GLOBAL OPTIMIZATION USING SPACE FILLING CURVES

Michaela BAILOVA, Jiri BOUCHALA, Petr VODSTRCIL

Department of Applied Mathematics, Faculty of Electrical Engineering and Computer Science, VSB–Technical University of Ostrava, 17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

michaela.bailova@vsb.cz, jiri.bouchala@vsb.cz, petr.vodstrcil@vsb.cz

DOI: 10.15598/aeee.v15i2.2303

Abstract. The existence of space filling curves opens the way to reducing multivariate optimization problems to the minimization of univariate functions. In this paper, we analyze the Hölder continuity of space filling curves and exploit this property in the solution of global optimization problems. Subsequently, an algorithm for minimizing univariate Hölder continuous functions is presented and analyzed. It is shown that the algorithm computes the approximate minimum with the guaranteed precision. The algorithm is tested on some types of two-dimensional functions.

Keywords

Dimension reduction, global optimization, Hölder continuity, Lipschitz continuity, space filling curves.

1. Introduction

Many engineering problems lead to a multivariate global optimization. Such issue can be difficult to solve. An approach presented in this paper is to turn the problem into its one-dimensional equivalent. A way how to provide such simplification is to use a space filling curve.

This paper deals with the problem of searching a global minimum

\[ F(y^*) = \min_{y \in D} F(y), \]

and a global minimizer \( y^* \in D \), where \( D \) is an \( N \)-dimensional hypercube defined as follows

\[ D = \{ y \in \mathbb{R}^N, -\frac{1}{2} \leq y_j \leq \frac{1}{2}, 1 \leq j \leq N \}. \]

The objective function \( F \) is assumed to satisfy the Lipschitz condition with a constant \( L, 0 < L < \infty \).

The main idea of the paper is to turn the multivariate optimization problem into its one-dimensional equivalent, which can be solved using some techniques evolved for univariate optimization. One way how to do so is to develop a continuous correspondence \( y \) mapping a one-dimensional interval onto the hypercube. The problem Eq. (1) turns into the following one

\[ F(y^*) = F(y(x^*)) = \min_{x \in [0,1]} F(y(x)), \]

where \( x^* \in [0,1] \). A complete analysis is done to prove that a univariate algorithm based on Hölder continuity can be used to find an approximation of the optimal value of \( F \) over the domain \( D \). The performance of the algorithm is illustrated by numerical experiments.

2. Space Filling Curves

Main ideas in this section are inspired by [1] and [3].

Definition 1. A space filling curve is a single-valued continuous correspondence \( y \) mapping the unit interval \([0,1]\) onto the hypercube \( D \) from Eq. (2).

If \( y \) is a space filling curve, then

\[ F(y^*) = \min_{x \in [0,1]} F(y(x)). \]

Though the concept of a space filling curve is useful for the analysis of the algorithm, the effective computation is usually based on a continuous correspondence \( y_n \) mapping the unit interval only into \( D \). The following theorem gives some information about the optimal value of \( F \circ y_n \) using the quality of \( y_n \).

Theorem 1. Let \( (y_n) \), where \( y_n : [0,1] \to D \), be a sequence of curves such that

\[ \sup_{y \in D} \text{dist}(y_n([0,1]), y) =: \varepsilon_n \to 0, \]

\[ \text{dist}(y_n([0,1]), y) \rightarrow 0, \]
for \( n \to \infty \) and let \( (x_n^*) \) be an arbitrary sequence in \([0,1]\) satisfying
\[
F(y_n(x_n^*)) = \min_{x \in [0,1]} F(y_n(x)). \tag{6}
\]

Then:

1. \((\forall n \in \mathbb{N}): 0 \leq F(y_n^*) - \min_{y \in D} F(y) \leq L \varepsilon_n, \tag{7}\)
2. if \( \tilde{y} \in D \) is an accumulation point of \( (y_n^*) \), then
\[
F(\tilde{y}) = \min_{y \in D} F(y). \tag{9}
\]

Proof.

1. Let us assume that
\[
F(y^*) := \min_{y \in D} F(y) \leq F(y_n^*) \leq F(y_n(x_n)), \tag{8}
\]
where \( x_n \in [0,1] \) is chosen so that
\[
\|y_n(x_n) - y^*\| \leq \varepsilon_n.
\]

Hence
\[
0 \leq F(y_n^*) - F(y^*) \leq F(y_n(x_n)) - F(y^*) \leq L \|y_n(x_n) - y^*\| \leq L \varepsilon_n. \tag{9}
\]

The last inequality is based on the Lipschitz continuity of \( F \) on \( D \) and on the quality of \( y_n \). This completes the proof of Eq. (7).

