématique du dispositif à dimensionner associé à l'ensemble des contraintes définies dans le cahier des charges définit un espace de solutions dans lequel l'algorithme d'optimisation utilisé pourra rechercher la solution optimale. Il déterminera ainsi les valeurs des variables définissant les dimensions et les caractéristiques de la structure, selon les besoins du cahier des charges.

Nous allons donc résoudre notre problème de dimensionnement en le reformulant en un problème d'optimisation.

1.B Définition des trois grands types de modèles.

Plusieurs types de modèles existent aujourd'hui pour modéliser un dispositif électromagnétique. Cependant, les exigences du concepteur en termes de rapidité et de précision guideront ses choix. Nous allons donner une définition précise des termes que nous allons désormais employer pour distinguer des modèles analytiques, semi-analytiques ou numériques. Il est nécessaire de préciser ici que les définitions qui seront données dans la suite de ce paragraphe sont « personnelles » par opposition à « rigoureusement mathématiques ». Elles sont simplement données ici pour aider le lecteur à comprendre notre propos.

Dans ce paragraphe, nous allons supposer que notre modèle, quel qu'il soit, aura n entrées distinctes et p sorties distinctes. Nous définissons les paramètres d'entrée de notre modèle comme le n -uplet (e_1, e_2, \dots, e_n) et, de même, nous définissons nos paramètres de sortie comme le p -uplet suivant (s_1, s_2, \dots, s_p) .

1.B.1 Modèle Analytique.

Nous définissons comme modèle analytique un modèle tel que :

$$\forall k \in \{1, \dots, p\}, \exists f_k : \mathfrak{R}^n \rightarrow \mathfrak{R} \text{ t.q. } s_k = f_k(e_1, e_2, \dots, e_n).$$

Chaque f_k vérifiant les propriétés suivantes :

- il contient exclusivement des équations explicites, continues et continûment dérivables permettant de calculer la valeur d'un réel dès que la valeur des n autres réels en fonction desquels il est exprimé est connue.
- Il ne contient aucune boucle. Une boucle est la définition d'une variable en fonction d'elle-même. C'est la condition qui est exprimée sur la deuxième ligne de l'équation ci-dessous.

$$s_k = \begin{cases} f(e_1, \dots, e_n) \\ s_k \neq e_i, \forall i \in \{1, \dots, n\} \end{cases} \quad (2)$$

Une boucle est contenue dans le système (3) qui met en équation un problème courant en automatique présenté Figure 1 :

$$\begin{cases} S = K \cdot e \\ e = E - G \cdot S \end{cases} \quad (3)$$

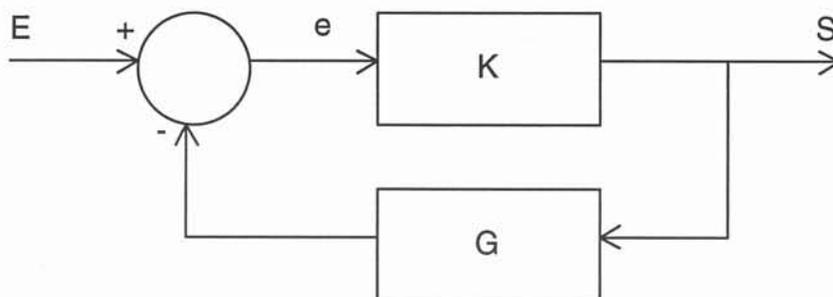


Figure 1.1 : Boucle de retour en automatique.

Dans ce cas, la boucle est bien visible, mais avec des modèles analytiques comprenant plusieurs dizaines d'équations, les boucles sont beaucoup plus difficiles à déceler. La boucle est également facile à « casser » car le système (3) est équivalent à (4) qui, lui, ne contient plus de boucle.

$$\begin{cases} S = \frac{K}{1+K \cdot G} \cdot E \\ e = E - G \cdot S \end{cases} \quad (4)$$

En revanche, avec des modèles plus complexes, la boucle peut être beaucoup plus difficile à casser et on peut parfois tomber sur des impossibilités. C'est le cas lorsque l'on aboutit à une fonction implicite. On peut prendre pour exemple le cas de réseaux de réductances qui dépendent non seulement des paramètres géométriques, mais aussi des valeurs des flux qui les traversent [COUTEL-99]. On aboutit à des équations du type $R(\phi) \cdot \phi = N \cdot I$. Dès qu'il contient de telles équations, nous ne considérons plus le modèle comme purement analytique. Nous allons voir par la suite dans quelle catégorie il va se classer.

1.B.2 Modèle Semi Analytique.

Nous définissons comme modèle semi analytique un modèle tel que :

$\exists q \in \{1, \dots, p\}$ t.q. :

- $\forall k \in \{1, \dots, q\}, \exists f_k : \mathfrak{R}^n \rightarrow \mathfrak{R}$ t.q. $s_k = f_k(e_1, e_2, \dots, e_n)$.
- $\exists (g_{q+1}, \dots, g_p)$ base d'un espace vectoriel de fonctions (de dimension $p-q$) t.q. : $\forall l \in \{q+1, \dots, p\}, \exists g_l : \mathfrak{R}^{n+1} \rightarrow \mathfrak{R}$ t.q. $g_l(e_1, e_2, \dots, e_n, s_l) = 0$.
- $p - q \ll p$.

Un modèle semi analytique est un modèle dont la plupart des équations sont analytiques et dont quelques unes ne s'écrivent pas sous la forme définie par (2). C'est à dire que, pour certains paramètres de sortie, nous n'avons pas d'expression analytique claire et explicite (comme pour (2)) nous permettant de les calculer.

Ainsi, lorsque l'on a un modèle analytique en apparence, mais qui contient une boucle sur plusieurs équations, on peut facilement montrer que ce modèle se classera dans la catégorie semi-analytique. On peut dire ici que quel que soit le paramètre de sortie s d'un modèle semi analytique, soit s est une fonction implicite des paramètres d'entrée, soit s se calcule en fonction de (e_1, e_2, \dots, e_n) avec un algorithme (ce qui revient pratiquement au même).

1.B.3 Modèle Numérique.

Nous définissons comme modèle numérique un modèle tel que :

$$\begin{aligned} \exists q \in \{1, \dots, p\} \text{ t.q. : } & \forall k \in \{1, \dots, q\}, \exists f_k : \mathfrak{R}^n \rightarrow \mathfrak{R} \text{ t.q. } s_k = f_k(e_1, e_2, \dots, e_n). \\ & \forall l \in \{q+1, \dots, p\}, \exists g_l : \mathfrak{R}^{n+1} \rightarrow \mathfrak{R} \text{ t.q. } g_l(e_1, e_2, \dots, e_n, s_l) = 0. \\ & q \ll p. \end{aligned}$$

La méthode des éléments finis est sans aucun doute celle qui définit le mieux les modèles que nous qualifions de numériques, bien qu'il en existe d'autres. Par exemple, la méthode des différences finies entre, elle aussi, dans cette catégorie de modèles. Ce sont, en règle générale, des modèles qui mettent en jeu la résolution de gros systèmes matriciels qu'il faut conditionner le mieux possible pour que l'inversion d'une matrice $N \times N$ (avec N grand) soit la plus aisée possible.

La méthode de Runge Kutta pour la résolution de systèmes d'équations différentielles entre aussi dans le cadre de cette définition car elle fait appel à des méthodes numériques pour résoudre les équations.

1.B.4 Analyse des caractéristiques des différents modèles en vue de leur utilisation pour le dimensionnement.

Maintenant que nous avons défini les trois grands types de modèles que nous serons amenés à utiliser, nous allons les comparer en insistant plus particulièrement sur les différences entre les modèles analytiques et numériques. Il faut bien garder à l'esprit que la finalité de la modélisation d'un dispositif, en ce qui nous concerne, est son dimensionnement et non la modélisation elle-même. A ce titre, si la meilleure précision d'un modèle numérique ne nous apporte rien pour le dimensionnement du dispositif concerné, nous nous contenterons d'un modèle analytique car ce qui intéresse le concepteur est avant tout la facilité d'utilisation [WURTZ-96].

La réflexion qui est d'usage pour le choix entre l'utilisation de modèles analytiques ou numériques est de trouver le meilleur compromis entre précision et rapidité. En effet, les modèles numériques sont réputés pour leur précision mais

demandent en général beaucoup de temps pour opérer un calcul. Les modèles analytiques sont, quant à eux, plus rapides mais moins précis.

1.B.4.1 Facilité de calcul des dérivées.

La première différence intéressante à souligner entre ces deux modèles est la facilité de calcul des dérivées qu'ils offrent.

Les **modèles analytiques offrent un calcul aisé des dérivées**. J'en veux pour preuve que l'on connaît les dérivées de toutes les fonctions classiques utilisées dans les modèles analytiques ainsi que leurs composées. Cela a d'ailleurs permis à certains logiciels de calcul mathématique de proposer le calcul formel des dérivées. Cette facilité de dérivation qu'offrent les modèles analytiques leur permet une convergence plus rapide, dans le cas d'un couplage avec un algorithme de type gradient, par exemple.

Dans le cas de **modèles numériques, le calcul des dérivées est beaucoup plus coûteux [NGYUEN-98]**. De plus, à défaut d'être moins fiable, il nécessite en tous cas des vérifications.

1.B.4.2 Connaissances consommables par le concepteur.

La différence sur laquelle il nous paraît le plus important d'insister est la somme de connaissances que l'on peut tirer de chacun des deux modèles.

D'un modèle analytique, on peut tirer des connaissances qualitatives très intéressantes, comme par exemple : quels sont les paramètres qui influent sur tel paramètre de sortie et en quelle mesure influent-ils ? Les modèles analytiques peuvent répondre à cette (ces) question(s) grâce à des formules claires et explicites.

Les modèles numériques ne font pas apparaître aussi clairement les dépendances entre paramètres, même si elles y sont prises en compte de manière plus précise. Ces informations sont contenues dans les modèles numériques, mais de façon implicite. L'utilisateur doit donc, pour obtenir les mêmes informations, lancer plusieurs résolutions et les étudier.

On peut résumer les constatations faites précédemment dans le tableau présenté dans le paragraphe suivant qui montre de manière schématique les

principales différences que nous avons relevées entre modèles analytique et numérique pour ce qui nous intéresse au final, le dimensionnement.

1.B.4.3 Eléments de choix de modèle.

Ces comparaisons étant faites, nous pouvons nous poser la question suivante :

Quel est le modèle dont le concepteur à besoin ?

- Est-ce le plus précis ? Ainsi, il aura de moins en moins de prototypes à construire avant d'obtenir un produit répondant parfaitement au cahier des charges.
- Est-ce le plus rapide ? Ainsi, il pourra concevoir le plus rapidement possible un prototype répondant à un cahier des charges donné. Ou encore, en un temps donné, il sera en mesure de traiter un plus grand nombre de cahiers des charges.

Réponse : Le concepteur veut tout

Il veut un modèle précis, car la construction de prototypes coûte cher, bien qu'elle soit indispensable. Une modélisation précise lui permet de savoir à quoi s'attendre et d'utiliser le prototype plus comme une vérification et un moyen d'améliorer encore son modèle que comme un simple essai de valeurs pour les paramètres de construction.

Il veut aussi un modèle rapide car il pourra essayer autant de jeux de paramètres d'entrée qu'il le désire pour satisfaire toutes les contraintes de son cahier des charges. Il pourra, grâce à un modèle rapide, concevoir plus en un temps plus court. Encore faut-il que la qualité ne s'en ressente pas.

En clair, le concepteur souhaite travailler *vite* et *bien* et ces deux possibilités lui sont offertes par les deux grands types de modèles dont il dispose. Pour travailler plus vite, il utilisera plutôt un modèle analytique, et pour avoir des résultats plus précis, il utilisera plutôt un modèle numérique. Cependant, dès que cela lui sera possible, il utilisera préférentiellement un modèle analytique car il pourra réfléchir réellement sur son problème grâce à la vision claire de l'influence de certains paramètres sur le résultat qui l'intéresse dans le cadre d'un cahier des charges donné.

Tableau 1 :Principales différences entre modèles analytique et numérique.

Analytique	Numérique
RAPIDE	Lent
Moins Précis	PRECIS
Calcul des dérivées aisé	On peut les calculer [NGUYEN-98]. Mais beaucoup PLUS coûteux
Informations Qualitatives <i>Qu'est ce qui influe sur quoi ?</i> <i>En quelles mesures ?</i>	Information Implicite
dérivées → convergence plus rapide	Convergence liée à l'utilisateur
Modèle de Conception	Modèle de Simulation

1.C Différentes approches pour le dimensionnement de moteurs à réluctance variable.

Dans ce paragraphe, nous nous proposons de mettre en évidence la difficulté de dimensionner un moteur à réluctance variable (MRV), puis de décrire différentes approches qui ont été imaginées pour dimensionner des MRV de types différents.

1.C.1 Difficulté de dimensionnement d'un moteur à réluctance variable.

La difficulté de dimensionnement d'un moteur à réluctance variable est principalement due à la difficulté de modélisation. La mise au point de processus de dimensionnement appropriés aux modèles est donc ardue.

C'est cette idée que l'on retrouve résumée dans la littérature sous la forme suivante qui est la plus rencontrée [MATT-95] :

« Pour modéliser un MRV, on ne peut pas se passer de modèles numériques ».

Nous pensons que cette obligation est due au fait que le flux magnétique n'est pas aussi bien canalisé dans ce moteur que dans les moteurs dits classiques. Nous appelons moteurs classiques les moteurs synchrones et asynchrones à courant alternatif et les moteurs à courant continu. Ils comprennent tous plusieurs bobinages ou des bobinages et des aimants permanents et un circuit magnétique dont une partie est mobile [GRELLET-97].

L'obligation d'en passer par un modèle numérique implique que la méthode de dimensionnement sera forcément plus compliquée que pour les moteurs classiques. En effet, pour ceux ci, d'une manière générale, les modèles analytiques suffisent, au moins pour les premières étapes du processus de dimensionnement [BOULES-90]. Pour les moteurs à réluctance variable, nous serons obligés, à un moment ou à un autre d'utiliser la complémentarité qui existe entre les modèles analytiques et numériques.

La difficulté de dimensionnement des moteurs à réluctance variable rend la tâche difficilement automatisable. En effet, beaucoup de méthodes de dimensionnement ont été développées, mais souvent les structures des moteurs pour lesquels on imagine ces méthodes diffèrent. De plus, comme la structure diffère, les modèles (liés à ces structures) diffèrent aussi. C'est ainsi que plusieurs approches de dimensionnement ont vu le jour.

Nous allons voir par la suite différentes approches qui ont été adoptées pour le dimensionnement de moteurs à réluctance variable. La première de celles qui seront présentées a été mise en œuvre sur un moteur à réluctance variable à double saillance [BRISSET-95] et utilise un modèle éléments finis paramétré avec un processus d'optimisation utilisant la méthode des plans d'expériences. La seconde approche est celle que nous avons adoptée sur le moteur à réluctance variable de type classique et consiste à utiliser un modèle semi analytique et un processus d'optimisation utilisant des algorithmes de type gradient. La dernière approche que nous allons introduire est celle qui a été la nôtre pour le moteur à réluctance variable à motifs. Nous allons voir que, quelle que soit l'approche de dimensionnement que suivra le concepteur de moteur à réluctance variable, il aura toujours besoin de cette complémentarité qui existe entre modèles analytique et numérique.

1.C.2 Avec un modèle éléments finis et une méthode d'interpolation adaptée à l'optimisation.

A l'Université des Sciences et Technologies de Lille, des chercheurs ont travaillé pour obtenir un outil d'aide à la conception [BRISSET-95]. La méthodologie de dimensionnement qu'ils proposent pour la conception de moteurs à réluctance variable à double saillance [BRISSET-95] est l'utilisation d'un modèle éléments finis paramétré avec un processus d'optimisation utilisant les plans d'expériences.

1.C.2.1 Présentation de la méthodologie.

1.C.2.1.1 Eléments Finis Paramétrés.

L'utilisation de la méthode des éléments finis paramétrés permet d'utiliser cette méthode numérique puissante et précise tout en pouvant changer les valeurs de certains paramètres de la géométrie qui auront été choisis pour leur influence particulière sur les performances du moteur à réluctance variable. Un choix pertinent de ces paramètres sera réalisé à l'aide de modèles analytiques très simples et très rapides, mais qui permettent quand même d'obtenir des informations importantes sur les interdépendances entre les paramètres. Le fait de paramétrer la géométrie permet de s'éviter, à chaque itération du processus de dimensionnement, de recréer une géométrie presque identique à la précédente. En effet, la méthode des éléments finis réclame la définition de points, de lignes et de surfaces pour définir le problème. Cette partie est fastidieuse et le paramétrage permet d'éviter de refaire tout ce travail à chaque fois que l'on change la valeur d'un paramètre géométrique.

1.C.2.1.2 Les plans d'expériences.

La technique des plans d'expériences [GOUPY-88], [PILLET-88] utilise la notion de plan optimal et ainsi se propose de faire des simulations où toutes les variables changent ensemble de façon organisée afin de prendre en compte les interactions et réduire le nombre de calculs nécessaires [BRISSET-95].

Cette méthode a été imaginée pour permettre d'obtenir un modèle de comportement d'un dispositif pour lequel l'obtention d'un point de fonctionnement par

l'expérience coûte cher. Elle trouve tout son intérêt lorsqu'elle est utilisée avec un modèle éléments finis, précis mais très lent. Ainsi, dès que la méthode des plans d'expériences permet d'éviter ne serait-ce qu'une simulation, elle fait gagner un temps précieux au concepteur.

Pour bien comprendre ce qu'est cette méthode d'interpolation, il faut d'abord définir les notions de plan factoriel, de facteur et de niveaux.

- **Facteur** : paramètre entrant en compte dans l'optimisation. Typiquement, les paramètres qui ont été choisis pour paramétrer le modèle « éléments finis » seront ceux qui seront appelés « facteurs » dans la méthode des plans d'expériences.
- **Niveaux** : Ce sont tout simplement les différentes valeurs que peuvent prendre les facteurs définis ci-dessus. Par exemple, si le paramètre x peut prendre les valeurs 6, 12 et 18, on dira que dans la méthode des plans d'expériences que le facteur x a trois niveaux.
- **Plan factoriel** : C'est l'espace dans lequel va s'effectuer l'optimisation. En effet, c'est le plan dans lequel on va retrouver tous les facteurs et tous leurs niveaux. Chaque expérience étant représentée par un point dans ce plan.

Pour illustrer ces trois définitions, nous allons représenter sur la Figure 1.2 l'exemple d'un plan factoriel à **trois** facteurs (x, y, z) et **deux** niveaux (+ ; -). Un tel plan est noté 2^3 , ce qui définit parfaitement le nombre d'expériences (nombre de points) à réaliser.

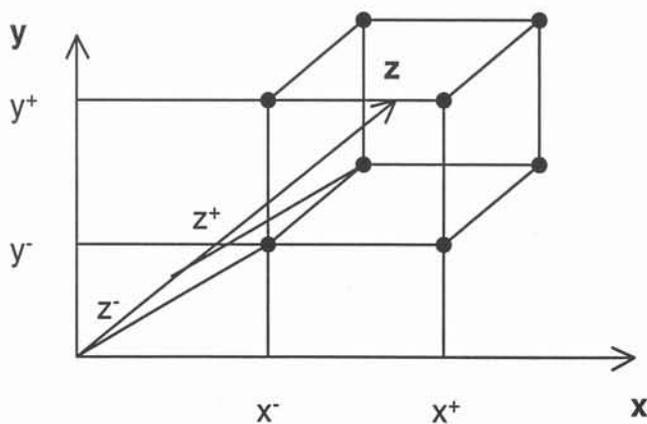


Figure 1.2 :Représentation d'un plan d'expériences à 3 facteurs et 2 niveaux.

Ensuite, on déduit des expériences un tableau des effets. En fonction des effets significatifs constatés sur chacune des variables, ou sur leurs interactions (on parle dans ce cas de contraste), on peut parfois se contenter d'un plan factoriel fractionnaire. Ceci permet de réduire le nombre de simulations nécessaires en perdant une quantité d'information sur certains effets, ce qui n'est pas forcément pénalisant. Cette méthodologie qui est résumée ici de façon schématique est remarquablement expliquée en détails avec des notions supplémentaires dans les références citées dans ce paragraphe et particulièrement dans [BRISSET-95].

1.C.2.2 Positionnement par rapport à cette approche.

L'optimisation qui est faite dans cette approche consiste à se déplacer intelligemment dans l'espace de dimension n (défini par le nombre de facteurs variables) sur l'hypersurface de dimension $n-1$ que l'on construit expérience par expérience (point par point). C'est-à-dire que l'on essaie de tendre le plus rapidement possible vers l'optimum en utilisant le moins d'expériences possible.

Il est vrai que la méthode des plans d'expériences a son intérêt lorsqu'elle est utilisée avec un modèle lent et qu'elle fait gagner du temps au concepteur dès qu'elle lui permet d'éviter les simulations inutiles. Elle a d'ailleurs été utilisée parce que le modèle la « réclamait » en quelque sorte. On peut constater que, dans ce cas, le modèle éléments finis a induit le choix de cette méthode d'interpolation.

Cependant, lorsque son modèle est rapide, le concepteur n'a pas besoin d'utiliser de telles méthodes. C'est l'idée qui est à la base de nos orientations en termes de modélisation. Nous pensons que le concepteur préfère utiliser un modèle analytique parce qu'il lui donnera des résultats rapidement. Ainsi, nous comptons, à chaque fois que cela nous sera possible, utiliser des modèles analytiques.

Même si nous savons que pour modéliser de manière suffisamment précise un moteur à réluctance variable, nous ne pourrions pas nous passer de l'utilisation, à un moment ou à un autre d'un modèle numérique, nous nous efforcerons, à chaque fois que nous le pourrions, de repousser ce moment le plus tard possible dans le processus de dimensionnement.

1.C.3 Avec un modèle semi analytique et une méthodologie d'optimisation de type gradient.

Lorsque nous avons travaillé sur le moteur à réluctance variable de type classique, nous avons à notre disposition un modèle déjà développé. Nous n'avons pas remis en cause ce modèle car c'est celui avec lequel travaillait le concepteur et qu'il le satisfaisait. Notre objectif était de lui fournir une aide au dimensionnement, sous forme de logiciel, par exemple.

Le problème qui s'est posé à nous est simplement le suivant : le modèle en question était semi analytique. Si il avait été analytique selon la définition donnée précédemment, cela n'aurait pas été un problème dans la mesure où nous développons au laboratoire un logiciel générique d'aide au dimensionnement de dispositifs modélisables analytiquement (PASCOSMA) [WURTZ-96].

L'idée que nous avons eue a été la suivante :

*Comme ce modèle ne nous aurait pas posé de problème si il avait été analytique, pourquoi ne pas essayer de le transformer de sorte à ce qu'il soit vu par la méthode d'optimisation **comme s'il était analytique**.*

Pour réussir, il faut que la partie semi analytique du modèle considéré soit dérivable. Nous verrons dans le chapitre 2 comment nous pouvons faire pour intégrer un modèle semi analytique dans une méthodologie d'optimisation de type gradient en s'appuyant sur des cas pratiques que nous avons résolus.

1.C.4 Avec deux modèles complémentaires.

Nous avons aussi travaillé sur un moteur à réluctance variable à motifs. Pour ce nouveau type de MRV, nous avons du réécrire un modèle.

Or, nous avons vu précédemment que, pour dimensionner un moteur à réluctance variable, on ne peut pas se passer de modèles numériques. Nous avons par ailleurs justifié pourquoi nous souhaitons utiliser le plus possible des modèles analytiques, particulièrement parce qu'ils nous facilitent le travail pour l'utilisation de méthodes de dimensionnement avec des algorithmes de type gradient.

Si nous voulons absolument utiliser un modèle et que la littérature nous en propose un autre, trois possibilités s'offrent à nous :

- On utilise seulement un modèle analytique et on court le risque de ne pas avoir de résultats suffisamment précis pour qu'ils servent un jour à un concepteur.
- On utilise seulement un modèle numérique et on oublie notre volonté d'offrir au concepteur une méthode de dimensionnement rapide.
- On utilise deux modèles : l'un analytique, l'autre numérique en essayant de tirer parti au mieux des avantages que chacun pourra nous apporter.

C'est cette troisième idée que nous avons suivie lors de notre travail sur le moteur à réluctance variable à motifs et que nous détaillerons dans le chapitre 3.

1.D Conclusion

Nous commençons par différencier la modélisation du dimensionnement. Ensuite, nous expliquons pourquoi nous préférons utiliser des modèles analytiques dérivables associés à des méthodes d'optimisation de type gradient. Notre approche du dimensionnement en est finalement la reformulation en un problème d'optimisation.

Nous proposons notre classification en trois catégories des modèles qui existent en électrotechnique. Nous donnons ainsi nos propres définitions des modèles analytique, semi analytique et numérique. Nous analysons ensuite les caractéristiques des différents modèles, principalement en termes de facilité de calcul des dérivées et de connaissances consommables par le concepteur. Enfin, on lui donne des éléments de choix des modèles pour leur utilisation en dimensionnement.

2 Les boîtes de calcul : Une méthodologie pour intégrer des modèles semi analytiques dans un processus d'optimisation de type gradient.

L'objectif de cette partie est de démontrer l'intérêt de la méthodologie des boîtes de calcul pour l'intégration générique des modèles semi-analytiques dans un processus d'optimisation de type gradient. Nous allons d'abord introduire les principales raisons de notre travail sur ce thème. Nous décrivons ensuite une méthodologie d'optimisation de type gradient (PASCOSMA) ainsi que les limites que nous y avons rencontrées dans la prise en compte des modèles semi analytiques.

Puis, nous proposerons un guide méthodologique pour gérer des modèles semi analytiques dans une méthodologie d'optimisation utilisant les gradients. Nous donnerons d'abord la définition de ce que nous appelons boîte de calcul en précisant son intérêt pour le dimensionnement. Nous expliquerons ensuite comment créer une boîte de calcul avec un modèle semi analytique. Nous illustrerons notre propos avec plusieurs exemples de boîtes de calcul réalisées et terminerons en donnant des exemples de boîtes de calcul qu'il pourrait être intéressant de réaliser.

Enfin, nous appliquerons notre méthodologie au dimensionnement d'un moteur à réluctance variable de type classique.

Pourquoi les boîtes de calcul ?

- La première approche de travail évoquée au chapitre précédent a été de ne pas remettre en cause le modèle semi analytique de MRV classique dont nous disposions. Nous voulions obtenir un processus de dimensionnement en utilisant ce modèle avec un processus d'optimisation basé sur un algorithme de type gradient [WURTZ-96]. Nous n'avons donc pas, en particulier, essayé de contourner les problèmes posés par les parties semi-analytiques du modèle en les remplaçant par d'autres, « plus analytiques ».
- Le processus d'optimisation de type gradient dont il est question fonctionne très bien dans le cadre de ses hypothèses. Parmi celles-ci, il faut que le modèle soit analytique, explicite et continûment dérivable. L'intérêt des boîtes de calcul est de proposer une méthodologie de travail en vue d'étendre le champ d'action de ce processus d'optimisation à des modèles semi-analytiques. Encore faudra-t-il être capable de bien définir le domaine de définition de telles boîtes...

2.A Méthodologie PASCOSMA et limites dans la prise en compte des modèles semi analytiques.

2.A.1 Présentation de la méthodologie PASCOSMA.

Au laboratoire d'électrotechnique de Grenoble, une méthodologie générique a été développée qui permet de générer automatiquement un logiciel d'optimisation sous contraintes à partir des équations du modèle analytique d'un système à dimensionner. Cette méthodologie s'appelle PASCOSMA [WURTZ-96]. Elle est composée de sept étapes que nous allons illustrer avec un exemple de modèle analytique très simple :

Saisie des équations du modèle analytique. Ces équations doivent permettre de relier explicitement , les unes aux autres, les grandeurs physiques intervenant dans le système. Ces grandeurs sont les paramètres de construction, les paramètres d'utilisation ainsi que les contraintes et les performances physiques qui en résultent. L'exemple simple que nous avons choisi est celui d'une résistance. Dans la méthodologie, on écrit des équations sous la forme suivante dans un fichier :

equ : $U = R \times I$
eqr : $R = \text{Rho} \times L / S$
eqp : $P = R \times I^2$
eqs : $S = \pi \times D^2 / 4$
eqv : $V = L \times S$

où R est la résistance,
U est la tension électrique appliquée,
I est le courant qui traverse la résistance,
Rho est la résistivité du matériau,
P est la puissance dissipée,
L est la longueur de la résistance,
S est la surface de celle-ci,
D est le diamètre,
et V est le volume total.

Recherche des paramètres d'entrée et de sortie du modèle. Dans cette phase, on classe l'ensemble des variables du modèle dans deux groupes distincts que nous appellerons les paramètres d'entrée et les paramètres de sortie. On réalise ce classement grâce à une procédure générique et automatisée. Les paramètres de sortie sont ceux des termes de gauche des équations. Les paramètres d'entrée sont ceux qui se trouvent au moins une fois dans le terme de droite d'une équation sans jamais être dans le terme de gauche. Chacune des variables du modèle est donc, après ce classement, une variable d'entrée ou une variable de sortie. Ainsi, dès que chacun des paramètres d'entrée a une valeur, tous les paramètres de sortie sont calculables puisqu'ils sont exprimables symboliquement en fonction des paramètres d'entrée par substitution.

Les paramètres de sortie, pour notre exemple, sont U, R, P, S et V et ceux d'entrée sont I, Rho, L et D.

Calcul symbolique et automatique des paramètres de sortie en fonction des paramètres d'entrée. Dans cette étape, les paramètres de sortie du modèle sont calculés symboliquement en fonction des paramètres d'entrée. Pour cela, on procède à une substitution symbolique de chaque paramètre de sortie par son expression en fonction des paramètres d'entrée dès qu'il est « à droite du signe = ».

Pour notre exemple, les équations que nous avons entrées initialement deviennent les suivantes :

$$\begin{aligned} \text{equ} : U &= 4 \times \text{Rho} \times L \times I / (\pi \times D^2) \\ \text{eqr} : R &= 4 \times \text{Rho} \times L / (\pi \times D^2) \\ \text{eqp} : P &= 4 \times \text{Rho} \times L / (\pi \times D^2) \times I^2 \\ \text{eqs} : S &= \pi \times D^2 / 4 \\ \text{eqv} : V &= L \times \pi \times D^2 / 4 \end{aligned}$$

Programmation automatique des expressions des paramètres de sortie en fonction des paramètres d'entrée (Génération du programme d'analyse). Les expressions symboliques des paramètres de sortie calculés à l'étape précédente sont programmées dans un langage informatique. Les lignes de programme du logiciel d'analyse sont ainsi programmées automatiquement.

Pour notre exemple, les lignes de programmes fortran générées automatiquement par le logiciel de calcul symbolique Macsyma [Macsyma] pour le calcul de R sont les suivantes :

```
SUBROUTINE R1(R3)
INCLUDE 'COMDDL.INC'
DOUBLE PRECISION R3
R3=(1.27323954473516d0*L*RHO)/D**2
RETURN
END
```

Calcul symbolique et automatique des dérivées partielles des paramètres de sortie en fonction des paramètres d'entrée. Ces dérivées sont calculées symboliquement et automatiquement par le logiciel Macsyma [Macsyma]. Ce logiciel est tout à fait approprié pour s'acquitter de cette tâche répétitive qui devient impossible pour un être humain dès que le modèle commence à contenir beaucoup de paramètres d'entrée et de sortie. Rappelons que pour un modèle qui contient Nb_Pe paramètres d'entrée et Nb_Ps paramètres de sortie, il faut calculer Nb_Pe * Nb_Ps dérivées.

Pour notre exemple, les dérivées symboliques de la résistance par rapport aux variables d'entrée du modèle sont les suivantes :

$$dR_{dRho} = 4 \times L / (\pi \times D^2)$$

$$dR_{dL} = 4 \times Rho / (\pi \times D^2)$$

$$dR_{dD} = -2 \times 4 \times Rho \times L / (\pi \times D^3)$$

$$dR_{dl} = 0$$

Programmation automatique des dérivées partielles des paramètres de sortie en fonction des paramètres d'entrée (Génération du programme de calcul de sensibilité). Les expressions symboliques des dérivées calculées lors de l'étape précédente sont programmées dans un langage informatique. Le fait que ces dérivées soient calculées par un programme évite les erreurs qui ne manqueraient pas d'arriver si un homme se chargeait de dériver (1^{ère} source d'erreur possible), puis de programmer ces dérivées (2^{ème} source d'erreur possible). Le logiciel de calcul de sensibilité est obtenu par programmation automatique.

Pour notre exemple, les programmes correspondant à la dérivée de R par rapport aux paramètres d'entrée sont les suivants :

```
SUBROUTINE R0RHO2(R0RHO3)
INCLUDE 'COMDDL.INC'
DOUBLE PRECISION R0RHO3
R0RHO3=(1.27323954473516d0*L)/D**2
RETURN
END
```

```
SUBROUTINE R0L2(R0L3)
INCLUDE 'COMDDL.INC'
DOUBLE PRECISION R0L3
R0L3=(1.27323954473516d0*RHO)/D**2
RETURN
END
```

```
SUBROUTINE R0I2(R0I3)
DOUBLE PRECISION R0I3
R0I3=0.0D0
RETURN
END
```

```
SUBROUTINE R0D2(R0D3)
INCLUDE 'COMDDL.INC'
DOUBLE PRECISION R0D3
R0D3=-((2.54647908947032d0*L*RHO)/D**3)
RETURN
END
```

Couplage du programme d'analyse et du programme de calcul de sensibilité avec un algorithme d'optimisation sous contraintes (Création du logiciel de dimensionnement automatique). On procède à une édition de liens des lignes de code du programme d'analyse et du programme de calcul de sensibilité avec l'algorithme d'optimisation sous contraintes. Dans ce logiciel, on trouve :

- la connaissance contenue dans le modèle, sous la forme des programmes d'analyse et de calcul de sensibilité.
- un processus d'exploration de l'espace des solutions pour un cahier des charges quelconque, représenté par l'algorithme d'optimisation sous contraintes.

