

Some Remarks on Application of Sandwich Methods in the Minimum Cost Flow Problem

Marta Kostrzewska¹, Lesław Socha²

¹Institute of Mathematics, University of Silesia, Katowice, Poland

²Faculty of Mathematics and Natural Sciences, Cardinal Stefan Wyszyński University, Warsaw, Poland

Email: marta.kostrzewska83@gmail.com, leslawsocha@poczta.onet.pl

Received November 27, 2011; revised December 25, 2011; accepted January 12, 2012

ABSTRACT

In this paper, two new sandwich algorithms for the convex curve approximation are introduced. The proofs of the linear convergence property of the first method and the quadratic convergence property of the second method are given. The methods are applied to approximate the efficient frontier of the stochastic minimum cost flow problem with the moment bicriterion. Two numerical examples including the comparison of the proposed algorithms with two other literature derivative free methods are given.

Keywords: Bicriteria Network Cost Flow Problem; Sandwich Algorithms; Efficient Frontier; Stochastic Costs

1. Introduction

The network cost flow problems which describe a lot of real-life problems have been studied recently in many Operation Research papers. One of the basic problems in this field are the bicriteria optimization problems. Although there exist exact computation methods for finding the analytic solution sets of bicriteria linear and quadratic cost flow problems (see e.g. [1,2]), Ruhe [3] and Zadeh [4] have shown that the determination of these sets may be very perplexing, because there exists the possibility of the exponential number of extreme nondominated objective vectors on the efficient frontier of the considered problems. The fact that efficient frontiers of bicriteria linear and quadratic cost flow problems are the convex curves in R^2 allows to apply the sandwich methods for a convex curve approximation in this field of optimization (see e.g. [5-8]). However, in some of these algorithms the derivative information is required. A derivative free method was introduced first by Yang and Goh in [8], who applied it to bicriteria quadratic minimum cost flow problems. The efficient frontiers of these problems are approximated by two piecewise linear functions called further approximation bounds, which construction requires solving of a number of one dimensional minimum cost flow problems. Unfortunately, the method introduced by Yang and Goh works under the assumption that the change of the direction of the tangents of the approximated function is less than or equal to $\frac{\pi}{2}$. Also, Siem *et al.* in [7] proposed an algorithm based only on

the function value evaluation, with the interval bisection partition rule and two new iterative strategies for the determination of the new input data point in each iteration. Authors gave the proof of linear convergence of their algorithm.

In this paper we consider the generalized bicriteria minimum cost flow problem. We are interested in minimizing two cost functions, which satisfy some additional assumptions. Two sandwich methods for the approximation of the efficient frontier of this problem are presented. In the first method, based on the algorithm proposed by Siem *et al.* [7], new points on the efficient frontier are computed according to the chord rule or the maximum error rule by solving proper convex network problems. In the second method, we modify the lower approximation function discussed in [8], what decreases the Hausdorff distance between upper and lower bounds. We give the proofs of the linear convergence property of the first method called the Simple Triangle Algorithm and the quadratic convergence property of the second method called the Trapezium Algorithm.

The paper is organized as follows. In Section 2, we state a nonlinear bicriteria optimization problem that can be treated as a generalized minimum cost flow problem. In Section 3, two new sandwich methods of approximation of the efficient frontier of the stated problem are presented and in Subsection 3.5 the corresponding algorithms are presented. In Section 4, we discuss the convergence of these algorithms. Section 5 includes the information how to use the methodology from Section 2 in the case of the stochastic minimum cost flow problem

with the moment bicriterion. To illustrate discussed methods in comparison with the algorithms presented in [7] and [8] two numerical examples are given. Finally, Section 6 contains the conclusions and future research direction. Proofs of lemmas and Theorem 2 are given in Appendix.

2. Problem Statement

Let G be the directed network with n nodes and m arcs. Let $R_+ = [0, +\infty)$. We consider the generalized minimum cost flow problem (GMCFP) defined as follows

$$\begin{aligned} & \min [h_1(x), h_2(x)]^T \\ & \text{s.t. } Dx = b, \\ & c_l \leq x \leq c_u, \end{aligned} \tag{1}$$

where $D \in R^{n \times m}$ is the node-arc incidence matrix, $b \in R^n$ is the net outflow of the nodes, vectors $c_l, c_u \in R_+^m$ are the lower and upper capacity bounds for the flow $x \in R_+^m$ and $h_1, h_2 : R^m \rightarrow R_+$ are the cost functions such that $h_2(\cdot)$ is a continuous and convex function and $h_1(x) = h(c^T x)$, where $c \in R_+^m$ and $h : R_+ \rightarrow R_+$ is a continuous, concave and strictly increasing function. Let

$$X = \{x \in R_+^m : Dx = b, c_l \leq x \leq c_u\}$$

be the *feasible set* of problem (1).

According to the concept of Pareto optimality we consider the relations \leq and $<$ in R^2 defined as follows

$$\begin{aligned} [a_1, a_2]^T \leq [b_1, b_2]^T & \Leftrightarrow a_1 \leq b_1 \text{ and } a_2 \leq b_2, \\ [a_1, a_2]^T < [b_1, b_2]^T & \Leftrightarrow a_1 < b_1 \text{ and } a_2 < b_2. \end{aligned}$$

Using these definitions in the field of the bicriteria programming, a feasible solution $x \in X$ is called the *efficient solution* of problem (1) if there does not exist a feasible solution $y \in X$ such that

$$[h_1(y), h_2(y)]^T \leq [h_1(x), h_2(x)]^T. \tag{2}$$

The set of all efficient solutions and the image of this set under the objective functions are called the *efficient set* and the *efficient frontier*, respectively.

Note that the efficient set X_E of problem (1) is the same as the efficient set of following problem

$$\begin{aligned} & \min [h^{-1}(h_1(x)), h^{-1}(h_2(x))]^T \\ & \text{s.t. } Dx = b, \\ & c_l \leq x \leq c_u. \end{aligned} \tag{3}$$

Moreover, from the fact that $h^{-1}(h_1(x)) = c^T x$ for all $x \in X$ and that the function $h^{-1} \circ h_2(\cdot)$ is a convex one it follows the next lemma.

Lemma 1

Efficient frontier of problem (3) is a convex curve in R^2 .

Proof: See Appendix.

3. The New Sandwich Methods

In this section, we introduce two new sandwich methods for the approximation of the efficient frontier of problem (3).

3.1. Initial Set of Points

Let $f_1 = h^{-1} \circ h_1(\cdot)$ and $f_2 = h^{-1} \circ h_2(\cdot)$ and suppose that

$$P_k = (f_1(x^k), f_2(x^k)) \tag{4}$$

for all $k = 1, 2, \dots, r$ are r given points on the efficient frontier of problem (1) such that x^1 and x^r are the lexicographical minimum for the first and the second criterion, respectively. Although we need only three given points on the efficient frontier to start the first method called the Simple Triangle Method (STM) and two points to start the second method called the Trapezium Method (TM), the described methodologies work for any number r of initial points, which may be obtained by solving scalarization problems corresponding to problem (3), *i.e.*

$$\begin{aligned} & \min (\lambda_k f_1(x) + (1 - \lambda_k) f_2(x)) \\ & \text{s.t. } x \in X, \end{aligned} \tag{5}$$

where $\lambda_k = \frac{k-1}{r-1}$ for all $k = 1, 2, \dots, r$.

Another possibility is to find lexicographical minima of problem (3) and then to solve $r-2$ convex programming problems with additional equality constraints

$$c^T x = c^T x^1 + \frac{k}{r-1} (c^T x^r - c^T x^1), \tag{6}$$

where $k = 1, \dots, r-2$. This method gives r points on the efficient frontier with the following property

$$f_1(x^{k+1}) - f_1(x^k) = \frac{1}{r-1} (f_1(x^r) - f_1(x^1)) \tag{7}$$

for all $k = 1, 2, \dots, r-1$.

3.2. Upper Bound

Suppose that the initial set $P = \{P_1, \dots, P_r\}$ of points on the efficient frontier is given and that the points are ordered according to the first criterion $f_1(x^k) < f_1(x^{k+1})$ for all $k = 1, \dots, r-1$.

