Multi-Objective Optimization Using Metaheuristics

Carlos A. Coello Coello

ccoello@cs.cinvestav.mx
CINVESTAV-IPN
Evolutionary Computation Group (EVOCINV)
Computer Science Department
Av. IPN No. 2508, Col. San Pedro Zacatenco
México, D.F. 07360, MEXICO

Eindhoven University of Technology
Eindhoven, The Netherlands July 2023
LECTURE 1

Motivation

Most problems in nature have several (possibly conflicting) objectives to be satisfied (e.g., design a bridge for which want to minimize its weight and cost while maximizing its safety). Many of these problems are frequently treated as single-objective optimization problems by transforming all but one objective into constraints.

Formal Definition

Find the vector $\vec{x}^* = \left[x_1^*, x_2^*, \ldots, x_n^*\right]^T$ which will satisfy the m inequality constraints:

$$g_i(\vec{x}) \leq 0 \quad i = 1, 2, \ldots, m \tag{1}$$

the p equality constraints

$$h_i(\vec{x}) = 0 \quad i = 1, 2, \ldots, p \tag{2}$$

and will optimize the vector function

$$\vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \ldots, f_k(\vec{x})]^T \tag{3}$$

Basic Concepts

In order to know how “good” a certain solution is, it is necessary to have some criteria to evaluate it. These criteria are expressed as computable functions of the decision variables, that are called **objective functions.**

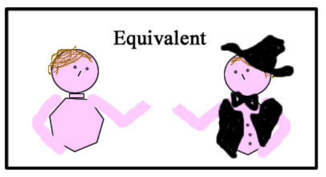
In real-world problems, some of these objective functions are in *conflict* with others, and some have to be minimized while others are maximized. These objective functions may be **commensurable** (measured in the same units) or **non-commensurable** (measured in different units).

Basic Concepts

In Operations Research, it is a common practice to differentiate among attributes, criteria, objectives and goals.

Attributes are often thought of as differentiating aspects, properties or characteristics of alternatives or consequences. **Criteria** generally denote evaluative measures, dimensions or scales against which alternatives may be gauged in a value or worth sense. **Objectives** are sometimes viewed in the same way, but may also denote specific desired levels of attainment or vague ideals. **Goals** usually indicate either of the latter notions. A distinction commonly made in Operations Research is to use the term **goal** to designate potentially attainable levels, and **objective** to designate unattainable ideals.

Basic Concepts



Several researchers use the terms **objective**, **criteria**, and **attribute** interchangeably to represent an MOP's goals or objectives (i.e., distinct mathematical functions) to be achieved. The terms **objective space** or **objective function space** are also used to denote the coordinate space within which vectors resulting from evaluating an MOP are plotted.

Basic Concepts

Ideal Objective Vector

It is an objective vector minimizing each of the objective functions. The components z_i^* of the **ideal objective vector** $\mathbf{z}^* \in \mathbb{R}^k$ are obtained by minimizing each of the objective functions individually, subject to the constraints. That is, it is obtained by solving:

$$\text{minimize } f_i(\vec{x}) \tag{4}$$

$$\text{subject to } \vec{x} \in \mathcal{F}, \text{ for } i = 1, \ldots, k \tag{5}$$

Basic Concepts

Ideal Objective Vector

The ideal vector is unreachable in most cases (except when there is no conflict among the objectives). However, the ideal vector is adopted by some mathematical programming techniques in which normally the idea is to minimize the distance of a solution with respect to such ideal vector.

Generating the ideal vector is not particularly complicated (except when some (or all) of the objective functions, when considered in isolation, presents multimodality). However, its generation has an additional computational cost that is not always affordable. Some multi-objective metaheuristics adopt an approximation of the ideal vector that is updated at each iteration.

Basic Concepts

Utopian Objective Vector

Some authors (e.g., Miettinen [1999]) also consider the concept of **utopian objective vector.**

The utopian objective vector is defined as: $\mathbf{z}^{**} \in \mathbb{R}^k$ and it is an infeasible objective vector whose components are formed by:

$$z^{**} = z_i^* - \varepsilon_i \tag{6}$$

for every $i = 1, \ldots, k$, where z_i^* is a component of the ideal objective vector and $\varepsilon_i > 0$ is a scalar which is relatively small, but computationally significant. Clearly, the utopian objective vector is strictly better (i.e., it strictly dominates) every Pareto optimal solution.

