MATHEMATICAL ENGINEERING
TECHNICAL REPORTS

Benders Decomposition Approach to Robust Mixed Integer Programming

Hiroo SAITO and Kazuo MUROTA

METR 2005–40 December 2005

DEPARTMENT OF MATHEMATICAL INFORMATICS
GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY
THE UNIVERSITY OF TOKYO
BUNKYO-KU, TOKYO 113-8656, JAPAN

WWW page: http://www.i.u-tokyo.ac.jp/mi/mi-e.htm

Benders Decomposition Approach to Robust Mixed Integer Programming

Hiroo Saito and Kazuo Murota

Department of Mathematical Informatics,
Graduate School of Information Science and Technology,
University of Tokyo, Tokyo 113-8656, Japan.
saito@misojiro.t.u-tokyo.ac.jp

December 24, 2005

Abstract

We consider mixed integer programming (MIP) problems with ellipsoidal uncertainty in problem data. Robust solutions to such problems are formulated as solutions of second-order cone programming problems with integer constraints, which we solve by an adaptation of the Benders decomposition technique towards MIP with conic constraints. Numerical computation against robust 0-1 knapsack problems and generalized assignment problems indicates that robustness can be achieved without substantial deterioration in optimal values.

Key words. mixed integer programming, robust optimization, ellipsoidal uncertainty, Benders decomposition

1 Introduction

Robust optimization, a methodology against data uncertainty in optimization, has attracted recent research interest. In continuous optimization, significant contributions were made by Ben-Tal and Nemirovski ([4, 8, 5, 6, 7, 9]) and El-Ghaoui et al. [16, 17]. Robust discrete optimization has also been studied, as is described in a monograph of Kouvelis and Yu [18] and a textbook of Bertsimas and Weismantel [13]; see also Averbakh [2], Averbakh and Berman [3], and Yaman, Karaşan, and Pinar [29]. In particular, Bertsimas and Sim [10, 11] and Atämturk [1] treat mixed integer programming (MIP) problems with some uncertainty described by intervals. Another paper [12] by Bertsimas and Sim, see also [27] by Sim, deals with 0-1 optimization problems with ellipsoidal uncertainty in objective functions and proposes an algorithm which finds a local optimal solution.

In this paper we consider MIP problems with ellipsoidal uncertainty in problem data. The robust counterpart is formulated as a second-order cone programming problem with integer constraints, which we solve by an adaptation of the Benders decomposition technique towards MIP with conic constraints. The relaxed problem resulting from the Benders decomposition is an ordinary MIP problem, for which we can make use of existing efficient softwares. The proposed algorithm is expected to converge to a global optimal solution under mild conditions. It is mentioned that the idea of the proposed approach has been presented in its crude form in [25] with preliminary computational experience, whereas the present paper is intended to be an improved presentation together with more extensive computational results.

The Benders decomposition is a computational technique well-established in the area of optimization and a number of variants with varying applicability and generality have been proposed (see, e.g., [22], [15]). The proposed variant is based on the duality of linear programming over symmetric cones and exploits the fact that conic programming can be solved efficiently by interior point methods. In the particular case of second-order cone programming problems with integer constraints, the proposed method enjoys an additional advantage that a cutting plane to be added to the relaxed problem can be obtained in an explicit form. This renders the proposed method significantly efficient.

As a natural alternative, it would be appropriate to mention the branch-and-bound approach to our problem. The idea is to branch on integer variables and derive upper-bounds by solving second-order cone programming problems obtained by relaxing the integer constraints. Even though second-order cone programming problems can be solved efficiently, it would be time-consuming to solve thousands of them. Our computational experience indicates that the proposed method compares favorably with the branch-and-bound method.

This paper is organized as follows. In Section 2, we present a MIP problem with second-order cone constraints as a robust counterpart of our problem with ellipsoidal uncertainty in linear constraints. In Section 3, we introduce a slightly more general problem, i.e., a MIP problem with a convex-cone constraint, and propose an adaptation of the Benders decomposition for it. This Benders decomposition takes a much simpler form when specialized to the case of second-order cone constraints, which is described in Section 4. In Section 5, we show computational results for robust 0-1 knapsack problems and generalized assignment problems.

2 Robust MIP with ellipsoidal uncertainty

We consider a mixed integer programming (MIP) problem with ellipsoidal uncertainty in linear constraints. Specifically we consider

$$\begin{array}{lll} \max\limits_{x,y}. & c^\top x + d^\top y & \\ \text{s. t.} & \tilde{a}_i^\top x + \tilde{b}_i^\top y \le f_i \quad (i = 1, \dots, m), \\ & x \in X, & \end{array} \tag{1}$$

where $c \in \mathbb{R}^{n_x}$, $d \in \mathbb{R}^{n_y}$, $\tilde{a}_i \in \mathbb{R}^{n_x}$, $\tilde{b}_i \in \mathbb{R}^{n_y}$, $f_i \in \mathbb{R}$, and $X = \{x \mid l \le x \le u,\ x \in \mathbb{Z}^{n_x}\}$ for some $l, u \in \mathbb{Z}^{n_x}$. Uncertainty lies in $\tilde{a}_i$ and $\tilde{b}_i$ in the following ellipsoidal manner:

$$\tilde{a}_i = a_i + P_i w_i, \quad \tilde{b}_i = b_i + Q_i w_i$$

with $\|w_i\| \le 1$, where $\|\cdot\|$ denotes the Euclidean norm, $a_i \in \mathbb{R}^{n_x}$, $b_i \in \mathbb{R}^{n_y}$, $w_i \in \mathbb{R}^{n_i}$, and P_i and Q_i are $n_x \times n_i$ and $n_y \times n_i$ matrices, respectively. We denote by $\mathcal{E}_i$ the set of such $(\tilde{a}_i, \tilde{b}_i)$, i.e.,

$$\mathcal{E}_i = \left\{ \begin{bmatrix} a_i \\ b_i \end{bmatrix} + \begin{bmatrix} P_i \\ Q_i \end{bmatrix} w_i \;\middle|\; \|w_i\| \le 1 \right\}.$$

