FOUNDATIONS OF COMPUTING AND DECISION SCIENCES
Vol. 38 (2013) No. 3

DOI: 10.2478/fcds-2013-0009 ISSN 0867-6356
e-ISSN 2300-3405

INTERVAL VERSIONS OF CENTRAL-DIFFERENCE METHOD FOR SOLVING THE POISSON EQUATION IN PROPER AND DIRECTED INTERVAL ARITHMETIC

Tomasz HOFFMANN [*], Andrzej MARCINIAK [†], Barbara SZYSZKA [‡]

Abstract. To study the Poisson equation, the central-difference method is often used. This method has the local truncation error of order $O(h^2 + k^2)$, where h and k are mesh constants. Using this method in conventional floating-point arithmetic, we get solutions including the method, representation and rounding errors. Therefore, we propose interval versions of the central-difference method in proper and directed interval arithmetic. Applying such methods in floating-point interval arithmetic allows one to obtain solutions including all possible numerical errors. We present numerical examples from which it follows that the presented interval method in directed interval arithmetic is a little bit better than the one in proper interval arithmetic, i.e. the intervals of solutions are smaller. It appears that applying both proper and directed interval arithmetic the exact solutions belong to the interval solutions obtained.

Keywords: Poisson equation, central-difference method, interval methods, proper interval arithmetic, directed interval arithmetic, floating-point interval arithmetic.

1 Introduction

Many scientific and engineering problems are described in the form of partial differential equations. If such equations cannot be solved analytically, we use approximate methods to solve them, usually providing all calculations in floating-point arithmetic. Using approximate methods we obtain solutions including some errors of methods,

[*]Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland, Institute of Computing Science, tomaszhof@gmail.com

[†]Poznan University of Technology, Piotrowo 2, 60-965 Poznan, Poland, Institute of Computing Science, Andrzej.Marciniak@put.poznan.pl

[‡]Poznan University of Technology, Piotrowo 3A, 60-965 Poznan, Poland, Institute of Mathematics, Barbara.Szyszka@put.poznan.pl

and floating-point arithmetic causes representation errors and rounding errors. Interval arithmetic makes it possible to represent any input data in the form of machine interval and perform all calculations in floating-point interval arithmetic which includes rounding errors. If an interval method used to solve a problem includes also the error of the method, then we can obtain a solution (in the form of interval) which contains all possible numerical errors.

In our previous papers [3], [4] we have considered an interval difference method for solving the Poisson equation

$$\frac{\partial^2 u}{\partial x^2}(x,y) + \frac{\partial^2 u}{\partial y^2}(x,y) = f(x,y). \tag{1}$$

In (1) the function f describes the input to the problem on a plane region R whose boundary will be denoted by Γ. We assume that this function is continuous together with its partial derivatives up to the second order.

To obtain a unique solution to (1), we usually apply the Dirichlet boundary conditions

$$u(x,y) = \varphi(x,y)$$

for all (x,y) on Γ. In general, the plane region R may be arbitrary, but further we will assume that R is a rectangular:

$$R = \{(x,y) : 0 < x < \alpha,\ \ 0 < y < \beta\}.$$

Thus, the problem is to find $u = u(x,y)$ satisfying the equation

$$\frac{\partial^2 u}{\partial x^2}(x,y) + \frac{\partial^2 u}{\partial y^2}(x,y) = f(x,y),\ \ 0 < x < \alpha,\ \ 0 < y < \beta, \tag{2}$$

with boundary conditions

$$u|_\Gamma = \varphi(x,y) = \begin{cases} \varphi_1(y) & \text{for} \quad x = 0, \\ \varphi_2(x) & \text{for} \quad y = 0, \\ \varphi_3(y) & \text{for} \quad x = \alpha, \\ \varphi_4(x) & \text{for} \quad y = \beta, \end{cases} \tag{3}$$

where

$$\varphi_1(0) = \varphi_2(0), \varphi_2(\alpha) = \varphi_3(0), \varphi_3(\beta) = \varphi_4(\alpha), \varphi_4(0) = \varphi_1(\beta),$$
$$\Gamma = \{(x,y) : x = 0, \alpha \ \text{ and } \ 0 \le y \le \beta \ \text{ or } \ 0 \le x \le \alpha \ \text{ and } \ y = 0, \beta\}.$$

should be noted that an application of interval arithmetic for solving the Poisson equation is know only from a few of papers. We can mention here [8] – [10] and [12] – [14]. But our approach is quite different–for known difference methods (the central-difference method in this paper) we construct interval analogies that include the errors of methods, and then we solve interval liner system of equations that erased by floating-point interval arithmetic.

