Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Multicriteria Decision Making (MCDM 2007)

An interactive approach to integer linear vector optimization problems using enumerative cuts

W. Habenicht
University of Hohenheim (510A)
D-70593 Stuttgart
Germany

***Abstract*- We present a conceptual framework of an interactive method for solving integer linear vector optimization problems. The method is based on an enumerative cut approach. It combines cutting planes with enumerative parts. In this method the user can perform a structured searching process in the non-dominated set.**

I. INTRODUCTION

Interactive approaches in vector optimization, generally, perform a structured search in the non-dominated set. In this paper we will show how this can be done for integer linear vector optimization problems. There are some approaches combining traditional approaches from classical integer programming. In an early work, Lee and Morris [8] combined dual cutting planes, branch-and-bound techniques and implicit enumeration in a goal programming approach. Ramesch, Zionts and Karwan [9] integrated the Zionts/Wallenius method into a branch and bound concept. Karavainova et al. [7] apply the Pareto Race to integer problems. Alves and Climaco [1] combine (classical) cutting planes and branch-and-bound routines. The method, we present here, uses enumerative cuts combining enumerative elements with a general type of cutting planes.

Enumerative cuts were introduced to solve ordinary integer linear optimization problems by Burdet [2], [3] and developed by Habenicht [4]. Up to our knowledge, this is the first time that this approach is used in vector optimization.

The paper is organized as follows. In chapter 2 integer linear vector optimization problems are introduced. The concept of enumerative cuts is presented in chapter 3. In chapter 4 we develop our interactive procedure. Chapter 5 gives some final remarks, open problems, and some directions of further research.

II. THE INTEGER LINEAR VECTOR OPTIMIZATION PROBLEM

This paper deals with the integer linear vector optimization problem (**ILVOP**):

$$\text{“Minimize” } \{\mathbf{y} = \mathbf{Cx} \mid \mathbf{x} \in \mathbf{X}\} \tag{1}$$

$$\mathbf{X} := \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{Ax} \le \mathbf{b}\} \tag{2}$$

Here, $\mathbb{Z}$ denotes the set of integers, $\mathbf{y} \in \mathbb{R}^k$ the (real valued) vector of outcomes, $\mathbf{C} \in \mathbb{R}^{kxn}$ the matrix of coefficients of the k criteria, $\mathbf{A} \in \mathbb{R}^{mxn}$ the coefficient matrix of the constraint set, and $\mathbf{b} \in \mathbb{R}^m$ the right hand side vector.

Let $\mathbf{Y} := \{\mathbf{y} = \mathbf{Cx} \mid \mathbf{x} \in \mathbf{X}\}$, then we can formulate **ILVOP** in an “outcome oriented “ way as:

$$\text{“Minimize” } \{\mathbf{y} \in \mathbf{Y}\} \tag{3}$$

The two formulations of **ILVOP** can be interpreted as two different views on the problem. (1), (2) stands for the traditional interpretation of a (vector) optimization problem in decision space: “Find some feasible solution **x** in decision space, according to the ‘minimize’- operator.” Whereas, (3) stands for a view on the problem in outcome space: “Find some feasible outcome **y,** according to the ‘minimize’-operator”.

Here, the “minimize”-operator is defined in the ordinary sense of pareto-optimality, i.e. find some $\mathbf{y} \in \mathbf{Y}^*$, with:

$$\mathbf{Y}^* := \{\mathbf{y} \in \mathbf{Y} \mid \mathbf{y}' \le \mathbf{y} \Rightarrow \mathbf{y}' = \mathbf{y}\} \tag{4}$$

$\mathbf{Y}^*$ is called the non-dominated set and

$$\mathbf{X}^* := \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{Cx} \in \mathbf{Y}^*\} \tag{5}$$

the efficient set. Throughout this paper we will use the following example:

$$\text{"Minimize"} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} -2x_1 + x_2 + 22 \\ x_1 - 2x_2 + 27 \end{pmatrix}$$

subject to:

$$\begin{aligned} x_1 + 5x_2 &\ge 20 \quad \text{(I)} \\ 13x_1 + x_2 &\le 122 \quad \text{(II)} \\ x_1 + 4x_2 &\le 47 \quad \text{(III)} \\ 4x_1 + x_2 &\le 43 \quad \text{(IV)} \\ x_1, x_2 &\ge 0, \text{integer} \end{aligned}$$