The upper approximation function $u^k(\cdot)$ of the frontier on the subinterval $[f_1(x^k), f_1(x^{k+1})]$ called the upper bound is defined as the straight line through the points P_k and P_{k+1} , that is

$$u^k(a) = f_2(x^k) + \frac{f_2(x^{k+1}) - f_2(x^k)}{f_1(x^{k+1}) - f_1(x^k)} (a - f_1(x^k)) \tag{8}$$

for $a \in [f_1(x^k), f_1(x^{k+1})]$.

3.3. Lower Bounds

In the algorithms proposed at the end of this section we will use two different definitions of the lower approximation functions called the lower bounds.

Definition 1

According to [7] the straight lines defined by the points P_{k-1} and P_k and the points P_{k+1} and P_{k+2} approximate the frontier from below so the lower bound $l^k(\cdot)$ on the interval $[f_1(x^k), f_1(x^{k+1})]$ may be constructed in the following form

$$l^k(a) = \begin{cases} u^{k-1}(a) & \text{for } a \in [f_1(x^k), a_k] \\ u^{k+1}(a) & \text{for } a \in [a_k, f_1(x^{k+1})], \end{cases} \quad (9)$$

where $k = 2, \dots, r-2$ and a_k is the point of intersection of two linear functions $u^{k-1}(\cdot)$ and $u^{k+1}(\cdot)$.

Moreover, we define the lower approximation bound $l^k(\cdot)$ on the most left and the most right interval as follows

$$l^1(a) = u^2(a) \text{ for } a \in [f_1(x^1), f_1(x^2)] \quad (10)$$

and

$$l^r(a) = \begin{cases} u^{r-1}(a) & \text{for } a \in [f_1(x^{r-1}), a_{r-1}] \\ f_2(x^r) & \text{for } a \in [a_{r-1}, f_1(x^r)], \end{cases} \quad (11)$$

where a_{r-1} is the point of intersection of function $u^{r-1}(\cdot)$ and the constant function $f_2(x^r)$.

If we compute new points on the efficient frontier due to the chord rule (see next section), then definition (9) may be modified, see Rote [9], in the following way

$$l^k(a) = \begin{cases} f_2(x^k) + \frac{f_2(x^{k+1}) - f_2(x^{k-1})}{f_1(x^{k+1}) - f_1(x^{k-1})} (a - f_1(x^k)) & \text{for } a \in [f_1(x^k), a_k] \\ f_2(x^{k+1}) + \frac{f_2(x^{k+2}) - f_2(x^k)}{f_1(x^{k+2}) - f_1(x^k)} (a - f_1(x^{k+1})) & \text{for } a \in [a_k, f_1(x^{k+1})], \end{cases} \quad (12)$$

where $k = 2, \dots, r-2$ and a_k is the point of intersection of corresponding two linear functions. Note that the lower bound is constructed by the tangents to P_k and P_{k+1} .

Moreover, the lower approximation bound $l^k(\cdot)$ on the most left and the most right interval are redefined as

follows

$$l^1(a) = f_2(x^2) + \frac{f_2(x^3) - f_2(x^1)}{f_1(x^3) - f_1(x^1)} (a - f_1(x^2)) \quad (13)$$

for $a \in [f_1(x^1), f_1(x^2)]$

and

$$l^r(a) = \begin{cases} f_2(x^r) + \frac{f_2(x^{r+1}) - f_2(x^{r-1})}{f_1(x^{r+1}) - f_1(x^{r-1})} (a - f_1(x^k)) & \text{for } a \in [f_1(x^k), a_k] \\ f_2(x^r) & \text{for } a \in [a_{r-1}, f_1(x^r)]. \end{cases} \quad (14)$$

Definition 2

The simple modification of the definition presented in [8] leads to the following form of lower approximation bound $l_*^k(\cdot)$ on the interval $[f_1(x^k), f_1(x^{k+1})]$

$$l_*^k(a) = \begin{cases} u^{k-1}(a) & \text{for } a \in [f_1(x^k), b_k] \\ f_2(y^k) + \frac{f_2(x^{k+1}) - f_2(x^k)}{f_1(x^{k+1}) - f_1(x^k)} (a - f_1(y^k)) & \text{for } a \in [b_k, c_k] \\ u^{k+1}(a) & \text{for } a \in [c_k, f_1(x^{k+1})], \end{cases} \quad (15)$$

where $k = 2, \dots, r-2$, constants b_k and c_k are the points of intersection of corresponding linear functions and y^k is the solution of the following convex network problem (the chord rule problem)

$$\min \left(f_2(x) - \frac{f_2(x^k) - f_2(x^{k+1})}{f_1(x^k) - f_1(x^{k+1})} f_1(x) \right) \quad (16)$$

s.t. $x \in X$.

Similar to the previous case, we define the lower approximation bound on the most left and the most right interval as follows

$$l_*^1(a) = \begin{cases} f_2(y^1) + \frac{f_2(x^2) - f_2(x^1)}{f_1(x^2) - f_1(x^1)} (a - f_1(y^1)) & \text{for } a \in [f_1(x^1), c_1] \\ u^2(a) & \text{for } a \in [c_1, f_1(x^2)] \end{cases} \quad (17)$$

and

$$l_*^r(a) = \begin{cases} u^{r-1}(a) & \text{for } a \in [f_1(x^{r-1}), b_{r-1}] \\ f_2(x^r) & \text{for } a \in [b_{r-1}, f_1(x^r)]. \end{cases} \quad (18)$$

Moreover, these definitions may be modified like in (12) using the tangents in points P_2, \dots, P_{r-1} in the following form

$$l_*^k(a) = \begin{cases} f_2(x^k) + \frac{f_2(x^{k+1}) - f_2(x^{k-1})}{f_1(x^{k+1}) - f_1(x^{k-1})} (a - f_1(x^k)) & \text{for } a \in [f_1(x^k), b_k] \\ f_2(y^k) + \frac{f_2(x^{k+1}) - f_2(x^k)}{f_1(x^{k+1}) - f_1(x^k)} (a - f_1(y^k)) & \text{for } a \in [b_k, c_k] \\ f_2(x^{k+1}) + \frac{f_2(x^{k+2}) - f_2(x^k)}{f_1(x^{k+2}) - f_1(x^k)} (a - f_1(x^{k+1})) & \text{for } a \in [c_k, f_1(x^{k+1})] \end{cases} \quad (19)$$

and

$$l_*^1(a) = \begin{cases} f_2(y^1) + \frac{f_2(x^2) - f_2(x^1)}{f_1(x^2) - f_1(x^1)} (a - f_1(y^1)) & \text{for } a \in [f_1(x^1), c_1] \\ f_2(x^2) + \frac{f_2(x^3) - f_2(x^1)}{f_1(x^3) - f_1(x^1)} (a - f_1(x^2)) & \text{for } a \in [c_1, f_1(x^2)] \end{cases} \quad (20)$$

and

$$l_*^r(a) = \begin{cases} f_2(x^r) + \frac{f_2(x^{r+1}) - f_2(x^{r-1})}{f_1(x^{r+1}) - f_1(x^{r-1})} (a - f_1(x^k)) & \text{for } a \in [f_1(x^{r-1}), b_{r-1}] \\ f_2(x^r) & \text{for } a \in [b_{r-1}, f_1(x^r)]. \end{cases} \quad (21)$$

If the approximation bounds $u^k(\cdot)$ and $l^k(\cdot)$ or $l_*^k(\cdot)$ are constructed for each $k = 1, \dots, r-1$, then we define the upper approximation function $u(\cdot)$ of the efficient frontier of problem (3) in the following way

$$u(a) = u^k(a) \text{ for } a \in [f_1(x^k), f_1(x^{k+1})] \quad (22)$$

and the lower approximation function due to equality

$$l(a) = l^k(a) \text{ for } a \in [f_1(x^k), f_1(x^{k+1})] \quad (23)$$

or

$$l_*(a) = l_*^k(a) \text{ for } a \in [f_1(x^k), f_1(x^{k+1})] \quad (24)$$

for all $k = 1, \dots, r-1$.

We note that after a small modification of the definition of the lower approximation function in the most right interval, any convex function may be approximated by the lower and upper bounds defined in this subsection.