Pour notre exemple, l'interface du logiciel de dimensionnement issu de notre modèle est présentée sur la Figure 2.1.

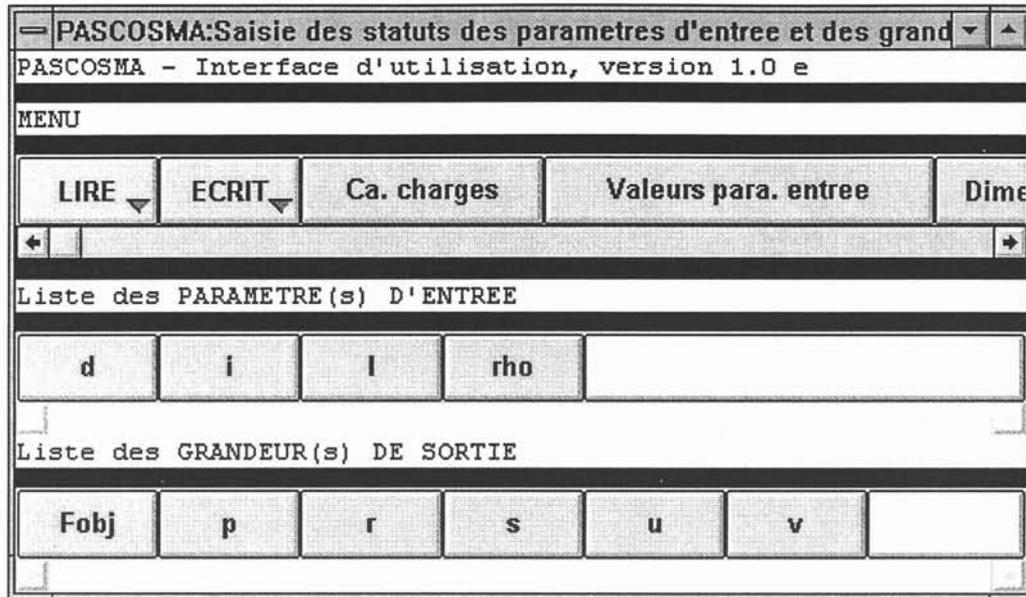


Figure 2.1: Interface graphique du logiciel de dimensionnement de notre exemple

On peut résumer sous forme schématique les sept étapes de génération automatique d'un logiciel de dimensionnement décrites précédemment comme cela est fait sur la Figure 2.2.

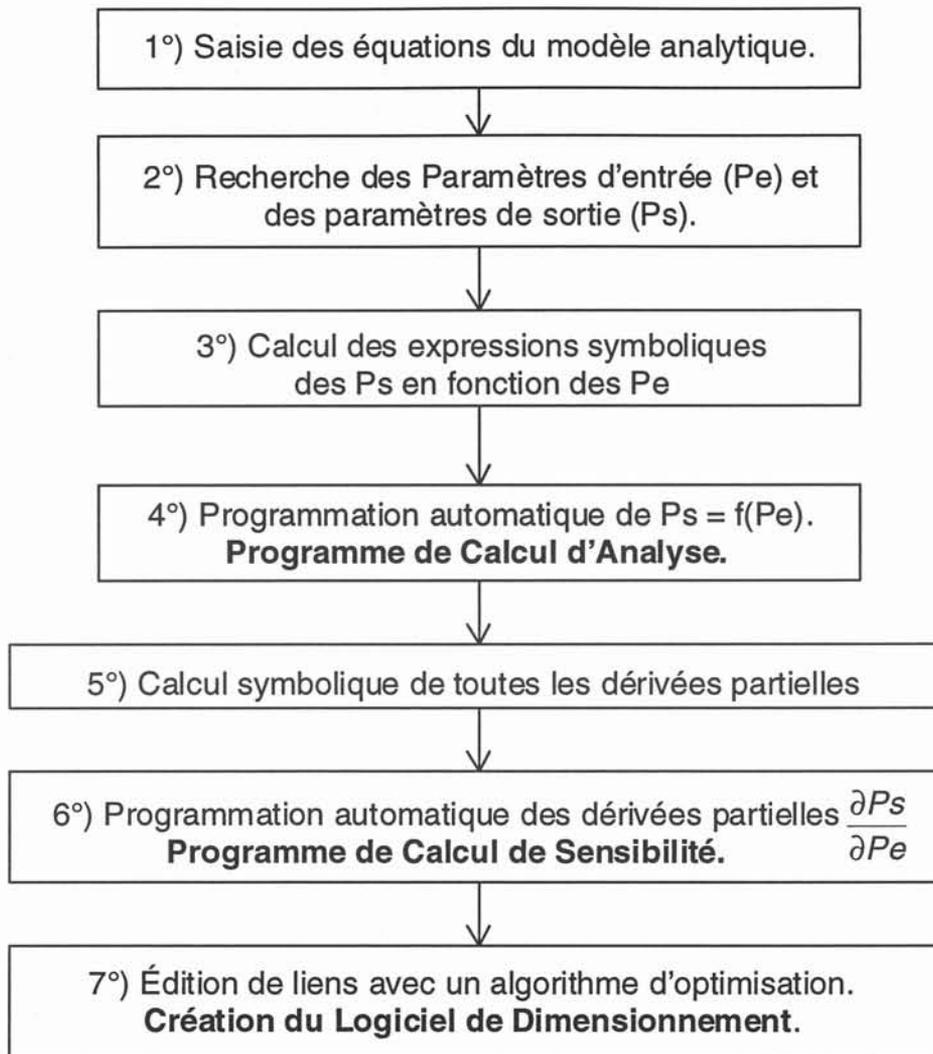


Figure 2.2 : les sept étapes de la méthodologie de génération automatique d'un logiciel de dimensionnement.

2.A.2 Limites de PASCOSMA dans la prise en compte des modèles semi analytiques.

La méthodologie qui vient d'être présentée fonctionne avec la condition que le modèle qui est saisi lors de la première étape soit un modèle analytique (donc explicite selon la définition donnée au chapitre 1). Cette méthodologie ayant déjà largement fait ses preuves avec son utilisation sur des modèles analytiques, nous avons voulu l'utiliser sur des modèles semi analytiques. C'est là que nous avons pu mettre en évidence quelques limites de cette méthodologie pour la prise en compte de ces modèles.

2.A.2.1 Les paramètres implicites.

La première limite est la prise en compte des paramètres implicites. Le modèle analytique qui doit être saisi ne doit pas comporter de variables implicites (c'est à dire une variable qui dépend d'elle même). Pour autant, même si le modèle contient une telle variable, il existe une parade qui utilise ce que nous offre le logiciel de dimensionnement en nous permettant de poser des contraintes d'égalité.

Soit V une variable implicite d'un modèle.

- Elle vérifie une équation du type $V = F(V)$, si on a pour elle une expression analytique explicite.
- L'équation $V - F(V) = 0$ ne connaît pas de racine exprimable sous forme symbolique.
- Comme on n'a pas le droit d'écrire l'équation sous la forme $V = F(V)$ dans la méthodologie, on introduit un paramètre CV que l'on exprime ainsi : $CV = V - F(V)$. Ensuite, il suffit de contraindre ce paramètre à 0 dans le logiciel de dimensionnement pour avoir ce que l'on veut.

Cette première limite n'est donc pas critique puisqu'il existe un moyen de contourner le problème.

C. Coutel a comparé différentes manières de résoudre ce problème et illustre la présence de paramètres implicites dans des modèles utilisés en électrotechnique avec un circuit magnétique comportant un aimant [COUTEL2-99] (Figure 2.3). Nous allons utiliser un de ces exemples pour illustrer notre méthodologie. Elle a montré que cette manière de faire n'est pas la plus efficace. Les performances se dégradent notamment lorsque le nombre de paramètres implicites est élevé [COUTEL3-99].

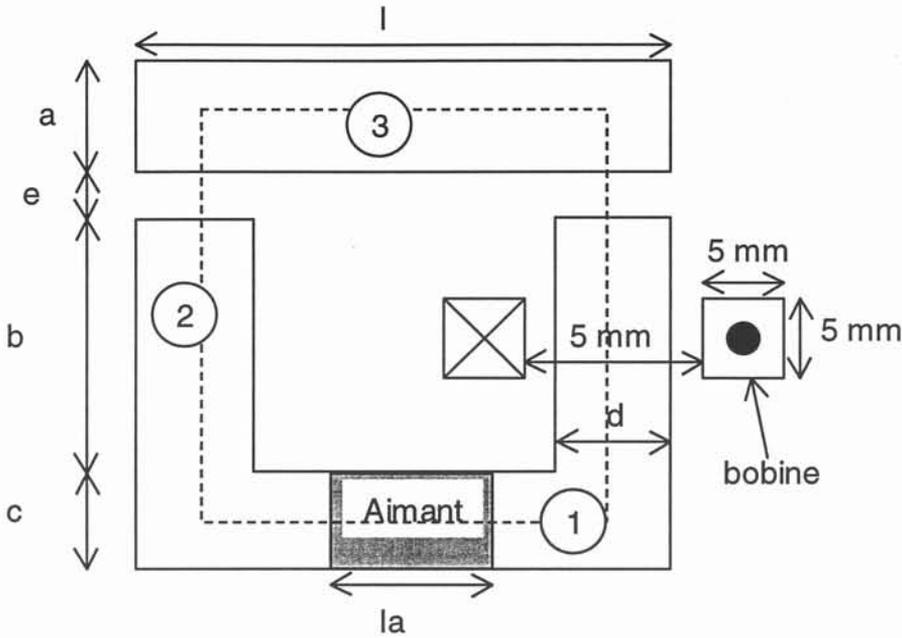


Figure 2.3 :Circuit magnétique comprenant un aimant

L'application du théorème d'Ampère pour ce circuit magnétique donne l'équation implicite suivante :

$$-(br - bam) \cdot \frac{la}{\mu_r \cdot \mu_0} + 2 \cdot e \cdot \frac{be}{\mu_0} + H(be) \cdot (2 \cdot b + c) + H\left(be \cdot \frac{d}{a}\right) \cdot (l - d) + H\left(be \cdot \frac{d}{c}\right) \cdot (l - la - d) = 0$$

- Où bpf l'induction dans la zone 1, (bpf = be * d / c)
- bcf l'induction de la zone 2, (bcf = be)
- bpm l'induction dans la zone 3, (bpm = be * d / a)
- br est l'induction rémanente de l'aimant, br = 1T,
- μ_r est la perméabilité relative de l'aimant, $\mu_r = 1$,

Le paramètre implicite à résoudre est be, l'induction dans l'entrefer. En raison de la non linéarité du matériau magnétique, approchée par la fonction :

$h(b) = 129,5 \times B + 76,1 \times B \times \exp(1.26 \times B^2)$, be ne peut pas être exprimé comme une fonction explicite de la, l, d, a, c, b et e et est donnée par une fonction implicite :

$$be = G(la, l, d, a, c, b, e)$$

2.A.2.2 Calcul des paramètres de sortie.

Maintenant, quand nous avons un modèle semi analytique dans lequel on est obligé **d'utiliser un algorithme** pour obtenir la valeur du paramètre de sortie en fonction des paramètres d'entrée, nous pouvons considérer l'utilisation de l'algorithme comme une simple équation. Ce calcul du paramètre de sortie S avec une équation ($S = f(V)$) où f est un algorithme) est possible si on a à notre disposition un programme informatique intégrable à la méthodologie générique traduisant cet algorithme.

Si nous ne l'avons pas, nous pouvons essayer de le programmer. On obtient donc une boîte comme celle décrite ci-dessous qui exprime S en fonction de V grâce à l'algorithme qu'elle contient.

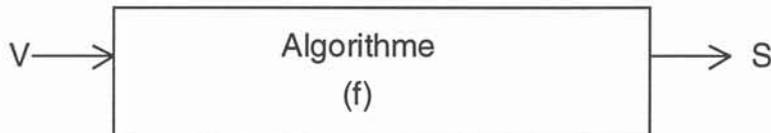


Figure 2.4 : calcul d'un paramètre de sortie avec un algorithme.

2.A.2.3 Calcul des dérivées des paramètres de sortie.

Les difficultés rencontrées avec les dérivées sont de deux types. Si elles sont explicites, elles ne sont pas forcément continues. De même, si elles sont continues, elles ne sont pas forcément exprimées de manière explicite.

2.A.2.3.1 Dérivées explicites mais pas continues

Les fonctions que nous utilisons dans nos modèles avec les algorithmes d'optimisation de type gradient (que nous privilégions) doivent être continues pour des raisons de convergence. Mais leurs dérivées peuvent ne pas être continues, lorsque nous sommes en présence d'un point anguleux. C'est par exemple le cas pour les fonctions continues affines par morceaux. Ceci n'est pas souhaitable pour des raisons de convergence.

La solution que nous proposons pour résoudre ce problème est, comme nous le verrons plus loin sur cet exemple, de rendre cette dérivée continue en

« arrondissant » le point anguleux. La méthode sera précisée plus loin, c'est pourquoi nous n'en dirons pas plus ici.

2.A.2.3.2 Dérivées continues mais pas explicites

Les racines d'un polynôme d'ordre quelconque, quant à elles, s'obtiennent grâce à un algorithme. Leurs dérivées par rapport aux racines ne sont pas fournies par un logiciel de calcul formel comme Macsyma. Dans ces cas là, on se retrouve face à la même situation que celle évoquée plus haut :

- soit on se passe des dérivées pour certains paramètres (ce qui nous interdit de les utiliser pour un dimensionnement).
- soit on les calcule numériquement, mais c'est justement ce que nous voulons éviter pour les raisons de convergence précisée au chapitre 1.
- soit on essaie de résoudre le problème pour pouvoir utiliser pleinement la méthodologie avec ce modèle. Dans ce cas, on crée une boîte contenant les programmes des dérivées que l'on a calculées comme cela est présenté sur la figure ci-dessous.

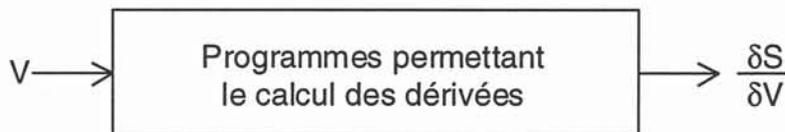


Figure 2.5 : Calcul de dérivées par programme.

2.A.2.4 Conclusion.

Pour conclure, dès que l'on est en présence d'un modèle semi analytique, il arrive inmanquablement qu'au moins un des deux problèmes présentés ci-dessus se pose. En conséquence, on ne peut pas mettre en marche la méthodologie décrite au paragraphe 2.A.1. parce qu'on ne se trouve plus dans le cadre de ses hypothèses. Elle réclame en effet un modèle analytique, et non semi analytique. Que faut-il alors faire pour pouvoir, malgré notre modèle semi analytique, utiliser quand même une méthodologie d'optimisation de type gradient ? C'est ce que nous allons présenter succinctement dans le paragraphe suivant. Le développement qui suivra

dans ce chapitre montrera en détail ce que nous proposons pour gérer les modèles semi analytiques.

2.A.3 Encapsulation du problème.

L'encapsulation des modèles semi analytiques est la solution que nous avons envisagée pour résoudre le problème que nous venons de poser. Nous allons présenter dans ce paragraphe ce que nous entendons par là. La méthodologie d'optimisation de type gradient telle que nous venons de la définir est composée de trois entités : *l'algorithme d'optimisation* fait appel à des programmes écrits dans un langage informatique, le *programme de calcul d'analyse* et le *programme de calcul de sensibilité*. Il se trouve que ces deux derniers sont générés automatiquement à partir des équations du modèle analytique.

Or, nous venons de voir que si cette génération automatique fonctionne parfaitement avec des modèles analytiques explicites, elle pose un problème pour la prise en compte de modèles semi analytiques.

L'algorithme d'optimisation, quant à lui, fait simplement appel à des programmes qui doivent renvoyer des valeurs :

- pour les paramètres de sortie.
- pour les dérivées des paramètres de sortie en fonction des paramètres d'entrée.

Nous nous proposons de créer des outils génériques permettant de prendre en compte des parties de modèles semi analytiques utilisées en électrotechnique. La généricité des outils développés permettra leur réutilisation facile grâce à une documentation associée par un utilisateur de la méthodologie générique qui se trouvera confronté au problème.

Nous avons donné dans ce paragraphe l'idée générale qui a guidé notre travail d'intégration des modèles semi analytiques à une méthodologie d'optimisation de type gradient : l'encapsulation. Nous allons la réaliser en écrivant nous mêmes, et à part de la méthodologie générique, les programmes permettant de calculer les paramètres de sortie et leurs dérivées par rapport aux paramètres d'entrée en fonction de ces derniers. Ainsi, nous pourrons insérer à une méthodologie d'optimisation de type gradient une partie de modèle semi analytique.

2.A.4 Outil générique

Nous venons de voir l'intérêt de l'encapsulation d'un modèle semi analytique dans des boîtes de calcul. Certaines parties semi analytiques reviennent souvent dans les modèles utilisés en électrotechnique. C'est pourquoi nous avons voulu créer des outils génériques permettant de créer automatiquement les boîtes de calcul adaptées à un problème semi analytique particulier. Ces outils pourraient être appelés générateurs de boîtes de calcul, parce qu'ils suivent le même principe que PASCOSMA pour la génération de ces boîtes. Nous utilisons une notion fondamentale qui permet l'extension de PASCOSMA aux boîtes de calcul qui est celle de **Fonction Extérieure**. Ces outils sont des générateurs de fonctions extérieures pour PASCOSMA.

2.A.4.1 Données nécessaires à la génération d'une boîte de calcul

Les données nécessaires à la génération d'une boîte de calcul sont :

- Les données composant l'interface d'entrée de la boîte de calcul
- Les données composant l'interface de sortie de la boîte de calcul
- Eventuellement une fonctionnelle si la boîte de calcul doit en contenir une. Nous appelons fonctionnelle une fonction qui est contenue dans la boîte de calcul et qui n'apparaît pas directement dans le calcul des paramètres de sortie. Typiquement, une fonctionnelle est la fonction avec laquelle on veut approcher un nuage de points avec la méthode des moindres carrés.

- Eventuellement des paramètres d'initialisation s'il y en a besoin. Ces paramètres d'initialisation servent de valeurs initiales aux paramètres de sortie implicites dont les algorithmes numériques ont besoin ou des bornes d'intervalles pour les sommes.

On donne ci-dessous l'exemple des données nécessaires à la génération d'une boîte de calcul de fonctions continues affines par morceaux.

Entrée.mac	
Interface d'entrée :	$(x_i, y_i)_{i \in \{1; \dots; n\}}$ $(\varepsilon_j)_{j \in \{2; \dots; n-1\}}$ x_{calc}
Interface de sortie :	y_{calc} $\left(\frac{\partial y_{calc}}{\partial x_i}, \frac{\partial y_{calc}}{\partial y_i} \right)_{i \in \{1; \dots; n\}}$ $\left(\frac{\partial y_{calc}}{\partial \varepsilon_j} \right)_{j \in \{2; \dots; n-1\}}$ $\frac{\partial y_{calc}}{\partial x_{calc}}$
Fonctionnelle :	Aucune
Paramètres d'initialisation :	Aucune

Figure 2.6 : Exemple de données nécessaires à l'outil générique de création de boîtes de calcul pour les fonctions continues affines par morceaux.

En donnant ces données à notre outil dans un fichier que nous appelons Entrée.mac, il est capable de nous générer les fichiers nécessaires à la création de notre boîte de calcul. Nous montrons sur la Figure 2.6 l'exemple des informations qu'il faut fournir au générateur pour qu'il crée une boîte de calcul comme celle qui est présentée Figure 2.10.

2.A.4.2 Fichiers générés par l'outil

Les fichiers qui sont générés par l'outil sont de deux types :

- Ceux qui réalisent le calcul des paramètres de sortie et de leurs dérivées partielles en fonction des paramètres d'entrée. Ceux-là réalisent vraiment la boîte de calcul au sens où nous l'entendons.
- Ceux qui permettent que la boîte de calcul soit assimilée par la méthodologie générique en tant que « simple équation ». Ceux-là sont indispensables pour créer un lien entre la boîte de calcul et l'environnement de dimensionnement qui l'utilise.

Ces deux types de fichiers peuvent être créés par l’outil générique dès qu’on lui a fourni les données qui lui sont nécessaires.

2.A.4.2.1 Fichiers réalisant la boîte de calcul

Ces fichiers sont des fichiers de programme. Ils permettent de calculer les valeurs de paramètres de sortie et celles des dérivées quelles que soient les valeurs fournies en entrée de la boîte. Pour que les boîtes de calcul que nous avons créées soient intégrables à la méthodologie PASCOSMA , nous les avons écrites en FORTRAN. Ainsi, elles seront homogènes avec les autres parties du code de calcul PASCOSMA.

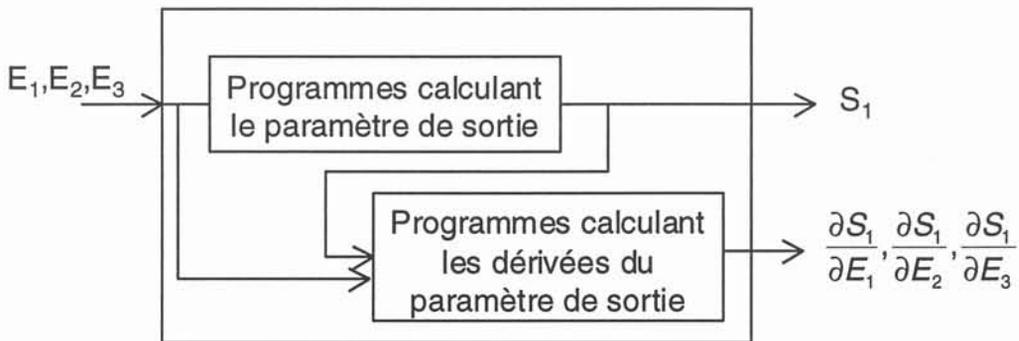


Figure 2.7 : Représentation Schématique d’une boîte de calcul à trois paramètres d’entrée et un paramètre de sortie.

```

Boite_de_Calcul.FOR
S1(E1,E2,E3)
dS1_dE1(E1,E2,E3)
dS1_dE2(E1,E2,E3)
dS1_dE3(E1,E2,E3)
    
```

Figure 2.8 : Programmes que devra contenir le fichier réalisant la boîte de calcul.

2.A.4.2.2 Fichiers d'assimilation (ou de liaison)

Ces fichiers sont ceux qui permettent à la boîte de calcul d'être « vue comme une équation » par la méthodologie générique. Ils sont donc indispensables. Contrairement à ceux qui font la boîte de calcul, ils dépendent de la méthodologie à laquelle nous voulons intégrer notre boîte. Cette méthodologie aura, quelle qu'elle soit, besoin de connaître les noms des programmes qu'il lui faudra appeler pour calculer tel paramètre de sortie ou telle dérivée.

Pour continuer l'exemple que nous avons donné au paragraphe précédent, nous devons déclarer l'existence de la fonction extérieure dans un fichier contenant les instructions suivantes :

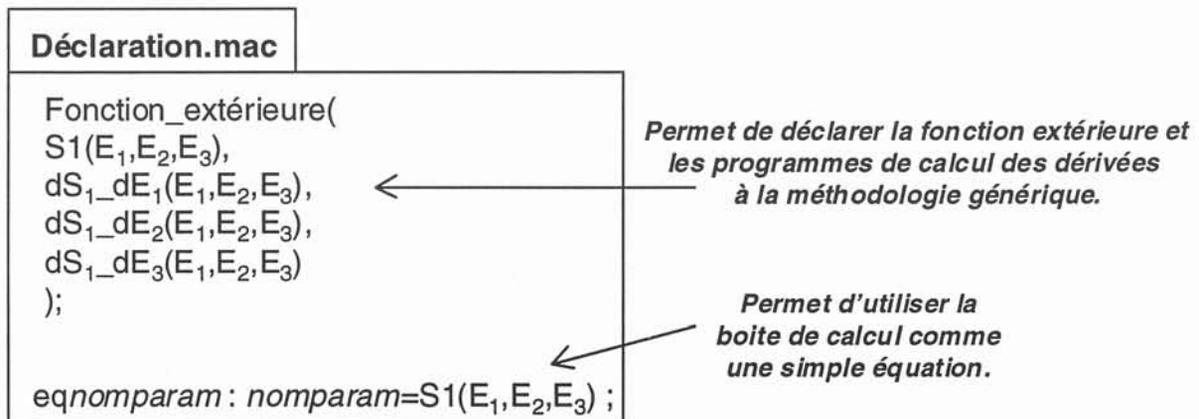


Figure 2.9 : Contenu d'un fichier d'assimilation (ou de liaison).

La première des deux instructions ci-dessous signifie à la méthodologie générique qu'elle devra appeler le programme $S_1(E_1, E_2, E_3)$ pour calculer la valeur du paramètre S_1 . La seconde indique quels sont les trois programmes qui correspondent aux dérivées de S_1 par rapport à chacune de ses variables d'entrée.

2.A.4.3 Fonctionnement général

On peut résumer le fonctionnement général d'un outil générique permettant de créer des boîtes de calcul par le schéma suivant.

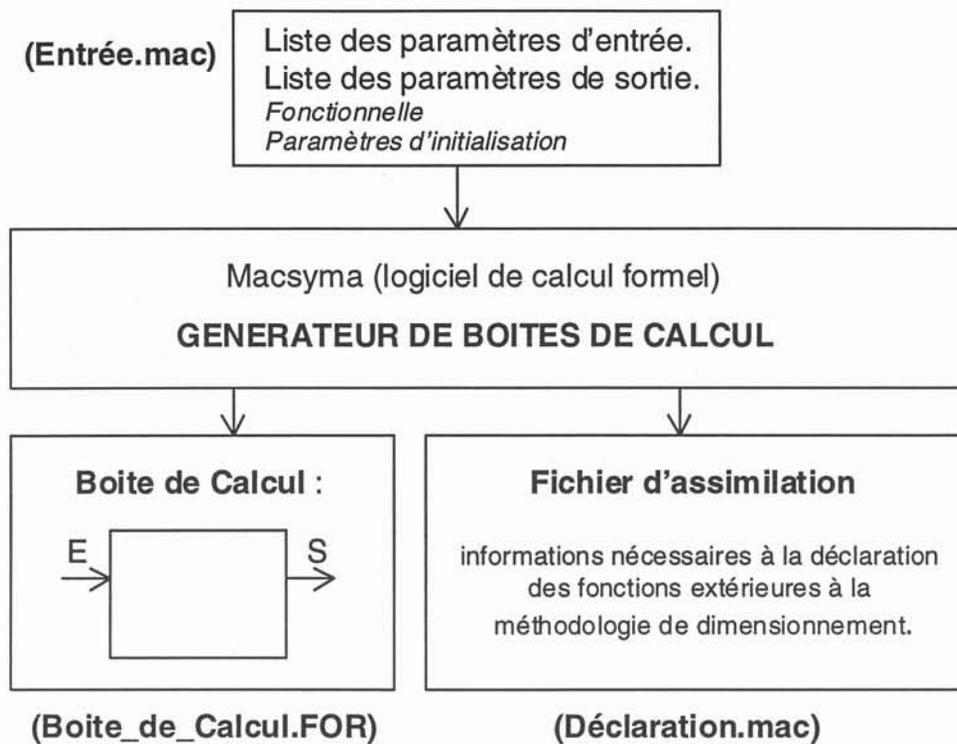


Figure 2.10 : Schéma général de fonctionnement d'un générateur de boîtes de calcul.

2.B Guide méthodologique proposé par les boîtes de calcul pour gérer des modèles semi analytiques.

La nécessité d'encapsuler la résolution du problème du calcul des paramètres de sortie d'un modèle semi analytique et de leurs dérivées a été présenté précédemment. Cette encapsulation nous a amené à baptiser cette structure « boîte de calcul » que nous commencerons par définir.

Ensuite, nous expliquerons comment faire pour créer une boîte de calcul avec un modèle semi analytique.

Nous détaillerons plus loin dans ce chapitre nos créations d'outils génériques permettant de créer des boîtes de calcul pour des problèmes que nous avons rencontrés sur le moteur à réluctance variable.

Enfin, nous nous terminerons cette partie en signalant quelques boîtes de calcul qu'il serait intéressant de créer pour un concepteur en électrotechnique.

2.B.1 Définition d'une boîte de calcul.

Dans ce paragraphe, nous allons définir ce que nous appelons une boîte de calcul, en la présentant comme une boîte noire, dont on va spécifier les interfaces d'entrée et de sortie. Nous expliquerons aussi pourquoi nous avons choisi ce formalisme et terminerons en précisant les intérêts d'une telle boîte pour le dimensionnement.

Avant tout, commençons par en donner une définition :

Définition : On appelle boîte de calcul une structure informatique calculant des paramètres de sortie en fonction de paramètres ainsi que les dérivées des paramètres de sortie par rapport aux paramètres d'entrée. Elle doit pouvoir être considérée comme une équation dans une méthodologie d'optimisation de type gradient.

2.B.1.1 Le concept de boîte noire.

Le concept de boîte noire est connu et utilisé depuis longtemps par les automaticiens. Il consiste à exprimer des variables de sortie en fonction de variables d'entrée avec un modèle à l'intérieur de la boîte sur lequel ils s'autorisent diverses opérations. La plus connue à mon sens est la transformée de Laplace, mais la transformée en Z existe aussi et d'autres doivent exister. Quelle que soit la transformée qu'il utilise, il travaille toujours avec les mêmes formalismes, ce qui lui permet d'appliquer les mêmes raisonnements lorsque deux modèles, même bien différents, sont en totale similitude à un endroit. On peut illustrer ce propos avec la démarche à suivre en automatique des procédés continus pour connaître le nombre d'intégrateurs qu'il faut insérer dans un système en boucle ouverte pour que la boucle fermée n'ait pas d'erreur. Ce raisonnement s'applique quel que soit le modèle et c'est bien le formalisme utilisé qui permet le raisonnement. Ce formalisme est le concept de boîte noire.

Nous avons construit nos boîtes de calcul en reprenant d'une certaine manière ce concept mais en l'appliquant au dimensionnement à l'aide d'algorithmes

d'optimisation de type gradient. La schématique habituelle que nous utilisons pour représenter une boîte de calcul est sans doute plus évocatrice (Figure 2.11).

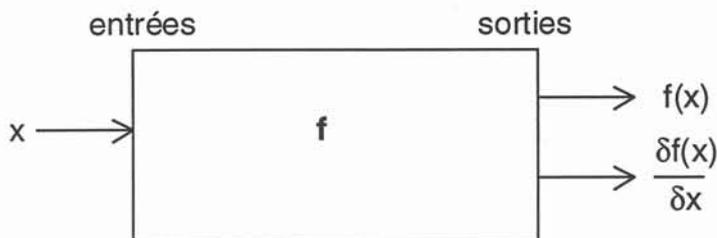


Figure 2.11 : Représentation schématique d'une boîte de calcul.

Toutes les boîtes de calcul sont conçues sur ce modèle. Elles ont donc l'avantage d'être toutes similaires entre elles et leur intégration au sein d'un processus d'optimisation de type gradient en sera très nettement facilitée.

De plus, cette représentation graphique est conforme à la définition car on voit bien apparaître les interfaces de la boîte de calcul que nous allons définir à présent.

2.B.1.2 Définition des interfaces.

2.B.1.2.1 Interface d'entrée.

L'interface d'entrée d'une boîte de calcul est constituée des paramètres d'entrée du modèle semi analytique que l'on veut mettre dans celle-ci. Par exemple, on peut définir une fonction continue affine par morceaux avec ses points anguleux, ou un polynôme par ses coefficients.

2.B.1.2.2 Interface de sortie.

L'interface de sortie d'une boîte de calcul est constituée des paramètres que l'on cherche à calculer avec celle-ci et des dérivées de chacun de ces paramètres de sortie par rapport à chacun des paramètres donnés en interface d'entrée. Par exemple, pour une fonction continue affine par morceaux, le paramètre de sortie est la valeur de la fonction au point de calcul donné en entrée. On doit donc donner sa dérivée par rapport à celui-ci, mais aussi par rapport à chacune des abscisses, des

ordonnées et des demi intervalles des points anguleux de la fonction qui peuvent, a priori, varier eux aussi au cours du processus d'optimisation (Figure 2.12).

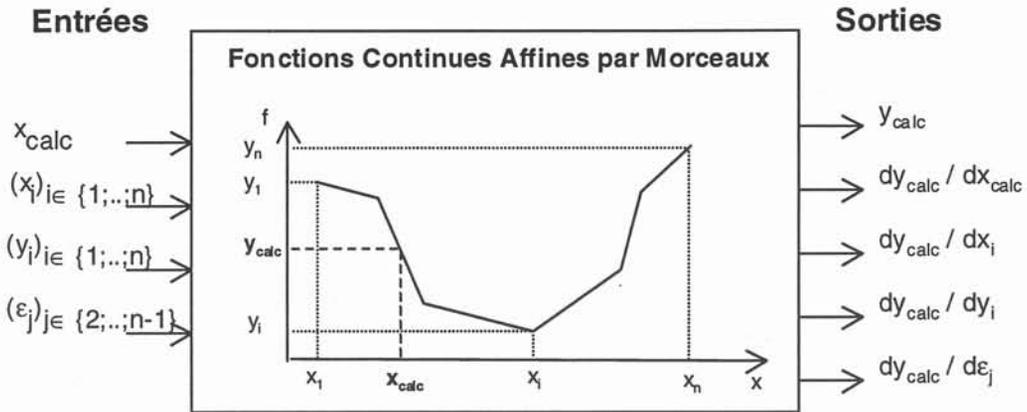


Figure 2.12 : Interfaces d'entrée et de sortie d'une boîte de calcul de fonctions continues affines par morceaux.

