

## Continuous Global Optimization: A Personal Perspective

**János D. Pintér**

Pintér Consulting Services (PCS) and  
Dalhousie University  
PCS: 129 Glenforest Drive  
Halifax, NS; Canada B3M 1J2  
+1-(902)-443-5910  
pinter@tuns.ca  
<http://www.tuns.ca/~pinter>

### 1. ACTUALITY: GOP'S ARE (ALMOST) UBIQUITOUS...

During the Mathematical Programming Symposium in Lausanne (August 1997), I had a conversation with Arne Drud, a well-respected scientist, software developer and consultant. When asked about the occurrence of global optimization problems (GOP's) in his own practice, he said that for 90% of the nonlinear models the existence of one local solution is not known a priori. He also added that, in many cases, model nonlinearity is relatively mild, and that it is often restricted to a subgroup of decision variables. Further, he said that in certain cases, the problem structure and the modeller's expertise allows a search to start at a reasonable place so that a local solution that is found is "reasonable". His observations — together with the experience of many OR practitioners, engineers, and scientists — clearly show the relevance of the subject. It is also implied that the most traditional repertoire of continuous mathematical programming techniques (linear programming and convex nonlinear programming) can be applied to such problems only with caution, problem-specific expertise, good modelling skills — and, perhaps, with some luck...

It is not difficult to realize why nonlinear models are important. Man-made systems, primarily those related to engineering (manufacturing, distribution, etc.), often can be modelled quite well by using exclusively continuous linear objective and constraint functions. (In the present discussion, we are not including

## Un point de vue personnel sur l'optimisation globale continue

**János D. Pintér**

Pintér Consulting Services (PCS) et  
Dalhousie University  
PCS : 129 Glenforest Drive  
Halifax, NS; Canada B3M 1J2  
+1-(902)-443-5910  
pinter@tuns.ca  
<http://www.tuns.ca/~pinter>

### 1. Les problèmes d'optimisation globale sont omniprésents (ou presque)

Dans le cadre du Symposium de programmation mathématique de Lausanne (août 1997), j'ai eu l'occasion de m'entretenir avec Arne Drud, un scientifique, développeur de logiciels et consultant réputé. Lorsque je l'ai interrogé sur la fréquence des problèmes d'optimisation globale dans ses travaux, il m'a répondu que 90 pour cent des modèles non linéaires analysés comportaient des optima multiples, ce qui en faisait des problèmes d'optimisation globale. Il a aussi ajouté que dans de nombreux cas, la non-linéarité des modèles était assez faible et souvent limitée à un sous-groupe de variables de décision. Il a ensuite poursuivi en disant qu'il arrivait parfois que la structure du problème et la compétence du modélisateur permettent une exploration du problème grâce à laquelle on peut éviter les pièges des optima locaux. Ses observations ainsi que l'expérience de nombreux professionnels en RO, ingénieurs et scientifiques illustrent clairement la pertinence de ce sujet. Bien sûr, lorsqu'on applique des techniques de programmation mathématique traditionnelles (programmation linéaire et programmation non linéaire convexe) à de tels problèmes, il faut faire preuve de prudence, d'une expertise spécifique aux problèmes, de bonnes techniques de modélisation, et, peut-être aussi, d'une certaine dose de chance. . .

Il n'est pas difficile de comprendre pourquoi les modèles non linéaires sont importants. Souvent, on arrive à modéliser assez bien les systèmes créés par l'homme, surtout ceux

associés à l'ingénierie (fabrication, distribution, etc.) en utilisant exclusivement des fonctions

models that include integer variables as they typically lead to problems that are much closer to the spirit of GO.) If one attempts, however, the analysis of natural — i.e., physical, chemical, biological, environmental, or even economic and societal— systems and their governing processes, then nonlinear functions start to play a significant role in the description. As a simple example, one may think of the most prominent (analytical) function forms in physics: probably, polynomials, power functions, the exponential-logarithmic pair and trigonometric functions come to mind first. For more sophisticated examples, one may think of natural phenomena and processes extensively discussed in textbooks on the fractal nature of the Universe. Perhaps more poetically (or simply), one can look at stones on the beach, watch the branches of a tree, wild flowers in a meadow, a range of rocky mountains, turbulent water flow in a stream, or patches of clouds in the sky... All these natural objects certainly show pronounced nonlinearity. Hence, it comes as no surprise that prescriptive (management) models that attempt to describe and optimize the behaviour of inherently nonlinear natural systems may very well lead to multiextremal decision problems.

GO is, arguably, the most challenging part of continuous nonlinear programming. The number of local (pseudo)solutions in a GOP is typically unknown and is often quite large. Furthermore, the quality of the various local and global solutions may differ significantly. GOP's, hence, can be very difficult; most classical (local search based) numerical approaches are — generally speaking — not directly applicable.

Recognizing the apparent need for GO methodology, sporadic work has been devoted to the study of GO models and strategies since the late fifties. In the mid-to late seventies, obvious and not so obvious heuristic methods were suggested. As well, attempts were made to extend convergent univariate algorithms to higher (say, 2,3,...) dimensions. For quite some

économiques linéaires et des contraintes continues. Dans le présent exposé, nous ne traitons pas des modèles qui comportent des variables en nombres entiers puisqu'en général, ils engendrent des problèmes qui se rapprochent beaucoup plus de l'optimisation globale.) Toutefois, si on tente d'analyser les systèmes naturels, c'est-à-dire physiques, chimiques, biologiques, environnementaux ou même économiques et sociétaux, ainsi que les principes qui les gouvernent, alors les fonctions non linéaires commencent à jouer un rôle déterminant dans leur représentation. À titre d'exemple, on peut penser aux formes les plus importantes en physique : on songera probablement en premier lieu aux polynômes, aux fonctions puissance, à la paire exponentielle et logarithmique et aux fonctions trigonométriques. Si l'on veut examiner des exemples plus complexes, on peut considérer les phénomènes et les processus naturels dont on parle abondamment dans les ouvrages sur la dimension fractale de l'univers. Sur un plan peut-être plus poétique (ou plus simple), il suffit de regarder les galets sur une plage, les branches d'un arbre, les fleurs sauvages dans une prairie, une chaîne de montagnes, l'eau agitée d'un ruisseau, ou des bandes de nuages dans le ciel... Tous ces éléments naturels présentent une non-linéarité marquée. Par conséquent, il n'est pas étonnant de constater que les modèles (de gestion) qui essaient de décrire et d'optimiser le comportement de systèmes fondamentalement non linéaires peuvent mener à des problèmes de décision présentant plusieurs extrêmes.

On peut dire que l'optimisation globale est le volet de la programmation non linéaire continue qui représente le plus grand défi. Le nombre de (pseudo)solutions locales dans un problème d'optimisation globale est habituellement inconnu et souvent assez important. En outre, la qualité des diverses solutions locales et globales peut varier considérablement. Ainsi, les problèmes d'optimisation globale peuvent se révéler très difficiles, étant donné qu'il est généralement impossible d'appliquer directement la majorité

des approches numériques classiques (fondées sur une recherche locale).