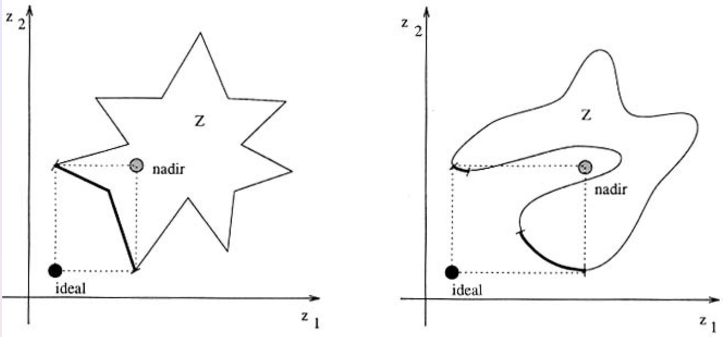
Basic Concepts

Nadir Objective Vector

It refers to the upper bounds of the Pareto optimal set. It is normally denoted as $\mathbf{z}^{nad}$ and its components are normally quite difficult to obtain.

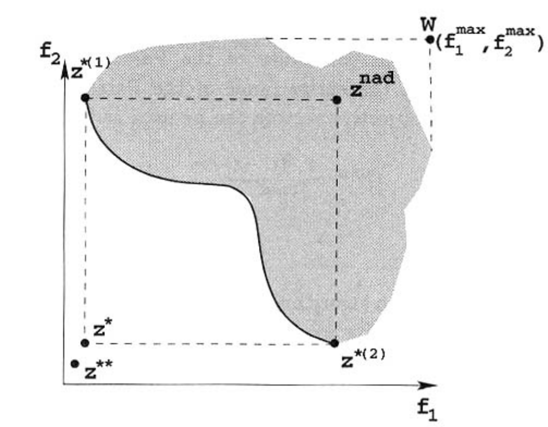
When computing the ideal vector, normally a **payoff table** is created using the decision vectors obtained. Row i of the payoff table displays the values of all the objective functions calculated at the point where f_i obtained its minimal value. Hence, z_i^* is at the main diagonal of the table. The maximal value of the column i in the payoff table can be selected as an estimate of the upper bound of the objective f_i for $i = 1, \ldots, k$ over the Pareto optimal set.

Basic Concepts



In this figure, a black circle is used to indicate the Ideal Objective Vectors and a gray circle is use to indicate the Nadir Objective Vectors. It is worth noting that the Nadir Objective Vector may be infeasible.

Basic Concepts



This figure shows the Ideal Objective Vector (z^*), the Utopian Objective Vector (z^{**}) and the Nadir Objective Vector (z^{nad}).

Basic Concepts

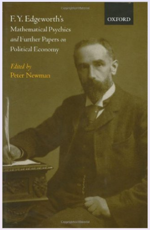
In multiobjective optimization problems, there are three possible situations:

- Minimize all the objective functions
- Maximize all the objective functions
- Minimize some and maximize others

For simplicity reasons, normally all the functions are converted to a maximization or minimization form. For example, the following identity may be used to convert all the functions which are to be maximized into a form which allows their minimization:

$$\max \; f_i(\vec{x}) = \min(-f_i(\vec{x})) \tag{7}$$

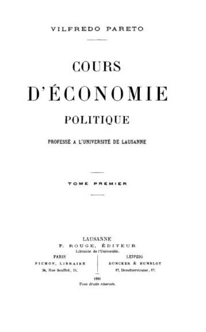
Basic Concepts



Having several objective functions, the notion of “optimum” changes, because in MOPs, the aim is to find good compromises (or “trade-offs”) rather than a single solution as in global optimization.

The notion of “optimum” that is most commonly adopted is that originally proposed by Francis Ysidro Edgeworth (in 1881) in his book entitled **Mathematical Psychics.**

Basic Concepts



VILFREDO PARETO

COURS
D'ÉCONOMIE
POLITIQUE

PROFESSÉ A L'UNIVERSITÉ DE LAUSANNE

TOME PREMIER

LAUSANNE
F. ROUGE, ÉDITEUR
Librairie de l'Université.

PARIS
PICHON, LIBRAIRE
24, Rue Soufflot, 24.

LEIPZIG
DUNCKER & HUMBLOT
17, Dresdnerstrasse, 17.

