Features of Industrial Optimization Problems

Luís Nunes Vicente, Univ. Coimbra

http//www.mat.uc.pt/~lnv

Workshop Investigação Operacional na Robótica

March 30, 2011 — APDIO, Coimbra

Presentation Outline

1. Industrial Contacts
2. Functions, Constraints, Variables
3. Structure, Uncertainty, Dimension
4. Surrogates, Global Optimization, Multicriteria Optimization, Multicomponent Optimization

Presentation Outline

1. Industrial Contacts
2. Functions, Constraints, Variables
3. Structure, Uncertainty, Dimension
4. Surrogates, Global Optimization, Multicriteria Optimization, Multicomponent Optimization

Software, Hardware, and Consulting Companies

- IBM Research (USA)

  $\longrightarrow$ 3 month postdoc, 1.5-year sabbatical time

  $\longrightarrow$ co-authorship of papers and book

- CERFACS (France)

  $\longrightarrow$ two 3-month visits

- Critical Software, Witt (Coimbra, Portugal)

  $\longrightarrow$ several contacts, supervision of graduate students

Aerospace Companies

- Astos Solutions, OHB (Germany)

  Skysoft/GMV, Spin.Works (Lisbon, Portugal)

  $\longrightarrow$ European Space Agency (ESA) projects and contacts

- Boeing (USA), SamTech (Belgium)

  $\longrightarrow$ several contacts, FCT-Portugal projects

Finance Companies

- Goldman Sachs (USA)

  $\longrightarrow$ several contacts, co-authorship of papers

- Bloomberg (USA), Hypo Real State, Allianz (Germany)

  $\longrightarrow$ several contacts

- Montepio Geral, BCP (Lisbon, Portugal)

  $\longrightarrow$ several contacts, FCT-Portugal projects

The University of Coimbra organized two Workshops on Optimization in Finance in 2005 and 2007 and currently offers a MS Program in Quantitative Finance.

The First Industrial Workshop in Mathematics...

Industrial Workshop, Coimbra, October 14-15, 2005

Organized by the Laboratory for Computational Mathematics (LMC) of the Centre for Mathematics of the University of Coimbra (CMUC).

`http://www.mat.uc.pt/~cmuc/lcm/workshop.html`

John Betts (The Boeing Company)
Pedro Champalimaud (ES Contact Center)
Andrew R. Conn (IBM Research)
Natércia Fernandes (ENGINUM)
Sam Howison (OCIAM, Mathematical Inst., Univ. of Oxford)
Pedro Júdice (JP Morgan Chase)
Jorge Barros Luís (Montepio Geral)
Andrey Romanenko (ENGINUM)
Marc Steinbach (Zuse Institute Berlin)

Presentation Outline

1. Industrial Contacts
2. Functions, Constraints, Variables
3. Structure, Uncertainty, Dimension
4. Surrogates, Global Optimization, Multicriteria Optimization, Multicomponent Optimization

Functions (objective and constraints)

A continuous (deterministic) optimization problem can be represented as

$$\begin{array}{rl} \text{minimize} & f(x) \\ \text{subject to} & x \in \Omega, \end{array}$$

where Ω is, most of the times, algebraically defined by a set of equalities $c_i(x) = 0$, $i \in \mathcal{E}$, and a set of inequalities $c_i(x) \geq 0$, $i \in \mathcal{I}$.

The index sets $\mathcal{E}$ and $\mathcal{I}$ may be infinite (semi-infinite optimization).

Variables may be further restricted to a cone...

Conic Optimization

The linear conic programming problem looks like:

$$\begin{array}{lrcl} \min & \langle c, x \rangle & & \\ \text{s.t.} & \langle a_i, x \rangle & = & b_i, \quad i = 1, \ldots, m, \\ & x & \in & C, \end{array}$$

where C is a cone.

Conic Optimization

The linear conic programming problem looks like:

$$\begin{array}{lrcl} \min & \langle c, x \rangle & & \\ \text{s.t.} & \langle a_i, x \rangle & = & b_i, \quad i = 1, \ldots, m, \\ & x & \in & C, \end{array}$$

where C is a cone.