Depuis la fin des années cinquante, on a reconnu qu'il fallait développer une méthodologie d'optimisation globale et des travaux consacrés à l'étude de modèles et de stratégies d'optimisation

time, there was only modest success. I remember reading the pioneering collections of papers (Dixon and Szegö, 1975, 1978) shortly after they were published. I was fascinated by the subject, and tried to explore the pros and cons of (and connections among) the potpourri of approaches that were presented.

Since then, the development of mathematical programming theory, as well as the rapidly increasing computational power has led to significant advances in GO related modelling, algorithms, and real-world applications. As of 1998, one can find probably some fifty books, a scientific journal (and some closely related others), thousands of research articles, and several informative WWW sites devoted to the subject. Just as important, there are also numerous other studies that refer to the need of GO methodology. At the end of this brief review article, a few pointers leading to more detailed technical information are provided.

## 2. MODEL TYPES

To enable a slightly more formal discussion, we shall consider GOP's in the following general form:

- (1)  $\min f(x)$   
 subject to  $x \in D \subset R^n$
- (2)  $D = \{ x \in R^n ; l \leq x \leq u ;$   
 $g_j(x) \leq 0, j=1, \dots, J \}$

The relations (1)-(2) model a constrained optimization problem defined by the following components:

- $x$  vector of decision variables
- $f$  objective function
- $D$  (non-empty) set of admissible decisions
- $g_j$  constraint functions; their collection will be denoted by  $g$
- $l, u$  explicit bounds defining an embedding box in  $R^n$

globale ont été entrepris ici et là. Dans la seconde moitié des années soixante-dix, des méthodes heuristiques, certaines évidentes, d'autres moins, ont été proposées. On a aussi tenté d'appliquer des algorithmes unidimensionnels convergents à des dimensions plus élevées (2, 3...). Assez longtemps, les résultats obtenus ont été plutôt modestes. Je me rappelle avoir lu les articles tout à fait novateurs de Dixon et Szegö, 1975, 1978, peu de temps après leur publication. J'étais fasciné par le sujet et j'ai exploré les diverses approches présentées ainsi que les liens entre elles.

Depuis, le développement de la théorie de la programmation mathématique et l'augmentation rapide de la puissance de calcul ont permis de réaliser des progrès considérables en ce qui trait à la modélisation et aux algorithmes de l'optimisation globale de même qu'à leur application à des problèmes réels. Aujourd'hui, on peut trouver une cinquantaine d'ouvrages, une revue scientifique (et plusieurs autres assez étroitement liées), des milliers d'articles de recherche et plusieurs sites Web instructifs consacrés à ce sujet. Il est également important de noter que plusieurs autres études soulignent la pertinence d'une méthodologie d'optimisation globale. Le lecteur trouvera à la fin de ce bref survol quelques références le renvoyant à une information technique plus détaillée.

## 2. Types de modèles

Pour permettre une discussion un peu plus rigoureuse, nous considérerons les problèmes d'optimisation globale sous la forme générale de :

- (1)  $\min f(x)$   
 sous la contrainte  $x \in D \subset R^n$
- (2)  $D = \{ x \in R^n ; l \leq x \leq u ;$   
 $g_j(x) \leq 0, j=1, \dots, J \}$

Les relations (1)-(2) définissent un problème d'optimisation sous contraintes dont les éléments sont :

- $x$  vecteur des variables de décision
- $f$  fonction économique
- $D$  ensemble (non vide) de décisions admissibles
- $g_j$  fonctions de contrainte; elles seront dénotées collectivement par  $g$
- $l, u$  bornes explicites définissant un enclassement dans  $R^n$

One can immediately observe that if all functions are continuous, then (by the classical theorem of Weierstrass) the optimal solution set to (1)-(2) is non-empty. As an added technical note, equality constraints can also be incorporated in the above model. For reasons of more straightforward computational tractability, however, we shall assume that  $D$  is a 'body' in  $R^n$  (i.e., it is the closure of its non-empty interior).

To illustrate this paradigm by just one prominent example which often leads to a corresponding GOP, one may think of a model fitting (calibration) problem arising from, say, chemical engineering practice. In this problem setting,  $x$  denotes the model parameters to be selected; their lower and upper bounds  $l$  and  $u$  typically come from physical and practical considerations. Further, the values of certain model functions ( $f$  and/or  $g$ ) may be determined by completing a sequence of side-calculations. Note that the latter may also include stand-alone ('black box') operations such as the execution of a simulation model, or the solution of a set of differential equations, etc. The objective function  $f$  expresses the overall suitability of the model fit (comparing model outcome with the available measurements), while the constraints  $g$  represent added restrictions concerning admissible model parameterizations. For reasons outlined above, it is obvious that

even the explicit functional form of  $f$  and/or  $g$  may not be known. Therefore it is advisable to make only minimal *a priori* assumptions (for example, continuity of  $f$  and  $g$ , if warranted) and rely only on those—unless other specific structural properties can be directly postulated or verified. (Note in this context that several environmental model calibration case studies leading to GOP's—together with other practical examples and an extensive list of references—are discussed by Pintér, 1996a.)

Various classifications of GO models are given in most textbooks devoted to this

On observe immédiatement que si toutes les fonctions sont continues, alors l'ensemble des solutions optimales (selon le théorème classique de Weierstrass) pour (1)-(2) n'est pas vide. D'un point de vue technique, ajoutons qu'on peut aussi incorporer des contraintes d'égalité dans le modèle ci-dessus. Cependant, à des fins de tractabilité computationnelle, nous ferons l'hypothèse que  $D$  est la fermeture de son intérieur non vide dans  $R^n$ .

Pour illustrer ce paradigme à l'aide d'un seul exemple engendrant souvent un problème d'optimisation globale correspondant, on peut penser à un problème d'ajustement (de calibrage) d'un modèle issu, par exemple, du secteur de l'ingénierie chimique. Dans le contexte de ce problème,  $x$  dénote les paramètres du modèle à choisir; leurs bornes  $l$  et  $u$  sont établies en fonction de considérations physiques et pratiques. De plus, on peut déterminer les valeurs de certaines fonctions du modèle ( $f$  ou  $g$ ) en effectuant une série de calculs périphériques. Notons que ces derniers peuvent aussi inclure des opérations indépendantes ("boîte noire") comme l'exécution d'un modèle de simulation ou la résolution d'un ensemble d'équations différentielles, etc. La fonction économique  $f$  exprime la congruence globale de l'ajustement du modèle (en comparant les résultats du modèle avec les mesures disponibles), alors que les contraintes  $g$  représentent des restrictions supplémentaires en ce qui a trait aux paramétrisations admissibles du modèle. Pour toutes ces raisons, il est clair que même la forme fonctionnelle de  $f$  ou de  $g$  peut ne pas être connue explicitement. Par conséquent, il

est plus prudent de poser des hypothèses *a priori* minimales (par exemple, la continuité de  $f$  et de  $g$ , si cela est justifié) et de ne se baser que sur elles — à défaut de pouvoir postuler ou vérifier directement d'autres propriétés structurelles spécifiques. (À cet égard, on peut trouver dans Pintér, 1996a plusieurs études de cas de calibrage de modèles environnementaux aboutissant à des problèmes d'optimisation globale ainsi que d'autres exemples pratiques et une liste détaillée de références.)