2 The Central-Difference Method

Partitioning the interval $[0, \alpha]$ into n equal parts of width h and the interval $[0, \beta]$ into m equal parts of width k provides a means of placing a grid on the rectangle R with mesh points $(x_i, y_j) = (ih, jk)$, where $h = \alpha/n, k = \beta/m, i = 0, 1, ..., n$ and $j = 0, 1, ..., m$. Assuming that the fourth order partial derivatives of u exist, for each mesh point in the interior of the grid we use the Taylor series in the variable x about x_i and in the variable y about y_j. This allows us to express the Poisson equation at the points (x_i, y_j) as

$$\begin{aligned} &\frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{h^2} + \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{k^2} \\ &\qquad - \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i, y_j) - \frac{k^2}{12}\frac{\partial^4 u}{\partial y^4}(x_i, \eta_j) = f(x_i, y_j), \\ &\qquad i = 1, 2, \ldots, n-1, \quad j = 1, 2, \ldots, m-1, \end{aligned} \tag{4}$$

where $\xi_i \in (x_{i-1}, x_{i+1})$, $\eta_j \in (y_{j-1}, y_{j+1})$, and the boundary conditions as

$$\begin{aligned} u(0, y_j) &= \varphi_1(y_j) \quad \text{for each } \ j = 0, 1, \ldots, m, \\ u(x_i, 0) &= \varphi_2(x_i) \quad \text{for each } \ i = 1, 2, \ldots, n-1, \\ u(\alpha, y_j) &= \varphi_3(y_j) \quad \text{for each } \ j = 0, 1, \ldots, m \\ u(x_i, \beta) &= \varphi_4(x_i) \quad \text{for each } \ i = 1, 2, \ldots, n-1. \end{aligned} \tag{5}$$

Omitting in (4) the partial derivatives, this results in a method, called the central-difference method, with local truncation error of order $O(h^2 + k^2)$. This method can be written in the form

$$\begin{aligned} &\frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{ij} + u_{i,j-1}}{k^2} = f_{ij}, \\ &\qquad i = 1, 2, \ldots, n-1, j = 1, 2, \ldots, m-1. \end{aligned}$$

where u_{ij} is an approximation to $u(x_i, y_j)$ and $f_{ij} = f(x_i, y_j)$. The above formulas together with (5) present a system of linear equations which may be solved by any known exact or iterative method.

3 Intervals Difference Methods

Let us assume that there exists a constant M such that

$$\left| \frac{\partial^4 u}{\partial x^2 \partial y^2} \right| \leq M \ \text{ for all } \ 0 \leq x \leq \alpha \ \text{ and } \ 0 \leq y \leq \beta, \tag{6}$$

and let

$$\frac{\partial^4 u}{\partial x^2 \partial y^2}(x, y) = \frac{\partial^4 u}{\partial y^2 \partial x^2}(x, y).$$

Since from the Poisson equation (1) it follows that

$$\frac{\partial^4 u}{\partial x^4}(x,y) = \frac{\partial^2 f}{\partial x^2}(x,y) - \frac{\partial^4 u}{\partial x^2 \partial y^2}(x,y),$$
$$\frac{\partial^4 u}{\partial y^4}(x,y) = \frac{\partial^2 f}{\partial y^2}(x,y) - \frac{\partial^4 u}{\partial y^2 \partial x^2}(x,y),$$

then it is obvious that we have

$$\begin{aligned} &\frac{\partial^4 u}{\partial x^4}(\xi,y) \in \Psi(X+[-h,h],Y) + [-M,M], \\ &\frac{\partial^4 u}{\partial y^4}(x,\eta) \in \Psi(X,Y+[-k,k]) + [-M,M], \end{aligned} \tag{7}$$