2. If \( \tilde{y} \) is an accumulation point of \( (y_n^*) \), then there exists a subsequence (for the sake of clarity labeled the same way as the original sequence) such that
\[
y_n^* \to \tilde{y}. \tag{10}
\]
Using the continuity of \( F \) leads to
\[
F(y_n^*) \to F(\tilde{y}), \tag{11}
\]
but, from Eq. (8), we also get
\[
F(y_n^*) \to F(y^*). \tag{12}
\]
It follows that
\[
F(\tilde{y}) = F(y^*) = \min_{y \in D} F(y). \tag{13}
\]

Remark 1. If \( y^* = \arg \min_{y \in D} F(y) \) is unique, then
\[
y_n^* \to y^*. \tag{14}
\]

Using \( y_n \) reduces the multidimensional optimization problem into its one-dimensional equivalent that can be solved by univariate algorithms. If such method provides a lower bound \( M_n \) of the one-dimensional function \( F \circ y_n \), then \( M_n \) is obviously a lower bound of \( F \) along \( y_n \). Is it possible to establish a lower bound for \( F \) over the whole \( D \)? The following theorem gives an answer.

**Theorem 2.** Assume that the curve \( y_n, n \in \mathbb{N}, \) satisfies the assumptions of Thm. 1 and
\[
(\forall x \in [0,1]): M_n \leq F(y_n(x)). \tag{15}
\]

Then the value
\[
M = M_n - L \varepsilon_n \tag{16}
\]
is a lower bound of \( F \) over the entire \( D \), i.e.
\[
M \leq \min_{y \in D} F(y). \tag{17}
\]

Proof. Using the result and notation of the previous theorem, we get
\[
(\forall y \in D): F(y) \geq F(y^*) = (F(y^*) - L \varepsilon_n) + F(y_n(x_n)) \geq -L \varepsilon_n + M_n. \tag{18}
\]

In what follows, we assume that \( F \circ y_n \) is Hölder continuous with some real constants \( H \geq 0, \alpha \in (0,1) \), i.e.
\[
(\forall x', x'' \in [0,1]): |F(y_n(x')) - F(y_n(x''))| \leq H |x' - x''|^\alpha. \tag{19}
\]

Let us describe such curve for \( N \)-dimensional case. At first, divide the domain \( D \) into \( 2^N \) equal hypercubes. Number all of the subcubes using the index \( z_1, 0 \leq z_1 \leq 2^N - 1 \). Each subcube with the index \( z_1 \) is designated \( D(z_1) \). Moreover,
\[
D = \bigcup_{i=0}^{2^N-1} D(z_i). \tag{20}
\]

Using the same approach, divide each of the subcubes from the previous partitioning into \( 2^N \) equal subcubes and number them with the index \( z_2 \), \( 0 \leq z_2 \leq 2^N - 1 \). Each subcube of the second partitioning is now designated \( D(z_1, z_2) \). Continuing the same process we get hypercubes \( D(z_1, z_2, \ldots, z_M) \) and the edge length will be \( 2^{-M} \). The total number of the subcubes in \( M \)-th partition will be equal to \( 2^{MN} \) and
\[
D \supset D(z_1) \supset D(z_1, z_2) \supset \ldots \supset D(z_1, z_2, \ldots, z_M). \tag{21}
\]

Now cut the interval \([0,1]\) into \( 2^N \) equal subintervals. Every single part is designated \( d(z_1), 0 \leq z_1 \leq 2^N - 1 \). In the same manner, cut all the subintervals once again, etc. Continuing the same process we get \( 2^{MN} \) subintervals with the length equal to \( 2^{-MN} \), which are designated \( d(z_1, z_2, \ldots, z_M) \). Moreover,
\[
[0,1] \supset d(z_1) \supset d(z_1, z_2) \supset \ldots \supset d(z_1, z_2, \ldots, z_M). \tag{22}
\]
Fig. 1: The first and the second partition of the unit interval. The interval 
\[ d(z_1, z_2, \ldots, z_M) = [x, x + 2^{-MN}], \]  
and can be referred to as \( d(M, x) \). The corresponding subcube \( D(z_1, z_2, \ldots, z_M) \) is designated \( D(M, x) \).