3.4. Error Analysis

Suppose that the approximation bounds have been built and let $\delta = \max\{\delta_1, \dots, \delta_{r-1}\}$, where δ_k denotes the approximation error on the interval $[f_1(x^k), f_1(x^{k+1})]$. We consider three different error measures called the *Maximum error measure (Maximum vertical error measure)* (δ_k^M), the *Hausdorff distance measure* (δ_k^H) and the *Uncertainty area measure* (δ_k^U) (see [7])

$$\delta_k^M = \max_{a \in [f_1(x^k), f_1(x^{k+1})]} |u_k(a) - l_k(a)|, \quad (25)$$

$$\delta_k^H = \max \left\{ \sup_{v \in L} \inf_{w \in U} \|v - w\|, \inf_{w \in U} \sup_{v \in L} \|v - w\| \right\}, \quad (26)$$

where

$$U = \{(a, u_k(a)) : a \in [f_1(x^k), f_1(x^{k+1})]\},$$

$$L = \{(a, l_k(a)) : a \in [f_1(x^k), f_1(x^{k+1})]\}$$

and

$$\delta_k^U = \int_{f_1(x^k)}^{f_1(x^{k+1})} |u_k(a) - l_k(a)| da. \quad (27)$$

If a measure δ (one of δ^M , δ^H , δ^U) does not satisfy a desired accuracy, we choose $k \in \{1, \dots, r\}$ for which $\delta = \delta_k$ and we determine the new point

$P^* = (f_1(x^*), f_2(x^*))$ on the efficient frontier of problem (3) such that $f_1(x^*) \in [f_1(x^k), f_1(x^{k+1})]$. New points on the efficient frontier may be computed according to the *chord rule* or the *maximum error rule*, that is by solving the optimization problem (16) or the following problem (the maximum error rule problem)

$$\begin{aligned} & \min f_2(x) \\ & \text{s.t. } x \in X, \\ & f_1(x) = a_k, \end{aligned} \quad (28)$$

where a_k is the point of intersection of linear functions $u^{k-1}(\cdot)$ and $u^{k+1}(\cdot)$. Note that if we construct the lower bound due to definition (15), then the chord rule problem (16) has been already solved.

After the determination of the new point, we rebuild

the set P of given points on efficient frontier due to following equality

$$P_i = \begin{cases} P_i & \text{for } i < k \\ P^* & \text{for } i = k \\ P_{i-1} & \text{for } i > k, \end{cases} \quad (29)$$

then we construct new upper and lower bounds and repeat the procedure until we obtain an error δ smaller than the prescribed accuracy.

The next lemma describes the relation between the approximation bounds of the efficient frontiers of problems (1) and (3).

Lemma 2

Let $l(\cdot)$ and $u(\cdot)$ be the lower and upper approximation bounds of the efficient frontier of problem (3) built due to the definitions (8) and (9) or (8) and (15), then $\bar{l}(\cdot) = h \circ l \circ h^{-1}(\cdot)$ and $\bar{u}(\cdot) = h \circ u \circ h^{-1}(\cdot)$ are the lower and upper approximation bounds of the efficient frontier of problem (1).

Moreover, the following inequality is satisfied

$$\bar{u}(h_1(x)) - \bar{l}(h_1(x)) \leq A[u(f_1(x)) - l(f_1(x))] \quad (30)$$

for all the efficient solutions $x \in X_E$ and $A = h_2(x^1)$.

Proof: See Appendix.

From Lemma 2 follows that in order to obtain the Maximum error between upper and lower approximation bounds $\bar{l}(\cdot)$ and $\bar{u}(\cdot)$ of problem (1) smaller than or equal to the accuracy parameter ϵ , we need to build the approximation bounds $l(\cdot)$ and $u(\cdot)$ of problem (3) for which the Maximum error is smaller than or equal to $\frac{\epsilon}{A}$.

3.5. Algorithms

We present two algorithms described in this section.

3.5.1. The Simple Triangle Algorithm (STA)

Input: Introduce an accuracy parameter $\epsilon > 0$ and an initial set of points on the efficient frontier $P = \{P_1, P_2, P_3\}$.

Step 1. Calculate lower and upper bounds $l(\cdot)$, $u(\cdot)$ and error δ . Check if $\delta > \epsilon$, then go to Step 2, otherwise stop.

Step 2. Choose interval $[f_1(x^k), f_1(x^{k+1})]$ for which the maximum error is achieved. Solve the quadratic problem (16) or (28) to obtain the new point P^* . Update set P , lower and upper bounds $l(\cdot)$, $u(\cdot)$ and error δ . Go to Step 3.

Step 3. Check if $\delta > \epsilon$, then go to Step 2, otherwise stop.

3.5.2. The Trapezium Algorithm (TA)

Input: Introduce an accuracy parameter $\epsilon > 0$ and an

initial set of points on the efficient frontier $P = \{P_1, P_2\}$.

Step 1. Solve problem (16) and calculate lower and upper bounds $l_*(\cdot)$, $u(\cdot)$ and error δ . Check if $\delta > \epsilon$, then go to Step 2, otherwise stop.

Step 2. Choose interval $[f_1(x^k), f_1(x^{k+1})]$ for which the maximum error is achieved. The new point $P^* = (f_1(y^k), f_2(y^k))$. Update set P , solve problem (13), calculate lower and upper bounds $l_*(\cdot)$, $u(\cdot)$ and error δ . Go to Step 3.

Step 3. Check if $\delta > \epsilon$, then go to Step 2, otherwise stop.

The geometric illustration of STA and TA is given in **Figure 1** and **Figure 2**, respectively. In **Figure 1(a)**, there is an illustration of an efficient frontier and the corresponding lower and upper bounds determined by three initial points

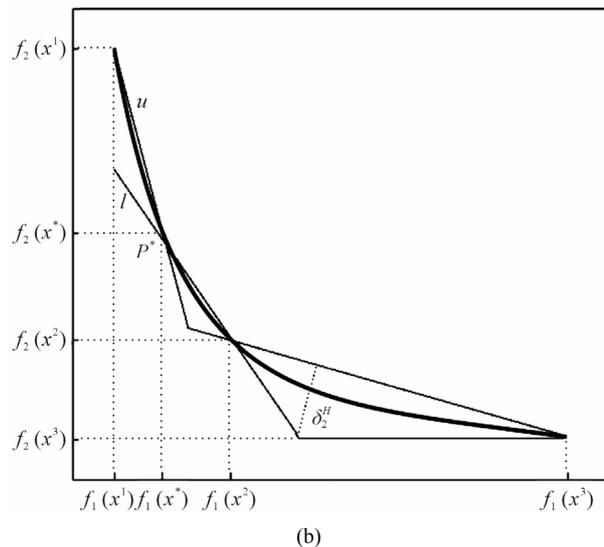
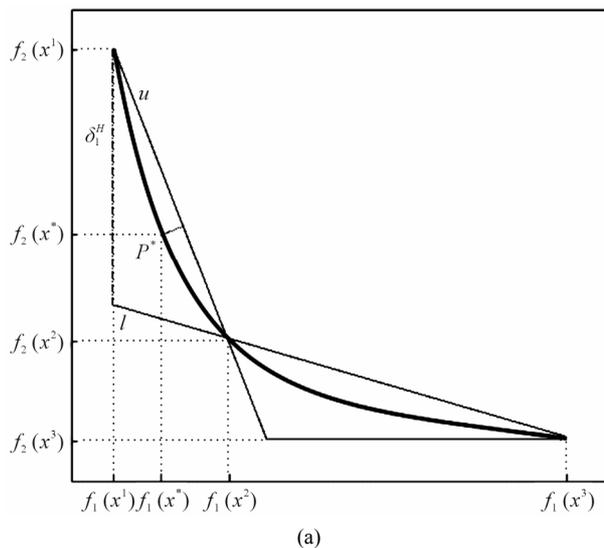


Figure 1. Lower and upper bounds built due to STA with the chord rule.

$$(f_1(x^1), f_2(x^1)), (f_1(x^2), f_2(x^2)), (f_1(x^3), f_2(x^3)).$$

Here one can observe that in the interval $[f_1(x^1), f_1(x^2)]$ the Hausdorff measure δ_1^H is the greatest. It means that we have to introduce a new point $P^* = (f_1(x^*), f_2(x^*))$ into the efficient frontier and determine new corresponding lower and upper bounds, what is illustrated in **Figure 1(b)**. Similar considerations are illustrated in **Figures 2(a)** and **2(b)**.