Pour un polynôme, les paramètres de sortie sont ses racines. On doit donc fournir également les dérivées de chacune des racines par rapport à chacun des coefficients.

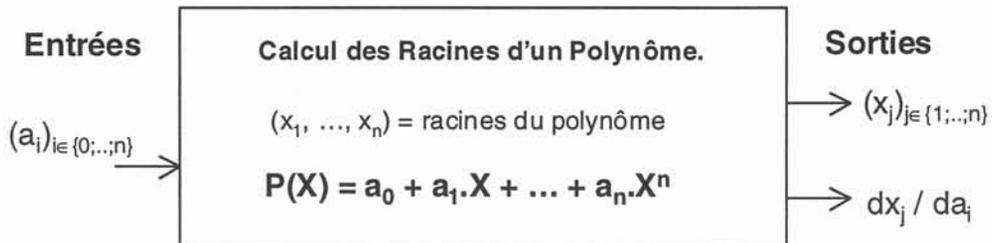


Figure 2.13 : Interfaces d'une boîte de calcul de racines de polynômes.

Nous voulons que la boîte de calcul soit équivalente à une équation dans une méthodologie d'optimisation de type gradient. C'est pour cette raison que nous devons fournir en sortie de notre boîte les valeurs des paramètres de sortie et de leurs dérivées.

2.B.1.3 Pourquoi ce formalisme ?

Nous pensons que ce formalisme peut nous être utile pour le dimensionnement. En effet, ce qui nous intéresse en dimensionnement n'est pas tant le modèle qui est utilisé mais plus ce que l'on peut en tirer pour trouver une solution (optimale si possible !) à un cahier des charges exprimé sous la forme la plus générique possible.

Nous avons défini les interfaces présentées ci dessus car nous avons constaté que ces informations sont celles que nous traitons avec un modèle analytique. En effet, ce que nous faisons d'une équation saisie sous la forme $U = R \times I$ est de la transformer en trois programmes fortran (1 + 2) correspondants, l'un au calcul de U quand on donne des valeurs à R et à I et les deux autres correspondants aux dérivées (Figure 2.14).

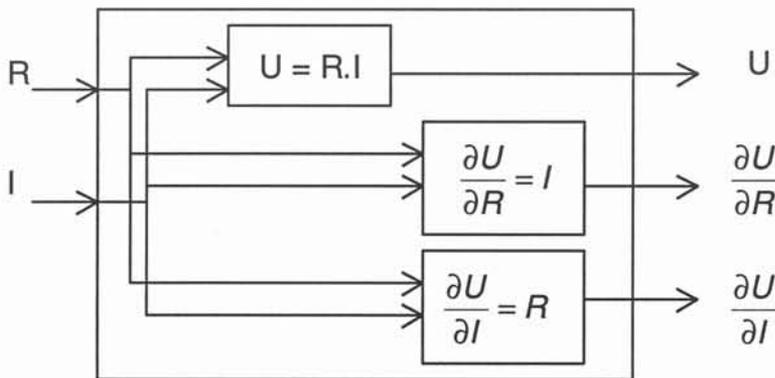


Figure 2.14 : Utilisation d'une équation analytique dans une méthodologie d'optimisation de type gradient.

La seule différence avec les autres modèles est que, pour les modèles analytiques, on est capable de créer les boîtes de calcul automatiquement sans intervention humaine. Pour créer des boîtes de calcul avec des modèles semi analytiques, on ne peut pas, pour l'instant, se passer de l'intervention humaine. C'est dans l'optique de s'en passer un jour que nous avons créé des outils génériques pour certains problèmes rencontrés en électrotechnique.

2.B.1.4 Intérêts pour le dimensionnement.

Avec ces boîtes de calcul, dès que l'on a les programmes permettant de calculer les valeurs des paramètres de sortie et leurs dérivées par rapport aux

paramètres d'entrée, il suffit d'y faire appel au niveau de la méthodologie générique pour que le modèle semi analytique soit traité comme une équation. On sait donc alors construire le programme d'analyse et le programme de calcul de sensibilité, comme nous l'avons montré dans le paragraphe 2.A.1. présentant la méthodologie PASCOSMA.

D'une manière générale, le fait de définir un cadre général de travail tel que les boîtes de calcul permet de faire travailler plusieurs personnes en parallèle sur un même projet. On peut imaginer des systèmes très complexes que l'on arrive à modéliser avec beaucoup de parties semi analytiques différentes. Pour réaliser un environnement de dimensionnement d'un tel dispositif, on pourrait faire travailler plusieurs personnes en parallèle en leur donnant à développer à chacune une boîte de calcul dont on aurait défini les interfaces au niveau du dessus. Dans ce cas, le projet serait un environnement de dimensionnement basé sur les boîtes de calcul.

2.B.1.4.1 Calcul d'analyse et Calcul de Sensibilité.

Comme nous l'avons vu précédemment, une boîte de calcul peut facilement s'intégrer à une méthodologie effectuant un calcul d'analyse seul car elle permet le calcul des paramètres de sortie. Si l'utilisateur ne souhaite pas utiliser les dérivées, il n'est pas obligé de le faire.

De même, les boîtes de calcul permettent le calcul des dérivées. Nous avons fait cela dans le but de les intégrer dans une méthodologie d'optimisation de type gradient, mais un autre utilisateur qui aurait besoin des dérivées du modèle pour les utiliser à d'autres fins peut également se servir de notre méthodologie de création de boîtes de calcul pour les créer conformes à son environnement. Il pourrait même utiliser les nôtres si ce dernier peut intégrer des programmes écrits en fortran.

2.B.1.4.2 Positionnement par rapport à PASCOSMA.

La méthodologie PASCOSMA permet de prendre en compte les modèles analytiques. Le formalisme des boîtes de calcul nous permet d'en élargir le champ d'action aux modèles semi analytiques pour lesquels on connaît un moyen de calculer les paramètres de sortie et leurs dérivées. Pour ceux que nous avons

rencontrés dans la modélisation des moteurs à réluctance variable, nous avons même créé des outils génériques permettant de créer automatiquement les boîtes de calcul intégrables à l'environnement existant traitant ces problèmes.

Pour présenter notre positionnement par rapport à PASCOSMA de façon schématique, nous pouvons regarder la Figure 2.15 qui présente les modèles analytiques totalement pris en charge de manière automatique par PASCOSMA, les modèles semi analytiques pour lesquels nous avons créé un outil générique permettant la création de boîtes de calcul et ceux pour lesquels il n'existe pas encore d'outil.

On voit bien avec ce schéma l'intérêt du formalisme proposé par les boîtes de calcul qui permettent, d'une manière générale, d'étendre le champ d'action de la méthodologie initiale aux modèles semi analytiques. Le développement d'outils génériques résolvant des problèmes particuliers permet simplement une utilisation plus conviviale de cette méthodologie.

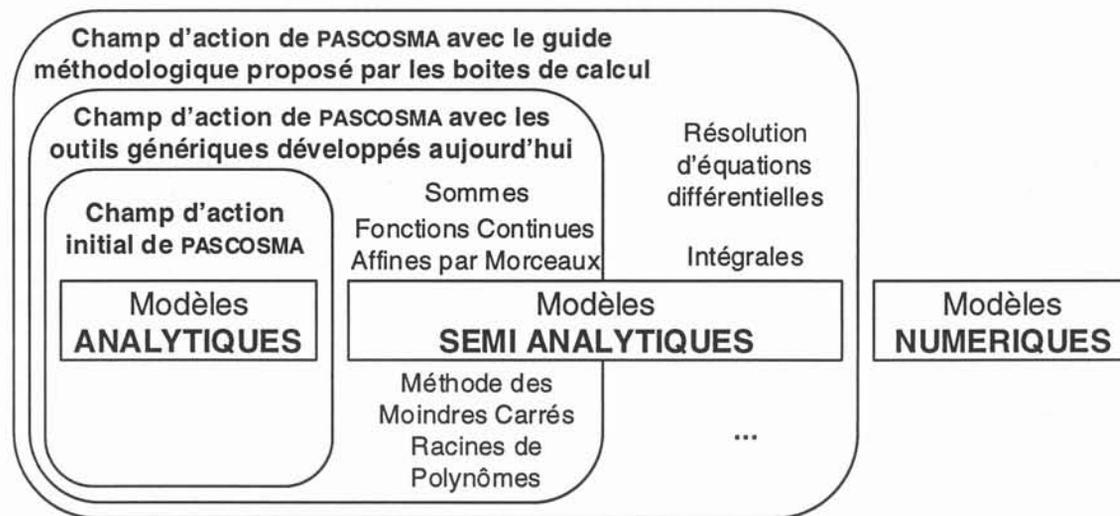


Figure 2.15 : Positionnement schématique par rapport à PASCOSMA.

2.B.1.4.3 Réintégration à PASCOSMA.

Les boîtes de calcul que nous avons créées l'ont toutes été pour être intégrées dans la méthodologie PASCOSMA. Cette nécessité d'intégration à PASCOSMA pour les valider nous a obligés à définir des liens entre la méthodologie et les boîtes de

calcul. Le problème particulier qui s'est posé est que PASCOSMA ne peut prendre en compte que des réels.

En effet, dans la présentation de la méthodologie PASCOSMA [WURTZ-96], le modèle analytique est utilisé de la manière présentée à gauche sur la Figure 2.16.

C'est à dire que chaque rond représenté sur la figure est un réel et les segments qui relient les paramètres de sortie aux paramètres d'entrée expriment la dépendance des uns par rapport aux autres via une équation.

Ce que nous permettons avec nos boîtes de calcul est la prise en compte de dépendances « non automatiquement dérivables » des paramètres de sortie en fonction des paramètres d'entrée. C'est ce qui est représenté par la boîtes de calcul insérées dans le même modèle dans la figure de droite.

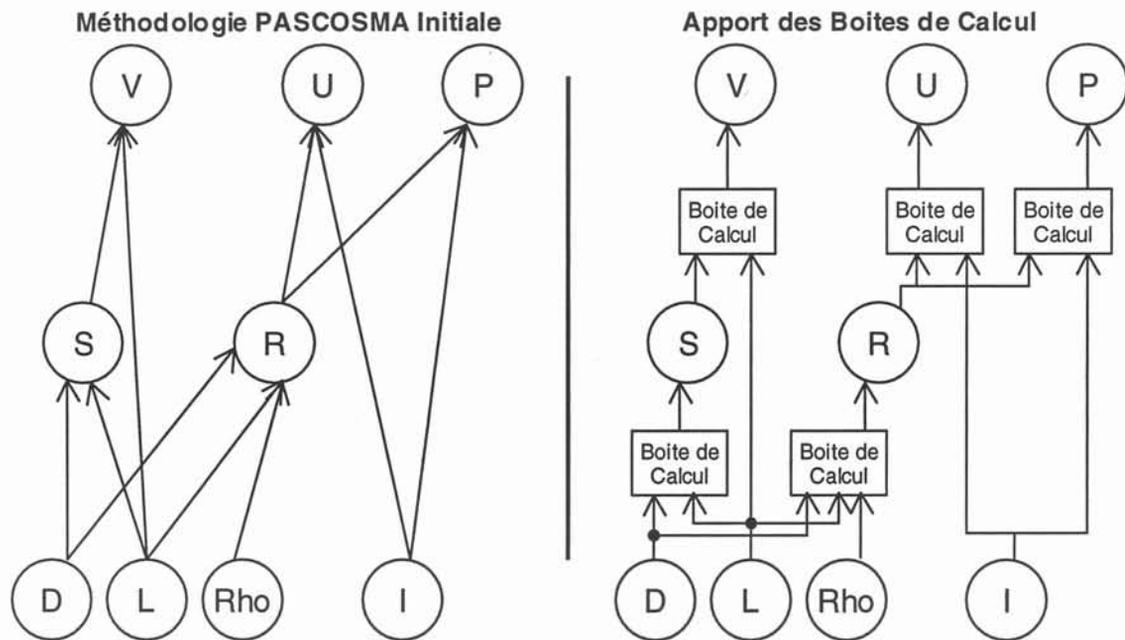


Figure 2.16 : Utilisation de Modèle dans la méthodologie PASCOSMA initiale.

Ainsi, lorsque la boîte de calcul que nous développons renvoie un tableaux de dimension n , il nous faut le transformer en les n réels qu'il contient. Ces n réels seront des variables que prendra en compte la méthodologie générique. Il en va de même pour les nombres complexes. Un nombre complexe, si on tient à l'utiliser dans un modèle PASCOSMA, devra être ramené à deux réels. On choisira donc selon les cas l'association Module et Argument ou Partie Réelle et Partie Imaginaire. C'est ce que nous avons dû faire, par exemple, lorsque nous avons crée l'outil générique

nous permettant d'obtenir les boîtes de calcul de racines de polynômes. Nous avons alors fait le choix (évident ici) d'exprimer les racines qui sont toutes a priori complexes pour l'algorithme en partie réelle et partie imaginaire.

Le même type de problème se pose aussi pour les paramètres d'entrée. Par exemple, pour les boîtes de calcul de sommes infinies, il faut transformer les entiers données comme bornes de l'intervalle de somme en « vrais entiers ». En effet, ils sont considérés comme des réels dans la méthodologie générique. Il en serait de même pour des entiers renvoyés comme des variables de sortie d'une boîte de calcul. Il faudrait les transformer en réels pour qu'ils deviennent « assimilables » pour la méthodologie PASCOSMA.

2.B.2 Comment créer une boîte de calcul avec un modèle semi analytique ?

Nous ne traiterons sans doute jamais de manière exhaustive toutes les parties semi analytiques que nous pourrions rencontrer dans les modèles. C'est pourquoi nous allons donner au lecteur dans les paragraphes qui suivent des éléments pour qu'il puisse créer lui même la boîte de calcul qui l'intéresse. Nous lui indiquerons en particulier quels sont les problèmes que nous avons identifiés et quelles sont les solutions que nous proposons pour chaque problème. Nous donnerons enfin une idée de l'implantation informatique que nous envisageons.

2.B.2.1 Problèmes identifiés.

Pour pouvoir utiliser avec le plus de chances de succès possible un processus de dimensionnement utilisant des **algorithmes de type gradient avec un modèle analytique** (ou une partie de modèle), il faut que celui-ci (celle-ci) soit calculable et continûment dérivable.

2.B.2.1.1 Sur les paramètres de sortie

Il faut qu'ils soient calculables car dès que l'on donne des valeurs aux paramètres d'entrée, il faut que l'on puisse calculer les valeurs des paramètres de sortie.

Pour insérer une partie de modèle dans une boîte de calcul, on devra donc être capable de donner des valeurs aux paramètres de sortie pour chaque jeu de valeurs donné aux paramètres d'entrées, par un moyen qu'il nous appartiendra de trouver.

2.B.2.1.2 Sur les dérivées

Il faut aussi que les paramètres de sortie soient **continûment dérivables** pour que, en chaque point où se trouve l'algorithme d'optimisation de type gradient, il ait une information cohérente qu'il soit capable d'interpréter. C'est l'algorithme d'optimisation de type gradient qui nous impose la continuité de la dérivée du modèle.

Il faudra donc, si on veut utiliser l'algorithme d'optimisation avec notre boîte de calcul, que le modèle semi analytique que l'on va « camoufler » soit au moins **dérivable** (on verra que l'on peut proposer des solutions s'il ne l'est pas « continûment »).

2.B.2.2 Solution sur les paramètres de sortie.

2.B.2.2.1 Problème implicite.

La plupart des modèles que les concepteurs utilisent en électrotechnique sont des équations analytiques comme les lois d'Ohm, de Faraday ou de Kirchhoff par exemple. Elles ont le grand mérite d'être très simples et pédagogiques. C'est sans doute pour cela que les concepteurs en électrotechnique aiment à se ramener à des schémas électriques équivalents très simplifiés mais qui, pour autant, rendent bien compte des principaux phénomènes physiques mis en jeu (par exemple : schéma équivalent d'un moteur asynchrone).

Malgré tout, certaines parties de modèles utilisés en électrotechnique demeurent implicites (c'est le cas pour la méthode des moindres carrés que l'on utilise pour approcher des courbes de saturation ou encore pour l'exemple que nous avons présenté Figure 2.3).

Pour résoudre les problèmes que pose une partie de modèle implicite, un outil générique a été développé par C. Coutel [COUTEL-99]. Il permet de prendre en compte les systèmes d'équations analytiques implicites sous la forme :

$$\begin{cases} g_1(e_1, \dots, e_n, s_1, \dots, s_p) = 0 \\ \dots \\ g_p(e_1, \dots, e_n, s_1, \dots, s_p) = 0 \end{cases}$$

avec (e_1, \dots, e_n) les paramètres d'entrée

et (s_1, \dots, s_n) les paramètres des sortie

Grâce à l'utilisation d'un algorithme numérique, on obtient les valeurs des paramètres de sortie. L'application du théorème des fonctions implicites nous permet d'exprimer les dérivées symboliques en fonction des valeurs obtenues pour les paramètres de sortie.

Le calcul des dérivées se fait de la manière suivante. Lorsque l'on veut calculer $\delta s_i / \delta e_j$, on effectue la dérivation de tous les g_i , $i \in \{1, \dots, p\}$ par rapport à e_j . Ces expressions sont programmables automatiquement à partir du moment où l'on dispose des expressions des g_i .

$$\frac{dg_i}{de_j}(e_1, \dots, e_n, s_1, \dots, s_p) = 0 = \frac{\partial g_i}{\partial e_j} + \sum_{k=1}^p \frac{\partial g_i}{\partial s_k} \cdot \frac{\partial s_k}{\partial e_j}$$

On calcule donc en même temps toutes les dérivées de tous les $\delta s_k / \delta e_j$, avec $k \in \{1, \dots, p\}$ en inversant la matrice du jacobien $\delta g_i / \delta s_k$.

La boîte de calcul équivalente qui résulte de tout ce travail a l'interface suivante :

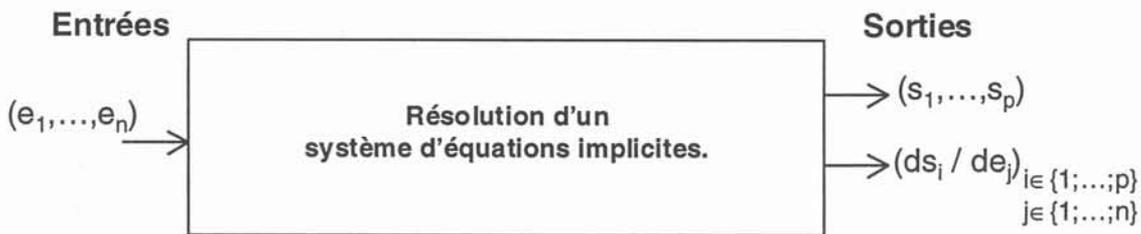


Figure 2.17 : Interface de la boîte de calcul des systèmes d'équations implicites.

Dans ce cas, on avait un algorithme à notre disposition, et on a créé un outil générique pour résoudre ce problème qui n'a pas manqué de se poser plusieurs fois à nous. Selon la difficulté du problème que l'utilisateur aura à résoudre, il lui est possible de développer lui même un algorithme à sa convenance.

2.B.2.2.2 *Modèle explicite sans formulation simple.*

Lorsque le modèle est explicite, la réalisation d'un programme permettant d'obtenir les paramètres de sortie doit être plus facile que lorsqu'il est implicite. En effet, il n'y a pas besoin de faire appel à des méthodes numériques pour résoudre le problème. Cependant, l'absence de formulation simple est un problème pour la prise en compte dans un modèle analytique. Le logiciel de calcul formel ne peut par exemple pas prendre en compte les fonctions continues affines par morceaux à cause de leur écriture difficile. Il nous faut dans ce cas définir les invariants qui caractérisent ces fonctions. La méthode des boîtes de calcul permet, une fois ces invariants définis, d'automatiser le travail de création des dérivées partielles et de programmation des formules. Nous verrons comment notre guide méthodologique permet de prendre en compte cette partie de modèle plus loin.

2.B.2.3 Solution sur la dérivée.

2.B.2.3.1 *Dérivée non continue.*

La non-continuité de la dérivée peut amener des défauts de convergence d'optimisation. En conséquence, il faut se débrouiller pour cette dérivée, non continue au départ, le devienne. Pour cela, on peut imaginer plusieurs méthodes pour obliger la dérivée à être continue. Celle que nous proposons nous paraît la plus simple et sera présentée dans le paragraphe concernant les fonctions continues affines par morceaux. Elle consiste à introduire une variable supplémentaire et à l'utiliser de sorte à obtenir au point anguleux un raccord de classe C^1 .

2.B.2.3.2 *Dérivée non explicite.*

Parfois les dérivées ne peuvent s'exprimer qu'une fois que les valeurs des paramètres de sortie sont calculées. C'est souvent le cas lorsque le problème est implicite. On attend alors que l'algorithme donne ses valeurs puis on les utilise dans des formules bien souvent génériques. Elles sont donc codées « en dur » dans les programmes informatiques. On peut schématiser cela comme présenté ci-dessous :

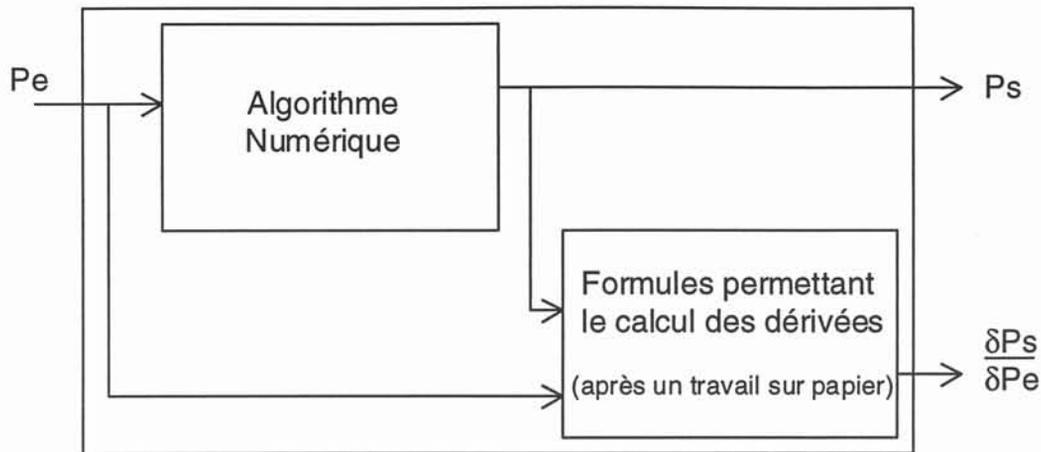


Figure 2.18 : Boîte de Calcul dont le calcul des dérivées dépend des valeurs des paramètres de sortie.

Pour des parties de modèles comme le calcul des racines d'un polynôme en fonction de ses coefficients, le calcul des dérivées des racines par rapport aux coefficients dépend de l'ordre de multiplicité de la racine. Ici, on se trouve donc non seulement dans le cas où le calcul de la dérivée dépend du paramètre de sortie qu'elle caractérise, mais aussi dans le cas où la formule utilisée pour le calcul dépend de l'ensemble des paramètres de sortie.

2.B.2.4 Implantation.

Lorsque l'on veut créer une boîte de calcul, il faut avant tout la créer « sur le papier ». C'est à dire qu'il faut être capable de dire comment on va faire pour calculer les $(Nb_Pe+1)*(Nb_Ps)$ valeurs que l'on doit calculer pour le problème que l'on se pose. En effet, on va calculer Nb_Ps valeurs de paramètres de sortie ainsi que $Nb_Pe * Nb_Ps$ dérivées.

Dès que l'on a répondu aux deux questions principales, qui sont : « Comment vais-je calculer les Ps ? » et « Comment vais-je calculer les dPs/dPe ? », l'implantation informatique de la boîte de calcul ne devient plus qu'une question de temps de développement informatique (qui peut quand même être assez long).

Une **difficulté supplémentaire** se pose si on veut créer un **outil générique** permettant de programmer automatiquement les boîtes de calcul répondant à un certain type de problème.

Des exemples d'implantation informatique sont donnés dans le paragraphe suivant avec des outils qui ont été réalisés.

2.B.3 Exemples de boîtes de calcul réalisées.

2.B.3.1 Fonctions continues affines par morceaux.

2.B.3.1.1 Position du problème.

On veut prendre en compte dans un modèle une fonction f continue, affine par morceaux (Figure 2.19). On s'autorise à arrondir plus ou moins les points anguleux de cette fonction en fonction du point considéré. Rappelons ici qu'il est préférable de s'autoriser un rayon de courbure significatif au niveau de chaque point pour que l'algorithme d'optimisation ait un meilleur fonctionnement.

On envisage l'arrondissement de la fonction au voisinage de ses points anguleux grâce à une **approximation quadratique réglable**. L'avantage de cette méthode est qu'elle transforme de manière simple une fonction de classe C^0 (Figure 2.12) en une fonction de classe C^1 .

Définition de la fonction :

Comme la fonction considérée est continue, il suffit de donner en entrée de la boîte de calcul les abscisses et les ordonnées de ses points anguleux $(x_i$ et $f(x_i)$, pour $i \in \{1, \dots, n\}$).

De plus, comme on veut que notre approximation soit **réglable**, il faut également donner en entrée de la boîte de calcul les valeurs des demi intervalles ε_i que l'on choisit de donner au voisinage de chaque x_i (ε_i pour $i \in \{2, \dots, n-1\}$). On arrondira ainsi plus ou moins notre point anguleux en fonction de la valeur que l'on donnera à ce paramètre. Il n'y a pas besoin de donner ε_1 ni ε_n car l'approximation de la fonction sera le premier morceau de droite entre x_1 et $x_2 - \varepsilon_2$ et sera le dernier morceau de droite entre $x_{n-1} + \varepsilon_{n-1}$ et x_n . Nous verrons même plus loin que cette définition sera prolongée au dehors du domaine de définition de la fonction pour d'autres raisons.

Le schéma suivant caractérise l'interface d'entrée de la boîte noire que l'on envisage de définir pour introduire une fonction continue affine par morceaux dans une méthodologie d'optimisation de type gradient :

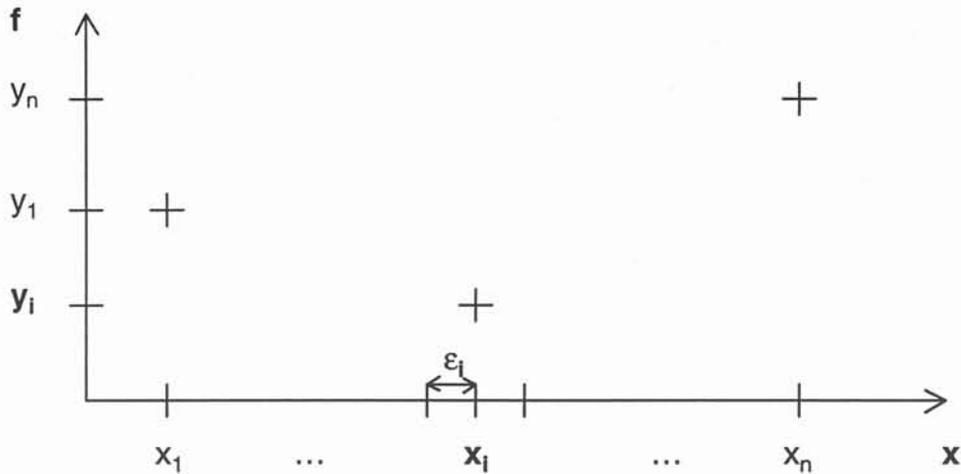


Figure 2.19 : Définition d'une fonction continue affine par morceaux.

On donne donc à l'entrée de la boîte de calcul les valeurs de x et celle des x_i , $f(x_i)$ et ε_i . On récupère en sortie, suivant la valeur de x donnée en entrée, les valeurs de $f(x)$ et de ses dérivées partielles par rapport aux paramètres d'entrée.

2.B.3.1.2 Type de problème posé.

Les deux types de problèmes que pose la réalisation d'une telle boîte de calcul sont les suivants.

- Les fonctions continues affines par morceaux sont explicites mais il n'existe pas de formule analytique simple pour les définir. Les formules sont simples si on s'autorise à utiliser le conditionnel « si ».
- De plus, les fonctions continues affines par morceaux ne sont pas continûment dérivables. Elles le sont presque partout au sens mathématique du terme. Les seuls points en lesquels elles ne sont pas dérivables sont les points anguleux (leur nombre est bien dénombrable et correspond au nombre de valeurs x_i et y_i).

2.B.3.1.3 Solution au problème.

Nous allons expliquer dans un premier temps la méthode que nous proposons pour résoudre le problème théorique, puis nous donnerons une idée de la manière dont nous avons programmé notre outil générique.

La méthode que nous proposons pour résoudre le problème d'intégration d'une fonction continue affine par morceaux à un processus d'optimisation de type gradient est d'arrondir les points anguleux de cette fonction grâce à une approximation quadratique réglable. Pour nous autoriser un rayon de courbure suffisant au niveau de ces points, nous définissons en plus de la fonction (définie avec les coordonnées de ses points anguleux $(x_i, y_i)_{i \in \{1; \dots; n\}}$) les demi intervalles $(\varepsilon_j)_{j \in \{2; \dots; n-1\}}$ que nous associons aux points anguleux.

Ainsi, sur les intervalles $(]x_j - \varepsilon_j; x_j + \varepsilon_j[)_{j \in \{2; \dots; n-1\}}$, nous pourrons rendre la fonction continûment dérivable en x_j en modifiant la dérivée de la manière présentée sur la Figure 2.20.

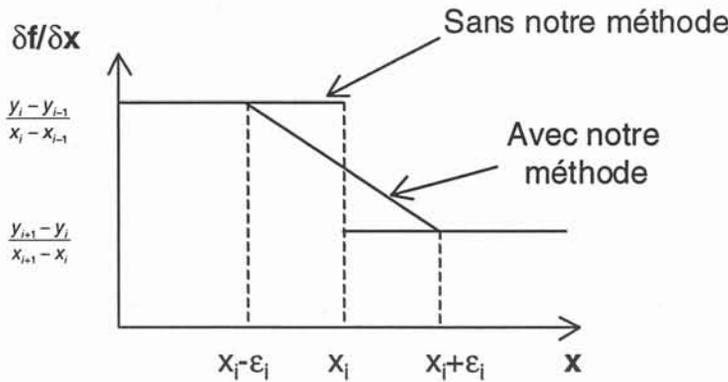


Figure 2.20 : Méthode proposée pour la dérivation d'une fonction continue affine par morceaux.

Cette dérivée devient alors continue et a pour expression analytique :

$$\frac{\partial f}{\partial x}(x) = \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} - \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \right) \cdot \frac{(x - x_i + \varepsilon_i)}{2 \cdot \varepsilon_i} + \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \tag{5}$$

L'intégration de cette dérivée nous donne la valeur de la fonction f sur l'intervalle considéré :

$$f(x) = C_1 \cdot \left(\frac{x^2}{2} - (x_i - \varepsilon_i) \cdot x \right) + C_2 \cdot (x - x_{i-1}) + C_3 \quad (6)$$

avec :

$$C_1 = \frac{1}{2\varepsilon_i} \cdot \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} - \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \right) \quad (7)$$

$$C_2 = \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \quad (8)$$

$$C_3 = \frac{(x_i - \varepsilon_i)^2}{4 \cdot \varepsilon_i} \cdot \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} - \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \right) + y_{i-1} \quad (9)$$

Sur les autres intervalles, $[x_1; x_2 - \varepsilon_2]$, $[x_{n-1} + \varepsilon_{n-1}; x_n]$ et $([x_k + \varepsilon_k; x_k - \varepsilon_k])_{k \in \{2; \dots; n-2\}}$, les expressions des dérivées et de la fonction restent inchangées car elles ne posent aucun problème.

Les dérivées sont de la forme :

$$\frac{\partial f}{\partial x}(x) = \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \quad (10)$$

et

$$f(x) = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \cdot (x - x_i) + y_i \quad (11)$$

Les expressions analytiques simples présentées ci-dessus ne sont valables que si les voisinages choisis pour les points $(x_j)_{j \in \{2; \dots; n-1\}}$ ne se chevauchent pas. En effet, il faut impérativement que les pentes aux points d'abscisses $x_j - \varepsilon_j$ et $x_j + \varepsilon_j$ soient celles des segments respectifs $[(x_{j-1}, y_{j-1}), (x_j, y_j)]$ et $[(x_j, y_j), (x_{j+1}, y_{j+1})]$. Nous avons écrit un programme permettant de détecter un tel problème et de le signaler à l'utilisateur.

De plus, comme nous allons insérer cette boîte de calcul dans un logiciel de dimensionnement, nous avons pensé à donner des valeurs à la fonction hors de son domaine de définition. Nous avons choisi de prolonger les pentes du bord du domaine de définition réel (Figure 2.21) [SAUVEY1-99].