1896
Tous droits réservés.

This notion was generalized by the italian economist Vilfredo Pareto (in 1896) in his book **Cours d'Economie Politique.** Although some authors call *Edgeworth-Pareto optimum* to this notion (originally called **ophelimity**) it is normally preferred to use the most commonly accepted term: **Pareto optimum.**

Basic Concepts

Pareto Optimality

We say that a vector of decision variables $\vec{x}^* \in \mathcal{F}$ is **Pareto optimal** if there does not exist another $\vec{x} \in \mathcal{F}$ such that $f_i(\vec{x}) \leq f_i(\vec{x}^*)$ for all $i = 1, \ldots, k$ and $f_j(\vec{x}) < f_j(\vec{x}^*)$ for at least one j (assuming that all the objectives are being minimized).

Other important definitions

In words, this definition says that $\vec{x}^*$ is **Pareto optimal** if there exists no feasible vector of decision variables $\vec{x} \in \mathcal{F}$ which would decrease some criterion without causing a simultaneous increase in at least one other criterion. This concept normally produces a set of solutions called the **Pareto optimal set**. The vectors $\vec{x}^*$ corresponding to the solutions included in the Pareto optimal set are called **nondominated**. The image of the Pareto optimal set is called the **Pareto front**.

Basic Concepts

Pareto Dominance

A vector $\vec{u} = (u_1, \ldots, u_k)$ is said to **dominate** $\vec{v} = (v_1, \ldots, v_k)$ (denoted by $\vec{u} \preceq \vec{v}$) if and only if u is partially less than v, i.e., $\forall i \in \{1, \ldots, k\},\ u_i \leq v_i \wedge \exists i \in \{1, \ldots, k\} : u_i < v_i$.

Pareto Optimal Set

For a given MOP $\vec{f}(x)$, the Pareto optimal set $(\mathcal{P}^*)$ is defined as:

$$\mathcal{P}^* := \{x \in \mathcal{F} \mid \neg\exists\, x' \in \mathcal{F}\ \ \vec{f}(x') \preceq \vec{f}(x)\}. \tag{8}$$

Pareto Front

For a given MOP $\vec{f}(x)$ and Pareto optimal set $\mathcal{P}^*$, the Pareto front $(\mathcal{PF}^*)$ is defined as:

$$\mathcal{PF}^* := \{\vec{u} = \vec{f} = (f_1(x), \ldots, f_k(x)) \mid x \in \mathcal{P}^*\}. \tag{9}$$

Basic Concepts

The concept of **Pareto dominance** implies that, for a solution to *dominate* another one, it should not be worse in any objective and must be strictly better in at least one of them.

Consequently, when comparing two solutions **A** and **B**, using Pareto dominance, there are three possible outcomes:

1. **A** dominates **B**.
2. **A** is dominated by **B**.
3. **A** and **B** are not dominated by each other (i.e., they are both non-dominated).

Basic Concepts

Properties of the dominance relation

Cormen et al. [1990] provide the properties of the dominance relation:

- **Reflection:** The dominance relation **is not reflexive**, because any relation p does not dominate itself.
- **Symmetry:** The dominance relation **is not symmetric** because $p \preceq q$ does not imply $q \preceq p$. In fact, the opposite is true. In other words, if p dominates q, then q does not dominate p. Therefore, the dominance relation is **asymmetric**.
- **Antisymmetry:** Since the dominance relation is not symmetric, it can't be **antisymmetric**.
- **Transitivity:** The dominance relation is **transitive**. This is because if $p \preceq q$ and $q \preceq r$, then $p \preceq r$.

Basic Concepts

Properties of the dominance relation

Another interesting property of the dominance relation is that if a solution *p* does not dominate another solution *q*, this does not imply that *q* dominates *p*.

For a binary relation to qualify as an order relation, it must be at least *transitive* [Chankong & Haimes, 1983]. Thus, the dominance relation is an order relation. However, since the dominance relation is not reflexive, it is a **strict partial order**.

Basic Concepts

Properties of the dominance relation

In general, if a relation is reflexive, antisymmetric and transitive, it is called (in a general sense) a **partial order**. A set in which a partial order is defined is called **partially ordered set.**

However, it is important to emphasize that the dominance relation is not reflexive and is not antisymmetric. Therefore, the dominance relation is not a partial order, but only a strict partial order.

