<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Stochastic Programming - Maximum Probability Model</td>
<td></td>
</tr>
<tr>
<td>Author(s)</td>
<td>Kataoka, Shinji</td>
<td></td>
</tr>
<tr>
<td>Citation</td>
<td>Hitotsubashi journal of arts and sciences, 8(1): 51-59</td>
<td></td>
</tr>
<tr>
<td>Issue Date</td>
<td>1967-09</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>Departmental Bulletin Paper</td>
<td></td>
</tr>
<tr>
<td>Text Version</td>
<td>publisher</td>
<td></td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://doi.org/10.15057/4330">http://doi.org/10.15057/4330</a></td>
<td></td>
</tr>
</tbody>
</table>
STOCHASTIC PROGRAMMING
—MAXIMUM PROBABILITY MODEL—

By Shinji Kataoka*

This paper presents a stochastic programming model where the probability of a stochastic linear objective function being greater than a specified value is maximized under deterministic linear constraints. A simple computational procedure is proposed here, utilizing Wolfe's method of quadratic programming in the long form, when the random variables in the objective function have a multinormal distribution.

I. Models and Formulation

Letting $x$ be a vector of $n$ decision variables, $x_1, x_2, \ldots, x_n$, consider the following optimization problem:

\begin{align*}
\max & \quad l(x) = P_r(\theta \leq \pi'x), \\
\text{subject to} & \quad Ax = b, \quad x \geq 0,
\end{align*}

where $P_r$ denotes probability and $\pi$ is a vector of which components are random variables $\pi_1, \pi_2, \ldots, \pi_n$ of known distributions. $\theta$ is a given constant (on which we impose an upper bound later, ref. Section III) and $A$ an $(m \times n)$-matrix and $b$ an $m$-vector. In this paper we call the problem (1) and (2) stochastic programming of maximum probability.

In order to transform the problem (1) and (2) into a computable model, we make an assumption on the random variables $\pi$s:

**Assumption 1:** The distribution function of the random variables $\pi$s is a multinormal distribution with mean values $\mu$ and a variance matrix $V=(v_{ij})$ such that $\mu_i = E\pi_i$, $v_{ij} = E(\pi_i - \mu_i)(\pi_j - \mu_j)$. Then for the mean and variance of $\pi'x$, we have

$$E(\pi'x) = \mu'x,$$

$$E(\pi'x - \mu'x)^2 = x'Vx.$$

By the above definition it is clear that the quadratic form $x'Vx$ is positive semi-definite. However for the sake of simplicity in the main part of our discussion we assume, for the time being,

**Assumption 2:**

$$x'Vx > 0,$$

and the case $x'Vx = 0$ will be discussed in the latter part of this section.

Define linear programming such that

* Professor (Kyoju) in Mathematics.
**Problem 0:**

\[
\max \ p'x, \\
\text{subject to } Ax=b, \ x \geq 0.
\]

For the later discussion we further assume

**Assumption 3:** The optimal solution of the Problem 0, \( \hat{x} \), is bounded.

Using the Assumption 1 and 2, the original problem, (1) and (2), is proved to be equivalent to the following non-linear programming:

**Problem 1:**

(3)

\[
\max \ h(x)=\frac{p'x-\theta}{\sqrt{x'Vx}} \\
\text{subject to } Ax=b, \ x \geq 0.
\]

**Proof:**

\[
Pr(\theta \leq \pi^* x) = Pr\left(\frac{p'x-\theta}{\sqrt{x'Vx}} \leq \frac{p'x-\theta}{\sqrt{x'Vx}}\right).
\]

Because of the Assumption 1, \((p'-\pi)x/\sqrt{x'Vx}\) is a normal random variable with the distribution \(N(0, 1)\). Therefore we have

\[
Pr(\theta \leq \pi^* x) = \int_{-\infty}^{h(x)} e^{-\frac{y^2}{2}} dy,
\]

and (3) follows.

Let us denote the optimal solution by \( \hat{x} \). Since, unfortunately, the objective function \( h(x) \) is not concave, the usual computation methods for the convex programming such as the gradient methods are not used directly to this Problem 1. In order to avoid this difficulty, we first consider the following problem as a help to proceed to the solution of the Problem 1.