An example is linear semi-definite programming (SDP):

$$\begin{array}{lrcl} \min & \langle C, X \rangle & & \\ \text{s.t.} & \langle A_i, X \rangle & = & b_i, \quad i = 1, \ldots, m, \\ & X = X^\top & \succcurlyeq & 0, \end{array}$$

where $\langle Y, W \rangle = \text{tr}(Y^\top W)$.

Conic Optimization

The linear conic programming problem looks like:

$$\begin{array}{lrcl} \min & \langle c, x \rangle & & \\ \text{s.t.} & \langle a_i, x \rangle & = & b_i, \quad i = 1, \ldots, m, \\ & x & \in & C, \end{array}$$

where C is a cone.

An example is linear semi-definite programming (SDP):

$$\begin{array}{lrcl} \min & \langle C, X \rangle & & \\ \text{s.t.} & \langle A_i, X \rangle & = & b_i, \quad i = 1, \ldots, m, \\ & X = X^\top & \succcurlyeq & 0, \end{array}$$

where $\langle Y, W \rangle = \mathrm{tr}(Y^\top W)$.

Another example is second-order cone programming, where

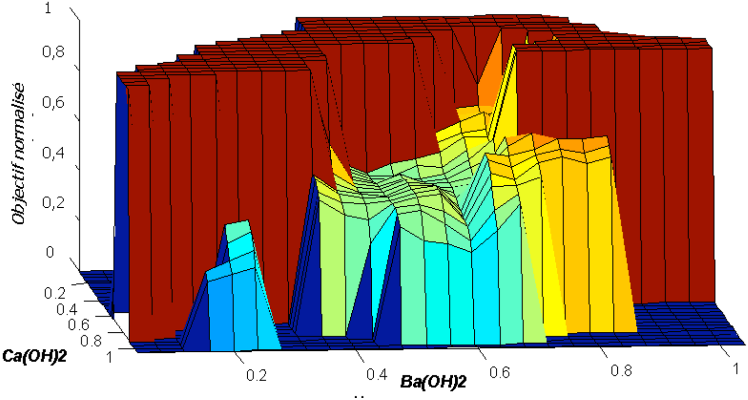
$$C \; = \; \{(y,t) \in \mathbb{R}^n \times \mathbb{R} : \; \|y\|_2 \leq t\}.$$

Functions (objective and constraints)

In practice, the functions:

- are typically continuous (better saying Lipschitz continuous)
- can be nonsmooth
- can be nonconvex
- can be noisy
- can provide only a few digits of accuracy
- can be costly to evaluate
- can be of black-box type (unavailability of derivatives)
- can be undefined in points of the putative domain

The objective function might not be smooth:



Relaxable and Unrelaxable Constraints

Constraints may appear in different ways.

- Unrelaxable constraints are those which must be satisfied in order to evaluate the objective function or other problem defining functions.

  Such constraints have to be satisfied at all iterations in an algorithmic framework.

- In contrast, relaxable constraints need only to be satisfied approximately or asymptotically.

Relaxable and unrelaxable constraints are also known as soft and hard constraints, or as open and closed constraints, respectively.

DFO Constraints

Constraints may also be:

- of black-box type — derivative-free optimization (DFO) constraints,
- or not — derivative-based constraints.

There are situations where DFO constraints need to be treated as unrelaxable.

These situations require the availability of a feasible starting point, are quite difficult to address in general, and are, fortunately, rare in practice.