By robust feasibility of a solution (x, y) we shall mean that (x, y) is feasible for all possible realizations of $(\tilde{a}_i, \tilde{b}_i)$ from $\mathcal{E}_i$ for $i = 1, \dots, m$. As is observed in [5], this leads us to the second-order cone constraint as follows. Recall that the second-order cone is a convex cone defined as

$$\mathcal{C} = \{(z_0, z_1) \in \mathbb{R} \times \mathbb{R}^n \mid z_0 \ge \|z_1\|\},$$

which is self-dual in that the dual cone of $\mathcal{C}$, defined by

$$\mathcal{C}^* = \{s \mid s^\top z \ge 0 \quad (\forall z \in \mathcal{C})\},$$

coincides with $\mathcal{C}$.

With the use of second-order cones $\mathcal{C}_i$ $(i = 1, \dots, m)$ in appropriate dimensions, the robust feasibility of a solution (x, y) is represented as follows:

$$\begin{aligned} & \forall (\tilde{a}_i\ \tilde{b}_i)^\top \in \mathcal{E}_i :\ \tilde{a}_i^\top x + \tilde{b}_i^\top y \le f_i \\ & \iff f_i \ge \max\{\tilde{a}_i^\top x + \tilde{b}_i^\top y \mid (\tilde{a}_i\ \tilde{b}_i)^\top \in \mathcal{E}_i\} \\ & \iff f_i \ge a_i^\top x + b_i^\top y + \|P_i^\top x + Q_i^\top y\| \\ & \iff h_i - A_i x - B_i y \in \mathcal{C}_i, \end{aligned}$$

where

$$A_i = \begin{bmatrix} a_i^\top \\ -P_i^\top \end{bmatrix}, \quad B_i = \begin{bmatrix} b_i^\top \\ -Q_i^\top \end{bmatrix}, \quad h_i = \begin{bmatrix} f_i \\ \mathbf{0} \end{bmatrix}.$$

Thus, the robust counterpart of (1) is formulated as

$$\begin{array}{ll} \max\limits_{x,y,\xi}. & c^\top x + d^\top y \\ \text{s. t.} & A_i x + B_i y + \xi_i = h_i \quad (i = 1, \dots, m), \\ & \xi_i \in \mathcal{C}_i \quad (i = 1, \dots, m), \\ & x \in X. \end{array} \tag{2}$$

We assume that (2) has an optimal solution.

With the notation

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_m \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ \vdots \\ B_m \end{bmatrix}, \quad G = \begin{bmatrix} I_1 & & \\ & \ddots & \\ & & I_m \end{bmatrix}, \quad h = \begin{bmatrix} h_1 \\ \vdots \\ h_m \end{bmatrix}, \quad \xi = \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_m \end{bmatrix} \tag{3}$$

and $\mathcal{K} = \mathcal{C}_1 \times \cdots \times \mathcal{C}_m$, we may further rewrite (2) in a more compact form:

$$\begin{array}{ll} \max\limits_{x,y,\xi}. & c^\top x + d^\top y \\ \text{s. t.} & Ax + By + G\xi = h, \\ & \xi \in \mathcal{K}, \\ & x \in X. \end{array} \tag{4}$$

Noting that (2) involves both integer and second-order cone constraints, we call it *mixed integer second-order cone programming.* We shall present a solution procedure for this problem based on the Benders decomposition technique.

Remark 1. In our problem formulation it is assumed that uncertainty lies only in the coefficients of the constraints, and not in the objective function nor in the right-hand side vector. This assumption, however, is not restrictive. If c, d or f_i is subject to uncertainty, we can consider an equivalent problem, with new variables t and $s = (s_1, \dots, s_m)$, of our form (2):

$$\begin{array}{ll} \max\limits_{x,y,t,s}. & t \\ \text{s. t.} & \tilde{a}_i^\top x + \tilde{b}_i^\top y - \tilde{f}_i s_i \leq 0 \qquad (i = 1, \dots, m), \\ & s_i \leq 1 \qquad (i = 1, \dots, m), \\ & -s_i \leq -1 \qquad (i = 1, \dots, m), \\ & -\tilde{c}^\top x - \tilde{d}^\top y + t \leq 0, \\ & x \in X. \end{array} \tag{5}$$

□

Remark 2. Interval uncertainty in linear constraints of (1) can be treated as follows. Uncertainty in $\tilde{a}_i$ and $\tilde{b}_i$ is represented as $\tilde{a}_i = a_i + P_i w_i$, $\tilde{b}_i = b_i + Q_i w_i$ with $w_i \in [-1, 1]^{n_i}$. We denote by $\mathcal{I}_i$ the set of such $(\tilde{a}_i, \tilde{b}_i)$. i.e.,

$$\mathcal{I}_i = \left\{ \begin{bmatrix} a_i \\ b_i \end{bmatrix} + \begin{bmatrix} P_i \\ Q_i \end{bmatrix} w_i \;\middle|\; w_i \in [-1, 1]^{n_i} \right\}.$$