**Problem 2:**

(4)

\[
\max \ g(x)=p'x-q\sqrt{x'Vx}, \\
\text{subject to } Ax=b, \ x \geq 0,
\]

where \( q \) is a constant (which is changed parametrically later). Let us denote the optimal solution by \( x^* \) or sometimes by \( x^*(q) \). This Problem 2 was treated by this author previously, where the concavity of the objective function \( g(x) \) was proved. Since \( x'Vx > 0 \) is further assumed in this case, \( g(x) \) is strictly concave over the convex domain \( Ax=b, \ x \geq 0 \). Hence we have

**Theorem 1:** The Problem 2 has one and only one optimal solution.

In the previous paper we also derived an algorithm for the Problem 2 introducing the following subsidiary quadratic programming:

**Problem 3:**

(5)

\[
\max \ f(x)=lp'x-\frac{1}{2}x'Vx,
\]
subject to \( Ax = b, \ x \geq 0, \)

where \( \lambda \) is a positive parameter. Let us denote the optimal solution by \( \hat{x} \) or \( \hat{x}(\lambda) \).

**Theorem 2:** A necessary and sufficient condition for the optimal solution of the Problem 3 \( \hat{x}(\lambda) \) to be that of the Problem 2 is that the value of the parameter \( \lambda \) satisfies

\[
\lambda q = \sqrt{\hat{x}(\lambda)' V \hat{x}(\lambda)}.
\]

**Proof:** Let the Lagrangian function of the Problem 2 be

\[
\varphi(x, u) = p'x - q \sqrt{x' V x} - u'(Ax - b),
\]

a necessary and sufficient condition for \( x^* \) to be optimal is that there exists a vector \( u^* \) such that

\[
\begin{align*}
\frac{\partial \varphi}{\partial x_j} &= p_j - q \frac{(x'^* V)_{jj} - (u'^* A)_{jj}}{\sqrt{x'^* V x'^*}} = 0 \text{ for } x^*_j > 0, \\
&= 0 \text{ for } x^*_j = 0,
\end{align*}
\]

(7)

\[
\begin{align*}
A x^* &= b.
\end{align*}
\]

(8)

On the other hand a necessary and sufficient condition for \( \hat{x}(\lambda) \) to be optimal for the Problem 3 is that there exists a vector \( t_\lambda \) such that

\[
\begin{align*}
\lambda p_j - (\hat{x}(\lambda)' V)_{jj} &- (u(\lambda)' A)_{jj} = 0 \text{ for } \hat{x}(\lambda)_j > 0, \\
&= 0 \text{ for } \hat{x}(\lambda)_j = 0,
\end{align*}
\]

(9)

\[
A \hat{x}(\lambda) = b.
\]

(10)

Suppose the condition (6) is satisfied by \( \hat{x}(\lambda) \) and let the solution be \( \lambda^* \), dividing (9) by \( \lambda^* \) and using (6), we have

\[
\begin{align*}
p_j - q \frac{(\hat{x}(\lambda^*)' V)_{jj} - (u(\lambda^*)' A)_{jj}}{\sqrt{\hat{x}(\lambda^*)' V \hat{x}(\lambda^*)}} &= 0 \text{ for } \hat{x}(\lambda^*)_j > 0, \\
&= 0 \text{ for } \hat{x}(\lambda^*)_j = 0.
\end{align*}
\]

Therefore \( x^* = \hat{x}(\lambda^*), \ u^* = u(\lambda^*)/\lambda^* \) will satisfy (7) and (8). Conversely when \( x^* \) satisfies (7) and (8), let \( \lambda = \sqrt{x'^* V x'^*} q, \ \hat{x} = x^* \) and \( u = u^* \), then these \( \hat{x} \) and \( u \) will fulfill (9) and (10).

Q.E.D.

So far we have assumed \( x' V x > 0 \) (Assumption 2). Let us examine the case \( x' V x = 0 \) here. By the well-known theorem on the positive semi-definite matrix, we have that \( x' V x = 0 \) implies \( V x = 0 \). Consequently for the original problem (1) and (2), we get the following linear programming problem:

\[
\begin{align*}
\text{max} & \quad p' x \\
\text{subject to} & \quad V x = 0, \\
& \quad A x = b, \quad x \geq 0.
\end{align*}
\]

(11)

Suppose an optimal solution \( x^* \) for this linear programming be obtained. If, comparing \( l(x^*) \) with the optimal value of the Problem 1 \( l(x^*), \ l(x^*) > l(x^*) \), we can conclude that the optimal solution will be \( x^* \). In a special case when the matrix \( V \) is positive definite, \( x' V x = 0 \) implies \( x = 0 \). Therefore we also have to examine whether \( l(0) > l(x^*) \) or not.

