Optimisation

Alexis Tsoukiàs

LAMSADE - CNRS, Université Paris-Dauphine
tsoukias@lamsade.dauphine.fr
http://www.lamsade.dauphine.fr/∼tsoukias

Napoli 31/05/2011

Outline

Is optimisation rational?

General Setting

$$\min F(x)$$
$$x \in S \subseteq K^n$$

where:
- x is a vector of variables
- S is the feasible space
- K^n is a vector space, $(\mathbb{Z}^n, \mathbb{R}^n, \{0,1\}^n)$.
- $F : S \mapsto \mathbb{R}^m$

Well known specific cases: m=1

- $F(x)$ is linear, S is a n-dimensional polytope: linear programming
  $\min cx$, $Ax \leq b$, $x \geq 0$.
- S is a n-dimensional polytope, but $F : \mathbb{R}^{n+m} \mapsto \mathbb{R}$: constraint satisfaction
  $\min y$, $Ax + y \leq b$, $x, y \geq 0$.
- $F(x)$ is linear, $S \subseteq \{0, 1\}^n$: combinatorial optimisation.
- $F(x)$ is convex and S is a convex subset of $\mathbb{R}^n$: convex programming

More challenging cases

- Instead of $\min_{x \in S} F(x)$ we get $\sup_{x \in S} x$. Practically we only have a preference relation on S (and thus we cannot define any "quantitative" function of x).

NB

The problem becomes tricky when the preference relation cannot be represented explicitly (for instance when $S \subseteq \{0, 1\}^n$)

- $m > 1$. We get

$$F(x) = \langle f_1(x) \cdots f_n(x) \rangle$$

Practically a problem mathematically undefinable ...

- Combinations of the two cases above as well as of the previous ones ...

More challenging cases

- Instead of $\min_{x \in S} F(x)$ we get $\sup_{x \in S} x$. Practically we only have a preference relation on S (and thus we cannot define any "quantitative" function of x).

NB

The problem becomes tricky when the preference relation cannot be represented explicitly (for instance when $S \subseteq \{0, 1\}^n$)

- $m > 1$. We get

$$F(x) = \langle f_1(x) \cdots f_n(x) \rangle$$

Practically a problem mathematically undefinable ...

- Combinations of the two cases above as well as of the previous ones ...

More challenging cases

- Instead of $\min_{x \in S} F(x)$ we get $\sup_{x \in S} x$. Practically we only have a preference relation on S (and thus we cannot define any "quantitative" function of x).

NB

The problem becomes tricky when the preference relation cannot be represented explicitly (for instance when $S \subseteq \{0, 1\}^n$)

- $m > 1$. We get

$$F(x) = \langle f_1(x) \cdots f_n(x) \rangle$$

Practically a problem mathematically undefinable ...

- Combinations of the two cases above as well as of the previous ones ...

More challenging cases

- Instead of $\min_{x \in S} F(x)$ we get $\sup_{x \in S} x$. Practically we only have a preference relation on S (and thus we cannot define any "quantitative" function of x).

NB

The problem becomes tricky when the preference relation cannot be represented explicitly (for instance when $S \subseteq \{0, 1\}^n$)

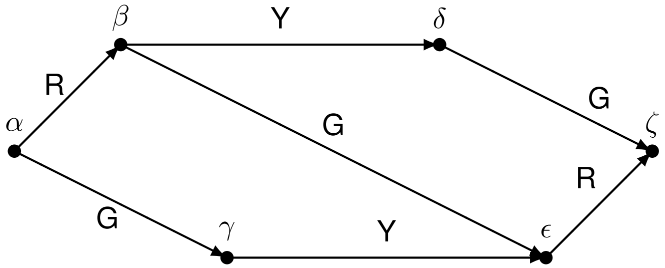
- $m > 1$. We get

$$F(x) = \langle f_1(x) \cdots f_n(x) \rangle$$

Practically a problem mathematically undefinable ...

- Combinations of the two cases above as well as of the previous ones ...