In **Figure 3** and **Figure 4** we see the illustration of STA and TA which lower bounds built due to definitions (9), (12) and (15), (19), respectively.

In Section 4 we study the convergence of described algorithms.

4. Convergence of the Algorithms

In this section we present the convergence results of presented algorithms based on proofs given in Rote [9]

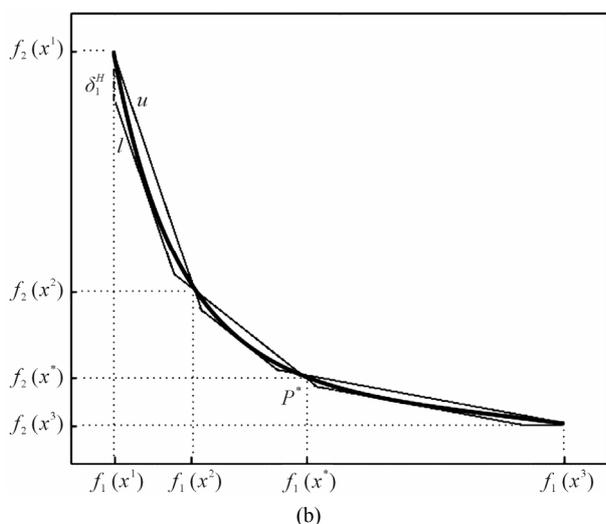
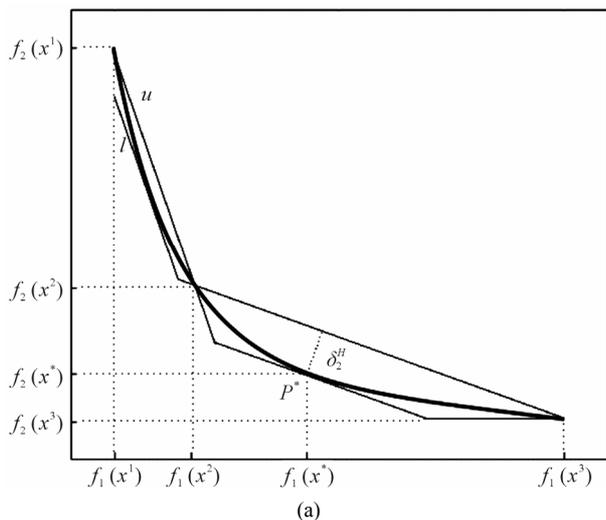


Figure 2. Lower and upper bounds built due to TA.

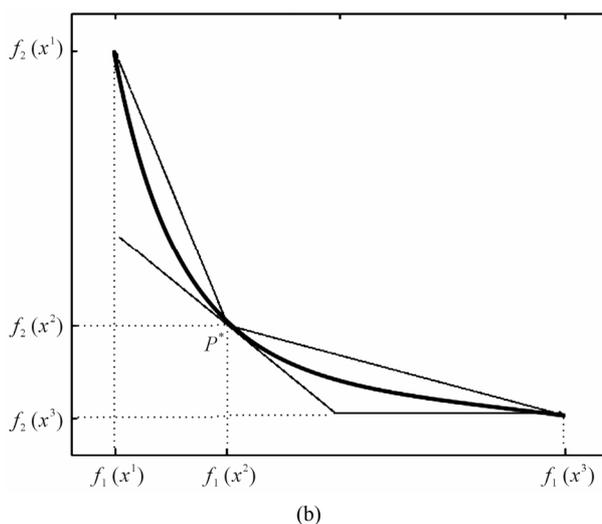
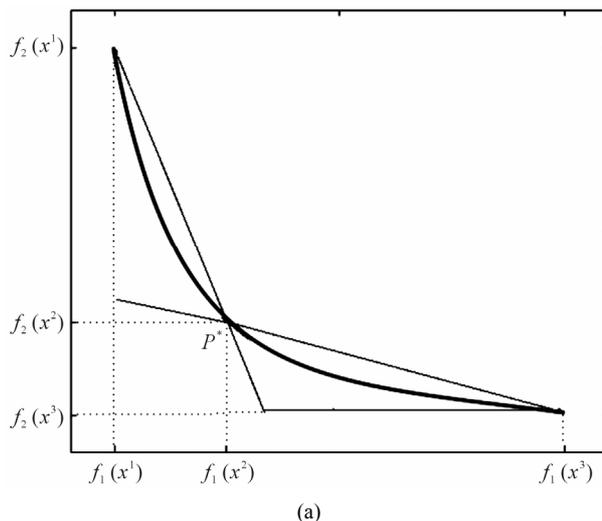


Figure 3. Lower bounds in STA built due to definitions (9) and (12) presented in (a) and (b), respectively.

and Yang and Goh [8]. First, we formulate two following remarks, which show the relation between the considered error measures.

Remark 1

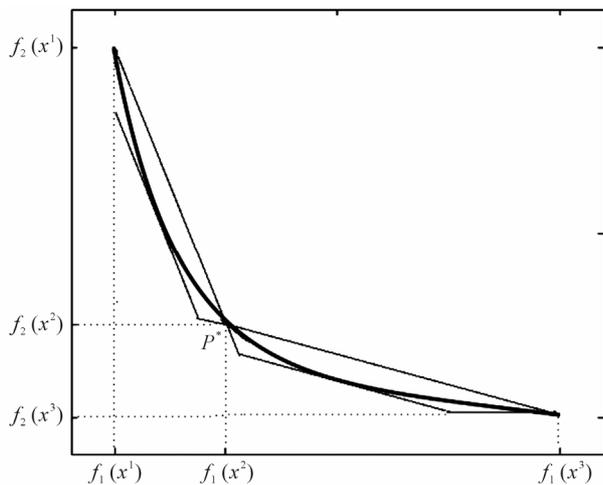
Suppose that the lower and upper approximation bounds on the interval $[f_1(x^k), f_1(x^{k+1})]$ have been built according to the definitions (8) and (9) or (15), then we have

$$\delta_k^M = \delta_k^H \sqrt{1 + \left(\frac{f_2(x^k) - f_2(x^{k+1})}{f_1(x^{k+1}) - f_1(x^k)} \right)^2}. \tag{31}$$

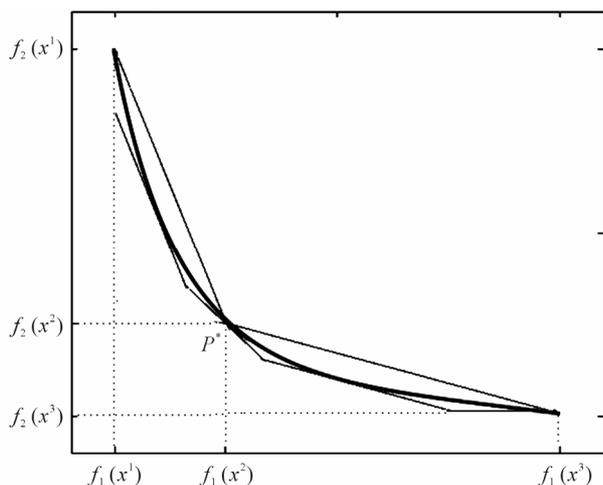
Moreover,

$$\delta_k^M \leq \delta_k^H \left(1 + \frac{f_2(x^k) - f_2(x^{k+1})}{f_1(x^{k+1}) - f_1(x^k)} \right). \tag{32}$$

See **Figure 5**.



(a)



(b)

Figure 4. Lower bounds in TA built due to definitions (15) and (19) presented in (a) and (b), respectively.

Remark 2

Suppose that the lower and upper approximation bounds on the interval $[f_1(x^k), f_1(x^{k+1})]$ have been build according to the definitions (8) and (9) or (15), then we have

$$\delta_k^U \leq \delta_k^H \sqrt{(f_1(x^k) - f_1(x^{k+1}))^2 + (f_2(x^k) - f_2(x^{k+1}))^2}. \tag{33}$$

Now, we suppose that the efficient frontier of problem (3) is given as a convex function $g : [a, b] \rightarrow R$ and the one-sided derivatives $g^+(a)$ and $g^-(b)$ have been evaluated.