Basic Concepts

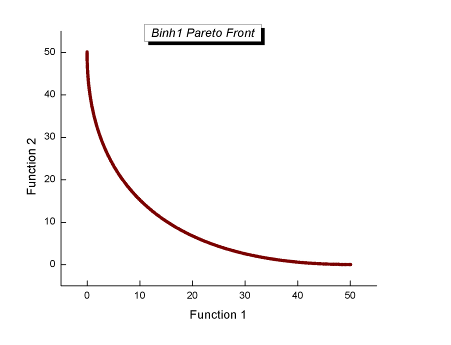
Weak Pareto Optimality

A point $\vec{x}^* \in \mathcal{F}$ is a **weakly Pareto optimal solution** if there is no $\vec{x} \in \mathcal{F}$ such that $f_i(\vec{x}) < f_i(\vec{x}^*)$, for $i = 1, \ldots, k$.

Strong Pareto Optimality

A point $\vec{x}^* \in \mathcal{F}$ is a **strongly Pareto optimal solution** if there is $\vec{x} \in \mathcal{F}$ such that $f_i(\vec{x}) \leq f_i(\vec{x}^*)$, for $i = 1, \ldots, k$.

Basic Concepts



Pareto Front

For a given multi-objective optimization problem $\vec{f}(x)$ and a Pareto optimal set $\mathcal{P}^*$, the Pareto Front ($\mathcal{PF}^*$) is defined as:

$$\mathcal{PF}^* := \{\vec{u} = \vec{f} = (f_1(x), \ldots, f_k(x)) \mid x \in \mathcal{P}^*\}. \qquad (10)$$

Basic Concepts

Pareto Front

In general, it is impossible to find an analytical expression that represents the line or hyper-surface corresponding to the Pareto Optimal Front. This is possible only in very simple (textbook) cases.

The normal procedure for generating (an approximation of) the Pareto optimal front of a problem is to compute all the (or as many as possible) feasible points and to obtain their corresponding objective function values. When we had obtained a sufficient number of such points, it is possible to determine which are the nondominated solutions from them (the Pareto optimal set). As indicated before, the image of the Pareto optimal set is the Pareto front.

Basic Concepts

Algorithm 1 to Obtain Nondominated Solutions (Simple Method)

Algorithm

Step 1: Set a counter $i = 1$ and create a (empty) set of nondominated solutions P'

Step 2: For a solution $j \in P$, where P is the population ($j \neq i$), check if the solution j dominates solution i. If it dominates it, go to Step 4.

Step 3: If there are more solutions left in P, increase j by one and go to Step 2; otherewise, $P' = P' \cup \{i\}$

Step 4: Increase i by one. If $i \leq N$, go to Step 2; otherwise, stop and declare P as the nondominated set.

Basic Concepts

Algorithm 2 to Obtain Nondominated Solutions (Continuous Update)

```
Algorithm
Step 1:  Initialize $P' = \{1\}$. Set a counter $i = 2$.
Step 2:  Set $j = 1$.
Step 3:  Compare solutions $i$ and $j$ from $P'$.
Step 4:  If $i$ dominates $j$, then delete the $j^{th}$ member of $P'$.
         If $j < |P'|$, then increase $j$ by one and go to Step 3.
         Otherwise, go to Step 5.
         Else if the $j^{th}$ member of $P'$ dominates $i$,
         increase $i$ by one and go to Step 2.
         Alternatively, if the $j^{th}$ member of $P'$ and $i$ are mutually
         nondominated, then increase $j$ by one if $j < |P'|$ and go to step 3;
         else go to step 5.
Step 5:  Insert $i$ in $P'$. If $i < N$, increase $i$ by one and
         go to Step 2.
         Otherwise, stop and declare $P'$ as the nondominated
         set.
```

Computational Efficiency

Both Algorithm 1 and Algorithm 2 shown before have an algoritmic complexity $O(MN^2)$ in the worst case. In this case M is the number of objectives and N is the number of solutions. However, in practice, Algorithm 2 requires about half of the computational effort required by Algorithm 1.

