Ignacy Kaliszewski

Janusz Miroforidis

MULTIPLE CRITERIA DECISION MAKING: FROM EXACT TO HEURISTIC OPTIMIZATION

Abstract

We propose to derive *assessments of outcomes* to Multiple Criteria Decision Making problems, instead of just outcomes, and carry decision making processes with the former. In contrast to earlier works in that direction, which to calculate assessments have made use of subsets of the *efficient set* (*shells*), here we provide formulas for calculations of assessments based on the use of *upper and lower approximations* (*upper and lower shells*) of the efficient set, derived by *evolutionary optimization*. Hence, by replacing shells, which are to be in general derived via optimization, with pairs of upper and lower shells, the need of exact optimization methods can be eliminated from Multiple Criteria Decision Making.

Keywords

Multiple criteria decision making, evolutionary optimization, parametric outcome assessments.

Introduction

Decision making, whether in economic or social domain, calls for multi aspect deliberations. The field of *Multiple Criteria Decision Making* (where "criteria" stands for "aspects") provides methodologies and supporting tools to cope with decision problems.

For a class of "complex" decision problems, where because of scale, bulk of data, and/or intricate framing a formal model is requested, efficient variants, and among them the most preferred variant (the decision), can be derived with the help of exact optimization methods. This in turn requires that the model has to be tied to an exact optimization package, which certainly precludes popular, lay and widespread use on Multiple Criteria Decision Making (MCDM) methods.

In a quest for simpler MCDM tools than those offered by now, it was proposed in Kaliszewski [3], [4] that the decision maker (DM) instead of evaluating exact *outcomes* (i.e. vectors of variant criteria values) would evaluate *assessments of outcomes*, provided with sufficient (and controlled) accuracy. Once the *most preferred outcome assessment* is derived, the closest (in a sense) variant is determined.

For an *efficient outcome* (i.e. outcome of an efficient variant) assessment calculations a subset of efficient variants (a *shell*) has to be known. As a shell can be derived (by exact optimization methods) prior to starting the decision process, replacing outcomes by their assessment relives MCDM from a *direct* dependence on exact optimization methods and packages.

In Miroforidis [7] it has been recently proposed to replace shells by somewhat weaker constructs, namely *lower shells* and *upper shells* and formulas for assessments of *weakly efficient outcomes* (i.e. outcomes of *weakly efficient variants*) have been derived. As lower and upper shells can be derived by *evolutionary optimization*, replacing shells by pairs of lower and upper shells leads to replacement of exact optimization methods (required to derive shells) by their evolutionary (*bona fide* heuristic) counterparts. This, in consequence, eliminates from MCDM the need of exact optimization methods and packages completely.

In this paper, on the base of the concept of lower and upper shells, we derive formulas for assessments of *properly efficient outcomes* (i.e. outcomes of *properly efficient variants*). These bounds subsume as a special case formulas derived in Miroforidis [7].

The outline of the paper is as follows. In Section 1 we provide basic definitions and notation. In Section 2 we derive formulas for assessments of properly efficient outcomes using lower and upper shells. Final Section concludes.

1. Definitions and notation

Let x denote a (decision) variant, $\mathcal{X}$ a variant space, X_0 a set of *feasible variants*, $X_0 \subseteq \mathcal{X}$. Then the underlying model for MCDM is formulated as:

$$\begin{gathered} \text{“}max\text{”}\, f(x) \\ x \in X_0 \subseteq \mathcal{X}, \end{gathered} \tag{1}$$

where $f : \mathcal{X} \to \mathcal{R}^k$, $f = (f_1,...,f_k)$, $f_i : X \to R$, are objective (criteria) functions, $i = 1,...,k$, $k \geq 2$; "*max*" denotes the operator of deriving all efficient variants in X_0 according to the definition of efficiency given below.

In MCDM to compare feasible variants x one makes use of their outcomes $f(x)$. Relations between outcomes in *outcome space* $\mathcal{R}^k$ induce relations between variants in variant space $\mathcal{X}$.

Below we make use of the following notation: $y = f(x)$.

Element $\bar{t}$ of T, $T \subseteq \mathcal{R}^k$, is:

– *efficient in* T, if $t_i \geq \bar{t}_i$, $i = 1,...,k$, $t \in T$, implies $t = \bar{t}$,
– *weakly efficient in* T, if there is no $t \in T$, such that $t_i > \bar{t}_i$, $i = 1,...,k$,
– *properly efficient in* T [1], if it is efficient and there exists a finite number $M > 0$ such that for each i we have

$$\frac{t_i - \bar{t}_i}{\bar{t}_j - t_j} \leq M$$

for some j such that $t_j < \bar{t}_j$, whenever $t \in T$ and $t_i > \bar{t}_i$.