for any $\xi \in (x-h, x+h)$ and any $\eta \in (y-k, y+k)$ where X and Y denote interval extensions of x and y, respectively, and $\Psi(X,Y)$ and $\Omega(X,Y)$ are interval extensions of $\frac{\partial^2 f}{\partial x^2}(x,y)$ and $\frac{\partial^2 f}{\partial y^2}(x,y)$ respectively. If we recall the Poisson equation at the mesh points (4) and write the partial derivatives at the right-hand side, it is easy now to write and interval analogy to this equation. Assuming that all interval extensions are proper, we have

$$\begin{aligned} & k^2 U_{i-1,j} + h^2 U_{i,j-1} - 2(h^2+k^2)U_{i,j} + k^2 U_{i+1,j} + h^2 U_{i,j+1} = \\ &= h^2k^2\Big(F_{i,j} + \frac{1}{12}\Big[h^2\Psi(X_i+[-h,h],Y_j) + k^2\Omega(X_i,Y_j+[-k,k]) + \\ &\qquad + (h^2+k^2)[-M,M]\Big]\Big), \\ &\qquad i = 1,2,\ldots,n-1, \quad j = 1,2,\ldots,m-1, \end{aligned} \tag{8}$$

where $F_{i,j} = F(X_i, Y_j)$, and where

$$\begin{aligned} U_{0,j} &= \Phi_1(Y_j) \quad \text{for each } j = 0,1,\ldots,m, \\ U_{i,0} &= \Phi_2(X_i) \quad \text{for each } i = 1,2,\ldots,n-1, \\ U_{n,j} &= \Phi_3(Y_j) \quad \text{for each } j = 0,1,\ldots,m \\ U_{i,m} &= \Phi_4(X_i) \quad \text{for each } i = 1,2,\ldots,n-1. \end{aligned} \tag{9}$$

$\Phi_1(Y), \Phi_2(Y), \Phi_3(Y)$ and $\Phi_4(Y)$ denote interval extensions of the functions $\varphi_1(y)$, $\varphi_2(y)$, $\varphi_3(y)$ and $\varphi_4(y)$, respectively.

The system of linear interval equations (8) – (9) can be solved in conventional (proper) floating-point interval arithmetic (see e.g. [2]) since all intervals are proper, i.e. for any interval $[a,b]$ we have $a \le b$.

But we can consider another interval analogy of (4). Namely, we can write (also using (7) and assuming that all interval extensions are proper)

$$
\begin{aligned}
&k^2U_{i-1,j} + h^2U_{i,j-1} - 2(h^2+k^2)U_{i,j} + k^2U_{i+1,j} + h^2U_{i,j+1} - \\
&-\frac{h^2k^2}{12}\Big(h^2\Psi(X_i+[-h,h],Y_j) + k^2\Omega(X_i,Y_j+[-k,k]) + (h^2+k^2)[-M,M]\Big) = \\
&= h^2k^2F_{i,j}, \\
&i = 1,2,\ldots,n-1, \quad j = 1,2,\ldots,m-1.
\end{aligned}
$$

Using directed interval arithmetic (see e.g. [7] and [11]), we can add at both sides of this equation the opposites to

$$-\frac{h^4k^2}{12}\Psi(X_i+[-h,h],Y_j),$$

$$-\frac{h^2k^4}{12}\Omega(X_i,Y_j+[-k,k])$$

and

$$-\frac{h^2k^2}{12}(h^2+k^2)[-M,M]$$

(the opposite of an interval, like the inverse of an interval, does not exist in proper interval arithmetic). We get

$$
\begin{aligned}
&k^2U_{i-1,j} + h^2U_{i,j-1} - 2(h^2+k^2)U_{i,j} + k^2U_{i+1,j} + h^2U_{i,j+1} = \\
&= h^2k^2\Big(F_{i,j} + \frac{1}{12}\Big[h^2\Psi(X_i+[-h,h],Y_j) + k^2\Omega(X_i,Y_j+[-k,k]) + \\
&\qquad + (h^2+k^2)[M,-M]\Big]\Big), \qquad (10) \\
&i = 1,2,\ldots,n-1, \quad j = 1,2,\ldots,m-1,
\end{aligned}
$$

The equation (10) differs from the equation (8) only by the last term on the right-hand side which is an improper interval. But using the directed floating-point interval arithmetic we can solve the system of equations (10) (together with (9)). If the interval solutions of this system are in the form of improper intervals, to get the proper intervals we can use the so-called proper projection of intervals, i.e. transform each interval $[a,b]$, for which $b<a$, to the interval $[b,a]$.