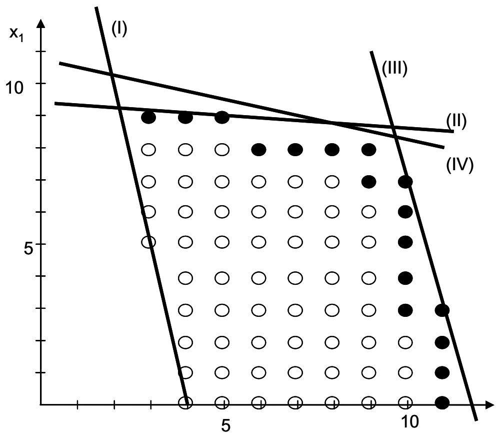

Fig. 1: Feasible set in decision space

III. ENUMERATIVE CUTS

Enumerative cuts are based on the concept of intersection cuts, also called convexity cuts. Like any cutting plane approach in integer linear optimization this concept uses LP-relaxations.

Let $\mathbf{X}^R := \{\mathbf{z}\in\mathbb{R}^{n+m} \mid (\mathbf{A},\mathbf{I})\mathbf{z}=\mathbf{b}, \mathbf{z}\geq\mathbf{0}\}$ be the feasible set of the LP-relaxation of **ILVOP**, with **I** the (mxm)-identity matrix, $\mathbf{z}=(\mathbf{x},\mathbf{s})$, and $\mathbf{s} := \mathbf{b}-\mathbf{A}\mathbf{x}\in\mathbb{R}^m$ the vector of slack variables. Moreover, let $\mathbf{BV} \subset \{1,\ldots,n+m\}$ be the index set of the basic variables ($|\mathbf{BV}| = m$), and $\mathbf{NBV} := \{1,\ldots,n+m\}\backslash\mathbf{BV}$ the index set of the non-basic variables of some basic solution of the LP-Relaxation. The non-negativity constraints of the non-basic variables define the cone $\mathbf{K}^B := \{\mathbf{z}\in\mathbb{R}^{\mathbf{n+m}} \mid (\mathbf{A},\mathbf{I})\mathbf{z}=\mathbf{b}, z_j\geq 0\ \forall j\in\mathbf{NBV}\}$, originating at the basic solution. $\mathbf{K}^B$ can also be formulated as $\mathbf{K}^B = \{\mathbf{z}\in\mathbb{R}^{n+m} \mid z_i = d_{io} - \Sigma_{j\in\mathbf{NBV}}\, d_{ij}z_j\ \forall j\in\mathbf{NBV}\}$, with d_{io} the right hand side coefficients and d_{ij} the other coefficients of the simplex tableau. The edges of the cone $\mathbf{K}^B$ are given by the columns of the simplex tableau, and the basic solution is given by $z_i^* = d_{io}\ \forall i\in\mathbf{BV}$, $z_i^* = 0$ else. The corresponding (LP-relaxed) solution of **IVLOP** is given by $x_i^* = z_i^*\ \forall i\leq n$.