So the obvious question is: **can we obtain nondominated solutions in a more efficient way (computational speaking)?**

Basic Concepts

Algorithm 3 to Obtain Nondominated Solutions

Theoretically, the most efficient algorithm that we can have for obtaining nondominated solutions is the one proposed by Kung et al. [1975]. This algorithm requires the set (or population) to be sorted based on the first objective. Then, the population is recursively divided in two halves: (S) superior and (I) inferior. Knowing that the first half is better than the second in terms of the first objective function, the inferior half is checked (in terms of Pareto dominance) with respect to the superior half.

Basic Concepts

Algorithm 3 to Obtain Nondominated Solutions

The solutions of (I) which are not dominated by any member of (S) are combined with the members of (S) to form a mixed population M. This union and dominance checking starts with the most inner case (in which there is only one member either in S or in I, after performing several recursive divisions of the population) and then the algorithm continues in a bottom up manner.

Algorithm 3 to Obtain Nondominated Solutions

This algorithm has a complexity $O(N(\log M)^{M-2})$ for $M \geq 4$ and $O(N \log N)$ for $M = 2$ y $M = 3$.

So, as we increase the number of objectives, Algorithm 3 also approximates the quadratic complexity of Algorithm 1 and Algorithm 2. However, for 2 or 3 objectives, this algorithm is clearly more efficient.

Basic Concepts

For more on Kung's Algorithm

H.T. Kung, F. Luccio, and F.P. Preparata, "**On finding the maxima of a set of vectors**", *Journal of the Association for Computing Machinery*, **22**(4):469–476, 1975.

For more information on this topic

- J.L. Bentley, H.T. Kung, M. Schkolnick, and C.D. Thompson, "**On the Average Number of Maxima in a Set of Vectors and Applications**", *Journal of the Association for Computing Machinery*, **25**(4):536–543, October 1978.
- Jon Louis Bentley, "**Multidimensional Divide-and-Conquer**", *Communications of the ACM*, **23**(4):214–229, April 1980.

Basic Concepts

For more information on this topic

- Jon L. Bentley, Kenneth L. Clarkson, and David B. Levine, **Fast Linear Expected-Time Algorithms for Computing Maxima and Convex Hulls**", *Algorithmica*, 9:168–183, 1993.
- Lixin Ding, Sanyou Zheng, and Lishan Kang, "**A Fast Algorithm on Finding the Non-dominated Set in Multi-objective Optimization**", in *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, Vol. 4, pp 2565–2571, IEEE Press, Canberra, Australia, December 2003.
- Michael A. Yukish, **Algorithms to Identify Pareto Points in Multi-Dimensional Data Sets**, PhD thesis, College of Engineering, Pennsylvania State University, USA, August 2004.

Basic Concepts

For more information on this topic

- Parke Godfrey, Ryan Shipley, and Jarek Gryz, "**Maximal Vector Computation in Large Data Sets**", Technical Report CS-2004-06, Department of Computer Science and Engineering, York University, Canada, December 2004.
- K.K. Mishra and Sandeep Harit, "**A Fast Algorithm for Finding the Non Dominated Set in Multiobjective Optimization**", *International Journal of Computer Applications*, **1**(25):35–39, 2010.
- Maxim Buzdalov and Anatoly Shalyto, "**A Provably Asymptotically Fast Version of the Generalized Jensen Algorithm for Non-dominated Sorting**", in Thomas Bartz-Beielstein et al. (Eds), *Parallel Problem Solving from Nature - PPSN XIII, 13th International Conference*, pp. 528–537. Springer. Lecture Notes in Computer Science Vol. 8672, Ljubljana, Slovenia, September 13-17 2014.

For more information on this topic

- Xingyi Zhang, Ye Tian, Ran Cheng and Jin Yaochu, “**An Efficient Approach to Nondominated Sorting for Evolutionary Multiobjective Optimization**”, *IEEE Transactions on Evolutionary Computation*, Vol. 19, No. 2, pp. 201–213, April 2015.
- Martin Drozdik, Youhei Akimoto, Hernan Aguirre and Kiyoshi Tanaka, “**Computational Cost Reduction of Nondominated Sorting Using the M-Front**”, *IEEE Transactions on Evolutionary Computation*, Vol. 19, No. 5, pp. 659–678, October 2015.
- Kent McClymont and Ed Keedwell, “**Deductive Sort and Climbing Sort: New Methods for Non-Dominated Sorting**”, *Evolutionary Computation*, Vol. 20, No. 1, pp. 1–26, Spring 2012.