In the interval methods (8) and (10) each of the known exact value belongs to its interval extension, i.e.

$$f_{ij} \in F_{ij}, \quad \frac{\partial^2 f}{\partial x^2}(\xi_i, y_j) \in \Psi(X_i+[-h,h],Y_j), \quad \text{etc.}$$

But (in our opinion) in general it is impossible to validate the solution (to prove analytically that the solution of (8) or (10) contains the exact solution, i.e. that $u_{ij} \in U_{ij}$). Thus, it can be only a hypothesis confirmed by numerical experiments (see Section 4).

We should also add a remark concerning the constant M (see (6)). In the examples presented in the next section, in which we have compared interval solutions with the exact ones, the constant M could be evaluated very easy, because the exact solutions were known in advance. In general (when the exact solution is unknown and nothing can be concluded about M from physical or technical properties or characteristics of the problem considered), we propose to find this constant by the procedure described below.

It is obvious that

$$\frac{\partial^4 u}{\partial x^2 \partial y^2}(x_i, y_j) =$$
$$= \lim_{h\to 0}\lim_{k\to 0}\Big(\frac{u_{i-1,j-1}+u_{i-1,j+1}+u_{i+1,j-1}+u_{i+1,j+1}}{h^2k^2} +$$
$$+ \frac{4u_{ij}-2(u_{i-1,j}+u_{i,j-1}+u_{i,j+1}+u_{i+1,j})}{h^2k^2}\Big).$$

We can calculate the constants

$$M_{nm} = \frac{1}{h^2k^2}\max_{i,j}\Big|u_{i-1,j-1}+u_{i-1,j+1}+u_{i+1,j-1}+u_{i+1,j+1}+$$
$$+4u_{ij}-2(u_{i-1,j}+u_{i,j-1}+u_{i,j+1}+u_{i+1,j})\Big|,$$

for $i = 1, 2, \ldots, n-1, j = 1, 2, \ldots, m-1$ and where u_{ij} are obtained by the conventional central-difference method (see Sec. 2), for a variety of n and m, say $n = m = 10, 20, 30, \ldots, N$, where N is sufficiently large. Then, we can plot M_{nm} against different $n = m$. The constant M can be easy determine from the obtained graph, since

$$\lim_{n\to\infty}\lim_{m\to\infty} M_{nm} \le M$$

(see Figures 3 and 6 in the next section).

4 Numerical Experiments

The realization of proper interval arithmetic in floating-point arithmetic consists in using downwardly directed rounding when calculating the left endpoint of a resulting interval and upwardly directed rounding when calculating the right endpoint of such an interval (see e.g. [2] or [5]). The realization of directed interval arithmetic in floating-point arithmetic is not too easy, because in every elementary operation we must obtain the resulting interval which contains all possible roundings. Several cases must be considered for every such an operation (see e.g. [7] and [11]).

To apply floating-point interval arithmetic one can use the Pascal-XSC or C-XSC scientific computer languages developed in the Universitt Karlsruhe. But in our experiments we have applied our own unit called *IntervalArithmetic* (see e.g. [5]) written in the Delphi Pascal programming language. This unit takes advantage of the Delphi Pascal floating-point Extended type[1] and makes it possible to represent any input numerical data in the form of machine interval, perform all calculations in floating-point interval arithmetic, use some standard interval functions and give results in the form of proper intervals.

We have carried out a number of numerical experiments for various functions $f(x, y)$ occurring in the Poisson equation (2) and various boundary conditions (3) using both: the method (8) with the conventional floating-point interval arithmetic and the method (10) with the directed floating-point interval arithmetic. In all examples considered and in both these methods, the exact solutions (if they are known) are included in the interval solutions obtained. Below we present two examples.