The following theorem based on Remark 3 and Theorem 1 in [8], Theorem 2 in [9] and Remark 1 and Remark 2 shows the quadratic convergence property of TA.

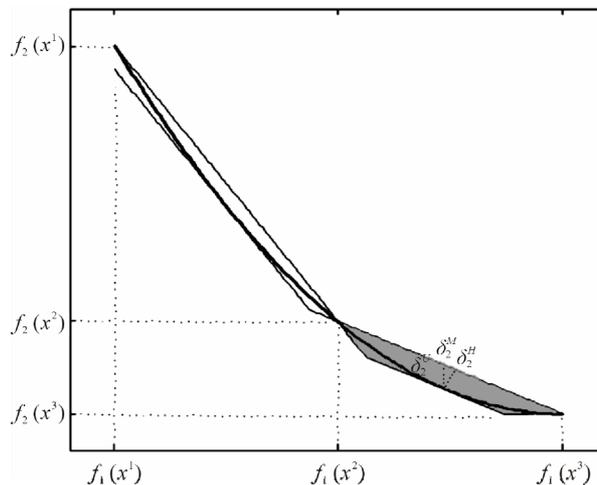


Figure 5. Illustration of error measures defined by (25)-(27).

Theorem 1

Let $L = b - a$ and $\Delta = g^-(b) - g^+(a)$ and suppose that the point $(c, g(c))$ was chosen to satisfy the following inequality

$$c \leq a + \epsilon \frac{b - a}{g(a) - g(b)}. \tag{34}$$

The number H of optimization problems (16) which have to be solved in order to obtain the Hausdorff distance between upper and lower bounds in TA smaller than or equal to ϵ satisfies the following inequality

$$H \leq \max \left\{ 2 \left\lceil \sqrt{\frac{L\Delta}{\epsilon}} \right\rceil - 1, 3 \right\}. \tag{35}$$

The third point $(c, g(c))$ on the efficient frontier has to be chosen if we want to avoid the problems with the leftmost interval in which the Hausdorff distance between the approximation bounds is equal to the maximum error measure between them. The explanation how to determine a point with property (34) is given in the proof of Theorem 2.

Corollary 1

Let $L = b - a$, $\Delta = g^-(b) - g^+(a)$. The number M of optimization problems (16) which have to be solved in order to obtain the Maximum error between upper and lower bounds in TA smaller than or equal to ϵ satisfies the following inequality

$$M \leq \max \left\{ 2 \left\lceil \sqrt{\frac{L\Delta}{\epsilon}} (1 + \gamma) \right\rceil - 1, 3 \right\}, \tag{36}$$

where $\gamma = \frac{g(b) - g(a)}{a - b}$.

The number U of optimization problems (16) which have to be solved in order to obtain the Uncertainty area error between upper and lower bounds in TA smaller than or equal to ϵ satisfies the following inequality

$$U \leq \max \left\{ 2 \left\lceil \sqrt{\frac{L\Delta\eta}{\epsilon}} \right\rceil - 1, 3 \right\}, \tag{37}$$

where $\eta = \sqrt{(g(a) - g(b))^2 + (a - b)^2}$.

Yang and Goh [8] noticed that the right directional derivative $g^+(a)$ may be close to $-\infty$, that is why using the fact that the Hausdorff distance is invariant under rotation it is better to consider the efficient frontier rotated by $\frac{\pi}{4}$ with the modified directional derivatives $\bar{g}^+(\bar{a})$ and $\bar{g}^-(\bar{b})$ and with $\bar{L} = \bar{b} - \bar{a}$ as the projective distance of the segment between points $(a, g(a))$ and $(b, g(b))$ onto the line $m(x) = -x$.

Also for STA we may find the upper bound for the number of optimization problems (16) or (28) which have to be solved. First, let us formulate the following lemma.

Lemma 3

Suppose that the convex function $g: [a, b] \rightarrow R$ is approximated from below by two linear functions $l_{ac}(\cdot)$ and $l_{cb}(\cdot)$ such that $l_{ac}(a) = g(a)$, $l_{cb}(b) = g(b)$ and $l_{ac}(c) = l_{cb}(c)$ for some $c \in (a, b)$. Let $m_{ac}(\cdot)$ and $m_{cb}(\cdot)$ be the lines which intersect the points $(a, g(a))$, $(c, g(c))$ and $(c, g(c))$, $(b, g(b))$, respectively (see **Figure 6**). If we denote by $\tan \alpha$, $\tan \beta$, $\tan \gamma$, $\tan \delta$ the slopes of lines $l_{ac}(\cdot)$, $l_{cb}(\cdot)$, $m_{ac}(\cdot)$, $m_{cb}(\cdot)$, respectively, then the following inequality is satisfied

$$\Delta_1 L_1 + \Delta_2 L_2 \leq \Delta L, \tag{38}$$

where $\Delta_1 = \tan \delta - \tan \alpha$, $\Delta_2 = \tan \beta - \tan \gamma$, $\Delta = \tan \beta - \tan \alpha$, $L_1 = c - a$, $L_2 = b - c$ and $L = b - a$.

Proof: See Appendix.

Note that Δ_1 and Δ_2 are the differences of the slopes of lower approximation functions computed according to

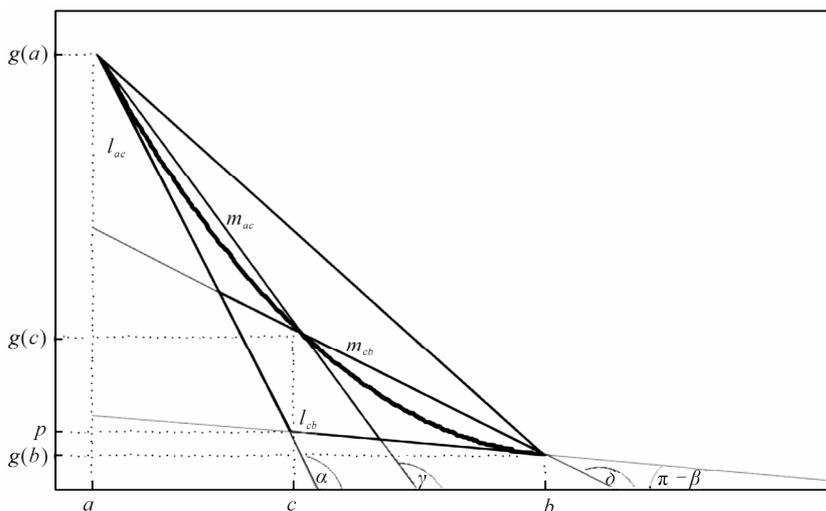


Figure 6. Illustration of functions and angels considered in Lemma 3.

STA for interval $[a, c]$ and $[c, b]$, respectively, and L_1 and L_2 are the lengths of these intervals.

The next theorem based on Lemma 5, Theorem 2 from [9] and Lemma 3 establishes the linear convergence property of STA.

Theorem 2

Let $(a, g(a))$, $(c, g(c))$ and $(b, g(b))$ are three initial points which are necessary to start STA and suppose that $(c, g(c))$ was chosen to satisfy the following inequality

$$c \leq a + \epsilon \frac{b - a}{g(a) - g(b)}. \tag{39}$$

Let $L = b - a$ and let $\Delta = -g^+(a)$, then the number H of convex optimization problems ((16) or (28)), which have to be solved in order to make the Hausdorff distance between upper and lower bound in STA with the chord rule or the maximum error rule smaller than or equal to ϵ satisfies the following inequality

$$H \leq \max \left\{ \left\lceil \frac{L\Delta}{\epsilon} \right\rceil - 4, 0 \right\}. \tag{40}$$

Proof: See Appendix.