Diverses classifications des modèles d'optimisation globale sont fournies dans la majorité des manuels consacrés à ce sujet; vous

subject: see, for instance, Horst and Pardalos (1995). Without going into much detail, an alphabetical listing of some of the most frequently applied models is provided below, in terms of specifying the properties of the model components in (1)-(2). For more on the technical terminology, the Reader is referred to general mathematical programming and GO textbooks; or to the very informative WWW sites of Greenberg (1998) and Neumaier (1998).

- BP - Bilinear / biconvex programming ( $f$  is bilinear or biconvex;  $D$  is convex)
- CO - Combinatorial optimization (note that a typical integer optimization problem can be rewritten, as an equivalent continuous GO model)
- CM - Concave minimization ( $f$  is concave;  $D$  is convex)
- CGO - Continuous global optimization ( $f$  and the components of  $g$  are merely continuous; hence,  $D$  and/or  $f$  may be nonconvex)
- CP - Complementarity problems (linear/nonlinear versions;  $f$  is the scalar product of two vector functions;  $D$  is typically convex)
- DC - Differential convex programming ( $f$  and each component of  $g$  can be represented as the difference of two suitable corresponding convex functions)
  
- FP - Fractional programming ( $f$  is the ratio of two continuous functions;  $D$  is convex)
- LGO - Lipschitz global optimization ( $f$  and the components of  $g$  are Lipschitz-continuous; hence,  $D$  and/or  $f$  may be nonconvex)
- MM - Minimax / maximin problems ( $f$  is the resulting nonconvex objective function;  $D$  is usually convex)
- MO - Multilevel optimization (e.g., in modelling non-cooperative games, involving hierarchies of decision-makers;  $f$  is an aggregate objective function;  $D$  is usually convex)
- MP - Multiplicative programming ( $f$  is the product of several convex functions; the components of  $g$  are convex, or general multiplicative

pouvez vous reporter notamment aux travaux de Horst et Pardalos (1995). Voici une liste alphabétique de quelques-uns des modèles les plus fréquemment utilisés qui précise les propriétés des éléments du modèle dans (1)-(2). Pour plus d'information sur la terminologie technique, le lecteur peut consulter des ouvrages généraux sur la programmation mathématique et sur l'optimisation globale ou les sites Web très instructifs de Greenberg (1998) et Neumaier (1998).

- DC - Programmation différentielle convexe (on peut représenter  $f$  et chaque élément de  $g$  comme la différence de deux fonctions convexes correspondantes appropriées)
- MC - Minimisation concave ( $f$  est concave;  $D$  est convexe)
- MM – Problèmes de minimax / maximin ( $f$  est la fonction économique non convexe obtenue;  $D$  est généralement convexe)
- OC - Optimisation combinatoire (on peut reformuler un problème type d'optimisation en nombres entiers sous la forme d'un modèle d'optimisation globale continue équivalent)
- OGC - Optimisation globale continue ( $f$  et les éléments de  $g$  sont seulement continus; par conséquent,  $D$  ou  $f$  peuvent ne pas être convexes)
- OGD - Problèmes d'optimisation globale distincte ( $f$  est une fonction distincte non convexe appropriée;  $D$  est généralement convexe)
- OGL - Optimisation globale lipschitzienne ( $f$  et les éléments de  $g$  sont des fonctions Lipschitz-continues; par conséquent,  $D$  ou  $f$  peuvent ne pas être convexes)
- OM - Optimisation multiniveau (par exemple, dans la modélisation de jeux non coopératifs qui comportent des hiérarchies de décideurs;  $f$  est une fonction économique agrégée;  $D$  est généralement convexe)
- OQ - Optimisation quadratique ( $f$  et éventuellement aussi les éléments de  $g$  sont des fonctions quadratiques indéfinies)
- PB - Programmation bilinéaire /

<p>functions)</p> <ul style="list-style-type: none"> <li>• NP - Network problems (<math>f</math> is nonconvex; the components of <math>g</math> are mostly linear or convex)</li> <li>• PP - Parametric nonconvex programming (<math>f</math> and/or <math>g</math> depend on a parameter vector)</li> <li>• QP - Quadratic optimization (<math>f</math>—and possibly also the components of <math>g</math>—are indefinite quadratic functions)</li> <li>• RCP - Reverse convex programming (at least one of the function components of <math>g</math> expresses a reverse convex constraint)</li> <li>• SGO - Separable GO problems (<math>f</math> is a suitable nonconvex, separable function; <math>D</math> is typically convex)</li> <li>• SP - nonconvex stochastic programming models in which <math>f</math> and/or <math>g</math> depend on random factors (possibly in a 'black box' fashion as outlined above)</li> </ul> <p>One can observe that the models listed are not mutually exclusive; in fact, some of them are hierarchically related. Specifically, it can be shown that <math>CM \subset DC \subset LGO \subset CGO</math>, and that these four model types jointly cover all of the other, more special, models listed above. (For instance, <math>QP \subset DC</math>.)</p> <p><b>3. SOLUTION APPROACHES</b></p> <p>To solve the GOP (1)-(2), in a strict mathematical sense, means to find the complete set of globally optimal solutions <math>X^*</math>, and the associated global optimum value <math>f^*=f(x^*)</math>. In most cases—at least in the realm of continuous GO—we need to replace this 'ambitious' objective by finding a verified estimate (ideally, both upper and lower bounds) of <math>f^*</math>, and corresponding approximation(s) of points from the set <math>X^*</math>. Again, for reasons of analytical tractability, we mostly assume that <math>X^*</math> is, at most, countable. Note that in many practical contexts the set of (global) optimizers consists only of a single point, or of several points.</p> <p>Looking at the above list of models, one can see that it includes some very well-structured model types (e.g., a concave minimization problem under linear</p>	<p>biconvexe (<math>f</math> est bilinéaire ou biconvexe; <math>D</math> est convexe)</p> <ul style="list-style-type: none"> <li>• PC - Problèmes de complémentarité (versions linéaire et non linéaire; <math>f</math> est le produit scalaire de deux fonctions vectorielles; <math>D</math> est généralement convexe)</li> <li>• PCI - Programmation convexe inverse (au moins un des éléments fonctionnels de <math>g</math> exprime une contrainte convexe inverse)</li> <li>• PF - Programmation fractionnaire (<math>f</math> est le rapport de deux fonctions continues; <math>D</math> est convexe)</li> <li>• PM - Programmation multiplicative (<math>f</math> est le produit de plusieurs fonctions convexes; les éléments de <math>g</math> sont convexes, ou ils sont des fonctions multiplicatives générales)</li> <li>• PP - Programmation non convexe paramétrique (<math>f</math> ou <math>g</math> dépendent d'un vecteur de paramètres)</li> <li>• PR - Problèmes de réseau (<math>f</math> est non convexe; les éléments de <math>g</math> sont majoritairement linéaires ou convexes)</li> <li>• PS - Modèles de programmation stochastique non convexe dans lesquels <math>f</math> ou <math>g</math> dépendent de facteurs aléatoires (peut-être à la manière d'une boîte noire, tel qu'il est indiqué ci-dessus)</li> </ul> <p>On constate que les modèles mentionnés ne s'excluent pas l'un l'autre; en fait, certains sont liés hiérarchiquement. On peut démontrer que <math>MC \subset DC \subset OGL \subset OGC</math> et que ces quatre types de modèles mis ensemble englobent tous les autres modèles, plus spécifiques, présentés ici. (Par exemple, <math>OQ \subset DC</math>.)</p> <p><b>3. Méthodes de résolution</b></p> <p>Au sens strictement mathématique, résoudre le problème d'optimisation globale (1)-(2) signifie trouver l'ensemble complet de solutions optimales globales <math>X^*</math> et la valeur de l'optimum global <math>f^*=f(x^*)</math> qui lui est associée. Dans la plupart des cas, en optimisation globale continue, on doit remplacer cet objectif "ambitieux" en établissant une estimation qualifiée (par des bornes supérieure et inférieure) de <math>f^*</math>, et l'approximation ou les approximations correspondantes des points provenant de l'ensemble <math>X^*</math>. De nouveau, à des fins de tractabilité analytique, on posera</p>
--	---