The robust feasibility of a solution (x, y) is represented as follows:

$$\begin{aligned}
&\forall(\tilde{a}_i\ \tilde{b}_i)^\top \in \mathcal{I}_i:\ \tilde{a}_i^\top x + \tilde{b}_i^\top y \le f_i \\
&\iff f_i \ge \max\{\tilde{a}_i^\top x + \tilde{b}_i^\top y \mid (\tilde{a}_i\ \tilde{b}_i)^\top \in \mathcal{I}_i\} \\
&\iff f_i \ge a_i^\top x + b_i^\top y + \max\{w_i^\top(P_i^\top x + Q_i^\top y) \mid w_i \in [-1,1]^{n_i}\} \\
&\iff \exists z_i^+, z_i^- \text{ such that } \left\{ \begin{array}{l} f_i \ge a_i^\top x + b_i^\top y + \mathbf{1}^\top z_i^+ + \mathbf{1}^\top z_i^-, \\ -z_i^- \le P_i^\top x + Q_i^\top y \le z_i^+, \quad z_i^+ \ge 0, \quad z_i^- \ge 0, \end{array} \right.
\end{aligned}$$

where $\mathbf{1}$ denotes a vector whose elements are all 1. Hence, the robust counterpart of (1) is formulated as a MIP problem:

$$\begin{array}{ll}
\max\limits_{x,y,z^+,z^-} & c^\top x + d^\top y \\
\text{s. t.} & a_i^\top x + b_i^\top y + \mathbf{1}^\top z_i^+ + \mathbf{1}^\top z_i^- \le f_i \quad (i = 1, \ldots, m), \\
& -z_i^- \le P_i^\top x + Q_i^\top y \le z_i^+ \quad (i = 1, \ldots, m), \\
& z_i^+ \ge 0, \quad z_i^- \ge 0 \quad (i = 1, \ldots, m), \\
& x \in X.
\end{array} \tag{6}$$

□

3 Benders decomposition for MIP with conic constraints

As a general framework for our Benders decomposition approach to mixed integer second-order programming, we consider an optimization problem slightly more general than (4), where $\mathcal{K}$ is a general closed convex cone rather than the direct product of second-order cones. That is, we consider

$$\begin{array}{ll}
\max\limits_{x,y,\xi}. & c^\top x + d^\top y \\
\text{s. t.} & Ax + By + G\xi = h, \\
& \xi \in \mathcal{K}, \\
& x \in X
\end{array} \tag{7}$$

with a closed convex cone $\mathcal{K}$.

According to the general recipe of the Benders decomposition, we first eliminate the second-order cone variable ξ through projection. Let Ω denote the feasible region of (7). We define the projection of Ω onto the (x, y)-space by

$$\text{proj}(\Omega) = \{(x, y) \mid \exists\xi \text{ such that } (x, y, \xi) \in \Omega\}.$$

Let $\mathcal{Q}$ be a convex cone defined by

$$\mathcal{Q} = \{v \mid G^\top v \in \mathcal{K}^*\},$$

where $\mathcal{K}^*$ is the dual cone of $\mathcal{K}$, i.e.,

$$\mathcal{K}^* = \{s \mid s^\top \xi \ge 0 \quad (\forall \xi \in \mathcal{K})\}.$$

The set of extreme rays of $\mathcal{Q}$ is denoted by $\mathrm{Extr}\,(\mathcal{Q})$. We assume that $\mathcal{Q}$ has an interior feasible solution, i.e., there exists a vector v such that $G^\top v \in \mathrm{int}\,\mathcal{K}^*$.

An explicit representation of $\mathrm{proj}(\Omega)$ can be obtained through the Farkas lemma for closed convex cones.

Proposition 1. *Suppose that $\mathcal{Q}$ has an interior feasible solution and is closed. Then we have*

$$\begin{aligned}\mathrm{proj}(\Omega) &= \left\{ (x, y) \;\middle|\; \begin{array}{l} v^\top(Ax + By) \le v^\top h \quad (\forall v \in \mathcal{Q}), \\ x \in X \end{array} \right\} \\ &= \left\{ (x, y) \;\middle|\; \begin{array}{l} v^\top(Ax + By) \le v^\top h \quad (\forall v \in \mathrm{Extr}\,(\mathcal{Q})), \\ x \in X \end{array} \right\}.\end{aligned}$$

Proof. It suffices to show the first equality because the second easily follows from the fact that any element of a closed convex cone is a nonnegative combination of its extreme rays.

For any $(x, y) \in \mathrm{proj}(\Omega)$, there exists a ξ such that $(x, y, \xi) \in \Omega$. Then, by the definition of the dual cone, it holds that

$$v^\top h = v^\top(Ax + By + G\xi) = v^\top(Ax + By) + v^\top G\xi \ge v^\top(Ax + By)$$

for any $v \in \mathcal{Q}$.

Conversely, take any (x, y) which satisfies $x \in X$ and $v^\top(h - Ax - By) \ge 0$ for any $v \in \mathcal{Q}$. Since $\mathcal{Q}$ has an interior feasible solution, the set $\{G\xi \mid \xi \in \mathcal{K}\}$ is closed. By the Farkas lemma (Lemma 1 below), there exists $\xi \in \mathcal{K}$ such that $G\xi = h - Ax - By$. □

We have used the following lemma (see, e.g., Theorem 3.2.3 in [24]) in the proof of Proposition 1.