Corollary 2

Let $(a, g(a))$, $(c, g(c))$ and $(b, g(b))$ are three initial points that are necessary to start STA and suppose that $(c, g(c))$ was chosen to satisfy the inequality (39). Let $L = b - a$ and let $\Delta = -g^+(a)$. The number M of optimization problems (16) which have to be solved in order to obtain the Maximum error between upper and lower bounds in TA smaller than or equal to ϵ satisfies the following inequality

$$M \leq \max \left\{ \left\lceil \frac{L\Delta}{\epsilon} (1 + \gamma) \right\rceil - 4, 0 \right\}, \tag{41}$$

where $\gamma = \frac{g(b) - g(a)}{a - b}$. The number U of optimization problems (16) which have to be solved in order to obtain the Uncertainty area error between upper and lower bounds in TA smaller than or equal to ϵ satisfies the following inequality

$$U \leq \max \left\{ \left\lceil \frac{L\Delta\eta}{\epsilon} \right\rceil - 4, 0 \right\}, \tag{42}$$

where $\eta = \sqrt{(g(a) - g(b))^2 + (a - b)^2}$.

Remark 3

Note that this theorem is true for every convex function $g : [a, b] \rightarrow R$ such that the one-sided derivative $g^+(a)$ has been evaluated. If we do not have the derivative information using TA with maximum error rule gives us the linear convergence property of this procedure.

The following theorem by Rote [9] establishes the quadratic convergence property of STA with the modified lower bound as in (12).

Theorem 3

Let $(a, g(a))$, $(c, g(c))$ and $(b, g(b))$ are three initial points which are necessary to start STA and suppose that $(c, g(c))$ was chosen to satisfy the following inequality

$$c \leq a + \epsilon \frac{b - a}{g(a) - g(b)}. \tag{43}$$

Let $L = b - a$ and let $\Delta = -g^+(a)$, then the number H of convex optimization problems (16), which have to be solved in order to make the Hausdorff distance between upper and lower bound in STA and TA with the modified lower bounds and the chord rule smaller than or equal to ϵ satisfies the following inequality

$$H \leq \max \left\{ \left\lceil \frac{\sqrt{L\Delta}}{\epsilon} \right\rceil - 2, 0 \right\}. \tag{44}$$

5. Examples

In this section we define the stochastic minimum cost flow problem with the moment bicriterion and present two numerical examples, which illustrate algorithm presented in Section 3. Similar to the classic bicriteria network cost flow problem, we consider the directed network G with n nodes and m arcs with the node-arc incidence matrix $D \in R^{n \times m}$, the net outflow vector $b \in R^n$ and the capacity bounds $c_l, c_u \in R_+$. Suppose that the cost per unit of flow through the arc $i \in \{1, \dots, m\}$ is described by the random variable $C_i : \Omega \rightarrow R_+$ and assume that each variable C_i has positive expected value $E[C_i] = c_i$ and let C be a random vector such that $C = [C_1, \dots, C_m]^T$. The stochastic minimum cost flow

problem with the moment bicriterion (SMCFP) is defined as follows

$$\begin{aligned} & \min [E[C^T x], E[f(C, x)]]^T \\ & \text{s.t. } Dx = b, \\ & \quad c_l \leq x \leq c_u, \end{aligned} \tag{45}$$

where $f : R_+^m \times R_+^m \rightarrow R_+$ is the continuous function such that $f(a, \cdot) : R_+^m \rightarrow R_+$ is convex for every $a \in R_+^m$.

Since $x \rightarrow E[C^T x]$ and $x \rightarrow E[f(C, x)]$ are linear and convex functions, respectively, then using Lemma 1 we find that the efficient frontier of problem (45) is a convex curve in R^2 .

The following two examples include the comparison of the results obtained by STA, TA with Yang and Goh's method [8] and the Siem *et al.*'s algorithm [7].

Example 1

We consider the simple stochastic minimum cost flow problem

$$\min [E[C^T x], f(x)]^T \tag{46}$$

s.t.

$$\begin{aligned} 2 \leq x_1 \leq 6, \quad 2 \leq x_2 \leq 6, \quad 0 \leq x_3 \leq 4, \\ 2 \leq x_4 \leq 6, \quad 2 \leq x_5 \leq 6, \quad x_1 + x_2 = 8, \\ x_1 = x_3 + x_4, \quad x_2 + x_3 = x_5, \quad x_4 + x_5 = 8, \end{aligned}$$

where $E[C_1] = 3$, $E[C_2] = 6$, $E[C_3] = 1$, $E[C_4] = 4$, $E[C_5] = 2$ and

$$f(x) = e^{\frac{x_1}{2}} + e^{\frac{x_2}{6}} + e^{x_3} + e^{\frac{x_4}{2}} + e^{\frac{x_5}{5}}. \tag{47}$$

The vectors $x^1 = (6, 2, 4, 2, 6)$ and $x^3 = (2, 6, 0, 2, 6)$ are the lexicographical minima due to the first and the second criterion of problem (46) and

$$P_1 = (f_1(x^1), f_2(x^1)) = (54, 82.12)$$

and

$$P_3 = (f_1(x^3), f_2(x^3)) = (62, 12.47)$$

are corresponding to these vectors points on the efficient frontier. To avoid the problem with the leftmost interval we have taken $P_2 = (54.15, 74.63)$ to be the third point of the efficient frontier necessary to start STA and TA.

The results' comparison of subsequent calls of TA and Yang and Goh's algorithm (YG) and the values of the Uncertainty area measure δ^U after each step of TA with the lower bounds defined in (19) (TA1) are presented in **Table 1**. The results of STA, when new points are computed according to the maximum error rule in common with the Maximum error measure (STAM) and when new points are computed according to the chord rule in common with the Hausdorff measure with the lower bounds defined by the Equation (9) (STAH) and

Table 1. The results of subsequent calls of TA and Yang and Goh’s method (YG) for the problem described in example 1.

Step	TA				YG		
	δ^H	δ^M	δ^U	δ^U (TA1)	δ^H	δ^M	δ^U
1	2.995	23.905	93.828	93.828	23.905	23.905	180.927
2	1.376	9.638	25.659	23.644	5.308	9.638	31.003
3	0.505	9.638	23.591	18.781	1.497	9.638	16.269
4	0.490	2.069	3.211	2.975	1.876	2.069	3.731
5	0.251	2.069	3.134	2.641	1.876	2.069	3.428
6	0.152	2.069	2.805	2.641	1.876	2.069	2.398
7	0.133	2.069	2.746	2.114	0.397	2.069	1.295
8	0.096	2.069	2.746	2.114	0.397	2.069	1.295
9	0.070	2.069	0.402	0.371	0.397	2.069	1.295
10	0.064	2.069	0.402	0.371	0.397	2.069	1.295

Table 2. The results of subsequent calls of STA with the chord rule (STAH, STAH1) and the maximum error rule (STAM) and Siem et al.’s method with interval bisection rule (SB) for the problem described in example 1.

Step	STAH			STAH1		
	δ^H	δ^M	δ^U	δ^H	δ^M	δ^U
1	6.553	52.303	205.288	6.553	52.303	205.288
2	4.163	24.116	40.901	2.902	10.540	28.462
3	1.744	21.671	26.547	1.170	4.556	4.259
4	1.225	21.671	26.547	0.401	4.556	4.259
5	0.760	8.633	5.064	0.367	4.556	4.259
6	0.648	8.633	4.084	0.167	4.556	3.610
7	0.485	8.633	4.084	0.126	4.556	2.727
8	0.305	8.107	3.567	0.122	2.543	0.908
9	0.269	8.107	3.567	0.115	2.543	0.908
10	0.236	2.836	0.676	0.107	2.543	0.908

Step	STAM			SB		
	δ^H	δ^M	δ^U	δ^H	δ^M	δ^U
1	6.553	52.303	205.288	3.321	49.630	99.260
2	5.283	24.961	82.440	2.389	31.580	31.580
3	3.335	12.086	32.670	2.389	21.090	12.878
4	1.952	11.916	8.199	2.389	8.710	12.878
5	1.952	4.728	8.199	2.389	6.439	12.878
6	0.912	4.460	2.925	0.934	4.230	4.230
7	0.912	3.122	2.925	0.934	3.057	2.126
8	0.912	3.019	2.925	0.934	2.814	2.126
9	0.912	2.274	1.838	0.934	2.730	2.126
10	0.912	1.715	1.838	0.934	2.126	2.126

(12) (STAH1) are summarized in **Table 2**. In the case of Yang and Goh’s method the errors are measured in the intervals set by the breakpoints of the upper bound. **Table 2** includes also the results of the method described in [7], which uses the interval bisection method of the computing new points with the Maximum error measure. After each step of algorithm we present the maximum values of three error measures: the Maximum error, the Hausdorff distance and the Uncertainty area. As we can notice TA and TA1 performs better than other algorithms giving in each step the smallest values of the Hausdorff distance measure and the Uncertainty area measure. Moreover, from **Table 2** we can conclude that STA with the chord rule and the Hausdorff distance gives smaller values of the Hausdorff measure in each step than to two other algorithms, which give comparable results.