concave minimization problem under linear or convex constraints) as well as more general models (e.g., differential convex, Lipschitz, or continuous GOP's). Hence, we can reasonably expect that the corresponding solution approaches will also vary to a considerable extent. On one hand, a very general search strategy should work for many cases, although its efficiency might be low for specialized problems. On the other hand, strictly specialized solvers will not work, as a rule, for problem classes outside their scope.

Again, only a non-technical, annotated classification of GO methods is provided here. For more details, consult several of the GO textbooks and WWW sites listed in the references.

#### EXACT METHODS

- 'Naïve' approaches  
Passive (simultaneous) grid search, and passive (pure) random search. Note that these are obviously convergent under mild assumptions, but are truly 'hopeless' in higher (already, say, 3,4,5,...) dimensional problems.
- Enumerative strategies  
These are based upon a complete (streamlined) enumeration of the possible solutions. They are applicable to combinatorial problems, or to certain well-structured continuous GOP's (e.g., concave programming).
- Homotopy (parameter continuation), trajectory methods and related approaches  
These methods have the 'ambitious' objective of visiting (enumerating) all stationary points of the objective function  $f$  (within  $D$ ). This leads to the list of all—global as well as local—optima.  
Applicable to smooth GO problems.
- Relaxation (outer approximation) strategies  
The GOP is replaced by a sequence of relaxed subproblems which are easier to solve. Successive refinement of subproblems to approximate initial problem is applied; cutting planes, more general cuts, diverse minorant constructions, etc. are also possible.

l'hypothèse que  $X^*$  est au plus dénombrable. Il faut noter que dans bon nombre de contextes pratiques, l'ensemble d'optima (globaux) n'est constitué que d'un seul point, ou de plusieurs points.

En examinant la liste des modèles ci-dessus, on constate qu'elle comprend des types de modèles très bien structurés (ex. un problème de minimisation concave sous des contraintes linéaires ou convexes) de même que des modèles plus généraux (ex. programmation différentielle convexe, optimisation lipschitzienne ou problèmes d'optimisation globale continus). Par conséquent, on peut raisonnablement en déduire que les méthodes de résolution correspondantes varieront dans une très large mesure. D'une part, une méthode de recherche très générale devrait fonctionner dans de nombreux cas, bien que son efficacité puisse être faible pour des problèmes spécifiques. D'autre part, certains solutionneurs spécialisés ne fonctionneront pas en principe pour des catégories de problèmes hors de leur champ d'application.

Nous n'avons donné ici qu'une classification annotée non technique des méthodes d'optimisation globale. Pour plus de détails, veuillez consulter les manuels et les sites Web sur l'optimisation globale cités en référence.

#### MÉTHODES EXACTES

- Approches " naïves "
  - Recherche passive sur treillis (simultanée) et recherche passive aléatoire (pure). Ces méthodes sont bien évidemment convergentes pour des hypothèses faibles, mais elles sont tout à fait inutilisables pour des problèmes de dimension plus élevée (déjà, disons, à 3,4,5...).
- Stratégies d'énumération  
Ces méthodes sont fondées sur une énumération complète (exhaustive) des solutions possibles. Elles s'appliquent aux problèmes combinatoires ou à certains problèmes d'optimisation globale continus bien structurés (par exemple, en programmation concave).
- Homotopie, méthodes de trajectoire et approches connexes  
Ces méthodes ont pour objectif " ambitieux " de visiter (dénombrer) tous les points stationnaires de la fonction économique  $f$  (à l'intérieur de  $D$ ). On obtient ainsi la liste de

<p>constructions, etc. are also possible. Applicable to diverse structured GO problems (e.g. concave minimization, DC problems).</p> <ul style="list-style-type: none"> <li>• Branch and bound algorithms Adaptive partition, sampling, and bounding procedures, similar in spirit to known integer linear programming methodology. This approach subsumes many specific cases, and allows for significant generalizations. Applicable to diverse structured GO problems (e.g., concave minimization, reverse convex programs, DC programming, Lipschitz optimization).</li> <li>• Bayesian search algorithms Stochastic function class models; estimation and update of problem-instance characteristics; myopic approximate decisions govern the search procedure. Applicable also to (merely) continuous GO problems.</li> <li>• Adaptive stochastic search methods</li> </ul> <p>Random sampling based search; parameter adjustment, clustering, and deterministic solution refinement options, statistical stopping rules, etc. can be added as enhancements. Applicable to both discrete and</p> <p>continuous GOP's under very general conditions.</p> <p><u>Heuristic Methods</u></p> <ul style="list-style-type: none"> <li>• 'Globalized' extensions of local search methods A preliminary (grid search or random search) global search phase, followed by local search. Applicable to smooth GO problems (differentiability is usually postulated for sake of the local search component).</li> <li>• Genetic algorithms, evolution strategies These heuristically 'mimic' biological evolution models; deterministic and stochastic variants (based on diverse evolution 'game rules') can be constructed. Applicable to discrete and continuous GOP's under mild structural requirements.</li> </ul>	<p>tous les optima, tant globaux que locaux. S'appliquent aux problèmes d'optimisation globale lisses.</p> <ul style="list-style-type: none"> <li>• Stratégies de relaxation (approximation externe) On remplace le problème d'optimisation globale par une séquence de sous-problèmes relaxés qui sont plus faciles à résoudre. On raffine successivement les sous-problèmes pour s'approcher du problème initial; des plans de coupures, des coupes plus générales ou diverses constructions de minorants, etc. sont aussi envisageables. S'appliquent à divers problèmes d'optimisation globale structurés (par exemple, la minimisation concave, les problèmes de programmation différentielle convexe).</li> <li>• Algorithmes "branch and bound" Procédures de partition adaptative, d'échantillonnage et de bornes, similaires aux méthodes de programmation linéaire en</li> </ul> <p>nombre entiers connues. Cette approche subsume beaucoup de cas particuliers et permet des généralisations importantes. S'appliquent à différents problèmes d'optimisation globale structurés (par exemple, à la minimisation concave, aux programmes convexes inverses, à la programmation différentielle convexe, à l'optimisation lipschitzienne).</p> <ul style="list-style-type: none"> <li>• Algorithmes de recherche bayésienne Modèles de classes de fonctions stochastiques; estimation et mise à jour des caractéristiques des instances; des décisions approximatives à courte vue régissent la procédure de recherche. S'appliquent également (ou exclusivement) aux problèmes d'optimisation globale continus.</li> <li>• Méthodes adaptatives de recherche stochastique Recherche fondée sur un échantillonnage aléatoire; on peut y ajouter des options de raffinement comme l'estimation progressive de paramètres, la technique de regroupement et une solution déterministe ainsi que des règles d'arrêt statistiques, etc. S'appliquent aux problèmes d'optimisation globale discrets et continus dans des conditions très générales.</li> </ul> <p><u>MÉTHODES HEURISTIQUES</u></p>
---	--