Example 1

Let us take into account the following boundary value problem:

$$\frac{\partial^2 u}{\partial x^2}(x,y) + \frac{\partial^2 u}{\partial y^2}(x,y) = 0, \quad u = u(x,y), \quad 0 < x < 1, \quad 0 < y < 1,$$
$$u|_\Gamma = \varphi(x,y) = \begin{cases} \varphi_1(y) = & \cos(3y) & \text{for} \quad x = 0, \\ \varphi_2(x) = & \exp(3x) & \text{for} \quad y = 0, \\ \varphi_3(y) = & \exp(3)\cos(3y) & \text{for} \quad x = 1, \\ \varphi_4(x) = & \exp(3x)\cos(3) & \text{for} \quad y = 1. \end{cases} \tag{11}$$

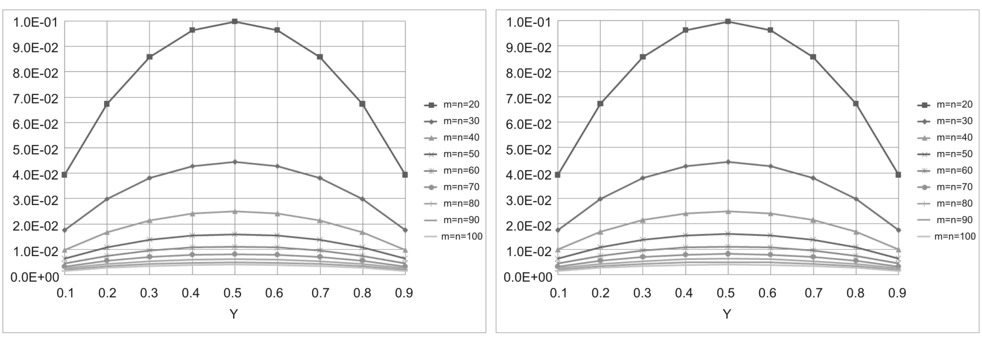

Figure 1: The widths of interval solutions obtained in proper (on the left-hand side) and directed (on the right-hand side) interval arithmetic for the problem (11)

[1]The Extended real type has larger precision and range than the double real type used in Pascal-XSC. Moreover, the realization of directed interval arithmetic in Pascal-XSC presented in [11] consists in finding two resulting intervals (called outward and inward rounding) for any elementary arithmetic operation, while in our *IntervalArithmetic* unit (the current version, still developed, is available in [1]) we choose an appropriate resulting interval on the basis of the widths of intervals (from all possible resulting intervals we always choose the worst case, i.e. the interval with the largest width S see [6] for details).

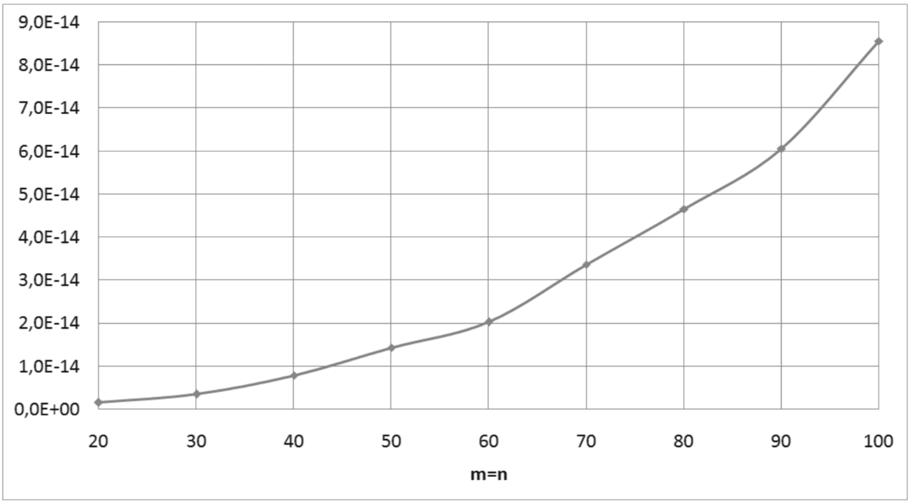

Figure 2: The differences of widths of the intervals obtained in both interval arithmetics for the problem (11): width(U_p) - width(U_d)

The exact solution is given by

$$u(x, y) = \exp(3x)\cos(3y). \tag{12}$$

In Table 1 we present the results obtained by conventional central-difference method (u_{conv}) and in proper and directed interval arithmetic at the center of the region Γ, while in Figure 1 and 2 we show the widths of interval solutions obtained in both arithmetic for different meshes (m and n). We have assumed that $M = 1627$. Of course, this estimation of

$$\left|\frac{\partial^4 u}{\partial x^2 \partial y^2}\right|$$

can be calculated from (12), but a similar estimation can be concluded from the graph presented in Figure 3.