Lemma 1. (Farkas lemma) *Let $\mathcal{K} \subseteq \mathbb{R}^n$ be a closed convex cone and G be an $m \times n$ matrix. Suppose that the set $\{G\xi \mid \xi \in \mathcal{K}\}$ is closed. Then the system $G\xi = b,\ \xi \in \mathcal{K}$ has a solution if and only if $b^\top v \ge 0$ for each v with $G^\top v \in \mathcal{K}^*$.* □

Proposition 1 yields another equivalent formulation of (7) as a mixed integer semi-infinite programming problem as follows:

$$\begin{array}{ll} \displaystyle\max_{x,y}. & c^\top x + d^\top y \\ \text{s. t.} & v^\top(Ax + By) \le v^\top h \quad (\forall v \in \mathrm{Extr}\,(\mathcal{Q})), \\ & x \in X. \end{array} \tag{8}$$

Noting that (8) has an infinite number of linear constraints, we consider a relaxed problem with a certain finite set $R \subset \mathrm{Extr}\,(\mathcal{Q})$. This gives a MIP problem:

$$\text{MIP}(R) \left| \begin{array}{ll} \max\limits_{x,y}. & c^\top x + d^\top y \\ \text{s. t.} & v^\top (Ax + By) \le v^\top h \quad (\forall v \in R), \\ & x \in X. \end{array} \right. \tag{9}$$

Let $(\bar{x}, \bar{y})$ be an optimal solution of (9). If $(\bar{x}, \bar{y})$ is feasible for (8), $(\bar{x}, \bar{y})$ is an optimal solution of (8). The feasibility of $(\bar{x}, \bar{y})$ can be checked efficiently by solving the following conic programming problem, say, by an interior point method.

For any (x, y), we consider a conic linear programming problem described by

$$\text{D}(x, y) \left| \begin{array}{ll} \min\limits_{v}. & (h - Ax - By)^\top v \\ \text{s. t.} & G^\top v \in \mathcal{K}^*. \end{array} \right. \tag{10}$$

Proposition 2. A vector (x, y) with $x \in X$ is feasible for (8) if and only if the optimal value of $\text{D}(x, y)$ is equal to 0.

Proof. Take any (x, y) with $x \in X$. The feasibility of (x, y) for (8) can be written as

$$\min\{v^\top (h - Ax - By) \mid v \in \text{Extr}\,(\mathcal{Q})\} \ge 0.$$

Since $\mathcal{Q}$ is a closed convex set, we have

$$\min\{v^\top (h - Ax - By) \mid v \in \text{Extr}\,(\mathcal{Q})\} = \min\{v^\top (h - Ax - By) \mid v \in \mathcal{Q}\}.$$

The optimal value of this problem is either 0 or $-\infty$ because $\mathcal{Q}$ is a cone. □

Hence, if $\text{D}(\bar{x}, \bar{y})$ is finite for the optimal solution $(\bar{x}, \bar{y})$ of the relaxed problem (9), then $(\bar{x}, \bar{y})$ is feasible, and hence optimal, for (8). When $\text{D}(\bar{x}, \bar{y})$ is unbounded, $(\bar{x}, \bar{y})$ is not feasible for (8), but we can obtain an infinite direction, i.e., $v \in \mathcal{Q}$ such that $v^\top (h - A\bar{x} - B\bar{y}) < 0$. Then we add a cutting plane to (9) corresponding to this infinite direction v.

The proposed procedure for (7) with a closed convex cone $\mathcal{K}$ is summarized as follows. We take a convergence tolerance $\epsilon > 0$.

Step 1. Find a $v \in \mathcal{Q}$ and put $R \leftarrow \{v\}$.

Step 2. Solve the relaxed problem (9) to obtain an optimal solution $(\bar{x}, \bar{y})$.

Step 3. Solve $\text{D}(\bar{x}, \bar{y})$:

(a) If the optimal value is 0, then $(\bar{x}, \bar{y})$ is optimal for (8). Terminate.

(b) If the optimal value is unbounded, find an infinite direction v. If $v^\top (A\bar{x} + B\bar{y} - h) < \epsilon$, terminate. Otherwise, set $R \leftarrow R \cup \{v\}$ and goto Step 2.

In Step 3 (b), we find an infinite direction by solving the following problem, where **1** denotes a vector whose elements are all 1.

$$\mathrm{D}'(x,y)\left|\begin{array}{ll}\min\limits_{v^+,v^-}. & (h-Ax-By)^\top(v^+-v^-)\\ \text{s. t.} & G^\top(v^+-v^-)\in\mathcal{K}^*, \quad v^+,v^-\geq \mathbf{0},\\ & \mathbf{1}^\top v^+ + \mathbf{1}^\top v^- \leq 1.\end{array}\right.$$

Proposition 3. $\mathrm{D}(x,y)$ is unbounded if and only if the optimal value of $\mathrm{D}'(x,y)$ is negative. □

4 Benders decomposition for robust MIP

The Benders decomposition, as described in the previous section, takes a much simpler form when specialized to the case of (3) for the robust MIP problem (2).

Since $\mathcal{K}=\mathcal{C}_1\times\cdots\times\mathcal{C}_m$ is a direct product of second order cones, we have $\xi\in\mathcal{K}$ if and only if $\xi_i\in\mathcal{C}_i$ for $i=1,\ldots,m$. Since G is an identity matrix and $\mathcal{C}_i^*=\mathcal{C}_i$ for $i=1,\ldots,m$, we also have $\mathcal{Q}=\mathcal{C}_1^*\times\cdots\times\mathcal{C}_m^*=\mathcal{C}_1\times\cdots\times\mathcal{C}_m$ and $\mathrm{Extr}\,(\mathcal{Q})=\mathrm{Extr}\,(\mathcal{C}_1)\times\cdots\times\mathrm{Extr}\,(\mathcal{C}_m)$. Accordingly, the relaxed problem (9) is defined, with reference to a family of finite sets $R_i\subset\mathrm{Extr}\,(\mathcal{C}_i)$ $(i=1,\ldots,m)$, as

$$\mathrm{MIP}(R_1,\ldots,R_m)\left|\begin{array}{ll}\max\limits_{x,y}. & c^\top x+d^\top y\\ \text{s. t.} & v_i^\top(A_ix+B_iy)\leq v_i^\top h_i \quad (\forall v_i\in R_i) \quad (i=1,\ldots,m),\\ & x\in X.\end{array}\right. \tag{11}$$