Now we are able to compute an optimal solution by solving a quadratic programming problem under the side condition (6). In the previous paper we presented an iteration method
and later made a successful computer program for it. However by computational experiences it turned out that a number of iterations were needed for attaining the optimum. This time, then, another simpler method of utilizing Wolfe’s long form quadratic programming is proposed.

II. Quadratic Programming

Among several computational methods for quadratic programming, Wolfe’s simplex method is most suitable to our present problem. In this section we make a survey of this method, partly modifying it for our purpose.

The quadratic programming to be discussed here is just formulated as the Problem 3:

$$\text{max } f(x) = \frac{1}{2} x' V x, \quad \text{subject to } A x = b, \quad x \geq 0.$$  

Making use of the theorem of Kuhn-Tucker, as before, we have a necessary and sufficient condition for $x$ to be an optimal solution:

\begin{align}
\lambda p_j - \sum_{k=1}^{n} v_{jk} x_k - \sum_{l=1}^{m} u_l a_{lj} & = 0 \quad \text{for } x_j > 0, \\
\sum_{k=1}^{n} a_{ik} x_k & = b_i, \quad i = 1, ..., m; \quad j = 1, ..., n.
\end{align}

Introducing slack variables $v(v_1, ..., v_n)$ and artificial variables $w(w_1, ..., w_m)$, $z(z_1, ..., z_n)$, we transform (12) and (13) into the followings:

\begin{align}
V x - v' + A' u^+ - A' u^- + E z' & = \lambda p, \\
A x & = b, \\
v x = 0, \\
x, v, u^+, u^-, w, z \geq 0,
\end{align}

where, because the vector $u$ may be positive or negative, two nonnegative vectors $u^+$ and $u^-$ are introduced such that $u = u^+ - u^-; \ E$ is a diagonal matrix,

$$E = \begin{pmatrix}
\varepsilon_1 \\
\varepsilon_2 \\
\varepsilon_n
\end{pmatrix}, \quad \varepsilon_j = \begin{cases}
+1 & \text{for } p_j \geq 0, \\
-1 & \text{for } p_j < 0.
\end{cases}$$

The condition (16) means that both $v_j$ and $x_j (j = 1, ..., n)$ never have positive values at the same time.

**Short Form:** A method to obtain an optimal solution for a specified value of $\lambda$.

Choosing $w$ and $z$ as an initial basis,

$$\text{minimize } \sum w_i + \sum z_j$$

under the side condition (16), that is, avoiding to let both $v_j$ and $x_j$ get into the basis simultaneously. If the value of the objective function is reduced to zero, computation is terminated, and otherwise the Problem 3 has no optimal solution.
Long Form: A method to get all optimal solutions for nonnegative values of $\lambda$.

**Phase 1:** Perform the short form for $\lambda=0$:
\[
Vx - v' + A'u^+ - A'u^- + z' - \mu p = 0,
\]
\[
Ax + w' = b,
\]
minimize $\sum v_i + \sum z_i$,
under the side condition (16) and the nonnegative conditions (17), keeping $\mu=0$.

**Phase 2:** Discarding $w$ and $z$ from the variables and using the final basis of the Phase 1 as the initial one,

maximize $\mu$
subjected to $Vx - v' + A'u^+ - A'u^- - \mu p = 0$,
and $Ax = b$,
under the conditions (16) and $x, v, u^+, u^- \geq 0$. Then the simplex algorithm will yield a finite sequence of $\mu$ and $x$-parts of their associate basic feasible solutions:

$0 = \mu_0 < \mu_1 < \ldots < \mu_K$, and $x^0, x^1, \ldots x^K$,
terminating with the vector $x^K = \bar{x}$ (for $\lambda \to \infty$), the optimal solution of the Problem 0, since we assume that $\bar{x}$ is bounded. The explicit expression of $\bar{x}(\lambda)$ will then be

(18) $\bar{x}(\lambda) = \frac{\lambda}{\mu_1} x^1$ for $0 = \mu_0 \leq \lambda \leq \mu_1$,
(19) $\bar{x}(\lambda) = \frac{\mu_{k+1} - \lambda}{\mu_{k+1} - \mu_k} x^k + \frac{\lambda - \mu_k}{\mu_{k+1} - \mu_k} x^{k+1}$ for $\mu_k \leq \lambda \leq \mu_{k+1}$,
(20) $\bar{x}(\lambda) = \bar{x}$ for $\mu_K < \lambda$,
\[k=1, 2, \ldots, K-1\].