The process of partitioning has to satisfy the following condition.

**Condition 1.** If the subintervals \( d(M, v') \) and \( d(M, v'') \), \( M \in \mathbb{N} \), have a common end point, then the corresponding subcubes \( D(M, v') \) and \( D(M, v'') \) have a common face.

Now consider a space filling curve \( y : [0, 1] \rightarrow D \) such that
\[ (\forall M \in \mathbb{N}_0) : y(d(z_1, \ldots, z_M)) \subset D(z_1, \ldots, z_M). \]  
Then \( F \circ y \) is Hölder continuous with the constants \( 2L\sqrt{N} + 3 \) and \( \frac{1}{2} \), i.e. for all \( x', x'' \in [0, 1] \)
\[ |F(y(x')) - F(y(x''))| \leq 2L\sqrt{N} + 3(|x' - x''| + \frac{1}{2}). \]  
The proof can be found in [1]. In what follows, we use Hilbert-type curves. The process of partitioning for two-dimensional Hilbert-type curve is illustrated by the following figure.

![Image](image1.png)

Fig. 2: The first and the second partition of the cube \( D \) for two-dimensional Hilbert-type curve.

For the practical computation, we use only "iterates" of the Hilbert curve. Consider the result of the \( M \)-th partition illustrated in this section. The construction of the "\( M \)-th iteration" of the Hilbert curve \( y_M \) is described for example in [1] and [3]. The curve \( y_M : [0, 1] \rightarrow D \),
\[ (\forall K < M) : x_i \in d(z_1, \ldots, z_K) \Rightarrow y_i \in D(z_1, \ldots, z_K). \]  
The curves \( y_M \) for \( M = 0, 1, 2 \) and \( N = 2 \) are illustrated by Fig. 3.

![Image](image2.png)

Fig. 3: Iterations of the Hilbert curve for \( N = 2 \).
is based on complex transformations and is completely described in [3]. The next goal is to prove that also $F \circ y_M$ satisfies the Hölder condition similar to Eq. (25).

**Theorem 3.** The function $F \circ y_M$, $M \in \mathbb{N}$, fulfills the Hölder condition with the constants $2L\sqrt{N+3}$ and $\frac{1}{N}$ on interval $[0,1]$, i.e. for all $x', x'' \in [0,1]$ 

$$|F(y_M(x')) - F(y_M(x''))| \leq 2L\sqrt{N+3}|x' - x''|^\frac{1}{N}. \quad (31)$$

Proof. Let $x', x'' \in [0,1]$, $x' \neq x''$. Then there exists $n \in \mathbb{N}_0$ such that 

$$2^{-(n+1)N} \leq |x' - x''| \leq 2^{2-nN}. \quad (32)$$

The proof can be divided into two parts.

1. Suppose that $n \geq M$. If $x'$ and $x''$ are from the same subinterval $d(M, x_i)$, then, according to the properties of $y_M$, the points $y_M(x')$ and $y_M(x'')$ belong to the same linear segment with end points $y_i$, $y_{i+1}$. From Eq. (28), Eq. (29) and Eq. (32), we get 

$$\|y_M(x') - y_M(x'')\| \leq \|y_i + (y_{i+1} - y_i)2^{MN}\|.$$ 

$$\|x' - x\| = 2^{MN}\|y_i - y||x' - x''| \leq 2^{2MN}2^{-M_2-nM}. \quad (33)$$

Using the first inequality in Eq. (32), it follows that 

$$\|y_M(x') - y_M(x'')\| \leq 2^{(M-n)(N-1)2}|x' - x''|^{1/N} \leq 2|x' - x''|^{1/N}. \quad (34)$$

Hence 

$$|F(y_M(x')) - F(y_M(x''))| \leq L\|y_M(x') - y_M(x'')\| \leq 2L|x' - x''|^{1/N}. \quad (35)$$

If $x' \in d(M, x_i)$ and $x'' \in d(M, x_{i+1})$, then the points $y_M(x')$, $y_M(x'')$ belong to two different segments with a common end point $y_i$. Using the previous result, we get 

$$\|y_M(x') - y_M(x'')\| \leq \|y_M(x') - y_{i+1}\| + \|y_M(x'') - y_{i+1}\| \leq 4|x' - x''|^{1/N}. \quad (36)$$

