The 7th International Symposium on Operations Research and Its Applications (ISORA'08)
Lijiang, China, October 31–Novemver 3, 2008

Disjoint Multi-linear Optimization in Imprecise Decision Analysis*

Xiaosong Ding† Jihong Zhang

School of International Business, Beijing Foreign Studies University
100089, Beijing, People's Republic of China

Abstract This paper presents a global optimization algorithm for handling disjoint multi-linear optimization programs arising in computational decision analysis when imprecise information prevails. We make use of an existing cutting plane method, namely, the polar cut, and the disjoint structural property to develop a new approach that is different from the traditional class of branch and bound methods.

1 Introduction

Most classical decision analysis approaches consist of a set of straightforward decision rules applied to precise estimates of probabilities and/or utilities no matter how unsure a decision maker is of his estimates. The requirement for numerically precise data has been considered unrealistic by an increasing number of researchers and decision-makers. In attempting to address real-life decision problems, a representation of imprecise information seems important. Despite a number of ambitious theories suggested, comparatively few have addressed the issues about computationally feasible algorithms for evaluating the decision structure [4, 5, 6, 7, 11, 14].

To evaluate a decision situation, many approaches are built upon the principle of maximizing the expected utility (PMEU) because it has been shown that both these and some other approaches have performances which are at best equal to that of the PMEU and at worst are significantly poorer [9]. Therefore, to improve PMEU, it should be supplemented with other qualitative rules rather than engaging in further modifications in pursuit of a reasonable rule [4]. However, using PMEU may lead to disjoint multi-linear programming (DMLP) problems with special structural properties when imprecise information prevails. In general, DMLP is computationally hard to solve. But for an interactive decision making process, it is necessary to develop an efficient computational procedure for solving such programs.

This paper intends to discuss the computational aspects of DMLP arising in the evaluation of an imprecise decision model by using PMEU and extend the work in [7]. The following section presents an imprecise decision framework that can be transformed into a

*partially supported by Natural Science Foundation of China under Grant #70771013 and Program for NCET-07-0105
†corresponding author. Email: dingxiaosong@bfsu.edu.cn

DMLP program. The third and fourth sections discuss the corresponding local and global optimization strategies for solving DMLP, respectively. The final section concludes this paper.

2 An Imprecise Decision Model

For simplicity, we will not go into relevant details of the representation and evaluation of a general decision situation, interested readers are encouraged to pursue [4] and the references therein.

Suppose we have an alternative represented as a decision tree

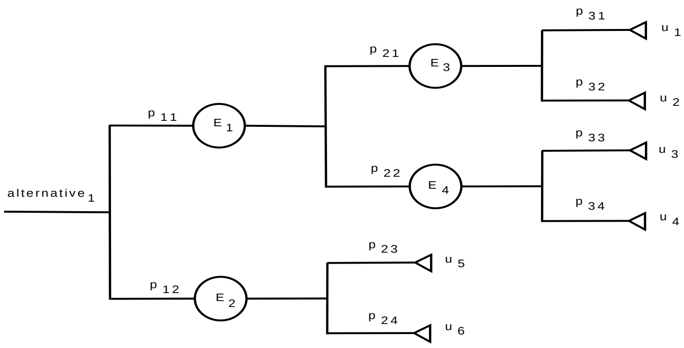

Figure 1: A Multi-linear Decision Model

To calculate the expected utility of alternative$_1$ with point values seems rather straightforward. It can be derived as

$$\begin{aligned}&p_{11}(p_{21}(p_{31}u_1+p_{32}u_2)+p_{22}(p_{33}u_3+p_{34}u_4))+p_{12}(p_{23}u_5+p_{24}u_6)\\&=p_{11}p_{21}p_{31}u_1+p_{11}p_{21}p_{32}u_2+p_{11}p_{22}p_{33}u_3+p_{11}p_{22}p_{34}u_4\\&+p_{12}p_{23}u_5+p_{12}p_{24}u_6\end{aligned}\qquad(1)$$

As imprecise information such as interval statements prevails [4, 5], we have various types of imprecise statements within each level that can be translated into linear constraints. It should be noted that currently we only allow constraints from the same level rather than different levels. For example, the interval statement $0.2 \le p_{21}+p_{23} \le 0.3$ is considered proper, while the interval statement $0.2 \le p_{11}+p_{23} \le 0.3$ is considered improper since p_{11} and p_{23} are from level 1 and level 2, respectively, and this will destroy the disjoint structural property.