Table 1: The interval solutions and the widths of intervals obtained in proper (U_p) and directed (U_d) interval arithmetic for the problem (11) at $(0.5, 0.5)$

Exact solution $u(0.5, 0.5) \approx 0.317022143580443$

$m = n$	$u_{conv}(0.5, 0.5)$	$U_p(0.5, 0.5)$	$width(U_p)$
20	0.317802799652435	[0.267957818017965, 0.367647781286905]	0.099690
40	0.317217584617279	[0.304737968870283, 0.329697200364276]	0.024959
60	0.317109029137083	[0.311561016819748, 0.322657041454418]	0.011096
80	0.317071021212368	[0.313949965550288, 0.320192076874449]	0.006242
100	0.317053426600120	[0.315055862461985, 0.319050990738256]	0.003995
$n = m$	$U_d(0.5, 0.5)$	$width(U_d)$	$width(U_p) - width(U_d)$
20	[0.267957818017966, 0.367647781286904]	0.099690	$1.596E-16$
40	[0.304737968870287, 0.329697200364273]	0.024959	$7.900E-15$
60	[0.311561016819759, 0.322657041454408]	0.011096	$2.030E-14$
80	[0.313949965550311, 0.320192076874426]	0.006242	$4.659E-14$
100	[0.315055862462027, 0.319050990738214]	0.003995	$8.565E-14$

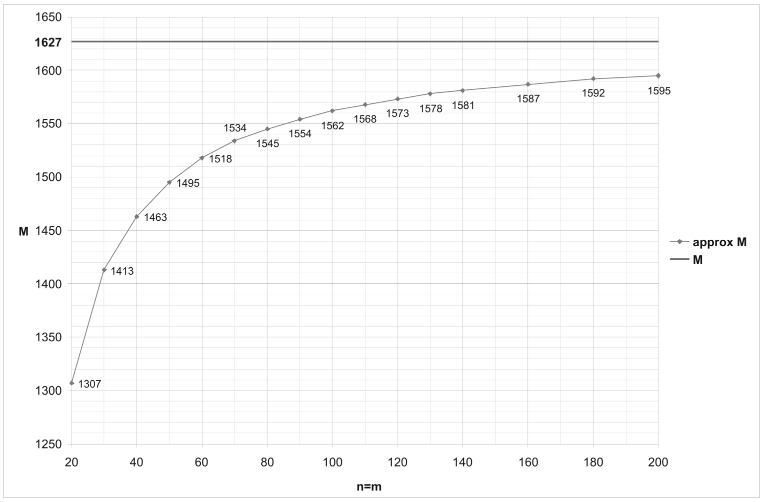

Figure 3: Approximations to the constant M for the problem (11)

Example 2

Let us consider another boundary value problem of the following form:

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2}(x,y) + \frac{\partial^2 u}{\partial y^2}(x,y) \quad &= \quad -2\pi^2 \sin(\pi x)\sin(\pi y), \quad u = u(x,y), \\ 0 < x < 1, \qquad & \quad\;\; 0 < y < 1, \\ u|_\Gamma \quad &= \quad 0, \end{aligned} \tag{13}$$

and with the exact solution

$$u(x,y) = \sin(\pi x)\sin(\pi y). \tag{14}$$

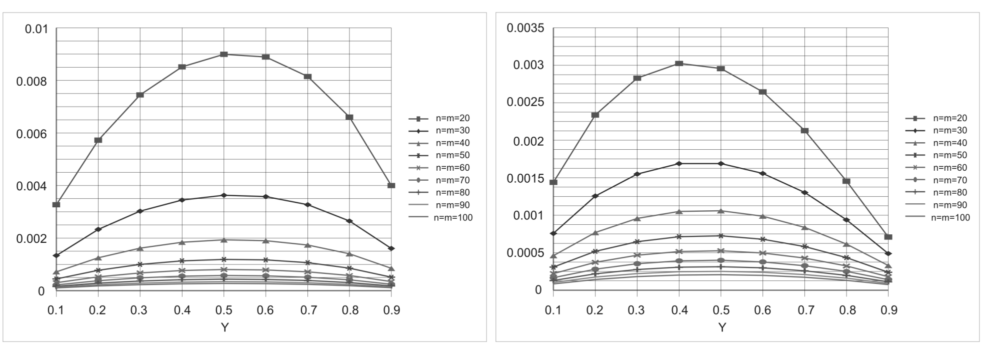

Figure 4: The widths of interval solutions obtained in proper (on the left-hand side) and directed (on the right-hand side) interval arithmetic for the problem (13)