<p>requirements.</p> <ul style="list-style-type: none"> <li>• Simulated annealing These techniques are based upon the physical analogy of cooling crystal structures that spontaneously attempt to arrive at a stable (globally or locally minimal potential energy) configuration. Applicable to discrete and continuous GOP's under mild structural requirements.</li> <li>• Tabu search The essential idea is to 'forbid' search moves to points already visited in the (usually discrete) search space, at least for the upcoming few steps. Tabu search has traditionally been used to solve combinatorial optimization problems, but it can be extended to the continuous case.</li> <li>• Approximate convex underestimation This strategy attempts to estimate the (large-scale, overall) convexity characteristics of the objective function based on directed sampling in <math>D</math>. Applicable to smooth GO problems.</li> </ul> <p><u>CONTINUATION METHODS</u></p> <p>These first transform the objective function into a more smooth ('simpler') function with fewer local minimizers, and then use a local minimization procedure to trace the minimizers back to the original function. Applicable to smooth GOP's.</p> <ul style="list-style-type: none"> <li>• Sequential improvement of local optima (tunneling, deflation, filled functions etc.) These usually operate on adaptively constructed auxiliary functions, to assist the search for gradually better optima. Applicable to smooth GO problems</li> </ul> <p>Note that certain overlaps exist among the categories listed; algorithm (search</p>	<ul style="list-style-type: none"> <li>• Prolongements " globalisés " des méthodes de recherche locale Phase de recherche globale préliminaire (recherche sur treillis ou aléatoire), suivie d'une recherche locale. S'appliquent aux problèmes d'optimisation globale lisses (on fait l'hypothèse de dérivabilité aux fins de la recherche locale).</li> <li>• Algorithmes génétiques, stratégies évolutives Ces méthodes " imitent " d'un point de vue heuristique les modèles d'évolution biologique; on peut construire des variantes déterministes et stochastiques (basées sur diverses " règles de jeu " de l'évolution). S'appliquent aux problèmes d'optimisation globale discrets et continus dans des conditions structurelles faibles.</li> <li>• Recuit simulé Ces techniques sont fondées sur l'analogie physique du refroidissement des structures cristallines qui tentent spontanément d'atteindre une configuration stable (énergie potentielle minimale globalement ou localement). S'appliquent aux problèmes d'optimisation globale discrets et continus dans des conditions structurelles faibles.</li> <li>• Recherche avec tabous Le principe fondamental de cette méthode consiste à "interdire " à la recherche de se diriger vers des points déjà visités dans l'espace de recherche (généralement discret), du moins dans les quelques étapes qui suivent. Habituellement, la recherche avec tabous est utilisée pour résoudre des problèmes d'optimisation combinatoires, mais on peut l'adapter au cas continu.</li> <li>• Sous-estimation convexe approximative Cette stratégie tente d'estimer les caractéristiques de convexité (globales, sur une grande échelle) de la fonction économique en fonction d'un échantillonnage orienté dans <math>D</math>. S'applique aux problèmes d'optimisation globale lisses.</li> </ul> <p><u>MÉTHODES DE PROGRAMMATION CONTINUE</u></p> <p>Ces méthodes transforment d'abord la fonction économique en une fonction plus lisse (plus simple) comportant moins de minima locaux puis elles utilisent une procédure de minimisation pour retracer les minima dans la fonction originale.</p>
---	--

strategy) combinations are often possible.

#### 4. SOFTWARE DEVELOPMENT

In spite of the very significant theoretical advances in GO, software development and standardized use lags behind. This can be explained by the inherent numerical difficulty of GOP's. Even 'simpler' problem instances—e.g., the indefinite quadratic programming problem— belong to the hardest class of mathematical programming problems. The computational difficulty of any given type of GOP can be expected to be some exponential function of the problem dimensionality  $n$ . (For a formal

discussion, consult e.g., the related chapters in Horst and Pardalos, 1995.)

Consequently, GO problems in  $R^n$ , with  $n$  being 10, 20, 50, or 100, ... may spell rapidly increasing, even straight enormous, complications. This is in spite of the fact that computational power seems to grow at an unbelievable pace—the 'curse of dimensionality' is here to stay...

About one and a half years ago, a survey on continuous GO software was prepared for the newsletter of the Mathematical Programming Society (Pintér, 1996b). Based on the responses of software authors and on information collected from the WWW, over 50 software products were listed in that review. Probably, by now the number of solvers aimed at GOP's should be near to a hundred.

Based upon that survey and gentle follow-up efforts to update the information, my impression is that many of the software products are still dominantly 'academic', as opposed to 'industrial', in character. Of course, there is nothing wrong with either—both are needed. Ideally, the research and implementation/testing/application effort put into their development has synergistic effects. Differences, however, need to be explicitly recognized. Highly desirable—and in the case of targeted 'industrial scope' software, *de facto* indispensable—features of GO software

S'appliquent aux problèmes d'optimisation globale lisses.

- Amélioration séquentielle des optima locaux (" tunneling ", " deflation ", " filled functions ", etc.)

Ces méthodes sont habituellement fondées sur des fonctions auxiliaires construites par adaptation pour aider la recherche d'optima graduellement supérieurs. S'appliquent aux problèmes d'optimisation globale lisses.

Veuillez noter que les catégories mentionnées se chevauchent parfois; des combinaisons d'algorithmes (de stratégies de recherche) sont souvent possibles.

#### 4. Développement de logiciels

En dépit des progrès très considérables réalisés dans le domaine de l'optimisation globale, le développement de logiciels et leur utilisation standardisée traînent encore de l'arrière. Cette situation est attribuable en partie à la difficulté numérique inhérente aux problèmes d'optimisation globale. Même les problèmes " plus simples ", par exemple, le problème de la programmation quadratique indéfinie, sont parmi les problèmes de programmation mathématique les plus ardues. On peut s'attendre à ce que la difficulté de calcul de tout type de problème d'optimisation globale soit une fonction exponentielle de la dimensionnalité du problème  $n$ . (Pour une analyse plus approfondie, le lecteur peut consulter les chapitres sur ce sujet dans l'ouvrage de Horst et Pardalos, 1995.) Par conséquent, les problèmes dans  $R^n$ , avec  $n$  de l'ordre de 10, 20, 50, ou 100... peuvent augmenter rapidement, voire engendrer d'énormes complications. Tout cela en dépit du fait que la puissance de calcul semble progresser à un rythme incroyable

Il y a environ un an et demi, nous avons réalisé un sondage sur les logiciels d'optimisation globale continue pour le bulletin de la Mathematical Programming Society (Pintér, 1996b). À partir des réponses fournies par les développeurs de logiciels et de l'information recueillie dans le Web, nous avons recensé une cinquantaine de logiciels. Aujourd'hui, il y a probablement près d'une centaine de solutionneurs conçus pour les problèmes d'optimisation globale.