To evaluate alternative$_1$ by using PMEU with imprecise information, we are supposed to compute two extreme values, i.e., maximum and minimum, of (1) subject to disjoint linear constraints in order to receive a range of expected utility for alternative$_1$. This will result in a special case of DMLP that is intrinsically hard to solve. If we take $x_i = p_i = (p_{i1},\dots,p_{in_i})', i = 1,\dots,n$, and $x_{n+1} = (u_1,\dots,u_{n_u})'$, the imprecise decision model of (1) can be transformed into DMLP as

$$\begin{array}{ll} \min & f(x_1,\ldots,x_{n+1}) = \sum_{t=1}^{T} \prod_{j\in J_t} y_j \\ \text{s.t.} & X_i = \{x_i \in R^{n_i} : A_i x_i \le b_i, x_i \ge 0\},\ i = 1,\ldots,n+1 \end{array} \tag{2}$$

where J_t denotes the index set, and we can have at most one decision variable from x_i for each J_t. Therefore, for each product term, at most one decision variable from x_i occurs. This property of the objective function and the disjoint linear constraint sets, X_is, demonstrate the disjoint property of DMLP.

DMLP is in fact a global optimization issue because of its non-convexity, and thereby requires a local optimization strategy and a global optimization strategy, which automatically switch between refinement and exploration [10].

Before we start, a key solution property of DMLP has to be stated. Suppose each X_i is nonempty and bounded, the global solution of DMLP must consist of the basic feasible points of X_is. This important solution property can be derived from [8].

3 Local Optimization

Given a DMLP program, we first need a local optimization strategy that can quickly locate a local optimum and then use a global optimization strategy to escape the located optimum.

On the basis of its solution property, we can have the following algorithm for the local optimization phase of DMLP.

Algorithm 1:

(a). Find feasible extreme points $\tilde{x}_i^1 \in X_i, i = 1,\ldots,n$.

(b). [1] Solve: $\min\{f(\tilde{x}_1^1,\ldots,\tilde{x}_n^1,x_{n+1}) | x_{n+1} \in X_{n+1}\}$, to yield $\tilde{x}_{n+1}^1$;
[2] Solve: $\min\{f(x_1,\tilde{x}_2^1,\ldots,\tilde{x}_{n+1}^1) | x_1 \in X_1\}$, to yield $\tilde{x}_1^2$;
[3] Solve: $\min\{f(\tilde{x}_1^2,x_2,\tilde{x}_3^1,\ldots,\tilde{x}_{n+1}^1) | x_2 \in X_2\}$, to yield $\tilde{x}_2^2$;
⋮
[n+1] Solve: $\min\{\tilde{x}_1^2,\ldots,\tilde{x}_{n-1}^2,x_n,\tilde{x}_{n+1}^1) | x_n \in X_n\}$, to yield $\tilde{x}_n^2$;
Set $\tilde{x}_i^1 \leftarrow \tilde{x}_i^2, i = 1,\ldots,n$, and repeat (b) until it converges to a solution $(\bar{x}_1,\ldots,\bar{x}_{n+1})$.

(c). Suppose $\bar{x}_{n+1}$ is non-degenerate, and for each $x_i, i = 1,\ldots,n$, let $\hat{x}_{n+1} \in N_{X_{n+1}}(\bar{x}_{n+1})$ be such that

$$\begin{array}{l} f(\bar{x}_1,\ldots,\bar{x}_{i-1},\hat{x}_i,\bar{x}_{i+1},\ldots,\bar{x}_n,\hat{x}_{n+1}) \\ = \min_{x_i\in X_i} f(\bar{x}_1,\ldots,\bar{x}_{i-1},x_i,\bar{x}_{i+1},\ldots,\bar{x}_n,\hat{x}_{n+1}) \\ < f(\bar{x}_1,\ldots,\bar{x}_{n+1}). \end{array}$$

Go to (b) with $\tilde{x}_i^1 \leftarrow \bar{x}_i, i = 1,\ldots,i-1,i+1,\ldots,n$, and $\tilde{x}_i^1 \leftarrow \hat{x}_i$.