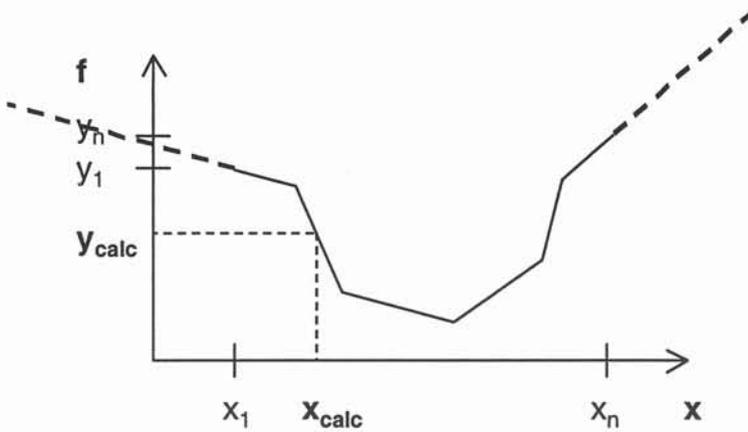


Figure 2.21 : Prolongement des pentes au bord du domaine de définition.

Programmation

La programmation de l'outil générique permettant d'obtenir des boîtes de calcul permettant d'intégrer des fonctions continues affines par morceaux comprend trois niveaux :

- Le premier niveau a été écrit en fortran et contient les routines de calcul des valeurs de la fonction et de ses dérivées ainsi que celles de vérification de la cohérence. Ces routines ont été écrites de manière définitive. Elles sont la traduction en langage fortran des formules présentées précédemment (1 à 5).
- On trouve dans le deuxième niveau cinq routines génériques écrites en fortran qui permettent de calculer tous les types de valeurs dont le logiciel pourrait avoir besoin : la valeur de la fonction f , les dérivées de f par rapport aux abscisses x_i , aux ordonnées y_i des points anguleux ainsi que par rapport aux voisinages choisis pour les différentes abscisses $(\varepsilon_j)_{j \in \{2; \dots; n-1\}}$. Ces routines sont elles aussi écrites une fois pour toutes et permettront un bon interfaçage avec le logiciel de dimensionnement.
- Le troisième niveau a été écrit dans le langage Macsyma [Macsyma] et permet de générer automatiquement les fonctions qui correspondent au problème particulier qui est défini comme paramètre d'entrée. Ce programme écrit également des informations nécessaires au logiciel de dimensionnement pour qu'il prenne en compte la boîte de calcul (en particulier la déclaration des fonctions et des gradients). Cette étape est la seule qui soit particulière au problème que l'on veut traiter.

Ainsi, pour créer une boîte de calcul correspondant à un problème particulier de fonctions continues affines par morceaux, il suffit de copier les fichiers fortran du premier niveau décrit et d'appeler le programme de troisième niveau dans le logiciel Macsyma qui génère automatiquement les fonctions FORTRAN manquantes. On peut résumer schématiquement la programmation de cet outil comme sur la Figure 2.22.

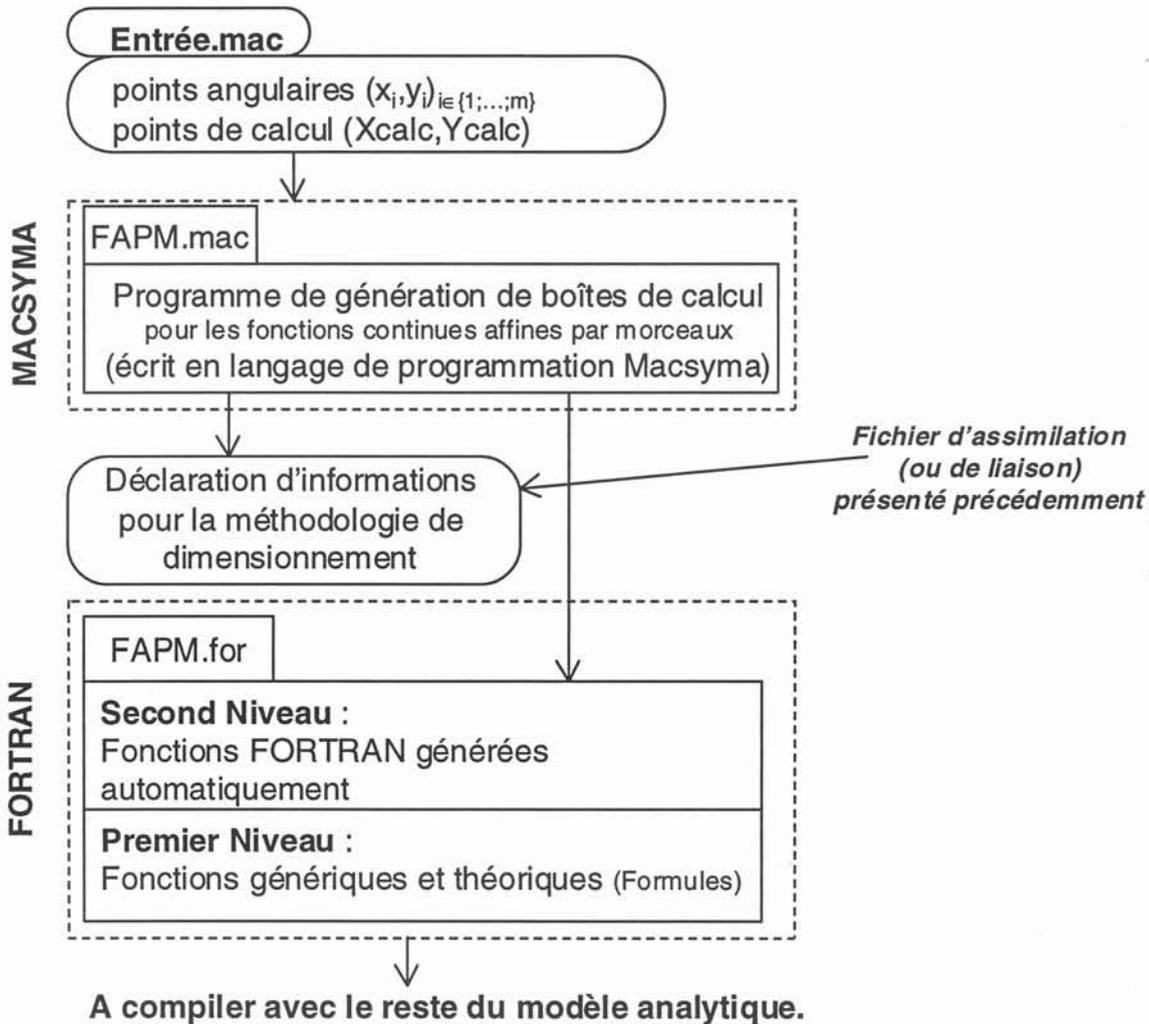


Figure 2.22 : Représentation schématique de la programmation de l'outil FAPM.

2.B.3.1.4 Type d'applications possibles.

Cette boîte de calcul nous a permis de résoudre notre problème d'intégration de la formule pour le calcul de l'inductance en fonction de la position angulaire du rotor dans un moteur à réluctance variable. La forme de l'inductance en fonction de la position peut se modéliser à l'aide de fonctions continues affines par morceaux dans

d'autres moteurs, notamment pour le moteur synchrone à pôles saillants. On voit sur la Figure 2.23 la forme de l'inductance en fonction de la position du rotor dans un MRV classique.

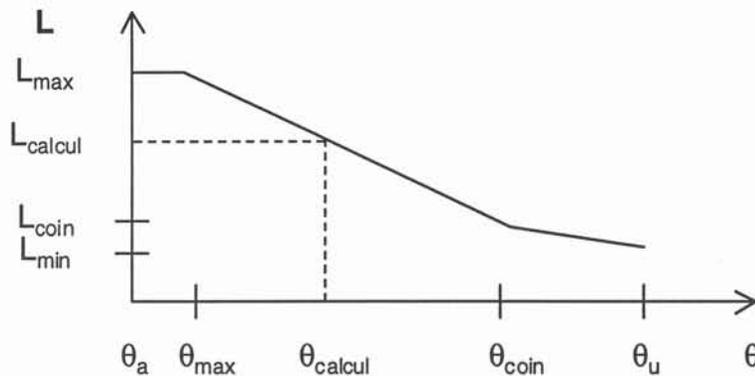


Figure 2.23 : Inductance en fonction de la position du rotor d'un MRV classique

2.B.3.2 Méthode des moindres carrés.

2.B.3.2.1 Position du problème.

L'approximation d'une courbe dans un certain intervalle par la méthode des moindres carrés [ANGOT-72] peut se résumer brièvement comme suit. Nous connaissons les valeurs de la fonction $y=\phi(x)$ en p points $(x_i)_{i \in \{1; \dots; p\}}$ et on cherche les valeurs des paramètres (a_1, \dots, a_n) tels que la fonctionnelle $f(a_1, \dots, a_n, x)$ approche le mieux ϕ selon le critère des moindres carrés (Figure 2.25). On cherche :

$$(a_1, \dots, a_n) = \min_{(a_1, \dots, a_n) \in \mathbb{R}^n} E(a_1, \dots, a_n, x_1, \dots, x_p, y_1, \dots, y_p) \quad (12)$$

Où $E(a_1, \dots, a_n, x_1, \dots, x_p)$ est l'erreur quadratique définie par :

$$E(a_1, \dots, a_n, x_1, \dots, x_p, y_1, \dots, y_p) = \sum_{i=1}^p (y_i - f(a_1, \dots, a_n, x_i))^2 \quad (13)$$

Notre problème est donc celui de la minimisation de la fonction $E(a_1, \dots, a_n, b_1, \dots, b_p)$ connaissant le $2p$ -uplet (b_1, \dots, b_{2p}) contenant les abscisses et les ordonnées des points à interpoler. Pour intégrer l'approximation d'une courbe par la méthode des moindres carrés à un processus d'optimisation de type gradient, nous devons considérer deux cas :

- Soit les points à interpoler sont fixés une fois pour toutes. Dans ce cas, il suffit d'utiliser la méthode des moindres carrés une seule fois en dehors du processus d'optimisation pour obtenir les valeurs de (a_1, \dots, a_n) .
- Soit les abscisses et les ordonnées des points sont paramétrés par ailleurs et sont susceptibles de varier au cours de l'optimisation. Dans ce cas, il devient nécessaire d'opérer la méthode des moindres carrés à chaque itération du processus d'optimisation.

C'est parce que le deuxième cas peut se produire que nous avons créé l'outil générique qui permet d'insérer dans un modèle analytique une optimisation par la méthode des moindres carrés.

2.B.3.2.2 Type de problème posé.

Le problème qui est posé par l'intégration à une méthodologie d'optimisation de type gradient de minimisation par la méthode des moindres carrés est celui de la résolution d'un problème de minimisation. Il est équivalent à un problème implicite comme nous allons le montrer par la suite.

Comme la famille de fonctions d'approximation est continue, on peut supposer que les valeurs des paramètres d'interpolation obtenues varieront de manière continue en fonction des paramètres d'entrée et donc que dériver ce processus ne pose pas de problème théorique.

2.B.3.2.3 Solution au problème.

L'approximation par la méthode des moindres carrés consiste à minimiser l'erreur quadratique. Reprenons notre problème général avec $E(a_1, \dots, a_n, b_1, \dots, b_p)$. Pour trouver le minimum (a_1, \dots, a_n) , nous avons choisi d'exprimer la condition pour qu'un point soit un minimum local. Les coefficients (a_{10}, \dots, a_{n0}) qui sont solution à notre problème à (b_1, \dots, b_p) donnés sont ceux qui vérifient :

$$\left\{ \begin{array}{l} \frac{\partial E(a_1, \dots, a_n; x_i; y_i)}{\partial a_1} = 0 = g_1(a_1, \dots, a_n; x_1, \dots, x_p; y_1, \dots, y_p) \\ \dots \\ \frac{\partial E(a_1, \dots, a_n; x_i; y_i)}{\partial a_n} = 0 = g_n(a_1, \dots, a_n; x_1, \dots, x_p; y_1, \dots, y_p) \end{array} \right. \quad (14)$$

On obtient un système de n équations implicites à n inconnues. Sous réserve d'être capable de résoudre ce système, on a résolu notre problème.

Il est à noter que cette méthode ne nous garantit pas d'obtenir le minimum global. Cependant, si ce système d'équations implicites ne comporte qu'une seule solution, cette méthode nous y amène directement. Dans l'équation (14), nous avons posé $g_i = (\delta E / \delta a_i)_{i \in \{1, \dots, n\}}$ pour simplifier l'écriture de l'équation implicite qu'il faut résoudre.

Nous disposons d'un outil informatique permettant de résoudre un système de n équations implicites à n inconnues [COUTEL-99]. Les hypothèses qui doivent être satisfaites pour se servir de cet outil sont, d'une part que les fonctions $(g_i)_{i \in \{1, \dots, n\}}$ doivent être de classe C^1 , et d'autre part que l'on doit connaître leur expression symbolique explicite. Or, chaque fonction g_i est en fait la dérivée de E par rapport à a_i . Donc, cela implique que la fonction E doit être de classe C^2 par rapport aux $(a_i)_{i \in \{1, \dots, n\}}$ et que chaque fonction $(\delta E / \delta a_i)_{i \in \{1, \dots, n\}}$ doit être de classe C^1 par rapport à chaque $(x_j)_{j \in \{1, \dots, p\}}$ et à chaque $(y_j)_{j \in \{1, \dots, p\}}$.

Comme nous voulons l'annulation de toutes les fonctions g , il ne faut pas que les dérivées $(\delta g / \delta a_i)_{i \in \{1, \dots, n\}}$ s'annulent dans les intervalles considérés. L'annulation de toutes les fonctions g nous donne le minimum pour E . L'utilisation de l'outil nous donne donc les $(a_i)_{i \in \{1, \dots, n\}}$ optimaux ainsi que leurs dérivées par rapport aux $(b_j)_{j \in \{1, \dots, p\}}$.

Programmation

Nous venons de montrer que notre problème de minimisation de l'erreur quadratique peut se ramener à un système de n équations implicites à n inconnues. Or, il existait déjà un outil permettant de résoudre ces systèmes d'équations. L'outil informatique nous permettant d'obtenir des boîtes de calcul a donc été réalisé en deux parties.

- La première partie de l'outil est un programme écrit dans le langage de programmation du logiciel Macsyma [Macsyma]. Il traite les données correspondant à notre problème de moindres carrés (contenues dans le fichier Entrée.mac) de manière à en faire un système de n équations implicites à n inconnues. C'est la traduction de ce qui a été écrit précédemment en un programme exécutable dans le logiciel Macsyma.
- Le résultat de l'exécution de ce programme (que sont les expressions analytiques des g_i) est passé en paramètre d'entrée à la seconde partie de l'outil qui est un autre programme écrit dans le langage de programmation du logiciel Macsyma. Ce programme est celui qui permet de gérer des systèmes d'équations implicites [COUTEL2-99]. Il reçoit les informations relatives au système de n équations à n inconnues et se charge automatiquement de l'appel à l'autre outil traitant ce problème. On peut résumer schématiquement la programmation de cet outil comme sur la Figure 2.24 avec l'exemple d'un polynôme du second degré comme fonctionnelle.

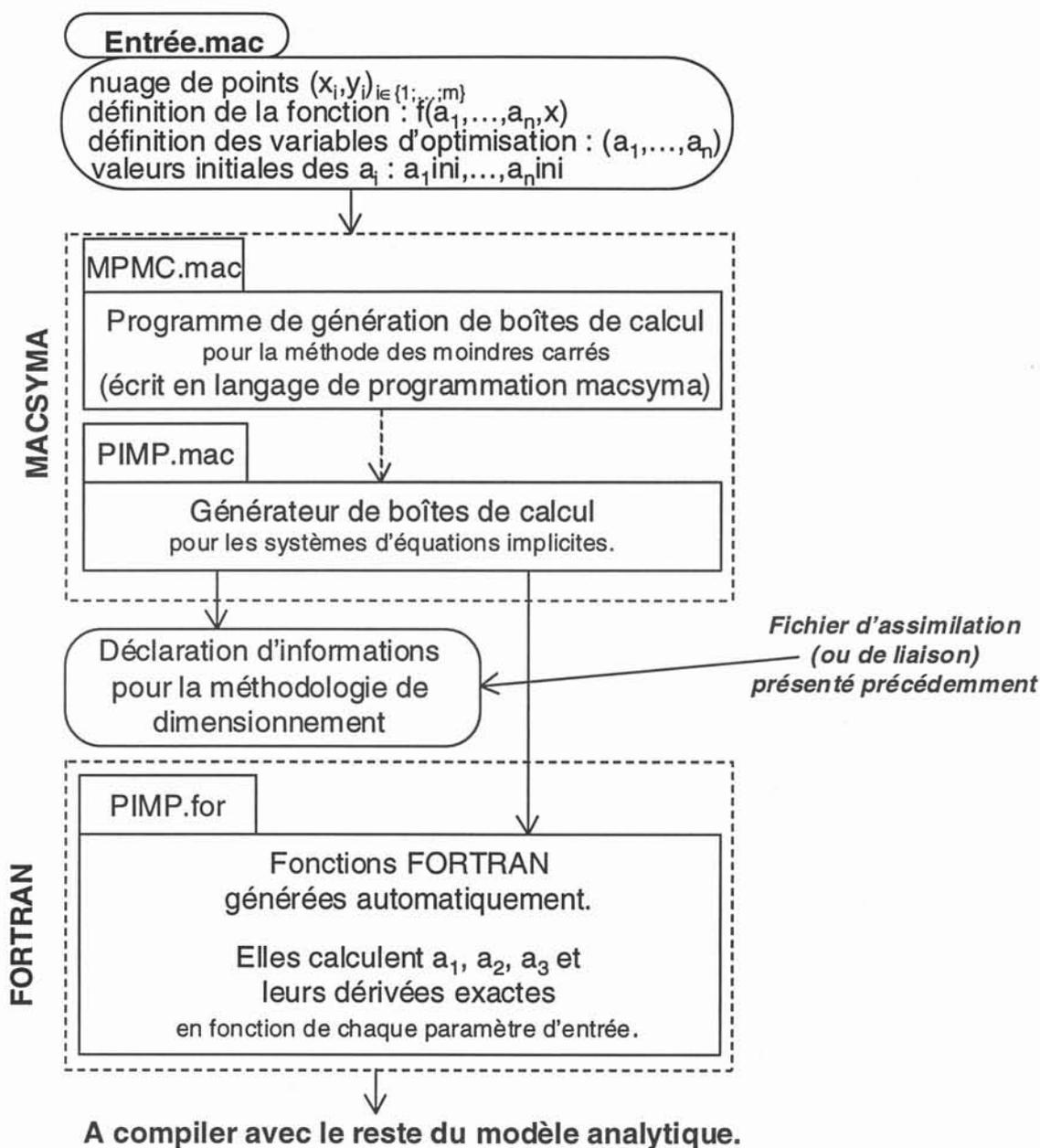


Figure 2.24 : Représentation schématique de la programmation de l'outil MPMC.

2.B.3.2.4 Type d'applications possibles.

Cet outil générique de création de boîtes de calcul nous a permis d'intégrer l'approximation de la courbe $\phi(i)$ de notre modèle de moteur à réluctance variable avec la méthode des moindres carrés à un processus d'optimisation de type gradient (Figure 2.25).

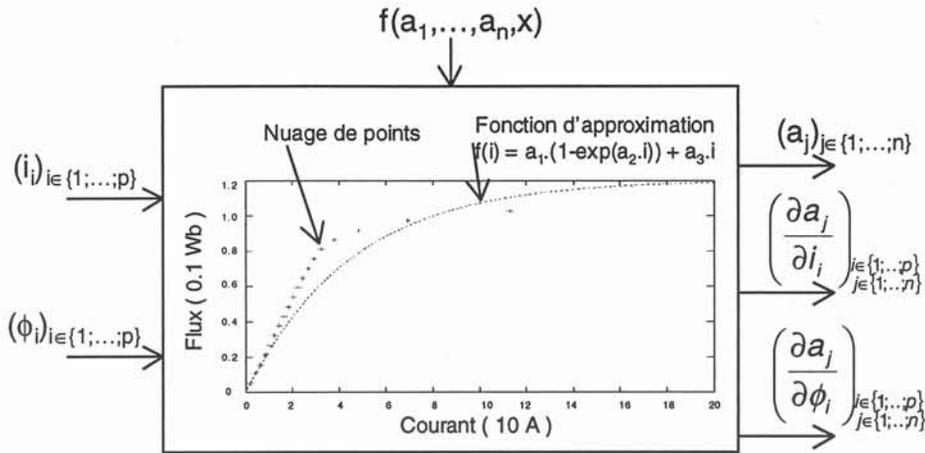


Figure 2.25 : Boîte de calcul réalisée pour la méthode des moindres carrés.

On peut imaginer d'autres applications pour cette boîte de calcul, comme par exemple la détermination quasi-automatique de coefficients correctifs que l'on pourrait introduire sur des modèles analytiques pour les faire « coller » aux résultats d'essais sur un domaine défini. On peut aussi imaginer d'utiliser une telle boîte de calcul pour interpoler la courbe d'aimantation $B(H)$ d'un matériau ferromagnétique.

2.B.3.3 Utilisation de sommes finies

2.B.3.3.1 Position du problème.

On veut prendre en compte dans un modèle analytique une fonction f dont l'expression est une somme telle que celle qui est décrite ici :

$$f = \sum_{i=i_{\min}}^{i_{\max}} g(a_1, \dots, a_p, i)$$

Avec :

i : variable muette, indice servant au calcul de la somme, mais n'étant pas une variable de la fonction f .

(a_1, \dots, a_p) : variables « principales » de la fonction f .

g : fonction continue et dérivable.

Nous avons créé un outil générique pour intégrer le calcul des sommes finies dans un processus d'optimisation de type gradient pour qu'il puisse accepter de traiter des modèles utilisant des sommes écrites avec le signe Σ , puis,

éventuellement, pour qu'un utilisateur puisse intégrer des sommes infinies, si il donne à l'indice n une valeur suffisamment grande. Les boîtes de calcul que nous génère automatiquement cet outil ont une interface du type de celle qui est présentée Figure 2.26.

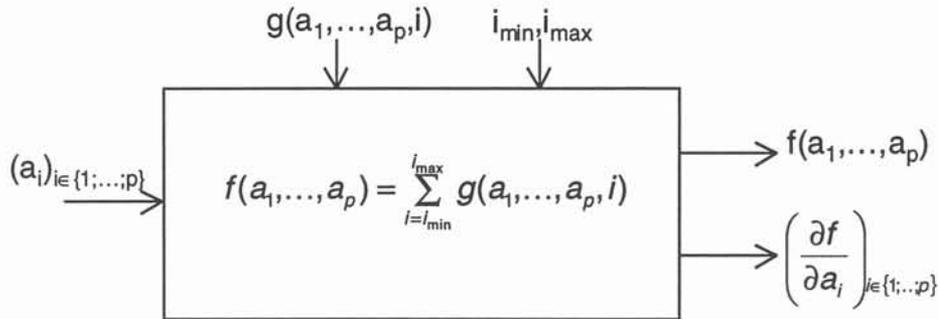


Figure 2.26 : Boîte de calcul permettant de calculer des sommes finies.

2.B.3.3.2 Type de problème posé.

Contrairement aux autres boîtes de calcul présentées dans ce chapitre, l'outil que nous avons créé pour pouvoir intégrer les sommes finies ne posait aucun problème théorique. Nous avons utilisé la méthodologie des boîtes de calcul pour résoudre un léger problème technique. En effet, le logiciel Macsyma [Macsyma] ne peut pas générer les instructions qui traduisent en FORTRAN une somme finie. En langage FORTRAN, on ne peut pas passer une fonction comme paramètre à une autre fonction. La programmation automatique avec le langage Macsyma trouve ici une de ses limites. L'outil qui a été développé est un simple programme écrit dans le langage de programmation spécifique à ce logiciel et qui génère automatiquement les boîtes de calcul à lui seul.

On peut quand même classer le problème posé parmi ceux qui sont explicites mais qui n'ont pas de formulation simple. Nous avons développé l'outil générique permettant d'obtenir les boîtes du type de celle qui est décrite sur la Figure 2.26.

2.B.3.3.3 Solution au problème.

Comme la réalisation d'un tel outil ne pose pas de difficultés majeures, il suffit pour calculer la valeur de f en un point, de créer un programme qui calcule une

somme de i_{\min} à i_{\max} des $g(a_1, \dots, a_p, i, i_{\min}, i_{\max})$. On vérifie quand même préalablement que i_{\min} et i_{\max} sont bien des entiers.

Pour le calcul des dérivées de f par rapport aux variables principales, il suffit de faire la même chose avec les dérivées de g par rapport à ces variables (g est de classe C^1).

2.B.3.3.4 Type d'applications possibles.

Les boîtes de calcul qui sont générées par cet outil ont été utilisées, par exemple, pour calculer l'inductance minimale d'un MRV de type classique [RADUN-94] :

$$L(l_1, l_2, dr) := 8 \cdot \mu_0 \cdot N_s^2 \cdot L_{fed} \cdot \sum_{n=0}^{100} \frac{\frac{\sin\left[\frac{(2n+1) \cdot \pi \cdot l_1}{l}\right]}{l_1} + \frac{\sin\left[\frac{(2n+1) \cdot \pi \cdot (l-l_2)}{l}\right]}{l_2}}{(\pi \cdot (2n+1))^2 \cdot \text{th}\left[\frac{(2n+1) \cdot \pi \cdot dr}{l}\right]} \cdot \left(1.04 - 1.14 \exp\left(-1.44 \frac{L_{fed}}{R_{eff}}\right)\right)^2$$

Elles ont également été utilisées dans un modèle de refroidisseur de composants de puissance [PERRET-00].

On pourrait aussi imaginer leur utilisation pour le calcul de signaux carrés par somme de séries de Fourier ou toute autre application faisant intervenir des calculs de sommes.

2.B.3.4 Calcul des racines d'un polynôme en fonction de ses coefficients.

2.B.3.4.1 Position du problème.

Nous avons remarqué que beaucoup de modèles analytiques utilisent des polynômes [POLOU-86], l'optimisation étant parfois réalisée avec la racine de ceux-ci. C'est le cas, par exemple, lorsque l'on veut calculer la hauteur d'un transformateur à partir de la puissance qu'il doit fournir et lorsqu'on lui impose une valeur d'inductance de court-circuit. C'est la raison pour laquelle nous avons souhaité créer un outil générique pour intégrer le calcul des racines d'un polynôme en fonction de ses coefficients à un processus d'optimisation utilisant les gradients.

La boîte de calcul que nous voulons créer à l'interface suivante :

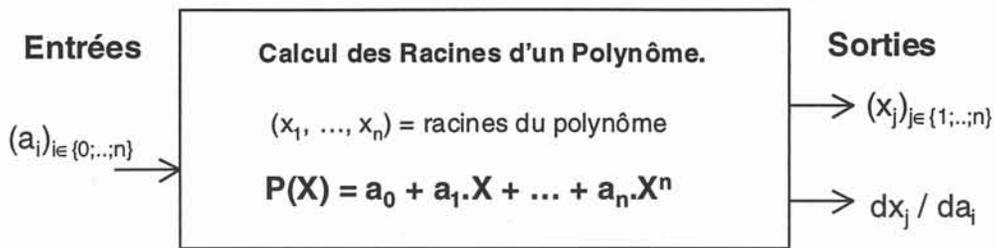


Figure 2.27 : Boîte de calcul des racines d'un polynôme.

2.B.3.4.2 Type de problème posé.

Le type de problème que pose la réalisation d'une telle boîte de calcul est avant tout que le calcul des racines est un problème implicite. En effet, le calcul des racines n'est explicite que pour les polynômes dont le degré est inférieur ou égal à 3. Des algorithmes numériques existent pour rechercher les racines d'un polynôme. Aussi, nous ne nous sommes pas intéressés à ce problème car nous avons trouvé un algorithme qui nous a donné entière satisfaction [Harwell].

Nous verrons plus loin que le calcul de la dérivée des racines en fonction des coefficients dépend de leurs valeurs, ainsi que de leurs ordres de multiplicité. Nous avons donc mis en place les programmes adéquats pour permettre à cette boîte de calculer les dérivées correctement.

2.B.3.4.3 Solution au problème.

Une fois que nous avons connaissance des n racines du polynôme et de leur ordre de multiplicité grâce à l'algorithme sus cité, on peut procéder au calcul des dérivées. En effet, la dérivée première d'une racine de polynôme dépend de son ordre de multiplicité.

- **Faisons le calcul pour une racine d'ordre 1 :**

Soit un polynôme $P(X)$ de degré n (supérieur ou égal à 2) :

$$P(X) = a_0 + a_1 \cdot X + a_2 \cdot X^2 + \dots + a_n \cdot X^n \quad (15)$$

Soit x_j une racine de P . En x_j , on a :

$$P(x_j) = 0 \quad (16)$$

De plus, on peut écrire :

$$dP = 0 \quad (17)$$

Car le polynôme P a des coefficients constants à chaque étape du processus d'optimisation (il a bien fallu fixer les $(a_i)_{i \in \{0; \dots; n\}}$ pour pouvoir trouver les n racines, dont x_j).

On peut donc écrire :

$$\frac{dP}{da_i} = 0, \forall i \in \{0, \dots, n\} \quad (18)$$

Or cette dérivée peut s'écrire sous la forme :

$$\frac{dP}{da_i} = \frac{dP}{dX} \cdot \frac{dX}{da_i} + \frac{dP}{da_i} \quad (19)$$

De (16) et (17), il vient assez facilement :

$$\frac{dx_j}{da_i} = - \frac{\frac{dP}{da_i}(x_j)}{\frac{dP}{dX}(x_j)}, \text{ car } \frac{dP}{dX}(x_j) \neq 0 \quad (20)$$

On voit clairement apparaître l'importance de l'ordre de multiplicité de x_j dans l'expression (20). En effet, si x_j est une racine d'ordre 1, la condition $(\delta P / \delta X)(x_j) \neq 0$ est vérifiée (c'est la définition d'une racine d'ordre 1). Par contre, si x_j était d'ordre supérieur ou égal à 2, on pourrait écrire les mêmes lignes **sauf** la dernière (20) qui mène à une division par zéro.

- **Pour calculer les dérivées d'une racine d'ordre supérieur à 1 :**

Soit q l'ordre de multiplicité de la racine x_j . q est alors le premier entier pour lequel on peut écrire :

$$P^{(q)}(x_j) \neq 0 \quad (21)$$

Si on note $f^{(i)}$ la $i^{\text{ème}}$ dérivée d'une fonction f . En suivant le même raisonnement que celui qui est décrit plus haut, on arrive à la formule suivante pour la dérivée de la racine x_j par rapport au coefficient a_i :

$$\frac{dx_j}{da_i} = -\frac{\frac{dP^{(q-1)}}{da_i}(x_j)}{\frac{dP^{(q-1)}}{dX}(x_j)}, \text{ car } \frac{dP^{(q-1)}}{dX}(x_j) = P^{(q)}(x_j) \neq 0 \quad (22)$$

On remarquera que la formule (20) valable pour une racine d'ordre 1 n'est autre que l'application de la formule (22) dans le cas particulier $q=1$. La programmation de l'outil POLY [SAUVEY-00] est similaire à celle de FAPM [SAUVEY1-99], qui a été présentée en détail précédemment.

2.B.3.4.4 Type d'applications possibles.

L'outil POLY a été développé par des étudiants de l'ENSIEG [Rapport] dans le cadre d'un projet que nous avons défini et encadré. Le besoin de développement d'un tel outil n'était pas immédiat mais nous avons pu juger de son utilité pour des modèles déjà existants en électrotechnique. On peut citer le cas d'un transformateur triphasé [POLOU-86] pour lequel la racine d'un polynôme est la hauteur optimale des colonnes d'un transformateur triphasé.

Certains modèles de commande de convertisseurs statiques nécessitent parfois la résolution de polynômes dont les degrés avoisinent la dizaine [GERBAUD-00].

On pourra sans doute trouver d'autres modèles utilisés en électrotechnique qui contiennent des polynômes. Pour tous ces modèles, notre outil sera déjà disponible et prêt à être utilisé afin de permettre leur utilisation dans un processus d'optimisation de type gradient.

2.B.4 Boîtes de calcul qu'il pourrait être intéressant de réaliser.

Nous pouvons imaginer une multitude de boîtes de calcul à intégrer à un processus d'optimisation de type gradient. Il faut avant tout traiter celles qui ont un intérêt pour le concepteur en électrotechnique. Nous allons présenter dans les paragraphes qui suivent deux boîtes de calcul qu'il nous semblerait intéressant de développer pour agrandir l'ensemble des modèles que peut prendre en compte la méthodologie générique d'optimisation développée au laboratoire. Il s'agit du calcul d'intégrales et du calcul des équations différentielles.

Nous présenterons ces boîtes de calcul différemment des précédentes, car, contrairement à celles-ci, nous n'avons pas développé les outils nécessaires à la création de ces boîtes. Nous commencerons donc par présenter ce qu'elles pourraient apporter à un concepteur en électrotechnique, puis nous donnerons une idée de comment les réaliser.

2.B.4.1 Calcul d'intégrales.

2.B.4.1.1 Type d'applications possibles.

Les calculs d'intégrales sont fréquents dans les modèles d'électrotechnique. En effet, le couple moyen d'un moteur à réluctance variable se calcule à l'aide d'une intégrale sur une période du couple instantané. De même, l'énergie emmagasinée dans un circuit magnétique se calcule aussi à l'aide d'une intégrale sur le terme $H \cdot dB$ de la courbe d'aimantation $H(B)$ du matériau.