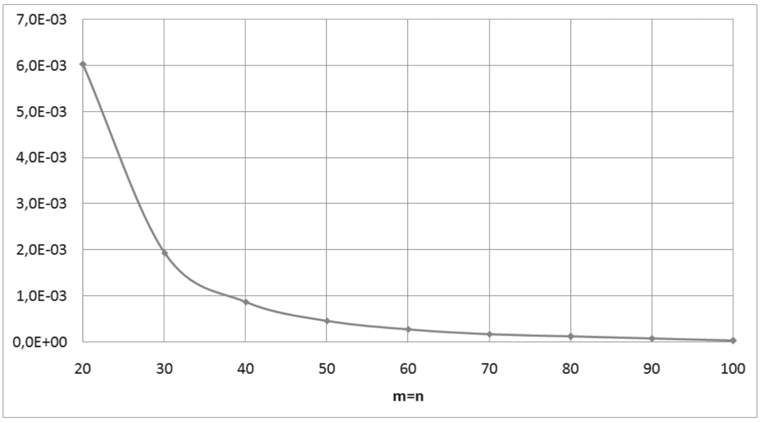

Figure 5: The differences of widths of the intervals obtained in both interval arithmetics for the problem (13): width(U_p) - width(U_d)

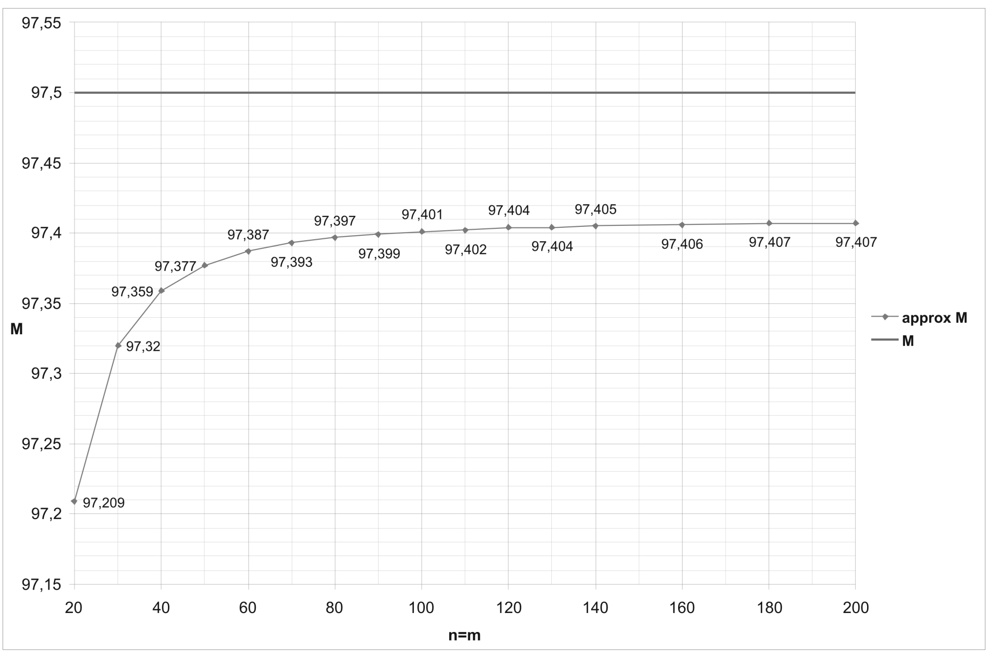

Figure 6: Approximations to the constant M for the problem (13)

The interval solutions obtained in proper and directed interval arithmetic at the center of the region Γ are presented in Table 2. In Figures 4 and 5 we present the graphs of interval widths. To solve the problem (13) by the interval difference method (8) and (10) we have assumed that $M = 97.5$. Applying the procedure described at the end of Section 3 we can obtain a similar value (see Figure 6).