(d). Terminate with $(\bar{x}_1,\ldots,\bar{x}_{n+1})$ as a local solution.

In Algorithm 1, $N_{X_{n+1}}(\bar{x})$ denotes the set of extreme points in X_{n+1} that are adjacent to $\bar{x}$, and each step only involves linear programming (LP) operations. The derived local solution has been proved to be a KKT point [8]. Intuitively, such a solution acts as a local solution in X_{n+1}^k and a global solution in $X_i, i = 1,\ldots,n$.

4 Global Optimization

Given a local solution derived by Algorithm 1, we need to generate a cutting plane so as to cut off this local solution and obtain the global optimality. Here we employ the deepest cut, i.e., the polar cut [15, 16].

For simplicity, we present the basic idea about the generation of a polar cut with respect to disjoint bilinear programming (DBLP). For an overview about DBLP, please refer to [1].

In general, DBLP can be stated as

$$\begin{array}{ll} \min & f(x,y) = c^t x + d^t y + x^t C y \\ \text{s.t.} & X_0 = \{x \in R^{n_1} : A_1 x \le b_1, x \ge 0\}, \\ & Y_0 = \{y \in R^{n_2} : A_2 y \le b_2, y \ge 0\}. \end{array}$$

For a DBLP program, let $\bar{x}$ be an extreme point of X_0 and let $x_j, j \in N$, be the n_1 non-basic variables at $\bar{x}$, where N is the index set for non-basic variables. Barring the degenerate case and denoting by $\overline{a}^j$ the columns of the simplex tableau in extended form, then X_0 has precisely n_1 distinct edges incident to $\bar{x}$. Each half line $\xi^j = \{x : x = \bar{x} - \overline{a}^j \lambda_j, \lambda_j \ge 0\}, j \in \overline{N}$, contains exactly one such edge [3].

Definition: The *generalized reverse polar* of Y_0 for a given scalar α is given by $Y_0(\alpha) = \{x : f(x,y) \ge \alpha\}$ for all $y \in Y_0$.

Let $(\bar{x}, \bar{y})$ be a local solution derived by some optimization method like Algorithm 1, let ξ^j be defined as above, let α be the current best objective value (CBOV) of DBLP, and let $\overline{\lambda}_j$ be defined by

$$\begin{cases} \max\{\lambda_j : f(\bar{x} - a^j\lambda_j, y) \ge \alpha \text{ for all } y \in Y_0\} \text{ if } \xi^j \not\subset Y_0(\alpha) \\ -\max\{\lambda_j : f(\bar{x} + a^j\lambda_j, y) \ge \alpha \text{ for some } y \in Y_0\} \text{ if } \xi^j \subset Y_0(\alpha) \end{cases}$$

Then $\sum_{j \in \overline{N}} x_j / \overline{\lambda}_j \ge 1$ determines a valid cutting plane, and we denote by $H^+(\bar{x})$ the positive half-space defined by $\sum_{j \in \overline{N}} x_j / \overline{\lambda}_j \ge 1$.

To generate a polar cut, when $\xi^j \not\subset Y_0(\alpha)$, the modified Newton method requires around three LP operations to obtain $\overline{\lambda}_j$ [15]. We present herein another approach based on the LP duality theory, which costs only one LP iteration.