À la lumière des résultats de ce sondage et du suivi assuré pour mettre cette information à jour, nous avons l'impression que beaucoup de

<p>include:</p> <ul style="list-style-type: none"> <li>• Well-specified hardware and software environment</li> <li>• Quality user guidance (understandable manual which contains a clearly outlined model development procedure, sensible modelling and troubleshooting tips, user file templates, and non-trivial numerical examples)</li> <li>• ‘Fool-proof’ solver selection and execution procedures</li> <li>• Good runtime communication and documentation including clear system output for all foreseeable situations, including proper error messages, and result file(s)</li> <li>• Visualization options which are especially desirable in nonlinear systems modelling, to avoid problem misrepresentation, and to assist alternative model development procedures</li> <li>• Reliable, high-quality user support</li> <li>• Continued product maintenance and development (since not only science progresses, but hardware devices, operating systems, as well as development platforms are in permanent change)</li> </ul> <p>Looking at this tentative list of requirements, it should be clear (as anyone who has attempted to provide software to be used by others could readily attest) that while the task is certainly not impossible, it is a rather tall order—especially in the present context.</p> <p>Based on publications and conference presentations, it is evident that across Canada there is high quality research related to GO. Information about software development, though, seems to be far more scarce—or just does not seem to reach this humble(?) author, even when information is solicited for the present issue</p>	<p>ces logiciels sont destinés au milieu universitaire, par opposition au secteur industriel. Bien sûr, nous n’avons rien contre ce qui se fait dans ce domaine, mais les deux champs d’activité sont importants. Idéalement, la recherche et les efforts consacrés à la réalisation, à la mise à l’essai et à l’application dans le cadre du processus de développement devraient avoir des effets synergiques. Cependant, il faut en définir les</p> <p>différences de façon explicite. Voici les caractéristiques souhaitables et, dans le cas des logiciels destinés au secteur industriel, <i>de facto</i> indispensables pour les solutionneurs de problèmes d’optimisation globale :</p> <ul style="list-style-type: none"> <li>• Un environnement matériel et logiciel bien défini</li> <li>• Des instructions de qualité pour l’utilisateur (manuel clair qui contient une procédure de développement des modèles bien formulée, conseils utiles pour la modélisation et le dépannage, modèles de fichiers d’utilisateurs et exemples numériques pertinents)</li> <li>• Des procédures fiables de sélection et d’exécution du solutionneur</li> <li>• Une bonne communication et documentation sur l’exécution incluant une sortie système claire pour tous les cas, des messages d’erreur appropriés et un ou plusieurs fichiers résultats</li> <li>• Des options de visualisation (particulièrement souhaitables pour la modélisation des systèmes non linéaires afin d’éviter une représentation inexacte des problèmes et pour faciliter les procédures de développement d’autres modèles)</li> <li>• Un support à l’utilisateur fiable et de grande qualité</li> <li>• Une mise à jour et un développement des produits continus (puisque non seulement la science progresse, mais les périphériques, les systèmes d’exploitation et les plates-formes d’élaboration évoluent aussi constamment)</li> </ul> <p>Si on examine ces exigences, il apparaît clairement (comme pourrait en attester toute personne qui a essayé de développer des logiciels pour d’autres utilisateurs) que si cette tâche n’est pas impossible, elle est tout de même</p>
--	---

of the CORS Bulletin...

A notable exception (known to me) is the set of GO tools developed by Chinneck and Dravnieks (1995, 1997). The Mprobe software provides an assessment of whether or not the feasible region is convex; further, for objective functions, it provides an

assessment of whether a global optimum is possible or whether a local optimum is likely. The GOS-BB package provides formulation assistance for GO and it includes post classification and diagnostic information. Finally, LSGRG(MIS) serves to isolate infeasible subsets of constraints in a given infeasible nonlinear programming problem.

May I mention at this point the LGO model development and solver system for general continuous and Lipschitz GO that attempts to address the points of the 'wish-list' presented above. LGO has been announced in the previous (February 1998) CORS Bulletin issue. For more details, please see (Pintér, 1997a), or the detailed User's Guide (Pintér, 1997b); my WWW site also contains relevant information.

##### **5. GO TEST PROBLEMS AND APPLICATIONS**

Global optimization problems are prevalent in applications described by nonlinear system models. For real-world examples, as well as for both simpler and more challenging test problems, consult, e.g., Bomze, Csendes, Horst and Pardalos (1997), Floudas and Pardalos (1990, 1996), Grossmann (1996), Migdalas, Pardalos and Värbrand (1997), Mistakidis and Stavroulakis (1997), Mockus, Eddy, Mockus, Mockus and Reklaitis (1996), Pintér (1996a). On the WWW, see the pages of Neumaier, and of the Argonne and Sandia National Laboratories; especially Neumaier's site provides numerous further links and pointers. A personal note here which, however, is firmly supported by some of the application studies listed—and by opinions heard from my non-mathematician clients. Quite often

ardue.

Si l'on en juge par les publications et les communications des congrès, on constate qu'à la grandeur du Canada, il se fait une recherche de grande qualité dans le domaine de l'optimisation globale. L'information

concernant le développement des logiciels semble cependant beaucoup plus rare, ou peut-être ne s'est-elle pas rendue jusqu'à votre humble auteur...

Une exception notable à cet égard est la trousse d'outils d'optimisation globale mise au point par Chinneck et Dravnieks (1995, 1997). Le logiciel Mprobe fournit une évaluation qui permet de déterminer si la région réalisable est convexe. Pour les fonctions économiques, il établit une évaluation qui permet de déterminer si un optimum global est possible ou si un optimum local est vraisemblable. La trousse GOS-BB fournit une aide à la formulation pour l'optimisation globale et elle inclut une information après classement et diagnostic. Enfin, le LSGRG(MIS) sert à isoler des sous-ensembles de contraintes non réalisables dans un problème donné de programmation non linéaire non réalisable.

Nous aimerions mentionner à ce stade le système de développement de modèles et de solveurs conçu pour les problèmes généraux d'optimisation continue et d'optimisation lipschitzienne qui a été annoncé dans le dernier bulletin de la SCRO. Pour plus de détails, se reporter à Pintér, 1997a ou au Guide de l'utilisateur détaillé (Pintér, 1997b). (Mon site Web contient aussi toute l'information pertinente à ce sujet.)

##### **5. Problèmes-tests et applications de l'optimisation globale**

Les problèmes d'optimisation globale sont fréquents dans les applications décrites par des modèles de systèmes non linéaires. Pour des exemples de problèmes réels et des problèmes tests, on peut consulter, par exemple, les travaux de Floudas et Pardalos (1990), Grossmann (1996), Pintér (1996a), ou les pages Web de Neumaier et des laboratoires nationaux d'Argonne et de Sandia. (Le site de Neumaier est particulièrement utile puisqu'il fournit de

real-world applications lead to problems which are qualitatively different from (and far more challenging, than many) tests favoured by most academic communities.