III. Theorems for Computational Procedures

Before proceeding to the computational procedures for the Problems 2 and 1, several fundamental theorems concerning the properties of the solutions for those problems are proved in this section. From (18), (19) and (20) we have

**Lemma 1:** The optimal solution $\bar{x}(\lambda)$ is a one-valued continuous function of $\lambda$.

Let $S(\lambda)$ be the standard deviation of the random variable $\pi' \bar{x}(\lambda)$:

(21) $S(\lambda) = \sqrt{\bar{x}(\lambda)' \bar{x}(\lambda)}$,
and especially let $S(\mu_k)$ be $S_k$. Then we have

**Lemma 2:** $S(\lambda)$ is a continuous bounded monotone nondecreasing function of $\lambda$.

**Proof:** The continuity and boundedness are easily derived from (18), (19) and (20). Letting $x$ and $v$ be the optimal solution of the Problem 3 for $\lambda_x$ and $\lambda_y$ respectively, we have

$\lambda_x \pi' x - \frac{1}{2} \pi' V x \leq \lambda_y \pi' y - \frac{1}{2} \pi' V y$, 

Dividing the first inequality by $\lambda_1$ and the second by $\lambda_2$, and adding them, we have

$$
(\lambda_1 - \lambda_2)(x'Vx - y'Vy) \geq 0,
$$

where the equality holds for $\lambda_1, \lambda_2 \geq \mu_k$.

**Theorem 3:** \( S(\lambda)/\lambda \) is constant for \( 0 < \lambda \leq \mu_1 \) and is monotone decreasing for \( \mu_1 < \lambda \leq \mu_k \).

**Proof:** The first part of the theorem is clear from (18). Since \( S(\lambda) \) is monotone increasing for \( 0 < \lambda \leq \mu_k \) and is constant for \( \lambda = \mu_k \), if \( S(\lambda)/\lambda \) did not decrease for \( 0 < \lambda \leq \mu_k \), the line \( q_\lambda \) \((q > 0)\) would have more than one separate intersections with the curve \( S(\lambda) \). This fact contradicts with the Theorem 1 and Theorem 2 because \( g(x) \) is strictly concave. Furthermore it is proved that on the curve \( S(\lambda) \) for \( \mu_k < \lambda < \mu_{k+1} \), there does not exist a line segment of which extended line goes through the origin: if it existed, from (19) we could conclude that

$$
\frac{x^k}{\mu_k} - \frac{x^{k+1}}{\mu_{k+1}}.
$$

On the other hand the constraints \( Ax \leq b \) must be satisfied by both \( x_k \) and \( x_{k+1} \), hence \( \mu_k \) has to be equal to \( \mu_{k+1} \). This fact contradicts with the assumption \( 0 < \mu_k < \mu_{k+1} \) \((k \neq 0)\). Q.E.D.

**Corollary 1:** The sequence \( q_i \) \((i=1, 2, ..., K)\) is monotone decreasing.

Let \( \lambda^*(q) \) be a positive root of the equation \( q\lambda = S(\lambda) \), then we have

**Corollary 2:** \( \lambda^*(q) \) is a continuous monotone decreasing function of \( q \) for \( q \leq q_1 \) and \( \lambda^*(q) = 0 \) for \( q > q_1 \).

**Lemma 3:** \( x^k \) is an optimal solution of the Problem 2 for \( q = q_1 \).

**Proof:** By the definition, \( x^k \) satisfies the Problem 3 for \( \lambda = \mu_k \) and the condition (16), then by the Theorem 2 \( x^k \) is proved to be the solution of the Problem 2 for \( q = q_1 \). Q.E.D.

**Lemma 4:** The solution of the Problem 2, \( x^*(q) \) is a continuous function of \( q \).

**Proof:** This is derived from the Theorem 2, Lemma 1 and Corollary 2. Q.E.D.

**Theorem 4:** The optimal value of the Problem 2, \( g^*(q) \), is a continuous monotone decreasing function of \( q \).