Consider DBLP and the first line of $\overline{\lambda}_j$, in which we need obtain

$$\begin{array}{l} \max\{\lambda_j : f(\bar{x} - e^j\lambda_j, y) \ge \alpha | A_2 y \le b_2, y \ge 0\} = \\ \max\{\lambda_j : \min_y [c^t(\bar{x} - a^j\lambda_j) + d^t y + (\bar{x} - a^j\lambda_j)^t C y] \ge \alpha \\ \qquad |A_2 y \le b_2, y \ge 0\} \end{array}$$

Using LP duality theory, the foregoing can be rewritten as

$$\max\{\lambda_j : \max_u[c^t(\bar{x}-a^j\lambda_j)+b_2^t u] \geq \alpha$$
$$|A_2^t u \leq d + C^t(\bar{x}-a^j\lambda_j), u \leq 0\} =$$
$$\max\{\lambda_j : \min_u[c^t a^j \lambda_j - b_2^t u] \leq c^t\bar{x}-\alpha$$
$$|C^t a^j \lambda_j + A_2^t u \leq C^t\bar{x}+d, u \leq 0\}$$
$$\Longleftrightarrow$$
$$\begin{array}{ll} \max_{(\lambda_j,u)} & \lambda_j \\ \text{s.t.} & \begin{bmatrix} c^t a^j & -b_2^t \\ C^t a^j & A_2^t \end{bmatrix} \begin{bmatrix} \lambda_j \\ u \end{bmatrix} \leq \begin{bmatrix} c^t\bar{x}-\alpha \\ C^t\bar{x}+d \end{bmatrix}, u \leq 0. \end{array}$$

Therefore, we can obtain $\overline{\lambda}_j$ when $\xi^j \not\subset Y_0(\alpha)$ by solving just one LP program. The cutting plane method for solving DBLP is built upon the traditional polar cut and its negative extension [15, 16]. For the case when $\xi^j \not\subset Y_0(\alpha)$, the new method based on the LP duality theory to generate $\overline{\lambda}_j$ can be applied. As for the case when $\xi^j \subset Y_0(\alpha)$, namely, for the computation of its negative extension, we still employ the modified Newton method.

Now we are ready to develop the generalized cutting plane method for solving DMLP by taking advantage of its disjoint property.

Algorithm 2:

(a). Let CBOV, $obj_0 = +\infty$, let the initial best feasible solution $\{(\hat{x}_1^0, \ldots, \hat{x}_{n+1}^0)\} = \emptyset$, and set the iteration number $k = 1$.
(b). If $X_{n+1}^k = \emptyset$, terminate with obj_{k-1} as the global minimum and $(\hat{x}_1^{k-1}, \ldots, \hat{x}_{n+1}^{k-1})$ as its corresponding global solution.
(c). Find a local solution by using Algorithm 1 with $X_{n+1} \leftarrow X_{n+1}^k$, and set $obj_k = \min\{obj_{k-1}, f(\bar{x}_1^k, \ldots, \bar{x}_{n+1}^k)\}$; $(\hat{x}_1^k, \ldots, \hat{x}_{n+1}^k) = \text{argmin}\{obj_{k-1}, f(\bar{x}_1^k, \ldots, \bar{x}_{n+1}^k)\}$.
(d). Compute $\overline{\lambda}_{jX_i}$ with respect to X_i for all $j \in N, i = 1, \ldots, n$.
(e). If either there exists no $\overline{\lambda}_{jX_i}$ such that $\xi^j \not\subset X_i(obj_k), i = 1, \ldots, n$; or there exists $\overline{\lambda}_{jX_i}$ such that $\xi^j \not\subset X_i(obj_k)$ but also exists $\overline{\lambda}_{jX_i} = 0$ such that $\xi^j \subset X_i(obj_k), i = 1, \ldots, n$, terminate with obj_k as the global minimum and $(\hat{x}_1^k, \ldots, \hat{x}_{n+1}^k)$ as its corresponding global solution.
(f). Let $\lambda_j = \min\{\overline{\lambda}_{jX_i}\}$ for all $j \in N, i = 1, \ldots, n$, generate a polar cut, and let $X_{n+1}^{k+1} = X_{n+1}^k \cap H^+(\bar{x}_{n+1}^k)$.
(g). Set $k = k+1$, and return to (b).