Hence 

$$|F(y_M(x')) - F(y_M(x''))| \leq 4L|x' - x''|^{1/N}. \quad (37)$$

2. Now suppose that $n \in \mathbb{N}_0$, $n > M$. If $x'$, $x''$ are both from $d(n, \bar{x}_i)$, then, from Eq. (30), $y_M(x')$, $y_M(x'') \in D(n, \bar{x}_i)$. The maximal distance can be estimated as follows 

$$\|y_M(x') - y_M(x'')\| \leq 2^{-n}\sqrt{N}. \quad (38)$$

If $x' \in d(n, \bar{x}_i)$, $x'' \in d(n, \bar{x}_{i+1})$, then $y_M(x') \in D(n, \bar{x}_i)$ and $y_M(x'') \in D(n, \bar{x}_{i+1})$. According to the Cond. 1, $D(n, \bar{x}_i)$ and $D(n, \bar{x}_{i+1})$ are contiguous. Therefore 

$$\|y_M(x') - y_M(x'')\| \leq 2^{-n}\sqrt{N+3}. \quad (39)$$

Using Eq. (32), we can derive the final estimate 

$$\|y_M(x') - y_M(x'')\| \leq 2\sqrt{N+3}|x' - x''|^{1/N}. \quad (40)$$

Since the function $F$ is Lipschitz continuous, it follows that 

$$|F(y_M(x')) - F(y_M(x''))| \leq L\|y_M(x') - y_M(x'')\| \leq 2L\sqrt{N+3}|x' - x''|^{1/N}, \quad (41)$$

which completes the proof. $\square$

**Remark 2.** Let $M_M$ be the lower bound of $F \circ y_M$ and let $N = 2$. Then 

$$(\forall y \in D) : M_M - L^{-2M} \leq F(y). \quad (42)$$

### 3. Optimization Algorithm

The algorithm introduced in this section is inspired by [1, 2] and is designed for the optimization of the univariate functions that are Hölder continuous. Consider $g : [a, b] \to \mathbb{R}$ satisfying the Hölder condition with the constants $H, \alpha$, $a, b \in \mathbb{R}$, $a < b$. Let $x_0, x_1, \ldots, x_n$ be the trial points obtained in the previous iterations. These points divide $[a, b]$ into $n$ intervals. Let $I_i$ be an interval with end points $x_j$, $x_k$ that are consecutive, $j, k \in \{0, \ldots, n\}$, $x_j < x_k$. Then the lower bounding function on the interval $I_i$ is constructed as follows:

$$l_i^n(x) := \max\{l_i^L(x), l_i^R(x)\}, \quad (43)$$

where 

$$l_i^L(x) = g(x_j) - H(x - x_j)\alpha, \quad (44)$$

$$l_i^R(x) = g(x_k) - H(x_k - x)\alpha. \quad (45)$$

The functions $l_i^L$ and $l_i^R$ are illustrated by Fig. 4. It can be shown that the function 

$$L^n(x) := l_i^n(x), \quad x \in I_i, i \in \{0, \ldots, n - 1\} \quad (46)$$

is a lower bounding of $g$ over $[a, b]$. 

![Fig. 4: The functions $l_i^L$ and $l_i^R$.](image-url)
The algorithm computes a value $y_i \in I_i$ as an $x$ coordinate of the intersection of $l_0^i$ and $l_0^i$. Then, the characteristic $M_i$ is computed in the following way

$$M_i = l_0^i(y_i) \leq l_0^i(y_i). \quad (47)$$

It is true that $M_i \leq \inf\{g(x), x \in I_i\}$. In Fig. 4 we can see a graphical interpretation of $y_i$ and $M_i$.