Même si nous n'avons pas écrit un outil générique permettant de traiter ce problème, nous pensons qu'il serait utile de pouvoir insérer le calcul d'intégrales dans une méthodologie d'optimisation de type gradient car il permettrait de calculer les valeurs moyennes inaccessibles aux modèles analytiques.

2.B.4.1.2 Position du problème.

L'intégrale I de a à b d'une fonction f dépendant des $p+1$ paramètres (a_1, \dots, a_p, t) s'écrit de cette manière :

$$I = \int_a^b f(a_1, \dots, a_p, t) \cdot dt$$

La boîte de calcul que nous souhaiterions réaliser se présente ainsi.

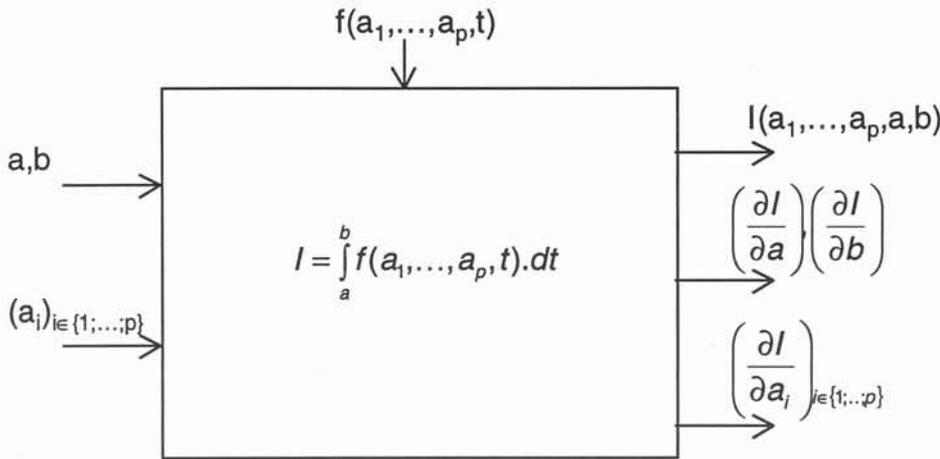


Figure 2.28 :Boite de calcul envisageable pour le calcul d'intégrales.

2.B.4.1.3 Type de problème posé.

L'outil générique qu'il faudra créer pour obtenir ces boîtes de calcul posera sans doute le même type de problèmes que celui qui a été développé pour les sommes finies. En effet, l'idée qui vient à l'esprit lorsque l'on est face à ce type de problème est le calcul par sommes de Riemann. Ainsi, on se ramène au cas des sommes finies.

$$\int_a^b f(a_1, \dots, a_p, t) dt = \lim_{n \rightarrow +\infty} \frac{(b-a)}{n} \cdot \sum f(a_1, \dots, a_p, a + i \cdot \frac{(b-a)}{n})$$

Comme l'outil permettant de traiter les sommes finies a été développé, il nous semble que la réalisation de celui permettant de calculer les intégrales sera possible également.

2.B.4.2 Equations différentielles.

2.B.4.2.1 Type d'applications possibles.

L'enjeu de la réalisation d'une boîte de calcul permettant de prendre en considération des équations différentielles dans une méthodologie d'optimisation de type gradient est considérable. En effet, nous serions alors capables de dimensionner en prenant en compte des contraintes imposées sur les transitoires.

Ce serait très intéressant pour dimensionner des alternateurs, par exemple. Le même problème se pose également lors de la résolution d'équations différentielles de circuits.

Cependant, pour pouvoir faire cela, il faut posséder un modèle qui sache prendre en compte le paramètre temps et son évolution. Donc, pour pouvoir développer une telle boîte de calcul, il faudra d'abord penser à étendre la méthodologie générique à la prise en compte du paramètre temps.

2.C Exemple d'application de la méthodologie des boîtes de calcul au dimensionnement d'un moteur à réluctance variable de type classique.

2.C.1 Structure du modèle de MRV Classique.

2.C.1.1 Structure du moteur.

Le MRV de type classique est en fait un moteur à réluctance variable à *double saillance* à *circuit magnétique unique*. Il est à double saillance parce qu'il utilise la variation de réluctance due au profil du rotor et du stator. Il est à circuit magnétique unique parce que les trois phases du moteur utilisent le même circuit magnétique. Nous verrons, par exemple, dans le troisième chapitre, que les moteurs à réluctance variable à motifs sont à circuit magnétique multiple.

Le stator et le rotor sont obtenus par empilage de tôles et le rotor n'est pas bobiné ce qui en fait un moteur très robuste et de maintenance facile. Les bobines sont placées autour des dents du stator. A partir de la position d'alignement de l'axe de symétrie d'une paire de pôles rotor avec celui des pôles d'une phase excitée, si on déplace le rotor d'un côté ou de l'autre, il apparaît un couple de rappel qui ramène le rotor sur la position d'alignement. Cette position est stable (position de détente). Elle correspond à un flux et à une inductance de phase maximum. Pour une position quelconque du rotor, le couple est produit par l'attraction magnétique des pôles rotor les plus proches des pôles stator excités depuis le début de recouvrement mutuel des pôles jusqu'à la position où il y a recouvrement total d'un pôle par l'autre (ce qui conduit à un couple nul tant que cette situation dure). C'est à dire qu'il y a production

de couple sur un arc polaire β égal au plus petit des arcs polaires des dents stator (β_s) ou rotor (β_r) comme on le voit sur la Figure 2.29.

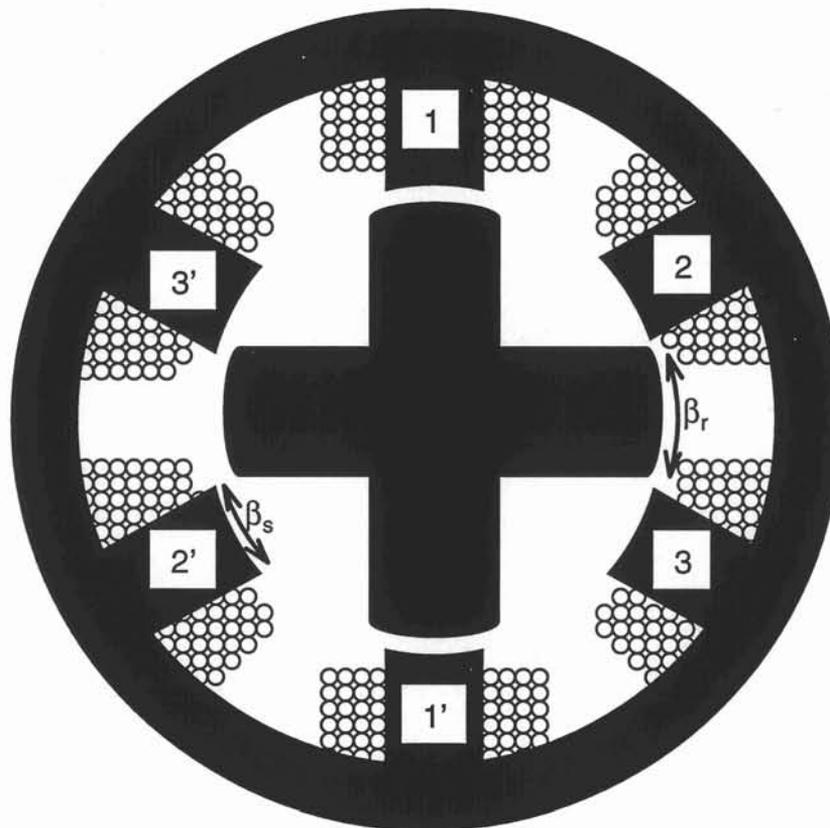


Figure 2.29 : Moteur à réluctance variable à 6 pôles stator et 4 pôles rotor.

2.C.1.2 Description du modèle semi analytique.

Nous disposons pour ce travail d'un modèle semi analytique de moteur à réluctance variable. Le modèle ne sera pas décrit dans les détails mais nous donnerons une idée de sa structure et une bonne vue d'ensemble de sa complexité. C'est un modèle valable à vitesse réduite - il suppose que le courant s'établit instantanément dans une phase à sa valeur maximale -, qui peut tenir compte de la saturation. Nous le présenterons dans ce paragraphe en différentes sections qui sont interdépendantes.

Calcul des paramètres géométriques et des paramètres de construction.

Tous les paramètres géométriques et de construction (dimensions du stator et du rotor, de la taille et de la largeur de la dent, des caractéristiques des

enroulements...) aussi bien que tous les paramètres qui dépendent d'eux (poids du stator, poids du rotor, poids de cuivre, encoches, ...) sont liés entre eux par des équations analytiques. Par exemple, le poids de la culasse stator m_{stat} est calculé avec l'équation suivante:

$$m_{stat} = \pi \cdot (res^2 - (ris + ds)^2) \cdot l_{fer} \cdot mv_{stat} \quad (23)$$

où res et ris sont des paramètres géométriques visibles dans la Figure 2.30, l_{fer} est la longueur de fer du moteur et mv_{stat} est le poids par unité du volume du fer utilisé. Avec une méthodologie d'optimisation de type de gradient comme logiciel de PASCOSMA, il est possible d'employer un nombre très élevé de paramètres pour définir une structure électrotechnique. La description de quelques paramètres de construction utilisés pour le modèle de MRV est présentée sur la Figure 2.30.

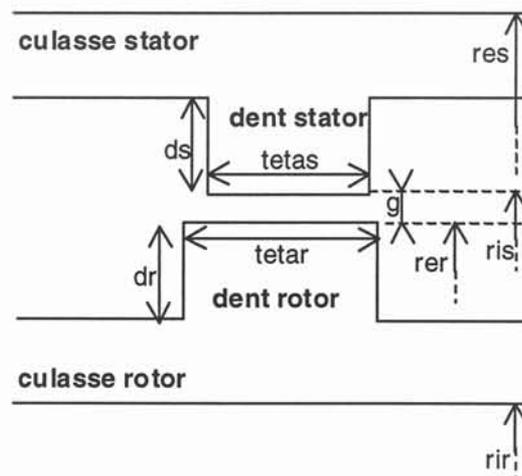


Figure 2.30 : Description de quelques paramètres géométriques du moteur.

Calcul des densités de flux magnétique.

Toutes les performances magnétiques sont écrites sous forme d'équations analytiques. Nous obtenons une expression analytique du champ d'excitation en fonction de la densité de flux $H(b)$ du fer que nous utilisons. Avec la description géométrique du moteur, nous exprimons analytiquement les lois de la conservation du flux pour la position alignée. Quand nous donnons une valeur à la densité de flux dans des dents du stator, nous pouvons connaître les densités de flux dans toutes les pièces du moteur. Puis, avec ce circuit magnétique équivalent, nous calculons la

force magnéto motrice requise et le courant dans l'enroulement. Nous obtenons ainsi la courbe $\Phi(i)$ pour la position alignée.

Calcul des inductances caractéristiques.

L'inductance maximale est calculée naturellement après le calcul des densités de flux et du courant correspondant. En effet, si nous donnons une valeur à la densité de flux, nous avons directement la valeur de tout le flux dans une phase en la multipliant par la section de passage. L'équation suivante donne automatiquement la valeur de l'inductance maximale L_{\max} :

$$L_{\max} = \frac{\text{Flux total dans la phase vu par la bobine}}{\text{Valeur du courant dans la bobine}} \quad (24)$$

L'inductance minimale et l'inductance de coin (inductance de la phase au moment où les dents stator et rotor commencent à se faire face) sont obtenues par la formule analytique que nous avons trouvée dans la littérature [RADUN-94], est présentée ci-dessous.

$$L(a, b, dr) = 8 \cdot \mu_0 \cdot N_s^2 \cdot l_{fer} \cdot l \cdot K \cdot \sum_{n=0}^{\infty} \frac{\frac{\sin((2n+1)\pi \cdot a)}{a} + \frac{\sin((2n+1)\pi \cdot (l-b))}{b}}{(\pi \cdot (2n+1))^2 \cdot \tanh\left(\frac{(2n+1)\pi \cdot dr}{l}\right)} \quad (25)$$

avec :

$$K = \left(1.06 + 1.16 \cdot \exp\left(-1.44 \cdot \frac{l_{fer}}{r_{er}}\right) \right)^2 \quad (26)$$

Nous pouvons dire qu'elle est obtenue avec la solution analytique des équations de Maxwell dans un cadre paramétré. Cette formule est une somme infinie et elle nous a posé des problèmes d'intégration dans la méthodologie générique. Nous les avons résolus avec la boîte de calcul de sommes finies présentée plus haut (§ 2.B.3.3).

Calcul des coefficients de magnétisation.

Les coefficients de magnétisation $a_i(\theta)$ sont ceux utilisés dans l'équation (27). Ils permettent de calculer la courbe $\Phi(\theta, i)$ où Φ est le flux de phase, θ est la position angulaire de rotor et i est le courant de phase. Ils sont étroitement liés à la géométrie du MRV, mais plus de détails sont disponibles dans [TORREY-95]. Nous pouvons donner une courte description de chaque paramètre. Dans l'équation (27), $a_1(\theta)$ peut être considéré comme la valeur du flux dans la phase quand elle passe du régime linéaire au régime saturé. Le coefficient $a_2(\theta)$ contrôle la courbure du coude de saturation. Le coefficient $a_3(\theta)$ est l'inductance incrémentale de phase aux forts courants.

$$\Phi(\theta, i) = a_1(\theta) \cdot [1 - \exp[-a_2(\theta) \cdot i]] + a_3(\theta) \cdot i \quad (27)$$

Comme une bonne précision de la courbe $\Phi(\theta, i)$ est fondamentale à la fiabilité du modèle, le maximum de soin doit être porté au calcul des coefficients de magnétisation. Dans notre modèle semi analytique, ils sont calculés en position alignée grâce à la méthode des moindres carrés. Les coordonnées des points qui sont utilisés dans la méthode des moindres carrés sont calculées avant, séparément, en fonction de l'induction. En effet, dans le modèle, nous avons des équations analytiques tout à fait simples pour $\Phi(b)$ et $i(b)$ où Φ est le flux total, i le courant en position alignée et b l'induction dans des dents du stator. Puis, les valeurs des coefficients de magnétisation en position non alignée sont obtenues avec des formules empiriques ($a_{1_u} = 0.1 \cdot a_{1_a}$ et $a_{3_u} = 0.95 \cdot L_{\min}$). Ensuite, leurs valeurs en fonction de la position angulaire du rotor sont déduites grâce à la boîte de calcul des fonctions continues affines par morceaux.

Calcul des performances électromagnétiques.

La bonne connaissance de la courbe $\Phi(\theta, i)$ est tellement importante que le calcul du couple dépend directement des coefficients de magnétisation. En effet, le couple est proportionnel à la dérivée de la co-énergie en fonction de la position angulaire de rotor [MILLER-93]. À n'importe quelle position, la co-énergie est la zone comprise au-dessous de la courbe de magnétisation. Comme nous pouvons intégrer formellement la courbe $\Phi(\theta, i)$ en fonction du courant de phase (27), la définition des

coefficients de magnétisation nous donne maintenant une expression analytique du couple en fonction du courant de phase actuelle et de la position angulaire du rotor:

$$\text{Couple}(\theta, i) = a_1(\theta) \cdot i + a_3(\theta) \cdot i^2/2 + [1 - \exp [a_2(\theta) \cdot i]] \cdot a_1(\theta) / a_2(\theta) \quad (28)$$

Notre capacité à dériver les coefficients de magnétisation en fonction des paramètres de construction du MRV donnent à la méthode d'optimisation de type de gradient la possibilité à converger rapidement vers une bonne solution.

2.C.2 Dimensionnement d'un MRV Classique avec PASCOSMA et les boîtes de calcul nécessaires.

Le modèle dont nous disposons sur le MRV Classique est donc un modèle semi-analytique. Ce modèle est principalement constitué d'équations analytiques explicites, continûment dérivables qui ne posent donc aucun problème pour être intégrées à un tel processus. Les deux principales parties de modèle qui nous ont posé des problèmes d'intégration sont les fonctions continues affines par morceaux et l'utilisation de la méthode des moindres carrés.

2.C.2.1 Utilisation d'une fonction affine par morceaux.

- **Une fonction continue affine par morceaux** est utilisée dans notre modèle pour exprimer l'inductance en fonction de la position angulaire du rotor (Figure 2.31). Cette modélisation de l'inductance est assez courante pour les MRV [GLINKA-98] car le couple d'un MRV dépend directement de la dérivée de l'inductance en fonction de la position : $C = \frac{i^2}{2} \cdot \frac{dL}{d\theta}$. Une telle modélisation permet donc de ramener le calcul du couple à $C = K \cdot i^2$ avec $K = \text{Constante}$. Pour pouvoir utiliser cette partie de modèle avec le processus d'optimisation de type gradient, j'ai utilisé le générateur de boîtes de calcul décrit au paragraphe 2.B.3.1.

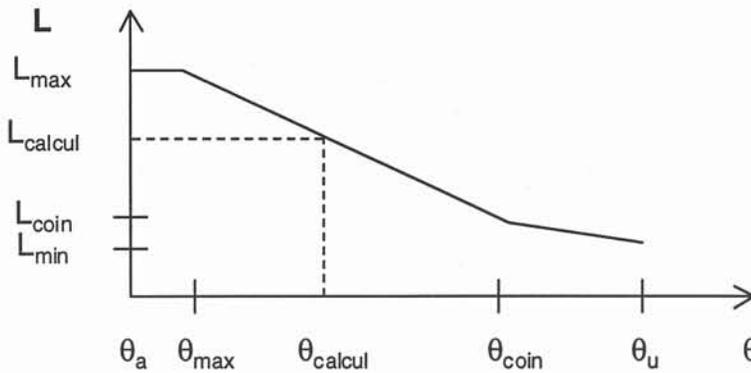


Figure 2.31 : Inductance en fonction de la position dans un MRV (fonctions continues affines par morceaux).

2.C.2.2 Méthode des moindres carrés.

Deux philosophies existent pour représenter mathématiquement le type de loi $b(h)$, l'approximation de fonctions ou l'interpolation [CHEVALIER-99]. La **méthode des moindres carrés** est l'une des principales méthodes que l'on peut utiliser pour l'approximation de fonctions. On utilise cette méthode dans notre modèle pour obtenir les paramètres a_1 , a_2 et a_3 d'une fonction définie à l'avance pour qu'elle « colle » au mieux aux points calculés par ailleurs dans le modèle (Figure 2.32). Nous présentons sur cette figure l'utilisation de la méthode des moindres carrés pour le calcul de ces coefficients dans la position alignée.

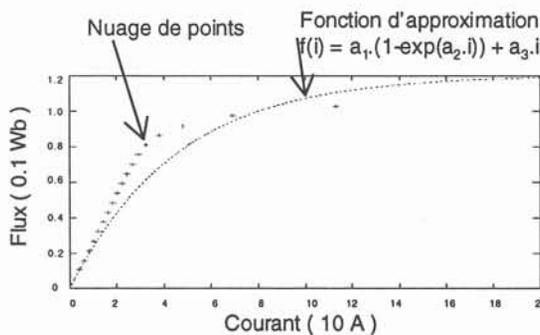


Figure 2.32 : Utilisation de la méthode des moindres carrés.

2.C.2.3 Résultats de dimensionnement.

Nous avons utilisé ces boîtes de calcul pour créer une application de dimensionnement de moteurs à réluctance variable de type classique avec le logiciel PASCOSMA. L'exécutable généré a pour taille 1.46 Mo et donne ses résultats d'optimisation dans des délais de l'ordre de la dizaine de secondes.

Nous avons réalisé l'optimisation décrite dans les figures de ce paragraphe.

Le moteur initial fournit un couple moyen de 5 N.m pour une masse de 5 kg mais a un échauffement de 215 K pour un courant efficace de 46 A. Le paramètre que nous avons choisi pour la fonction objectif est la masse ($z_{masse}=1$ et $z_{lfer}=0$). On retrouve ces informations dans les fenêtres de gauche de la Figure 2.33.

Le cahier des charges que nous avons imposé est présenté sur la Figure 2.34. On souhaite passer du moteur initial (qui fournit 5 N.m) à un moteur qui fournit 2 N.m. On souhaite également que l'échauffement après optimisation ait une valeur inférieure à 150 K. Par ailleurs, on contraint le courant à ne pas dépasser 50A. Les inductions dans les différentes parties du circuit magnétique sont contraintes dans des plages de variation raisonnables.

On trouve la liste exhaustive des paramètres du moteur après optimisation dans les deux fenêtres de droite de la Figure 2.33. L'algorithme d'optimisation a trouvé cette solution en 30 itérations et une dizaine de secondes.

On constate également que l'algorithme d'optimisation a normalement fonctionné car il a bien diminué la fonction objectif : la masse est passée de 5 à 3.3 kg. La longueur de fer y a largement contribué passant de 65 à 43 mm. On présente sur la Figure 2.35 les évolutions respectives de la fonction objectif, la longueur de fer, du couple moyen et de l'échauffement pour cette optimisation.

Nous donnons maintenant la nomenclature des paramètres d'entrée et de sortie du modèle que l'on retrouve dans la Figure 2.33 et la Figure 2.34.

Paramètres d'entrée :

- aunaini : Paramètre d'initialisation du paramètre de sortie auna (moindres carrés).
- bdents : Induction dans les dents stator.
- bref : Induction de référence pour le calcul des pertes fer
- csnbpts : Nombre de points pour le calcul de l'ondulation de couple *ondul*.

dr :	Hauteur de la dent rotor.
ds :	Hauteur de la dent stator.
entrefe :	Entrefe.
ldecorn :	Paramètre artificiel permettant d'utiliser la formule (23) pour le calcul de l'inductance de coin.
lfer :	Longueur de fer.
ns :	Nombre de spires.
pferm :	Pertes fer massiques à bref.
pr :	Nombre de dents au rotor.
ps :	Nombre de dents au stator.
res :	Rayon extérieur du stator.
rir :	Rayon intérieur du rotor.
ris :	Rayon intérieur du stator.
section :	Section des fils de cuivre du bobinage stator.
teta :	Angle pour lequel on calcule la valeur d'inductance.
tetar :	Angle d'une dent rotor (en degrés).
tetas :	Angle d'une dent stator (en degrés).
zfer :	Coefficient de prise en compte du paramètre lfer dans la fonction objectif <i>fob</i> .
zmasse :	Coefficient de prise en compte du paramètre masse dans la fonction objectif <i>fob</i> .

Paramètres de sortie :

fob :	Fonction objectif ($fob = zfer \cdot lfer + zmasse \cdot masse$).
adea :	Coefficient a_2 dans la position alignée.
adeu :	Coefficient a_2 dans la position non alignée.
aire :	Surface disponible pour le logement du bobinage stator.
atra :	Coefficient a_3 dans la position alignée.
atru :	Coefficient a_3 dans la position non alignée.
auna :	Coefficient a_1 dans la position alignée.
aunu :	Coefficient a_1 dans la position non alignée.
bculasr :	Induction dans la culasse rotor.
bculass :	Induction dans la culasse stator.
bdentr :	Induction dans la dent rotor.
bentref :	Induction dans l'entrefe.
cmoy :	Couple moyen du moteur.
courant :	Courant nécessaire pour obtenir l'induction bdents.
diamfil :	Diamètre des fils du bobinage stator.
echauff :	Echauffement du moteur.
energ :	Energie calculée à l'angle teta.
ieff :	Courant efficace.
indcalc :	Inductance calculée à l'angle teta.
lcorner :	Inductance de coin.
lmax :	Inductance maximale.
lmin :	Inductance minimale.
masse :	Masse du moteur.
ondul :	Ondulation de couple.
pfer :	Pertes fer.
pjoule :	Pertes Joule.
pmoteur :	Pertes Moteur.
rempli :	Coefficient de remplissage.
rer :	Rayon extérieur du rotor.
resist :	Résistance du bobinage stator.
rth :	Résistance thermique du moteur.
tetatr :	Angle correspondant à θ_{coin} sur la Figure 2.31.
tetau :	Angle de la position non alignée.
tetaun :	Angle correspondant à θ_{max} sur la Figure 2.31.
tr :	Angle d'une dent rotor (en radians).
ts :	Angle d'une dent stator (en radians).

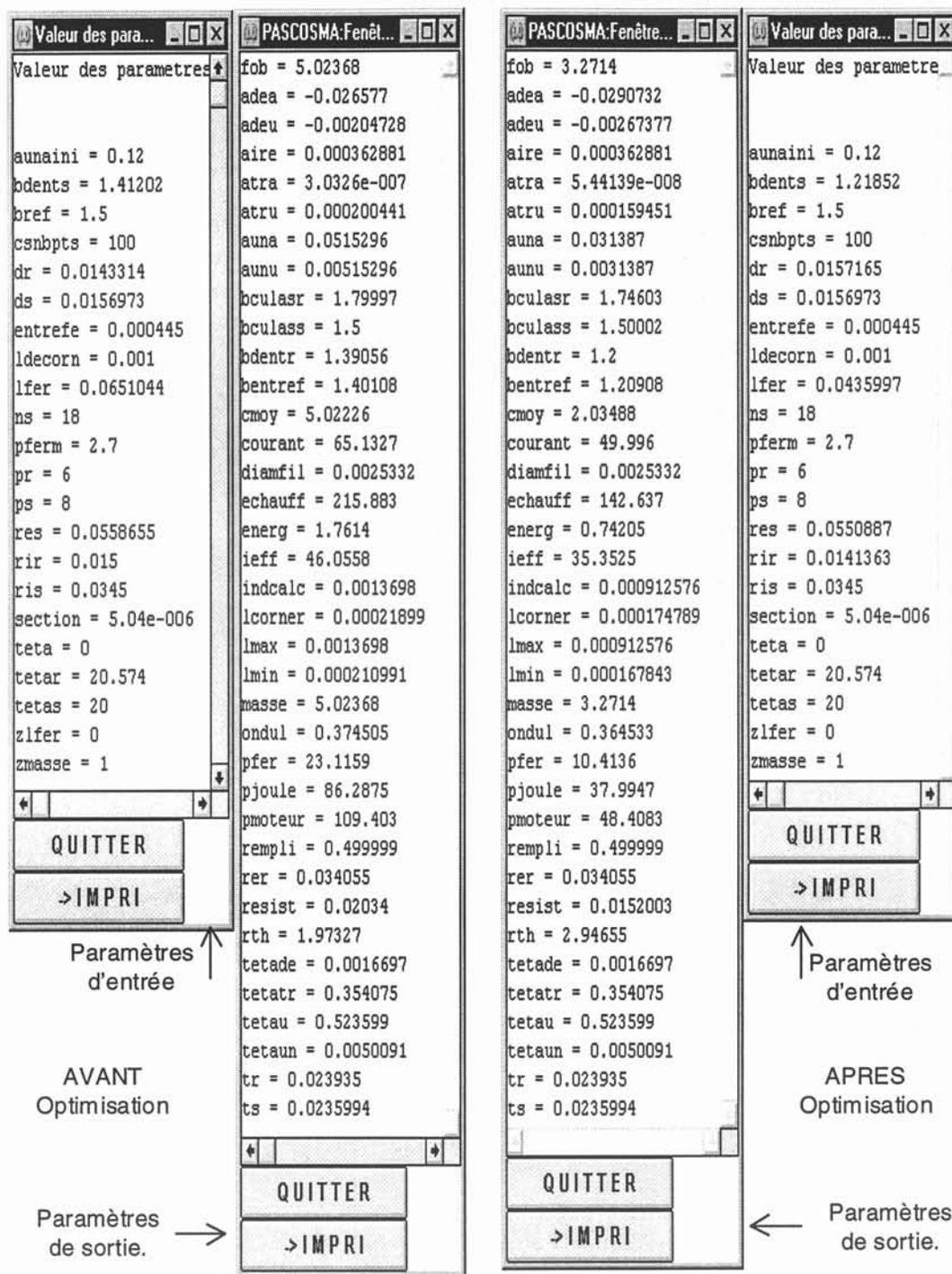


Figure 2.33 : Paramètres d'entrée et de sortie avant et après l'optimisation.

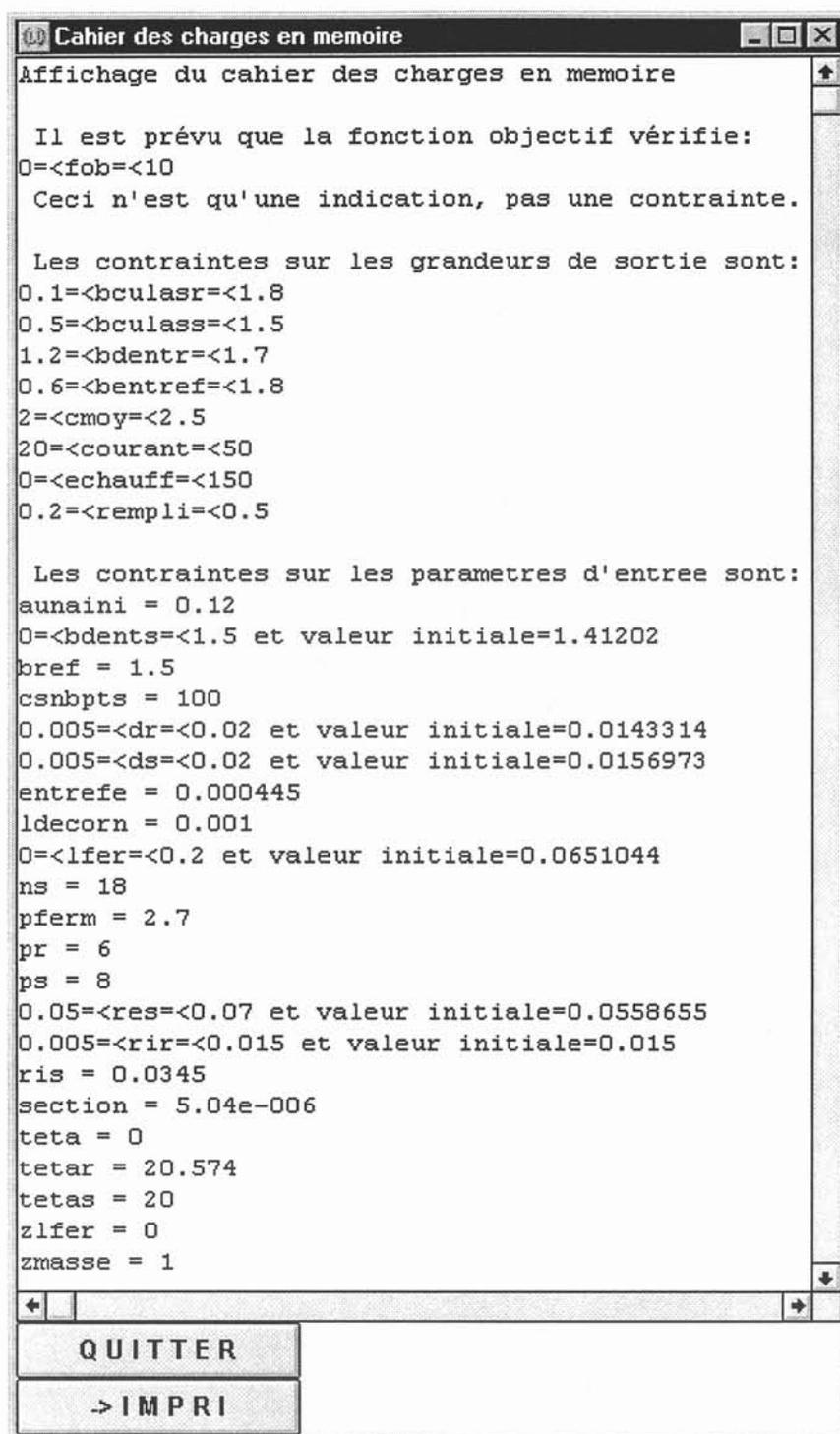
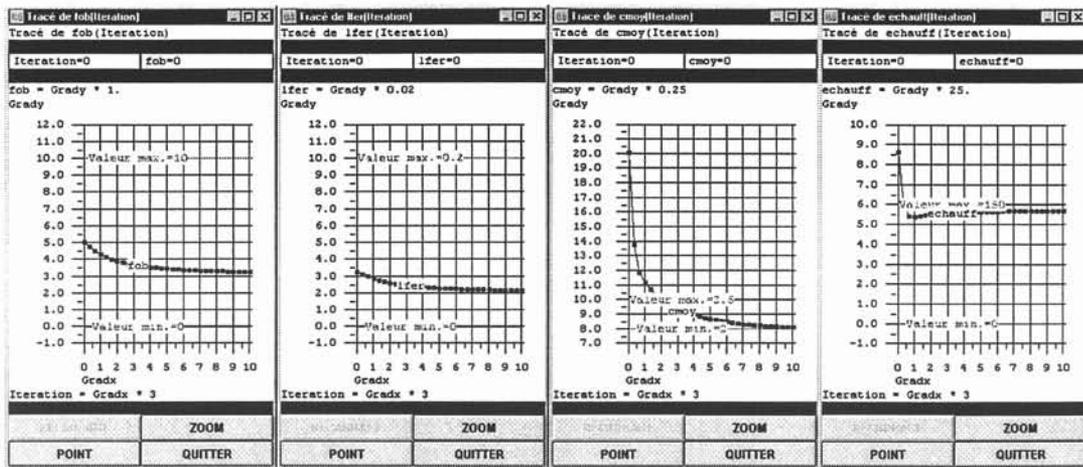


Figure 2.34 : Cahier des charges de l'optimisation réalisée.



Évolution de la fonction objectif

Évolution de la longueur de fer

Évolution du couple moyen

Évolution de l'échauffement

Figure 2.35 : Evolution de certains paramètres au cours de l'optimisation.