Hidden Constraints

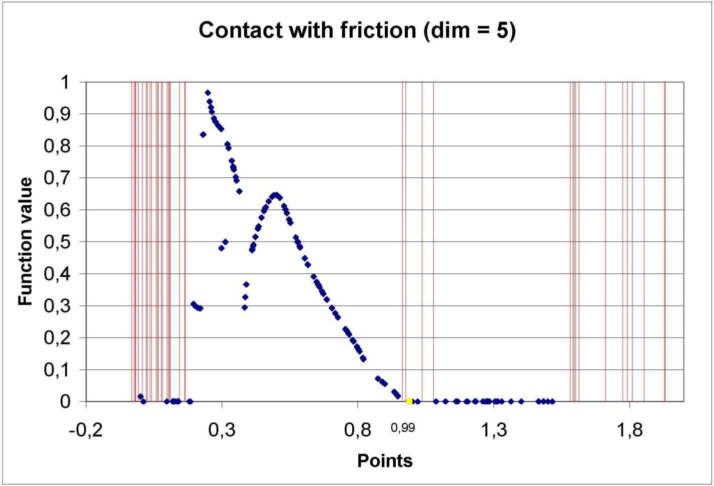
An extreme case of DFO unrelaxable constraints that does occur in practice are the so-called hidden constraints.

Hidden constraints are not part of the problem specification/formulation and their manifestation comes in the form of some indication that the objective function could not be evaluated.

For example, the objective function $f(x)$ may be computed by a simulation package which may not converge for certain (unknown a priori) values of input parameters, failing to produce the objective function value.

So far these constraints are treated in practical implementations by a heuristic approach or by using the extreme barrier function approach.

The objective function might not be well defined (hidden constraints):



Variables (integer or categorical)

A significant number of practical problems is formulated using integer or binary variables:

$\longrightarrow$ an important tool in Operations Research and Management Sciences.

A particularly difficult type of integer variables are categorical variables:

$\longrightarrow$ categorical variables do not admit branching or rational approximations and appear due to, e.g., manufacturing requirements.

Variables (implicitly densely discrete)

Many optimization problems are only apparently continuous.

Many applications involve an underlying discrete structure (most of the times unknown).

The evaluation of the objective function is made by first rounding or projecting variables (to where the evaluation is possible or desirable).

The underlying discrete structure is unknown to the optimizer and its manifestation is detected only when a evaluation is demanded.

Sparse Optimization — a new paradigm

It is well known now that linear ℓ_0-recovery

$$\begin{array}{lll} \min & \|x\|_0 & = |\{i : \ x_i \neq 0\}| \\ \text{s.t.} & Ax \ = \ b, & \end{array}$$

which is NP-hard, ...

It is well known now that linear ℓ_0-recovery

$$\begin{array}{lll} \min & \|x\|_0 & = |\{i : \; x_i \neq 0\}| \\ \text{s.t.} & Ax \; = \; b, & \end{array}$$

which is NP-hard, ...

... can be achieved (under certain conditions) by linear ℓ_1-recovery

$$\begin{array}{ll} \min & \|x\|_1 \\ \text{s.t.} & Ax \; = \; b, \end{array}$$

which can be done in polynomial time.

Sparse Optimization — a new paradigm

It is well known now that linear ℓ_0-recovery

$$\begin{array}{lll} \min & \|x\|_0 & = |\{i : \ x_i \neq 0\}| \\ \text{s.t.} & Ax \ = \ b, & \end{array}$$

which is NP-hard, ...

... can be achieved (under certain conditions) by linear ℓ_1-recovery

$$\begin{array}{ll} \min & \|x\|_1 \\ \text{s.t.} & Ax \ = \ b, \end{array}$$

which can be done in polynomial time.

Many application problems welcome nullity of variables.

Presentation Outline

1. Industrial Contacts
2. Functions, Constraints, Variables
3. Structure, Uncertainty, Dimension
4. Surrogates, Global Optimization, Multicriteria Optimization, Multicomponent Optimization

Structure

Often, there is a structure in practical optimization problems:

- Networks (transportation and telecommunication problems).
- Differential equations (engineering problems) — parameter estimation, optimal control & optimal design, shape optimization, topology optimization.
- Scenario tree (stochastic programming).
- Sparsity, partial separability, ...

Uncertainty — Stochastic Optimization

Optimization problems where decisions must be taken over time or under uncertainty are often modelled by Stochastic Programming formulations.

Some of the most well known applications can be found in asset/liability and risk management in Finance and in protection strategies, production planning, and pollution control in Energy.