In Algorithm 2, N_{n+1}^k is either the original feasible when $k = 1$, or the reduced feasible after we introduce some polar cuts in k^{th} iteration when $k \neq 1$. It is in step (d) that we compute $\overline{\lambda}_j$s in order to generate either a traditional polar cut or its negative extension depending on whether $\xi^j \not\subset X_i(obj_k)$ is satisfied. The convergence proof of Algorithm 2 can be derived from the work in [16].

5 Conclusions

The proposed approach for dealing with DMLP incurred by imprecise decision analysis takes advantage of the disjoint structural property in local optimization and polar cuts in global optimization. It appears to be a different approach from the popular class of branch and bound methods, e.g., [12, 13], which are designed for solving either multilinear or multiplicative programs. Therefore, we need a further investigation in its computational performance against other counterparts.

Another interesting topic seems to be the generation of a polar cut with respect to a degenerate solution. The approach that incorporates disjunctive cuts seems relatively expensive [15]. Hence, it is necessary to investigate the performance of other possible approaches [2, 3].

References

[1] Al-Khayyal, F.A.: 1990, Jointly Constrained Bilinear Programs and Related Problems: An Overview, *Computers & Mathematics with Application*, **Vol. 19. 11**, pp. 53–62.

[2] Alarie, S., Audet, C., Jaumard, B. and Savard, G.: 2001, Concavity Cuts for Disjoint Bilinear Programming, *Mathematical Programming*, **Vol. 90. 2**, pp. 373–398.

[3] Balas, E.: 1971, Intersection Cuts – a New Type of Cutting Planes for Integer Programming, *Operations Research*, **Vol. 19**, pp. 19–39.

[4] Danielson, M.: 1997, Computational Decision Analysis, doctoral thesis, Department of Computer and Systems Sciences, Stockholm University and Royal Institute of Technology.

[5] Danielson, M. and Ekenberg, L.: 1998, A Framework for Analysing Decisions under Risk, *European Journal of Operational Research*, **Vol. 104. 3**, pp. 474–484.

[6] Danielson, M. and Ekenberg, L.: 2007, Computing Upper and Lower Bounds in Interval Decision Trees, *European Journal of Operational Research*, **Vol. 181. 2**, pp. 808–816.

[7] Ding, X. and Al-Khayyal, F.: 2006, Accelerating Convergence of Cutting Plane Algorithms for Disjoint Bilinear Programming, *Journal of Global Optimization*, **Vol. 38. 3**, pp. 421–436.

[8] Konno, H.: 1976, A Cutting Plane Algorithm for Solving Bilinear Programs, *Mathematical Programming*, **Vol. 11**, pp. 14–27.

[9] Malmnäs, P.-E.: 1996, Evaluations, Preferences, and Choice Rules, Report, Department of Philosophy, Stockholm University.

[10] Pardalos, P.M. and Romeijn, H.E.: 2002, Handbook of Global Optimization, Volume 2, Kluwer Academic Publishers, the Netherlands.

[11] Rios Insua, D. and French, S.: 1991, A Framework for Sensitivity Analysis in Discrete Multi-objective Decision-making, *European Journal of Operational Research*, **Vol. 54**, pp. 176–190.

[12] Ryoo, H.S. and Sahinidis, N.V.: 2001, Analysis of Bounds for Multilinear Functions, *Journal of Global Optimization*, **Vol. 19(4)**, pp. 403–424.

[13] Ryoo, H.S. and Sahinidis, N.V.: 2003, Global Optimization of Multiplicative Programs, *Journal of Global Optimization*, **Vol. 26(4)**, pp. 387–418.

[14] Salo, A. and Hämäläinen, R.P.: 1995, Preference Programming through Approximate Ratio Comparisons, *European Journal of Operational Research*, **Vol. 82(3)**, pp. 458–475.

[15] Sherali, H.D. and Shetty, C.M.: 1980, A Finitely Convergent Algorithm for Bilinear Programming Problems Using Polar Cuts and Disjunctive Face Cuts, *Mathematical Programming*, **Vol. 19**, pp. 14–31.

[16] Vaish, H. and Shetty, C.M.: 1977, A Cutting Plane Algorithm for the Bilinear Programming Problem, *Naval Research Logistics Quarterly*, **Vol. 24**, pp. 83–94.