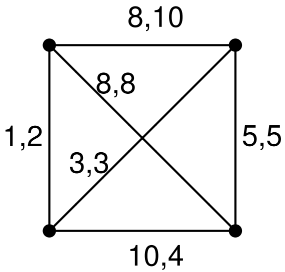
Example



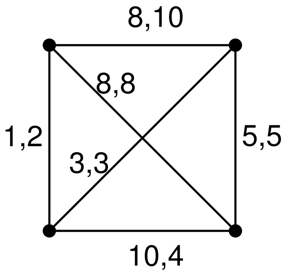
- R: dangerous
- Y: fairly dangerous
- G: not dangerous

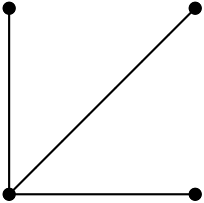
Which is the safest path in the network?

Example 2

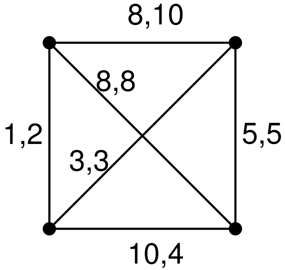


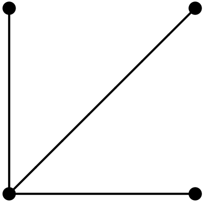
Example 2



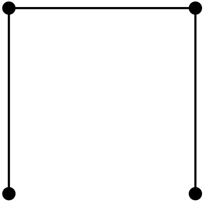


Example 2



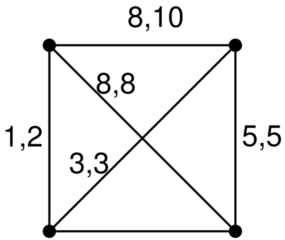


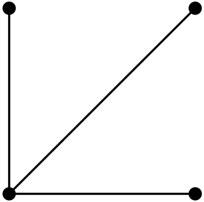
Sol. 1: 14,9



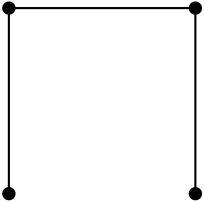
Sol. 2: 8,17

Example 2

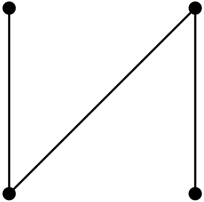




Sol. 1: 14,9



Sol. 2: 8,17



Robust: 9,10

First Idea

Find all "non dominated solutions" and then explore it appropriately (straightforward or interactively) until a compromise is established. BUT:

- The set of all such solutions can be extremely large, an explicit enumeration becoming often intractable.
- Depending on the shape and size of the set of the "non dominated solutions", exploring the set can be intractable.

Further Ideas

- Instead trying to construct the whole set of "non dominated solutions", concentrate the search of the compromise in an "interesting" subset. Problem: how to define and describe the "interesting" subset?
- Aggregate the different objective functions (the criteria) to a single one and then apply mathematical programming:
  - scalarising functions;
  - distances.

Scalarising Functions

We transform

$$\min_{x \in S} [f_1(x) \cdots f_n(x)]$$

to the problem

$$\min_{x \in S} \lambda^T F(x)$$

λ being a vector of trade-offs. Problem: how do we get them?

Scalarising Functions

We transform

$$\min_{x \in S}[f_1(x) \cdots f_n(x)]$$

to the problem

$$\min_{x \in S} \lambda^T F(x)$$

λ being a vector of trade-offs. Problem: how do we get them?

This turns to be a parametric optimisation problem

Add Constraints

We transform

$$\min_{x \in S}[f_1(x) \cdots f_n(x)]$$

to the problem

$$\min_{x \in S} f_k(x)$$
$$\forall j \neq k f_j \leq \epsilon_j$$

ϵ_j being a vector of constants. Problem: how do we get them?

Add Constraints

We transform

$$\min_{x \in S}[f_1(x) \cdots f_n(x)]$$

to the problem

$$\min_{x \in S} f_k(x)$$
$$\forall j \neq k f_j \leq \epsilon_j$$

ϵ_j being a vector of constants. Problem: how do we get them?