Stochastic variables can be anticipative (independent of future realizations of random events), possibly deterministic, or adaptive (otherwise), typically random $\longrightarrow$ recourse models.

A two-stage SP problem:

$$\begin{aligned} \max_{x,\omega\in\Omega} \quad & a^\top x + E\left(c(\omega)^\top y(\omega)\right) \\ \text{s.t.} \quad & Ax = b \\ & B(\omega)x + C(w)y(w) = d(w), \ \forall w \in \Omega \\ & x \geq 0, \quad y(\omega) \geq 0, \ \forall \omega \in \Omega. \end{aligned}$$

A two-stage SP problem:

$$\begin{aligned} \max_{x,\omega\in\Omega} \quad & a^\top x + E\left(c(\omega)^\top y(\omega)\right) \\ \text{s.t.} \quad & Ax = b \\ & B(\omega)x + C(w)y(w) = d(w), \ \forall w \in \Omega \\ & x \geq 0, \quad y(\omega) \geq 0, \ \forall \omega \in \Omega. \end{aligned}$$

Setting $\Omega = \{\omega_1, \ldots, \omega_S\}$, the deterministic version becomes

$$\begin{array}{rllll} \max_{x,y_1,\ldots,y_S} & a^\top x + p_1 c_1^\top y_1 + \cdots + p_S c_S^\top y_S & & & \\ \text{s.t.} & Ax & & & = b \\ & B_1 x & + C_1 y_1 & & = d_1 \\ & & & & \vdots \\ & B_S x & & + C_S y_S & = d_S \\ & x, y_1, \ldots, y_S \geq 0. & & & \end{array}$$

Constraints can involve these variables and random data and/or be imposed with some probability (chance constraints).

The solution of a stochastic programming problem is usually done by solving a corresponding deterministic one, which, to be realistic, must take into consideration a considerably large number of scenarios.

Uncertainty — Robust Optimization

Uncertainty can also appear in the problem data due to, e.g., estimation errors or incompletion.

In Robust Optimization, the immunization against data uncertainty is made by letting the uncertain parameters becoming variables in uncertainty sets P in which one looks for a safe, worst case scenario:

$$\min_{(x,p)\in\mathbb{R}^n\times P} \quad \max_{p\in P} f(x,p).$$

Conic Optimization (in particular, semi-definite and second-order programming) are main tools in Robust Optimization.

Dimension

A number of continuous problems can easily become extremely large:

- Linear programming problems associated with mixed integer problems (e.g., airline scheduling).
- Discretized stochastic programming problems (e.g., asset/liability management (ALM) problems).
- Nonlinear programming problems arising from PDE-constrained optimization (e.g., dynamic optimization problems in chemical engineering).

One can now solve instances of continuous problems with derivatives with a few hundred thousand variables and constraints in a reasonable amount of serial computing time.

Dimension

Dimension must be understood differently for:

- Integer programming and combinatorial optimization problems.
- Derivative-free optimization (DFO) problems.
- Global optimization problems.

For these classes of problems, extremely large might mean a couple of hundred variables.

Presentation Outline

1. Industrial Contacts
2. Functions, Constraints, Variables
3. Structure, Uncertainty, Dimension
4. Surrogates, Global Optimization, Multicriteria Optimization, Multicomponent Optimization

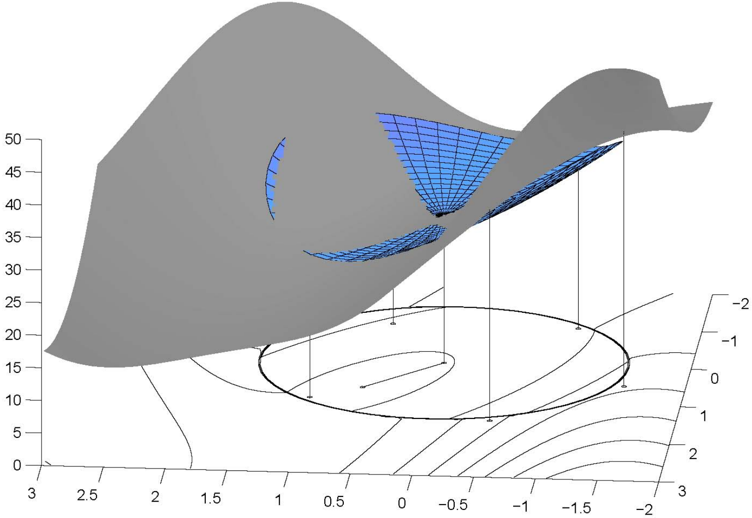
Surrogate Models

In engineering modeling it is frequently the case that the function to be optimized is expensive to evaluate.