The interval $I_i$ with the lowest value of $M_i$ is chosen and $x_{n+1} := y_i$ becomes the next trial point, that divides $I_i$ into two subintervals. For both of them, new characteristics are computed. The algorithm goes on until

$$\text{diam}(I_i) \leq \delta. \quad (48)$$

Let us denote the approximation of the minimizer generated by the algorithm by $\bar{x}$. Let $\varepsilon$ be the desired precision of the algorithm, i.e. the goal is to find $\bar{x} \in [a, b]$ such that

$$g(\bar{x}) - \min_{x \in [a, b]} g(x) \leq \varepsilon. \quad (49)$$

The value $\delta = \delta_2$ in Eq. (18) can be chosen so that

$$\delta \leq 2 \left( \frac{\varepsilon}{H} \right)^{\frac{1}{\alpha}}. \quad (50)$$

The steps of the algorithm can be described as follows:

- **First iteration**: Set $x_0 = a$ and $x_1 = b$ and compute the values $g(x_0)$ and $g(x_1)$. Then the functions $l_0^0$ and $l_0^1$ are constructed, similarly to the Fig. 4. The point $y_0$ is found as the $x$ coordinate of their intersection and the characteristic $M_0$ is computed using Eq. (47). If $M_0 = g(y_0)$, the optimization is done and $\bar{x} = y_0$. Otherwise, the algorithm sets the next trial point $x_2 = y_0$, that divides $I_0$ into two subintervals, $I_0 = [x_0, x_2]$ and $I_1 = [x_2, x_1]$. For both intervals, the values $y_0, y_1$ and the corresponding characteristics $M_0, M_1$ are computed.

- **n-th iteration** Let $x_0, x_1, \ldots, x_n$ be the trial points gained from the previous iterations, not necessarily sorted. Let $I_0, I_1, \ldots, I_{n-1}$ be the intervals (generated in the previous steps) with the end points $x_i, \ i \in \{0, \ldots, n\}$. These intervals are characterized by the values $y_0, y_1, \ldots, y_{n-1}$ and $M_0, M_1, \ldots, M_{n-1}$ computed in a way described above. The algorithm chooses the interval $I_i$ such that

$$M_i = \min\{M_0, M_1, \ldots, M_{n-1}\}, \quad (51)$$

and sets $x_{n+1} = y_i$. If $M_i = g(y_i)$, then $\bar{x} = y_i$ and the optimization is done. Otherwise, the point $y_i$ divides $I_i$ into two subintervals, $I_i$ and $I_n$. For both subintervals, the points $y_i, y_n$ and the characteristics $M_i, M_n$ are computed. If

$$\text{diam}(I_i) \leq \delta, \quad (52)$$

the algorithm computes the approximation of the optimal value

$$g^* = \min \{g(x_i), 0 \leq i \leq n + 1\}, \quad (53)$$

and the minimizer

$$\bar{x} = \arg \min \{g(x_i), 0 \leq i \leq n + 1\}. \quad (54)$$

Otherwise, the next iteration is done in the same manner.

For illustration, the first 6 iterations for the function

$$g(x) = (x - 0.3)^2 + 1, \quad (55)$$

on the interval $[0, 1]$ were chosen. The red lines illustrate the curves $l_0^0$ and $l_0^1$ and the optimal values of $M_i$ are marked by the red dots.

The algorithm can be used to compute the optimal value of $F$ along the curve $y_M$. As we have shown in the previous section, the function $F \circ y_M$ is Hölder continuous on $[0, 1]$ with the constants $\alpha = \frac{1}{N}$ and $H = 2L\sqrt{N} + 3$. In the following observations, we assume that $N = 2$. We get

$$\sup_{y \in D} \text{dist}(y_M([0, 1]), y) = \frac{1}{2^M}. \quad (56)$$
Let us return to the problem Eq. (1). If $\varepsilon > 0$ is a desired precision, i.e. if we want to find $\tilde{y} \in D$ such that
\[
F(\tilde{y}) - \min_{y \in D} F(y) \leq \varepsilon,
\]
then we can choose the parameters $M$ and $\delta$ so that
\[
M \geq \log_2 \frac{2L}{\varepsilon}, \quad \delta \leq 2 \left( \frac{\varepsilon}{2H} \right)^2.
\]

To prove the statement, we use Thm. 1 and the quality of $y_M$, the inequality $\sqrt{D} \leq |\leq 10| + |\leq 10|$ and a triangular inequality and we get
\[
|F(\tilde{y}) - \min_{y \in D} F(y)| \leq |F(\tilde{y}) - \min_{x \in [0,1]} F(y_M(x))| + |\min_{x \in [0,1]} F(y_M(x)) - \min_{y \in D} F(y)| \leq \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon.
\]

The algorithm described above is rather theoretical. For the practical computation, some simplifications have to be made.