GO problems are, and will remain, difficult. However, algorithm development, and more powerful machines (including parallelization options) push the solvability frontier farther. Let me illustrate this point with my own experience. Six to seven years ago I was glad to be able to solve somewhat 'tricky' GO problems in up to 5-6 variables (with my solver of the day, on those machines). In recent years my LGO solver has been routinely applied to optimization problems which have several tens to a few hundred variables. (Due to the exponential complexity of GOP's, this is non-trivial progress; plus, hopefully, the GO solvers themselves are getting better...) The present standard educational version being shipped handles 20 variables and 20 constraints. The largest, truly difficult, decision models solved so far by LGO have a few hundred variables. Without the slightest doubt, however, there is still much room to progress...

We conclude with a rather long (but very far from exhaustive) list of existing and potential areas of GO applications. The first part of this listing draws upon information gathered from the WWW and from literature. For more details, consult e.g., the Argonne National Laboratory MINPACK-2 test problem collection (from 1993, with added notes by Averick and Moré), and Neumaier's WWW pages. This partial list includes the following examples (the GO applications are arranged alphabetically, without further grouping):

- Chebychev quadrature
- chemical and phase equilibria
- coating thickness standardization
- combustion of propane (full and reduced formulation)
- control system analysis and design
- database optimization
- elastic-plastic torsion

nombreux liens et références.) Nous nous permettons ici de faire un commentaire personnel qui est cependant grandement partagé par bon nombre de clients qui ne sont pas mathématiciens. Très souvent, les

applications réelles engendrent des problèmes qui sont différents qualitativement et plus exigeants que beaucoup des tests privilégiés par le milieu universitaire.

Les problèmes d'optimisation sont difficiles et ils le resteront. Cependant, le développement d'algorithmes et de machines plus puissantes, notamment les options de parallélisation, repoussent toujours plus loin les limites de la résolution. Laissez-moi vous démontrer ce point à partir de ma propre expérience. Il y a six ou sept ans, j'étais heureux d'arriver à résoudre certains problèmes d'optimisation globale assez complexes présentant jusqu'à 5 ou 6 variables. Dans les dernières années, on a utilisé régulièrement mon solveur d'optimisation globale lipschitzienne pour résoudre des problèmes d'optimisation présentant des variables de plusieurs dizaines (20 à 100). La version éducative standard actuellement distribuée peut gérer 20 variables et 20 contraintes. Les modèles de décision les plus grands et les plus difficiles résolus jusqu'ici à l'aide de l'optimisation globale lipschitzienne présentent quelques centaines de variables. Ce qui démontre bien qu'on peut toujours aller plus loin ...

Nous allons conclure cet article par une liste assez longue, bien que non exhaustive, des champs d'application actuels et potentiels de l'optimisation globale. La première partie de cette liste est basée sur l'information recueillie dans le Web et sur la documentation disponible. Pour plus de détails, on peut consulter par exemple les problèmes-tests MINPACK-2 du laboratoire national d'Argonne (de 1993, avec des ajouts de Averick et Moré) et les pages Web de Neumaier. Les exemples d'applications sont classés par ordre alphabétique, sans autre critère de catégorisation :

- ajustement exponentiel de données
- ajustement gaussien de données
- allumage par combustible solide
- analyse de réactions enzymatiques

<ul style="list-style-type: none"> <li>• enzyme reaction analysis</li> <li>• exponential data fitting</li> <li>• flow in a channel</li> <li>• flow in a driven cavity</li> <li>• Gaussian data fitting</li> <li>• Ginzburg-Landau problem</li> <li>• human heart dipole</li> <li>• hydrodynamics modelling</li> <li>• incompressible elastic rods</li> <li>• isomerization of alpha-pinene (in several versions)</li> <li>• localization of brain activity</li> <li>• minimal surfaces</li> <li>• molecular configuration problems</li> <li>• optimal design with composites</li> <li>• pressure distribution in a journal bearing</li> <li>• Ramsey graphs</li> <li>• solid fuel ignition</li> <li>• spherical codes</li> <li>• steady-state combustion</li> <li>• swirling flow between disks</li> <li>• thermistor resistance analysis</li> <li>• thin film design</li> <li>• VLSI CAD</li> </ul> <p>The second list of GO applications draws mainly upon my own practice. Most of these problem-types are discussed in Pintér (1996a), but some of the work is more recent, or still in progress. (See also the extensive list of further applications collected in the book.)</p> <ul style="list-style-type: none"> <li>• 'black box' system design and operation</li> <li>• combination of deterministic or statistical expert opinions (forecasts, votes)</li> <li>• data classification (cluster analysis)</li> <li>• data visualization</li> <li>• extremal energy problems (in mathematical, physical, chemical and biological modelling)</li> <li>• inverse model fitting (calibration) to observation data sets</li> <li>• nonlinear approximation (including</li> </ul>	<ul style="list-style-type: none"> <li>• analyse des résistances d'une thermistance</li> <li>• analyse et conception des systèmes de contrôle</li> <li>• codes sphériques</li> <li>• combustion de propane (formulations complète et réduite)</li> <li>• combustion en régime stationnaire</li> <li>• conception de circuits intégrés</li> <li>• conception de pellicules fines</li> <li>• conception optimale avec matériaux composites</li> <li>• dipôle du cœur humain</li> <li>• distribution de pression dans un palier</li> <li>• équilibre chimique et de phase</li> <li>• flux dans un conduit</li> <li>• flux dans une cavité attaquée</li> <li>• flux tourbillonnant entre des disques</li> <li>• graphes de Ramsey</li> <li>• isomérisation d'alpha-pinènes (en plusieurs versions)</li> <li>• localisation de l'activité cérébrale</li> <li>• modélisation hydrodynamique</li> <li>• optimisation de bases de données</li> <li>• poutres élastiques incompressibles</li> <li>• problème de Ginzburg-Landau</li> <li>• problèmes de configuration moléculaires</li> <li>• quadrature de Tchebychev</li> <li>• standardisation de l'épaisseur des revêtements</li> <li>• surfaces minimales</li> <li>• torsion élasto-plastique</li> </ul> <p>La seconde liste d'applications de l'optimisation globale est établie principalement à partir de mon expérience. Vous retrouverez bon nombre de ces types de problèmes dans Pintér (1996a), mais certaines applications sont plus récentes ou encore en cours. (Se reporter également à la liste détaillée incluse dans le volume.)</p> <ul style="list-style-type: none"> <li>• affectation des ressources (découpage, ordonnancement, chargement, mise en paquets) et problèmes connexes</li> <li>• ajustement de modèle inverse (calibrage) pour des ensembles de données d'observation</li> <li>• analyse et gestion des risques</li> <li>• approximation non linéaire (y compris l'ajustement de courbes ou de surfaces)</li> </ul>
---	--

- curve or surface fitting)
- optimized design and tuning of equipment and instruments
- resource allocation (cutting, sequencing, loading, packing) and related problems
- risk analysis and management
- robust product/mixture design
- systems of nonlinear equations and inequalities

We are especially interested to hear about GO (test or real) challenges, as well as about other prospective application areas.