**Proof:** Since \( g^* = p'x^*(q) - q\sqrt{x^TVx} \) and \( x^* \) is continuous in \( q \), then \( g^*(q) \) is continuous. Suppose \( x, y \) be optimal solutions of the Problem 2 for \( q_x \) and \( q_y \) \((q_x < q_y)\) respectively. If it were that

$$
p'x - q_x\sqrt{x'Vx} \leq p'y - q_y\sqrt{y'Vy},$$

because

$$
p'y - q_y\sqrt{y'Vy} < p'y - q_x\sqrt{y'Vy},$$

then

$$
p'x - q_x\sqrt{x'Vx} < p'y - q_x\sqrt{y'Vy}.$$
Therefore this contradicts with the assumption of \( x \) being a unique optimum for the parameter \( q_x \).

**Q.E.D.**

**Corollary 3:**

\[ x^*(q_i) = \hat{x}(\mu_i) = x^i. \]

Defining \( g_i = p'x^i - q_i \), we have

**Corollary 4:** The sequence \( g_i \) is monotone decreasing.

As is easily seen, the objective function of the Problem 2, \( g(x) \), is convex for \( q \geq 0 \), and the maximum value of \( g^*(q) \) is attained at \( q = 0 \). Hence the domain of \( \theta \) for the present computational method being available is \( \theta \leq p' \bar{x} \), where \( \bar{x} \) is an optimal solution of the Problem 0.

### IV. Computational Procedures

Preparing the above mentioned definitions and theorems, now we are going to show a process for computation of the Problem 2 and 1.

**Problem 2:**

\[
\max g(x) = p'x - q_x \sqrt{x'Vx}, \\
\text{subject to } Ax = b, \quad x \geq 0.
\]

(A1) Generate the Problem 3:

\[
\max f(x) = \lambda p'x - \frac{1}{2} x'Vx, \\
\text{subject to } Ax = b, \quad x \geq 0.
\]

(A2) Solve the Problem 3 in the long form and get series \( 0 = \mu_1 < \mu_2 < ... < \mu_K \) and \( x^0, x^1, ..., x^K (= \bar{x}) \).

(A3) Compute \( s_i = \sqrt{x'^iVx^i}, \quad q_i = s_i/\mu_i \).

(A4) Determine an integer \( k \) such that \( q_k \leq q < q_{k+1} \).

(A5) Compute

\[
s_2 = \frac{1}{(\mu_{k+1} - \mu_k)^2} (x^{k+1} - x^k)'V(x^{k+1} - x^k),
\]

\[
s_1 = \frac{\mu_k \mu_{k+1}}{(\mu_{k+1} - \mu_k)^2} (x^{k+1} - x^k)'V \left( \frac{x^{k+1}}{\mu_{k+1}} - \frac{x^k}{\mu_k} \right),
\]

\[
s_0 = \frac{(\mu_k \mu_{k+1})^2}{(\mu_{k+1} - \mu_k)^2} \left( \frac{x^{k+1}}{\mu_{k+1}} - \frac{x^k}{\mu_k} \right)'V \left( \frac{x^{k+1}}{\mu_{k+1}} - \frac{x^k}{\mu_k} \right).
\]

(A6) Solve a quadratic equation of \( \lambda \), \( s_2 \lambda^2 - 2s_1 \lambda + s_0 = q^2 \lambda^2 \), and let a positive root be \( \lambda^* \).

(A7) Then the optimal solution will be

\[
x^* = \hat{x}(\lambda^*) = \frac{\mu_{k+1} - \lambda^*}{\mu_{k+1} - \mu_k} x^k + \frac{\lambda^* - \mu_k}{\mu_{k+1} - \mu_k} x^{k+1},
\]

and \( g^* = p'x^* - q \lambda^* \).
Problem 1:

\[ \max h(x) = \frac{p' x - \theta}{\sqrt{x' V x}}, \]
subject to \( Ax = b, \quad x \geq 0. \)

(B1) same as (A1).

(B2) same as (A2).