The problem $\mathrm{D}(x,y)$ in (10) can easily be solved in this special case. Indeed the problem is decomposed into m independent problems, and each of them admits an explicit solution. To be specific, suppose that we have a solution $(\bar{x},\bar{y})$ to the relaxed problem (11) and put $\bar{\xi}_i=h_i-A_i\bar{x}-B_i\bar{y}$. If $\bar{\xi}_i$ lies in $\mathcal{C}_i$ for each $i=1,\ldots,m$, then we are done. Otherwise, for every i such that $\bar{\xi}_i\notin\mathcal{C}_i$ we find an infinite direction v_i from

$$\mathrm{D}_i''(\bar{x},\bar{y})\left|\begin{array}{ll}\min\limits_{v_i}. & \bar{\xi}_i^\top v_i\\ \text{s. t.} & v_i\in\mathcal{C}_i, \quad v_i(0)=1,\end{array}\right. \tag{12}$$

where $v_i(0)$ denotes the 0-th element of the vector v_i. Fortunately, the optimal solution to (12) can be given explicitly as follows.

Proposition 4. *The optimal solution to* (12) *is given by*

$$v_i=\begin{cases}(1,-(P_i^\top\bar{x}+Q_i^\top\bar{y})/\|P_i^\top\bar{x}+Q_i^\top\bar{y}\|)^\top & (P_i^\top\bar{x}+Q_i^\top\bar{y}\neq\mathbf{0}),\\ (1,\mathbf{0})^\top & (P_i^\top\bar{x}+Q_i^\top\bar{y}=\mathbf{0}).\end{cases} \tag{13}$$

Proof. This is geometrically easy to see, but we can also show this as follows. It is clear that v_i given by (13) is feasible for (12) in each case. The dual problem of (12) is

$$\mathrm{P}_i''(\bar{x}, \bar{y}) \left| \begin{array}{ll} \max\limits_{\xi_i, \mu_i}. & \mu_i \\ \text{s. t.} & \xi_i + \mu_i \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} = \bar{\xi}_i, \quad \xi_i \in \mathcal{C}_i. \end{array} \right. \tag{14}$$

Suppose that $P_i^\top \bar{x} + Q_i^\top \bar{y} \neq \mathbf{0}$. Then

$$\mu_i = (f_i - a_i^\top \bar{x} - b_i^\top \bar{y}) - \|P_i^\top \bar{x} + Q_i^\top \bar{y}\|, \quad \xi_i = \begin{bmatrix} \|P_i^\top \bar{x} + Q_i^\top \bar{y}\| \\ -P_i^\top \bar{x} - Q_i^\top \bar{y} \end{bmatrix}$$

is a feasible solution of (14) whose objective value is identical to that of (13) for (12). This shows the optimality of (13). The other case can be treated similarly. □

The proposed procedure for the robust MIP problem (2) is summarized as follows. We take a convergence tolerance $\epsilon > 0$.

Step 1. For each $i = 1, \ldots, m$, find a v_i that satisfies $v_i \in \mathcal{C}_i$, say $v_i = (1, \mathbf{0})^\top$, and put $R_i \leftarrow \{v_i\}$.

Step 2. Solve the relaxed problem (11) with $R_1, \ldots, R_m$ to find an optimal solution $(\bar{x}, \bar{y})$.

Step 3. Check for the feasibility of $(\bar{x}, \bar{y})$:

(a) If $h_i - A_i\bar{x} - B_i\bar{y} \in \mathcal{C}_i$ for each $i = 1, \ldots, m$, output $(\bar{x}, \bar{y})$, which is optimal for (2), and terminate.

(b) Otherwise, let $I = \{i \mid h_i - A_i\bar{x} - B_i\bar{y} \notin \mathcal{C}_i\}$ and

$$v_i = \begin{bmatrix} 1 \\ -(P_i^\top \bar{x} + Q_i^\top \bar{y})/\|P_i^\top \bar{x} + Q_i^\top \bar{y}\| \end{bmatrix}$$

for every $i \in I$. If $v_i^\top (A_i\bar{x} + B_i\bar{y} - h_i) < \epsilon$ for every $i \in I$, terminate. Otherwise, put $R_i \leftarrow R_i \cup \{v_i\}$ for each $i \in I$ and goto Step 2.

5 Computational results

This section presents computational results to demonstrate the proposed Benders decomposition method. In particular, we compare optimal values of robust counterpart problems with those of the nominal problems, which are free from uncertainty. We consider the knapsack problem and the generalized assignment problem.

Computations are performed on a PC with Intel Pentium M 1.2GHz CPU and 512MB RAM. Our implementation uses glpk-4.8 [19] to solve MIP problems.

5.1 Knapsack problem

We deal with the robust 0-1 knapsack problem

$$\begin{array}{ll} \max\limits_{x}. & c^\top x \\ \text{s. t.} & \tilde{a}^\top x \le f, \;\; x \in \{0,1\}^n \end{array}$$

which is of the form of (1) with $m = 1$, $n_x = n$, $n_y = 0$, and $X = \{0,1\}^n$.

The instances are generated as follows. We put $f = 4000$ and randomly generate 1000 instances with $c_j \in \{80, 81, \ldots, 120\}$ and $a_j \in \{100, 101, \ldots, 200\}$ $(j = 1, \ldots, n)$ for each of $n = 50, 100, 150,$ and 200. We set $P = \mathrm{diag}(\alpha, \ldots, \alpha)$; computation time is observed for $\alpha = 2.0$ by varying n, whereas the relationship between α and optimal objective value is observed by varying the value of α.