This turns to be a parametric optimisation problem

Tchebychev Distances

We transform

$$\min_{x \in S} [f_1(x) \cdots f_n(x)]$$

to the problem

$$\min_{x \in S} [\max_{j=1 \cdots m} w_j(f_j(x) - y_j)]$$

w_j being a vector of trade-offs. Problem: how do we get them?
y_j being a special point (for instance the ideal point) in the objective space

Combinatorial Optimisation

What happens if we have to choose among collections of objects, while we only know the values of the objects?

1. Knapsack Problems
2. Network Problems
3. Assignment Problems

Combinatorial Optimisation

What happens if we have to choose among collections of objects, while we only know the values of the objects?

1. Knapsack Problems
2. Network Problems
3. Assignment Problems

What if there are interactions (positive or negative synergies) among the chosen objects?

The Choquet Integral

Given a set N, a function $v : 2^N \mapsto [0, 1]$ such that:
- $v(\emptyset) = 0$, $V(N) = 1$
- $\forall A, B \in 2^N : A \subseteq B \;\; v(A) \leq v(B)$

is a capacity

The Choquet Integral

Given a set N, a function $v : 2^N \mapsto [0,1]$ such that:
- $v(\emptyset) = 0$, $V(N) = 1$
- $\forall A, B \in 2^N : A \subseteq B \;\; v(A) \leq v(B)$

is a capacity

We use the Choquet Integral

$$C_v(f) = \sum_{i=1}^{n} [f(\sigma(i)) - f(\sigma(i-1))] v(A_i)$$

which is a measure of a capacity where:
- f represent the value function for x;
- $\sigma(i)$ represents a permutation on A_i such that:

$f(\sigma(0)) = 0$ and $f(\sigma(1)) \leq \cdots f(\sigma(n))$

Several Models Together

The Choquet Integral contains as special cases several models:

- The weighted sum.
- The k-additive model
- The expected utility model.
- The Ordered Weighted Average model
- The Rank Depending Utility model

Lessons Learned

- Optimising is not necessary "rational".
- Optimising multiple objectives simultaneously is ill defined and "difficult".
- We can improve using preference based models.
- We need to (and we can) take into account the possible interactions among objects or among objectives.
- We need "good" approximation algorithms.

Lessons Learned

- Optimising is not necessary “rational”.
- Optimising multiple objectives simultaneously is ill defined and “difficult”.
- We can improve using preference based models.
- We need to (and we can) take into account the possible interactions among objects or among objectives.
- We need “good” approximation algorithms.

Lessons Learned

- Optimising is not necessary “rational”.
- Optimising multiple objectives simultaneously is ill defined and “difficult”.
- We can improve using preference based models.
- We need to (and we can) take into account the possible interactions among objects or among objectives.
- We need “good” approximation algorithms.

Lessons Learned

- Optimising is not necessary “rational”.
- Optimising multiple objectives simultaneously is ill defined and “difficult”.
- We can improve using preference based models.
- We need to (and we can) take into account the possible interactions among objects or among objectives.
- We need “good” approximation algorithms.

Lessons Learned

- Optimising is not necessary “rational”.
- Optimising multiple objectives simultaneously is ill defined and “difficult”.
- We can improve using preference based models.
- We need to (and we can) take into account the possible interactions among objects or among objectives.
- We need “good” approximation algorithms.

Books

- Deb K., *Multi-Objective Optimization using Evolutionary Algorithms*, J. Wiley, New York, 2001.
- Ehrgott M., *Multicriteria Optimisation*, Springer Verlag, Berlin, 2000.
- Ehrgott M., Gandibleux X., *Multiple Criteria Optimization. State of the art annotated bibliographic surveys*, Kluwer Academic, Dordrecht, 2002.
- Figueira J., Greco S., EhrgottM., *Multiple Criteria Decision Analysis: State of the Art Surveys*, Springer Verlag, Berlin, 2005.
- M. Grabisch, T. Murofushi, M. Sugeno, and J. Kacprzyk. *Fuzzy Measures and Integrals. Theory and Applications.* Physica Verlag, Berlin, 2000.
- Kouvelis P., Yu G., *Robust discrete optimization and its applications*, Kluwer Academic, Dodrecht, 1997.
- Steuer R.E., *Multiple Criteria Optimization: Theory, Computation and Application*, John Wiley, New York, 1986.