(B3) Compute
\[ S_i = \sqrt{x_i' V x_i}, \quad q_i = S_i / \mu_i \]
and
\[ g_i = p' x_i - q_i S_i \quad \text{for} \quad i = 1, \ldots, K. \]

(B4) Determine an integer \( k \) such that \( g_k \leq \theta < g_{k+1}. \)

(B5) Solve the following simultaneous equations of \( \lambda \) and \( q. \)
\[ \theta = p' \mathcal{E}(\lambda) - q S(\lambda), \]
\[ q \lambda = S(\lambda), \]
\[ \mathcal{E}(\lambda) = \frac{\mu_{k+1} - \lambda}{\mu_{k+1} - \mu_k} x^k + \frac{\lambda - \mu_k}{\mu_{k+1} - \mu_k} x^{k+1}, \]
or for this purpose, compute \( s_0, s_1, s_9 \) and
\[ m_1 = \frac{p'(x^k - x^k)}{\mu_{k+1} - \mu_k}, \]
\[ m_0 = \frac{\mu_k \mu_{k+1}}{\mu_{k+1} - \mu_k} p' \left( \frac{x^{k+1}}{\mu_{k+1}} - \frac{x^k}{\mu_k} \right), \]
and solve a quadratic equation of \( \lambda: \)
\[ (s_1 - m_1) x^2 + (-2 s_1 + m_0 + \theta) x + s_0 = 0. \]

(B6) Letting \( \tilde{\lambda} \) be a positive root of the equation above, compute
\[ \tilde{q} = \sqrt{(m_1 \tilde{\lambda} - m_0 - \theta) / \tilde{\lambda}}. \]

Then the optimal solution of the Problem 1 will be given by
\[ \tilde{x} = \frac{\mu_{k+1} - \tilde{\lambda}}{\mu_{k+1} - \mu_k} x^k + \frac{\tilde{\lambda} - \mu_k}{\mu_{k+1} - \mu_k} x^{k+1}, \]
and
\[ h(\tilde{x}) = \frac{p' \tilde{x} - \theta}{\sqrt{\tilde{x}' \tilde{x}}} = \tilde{q}. \]

Numerical Example:

Problem 0: \( \max x_1 + x_2, \) subject to \( 2x_1 + x_2 + x_3 = 3; \) \( x_1, x_2, x_3 \geq 0. \)

optimal solution: \( x = (0, 3, 0). \)

Problem 1: \( \max h(x) = \frac{x_1 + x_2 - 1}{\sqrt{x_1^2 + x_2^2}}, \) subject to the same as above.
Problem 3: \( \max f(x) = \lambda(x_1 + x_2) - \frac{1}{2} (x_1^2 + x_2^2), \) 
\( 2x_1 + x_2 + x_3 = 3; \ x_1, x_2, x_3 \geq 0. \)

Solve the Problem 3:
\( \mu_1 = 1, \ \mu_2 = 6; \ x^1 = (1, 1, 0), \ x^2 = (0, 3, 0) \)

Compute \( S_1, q_1, g_1: \)
\( S_1 = \sqrt{2}, \ S_2 = 3; \ q_1 = \sqrt{2}, \ q_2 = 1/2; \ g_1 = 0, \ g_2 = 3/2. \)

Determine \( k: \) since \( 0 < \theta = 1 < 3/2, \) then \( k = 1. \)

Compute \( s \) and \( m: \)
\( s_1 = 1/5, \ s_2 = 0, \ s_3 = 9/5; \ m_1 = 1/5, \ m_2 = -9/5. \)

Solve: \( (-9/5 + 1)\lambda + 9/5 = 0, \ \lambda = 9/4. \)

Compute optimal solution:
\( \hat{x} = (3/4, 6/4, 0), \ \hat{q} = h(\hat{x}) = \sqrt{5}/3. \)

Problem 2: \( \max g(x) = x_1 + x_2 - (\sqrt{5}/3) \sqrt{x_1^2 + x_2^2}, \)
\( 2x_1 + x_2 + x_3 = 3; \ x_1, x_2, x_3 \geq 0. \)

(A1), (A2) same as (B1), (B2).

Compute \( S_1, q_1: \)
\( S_1 = \sqrt{2}, \ S_2 = 3; \ q_1 = \sqrt{2}, \ q_2 = 1/2. \)

Determine \( k: \)
\( \sqrt{2} < q = \sqrt{5}/3 < 1/2, \ k = 1. \)

Compute \( s_2, s_1, s_0: \)
\( s_2 = 1/5, \ s_1 = 0, \ s_0 = 9/5. \)

Solve \( \lambda, \) and obtain optimal solution:
\( (1/5)x^2 + 9/5 = (5/9)\lambda^2, \ \lambda^* = 9/4, \)
\( x^* = (3/4, 6/4, 0), \ g^* = 1. \)

References