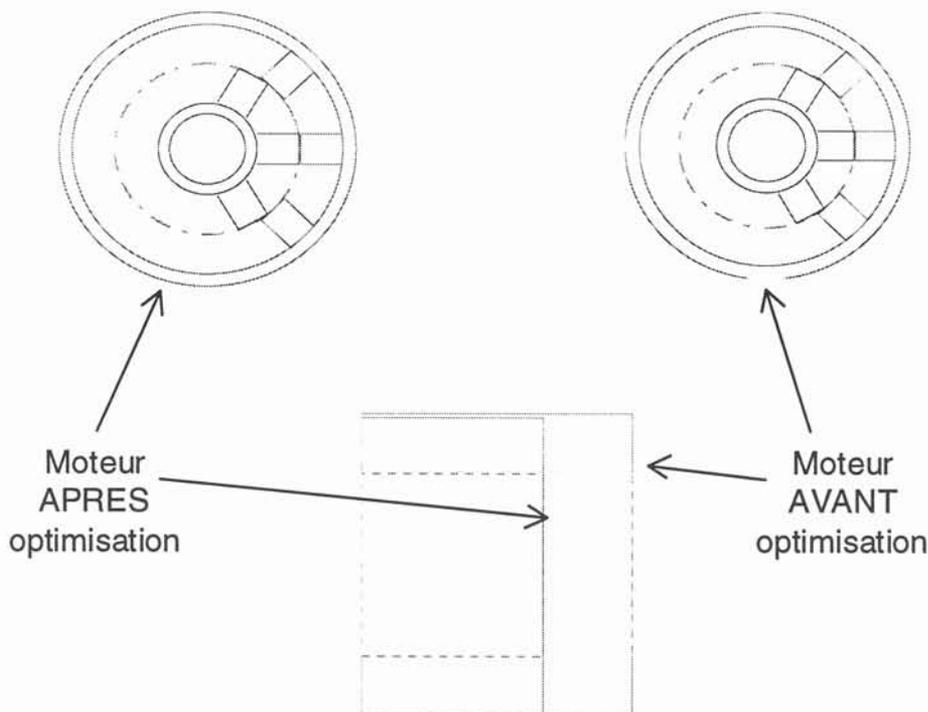


Figure 2.36 : Formes générales des moteurs avant et après l'optimisation.

On voit clairement sur la Figure 2.36 que la forme de la tôle avant et après l'optimisation n'a pas vraiment changé. En revanche, on retrouve bien la décroissance significative de la longueur de fer que l'on a vue sur la Figure 2.35. Les

informations données par les figures d'évolution des paramètres au cours du dimensionnement et celles données par la Figure 2.36 peuvent paraître redondantes mais la vue de la forme géométrique des tôles rotor et stator peut permettre au concepteur électrotechnicien de voir très rapidement si son dimensionnement est correct ou s'il y a un problème flagrant sur l'un des paramètres.

2.D CONCLUSION.

Dans ce chapitre, nous avons présenté la méthodologie développée au laboratoire (PASCOSMA), qui prévoit la génération automatique d'un logiciel de dimensionnement à partir des équations analytiques système à dimensionner. Nous en avons ensuite mis en évidence certaines limites pour la prise en compte des modèles semi analytiques. La solution que nous avons proposé est l'encapsulation des parties de modèle qui posent problème.

Nous avons ensuite défini les boîtes de calcul qui offrent un guide méthodologique pour mettre en œuvre cette solution. En fonction du modèle semi analytique que l'on veut utiliser. Nous avons illustré cela avec les exemples d'outils génériques que nous avons créés pour générer des boîtes de calculs sur des problèmes particuliers que nous avons rencontrés. Nous avons enfin donné des perspectives à ce travail en donnant des idées d'outils génériques qu'il serait intéressant de créer pour intégrer des modèles semi analytiques que l'on rencontre le plus souvent en électrotechnique.

Pour l'étude que nous avons menée sur le moteur à réluctance variable classique, nous avons utilisé un modèle semi analytique que nous n'avons pas remis en cause. Nous avons voulu l'intégrer à un processus d'optimisation de type gradient afin d'obtenir un outil de dimensionnement. Nous avons résolu les deux principaux problèmes que ce modèle posait grâce à deux boîtes de calcul. Nous avons donc créé les deux premiers outils génériques pour les fonctions continues affines par morceaux et pour la méthode des moindres carrés.

3 Modélisation en vue du dimensionnement d'un moteur à réluctance variable à motifs.

Dans ce chapitre, en ce qui concerne le processus de conception, nous nous plaçons avant la réalisation du premier prototype. Cela signifie que nous allons travailler dans l'ignorance de valeurs de mesures. En effet, pour pouvoir faire des mesures, il faut pouvoir disposer d'un prototype.

Nous commencerons par présenter la structure du moteur à réluctance variable à motifs qui nous intéresse. C'est un MRV à motifs à flux radial selon la définition donnée dans [SEGUIER-94].

Ensuite, nous savons que nous allons avoir besoin de deux modèles complémentaires pour pouvoir être efficaces : l'un sera analytique et rapide, l'autre sera numérique et précis, et chacun aura son rôle à jouer dans le processus de dimensionnement.

Les deux sources que sont ces deux modèles constituent autant d'informations utiles pour aider le concepteur à prendre une décision.

3.A Présentation de la structure du moteur à réluctance variable à motifs à flux radial.

Dans un premier temps, nous allons décrire la structure du moteur qui est fondée sur l'assemblage de ce que nous appelons des noyaux magnétiques. Nous continuerons par la description de l'électronique de puissance associée au moteur et par une brève présentation de l'électronique de commande. Notre conclusion insistera sur la sensibilité à la modélisation du noyau d'un modèle global du moteur.

3.A.1 Structure du moteur.

Notre MRV à motifs à flux radial se compose de trois couronnes électriquement indépendantes et mécaniquement solidaires. Chacune d'elles est composée d'un stator et d'un rotor et a la forme d'un cylindre plat (dont la hauteur est petite par rapport au rayon). Les trois couronnes sont empilées sur le même axe qui est l'axe de rotation du moteur. De plus, ce moteur est à structure inversée, c'est à dire que son rotor est situé en périphérie.

Nous décrirons ainsi la constitution du moteur. Nous présenterons d'abord le noyau magnétique. Ensuite, nous verrons la disposition de ces circuits magnétiques

élémentaires à l'intérieur d'une couronne. Enfin, nous présenterons l'assemblage de ces trois phases pour créer un moteur à réluctance variable.

3.A.1.1 Noyaux magnétiques.

Un noyau est un circuit magnétique constitué d'un entrefer et de deux U qui seront respectivement placés dans les parties stator et rotor du moteur (Figure 3.1). Le bobinage du moteur sera logé dans la partie du noyau magnétique située au stator (que nous appellerons fenêtre), ce qui explique pourquoi le U le plus grand est forcément situé au stator. Nous avons un U au rotor (et non un I) parce que ce qui nous intéresse est l'effet de réluctance qui existe entre les deux parties du noyau. A courant fixé, un moteur réalisé avec un rotor en I fournirait un couple inférieur car le flux magnétique serait plus dispersé au niveau de l'entrefer où il doit fournir la force réluctante.

Les noyaux magnétiques sont constitués de tôles « fer-silicium » à cristaux orientés. Ces tôles présentent des pertes fer faibles par rapport à d'autres matériaux. Elles sont usuellement utilisées dans la fabrication de transformateurs.

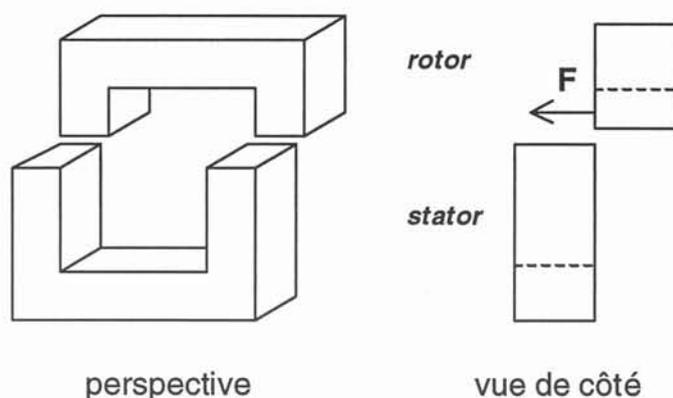


Figure 3.1 : représentation d'un noyau magnétique.

3.A.1.2 Création d'une couronne par assemblage de noyaux.

Pour créer une couronne, on répartit des noyaux sur l'entrefer de sorte que lorsqu'on regarde au niveau de l'entrefer, on ait une distance entre les noyaux égale à l'épaisseur d'un noyau. Les deux parties sont assemblées par serrage et collage au rotor et au stator sur des culasses en aluminium. Le bobinage est logé dans la

fenêtre des noyaux magnétiques situés au stator. On fait ensuite couler une résine organique de sorte à améliorer la cohésion des deux ensembles stator et rotor et diminuer les bruits.

Du point de vue électrique, une couronne est une phase du moteur. Chacune de ces phases est électriquement indépendante des autres et sera donc alimentée de manière complètement autonome.

De plus, dans un moteur à réluctance variable à motifs, chaque phase est indépendante des autres du point de vue magnétique, ce qui constitue une différence fondamentale avec le MRV de type classique.

La Figure 3.2 montre comment les noyaux magnétiques sont disposés pour réaliser une couronne.

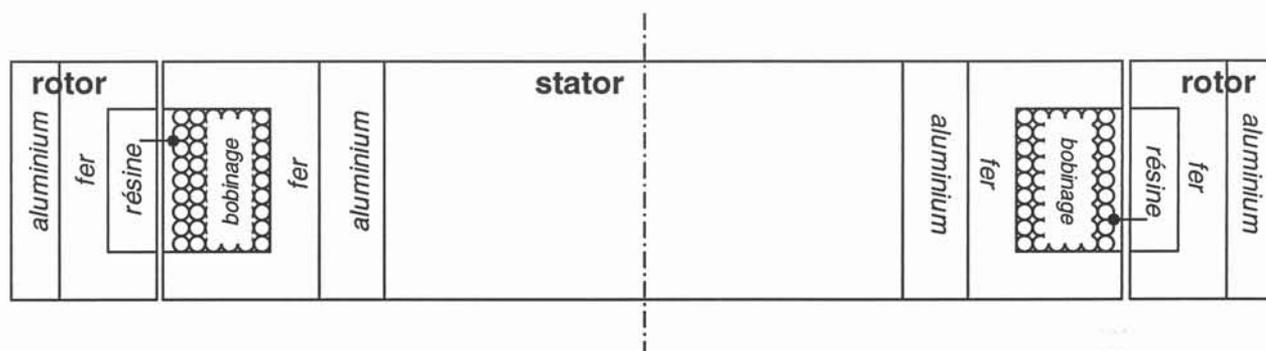


Figure 3.2 : représentation d'une couronne.

3.A.1.3 Création d'un moteur par assemblage de couronnes.

Pour créer un moteur, à trois phases par exemple, il suffit d'assembler mécaniquement de façon appropriée trois couronnes identiques. Pour notre moteur, par exemple, les trois couronnes ont été assemblées de la façon suivante :

- Les trois rotors sont mécaniquement solidaires et disposés de sorte que les noyaux magnétiques qu'ils supportent soient alignés.
- Les trois stators sont mécaniquement solidaires et disposés de sorte que les noyaux magnétiques qu'ils supportent soient décalés d'un tiers de période mécanique. Ainsi, les couples créés par les trois enroulements s'ajouteront de manière plus efficace.

Pour alimenter au mieux un tel moteur, il ne reste plus qu'à alimenter les trois phases du stator avec des tensions décalées d'un tiers de période électrique. Ainsi, on se rapproche du champ tournant utilisé dans les moteurs classiques. Les trois phases sont décalées géométriquement de $2\pi/3$, de même que les trois phases électriques sont décalées dans le temps de $2\pi/3$.

La Figure 3.3 donne une représentation de la manière selon laquelle les trois couronnes sont assemblées pour créer un moteur à réluctance variable à trois phases. Pour des raisons de commodité, nous avons représenté sur cette figure la disposition relative des couronnes au niveau de l'entrefer en vue développée.

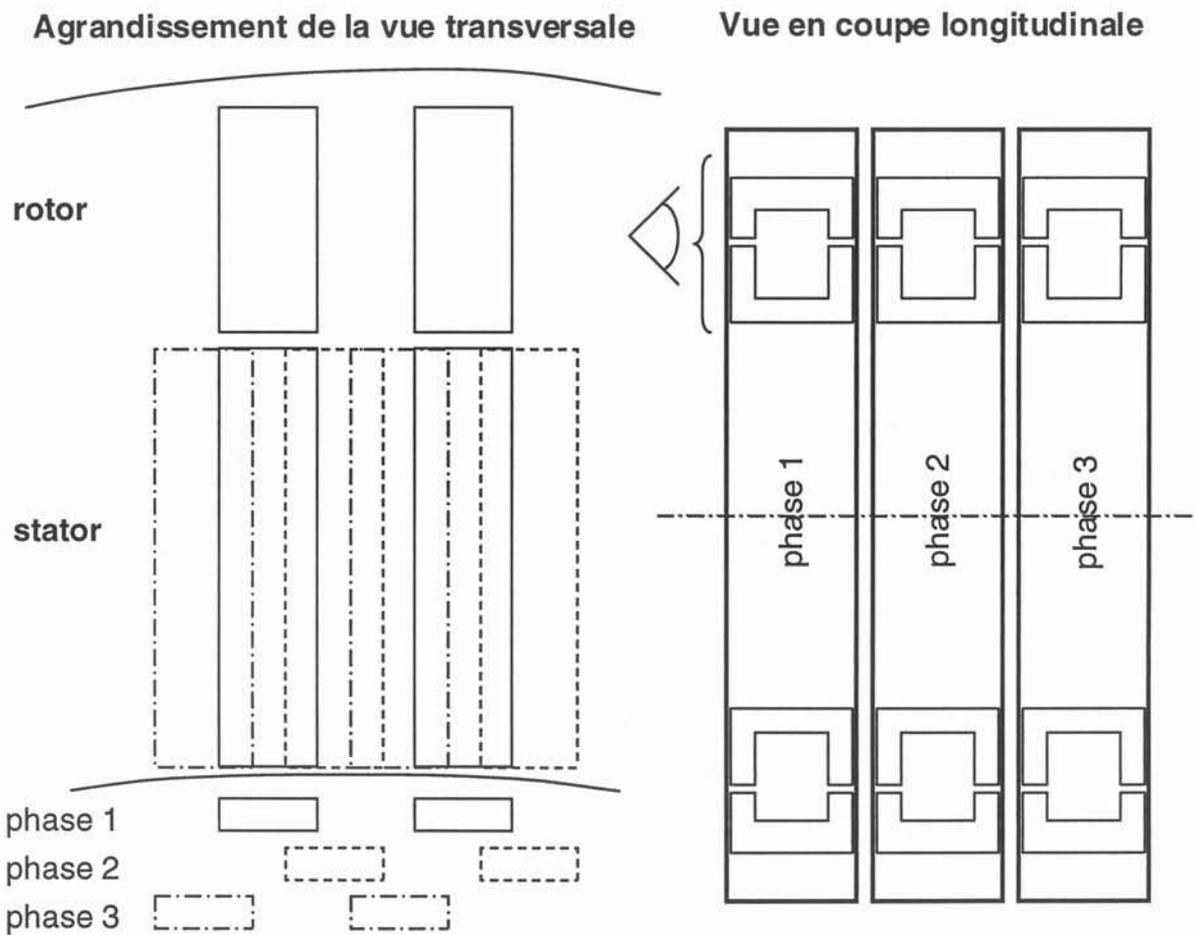


Figure 3.3 : représentation d'un moteur à trois phases

3.A.1.4 Conclusion.

En conclusion de la présentation de la structure du moteur, on peut remarquer sa modularité. En effet, l'élément de base de cette structure modulaire est le noyau magnétique. On utilise plusieurs noyaux pour faire une couronne et ensuite plusieurs couronnes pour faire un moteur.

Cette forte modularité structurelle du MRV à motifs nous amène à la réflexion suivante : La modélisation du noyau magnétique est fondamentale et constituera la base de tout modèle que l'on voudra créer de ce moteur.

Si il existait un *modèle parfait* de ce noyau, en particulier de la force réluctante qu'il fournit en fonction des Ampères tours que l'on y injecte et de la position, la modélisation d'un moteur à réluctance l'utilisant ne poserait pas de problème car il suffirait de multiplier cette force par un coefficient multiplicateur adéquat dépendant presque uniquement du nombre de noyaux dans le moteur et du nombre de phases.

En termes de conception « industrielle », les moteurs à motifs sont donc assez séduisants. Ils permettent, dans le cas où l'on possède une connaissance assez fine du noyau magnétique, de concevoir un moteur par jeu d'assemblage (un peu comme un jeu de *LEGO*).

3.A.2 *Electronique associée.*

3.A.2.1 *Electronique de puissance.*

La structure du convertisseur qui alimente chacune des trois phases du moteur est celle qui est représentée sur la Figure 3.4 pour une phase [Rapport2].

Signalons que le courant passant dans une phase a une valeur limite que l'on appellera I_{max} . On peut alors distinguer deux cas de fonctionnement selon que le courant dans la phase atteint ou non cette valeur. Nous décrivons le fonctionnement de la structure dans ces deux cas.

Cas où le courant dans la phase n'atteint pas I_{max} :

Pour appliquer la tension positive U aux bornes d'une phase, on commande les deux interrupteurs T_1 et T_2 . Le courant se met donc à croître dans la phase. Lorsque l'on désire annuler le courant, il suffit de couper les deux interrupteurs. Comme la phase est de nature inductive, le courant va circuler par les deux diodes D_1 et D_2 et ainsi imposer une tension négative égale à $-U$ aux bornes de la phase. Comme la

tension appliquée est négative, le courant va décroître jusqu'à annulation. Une fois le courant annulé, la tension reste nulle aux bornes de l'enroulement jusqu'au prochain amorçage de T_1 et T_2 .

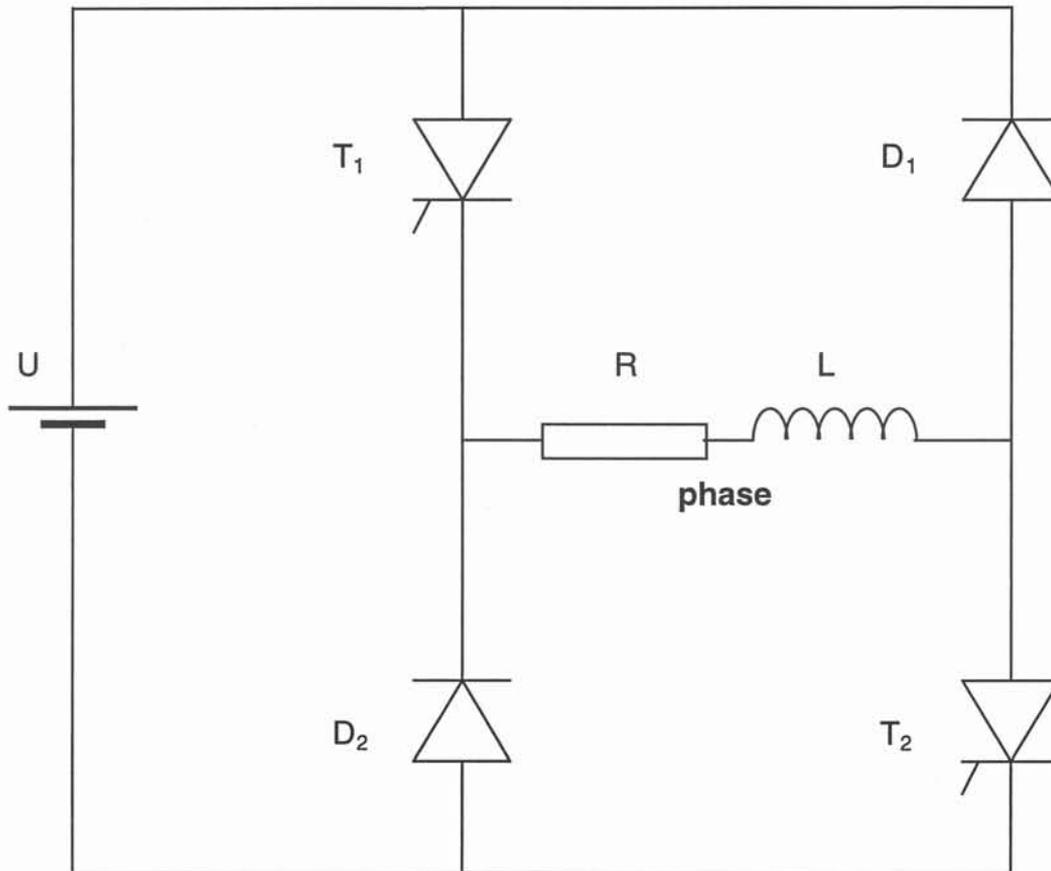


Figure 3.4 : structure du convertisseur alimentant une phase du moteur.

Cas où le courant dans la phase atteint I_{max} :

Au début, le fonctionnement est le même que dans le cas précédent. On commande donc les deux interrupteurs jusqu'à ce que le courant atteigne sa valeur maximale. Ensuite, on laisse l'interrupteur T_2 constamment conducteur et on alterne les phases de conduction et de non-conduction pour T_1 .

- Quand T_1 ne conduit pas, le courant circule dans la roue libre constituée de la diode D_2 et de l'interrupteur T_2 . Durant cette période, la tension aux bornes de l'enroulement est nulle et le courant décroît.
- Quand le courant a suffisamment décré, on réamorçage T_1 . La tension aux bornes de la phase est à nouveau égale à E . Le courant croît de nouveau.

- En alternant ces phases à l'aide d'un comparateur à hystérésis, on maintient le courant à sa valeur maximale jusqu'au moment où l'on doit l'annuler.

Au moment où l'on doit éteindre le courant dans la phase, on coupe les deux interrupteurs et tout se passe comme dans le cas précédent.

3.A.2.2 Electronique de commande.

Le couple moyen délivré par le moteur est fonction des instants d'application des tensions positive et négative aux bornes des phases du moteur. Les convertisseurs alimentant les trois phases doivent être commandés par des signaux qui tiennent compte de la position du rotor par rapport au stator.

La position du rotor par rapport au stator est traduite, pour chacune des phases, par un signal en dents de scie allant de 0 (pour la position alignée) à $2 \cdot E$ (pour la prochaine position alignée de la phase) [Rapport2]. On voit les signaux pour chacune des phases sur la Figure 3.5. La position d'inductance minimale est donc repérée par

la tension E . L'angle θ est donc repéré par l'équation : $\theta = \frac{2 \cdot \pi}{2 \cdot E} \cdot V$

Il s'avère que le calage des capteurs est une tâche très difficile et elle l'est d'autant plus que le nombre de noyaux magnétiques par couronne (phase) est élevé (48 noyaux par phase dans notre cas). En effet, la précision angulaire que l'on peut obtenir sur une pièce donnée en mécanique est fixe alors que le pas polaire angulaire diminue à chaque fois que l'on ajoute un noyau à la couronne. C'est ce qui fait toute la difficulté de la capture de position lorsque l'on a un nombre important de noyaux magnétiques.

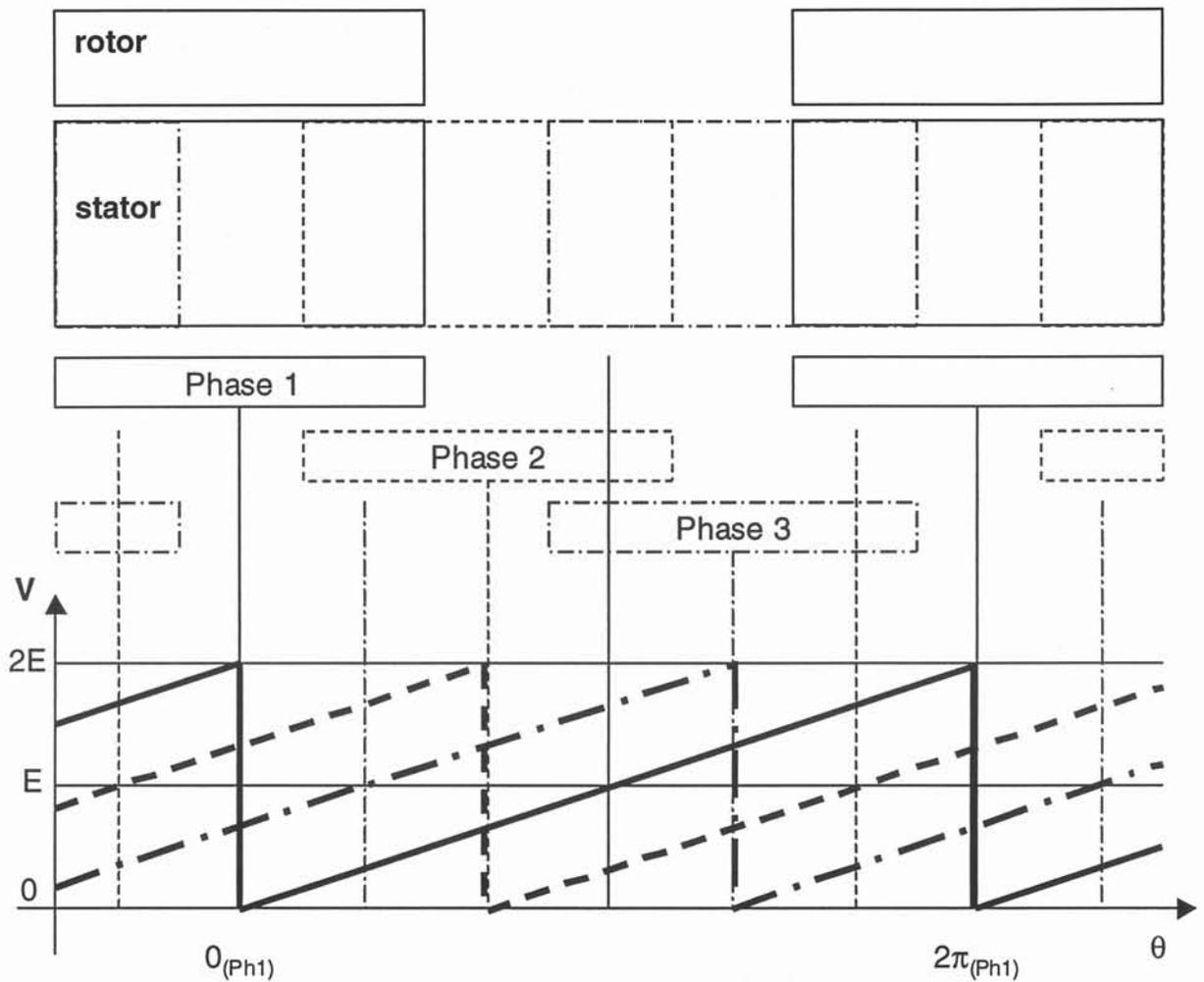


Figure 3.5 : Signaux électroniques représentant la position du rotor.

3.B Modèle analytique.

Le modèle analytique que nous présentons a été construit de façon modulaire et nous permet d'exprimer les performances d'un MRV à motifs en fonction de ses paramètres de construction en deux étapes : La première exprime les paramètres intermédiaires, que sont par exemple l'inductance et la résistance d'une phase du moteur, en fonction des paramètres de construction grâce à des réseaux de réductances. La seconde étape permet d'exprimer les performances du moteur en fonction des paramètres intermédiaires par une résolution symbolique de l'équation différentielle de circuit, grâce à des hypothèses que nous justifierons.

3.B.1 Construction modulaire du modèle.

3.B.1.1 Raison structurelle de ce choix.

Dans la description du moteur qui a précédé, nous avons présenté le noyau magnétique, puis la couronne qui est un assemblage de noyaux magnétiques, et enfin le moteur qui est un assemblage de couronnes. De même, dans la modélisation analytique que nous proposons, il nous a paru logique de distinguer ces trois niveaux d'entrées-sorties.

3.B.1.2 Méthode de création du modèle.

Nous avons voulu décrire le noyau avec ses paramètres géométriques et les ampère tours qui le traversent. Cette manière de décrire le noyau s'inscrit dans notre volonté de modularité. En effet, à ce niveau de description, on ignore quelle sera la bobine qui fournira les ampère tours au circuit magnétique. En revanche, pour la description de la couronne, nous donnons ses paramètres géométriques (comme le diamètre intérieur, par exemple) et sa définition électrique se fait cette fois ci avec une description de bobine (diamètre du fil utilisé, nombre de spires, ...). De même, on définit le moteur comme un assemblage de couronnes (comme indiqué précédemment) et c'est par exemple à ce niveau que l'on donne une valeur maximale au courant pour le fonctionnement à basse vitesse. De ces données, on pourra finalement déduire les ampère tours du noyau.

3.B.1.3 Intérêts pour le modèle.

3.B.1.3.1 Capitalisation.

L'avantage de définir le paramètre électrique du noyau avec les ampère tours est que l'on peut utiliser le même modèle de noyau si on choisit de bobiner une couronne avec du fil plat. On n'aura rien à modifier au niveau du modèle de noyau. On fera simplement un autre modèle de couronne (par exemple) qui diffèrera du précédent au niveau du fil employé.

Ainsi, le modèle du noyau est capitalisé dans le sens où, dès que l'on décide de changer, sur une structure existante, quelque chose de non relatif au noyau, on peut reprendre son modèle tel qu'il était précédemment sans rien avoir à changer.

3.B.1.3.2 Maintenance.

Un autre avantage considérable qu'offre la modularité à un modèle est sa facilité de « maintenance ». En effet, il semble plus facile d'aller corriger une erreur à un endroit lorsque celui ci peut être localisé par éliminations successives de certaines parties de modèle.

Cette démarche rappelle un peu celle d'un programmeur qui traque une erreur dans son programme. Il peut procéder par dichotomie en testant successivement les différentes parties de son programme jusqu'à localiser de plus en plus finement l'endroit où elle se cache.

3.B.1.4 Intérêts pour le processus de dimensionnement.

Les avantages qu'offre la construction modulaire d'un modèle ne se limite pas seulement à celui ci. Nous allons voir qu'elle permet également de gagner du temps dans la modélisation de nouveaux produits de même qu'elle permet d'en gagner dans la création d'un outil de dimensionnement.

3.B.1.4.1 Nouveaux produits.

Un autre avantage à la définition modulaire du modèle est qu'il offre la possibilité d'être réutilisé, en partie, dans la modélisation rapide de nouveaux produits. Cette modélisation peut être très rapide quand on reprend exactement un motif déjà utilisé.

Ainsi, si l'on souhaite utiliser le même noyau magnétique dans une autre structure (un actionneur avec un seul noyau, par exemple), l'avantage d'avoir pensé à sa définition en faisant le moins d'hypothèses possible est que l'on pourra reprendre, là aussi, intégralement, le même modèle pour le noyau (puisque'il aura déjà été utilisé pour le MRV).

En fait, on peut faire ici un parallèle entre l'avantage qu'offre la modularité du modèle pour le dimensionnement de nouveaux produits que celui qu'il offre pour le modèle en termes de capitalisation. Ces deux avantages sont sensiblement liés.

3.B.1.4.2 Création d'un outil de dimensionnement.

De la même manière qu'un modèle construit de façon modulaire peut être réutilisé dans la modélisation de nouveaux produits, sa réutilisation dans un environnement de dimensionnement est facilitée dans la mesure où le programme « calcul de sensibilité » existe déjà également. L'intérêt pour le processus de dimensionnement est alors double : Il dispose à la fois d'un programme d'analyse relatif à un modèle connu et déjà utilisé et du programme de calcul de sensibilité correspondant.

Tout cela se traduit par un gain de temps, tant dans la partie de création du modèle que dans celle de création de l'outil de dimensionnement.

3.B.2 Modélisation du comportement statique.

3.B.2.1 Hypothèses.

Les hypothèses qui sont faites pour ce modèle du comportement statique du moteur sont celles des réseaux de réluctances sur lesquels ce modèle sera fondé. Elles sont les suivantes :

- **Les réluctances du fer sont linéaires.** Cela implique que la saturation magnétique du noyau de fer ne sera pas prise en compte. La conséquence importante que cette hypothèse induit sur le modèle est que l'inductance d'une phase, pour une position donnée, est indépendante du courant [$L(i) = \text{Constante}$].
- **Ce modèle est un modèle statique.** Nous supposons donc qu'à chaque instant, la position du noyau varie très peu par rapport à l'évolution des grandeurs électriques. Cette hypothèse sera donc de plus en plus fautive à mesure que l'on voudra utiliser ce modèle à des vitesses élevées. Elle nous interdit en particulier la prise en compte des courants de Foucault.

3.B.2.2 Description du modèle.

Ce modèle est constitué de plusieurs sous modèles qui se font appel les uns aux autres (Figure 3.6). Il est donc modulaire. Ainsi, au niveau du modèle principal construit sur ces fondations, on donne les valeurs des paramètres de construction d'une couronne et on obtient les valeurs des paramètres intermédiaires que sont les différentes valeurs d'inductances et la résistance d'une phase.