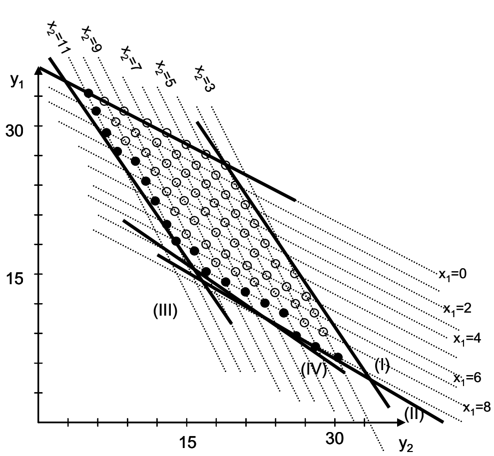

Fig. 2: Feasible Set in Outcome Space

Let $\mathbf{C} \subset \mathbb{R}^n$ be a closed and convex set that contains $\mathbf{x}^*$ in its interior, then we call **C** a **cut generating set**. For $j\in\mathbf{NBV}$, let $\lambda_j^* := \sup\{\lambda_j \mid \mathbf{x}^*-\mathbf{d}^j\lambda_j \in \mathbf{C}\}$. If $\lambda_j^*<\infty$, then $\mathbf{x}^j := \mathbf{x}^* - \mathbf{d}^j\lambda_j^*$ is an intersection point of the edge of $\mathbf{K}^B$ generated by the non-basic variable z_j with the boundary of **C** ($\mathbf{d}^j$ denotes the column of the simplex tableau for non-basic variable z_j). The half space $\mathbf{S}(\mathbf{C}) := \{\mathbf{x} = \mathbf{x}^*-\mathbf{d}^j z_j \mid \Sigma_{j\in\mathbf{NBV}}\, 1/\lambda^*\, z_j \geq 1\}$ is called the **cut generated by C**.

Obviously, the intersection points $\mathbf{x}^j$ lie on the boundary of $\mathbf{S}(\mathbf{C})$ and the following theorem holds [4]:

Theorem: Let **C** be a cut generating set and $\mathbf{S}(\mathbf{C})$ the generated cut, then

$$\mathbf{K}^B\backslash\ \mathbf{S}(\mathbf{C})\subset \mathrm{int}(\mathbf{C}) \quad (4)$$

In (4), int(**C**) denotes the interior of **C**. Classical cutting plane algorithms use this result by choosing cut generating sets that do not contain any integer point in its interior. In fact, the classical Gomory cuts are special cases of such intersection cuts.

If we choose cut generating sets that have integer points in its interior we can get deeper cuts, but these integer points may be cut off. Hence, we have to enumerate the integer points in the interior of the cut generating set. In our approach we use hypercubes of the form $\mathbf{Q} := \{\mathbf{x}\in\mathbb{R}^n \mid \mathbf{u} \leq \mathbf{x} \leq \mathbf{o}\}$ with $\mathbf{u}, \mathbf{o} \in\mathbb{Z}^{\mathbf{n}}$ as cut generating sets.

In Figure 3 we show for our example the cut generating set $\mathbf{C} = \{5\leq x_1\leq 10, 6\leq x_2\leq 12\}$ together with the generated cut $\mathbf{S}(\mathbf{C})$.

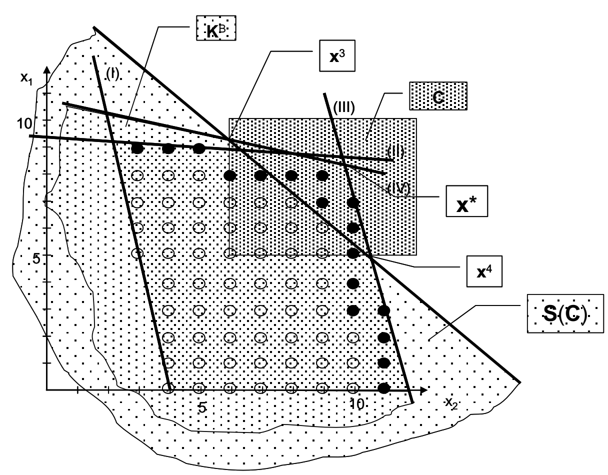

Fig. 3: Enumerative Cut

IV. THE INTERACTIVE APPROACH

We start with a rudimentary description of our approach:

- Step 0: Let $\mathbf{x}^o$ be the best solution of **ILVOP** known so far. (If no solution is known, let $\mathbf{x}^o$ be some dummy solution with a sufficiently bad evaluation).
- Step 1: Transform **ILVOP** to a single criterion problem by some positive weighting of the criteria and solve the LP-Relaxation
- Step 2: The DM chooses a cut generating set in outcome space.
- Step 3: Let the DM choose the best feasible (integer) solution $\mathbf{x}'$ in int($\mathbf{C}$). If the DM prefers $\mathbf{x}'$ **to** $\mathbf{x}^o$, let $\mathbf{x}^o$:=$\mathbf{x}'$.
- Step 4: Introduce S($\mathbf{C}$) and present the vertices of the facet produced by the introduction of S($\mathbf{C}$) into the Relaxation to the DM. Ask for the best vertex and denote it by $\mathbf{x}^+$.
- Step 5: If the DM prefers $\mathbf{x}^*$ to $\mathbf{x}^+$, stop. Otherwise, choose as the new basic solution of the LP-Relaxation and go to step 2.