Variant $\bar{x} \in A \subseteq \mathcal{X}$ is called *efficient* (*weakly efficient*, *properly efficient*) in A if $\bar{y} = f(\bar{x})$ is efficient (weakly efficient, properly efficient) in $f(A)$. We denote the set of efficient variants of X_0 by N.

We define on $\mathcal{X}$ the *dominance relation* $\succ$,

$$x' \succ x \Leftrightarrow f(x') >> f(x),$$

where $>>$ denotes $f_i(x') \geq f_i(x)$, $i = 1,...,k$, and $f_i(x') > f_i(x)$ for at least one i. If $x' \succ x$, then we say that x is *dominated* by x' and x' is *dominating* x.

The following definitions of lower and upper shells come from [7].

Lower shell is a finite nonempty set $S_L \subseteq X_0$, elements of which satisfy

$$\forall_{x \in S_L} \neg \exists_{x' \in S_L} x' \succ x. \tag{2}$$

By condition (2) all elements of shell S_L are efficient in S_L.

For a given lower shell S_L we define *nadir point* $y^{nad}(S_L)$ as

$$y_i^{nad}(S_L) = \min_{x \in S_L} f_i(x), \ i = 1,...,k.$$

Upper shell is a finite nonempty set $S_U \subseteq \mathcal{X} \setminus X_0$, elements of which satisfy*

$$\forall_{x \in S_U} \neg \exists_{x' \in S_U} \, x \succ x' \,, \tag{3}$$

$$\forall_{x \in S_U} \neg \exists_{x' \in N} \, x' \succ x \,, \tag{4}$$

$$\forall_{x \in S_U} \; f_i(x) > y_i^{nad}(S_L), \; i = 1,\dots,k \,. \tag{5}$$

Below we make use of a selected element of outcome space $\mathcal{R}^k$, denoted y^*, defined as

$$y_i^* = \hat{y}_i + \varepsilon, \quad i = 1,\dots,k \,,$$

where ε is any positive number and $\hat{y}$ is the *utopian element* of $\mathcal{R}^k$, calculated as

$$\hat{y}_i = \max_{y \in f(X_0) \cup f(S_U)} y_i \,, \; i = 1,\dots,k \,,$$

and we assume that all these maxima exist.

We assume that all efficient outcomes are ρ-properly efficient, i.e. they can be derived by solving the optimization problem

$$\min_{y \in f(X_0)} \max_i \lambda_i ((y_i^* - y_i) + \rho e^k (y^* - y)) \,, \tag{6}$$

where $\lambda_i > 0, \; i = 1,\dots,k$, and $\rho > 0$ (cf. e.g. [8], [6], [2], [4]).

By condition (3) all elements of upper shell S_U are efficient in S_U. We also assume that they all are ρ-properly efficient in S_U, i.e. they can be derived by solving the optimization problem

$$\min_{y \in f(S_U)} \max_i \lambda_i ((y_i^* - y_i) + \rho e^k (y^* - y)) \,, \tag{7}$$

where $\lambda_i > 0, \; i = 1,\dots,k$, and $\rho > 0$ has the same value as for ρ-properly efficient outcomes (elements of $f(X_0)$) defined above.

---

* In [7] condition (5) has the form $f(x) >> y^{nad}(S_L)$. We have had to strengthen this condition to deal with proper efficiency in formula (11) below [5].

2. Parametric bounds on outcomes

An outcome which is not derived explicitly (i.e. it is not an *explicit outcome*) but is only designated by selecting vector λ for the purpose to solve the optimization problem (6), is called an *implicit outcome*.

We use lower and upper shells of N to calculate parametric bounds on implicit outcomes, with weights λ as parameters.

We are aiming at the following. Suppose vector of weights λ is given. Let $y(\lambda)$ denote an implicit properly efficient outcome of $f(X_0)$, which would be derived if optimization problem (6) were solved with that λ. Let $L(y(\lambda))$ and $U(y(\lambda))$ be vectors of lower and upper bounds on components of $y(\lambda)$, respectively. These bounds form an *assessment* $[y(\lambda)]$ of $y(\lambda)$,

$$[y(\lambda)] = \{L(y(\lambda)), U(y(\lambda))\}.$$

To simplify notation we put $L(y(\lambda)) = L(\lambda)$ and $U(y(\lambda)) = U(\lambda)$.