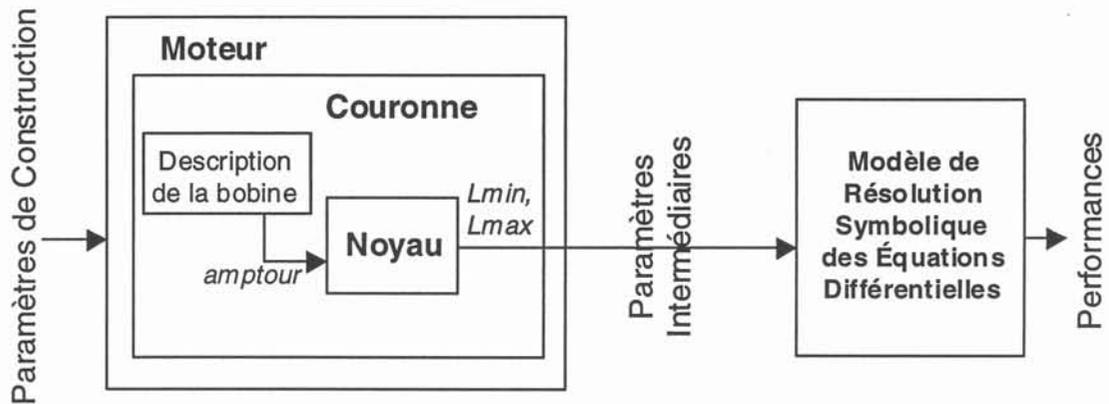


Figure 3.6 : représentation schématique de la modularité du modèle.

3.B.2.2.1 Modèle de noyau par réseaux de ré reluctances.

Le modèle de noyau est celui dans lequel on trouve vraiment la modélisation avec les réseaux de ré reluctances car c'est celui que l'on souhaite pouvoir réutiliser quelle que soit l'application.

On donne ci-dessous les interfaces d'entrée et de sortie de ce modèle en distinguant les paramètres selon quelques catégories.

Paramètres d'entrée :

Paramètres géométriques :

dr :	hauteur de la dent rotor.
ds :	hauteur de la dent stator.
l _{bob} :	distance minimale entre les conducteurs et l'entrefer.
prof :	largeur de l'encoche.
ent :	entrefer.
l _{fer} :	épaisseur du circuit magnétique.
ts :	profondeur du noyau.

Paramètres électriques :

amptour : force magnétomotrice passant dans l'encoche.

Constantes utilisées :

muzero : perméabilité relative du vide.
 mvfer : masse volumique du fer.
 ndiv : nombre de subdivisions dans les éléments de Roters [ROTERS-41].

Paramètres de sortie :

masseun : masse d'un noyau
 secbob : section d'encoche.
 bmin : induction dans la position d'inductance minimale (servira au calcul de L_{min}).
 bmax : induction dans la position d'inductance maximale (servira au calcul de L_{max}).
 + les différentes réluctances calculées dans ce modèle que nous ne détaillerons pas ici.

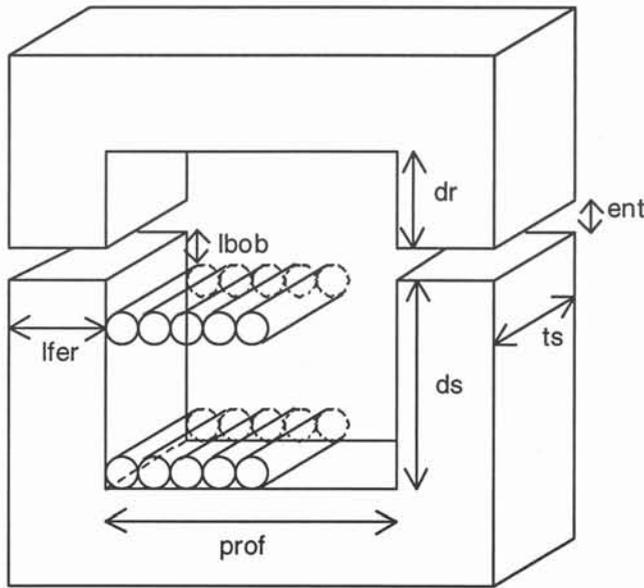


Figure 3.7 : visualisation des paramètres géométriques du noyau.

Ce modèle présuppose la forme des lignes de champ dans l'air et utilise des éléments de géométries prédéfinies comme des jeux de **LEGO** pour construire une géométrie conforme à ce qu'il suppose. Ensuite, comme les réluctances de ces géométries sont connues, il suffit d'utiliser les règles de somme de réluctances pour trouver la réluctance totale du circuit magnétique. On trouve la base de cette théorie dans [ROTERS-41]. On retrouvera des formules se basant sur cette théorie dans [GENTILHOMME-91] et [DESHPANDE-95].

On établit un réseau de réluctances pour chaque position nous intéressant (L_{min} et L_{max}), et, pour chacune de ces positions, la réluctance totale sert ensuite à calculer les inductions respectives dans le noyau. Pour notre modèle, on retrouve les deux valeurs d'induction (b_{min} et b_{max}) correspondant aux deux positions particulières de la couronne.

Nous donnons avec la figure ci-dessous un exemple de réseau de réluctances pour la position d'inductance minimale.

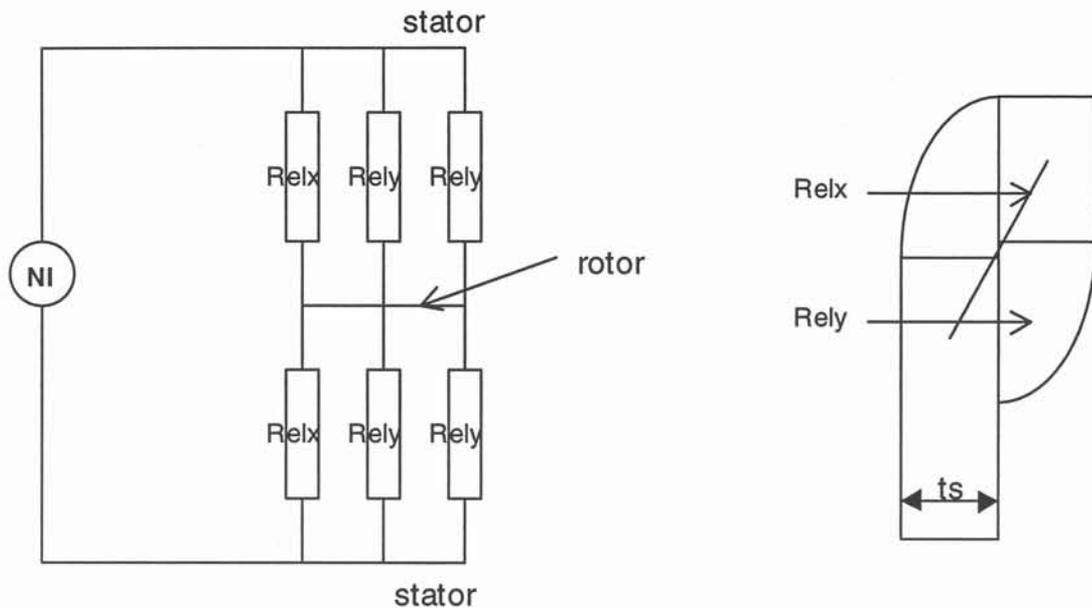


Figure 3.8 : Réseau de réluctances pour la position d'inductance minimale.

3.B.2.2.2 *Modèle de couronne.*

Dans ce modèle, on utilise le modèle de noyau décrit plus haut pour dimensionner une couronne faite d'un nombre de noyaux $nnoyaux$ déduit du rayon d'entrefer $rent$ et de la profondeur du noyau ts grâce à la relation suivante :

$$nnoyaux = \frac{1}{2} \cdot \frac{2\pi}{\arctan\left(\frac{ts}{rent}\right)}$$

C'est ce modèle qui donne les valeurs des inductances minimale et maximale ainsi que la valeur de la résistance d'une phase.

Le modèle de couronne est un modèle qui a les interfaces d'entrée et de sortie suivantes :

Paramètres d'entrée :

On retrouve parmi les paramètres d'entrée du modèle de couronne ceux du modèle de noyau à l'exception de μ dont la valeur sera calculée à l'aide des paramètres d'entrée spécifiques à la couronne décrite.

Paramètres géométriques :

dfil : diamètre des fils.
 nspires : nombre de spires.
 rent : rayon d'entrefer.
 inter : coefficient donnant l'interstice entre deux noyaux en fonction de ts.

Paramètres électriques :

tension : tension aux bornes de la bobine.
 intens : intensité maximale dans les fils.

Constantes utilisées :

rhocuv : résistivité du cuivre.

Paramètres de sortie :

On retrouve également ceux du modèle de noyau avec cette fois-ci, amptour qui est calculée à l'aide des paramètres électriques et géométriques de la couronne.

Paramètres électriques :

amptour : force magnétomotrice donnée en entrée de la boîte de calcul noyau.

Paramètres géométriques :

te : Interstice entre deux noyaux au niveau de l'entrefer.
 tssrent : valeur du rapport ts/rent.
 nnoyaux : nombre maximal de noyaux que l'on peut mettre dans la couronne.
 rstat : rayon de base des noyaux stator.
 rrot : rayon de base des noyaux rotor.
 rint : rayon intérieur de la couronne.
 rext : rayon extérieur de la couronne.
 resist : résistance du bobinage.
 dencou : densité de courant dans les conducteurs.
 rempli : coefficient de remplissage de l'encoche.
 masferc : masse de fer de la couronne.
 indmin et indmax : inductances caractéristiques de la couronne.

On définit la couronne avec les paramètres géométriques et électriques qui la caractérisent vraiment.

3.B.3 Modélisation du comportement dynamique.

La modélisation du comportement dynamique du moteur revient à résoudre l'équation de circuit qui s'exprime sous la forme de l'équation différentielle suivante :

$$E = r.i(t) + (L + \lambda.\omega.t) . di(t)/dt , \text{ avec } r = resist \pm \lambda\omega.$$

Nous sommes obligés de résoudre cette équation de circuit pour connaître les performances du moteur, car le couple dépend directement du courant.

3.B.3.1 Hypothèses.

Les hypothèses qui ont été faites pour construire ce modèle de résolution symbolique de l'équation de circuit sont les suivantes :

- **L'inductance d'une phase en fonction de la position est linéaire.** Cette hypothèse est assez courante et ne s'éloigne de la réalité qu'aux positions d'inductance minimale et maximale. Les valeurs de l'inductance minimale et maximale sont indépendantes du courant. En ces points, on n'a pas les points anguleux de notre modèle mais plutôt des arrondis, comme on peut le voir sur la Figure 3.9. En cas de forte saturation dans la position alignée, l'arrondi sera plus nettement prononcé.

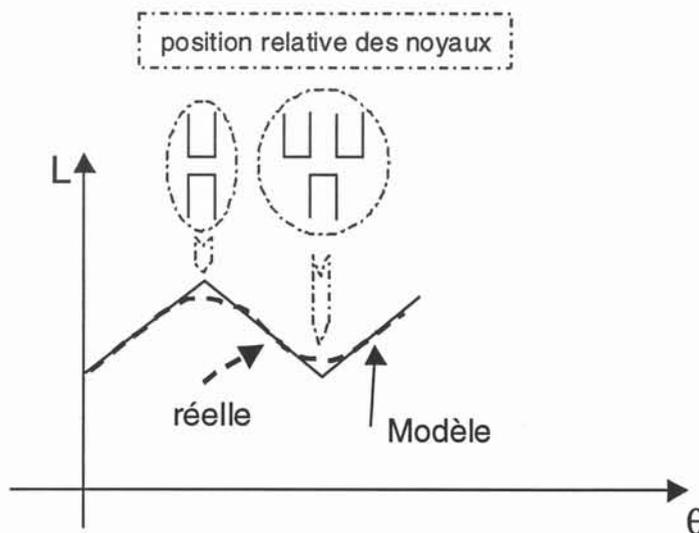


Figure 3.9 : Inductance du MRV à motifs en fonction de la position

Cette hypothèse de linéarité de l'inductance en fonction de la position est bienvenue dans la mesure où les réseaux de réluctances nous donnent des valeurs d'inductance constantes qui, justement, ne dépendent pas du courant. Le couplage du modèle de résolution symbolique des équations différentielles de circuit avec les réseaux de réluctances est donc parfaitement opportun.

- **Ce modèle est un modèle dynamique.** Comme il consiste à résoudre une équation différentielle, ce modèle est dynamique car les valeurs de ses variables sont différentes dans le temps.

3.B.3.2 Description du modèle.

Ce modèle doit prendre comme interface d'entrée des paramètres intermédiaires du modèle qui sont fournis par les réseaux de réluctances via le modèle de couronne et fournir en interface de sortie les performances du moteur, comme le couple, par exemple.

Ses paramètres d'entrée sont les suivants :

Nrpm : vitesse angulaire du rotor (en tr/min).
U : tension aux bornes d'une phase (en V).
Imax : Courant maximum autorisé pour la régulation (en A).

Ndent : nombre de noyaux sur une phase.
Lmax : Inductance maximale d'une phase (en henry).
Lmin : Inductance minimale d'une phase (en henry).
R : résistance du bobinage d'une phase (en ohms).

Frac_ON : fraction définissant l'angle d'allumage des transistors.
Frac_OFF : fraction définissant l'angle d'extinction des transistors.

Et les paramètres de sortie sont les suivants :

Itotal(θ ,Nr) : intensité en fonction du temps et de la vitesse.
Courant_Moy_Phase : courant moyen dans la phase.
Couple(θ ,Nr) : couple dans la phase en fonction du temps et de la vitesse.
Couple_Moy_Stack : couple moyen dans la phase.

Courant_Batt(θ ,Nr) : courant issu de la batterie.
Couple_Mot(θ ,Nr) : couple délivré par le moteur.
Puissance_Moteur(Nr) : puissance du moteur.

A l'aide de ces paramètres, le modèle calcule les formes d'onde du courant avec la solution symbolique de l'équation différentielle qui se pose en fonction de la séquence dans laquelle on se trouve.

La période électrique est en effet divisée en 4 sous séquences qui sont représentées sur la Figure 3.10.

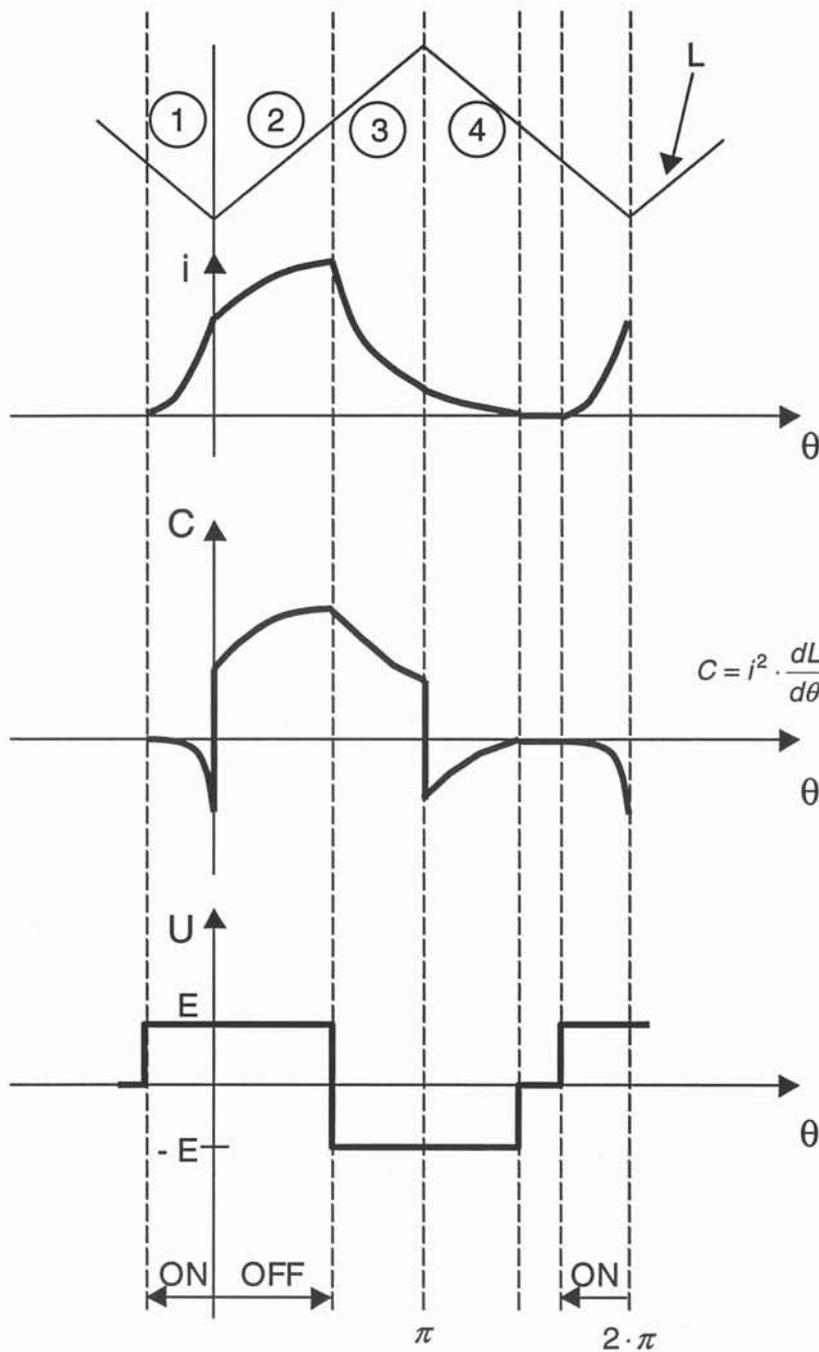


Figure 3.10 : Formes du courant, du couple et de la tension pour les différentes séquences de résolution.

- 1) de ON à 0 : L'inductance est décroissante depuis l'instant d'allumage (que l'on choisit) jusqu'à sa valeur minimale. Le courant commence donc à s'établir dans la phase et crée, pendant cette séquence, un couple résistant (négatif).

- 2) de 0 à OFF : L'inductance est croissante. On a choisi d'éteindre l'alimentation à OFF. Le courant continue donc à croître dans la phase, mais crée maintenant un couple positif parce que l'inductance est croissante.
- 3) de OFF à π : L'inductance est encore croissante jusqu'à sa valeur maximale, donc le couple sera moteur, mais le courant dans la phase commence à décroître parce que l'on a coupé ou inversé la tension. Ainsi, la première valeur de couple négatif sera moins élevée que si on avait coupé à π électrique.
- 4) de π à l'extinction du courant : ($\theta_{\text{extinction}} < 2\pi\text{-ON}$) L'inductance recommence à décroître, donc le couple redevient négatif. Le courant décroît jusqu'à devenir nul. La tension, si elle était inversée de sorte à faire décroître plus vite le courant, s'annule en même temps que celui-ci. Le courant doit s'annuler avant $2\pi\text{-ON}$ pour que la conduction soit bien discontinue et que notre modèle soit bien dans ses conditions d'application.
- La différence qui existe entre les 4 équations différentielles à résoudre en fonction de la séquence dépend, comme les 4 séquences, du sens de l'inductance et de la valeur de la tension (+E ou -E).

Elles ont la forme suivante :

$$\pm E = r \cdot i(t) + (L \pm \lambda \cdot \omega \cdot t) \cdot \frac{di}{dt}$$

↑

↑

signe de la tension croissance ou décroissance de l'inductance
en fonction de la position ($\theta = \omega \cdot t$)

- L'intérêt de fermer plus tôt et d'ouvrir plus tôt les interrupteurs est d'augmenter la puissance du moteur à grande vitesse par rapport à un allumage à 0 et une coupure à π . en effet, allumer plus tôt permet de faire monter le courant à une certaine valeur (supérieure à 0) au moment où l'inductance change de signe. Cela permet d'obtenir, dès cet instant, un couple positif qui ne part pas d'une valeur nulle. On gagne ici au niveau du

couple moyen. Un raisonnement analogue permet de comprendre pourquoi on ouvre les interrupteurs avant π électrique aux grandes vitesses. Cela permet d'arriver dans la zone de couple négatif avec un courant réduit par rapport à sa valeur maximale.

3.C Modèle numérique.

Le modèle numérique que nous présentons exprime les performances d'un MRV à motifs en deux étapes, en passant lui aussi par des paramètres intermédiaires qui correspondent à la même connaissance mais ne sont pas rigoureusement identiques à ceux que nous venons de voir. La première étape de ce modèle est le calcul par éléments finis d'une matrice exprimant le flux circulant dans un noyau en fonction de la position relative des noyaux et des Ampère tours circulant dans la fenêtre du stator. La seconde étape consiste à résoudre l'équation différentielle de circuit grâce à la méthode de Runge-Kutta

3.C.1 Modélisation par éléments finis.

On réalise plusieurs simulations éléments finis pour obtenir à chaque fois une valeur de flux circulant dans le circuit magnétique constitué des deux demi noyaux en fonction de différentes positions et de différents niveaux de courant (d'Ampère tours) dans la fenêtre stator. Nous allons voir plus loin quel est le modèle que nous avons choisi d'utiliser. Nous avons choisi de prendre 10 valeurs d'Ampère tours et 10 valeurs de position. Ainsi, nous devons réaliser 100 points de simulation. On obtient ces 100 valeurs grâce à un fichier batch.

3.C.1.1 Hypothèses

La saturation du circuit magnétique est modélisée. En effet, le logiciel que nous utilisons permet de prendre en compte les courbes de saturation des matériaux magnétiques [Flux2D].

Ce modèle est un modèle statique. Nous avons choisi d'utiliser le module de magnétostatique du logiciel éléments finis. Donc, comme pour le modèle analytique, la prise en compte des courants de Foucault nous sera impossible.

3.C.1.2 Difficulté d'obtenir la courbe $\Phi(i,\theta)$

L'obtention de cette matrice de valeurs est difficile. En effet, **plusieurs modélisations 2D du noyau sont possibles** et il va nous falloir choisir. Nous pensons qu'aucune ne sera complètement satisfaisante parce que nous jugeons que notre problème est typiquement 3D. Les effets de frange [BOUCHARD-95] dans la troisième dimension ne nous apparaissent pas comme négligeable, contrairement aux hypothèses que l'on fait habituellement lorsque l'on utilise un logiciel éléments finis 2D.

Nous avons choisi une modélisation particulière parmi celles que nous allons présenter dans le paragraphe suivant. Si notre choix se révélait mauvais, ce ne serait pas dramatique pour deux raisons particulières :

- Il nous faut un modèle qui soit capable de nous donner une matrice $\Phi(i,\theta)$. En effet, nous avons bien besoin de cette information pour développer la deuxième partie du modèle qui utilisera cette information en interface d'entrée. Quel que soit ce modèle choisi, il satisfera à cette exigence.
- Une fois que la deuxième partie du modèle est développée, on peut utiliser un autre modèle (meilleur que le premier) pour obtenir une nouvelle matrice $\Phi(i,\theta)$. Si le modèle développé fonctionnait avec la première, il fonctionnera avec la deuxième.

L'essentiel, pour cette étape du modèle, est d'arriver à obtenir une surface $\Phi(i,\theta)$ à partir de la géométrie.

3.C.1.3 Réflexion sur les différents modèles possibles.

Comme nous pensons que notre problème est typiquement tridimensionnel en raison des dimensions du circuit magnétique. Il nous semble que, pour avoir une modélisation correcte, on ne peut pas se passer d'une simulation 3D. Néanmoins, comme nous avons conscience des limites de la modélisation 2D et que nous n'avons finalement besoin que d'une surface $\Phi(i,\theta)$, nous avons réfléchi aux différents modèles qu'il nous semblait possible de réaliser avec Flux2D. Nous allons présenter trois géométries avec les figures suivantes et exposer brièvement les avantages et les inconvénients que nous leurs avons trouvés.

3.C.1.3.1 Géométrie « Kruse ».

La géométrie présentée ici présente l'avantage de donner directement la force qui s'exerce sur les noyaux en fonction du courant. On crée une zone de perméabilité infinie de sorte à « transporter » le flux sans pertes d'ampère tours. Cela nous permet de ramener la définition du courant dans le plan qui nous convient. Cette géométrie est donc fonctionnelle.

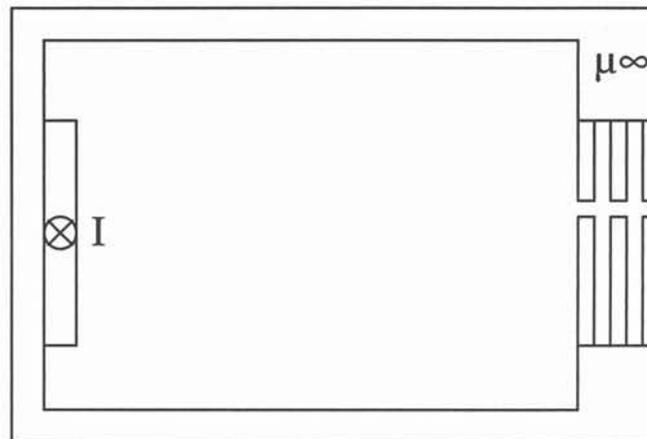


Figure 3.11 : Géométrie « Kruse ».

Son inconvénient majeur est qu'elle ne permet pas l'utilisation d'une bande de glissement. D'autre part, on effectue un maillage inutile dans toute la zone située à l'intérieur du circuit magnétique. Ce maillage inutile allonge le temps de calcul. De plus, on est obligé de modéliser plus d'un noyau pour modéliser l'influence des autres dans une position intermédiaire.

La perméabilité « infinie » ne l'est pas vraiment. On lui attribue une valeur grande devant les autres données du problème. L'acheminement du flux dans cette région devient donc source d'erreur puisqu'on modélise quelque chose qui n'existe pas vraiment. Même si nous pensons que l'erreur commise est négligeable, nous ne sommes pas en mesure de le prouver.

3.C.1.3.2 Géométrie « Dirichlet ».

Pour ce modèle, on décrit trois noyaux pour modéliser la force s'exerçant sur celui du milieu en fonction de la position. La condition de Dirichlet est imposée sur les bords en traits gras sur la Figure 3.12 de sorte à imposer une valeur de flux

circulant dans le sens vertical. Cette géométrie nous permettra donc de calculer la force en fonction du flux (et non plus du courant) circulant dans le circuit magnétique.

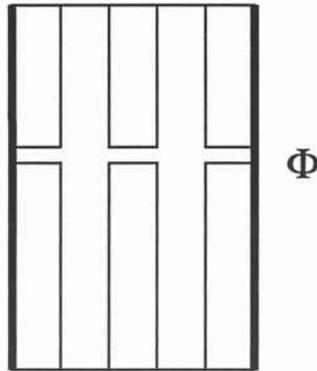


Figure 3.12 : Géométrie « Dirichlet ».

Les avantages de cette géométrie sont qu'elle permet d'utiliser une bande de glissement et qu'elle nécessite un maillage beaucoup moins important que la géométrie précédente.

Son principal inconvénient est qu'elle est peu fonctionnelle. Directement, on a la force en fonction du flux, et pour connaître le flux en fonction du courant, il faut modéliser tout le circuit magnétique. De plus, les conditions de Dirichlet nous obligent à modéliser plus d'un noyau. On pourrait ajouter une demi dent du côté où il le faut pour modéliser mieux le parcours du flux dans la position d'inductance minimale (Figure 3.13) mais cela alourdirait encore le travail.

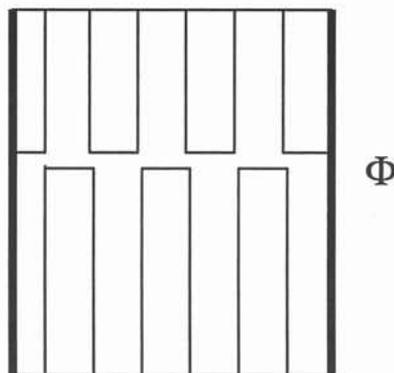


Figure 3.13 : Inconvénient de la géométrie « Dirichlet » dans la position d'inductance minimale.

3.C.1.3.3 Géométrie « Dirichlet » cyclique.

Pour ce modèle, on ne décrit plus qu'un seul noyau car, cette fois-ci, des conditions aux limites cycliques sur le bord du domaine sont venues remplacer les

conditions de Dirichlet. La condition de Dirichlet est aussi représentée en traits gras sur la Figure 3.14 de sorte à imposer une valeur de flux circulant dans le sens vertical. Cette géométrie nous permettra, comme sa précédente, de calculer la force en fonction du flux.

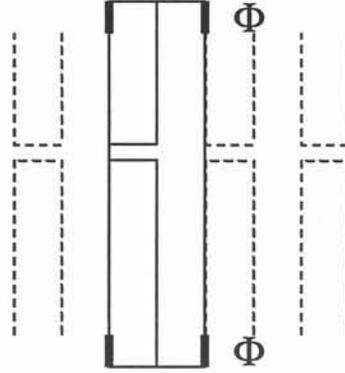


Figure 3.14 : Géométrie « Dirichlet » cyclique.

Le maillage est encore réduit avec cette géométrie puisqu'on ne maille plus qu'un seul noyau. On peut toujours y utiliser la bande de glissement.

Son inconvénient reste qu'elle est toujours aussi peu fonctionnelle que la géométrie « Dirichlet ».

3.C.1.3.4 Choix du modèle.

Nous avons finalement choisi, parmi ces trois modèles, celui qui exprime directement le flux en fonction du courant (géométrie « Kruse ») parce qu'il nous semble plus pratique à utiliser pour lancer les batchs. Il nous a permis d'obtenir les résultats qui ont servi de base au développement de la seconde partie du modèle.

3.C.1.4 Approche modulaire.

Nous avons adopté une approche modulaire aussi pour le modèle numérique que nous avons construit, même si elle est moins évidente. En effet, la première partie du modèle consiste à donner une surface $\Phi(i, \theta)$ en sortie pour une géométrie donnée en entrée.

Nous avons commencé à travailler avec un modèle qui nous paraît le plus facile à utiliser, mais la simplicité de l'interface de sortie nous permettra d'utiliser le modèle

éléments finis (2D ou 3D) de notre choix plus tard, si on veut éventuellement améliorer la précision sur cette surface.

On pourra même, pour un objectif industriel, utiliser des prototypes pour caractériser cette courbe et utiliser l'expérience acquise sur un prototype pour dimensionner d'autres structures.

Ce qu'il est intéressant de signaler dans notre approche du problème n'est pas , à notre avis, de savoir quel est le modèle éléments finis qu'il aurait été judicieux ou on de prendre, mais plutôt **la modularité**. En effet, l'information qu'il nous intéresse de connaître est la valeur de l'inductance en fonction du courant et de la position angulaire du rotor. Cette information est contenue dans la surface $\Phi(i, \theta)$.

Notons qu'elle correspond bien à ce que nous avons défini comme paramètres intermédiaires dans le modèle analytique comme L_{\min} et L_{\max} . La différence dans ce cas est que l'on connaît non seulement l'inductance en fonction de la position, mais également que cette inductance n'est plus constante en fonction du courant.

3.C.2 Résolution de l'équation différentielle de circuit avec la méthode de Runge Kutta.

3.C.2.1 Hypothèses.

Les hypothèses qui sont faites pour ce modèle numérique de résolution de l'équation différentielle de circuit avec la méthode Runge Kutta sont réduites au minimum. Nous avons en interface d'entrée une surface $\Phi(i, \theta)$ comme celle qui est présentée Figure 3.15 et nous allons « nous promener dessus » à chaque pas de l'équation différentielle.

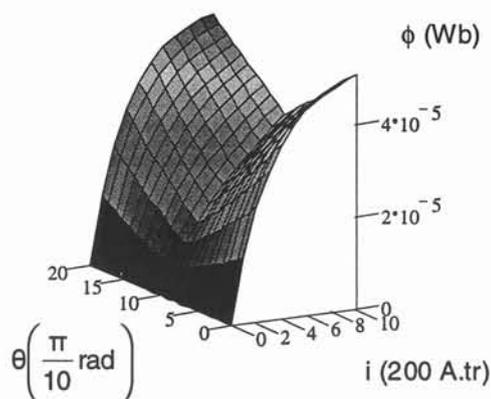


Figure 3.15 : Surface $\Phi(i, \theta)$ permettant la résolution de l'équation différentielle avec la méthode de Runge Kutta

3.C.2.2 Description du modèle.

Ce modèle prend en compte comme interface d'entrée la matrice $\Phi(i, \theta)$ qui constitue les paramètres intermédiaires du modèle et calcule les performances du moteur grâce à quelques paramètres supplémentaires fournis en entrée.

Notre modèle de résolution d'équation différentielle de circuit avec la méthode de Runge Kutta a les interfaces d'entrée et de sortie suivantes :

Paramètres d'entrée :

Matrice des points calculés dans Flux2D. Courbe en 3D $\phi(i, \theta)$.

- resist : résistance du bobinage de la phase (en ohms).
- omega : vitesse angulaire du rotor (en rad/s).
- ubatt : tension disponible aux bornes de la batterie (en V).
- Imax : Limitation de courant pour les basses vitesses (en A).
- nspires : nombre de spires de la bobine.
- nnoyaux : nombre de noyaux dans la couronne.
- nstack : nombre de phases du moteur.
- Nper : nombre de périodes de visualisation des signaux.
- Frac_ON : fraction définissant l'angle d'allumage des transistors.
- Frac_OFF : fraction définissant l'angle d'extinction des transistors.
- npts : nombre de points pris pour la résolution avec Runge-Kutta.

Paramètres de sortie :

- tension(t) : tension de commande (définie en fonction des angles de commande).
- i(t) : intensité en fonction du temps.

Couple_Moy_Stack :	couple moyen fourni par une couronne.
Couple_Moteur :	couple fourni par le moteur en fonction du temps.
Courant_Moy_Phase :	courant moyen dans une couronne.
Courant_Batterie :	courant fourni par la batterie en fonction du temps.
Couple_Moyen_Moteur :	couple moyen fourni par le moteur.
Courant_Moyen_Batterie :	courant moyen fourni par la batterie.
Puissance_Moteur :	puissance fournie par le moteur.