Basic Concepts

Optimality Conditions

Fritz-John's Necessary Condition. A necessary condition for $\mathbf{x}^*$ to be Pareto optimal is that there exist vectors $\lambda \geq 0$ and $\mathbf{u} \geq 0$ (where $\lambda \in \mathbb{R}^M$, $\mathbf{u} \in \mathbb{R}^J$ and $\lambda, \mathbf{u} \neq 0$) such that the following conditions hold:

1. $\sum_{m=1}^{M} \lambda_m \nabla f_m(\mathbf{x}^*) - \sum_{j=1}^{f} u_j \nabla g_j(\mathbf{x}^*) = 0$
2. $u_j g_j(\mathbf{x}^*) = 0$ for every $j = 1, 2, \ldots, J$.

Basic Concepts

Optimality Conditions

These conditions are very similar to the Kuhn-Tucker conditions of optimality for single-objective problems. The difference lies on the addition (in this case) of the vector of the gradients of the objectives.

For an unconstrained multi-objective optimization problem, the previous theorem requires the following condition:

$$\sum_{m=1}^{M} \lambda_m \nabla f_m(\mathbf{x}^*) = 0 \tag{11}$$

for a solution to be Pareto optimal.

Basic Concepts

Optimality Conditions

For nonlinear objective functions, it is expected that the partial derivatives are nonlinear. For a given vector λ, it is possible to check the non-existence of a Pareto optimal solution using the previously defined conditions.

If the necessary conditions are not satisfied, then there does not exist a Pareto optimal solution corresponding to the given vector λ. It is worth noting, however, that since this is a necessary condition, the existence of a solution that is Pareto optimal is not guaranteeed. In other words, a solution that satisfies these conditions is not necessarily Pareto optimal.

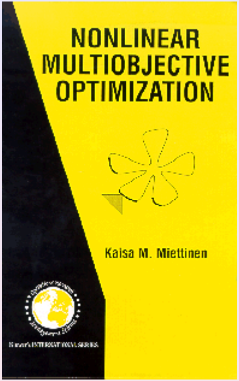
Basic Concepts

Optimality Conditions

Kuhn-Tucker's Sufficiency Conditions for Pareto Optimality: Let's assume that the objective functions are convex and the constraints are non-convex. Let's assume that the objective functions and the constraints are continuously differentiable in a feasible solution $\mathbf{x}^*$. A sufficient condition for $\mathbf{x}^*$ to be Pareto optimal is that there exist vectors $\lambda > 0$ and $u \geq 0$ (where $\lambda \in \mathbb{R}^M$ and $u \in \mathbb{R}^J$) such that the following equations hold:

1. $\sum_{m=1}^{M} \lambda_{\mathbf{m}} \nabla f_i(\mathbf{x}^*) - \sum_{j=1}^{J} u_j \nabla g_j(\mathbf{x}^*) = 0$
2. $u_j g_j(\mathbf{x}^*) = 0$ for every $j = 1, 2, \ldots, J$.

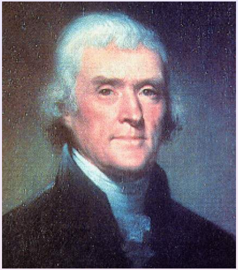
Basic Concepts


NONLINEAR MULTIOBJECTIVE OPTIMIZATION
Kaisa M. Miettinen
Kluwer's INTERNATIONAL SERIES

For more information

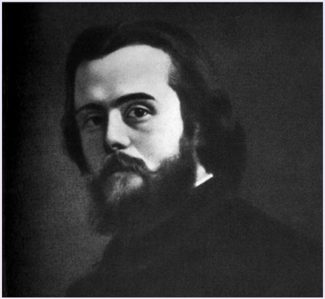
Kaisa M. Miettinen, **Nonlinear Multiobjective Optimization**, Kluwer Academic Publishers, Boston, Massachusetts, 1999.

Historical Highlights of Multi-Objective Optimization



Multiobjective optimization is an intrinsic part of economic equilibium theory and, as such, it can be said to have been founded by Adam Smith in his famous treatise entitled **An Inquiry into the Nature and Causes of the Wealth of Nations**, in 1776.

Historical Highlights of Multi-Objective Optimization



The concept of general economic equilibrium is normally attributed to Léon Walras (1834-1910). Within economic equilibrium theory, the most relevant works (besides those of Walras) are those from Jevons and Menger on utility theory, and the work on welfare theory by Edgeworth and Pareto, spanning the period from 1874 to 1906.