Table 1 shows the differences in optimal values between robust and nominal problems, where the discrepancy in percentage of the optimal objective values is measured by

$$\left| \frac{\text{nominal value} - \text{robust value}}{\text{nominal value}} \right| \times 100.$$

We can see that robustness is achieved without substantial deterioration in optimal values.

Table 1: Difference in optimal values between robust and nominal problems (%)

(1000 instances for each n)

n	50	100	150	200	250	300
max.	1.24	0.83	0.73	0.72	0.85	0.73
avr.	0.52	0.43	0.39	0.39	0.39	0.40
min.	0.21	0.25	0.25	0.26	0.26	0.27

Table 2 shows the number of iterations in the Benders decomposition and Figure 1 is a histogram of the number of iterations for instances with $n = 300$. These results show that almost all instances are solved within a few iterations.

Table 3 shows the computation time of the Benders decomposition method, where “s.d.” means the standard deviation. Robust problems can be solved within 5 or 6 minutes, mostly with reasonable increase in computation time compared with that for nominal problems. It mentioned, however, that in some instances the computation time is very long.

Figure 2 shows how the objective value changes in iterations. We see that we obtain near-optimal values within a few iterations.

Table 2: Number of iterations in the Benders decomposition

	(1000 instances for each n)					
n	50	100	150	200	250	300
max.	8	7	6	7	9	11
avr.	2.06	2.44	2.34	2.29	2.30	2.62
min.	2	2	2	2	2	2

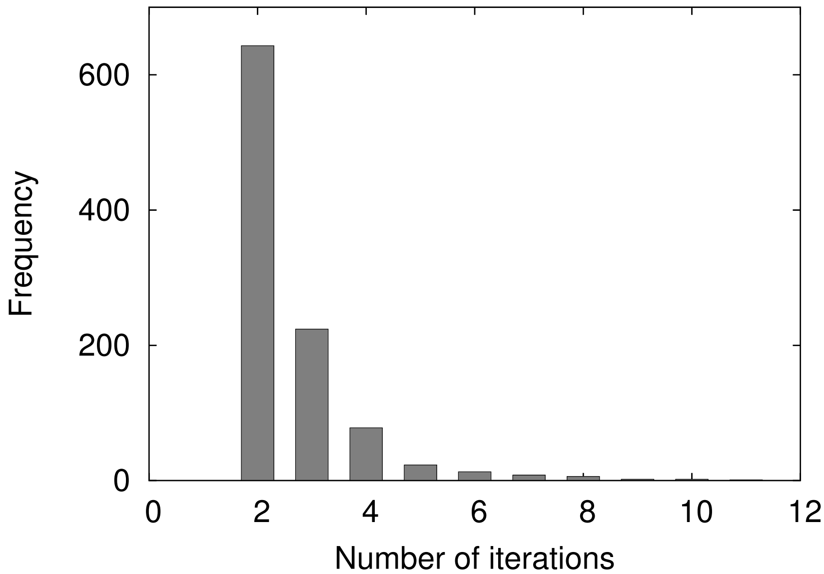

Figure 1: Histogram of the number of iterations ($n = 300$)

Table 3: Computation time (in seconds) for the knapsack problem

(1000 instances for each n)

	n	50	100	150	200	250	300
robust	max.	501.25	371.31	73.64	3540.45	5659.24	126913.93
	upper quartile	0.17	0.51	0.96	1.79	3.16	9.66
	median	0.06	0.24	0.45	0.75	1.23	2.29
	lower quartile	0.03	0.14	0.28	0.44	0.64	1.00
	min.	0.01	0.02	0.08	0.11	0.16	0.26
	s.d.	21.31	11.84	2.99	120.79	190.45	4324.85
nominal	max.	481.96	44.26	33.86	3531.88	978.95	6154.12
	upper quartile	0.08	0.19	0.37	0.65	1.18	2.13
	median	0.03	0.10	0.18	0.30	0.47	0.72
	lower quartile	0.02	0.05	0.11	0.18	0.28	0.41
	min.	0.01	0.01	0.01	0.01	0.02	0.03
	s.d.	18.41	1.57	1.31	111.64	37.68	260.61

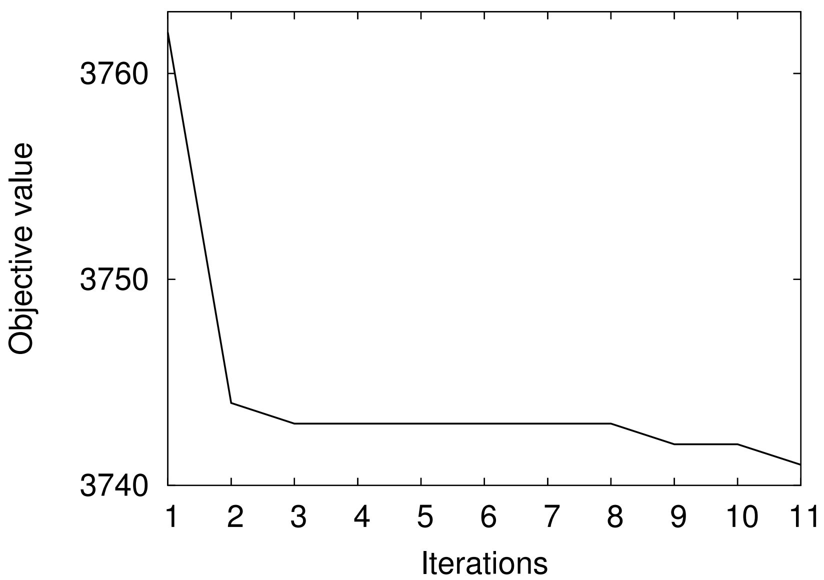

Figure 2: Objective value of the knapsack problem at each iteration of an instance with $n = 300$

Figure 3 shows the relationship between the magnitude of uncertainty α with the optimal objective values. The parameter α is varied from 0.0 to 20.0 by 0.5. The optimal value gets smaller as the magnitude α of uncertainty gets larger. We can see that this decrease is step-wise due to discrete nature of the problem.