To calculate bounds (assessments) one needs to know a pair of lower and upper shells. As can be seen below, computational costs to calculate such bounds are negligible as compared to derivation of efficient outcomes by exact optimization methods.

Formulas we show may at the first glance look complicated, but in fact they consist of no more than operations of adding and taking maxima over finite sets of numbers.

Proofs of formulas can be found in [5].

Let $\overline{L}_i$ and $\overline{U}_i$ be such that for each $y \in f(X_0)$ the following holds

$$\overline{L}_i \le y_i \le \overline{U}_i, \quad i = 1,...,k. \tag{8}$$

2.1. Lower Bounds

Below we give a formula to calculate lower bounds on outcome components. For a given vector of weights λ, $\lambda_i > 0$, $i = 1,...,k$, let $y(\lambda)$ be an implicit properly efficient outcome, which would be derived if optimization problem (6) were solved with that λ.

For a given lower shell S_L the lower bounding formula is

$$y_i(\lambda) \geq L_i(S_L, \lambda)$$
$$\max\{y_i^* - (\lambda_i(1+\rho))^{-1} \max_{y \in f(S_L)}[\max_j \lambda_j((y_j^* - y_j) \tag{9}$$
$$+ \rho e^k(y^* - y))] + \frac{\rho}{1+\rho} \sum_{j \neq i}^{k} (y_j^* - U_j(\lambda)), \overline{L}_i\},\ i = 1,\ldots,k\ ,$$

where $U_j(\lambda)$ are such that $y_j \leq U_j(\lambda),\ j = 1,\ldots,k,\ j \neq i$. One possible selection of $U_j(\lambda)$ is $\overline{U}_j,\ j = 1,\ldots,k,\ j \neq i$, where $\overline{U}_j$ is defined by (8). Here we extend notation $L(\lambda)$ to $L(S_L, \lambda)$ to stress dependence of lower bounds on lower shells S_L.

Putting $\rho = 0$ in (9) we get the lower bounding formula for weakly efficient outcomes, derived in [7].

2.2. Upper Bounds

Below we give a formula to calculate upper bounds on outcome components. For a given vector of weights $\lambda,\ \lambda_i > 0,\ i = 1,\ldots,k$, let $y(\lambda)$ be, as previously, an implicit properly efficient outcome, which would be derived if optimization problem (6) were solved with that λ.

Suppose that an upper shell S_U is given. To calculate upper bounds on components of efficient outcomes, for each element $\bar{y}$ of $f(S_U)$ we have to know vector $\lambda,\ \lambda_i > 0,\ i = 1,\ldots,k$, such that $\bar{y}$ solves optimization problem (6) on $f(S_U)$ for that λ. To stress the association between λ and $\bar{y}$ we denote $\lambda = \lambda(\bar{y})$.

It is easy to show that any ρ-properly efficient element $\bar{y}$ of $f(S_U)$ solves optimization problem (6) on $f(S_U)$ with $\lambda = \overline{\lambda}(\bar{y})$, where

$$\overline{\lambda}_i(\bar{y}) = ((y_i^* - \bar{y}_i) + \rho e^k(y^* - \bar{y}))^{-1},\ i = 1,\ldots,k\ . \tag{10}$$

Indeed, for $\overline{\lambda}_i(\bar{y}),\ i = 1,\ldots,k$, we clearly have, by the definition of y^*,

$$\overline{\lambda}_i(\bar{y}) > 0\ ,$$

and

$$\overline{\lambda}_i(\bar{y})((y_i^* - \bar{y}_i) + \rho e^k(y^* - \bar{y})) = 1\ .$$

Since all elements of S_U are ρ-properly efficient in S_U, $\bar{y}$ is a solution of optimization problem (7) for some $\lambda_i > 0,\ i = 1,...,k$, i.e. for all $y \in S_U$

$$\max_i \lambda_i ((y_i^* - y_i) + \rho\, e^k (y^* - y)) \geq \max_i \lambda_i ((y_i^* - \bar{y}_i) + \rho\, e^k (y^* - \bar{y})) .$$

Hence, for some j

$$\lambda_j ((y_i^* - y_j) + \rho e^k (y^* - y)) \geq \lambda_j ((y_j^* - \bar{y}_j) + \rho e^k (y^* - \bar{y})) ,$$

and

$$(y_j^* - y_j) + \rho e^k (y^* - y) \geq (y_j^* - \bar{y}_j) + \rho e^k (y^* - \bar{y}) .$$