Now, we explain the steps in some detail. In step 1 we may ask the decision maker (DM) for some (positive) weighting of the criteria, and we transform **ILVOP** to a single criterion problem by building the weighted sum of the criteria. If no weights are available from the DM, we may use equal weights. Then we solve the LP-relaxation. In Figure 4 the result is shown in outcome space.

The outcome vector $\mathbf{y}^*$ of the optimal solution $\mathbf{x}^*$ of the relaxation is shown to the DM, and he is asked for the region in outcome space around $\mathbf{y}^*$ where to search for better solutions. Precisely, he is asked for lower and upper bounds on all criteria. In Figure 4 we assume that the DM has defined the bounds $12 \le y_1 \le 30$ and $9 \le y_2 \le 21$.

This defines a cut generating set $\mathbf{C}^*$ in outcome space. Since the enumeration process takes place in decision space, Figure 5 shows $\mathbf{C}^*$ in decision space. Obviously, $\mathbf{C}^*$ is not very well suited for the enumeration in decision space. Moreover, the only points that are cut off by S($\mathbf{C}^*$) lie in the simplex $\mathbf{S}^* = \mathbf{K}^B \setminus \mathbf{S}(\mathbf{C}^*)$. Hence, we choose as cut generating

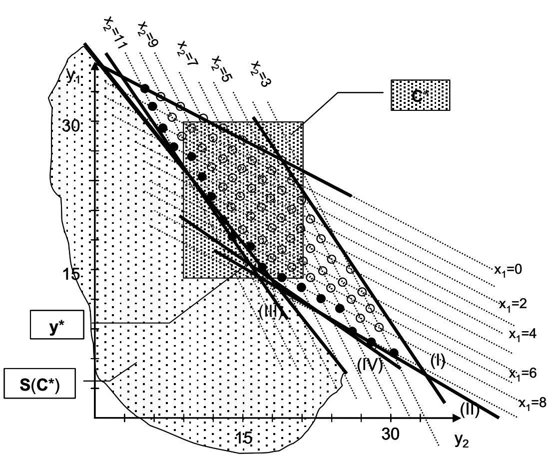

Fig. 4: Starting solution in outcome space

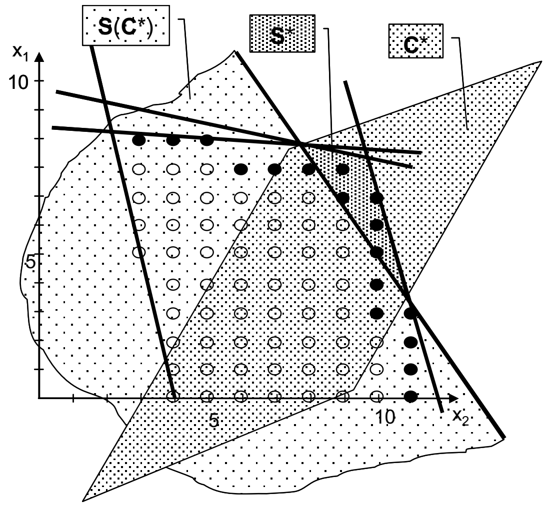

Fig. 5: $\mathbf{C}^*$ in decision space

set $\mathbf{C}$ the smallest hypercube in decision space with integral bounds that contains $\mathbf{S}^*$. This is shown in Figure 6.

It should be mentioned that S($\mathbf{C}$) is stronger than S($\mathbf{C}^*$). A crucial point in the choice of the cut generating set lies in the fact that the DM does not know how many integer points are contained in the interior of $\mathbf{C}$ when he chooses $\mathbf{C}^*$. If he chooses it too small, there may be no integer points in the interior of $\mathbf{C}$. If he chooses $\mathbf{C}^*$ too large, there are too many points to be enumerated. This is a crucial point because the DM has to choose the best one out of them.