#### ACKNOWLEDGEMENTS

First, I would like to thank Rick Caron for the kind invitation to prepare this review, as well as for his comments.

I am indebted to the owners of the WWW sites referred to above (and also to many other site owners), for the useful information altruistically collected and provided on their pages.

This work was completed during my visit at Wolfram Research, Inc.—thanks for all support, a very friendly work atmosphere, and good coffee at Espresso Point...

Finally, I wish to thank a (surprisingly large) number of colleagues from around the world—and also a few of my clients—for their advice, criticism, feedback, and support. They will be properly listed in a forthcoming book (or two) on the subject.

#### REFERENCES

Only a few references—including a classical textbook, and some others from the 1990's, as well as several WWW sites—are listed. Note that, for instance, the rapidly growing 'Nonconvex

*Optimization And Its Applications*' book series by Kluwer has also several other recent volumes closely related to the subject. (See <http://kapis.www.wkap.nl/series.htm/NOIA>

- classement de données (analyse de regroupement)
- combinaison d'opinions d'experts déterministes ou statistiques (sondages, votes)
- conception et exploitation du système de la boîte noire (oracle)
- conception optimisée et réglage d'équipement et d'instruments
- conception robuste de produits ou mélanges
- problèmes d'énergie extrême (modélisations mathématique, physique, chimique et biologique)
- systèmes d'équations et d'inéquations non linéaires
- visualisation de données

Nous sommes vivement intéressés à entendre parler de problèmes-tests ou de problèmes réels qui présentent des défis en matière d'optimisation globale, ainsi que d'éventuels nouveaux champs d'application.

#### Remerciements

J'aimerais d'abord remercier Rick Caron de sa gentille invitation à rédiger cet article de synthèse ainsi que de ses précieux commentaires. Je suis également reconnaissant aux propriétaires des sites Web cités plus haut pour l'information qu'ils rassemblent et présentent dans leurs pages. Enfin, j'aimerais remercier un nombre important de collègues du monde entier et aussi quelques-uns de mes clients pour leurs conseils, leurs critiques, leurs commentaires et leur appui.

#### RÉFÉRENCES

Quelques références clés seulement, incluant un manuel classique et d'autres ouvrages des années 90, ainsi que plusieurs sites Web. Veuillez noter, à titre d'exemple, que la série de livres "*Nonconvex Optimization And Its Applications*" de Kluwer, qui connaît une expansion très rapide, comprend plusieurs autres volumes récents étroitement liés à ce sujet. (Voir

<http://kapis.www.wkap.nl/series.htm/NOIA> pour obtenir une liste à jour. )

for a current listing.)

- Argonne National Laboratory (1993) *MINPACK-2 Test Problem Collection*. (See also the Accompanying notes titled 'Large-scale optimization: Model problems', by B.M. Averick and J.J. Moré; <http://www-c.mcs.anl.gov/home/more/tprobs/tprobs.html>.)
- Bomze, I.M., Csendes, T., Horst, R., and Pardalos, P.M., eds. (1997) *Developments in Global Optimization*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Chinneck, J.W. (1995) Analyzing infeasible nonlinear programs. *Computational Optimization and Applications* 4, 167-179.
- Dixon, L.C.W. and Szegö, G.P., eds. (1975, 1978) *Towards Global Optimisation*. Vols. 1-2. North-Holland, Amsterdam.
- Dravnieks, E.W. and Chinneck, J.W. (1997) Formulation assistance for global optimization problems. *Computers and Operations Research* 24, 1151-1168.
- Drud, A. (1997) Personal communication.
- Floudas, C.A. and Pardalos, P.M. (1990) *A Collection of Test Problems for Constrained Global Optimization Algorithms*. Lecture Notes in Computer Science 455, Springer, Berlin / Heidelberg / New York.
- Floudas, C.A. and Pardalos, P.M., eds. (1992) *Recent Advances in Global Optimization*. Princeton University Press, Princeton.
- Floudas, C.A. and Pardalos, P.M., eds. (1996) *State of the Art in Global Optimization: Computational Methods and Applications*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Grossmann, I.E., ed. (1996) *Global Optimization in Engineering Design*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Glover, F and Laguna, M. (1997) *Tabu Search*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Greenberg, H.J. (1998) *Mathematical Programming Glossary* (<http://www-math.cudenver.edu/~hgreenbe/glossary/glossary.html>).
- Hansen, E.R. (1992) *Global Optimization Using Interval Analysis*. Marcel Dekker, New York.
- Horst, R. and Tuy, H. (1996) *Global Optimization – Deterministic Approaches*. Springer, Berlin / Heidelberg / New York. (3rd Edn.)
- Horst, R. and Pardalos, P.M., eds. (1995) *Handbook of Global Optimization*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Journal of Global Optimization* (since 1991, Kluwer Academic Publishers).
- Kearfott, R.B. (1996) *Rigorous Global Search: Continuous Problems*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Michalewicz, Z. (1996) *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, Berlin / Heidelberg / New York. (3rd Edn.)
- Migdalas, A., Pardalos, P.M. and Värbrand, P., eds. (1997) *Multilevel Optimization: Algorithms and Applications*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Mistakidis, E.S., Stavroulakis, G.E. (1997) *Algorithms, Heuristics and Engineering Applications by the F.E.M.* Kluwer Academic Publishers, Dordrecht / Boston / London.
- Mockus, J., Eddy, W., Mockus, A., Mockus, L. and Reklaitis, G. (1996) *Bayesian Heuristic Approach to Discrete and Global Optimization (Algorithms, Visualization, Software, and Applications)*. Kluwer Academic Publishers, Dordrecht / Boston / London.
- Neumaier, A. (1990) *Interval Methods for Systems of Equations*. Cambridge University Press, Cambridge.
- Neumaier, A. (1998) *Global Optimization Web Pages* (<http://solon.cma.univie.ac.at/~neum/glopt.html>).
- Pintér, J.D. (1996a) *Global Optimization in Action (Continuous and Lipschitz Optimization:*



*Algorithms, Implementations and Applications*). Kluwer Academic Publishers, Dordrecht / Boston / London.

Pintér, J.D. (1996b) Continuous global optimization software: A brief review. *Optima* 52, 1-8.

Pintér, J.D. (1997a) LGO – A program system for continuous and Lipschitz global optimization; pp. 183-197. in: Bomze, Csendes, Horst, and Pardalos, eds. (1997).

Pintér, J.D. (1997b) *LGO – A Model Development System for Continuous Global Optimization. User's Guide*. Pintér Consulting Services, Halifax, NS.

Sandia National Laboratories (1998) *Global Optimization WWW Survey*. (Maintained by W. E. Hart; <http://www.cs.sandia.gov/opt/survey/main.html>).

Zhigljavsky, A.A. (1991) *Theory of Global Random Search*. Kluwer Academic Publishers, Dordrecht / Boston / London.

Congrès conjoint INFORMS / SCRO-CORS du printemps 1998

(40e Congrès annuel de la SCRO)

Montréal, Québec, du 26 au 29 avril 1998

« [www.crt.umontreal.ca/mtl98/](http://www.crt.umontreal.ca/mtl98/) »