Historical Highlights of Multi-Objective Optimization

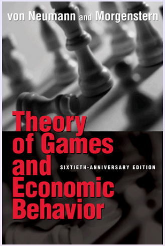
A second area that is considered to be one of the main precursors of multi-objective optimization is the inception of the psycological theory of games and the notion of (game) strategy.

Games of chance have a very ancient history. However, Félix Édouard Justin Émile Borel (1871-1956) is normally considered as the one who started the psicological theory of games and the one who introduced the formal definition of strategies that are based on analyzing the psychology of the opponent.

Game Theory

The so-called **game theory** can be traced back to a work by Borel from 1921. However, many historians normally attribute the origins of game theory to a paper from the famous hungarian mathematician John von Neumann which was orally presented in 1926 and published in 1928.

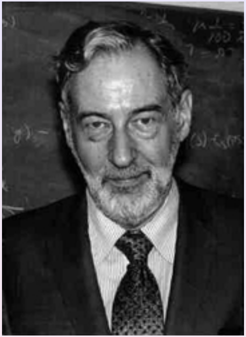
Historical Highlights of Multi-Objective Optimization


von Neumann and Morgenstern
Theory of Games and Economic Behavior
SIXTIETH-ANNIVERSARY EDITION

Game Theory

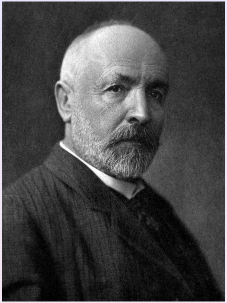
In 1944, John von Neumann and Oskar Morgenstern mentioned (in their famous book on **Game Theory**) that they had found a problem in economics that was a “peculiar and disconcerting mixture of several problems in conflict with each other” which could not be solved with the classical optimization methods known at that time. It remains a mystery why is that von Neummann did not get interested in this peculiar problem.

Historical Highlights of Multi-Objective Optimization



In 1951, Tjalling C. Koopmans edited a book entitled **Activity Analysis of Production and Allocation**, in which the concept of **efficient** vector (which is the same as a **nondominated vector**) was used in a meaningful way for the very first time.

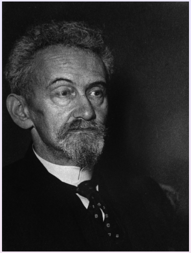
Historical Highlights of Multi-Objective Optimization



Mathematical Foundations

The origins of the mathematical foundations of multi-objective optimization can be traced back to the period from 1895 to 1906 in which Georg Cantor and Felix Hausdorff established the foundations of ordered spaces of infinite dimensions.

Historical Highlights of Multi-Objective Optimization

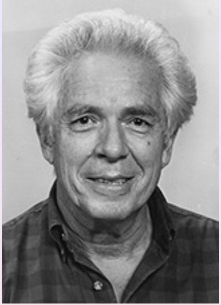


Mathematical Foundations

Cantor also introduced equivalent classes and established the first set of sufficiency conditions for the existence of a utility function.

Hausdorff provided the first example of a complete ordering.

Historical Highlights of Multi-Objective Optimization



Mathematical Foundations

However, it was the concept of the **maximum vector problem** introduced by Harold W. Kuhn and Albert W. Tucker (1951) which allowed multi-objective optimization to become a mathematical discipline on its own.

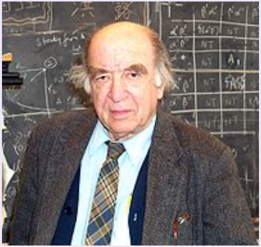
Historical Highlights of Multi-Objective Optimization

Mathematical Foundations

It is well-known that the now famous conditions of optimality commonly attributed to Kuhn and Tucker had been previously stated and proved by W. Karush in an unpublished Masters thesis in 1939.

Kuhn and Tucker gave credit to Karush, which is the reason why many books call them **Karush-Kuhn-Tucker (KKT) Conditions**.

Historical Highlights of Multi-Objective Optimization



Mathematical Foundations

Nevertheless, the theory of multi-objective optimization remained practically unexplored during the 1950s. It was until the 1960s, in which the mathematical foundations of the area were consolidated when Leonid Hurwicz generalized Kuhn and Tucker's results to topological vector spaces.