This problem can be solved by generalizing S($\mathbf{C}^*$) to $\mathbf{S}_\alpha(\mathbf{C}^*) := \{\mathbf{x} = \mathbf{x}^* - \mathbf{d}^j z_j \mid \Sigma_{j \in \mathbf{NBV}} 1/\lambda^* z_j \ge \alpha\}$. In this way, we can control the depth of the cut and, hence, the extension of $\mathbf{C}$ by the parameter α. We are choosing the maximal value of α such that the number of integer points in the interior of C is not greater than some prespecified value.

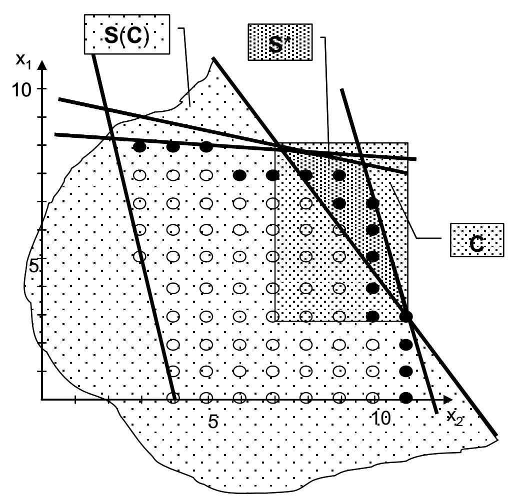

Fig. 6: Cut generating set in decision space

The enumeration of the integer points in the interior of **C** can be done in different ways. One approach is the **O**utcome based **N**eighbourhood **S**earch (**ONS**) developed by the author [5]. In this approach generalized quad trees are used to identify the non-dominated points.

In step 4 we introduce the cut **S**(**C**) and look at the vertices of the created facet. If the best solution found so far is preferred to all the vertices of the facet, we have reached some kind of local optimum und we propose to stop. If the preferences meet some special properties (for instance, pseudo concavity of the utility function) we conjecture that $\mathbf{x}°$ is globally optimal. If $\mathbf{x}°$ is not preferred to all vertices, we choose the best vertex as a new basic solution of the LP-relaxation and perform another iteration.

V. CONCLUDING REMARKS

The aim of this paper was to sketch out the applicability of the enumerative cutting plane approach to integer linear vector optimization problems. We presented a concept that has to be specified with respect to many aspects. One crucial point is the size of the cut generating sets. This point is highly correlated to the problem of finding good enumeration schemes. We are looking for enumeration schemes where the decision maker may choose the best one out of a great number of efficient solutions by only performing a moderate number of pair wise comparisons. One promising approach is the **ONS**-approach. Another point is the enumeration of the vertices of the facet generated by the cut. Here, we will rely on the work of Isermann [6] and others.

REFERENCES

[1] Alves, M., J. and J. Climaco A note on a decision support system for multiobjective integer and mixed-integer programming problems, European Journal of Operational Research 155 (2004), 258-265

[2] Burdet, C. A. Enumerative Inequalities in Integer Programming, Mathematical Programming 2 (1972), 32-64

[3] Burdet, C. A. Enumerative Cuts I, Operations Research 21 (1973), 61-89

[4] Habenicht, W. Ausgewählte Enumerationsstrategien zur Lösung ganzzahliger linearer Optimierungsprobleme, Anton Hain 1976 (in German)

[5] Habenicht, W. Outcome-Based Neighbourhood Search (ONS), in Bouyssou, D. et al (ed.) Aiding Decisions with Multiple Criteria, Kluwer 2002, 527-541.

[6] Isermann, H. The enumeration of the set of all efficient solutions for a linear multiple objective program, Operations Research Quarterly 28 (1977), 711-725

[7] Karainova, J., P. Korhonen, S. Narula, J. Wallenius and V. Vassilev A reference direction approach to multiple objective integer linear Programming, European Journal of Operational Research 81 (1995), 176-187

[8] Lee, S.,M. and R. L. Morris Integr Goal Programming Methods, TIMS Studies in Management Science 6 (1977), 273-289

[9] Ramesh, R., S. Zionts, M.,H. Karwan A class of practical interactive branch and bound algorithms for multicriteria integer programming, European Journal of Operational Research 26 (1986), 161-172