La première partie de ce modèle consiste à transformer une matrice contenant 100 points de calcul en une surface non bosselée (ce qui interdit l'usage de splines). Cette exigence se justifie par le besoin que nous avons plus loin des dérivées de cette surface par rapport à ces deux axes (i et θ) pour résoudre notre équation différentielle. Le modèle que nous avons utilisé pour approcher notre surface est le modèle de Torrey-Lang [TORREY-90] dont la formule est donnée ci-dessous :

$$\Phi(i, \theta) = a_1(\theta) \cdot [1 - e^{a_2(\theta)i}] + a_3(\theta) \cdot i$$

Ensuite, une fois que la surface est obtenue, on passe à la résolution proprement dite. Celle-ci est préprogrammée dans le logiciel Mathcad que nous utilisons pour développer notre modèle. Il n'a besoin que de valeurs initiales pour les variables d'état que nous définissons comme le courant et la position et de l'expression symbolique des dérivées de ces variables par rapport au temps. Il faut aussi lui indiquer la durée sur laquelle il doit résoudre l'équation ainsi que le nombre de pas qu'on lui accorde pour cela.

Il reste enfin l'exploitation des résultats de la résolution. ceux-ci sont données dans un tableau de points. On peut tirer de ce tableau des valeurs de i et de θ en fonction du temps. Seul i nous intéresse. De cette courbe $i(t)$, on tire la courbe du couple en fonction du temps grâce à la formule : $C(t) = \frac{i}{2} \cdot \frac{d\Phi}{d\theta}$.

Un calcul de moyenne sur $C(t)$ nous donne la valeur du couple moyen. On obtient donc la puissance du moteur en multipliant cette valeur par la vitesse que l'on donne en paramètre d'entrée.

3.D Proposition d'une méthode de dimensionnement.

Nous avons vu au cours de ce développement au moins deux manières de modéliser un moteur à réluctance variable à motifs. Le modèle analytique est plus facile et plus rapide à développer que le modèle numérique. Cependant, et bien que

nous n'ayons pas pour le moment les éléments pour le prouver, nous pensons que le modèle numérique sera plus précis. Nous allons maintenant proposer une méthode de dimensionnement s'appuyant sur cette base de réflexion.

3.D.1 Comparaison des deux modèles

Pour comparer nos deux modèles (analytiques et numérique), nous avons choisi de tracer les deux courbes de couple moyen en fonction de la vitesse pour un moteur identique dans chacun des deux modèles (Figure 3.16).

Il est important de préciser de quelle manière sont obtenues ces courbes. En effet, pour une vitesse donnée, la valeur de couple que calcule le modèle est très fortement dépendante des valeurs données aux angles d'allumage et d'extinction des interrupteurs. Pour l'exemple que nous présentons, nous avons réalisé une mini optimisation « à la main » avec les angles ON et OFF de sorte à obtenir le couple maximal pour chacun des deux modèles, et ce, pour chaque point de calcul. Il en résulte donc que pour une même vitesse, les angles d'allumage et d'extinction ne sont pas nécessairement identiques pour les deux modèles (Figure 3.17).

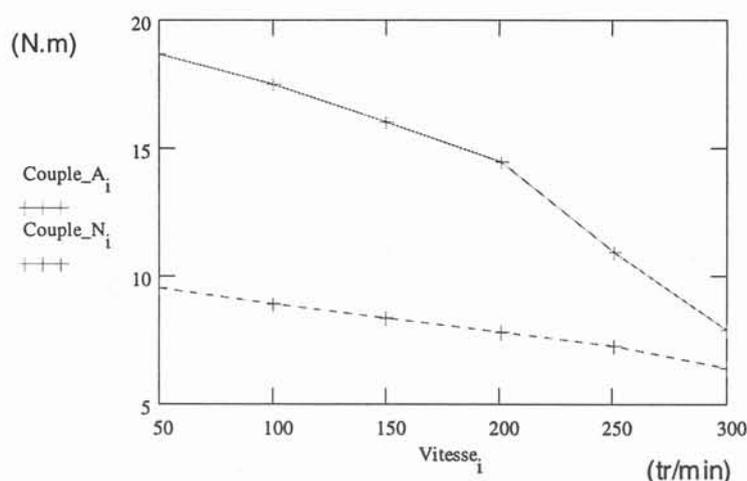


Figure 3.16 : Courbes de couple moyen maximum = f(vitesse) données par les 2 modèles

On constate que les deux modèles nous donnent deux courbes différentes, la courbe du modèle analytique donnant des valeurs de couple pratiquement deux fois plus grandes que celles du modèle numérique.

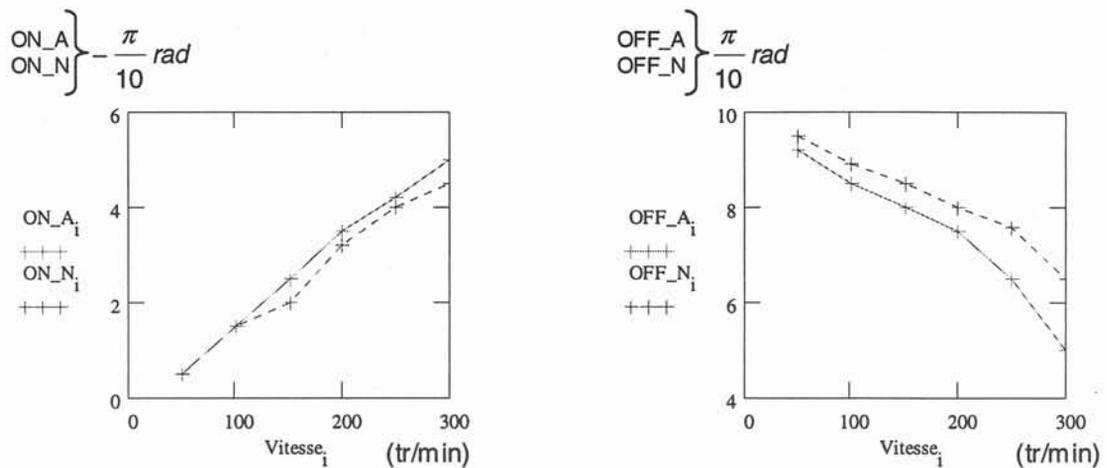


Figure 3.17 : Courbes des angles d'allumage et d'extinction des interrupteurs en fonction de la vitesse

Sur les courbes donnant l'évolution des angles d'allumage et d'extinction trouvés en fonction de la vitesse, on remarque qu'ils ont sensiblement les mêmes valeurs (Figure 3.17). Dans les deux cas, ils évoluent dans le même sens. Cette information peut paraître superflue, mais comme nous n'avons pas d'éléments de validation de notre modèle, toutes les informations convergentes données par les deux modèles sont autant de présomptions favorables.

3.D.2 Plusieurs sources d'incertitudes dans les modèles.

Avant de présenter la méthode de dimensionnement que nous proposons avec ces deux modèles, nous allons donner les sources d'incertitudes qui les concernent.

Le premier modèle que nous avons présenté est le modèle analytique. La première partie de ce modèle est l'utilisation de réseaux de réductances. Les fortes hypothèses qu'ils supposent comme la linéarité du fer sont autant de sources d'incertitude sur le résultat final. De même, la seconde partie du modèle utilise une hypothèse de totale linéarité de l'inductance en fonction de la position.

Ainsi, les hypothèses faites sur le modèle analytique ne nous permettent pas de valider sa précision.

Le second modèle que nous avons présenté est le modèle numérique. Nous avons précisé lorsque nous l'avons présenté que notre problème est typiquement

tridimensionnel. Nous pensons que l'utilisation d'un logiciel éléments finis 2D nous introduit une erreur de précision finale sur notre modèle qu'il nous est impossible d'évaluer. En perspective de ce travail, une campagne de simulation 3D répondrait sans doute à cette question. La résolution de l'équation différentielle de circuit avec la méthode de Runge Kutta n'introduit pas d'hypothèses supplémentaires sur le modèle. Elle s'appuie complètement sur la surface $\Phi(i, \theta)$ donnée par les éléments finis.

En conséquence, la qualité du modèle numérique que nous avons développé dépend quasiment totalement de la qualité de la surface $\Phi(i, \theta)$. Ne pouvant pas valider cette qualité, nous ne sommes pas en mesure de garantir la précision de ce modèle numérique.

3.D.3 Méthode de dimensionnement proposée.

La méthode de dimensionnement que nous proposons est représentée schématiquement sur la Figure 3.18. Elle comporte quatre étapes principales que nous allons développer dans les paragraphes suivants :

- Le rapprochement des deux modèles complémentaires que sont le modèle analytique et le modèle numérique.
- L'application de dimensionnement avec le premier modèle (analytique) et la méthodologie générique de dimensionnement présentée en détails dans le deuxième chapitre.
- La prise de décision après cette optimisation et la consultation du deuxième modèle et la construction d'un prototype.
- Les essais sur le prototype construit et la correction éventuelle des modèles qui en résulte.

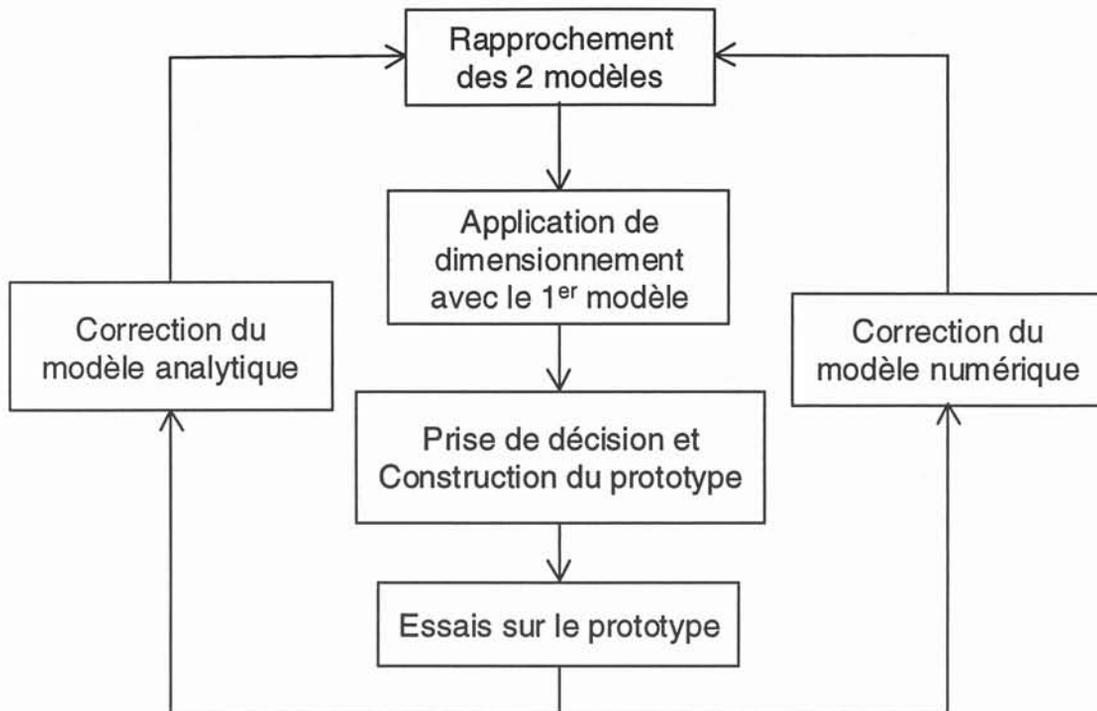


Figure 3.18 : Graphe de la méthode de dimensionnement proposée.

3.D.3.1 Rapprochement des deux modèles.

Nous venons de voir dans le paragraphe précédent que nous ne pourrions valider ni la précision du modèle analytique, ni celle du modèle numérique. Cependant, nous avons quand même deux modèles à notre disposition. Même si ces deux modèles ne donnent pas exactement les mêmes courbes de couple en fonction de la vitesse, nous avons remarqué que les informations qualitatives qu'ils donnent sont dans le même sens. Ainsi nous pensons pouvoir les utiliser pour effectuer un premier dimensionnement.

3.D.3.2 Application de dimensionnement avec le premier modèle.

Comme le premier modèle est totalement analytique, il est tout indiqué pour servir de base à la création d'une application de dimensionnement. On peut d'ores et déjà utiliser les présomptions que l'on peut se faire à la vue de la Figure 3.16 à l'utilisation du premier modèle. Comme il donne des valeurs quasiment doubles de celles du modèle numérique, on pourra utiliser ce modèle pour dimensionner un

moteur en appliquant sur le cahier des charges le rapport 2 reflétant le quotient analytique / numérique constaté. On prend ainsi un coefficient de sécurité par rapport à l'incertitude sur la précision des modèles. Ce coefficient nous est accessible grâce au fait que nous avons aussi développé le modèle numérique. L'écart relatif entre les deux modèles nous donne en effet un élément d'appréciation de la confiance que l'on peut avoir dans nos deux modèles.

3.D.3.3 Prise de décision et construction du prototype.

Une fois qu'un premier dimensionnement a eu lieu avec le modèle analytique, le concepteur peut utiliser le modèle numérique pour conforter son jugement et décider des dimensions finales de construction de son prototype. Comme à cette étape de son processus de conception, le concepteur n'a aucun élément de validation pour chacun de ses modèles, il peut tout à fait se tromper dans les dimensions qu'il définira. Cette erreur éventuelle ne lui serait pas imputable, mais plutôt aux modèles qui peuvent tout à fait être assez proches l'un de l'autre, mais complètement décalés de la réalité.

3.D.3.4 Essais et correction des modèles.

Les données du processus de conception sont totalement différents dès que le premier prototype est construit, même si le dimensionnement initial a été réalisé avec des modèles erronés. En effet, ce changement fondamental est dû au fait que pour la première fois au cours de ce processus, le concepteur va pouvoir réaliser des essais sur quelque chose de concret. Il aura des valeurs de mesure à comparer aux valeurs de ses modèles. Il aura donc des données à sa disposition :

- pour les valider s'ils ont donné des valeurs pas trop éloignées des performances mesurées.

Ou :

- pour les corriger afin qu'ils correspondent mieux aux premières mesures effectuées sur le prototype.

3.E Conclusion.

Dans ce chapitre, nous avons dans un premier temps présenté la structure du moteur à réluctance variable à motifs à flux radial sur laquelle nous avons travaillé . Nous avons distingué la description modulaire du moteur de l'électronique qui lui est nécessaire pour fonctionner.

Nous avons ensuite présenté les deux modèles que nous avons choisi de créer pour des raisons de complémentarité. Nous avons bien insisté sur l'intérêt de la modularité selon laquelle nous avons créé le modèle analytique. Il se compose de réseaux de réluctances linéaires et d'une résolution symbolique de l'équation différentielle de circuit aux prix d'hypothèses simplificatrices assez fortes. Le modèle numérique que nous avons créé s'appuie, quant à lui, sur un modèle éléments finis pour calculer les points d'une surface $\Phi(i, \theta)$ qui exprime le flux dans le noyau en fonction des ampères tours et de la position angulaire du rotor. Cette surface sert ensuite de base à un modèle de résolution de l'équation différentielle de circuit avec la méthode de Runge Kutta. La précision du modèle numérique global dépend de celle avec laquelle on approche cette surface.

Enfin, nous proposons une méthode de dimensionnement qui s'appuie sur les informations fournies par nos deux modèles. Bien que différentes, les deux sources d'information fournies par ces modèles sont souvent convergentes et peuvent s'avérer utiles. Nous avons uniquement, pour l'instant, un outil d'aide à la décision pour le concepteur. Il serait déraisonnable pour l'instant de parler d'outil de dimensionnement.

Cependant, dès que l'on aura des informations sur la validité des modèles, on pourra commencer à travailler sur un processus de dimensionnement avec les modèles modifiés, s'ils deviennent précis.

Conclusion

Dans la première partie de ce mémoire, nous avons présenté la problématique de dimensionnement des moteurs à réluctance variable. Nous souhaitons utiliser le plus possible des modèles analytiques parce qu'ils sont dérivables et ainsi apportent des informations qualitatives au concepteur sur l'interaction entre les paramètres. Ils permettent également de converger plus rapidement et plus précisément vers une solution. Nous préférons utiliser des algorithmes de type gradient car ils prennent bien en compte les dérivées ainsi que les contraintes.

Nous avons également défini les trois grands types de modèles qu'un concepteur en électrotechnique est susceptible de rencontrer et nous avons analysé leurs caractéristiques en vue de leur utilisation pour le dimensionnement. Nous avons aussi présenté différentes approches qui ont été proposées pour dimensionner des moteurs à réluctance variable.

Dans la deuxième partie, nous avons d'abord présenté la méthodologie PASCOSMA, puis ses limites dans la prise en compte des modèles semi analytiques. La solution que nous avons proposée est l'encapsulation des problèmes que posent les parties semi analytiques des modèles. Nous avons réalisé cette encapsulation pour les parties de modèles qui gênaient l'intégration de notre modèle de moteur à réluctance variable de type classique à PASCOSMA.

Pour que ces parties de modèles (fonctions continues affines par morceaux, méthode des moindres carrés, racines de polynômes, ...) ne posent plus de problèmes à l'avenir, nous avons réalisé de des outils génériques et donné un guide méthodologique qui permettra d'identifier plus rapidement les problèmes que poseront de nouvelles parties de modèles semi analytiques.

Enfin, nous avons utilisé nos boîtes de calcul pour dimensionner un moteur à réluctance variable de type classique.

Pour poursuivre ce travail, il serait intéressant de développer d'autres outils génériques de création de boîtes de calcul utiles au concepteur en génie électrique, comme par exemple les intégrales ou la résolution d'équations différentielles. D'autres doivent exister, et nous les identifierons en continuant à travailler sur des modèles de dispositifs électrotechniques.

La troisième partie de ce mémoire est consacrée à la modélisation d'un moteur à réluctance variable à motifs à flux radial. Après avoir présenté la structure du moteur et de l'électronique qui lui est associée, nous avons présenté les deux modèles que nous avons développés. La construction modulaire de ces modèles présente le double intérêt de les rendre capitalisables et d'en faciliter la maintenance. De plus, elle facilite la modélisation rapide de nouveaux produits.

Le modèle analytique modélise le comportement statique du moteur grâce à une modélisation des noyaux à l'aide de réseaux de réluctances et résout symboliquement l'équation différentielle de circuit grâce à une approximation linéaire de l'inductance en fonction de la position.

Le modèle numérique, quant à lui, permet d'obtenir une surface $\Phi(i,\theta)$ en fonction de la géométrie d'un noyau avec la méthode des éléments finis. Ensuite, la méthode de Runge Kutta nous permet de résoudre numériquement l'équation différentielle de circuit.

Enfin, nous nous sommes appuyés sur la complémentarité de ces deux modèles pour proposer une méthode de dimensionnement. Celle-ci prévoit le rapprochement des deux modèles puis le dimensionnement avec le modèle analytique et la construction d'un prototype. Avec les valeurs de mesures obtenues sur celui-ci, le concepteur aura des données concrètes avec lesquelles il pourra corriger ses modèles.

Suite à ce travail, de nombreuses voies sont encore à explorer.

Nous avons proposé une méthode de dimensionnement. Il serait intéressant de l'utiliser afin de dimensionner réellement un moteur à réluctance variable à motifs. Cela nous permettrait de valider ou bien de corriger notre approche de la conception de ce moteur.

Cela nous permettrait aussi de connaître la validité de nos deux modèles. Nous pourrions ainsi décider de l'opportunité de passer à une modélisation en trois dimensions.

Enfin, l'aspect modulaire de ces modèles nous a semblé intéressant. Sans doute ce sujet mériterait-il d'être testé sur des structures électrotechniques plus classiques car la capitalisation des modèles et leur maintenance sont des préoccupations fortes dans le domaine de la conception.

1
2
3
4
5

Bibliographie

- [ALGER-70] P.L. Alger
« Induction Machines »,
Gordon and Breach Science Publisher ed., New York, USA,
1970.
- [ANGOT-72] A.Angot,
« Compléments de Mathématiques à l'usage des ingénieurs
de l'électrotechnique et des télécommunications »,
Editions Masse & Cie, Paris, 1972.
ISBN
- [ATHANI-94] V.V. Athani, V.N. Walivadekar,
« Equivalent Circuit for Switched Reluctance Motor »,
Electric Machines and Power Systems DA.1994, ISSN 0731-
356X, USA, DA.1994, vol.22, n°4, pp533-43.
- [BELOT] « Calcul des Machines Electriques »,
Cours de l'ESE
- [BENOIST-94] D. Benoist, Y. Tourbier, S. Germain-Tourbier
« Plans d'expériences : Construction et Analyse »
Tec et Doc, Lavoisier, 1994
ISBN 2-85206-988-1
- [BESBES-98] M. Besbes, F. Camus, M. Gabsi,
« Local and Global Magnetic Forces in Vibration
Computation »,
4th International Workshop on Electric and Magnetic Fields,
Marseille (France), 12-15 May 1998.
- [BOUCHARD-95] R.P. Bouchard, G. Olivier,
« Circuits et Machines électriques »,
Editions de l'école Polytechnique de Montréal, 1995,
ISBN 2-553-00428-1
- [BOULES-90] N. Boules,
« Design optimisation of permanent magnet DC Motors »,
IEEE Transactions on Ind. Appl., vol 26, no 4, pp786-792,
July/August 1990.
- [BRISSET-95] S. Brisset,
« Outil et Méthodologie pour la Conception de Moteurs à
Réductance Variable à Double Saillance »,
Thèse de Doctorat de l'U.S.T.L., janvier 1995, Lille, France.
- [CHEVALIER-99] T. Chevalier,
« Modélisation et mesure des pertes fer dans les machines
électriques, Application au moteur asynchrone »,
Thèse de Doctorat INPG, Décembre 1999, Grenoble, France.

- [CHIRICOZZI-96] E. Chiricozzi, G. Conti, F. Parasiliti, M. Villani,
« Design Solutions to Optimize Torque Ripple in Synchronous Reluctance Motors »,
Proceedings of IEEE ICEM 1996, vol.2, pp.148-53.
- [COCHRAN-89] P.L. Cochran,
« Polyphase induction motors – analysis, design and applications »,
ed. Marcel Dekker, 675 p, 1989,
ISBN 0-8247-4.
- [COUTEL1-99] C. Coutel,
« Contribution méthodologique à la conception sous contraintes de dispositifs électromagnétiques »,
Thèse de Doctorat INPG, octobre 1999, Grenoble, France.
- [COUTEL2-99] C. Coutel, F. Wurtz, J. Bigeon,
« a comparative study of two methods for constrained optimisation with analytical models dealing with implicit parameters »,
IEEE Transactions on Magnetics, vol. 35, n°3, May 1999
- [COUTEL3-99] C. Coutel, F. Wurtz, J. Bigeon,
« Application of a new method of constrained optimisation with implicit parameters to an electrical contactor »,
IEEE- IEMDC'99, 9-12 Mai 1999, Seattle, Washington USA.
- [CRIVII-97] M. Crivii, M. Jufer,
« 2-Phase SRM Modelling using a Finite Element Method Coupled with the Driver »,
Proceedings of IEEE EPE 1997, pp.3.596-3.601.
- [DESHPANDE-95] U.S. Deshpande, J.J. Cathy, E. Richter,
« High Force Density Linear switched Reluctance Machine »,
Transactions on Industry applications, vol. 31, n°2,
March/April 1995
- [DE VASC-94] J.A. De Vasconcelos,
« Optimisation de forme des structures électromagnétiques »,
Thèse de Doctorat de l'Ecole Doctorale de Lyon des Sciences pour l'Ingénieur, 1994, Lyon, France.
- [ESPANET-99] C. Espanet,
« Modélisation et conception optimale de moteurs sans balais à structure inversée. Application au moteur roue »,
Thèse de Doctorat de l'Université de Franche Comté, 1999.

- [FAIZ-93] J. Faiz, J.W. Finch,
« Aspects of Design Optimisation for Switched Reluctance Motors »,
IEEE Transactions on Energy Conversion, vol. 8, n°4 p704-13, Dec.1993, ISSN 0885-8969.
- [GENTILHOMME-91] A. Gentilhomme,
« Cocase, Un système expert d'aide à la conception d'appareillages électriques »,
Thèse de Doctorat INPG, 1991, Grenoble, France.
- [GERBAUD-00] L. Gerbaud,
« Résolution symbolique d'équations différentielles ordinaires par la décomposition de Dunford en vue de l'étude de circuits d'électronique de puissance »,
NUMELEC '2000, 20-22 mars 2000, Poitiers, France.
- [GLINKA-98] T.J.E. Miller, M. Glinka, M. Mc Gilp, C. Cossar, G. Gallegos-Lopez, D. Ionel, M. Olaru,
« Ultra-Fast Model of the Switched Reluctance Motor »,
Proceedings of IEEE IAS 1998.
- [GOUPY-88] J. Goupy,
« La Méthode des Plans d'Expérience : Optimisation du choix des essais et de l'interprétation des résultats »,
Editions Dunod, 1988.
ISBN 2-04018-732-4
- [GRELLET-97] G. Grellet, G. Clerc,
« Actionneurs Electriques, Principes, Modèles, Commande »,
Editions Eyrolles, Paris, 1997,
ISBN 2-212-09352-7.
- [Harwell] Harwell Subroutine Library, release 1995, routine PA17,
disponible chez R.S.I., ZIRST, Chemin du Pré Carré, 38240 Meylan, France.
- [IVANES-94] M. Ivanes, R. Perret,
« Eléments de Génie Electrique,
Connaissances de Base et Machines Statiques »,
Editions Hermes, Paris, 1994,
ISBN 2-86601-407-3.
- [JUFER-95] M. Jufer,
« Traité d'Electricité, volume IX, Electromécanique »,
Presses polytechniques et universitaires romandes, 1995.
ISBN : 2-88074-285-4

- [KADDED-93] K. Kadded,
« Optimisation de formes de machines électriques à l'aide d'un logiciel éléments finis et de la méthode des pénalités intérieures étendue »,
Thèse de Doctorat INPG, 1993, Grenoble, France.
- [KAUFFMANN-97] J.M. Kauffmann, C. Espanet, F. Wurtz, J. Bigeon,
« Application of a new optimisation approach to the design of electrical wheels »,
Proceedings of IEMDC'97, Milwaukee Wisconsin USA, MB3-3.
- [KRUSE1-98] R. Kruse, G. Pfaff, C. Pfeiffer, H.J. Wehner,
« New Concept of a Low Speed Direct Servodrive »,
Proceedings of Intelligent Motion, May 1998 pp.1-10.
- [KRUSE2-98] R. Kruse, G. Pfaff, C. Pfeiffer,
« Transverse Flux Reluctance Motor for Direct Servodrives Applications »,
Proceedings of IEEE IAS 1998, pp.655-62.
- [LESENNE-80] J. Lesenne, F. Notelet, G. Segulier
« Introduction à l'électrotechnique approfondie »
Editions Techniques et Documentation, Paris, 1980
ISBN 2-85206-089-2
- [LIU-97] C.T. Liu, Y.N. Chen,
« On the Feasible Polygon Classifications of Linear Switched Reluctance Machines »,
Proceedings of IEEE IEMDC 1997, TB1-11.
- [MacSYMA] MacSYMA Reference Manual version 16",
MacSYMA Inc., 20 Academy Street Arlington, MA U.S.A.
- [MATT-95] D. Matt, J.F. Llibre,
« Performances comparées des machines à aimants et à réluctance variable. Maximisation du couple massique ou volumique. »,
Journal de Physique III, ISSN 1155-4320, Fra, DA.95, vol.5, n°10, pp1621-41.
- [MICKAELIDES-95] A. Mickaelides, C. Pollock,
« Design and Performance of a High Efficiency 5-Phase Switched Reluctance Motor »,
Proceedings of IEEE IEMDC 1995, pp.266-70.
- [MILLER-93] T.J.E. Miller,
« Switched Reluctance Motors and Their Control »,
Magna Physics Publishing, Oxford Science Publications,
Clarendon Press, Oxford, 1993, ISBN 0-19-859387-2.

- [MOON-98] J.W. Moon, S.G. Oh, J.W. Ahn, Y.M. Hwang,
« Switched Reluctance Motor with 2-Phase Excitation »,
Proceedings of IEEE IAS 1998, pp.547-52
- [NAGEL] N.J. Nagel, R.D. Lorenz,
« Rotating Vector Methods for Sensorless, Smooth Torque
Control of a Switched Reluctance Motor Drive »,
- [NGUYEN-98] T.N. Nguyen,
« Calculs Elements Finis Paramétrés à l'aide des dérivées
d'ordre élevé – Applications à l'électromagnétisme »,
Thèse de Doctorat INPG, Septembre 1998, Grenoble, France.
- [PERRET-00] C. Perret, J. Fandino, C. Schaeffer, J. Boussey, S. Râel,
« Conception et optimisation d'un microrefroidisseur en cuivre
pour composants électroniques »,
Soumis à la Revue Internationale de Génie Electrique, 23p.
- [PILLAY-98] P. Pillay, W. Cai,
« An Investigation into Vibration in Switched Reluctance
Motors »,
Proceedings of IEEE IAS 1998, pp 341-50.
- [PILLET 88] M. Pillet,
« Pratique des plans d'expérience : Méthodologie Taguchi »,
Les éditions d'organisation, 1997.
ISBN 2-70812031
- [POLOU-86] M. Poloujadoff, R.D. Findlay,
« A Procedure for Illustrating the Effect of Variation of
Parameters on Optimal Transformer Design »,
IEEE-PES Winter Meeting 1986.
- [POWELL-85] M.J.D. Powell,
« On the Quadratic Programming Algorithm of Goldfarb and
Idnani »,
Mathematical Programming Study 25, 1985, pp.46-61.
- [RADUN-94] A.V. Radun,
« Design considerations for the switched reluctance motor »,
Proceedings of IAS'94, vol.1.
- [RAM 73] R. Ramarathnam, B.G. Desai, V. Subba Rao,
« A comparative study of minimisation techniques for
optimisation of induction motor design »,
IEEE Transactions on PAS, v. PAS-22, No 5, pp. 1448-54,
1973.

- [Rapport] F. Bar, P. Rosnet,
« Rapport de Projet d'option MCS : Documentation de l'outil POLY »,
Documentation interne équipe CDI, LEG, février 1999.
- [Rapport2] « Etude, commande et dimensionnement d'un moteur à réluctance variable d'un nouveau type »,
Rapport d'étude Radio-Energie / Supelec, 1998.
- [RICHARDSON-95] K.M. Richardson, C. Pollock, J.O. Flower,
« Design of a Switched Reluctance Sector Motor for an Integrated Motor/Propeller Unit »,
Proceedings of IEEE IEMDC 1995, pp.266-70.
- [ROTTERS-41] H.C. Roters
« Electromagnetic Devices »
Wiley, New York, 1941
- [SAHIN-96] F. Sahin, H.B. Ertan, K. Leblebicioglu,
« Optimum Geometry for Torque Ripple Minimisation of Switched Reluctance Motors »,
Proceedings of IEEE ICEM 1996, vol.2, pp.110-5.
- [SALD-88] R.R. Saldanha,
« Optimisation en électromagnétisme par application adjointe des méthodes de programmation non linéaire et de la méthode des éléments finis »,
Thèse de Doctorat, INPG, 1988, Grenoble, France.
- [SAUVEY1-99] C. Sauvey, F. Wurtz, J.Bigeon, F. Binet,
« A methodology for using morsels continuous linear functions in gradient optimisation of electrical devices »,
IEEE IEMDC '99, 9-12 mai 1999, Seattle, Washington USA.
- [SAUVEY2-99] C. Sauvey, F. Wurtz, J.Bigeon, F. Binet,
« A methodology for using least squares minimisation in gradient optimisation of electrical devices »,
IEEE IEMDC '99, 9-12 mai 1999, Seattle, Washington USA.
- [SAUVEY3-99] C. Sauvey, F. Wurtz, J.Bigeon,
« Analytical gradient type optimisation of a Switched Reluctance Motor »,
IEEE COMPUMAG '99, 25-28 octobre 1999, Sapporo, Japon.
- [SAUVEY-00] C. Sauvey, F. Bar, P. Rosnet, F. Wurtz, J. Fandino, J.Bigeon,
« Intégration du calcul des racines d'un polynôme dans une méthodologie d'optimisation de type gradient »,
NUMELEC '2000, 20-22 mars 2000, Poitiers, France

- [SEGUIER-86] G. Séguier,
« L'électronique de Puissance, Les fonctions de base et leurs principales applications »,
Editions Dunod, 1986,
ISBN : 2-04-019861-X
- [SEGUIER-94] G. Séguier, F. Notelet,
« Electrotechnique Industrielle », 2^{ème} Edition,
Editions Technique et Documentation, Paris, 1994,
ISBN 2-85206-979-2.
- [TANG-96] Y. Tang, J.A. Kline,
« Modelling and Design Optimisation of SRM by Boundary Element Analysis and Simulation »,
IEEE Transactions on Energy Conversion, vol. 11, n°4 p673-80, Dec.1996, ISSN 0885-8969.
- [TORREY-90] D.A. Torrey, J.H. Lang
« Modelling a Non Linear variable-reluctance motor drive »
IEEE Proceedings, vol. 137, Pt B, N°5, sept 1990
- [TORREY-95] D.A. Torrey,
« Analytical modelling of variable reluctance machine magnetisation characteristics »,
IEE Proc-Electr. Power Appl, vol.142, n°1, January 1995.
- [VEINOTT 60] C.G. Veinott,
« Synthesis of induction motor designs on a digital computer »,
AIEE Transactions, Power Apparatus and Systems, vol.79, pp.12-18, 1960.
- [WURTZ-96] F. Wurtz,
« Une nouvelle approche pour la conception sous contraintes de machines électriques »,
Thèse de Doctorat INPG, mai 1996, Grenoble, France.