Historical Highlights of Multi-Objective Optimization

Kenneth J. Arrow did some very important pioneering work in the 1950s using the concept of admissible points and stating his famous *impossibility theorem* which relates to multi-criteria decision making.

Historical Highlights of Multi-Objective Optimization

Goal Programming

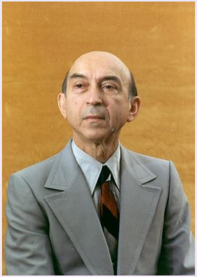
Perhaps the most important outcome from the 1950s was the development of **Goal Programming**, which was originally introduced by Abraham Charnes and William Wager Cooper in 1957. However, Goal Programming became popular in the 1960s.

Historical Highlights of Multi-Objective Optimization

Applications

The first application of multi-objective optimization outside economics was done by Koopmans (1951) in production theory. Later on, Marglin (1967) developed the first applications of multi-objective optimization in water resources.

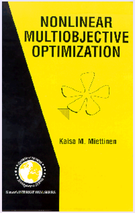
Historical Highlights of Multi-Objective Optimization



Applications

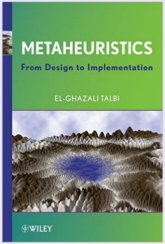
The first engineering application of multi-objective optimization reported in the literature is a paper published by Lofti Zadeh in the early 1960s (related to automatic control). However, multi-objective optimization applications generalized until the 1970s.

Current State



Currently, there are some 30 mathematical programming techniques for nonlinear multi-objective optimization. However, they have several limitations. For example, some of them require that the objectives (and the constraints) are differentiable. Other approaches cannot be applied to disconnected or to non-convex Pareto fronts. Additionally, most of them generate a single solution per algorithmic execution.

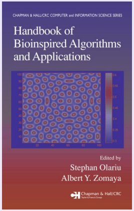
Current State



This has motivated the use of metaheuristics (particularly, bio-inspired metaheuristics).

A **metaheuristic** is a high-level search procedure that applies some form of rule or set of rules based on some source of knowledge, in order to explore the search space in a more efficient way.

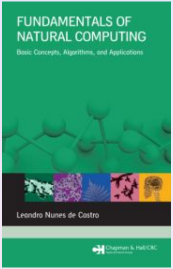
Current State


CHAPMAN & HALL/CRC COMPUTER and INFORMATION SCIENCE SERIES
Handbook of Bioinspired Algorithms and Applications
Edited by
Stephan Olariu
Albert Y. Zomaya
Chapman & Hall/CRC

From the many metaheuristics currently available, one particular class has become very popular in the last 30 years: **bio-inspired metaheuristics**.

Bio-inspired metaheuristics use rules that are inspired on some biological metaphore (e.g., in the case of evolutionary algorithms, the inspiration is Darwin's survival of the fittest principle). Most bio-inspired metheuristics are stochastic search techniques (e.g., evolutionary algorithms, particle swarm optimization, ant colony optimization, etc.).

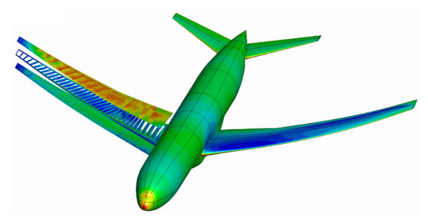
Current State


FUNDAMENTALS OF NATURAL COMPUTING
Basic Concepts, Algorithms, and Applications
Leandro Nunes de Castro
Chapman & Hall/CRC

Most bio-inspired metaheuristics operate on a set of solutions (normally called *population*) at each iteration. A clever use of this population in multi-objective optimization, allows the generation of several elements of the Pareto optimal set in a single algorithmic execution.

Also, bio-inspired metaheuristics require little information about the domain (e.g., they don't require derivatives) and are less susceptible to the shape or continuity of the Pareto front.

Current State



In spite of their several advantages, bio-inspired metaheuristics also have some disadvantages. One of them is that they cannot guarantee convergence to the true Pareto front of a problem in most practical cases. Another one (which is more relevant in practical applications) is that their computational cost is normally significantly higher than that of mathematical programming techniques. This is due to their stochastic nature, which requires sampling several solutions to find an appropriate search direction. This may be unaffordable in some applications (e.g., in aeronautical engineering).