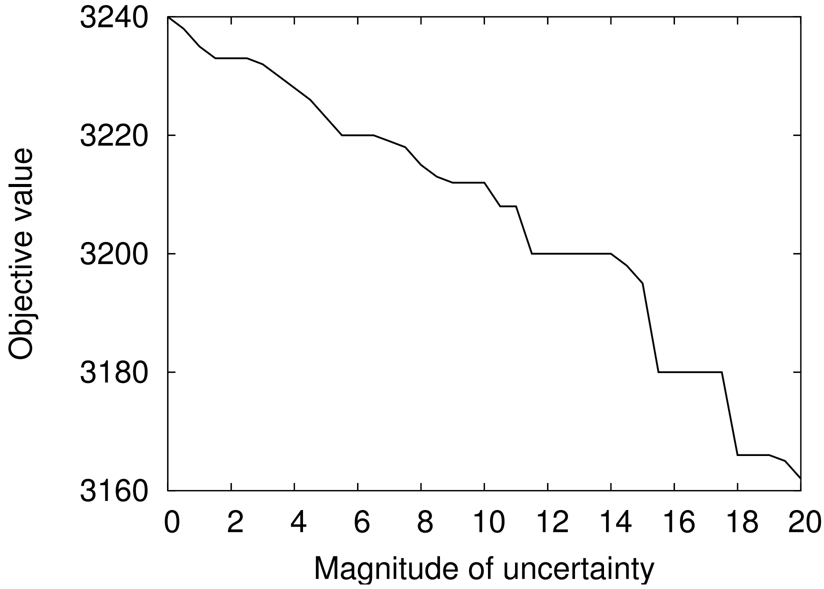

Figure 3: Relationship between uncertainty and the optimal objective value for an instance of the knapsack problem with $n = 50$

We next compare the Benders decomposition approach with the branch-and-bound approach. We implemented a branch-and-bound method in C++ language using ss-4.3.3 [20] to solve relaxed second-order cone programming problems. We tried with relatively small instances ($n = 50$) to find that the branch-and-bound takes about 15 hours, whereas the Benders decomposition method takes only a few seconds. The branch-and-bound method is slow because it takes time to solve many SOCP problems.

Finally we compare the ellipsoidal uncertainty with the interval uncertainty. According to Remark 2, the robust counterpart of the 0-1 knapsack problem with interval uncertainty reads as follows:

$$\begin{array}{ll} \max\limits_{x,z^+,z^-}. & c^\top x \\ \text{s. t.} & a^\top x + \mathbf{1}^\top z^+ + \mathbf{1}^\top z^- \leq f, \\ & -z^- \leq P^\top x \leq z^+, \quad z^+ \geq 0, \quad z^- \geq 0, \quad x \in \{0,1\}^n. \end{array}$$

Interval uncertainty and ellipsoidal uncertainty are compared in Table 4 in terms of the optimal objective values of the robust problems. The third rows of the table show the ratios in percentage of the two optimal objective values:

$$\left|\frac{\text{optimal value with interval uncertainty}}{\text{optimal value with ellipsoidal uncertainty}}\right| \times 100.$$

Table 4 shows that ellipsoidal uncertainty is superior to interval uncertainty in that optimal value does not change so much. It may be said that the ellipsoidal uncertainty is more appropriate than the interval uncertainty, the latter of which is often too "conservative" as discussed in the literature (see, e.g., [9]).

Table 4: Comparison of optimal values with interval and ellipsoidal uncertainty for the 0-1 knapsack problem (%)

(100 instances for each n)

	n	50	100	150	200	250	300
ellipsoidal	max.	100	99.89	99.87	99.89	99.87	99.84
/nominal	min.	99.10	99.47	99.55	99.59	99.59	99.46
interval	max.	99.34	98.90	98.71	98.68	98.71	98.84
/nominal	min.	98.27	98.19	98.14	98.09	98.20	98.07
interval	max.	99.49	99.19	99.10	99.04	98.95	99.84
/ellipsoidal	min.	98.57	98.44	98.30	98.34	98.42	98.33

5.2 Generalized assignment problem

Let M and N be the sets of machines and jobs, respectively. Each job $j \in N$ must be processed on one of the machines, say $i \in M$, and it requires cost c_{ij}. In addition, each machine has a capacity, denoted by b_i for $i \in M$, which limits the number of jobs to be assigned. The generalized assignment problem is to determine an optimal assignment that minimizes the total processing cost (see, e.g., [21, 26, 28] for details). This problem can be formulated with binary variables x_{ij} for $i \in M$ and $j \in N$ as follows:

$$\text{GAP} \quad \left| \begin{array}{lll} \min\limits_{x}. & \sum\limits_{i\in M}\sum\limits_{j\in N} c_{ij}x_{ij} & \\ \text{s. t.} & \sum\limits_{i\in M} x_{ij} = 1 & (\forall j \in N), \\ & \sum\limits_{j\in N} a_{ij}x_{ij} \leq b_i & (\forall i \in M), \\ & x_{ij} \in \{0,1\} & (\forall i \in M,\ \forall j \in N). \end{array} \right. \qquad (15)$$

Computation has been done in the following manner. The benchmark problems distributed by [14, 23, 28] are employed. We pick a constraint, say i-th constraint, to incorporate uncertainty by $P_i = 0.1 \times \text{diag}(a_{i1}, \ldots, a_{in})$.