Example 2

We consider the stochastic minimum cost flow problem in the network with 12 nodes and 17 arcs. We would like to minimize the mean value of the total cost of the flow through the network and the second moment of total cost, that is we solve the following problem

$$\begin{aligned} & \min \left[E[C^T x], E[(C^T x)^2] \right]^T \\ & \text{s.t. } Dx = b, \\ & \quad c_l \leq x \leq c_u, \end{aligned} \tag{48}$$

where $D \in R^{12 \times 17}$, $b^T = [10, 0, \dots, 0, -10]$, $c_l^T = [0, \dots, 0]$ and $c_u^T = [6, 6, \dots, 6, 6]$ and $C = [C_1, \dots, C_{17}]^T : \Omega \rightarrow R_+^{17}$ is the random cost vector such that C_i and C_j are mutually independent for all $i, j \in A$ such that $i \neq j$. We have chosen the values of $E[C_1], \dots, E[C_{17}]$ from the interval $[0, 2]$ and the values of the second moments $E[C_1^2], \dots, E[C_{17}^2]$ from the interval $[1, 3]$. The points on the efficient frontier corresponding to the lexicographical minima of problem (48) and the second criterion are $P_1 = (62.244, 6072.87)$ and

$P_2 = (65.293, 5145.63)$. In this example only TA and the Yang and Goh’s method are considered, since from Example 1. follows that these two algorithms work faster in comparison to STA and Siem et al.’s method. **Table 3** includes the comparison of TA with the method presented in [8]. After each step of the considered methods we present the values of Hausdorff distance, Maximum error and Uncertainty area measure and a new evaluated point. As we can notice TA performs better in comparison to Yang and Goh’s algorithm giving in each step the smallest value of the Hausdorff distance between upper and lower approximation bounds.

6. Conclusions

Two sandwich algorithms (the Simple Triangle Algorithm

Table 3. The results of subsequent calls of TA and Yang and Goh's method (YG) for the problem described in example 2.

Step	$(f_1(x^*), f_2(x^*))$	TA			YG		
		δ^H	δ^M	δ^U	δ^H	δ^M	δ^U
1		1.507	458.259	1052.1	1.507	458.259	1397.42
2	(63.410, 5259.94)	0.598	117.196	75.559	36.287	117.196	129.246
3	(64.194, 5176.09)	0.282	117.196	70.439	14.101	117.196	126.457
4	(64.733, 5153.34)	0.168	117.196	68.847	14.101	117.196	126.457
5	(62.824, 5551.23)	0.141	29.355	8.614	28.735	29.355	17.012
6	(65.013, 5147.54)	0.134	29.355	8.541	28.735	29.355	17.012
7	(63.584, 5227.26)	0.071	29.355	8.541	28.735	29.355	17.012
8	(65.1528, 5146.1)	0.058	29.355	8.534	28.735	29.355	17.012
9	(63.112, 5379.36)	0.053	29.355	8.523	7.254	29.355	15.961
10	(64.438, 5163.53)	0.042	29.355	8.523	7.254	29.355	15.961

and the Trapezium Algorithm) for the approximation of the efficient frontier of the generalized bicriteria minimum cost flow problem have been described. Both methods have been applied in the field of the stochastic minimum cost flow problem with the moment bicriterion.

The Simple Triangle Algorithm uses the lower bound proposed by Siem *et al.* in [7] with the maximum error rule or the chord rule, what causes faster decrease of the Maximum error measure and the Hausdorff distance measure and, as a result, reduces the number of steps of algorithm in comparison to the Siem *et al.*'s method. We have established the linear convergence property of this algorithm with both partition rules. If the lower bound in the Simple Triangle Algorithm is defined as in [9], according to the definition (12) and new points of the efficient frontier are computed due to the chord rule then we have the quadratic convergence property of the algorithm.

From the numerical examples follows that the Trapezium Algorithm performs better in comparison to all of the mentioned derivative free algorithms (Siem *et al.*'s method, Yang and Goh's method) and gives in each step the smallest values of the Hausdorff distance measure between lower and upper bound. It also works faster than Rote's triangle algorithm. Moreover, Trapezium Algorithm may be used for approximation of any convex function without the assumption that the change of the direction of the tangents of this function is less than or equal to $\frac{\pi}{2}$, see [8]. The quadratic convergence property of the Trapezium Algorithm has been established.

For further research we are interested in the construc-

tion of a method for solving the stochastic minimum cost flow problem with the moment multicriterion.

REFERENCES

- [1] A. Sedeno-Noda and C. Gonzalez-Martin, "The Biobjective Minimum Cost Flow Problem," *European Journal of Operational Research*, Vol. 124, No. 3, 2000, pp. 591-600. [doi:10.1016/S0377-2217\(99\)00191-5](https://doi.org/10.1016/S0377-2217(99)00191-5)
- [2] X. Q. Yang and C. J. Goh, "Analytic Efficient Solution Set for Multi-Criteria Quadratic Programs," *European Journal of Operational Research*, Vol. 92, No. 1, 1996, pp. 166-181. [doi:10.1016/0377-2217\(95\)00040-2](https://doi.org/10.1016/0377-2217(95)00040-2)
- [3] G. Ruhe, "Complexity Results for Multicriterial and Parametric Network Flows Using a Pathological Graph of Zadeh," *Zeitschrift für Operations Research*, Vol. 32, No. 1, 1988, pp. 9-27. [doi:10.1007/BF01920568](https://doi.org/10.1007/BF01920568)
- [4] N. Zadeh, "A Bad Network for the Simplex Method and Other Minimum Cost Flow Algorithms," *Mathematical Programming*, Vol. 5, No. 1, 1973, pp. 255-266. [doi:10.1007/BF01580132](https://doi.org/10.1007/BF01580132)
- [5] R. E. Burkard, H. W. Hamacher and G. Rote, "Sandwich Approximation of Univariate Convex Functions with an Applications to Separable Convex Programming," *Naval Research Logistics*, Vol. 38, 1991, pp. 911-924.
- [6] B. Fruhwirth, R. E. Burkard and G. Rote, "Approximation of Convex Curves with Application to the Bi-Criteria Minimum Cost Flow Problem," *European Journal of Operational Research*, Vol. 42, No. 3, 1989, pp. 326-338. [doi:10.1016/0377-2217\(89\)90443-8](https://doi.org/10.1016/0377-2217(89)90443-8)
- [7] A. Y. D. Siem, D. den Hertog and A. L. Hoffmann, "A Method for Approximating Univariate Convex Functions Using Only Function Value Evaluations," *Center Discussion Paper*, Vol. 67, 2007, pp. 1-26.
- [8] X. Q. Yang and C. J. Goh, "A Method for Convex Curve

Approximation,” *European Journal of Operational Research*, Vol. 97, No. 1, 1997, pp. 205-212.
[doi:10.1016/0377-2217\(95\)00368-1](https://doi.org/10.1016/0377-2217(95)00368-1)

rithm for Approximating Convex Functions,” *Computing*, Vol. 48, No. 3-4, 1992, pp. 337-361.
[doi:10.1007/BF02238642](https://doi.org/10.1007/BF02238642)

[9] G. Rote, “The Convergence Rate of the Sandwich Algo-

Appendix

We present the proofs of Lemma 1, Lemma 2, Lemma 3 and Theorem 2.

Proof of Lemma 1

Let $f_1(\cdot) = h^{-1} \circ h_1(\cdot)$, $f_2(\cdot) = h^{-1} \circ h_2(\cdot)$ and let $P_k = (f_1(x^k), f_2(x^k))$ for $k=1,2$ be two given points on the efficient frontier of problem (3) and let $\lambda \in [0,1]$.