For the two-dimensional case it is not difficult to compute the intersection of $l^L_i$ and $l^R_i$, but for the high-dimensional problem it is more useful to approximate $l^L_i$ and $l^R_i$ by two lines, as we can see in Fig. 6. The value $\tilde{y}_i$ can be computed as the first coordinate of their intersection and
\[
\tilde{M}_i = \min\{l^L_i(\tilde{y}_i), l^R_i(\tilde{y}_i)\}.
\]

It is also more convenient to use an approximated value of the Hölder constant. In [2], two possible approximations are proposed.

The choice of the interval $I_i$ can be improved. Two methods are presented in [2].

4. Numerical Experiments

For our experiments, three types of functions were chosen. Let us start with the function
\[
F_1(x, y) = (x - 0.3)^2 + (y - 0.7)^2 + 1.
\]

It is not difficult to realize that $y^* = [0.3, 0.7]$ and $F(y^*) = 1$. The algorithm was tested for $M = 1, 2, \ldots, 10$ and the precision $10^{-3}$. The results are summarized in the Tab. 1.

![Fig. 7: The function $F_1$ together with the fifth level of the Hilbert curve and the approximate minimizer.](image)

Tab. 1: Results summarization.

| $M$ | $F(y_M(\tilde{x})) - F(y^*)$ | $\text{ran}([\tilde{x}])$ | $|\text{ran}([\tilde{x}]) - y^*|_{\|}$ |
|-----|-----------------------------|--------------------------|---------------------------------|
| 1   | $4 \cdot 10^{-2}$           | [0.2998, 0.3000]          | 0.0000                          |
| 2   | $2.5 \cdot 10^{-3}$         | [0.3000, 0.7000]          | 0.0500                          |
| 3   | $5.1 \cdot 10^{-3}$         | [0.3000, 0.7158]          | 0.0720                          |
| 4   | $1.5 \cdot 10^{-3}$         | [0.3125, 0.7016]          | 0.0126                          |
| 5   | $1.5 \cdot 10^{-4}$         | [0.3016, 0.6870]          | 0.0126                          |
| 6   | $7.6 \cdot 10^{-5}$         | [0.3082, 0.7034]          | 0.0087                          |
| 7   | $1.1 \cdot 10^{-4}$         | [0.2981, 0.6994]          | 0.0110                          |
| 8   | $5.6 \cdot 10^{-5}$         | [0.2930, 0.7026]          | 0.0075                          |
| 9   | $2.9 \cdot 10^{-5}$         | [0.2949, 0.7019]          | 0.0054                          |
| 10  | $1.8 \cdot 10^{-5}$         | [0.2959, 0.7013]          | 0.0043                          |

Consider next the multiextremal function
\[
F_2(x, y) = -0.5 \sin(2\pi x) \sin(2\pi y) + 1.
\]

The algorithm found one of the minimizers and the result is illustrated by Fig. 8.

The last tested function is
\[
F_3(x, y) = \sqrt{(x - 0.6)^2 + (y - 0.4)^2 + 0.5}.
\]

As we can be seen Fig. 9 the algorithm works also for the nonsmooth function.
5. Conclusion

In this paper, we described a minimization algorithm based on reducing the problem by means of Hilbert-type space filling curve. Subsequently, complete mathematical analysis was done to show that the function $F \circ y_M$ is Hölder continuous. Thus an algorithm for minimizing univariate Hölder continuous functions could be used to find an approximation of the optimal value of $F$. It was shown that the algorithm computed the approximate minimum of $F$ with the guaranteed precision. The algorithm and the analysis will be extended for the dimensions $N > 2$ and used for some practical problems of the multivariate optimizations.

Acknowledgment

The work is partially supported by Grant of SGS No. SP2017/122. VSB–Technical University of Ostrava, Czech Republic.

References


About Authors

Michaela BAILOVA was born in Ostrava, Czech Republic. She received her M.Sc. from VSB–Technical University of Ostrava in 2015. Her research interests include functional and numerical analysis.

Jiri BOUCHALA was born in Novy Jicin, Czech Republic. He received his Ph.D. from Faculty of Applied Sciences, University of West Bohemia in Pilsen in 2000. His research interests include functional analysis, boundary element methods, variational methods, partial differential equations.

Petr VODSTRCIL was born in Svitavy, Czech Republic. He received his Ph.D. from Masaryk University in Brno in 2005. His research interests include calculus and functional analysis.