Thus,

$$\bar{\lambda}_j (\bar{y})((y_j^* - y_j) + \rho\, e^k (y^* - y)) \geq \bar{\lambda}_j (\bar{y})((y_j^* - \bar{y}_j) + \rho\, e^k (y^* - \bar{y}))$$

In consequence, for each $y \in S_U$ we have

$$\max_i \bar{\lambda}_i (\bar{y})((y_i^* - y_i) + \rho e^k (y^* - y)) \geq \max_i \bar{\lambda}_i (\bar{y})((y_i^* - \bar{y}_i) + \rho e^k (y^* - \bar{y})) = 1.$$

Hence, $\bar{y}$ is a solution of (6) on $f(S_U)$ with $\lambda = \bar{\lambda}_i(\bar{y})$.

For a given upper shell S_U the upper bounding formula is

$$y_i(\lambda) \leq U_i(S_U, \lambda)$$

$$\min\{\min_{\bar{y}\in f(s_U)}[\min_{l\in I(\lambda)}\{y_l^* + \frac{\rho}{1+\rho}\sum\nolimits_{j\neq l}^{k} y_j^* - \bar{\lambda}_l(\bar{y})^{-1}(1+\rho)^{-1} \tag{11}$$

$$-\frac{\rho}{1+\rho}\sum\nolimits_{j\neq i}^{k} L_j(\lambda)\}]\},\bar{U}_i\},\ i = 1,...,k ,$$

where $I(\lambda)$ is a subset of indices $\{1,2,...,k\}$ such that $l \in I(\lambda)$ if $t^l = \min\{t^1,...,t^k\}$, where

$$t^i = ((\tau_i + \rho\, e^k \tau)\bar{\lambda}_i(\bar{y}))^{-1} ,$$

τ is defined by formula

$$\tau = y^* - \bar{y} , \tag{12}$$

and $L_j(\lambda)$ are such that $y_j(\lambda) \geq L_j(\lambda),\ = 1,...,k,\ j \neq l$. One possible selection of $L_j(\lambda)$ is $\bar{L}_j,\ j = 1,...,k,\ j \neq i$, where $\bar{L}_j$ is defined by (8). Here we extend notation $U(\lambda)$ to $U(S_U, \lambda)$ to stress dependence of lower bounds on upper shells S_U.

Putting $\rho = 0$ in (11) we get the upper bounding formula for weakly efficient outcomes, derived in [7].

Concluding remarks and directions for further research

The obvious advantage of replacing shells, which are to be derived by solving optimization problems, with their lower and upper counterparts S_L and S_U, which can be derived, as in [7], by evolutionary computations, would be complete elimination of exact optimization from MCDM.

The open question is the quality (tightness) of assessments when $S_L \not\subset N,\ S_U \not\subset N$. This question imposes itself on the same question with respect to assessments derived with $S_L = S_U \subset N$, addressed in Kaliszewski [3], [4]. However, if S_L and S_U derived by evolutionary computations are "close" to N there should be no significant deterioration in the quality of assessments. Indeed, preliminary experiments with some test problems reported in [7], confirm such expectations.

To make condition (4) of the definition of upper shells operational one has to replace N by S_L, for obviously N is not known (for details cf. [7]), but with such a replacement formulas (9) and (11) remain valid (though in principle they become weaker).

References

[1] Geoffrion A.M.: *Proper Efficiency in Vector Optimization*. J. Appl. Math. Anal., 1968, 22, pp. 618-630.

[2] Ehrgott M.: *Multicriteria Optimization*. Springer, 2005.

[3] Kaliszewski I.: *Out of the Mist – Towards Decision-maker-friendly Multiple Criteria Decision Making Support*. Eur. J. Oper. Res., 2004, 158, pp. 293-307.

[4] Kaliszewski I.: *Soft Computing for Complex Multiple Criteria Decision Making*. Springer, 2006.

[5] Kaliszewski I.: Multiple Criteria Decision Making: Outcome Assessments with Lower and Upper Shells. Systems Research Institute Report, RB/9/2008.

[6] Miettinen K.M.: *Nonlinear Multiobjective Optimization*. Kluwer, Boston 1999.

[7] Miroforidis J.: *Decision Making Aid for Operational Management of Department Stores with Multiple Criteria Optimization and Soft Computing*. PhD Thesis, Systems Research Institute, Warsaw, 2010.

[8] Wierzbicki A.P.: *Reference Point Approaches*. In: *Multicriteria Decision Making – Advances*. In: *MCDM: Models, Algorithms, Theory and Applications*. Eds. T. Gal, Th. Stewart, Th. Hanne. Kluwer 1999.