This means that there is a single machine whose capacity has some uncertainty for each job. Table 5 summarizes the results. In particular, the difference in the optimal objective values between the robust and nominal problems is small.

Table 5: Computation time (in seconds) and number of iterations for the generalized assignment problem

instances	$\|M\|$	$\|N\|$	time	iterations	difference (%)
a05100	5	100	0.57	3	0.12
a05200	5	200	0.87	1	0
a10100	10	100	0.64	1	0
a10200	10	200	1.29	1	0
a20100	20	100	2.32	1	0
a20200	20	200	96.17	2	0.04
b05100	5	100	1687.71	3	0.54
b05200	5	200	72699.50	10	0.25
b10100	10	100	119.82	1	0
c05100	5	100	527.34	2	0.47
c05200	5	200	4165.73	3	0.17

Acknowledgments

The authors thank Akiko Yoshise for communicating references on conic programming. This work is supported by the 21st Century COE Program on Information Science and Technology Strategic Core, and by a Grant-in-Aid for Scientific Research from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

[1] A. Atämturk, Strong formulations of robust mixed 0-1 programming, *Math. Program.* to appear.

[2] I. Averbakh, On the complexity of a class of combinatorial optimization problems with uncertainty, *Math. Program.* **90** (2001) 263–272.

[3] I. Averbakh and O. Berman, Algorithms for the robust 1-center problem on a tree, *European J. Oper. Res.* **123** (2000) 292–302.

[4] A. Ben-Tal, A. Goryashko, E. Guslitzer and A. Nemirovski, Adjustable robust solutions of uncertain linear programs, *Math. Program.* **99** (2004) 351–376.

[5] A. Ben-Tal and A. Nemirovski, Robust convex optimization, *Math. Oper. Res.* **23** (1998) 769–805.

[6] A. Ben-Tal and A. Nemirovski, Robust solutions of uncertain linear programs, *Oper. Res. Lett.* **25** (1999) 1–13.

[7] A. Ben-Tal and A. Nemirovski, Robust solutions of linear programming problems contaminated with uncertain data, *Math. Program.* **88** (2000) 411–424.

[8] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization — Analysis, Algorithms and Engineering Applications*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2001.

[9] A. Ben-Tal and A. Nemirovski, Robust optimization—methodology and applications, *Math. Program.* **92** (2002) 453–480.

[10] D. Bertsimas and M. Sim, Robust discrete optimization and network flows, *Math. Program.* **98** (2003) 49–71.

[11] D. Bertsimas and M. Sim, The price of robustness, *Oper. Res.* **52** (2004) 35–53.

[12] D. Bertsimas and M. Sim, Robust discrete optimization under ellipsoidal uncertainty sets, http://web.mit.edu/dbertsim/www/papers.html (2004).

[13] D. Bertsimas and R. Weismantel, *Optimization over Integers*, Dynamic Ideas, Belmont, 2005.

[14] D. Cattrysse, M. Salomon and L.N. Van Wassenhove, A set partitioning heuristic for the generalized assignment problem, *European J. Oper. Res.* **72** (1994) 167–174.

[15] A.M. Geoffrion, Generalized Benders decomposition, *J. Optim. Theory Appl.* **10** (1972) 237–260.

[16] L. El-Ghaoui and H. Lebret, Robust solutions to least-squares problems with uncertain data, *SIAM J. Matrix Anal. Appl.* **18** (1997) 1035–1064.

[17] L. El-Ghaoui, F. Oustry and H. Lebret, Robust solutions to uncertain semidefinite programs, *SIAM J. Optim.* **9** (1999) 33–52.

[18] P. Kouvelis and G. Yu, *Robust Discrete Optimization and Its Applications*, Kluwer, Norwell, 1997.

[19] A. Makhorin, GLPK—GNU Linear Programming Kit. (http://www.gnu.org)

[20] M. Muramatsu, SS—SOCP Solver.
(http://jsb.cs.uec.ac.jp/~muramatu/SS/)

[21] R.M. Nauss, Solving the generalized assignment problem: An optimizing and heuristic approach, *INFORMS J. Comput.* **15** (2003) 249–266.

[22] G.L. Nemhauser and L.A. Wolsey, *Integer and Combinational Optimization*, John Wiley & Sons, New York, 1988.

[23] I.H. Osman, Heuristics for the generalised assignment problem: Simulated annealing and tabu search approaches, *OR Spektrum* **17** (1995) 211–225.

[24] J. Renegar, *A Mathematical View of Interior-Point Methods in Convex Optimization*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2001.

[25] H. Saito and K. Murota, Benders decomposition for robust mixed integer programming (in Japanese), *Proc. Spring Conference of the Operations Research Society of Japan*, 2005, 236–237; also a paper (in Japanese) submitted.

[26] M. Savelsbergh, A branch-and-price algorithm for the generalized assignment problem, *Oper. Res.* **45** (1997) 831–841.

[27] M. Sim, *Robust Optimization*, Ph.D. Thesis, Operations Research Center, MIT, June 2004.
(http://www.bschool.nus.edu/STAFF/dscsimm/research.htm)

[28] M. Yagiura, T. Ibaraki and F. Glover, An ejection chain approach for the generalized assignment problem, *INFORMS J. Comput.* **16** (2004) 133–151.

[29] H. Yaman, O.E. Karaşan and M. Ç. Pinar, The robust spanning tree problem with interval data, *Oper. Res. Lett.* **29** (2001) 31–49.