The problem to be (approximately) solved may require extensive simulation of systems of differential equations, possibly associated with different disciplines, or may involve other time consuming numerical computations.

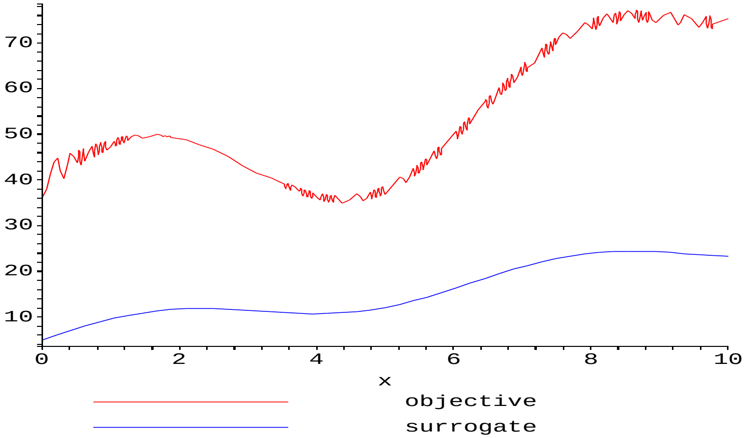
Engineers frequently consider surrogate models for the true function (which might be already a model itself).

Example (good local surrogate)



Example

A surrogate is not necessarily a good approximation of the true function:



Physical Surrogate Models

By physical surrogate models we mean surrogate models built from a physical or numerical simplification of the true problem functions.

One can think, for instance, of a coarser mesh discretization in numerical PDEs or of a linearization of term sources or equations, as ways of obtaining physical surrogate models.

Physical surrogate models are in many circumstances based on some knowledge of the physical system or phenomena being modeled, and thus any particular such model is difficult to exploit across different problems.

Functional Surrogate Models

Functional surrogate models are algebraic representations of the true problem functions.

One can say that functional models are typically based on the following components: a class of basis functions, a procedure for sampling the true functions, a regression or fitting criterion, and some deterministic or stochastic mathematical technique to combine them all.

Functional surrogate models have a mathematical nature different from the true, original functions. The knowledge of truth is revealed implicitly in the values of their coefficients.

Functional surrogate models are not (at least entirely) specific to a class of problems.

They are strongly dependent on samples of the true function and are applicable to a wide variety of problems.

Surrogate Models

A surrogate model can be used only for modeling and analysis, in order to gain insight about problem features and behavior without many expensive evaluations.

A surrogate model can take the place of the true function for the purposes of optimization.

A surrogate model is typically less accurate or has less quality than the true function, but is cheaper to evaluate or consumes fewer computing resources.

Several evaluations of the surrogate model can still be less expensive than one evaluation of the true function.

Global Optimization

Very recently Global Optimization regain increasing importance because it has been observed that a significant number of users of DFO packages demand global optimizers even when originally thought otherwise.

It is also frequent the case that the problems are single objective only because the user has linearly combined different goals.

Multicomponent Optimization

What has been observed in Optimization in general in the last couple of decades has been a focus on specialized classes of problems where structure can be explored efficiently.

The result has been the appearance of several subareas and individualized topics for which the best international researchers have developed theories and algorithms and written high quality software.

As a result, one is faced with hundreds of codes, most of them of public domain, written in different languages and using different interfaces, and, more importantly, directed to specific problems.

This current status of the optimization theory and software poses a problem for contemporary application problems which are typically multicomponent.