If $f_1(x^1) = f_1(x^2)$ then for all b such that $f_2(x^1) \geq b \geq f_2(x^2)$ the point $(f_1(x^k), b)$ lies also on the efficient frontier of problem (3).

Suppose now that $c^T x^1 = f_1(x^1) \neq f_1(x^2) = c^T x^2$. Note that for

$$C = \{x \in X : c^T x = \lambda c^T x^1 + (1-\lambda)c^T x^2\}, \tag{49}$$

$$B_1 = \{y \in X : c^T y = c^T x^1\} \tag{50}$$

and

$$B_2 = \{z \in X : c^T z = c^T x^2\}, \tag{51}$$

we have $\lambda B_1 + (1-\lambda)B_2 \subseteq C$, what yields

$$\min_{x \in C} f_2(x) \leq \min_{y \in B_1, z \in B_2} f_2(\lambda y + (1-\lambda)z). \tag{52}$$

Due to the convexity of the function f_2 we have

$$\begin{aligned} \min_{x \in C} f_2(x) &\leq \min_{y \in B_1, z \in B_2} f_2(\lambda y + (1-\lambda)z) \\ &\leq \min_{y \in B_1} \lambda f_2(y) + \min_{z \in B_2} (1-\lambda) f_2(z), \end{aligned} \tag{53}$$

what proves the convexity of the efficient frontier of problem (3). ■

Proof of Lemma 2

Let $x \in X_E$, then we have

$$l(f_1(x)) \leq f_2(x) \leq u(f_1(x)), \tag{54}$$

what is equivalent to the following inequality

$$l(h^{-1}(h_1(x))) \leq h^{-1}(h_2(x)) \leq u(h^{-1}(h_1(x))) \tag{55}$$

and from the fact that h is increasing we have

$$\begin{aligned} \bar{l}(h_1(x)) &= h(l(h^{-1}(h_1(x)))) \leq h_2(x) \\ &\leq h(u(h^{-1}(h_1(x)))) = \bar{u}(h_1(x)). \end{aligned} \tag{56}$$

Moreover,

$$\begin{aligned} &\bar{u}(h_1(x)) - \bar{l}(h_1(x)) \\ &= h(u(h^{-1}(h_1(x)))) - h(l(h^{-1}(h_1(x)))) \\ &= h(u(f_1(x))) - h(l(f_1(x))) \\ &\leq \max_{y \in [l(f_1(x^r)), u(f_1(x^1))]} h(y) [u(f_1(x)) - l(f_1(x))] \tag{57} \\ &= \max_{y \in [f_2(x^r), f_2(x^1)]} h(y) [u(f_1(x)) - l(f_1(x))] \\ &= h(f_2(x^1)) [u(f_1(x)) - l(f_1(x))] \\ &= h_2(x^1) [u(f_1(x)) - l(f_1(x))]. \end{aligned}$$

Proof of Lemma 3

Let $c = \lambda a + (1-\lambda)b$, where $\lambda \in (0,1)$ and let $l_{ac}(c) = l_{cb}(c) = p$. Without loss of generality we may assume that $g(a) \geq g(b)$, then we have $g(a) \geq g(c)$.

From the equalities

$$\begin{aligned} \tan \alpha &= \frac{p-g(a)}{c-a}, \tan \beta = \frac{g(b)-p}{b-c}, \\ \tan \gamma &= \frac{g(c)-g(a)}{c-a}, \tan \delta = \frac{g(b)-g(c)}{b-c} \end{aligned} \tag{58}$$

follows

$$\Delta = \frac{g(b)-p}{\lambda(b-a)} - \frac{p-g(a)}{(1-\lambda)(b-a)}, L = b-a, \tag{59}$$

$$\Delta_1 = \frac{g(b)-g(c)}{\lambda(b-a)} - \frac{p-g(a)}{(1-\lambda)(b-a)}, L_1 = (1-\lambda)(b-a) \tag{60}$$

$$\Delta_2 = \frac{g(b)-p}{\lambda(b-a)} - \frac{g(a)-g(c)}{(1-\lambda)(b-a)} \text{ and } L_2 = \lambda(b-a). \tag{61}$$

Now, it is easy to show that inequality (38) is equivalent to

$$(2\lambda^2 - 2\lambda + 1)(g(c) - p) \geq 0. \tag{62}$$

Using the fact that $g(c) \geq p$ we complete the proof. ■

Proof of Theorem 2

Suppose that we have found point $(c, f(c))$ with property (39). It is possible to find such a point as a solution of a convex programming problem with one additional constraint similar to problem (6). From condition (39) follows that $\delta_1^M \leq \epsilon$ and $\delta_1^H \leq \epsilon$.

Now, if we consider the interval $[c, b]$, then we have $c-b \leq L$ and

$$\frac{f(c)-f(b)}{b-c} \leq \Delta.$$

Similar to [9] we prove the theorem by induction on number $N = N\left(\frac{L\Delta}{\epsilon}\right)$.

The induction basis, $N\left(\frac{L\Delta}{\epsilon}\right) = 0$, is equivalent to

Lemma 1 from [5], which also holds for lower approximation function built according to definition (9).

Suppose that $N \geq 1$. If after one step of STA $\delta \leq \epsilon$, then we have had only one additional evaluation and the thesis is true.

In the other case, let $d \in [c, b]$ be the new computed point and let $L_1 = d-c$ and $L_2 = b-d$. Let Δ_1 and Δ_2 denote the slope differences of the linear functions

building the lower bounds in the interval $[c, d]$ and $[d, b]$, respectively. We can assume without loss of generality that the error δ exceeds ϵ in the right subinterval. Lemma 5 and Lemma 1 from [9] used for the lower approximation bounds built due to definition (9) give the following inequalities

$$\frac{L_1\Delta_1}{\epsilon} > 1 \tag{63}$$

and

$$\frac{L_2\Delta_2}{\epsilon} > 4. \tag{64}$$

Lemma 3 and inequality (63) gives

$$\frac{L_2\Delta_2}{\epsilon} < \frac{L\Delta}{\epsilon}. \tag{65}$$

Similarly, Lemma 3 and inequality (64) gives

$$\frac{L_1\Delta_1}{\epsilon} < \frac{L\Delta}{\epsilon}. \tag{66}$$

From (65) and (66) we have

$$N\left(\frac{L_1\Delta_1}{\epsilon}\right) < N\left(\frac{L\Delta}{\epsilon}\right) \text{ and } N\left(\frac{L_2\Delta_2}{\epsilon}\right) < N\left(\frac{L\Delta}{\epsilon}\right). \tag{67}$$

Now the induction hypothesis can be applied for $N\left(\frac{L_1\Delta_1}{\epsilon}\right)$ and $N\left(\frac{L_2\Delta_2}{\epsilon}\right)$. If $N\left(\frac{L_1\Delta_1}{\epsilon}\right) = 0$, then the theorem's thesis follows directly.

Otherwise, from Lemma 3 we have

$$\begin{aligned} & \max_{\Delta_1 L_1 + \Delta_2 L_2 \leq \Delta L, L_1 + L_2 = L, L_1 \Delta_1 > \epsilon, L_2 \Delta_2 > 4\epsilon} \left(1 + N\left(\frac{L_1\Delta_1}{\epsilon}\right) + N\left(\frac{L_2\Delta_2}{\epsilon}\right) \right) \\ &= \max_{\Delta_1 L_1 + \Delta_2 L_2 \leq \Delta L, L_1 + L_2 = L, L_1 \Delta_1 > \epsilon, L_2 \Delta_2 > 4\epsilon} \left(1 + \left\lceil \frac{L_1\Delta_1}{\epsilon} \right\rceil + \left\lceil \frac{L_2\Delta_2}{\epsilon} \right\rceil - 7 \right) \\ &\leq \left\lceil \frac{L\Delta}{\epsilon} \right\rceil - 6 \leq \left\lceil \frac{L\Delta}{\epsilon} \right\rceil = N\left(\frac{L\Delta}{\epsilon}\right), \end{aligned} \tag{68}$$

what ends the proof. ■