Table 2: The interval solutions and the widths of intervals obtained in proper (U_p) and directed (U_d) interval arithmetic for the problem (13) at $(0.5, 0.5)$

Exact solution $u(0.5, 0.5) = 1$

$m = n$	$u_{conv}(0.5, 0.5)$	$U_p(0.5, 0.5)$	$width(U_p)$
20	1.00205870676453	[0.994303172294329, 1.00329669988280]	0.008994
40	1.00051420047815	[0.998596029202215, 1.00052866286388]	0.001933
60	1.00022849438547	[0.999387775747690, 1.00019106751678]	0.000803
80	1.00012852038354	[0.999660751812287, 1.00009518839725]	0.000434
100	1.00008225076221	[0.999785209721521, 1.00005621380286]	0.000271

$n = m$	$U_d(0.5, 0.5)$	$width(U_d)$	$width(U_p) - width(U_d)$
20	[0.997322654757582, 1.00027721741955]	0.002955	$6.039E-03$
40	[0.999032949851789, 1.00009174221431]	0.001059	$8.738E-04$
60	[0.999526124430187, 1.00005271883428]	0.000527	$2.767E-04$
80	[0.999721122107072, 1.00003481810246]	0.000313	$1.207E-04$
100	[0.999816800768480, 1.00002462275590]	0.000208	$1.011E-05$

It is important that although the calculations by the method (10) in directed floating-point interval arithmetic are longer in time (approximately 15%) than by the method (8) in conventional one, the method (10) yields interval solutions with smaller widths. Depending on the problem considered, the differences in widths may be decreasing or increasing in the number of mesh points, but in all cases the widths of intervals for directed interval arithmetic are smaller. This conclusion refers not only to the examples presented, but also to many other experiments provided by us. In our opinion, it is the main advantage of applying the floating-point directed interval arithmetic instead of the floating-point proper one.

5 Conclusions and further studies

Interval methods for solving partial-differential equation problems in floating-point interval arithmetic give solutions in the form of intervals which contain all possible numerical errors, i.e. representation errors, rounding errors, and errors of methods. In further studies we plan to use other (faster) exact methods for solving the system of linear interval equations occurring in the problem considered. We will also try to solve a generalized Poisson equation of the form

$$a(x,y)\frac{\partial^2 u}{\partial x^2}(x,y) + b(x,y)\frac{\partial^2 u}{\partial y^2}(x,y) = f(x,y),$$

$$a(x,y)b(x,y) > 0,$$

with some boundary conditions, and other partial-differential problems using interval difference methods (not only based on the central-difference formula).

References

[1] Delphi Pascal IntervalArithmetic Unit, `http://www.cs.put.poznan.pl/amarciniak/DEL-wyklady/IntervalArithmetic.pas`

[2] Hammer, R., Hocks, M., Kulisch, U., Ratz, D.: Numerical Toolbox for Verified Computing I: Basic Numerical Problems, Springer, Berlin (1993)

[3] Hoffmann, T., Marciniak, A.: Solving the Poisson Equation by an Interval Difference Method of the Second Order, Computational Methods in Science and Technology 19 (1), 13–21, (2013)

[4] Marciniak, A.: An Interval Difference Method for Solving the Poisson Equation– the First Approach, Pro Dialog 24, 49-61, (2008)

[5] Marciniak, A.: Selected Interval Methods for Solving the Initial Value Problem, Publishing House of Poznan University of Technology, Poznan (2009)

[6] Marciniak, A.: On Realization of Floating-Point Directed Interval Arithmetic, 2012, `http://www.cs.put.poznan.pl/amarciniak/KONF-referaty/DirectedArithmetic.pdf`

[7] Markov, S.: On Directed Interval Arithmetic and its Applications, Journal of Universal Computer Science 7, 514-526, (1995)

[8] Nakao, M.T.: A Numerical Approach to the Proff of Existence of Solutions for Elliptic Problems, Japan Journal of Industrial and Applied Mathematics 5 (2), 313-332, (1988)

[9] Nakao, M.T.: Numerical Verification Methods for Solutions of Ordinary and Partial Differential Equations, Numerical Functional Analysis and Optimization 22, 321-356, (2001)

[10] Nakao, M.T., Watanabe, Y.: An Efficient Approach to the Numerical Verification for Solutions of Elliptic Differential Equations, Numerical Algorithms 37, 311-323, (2004)

[11] Popova, E.D.: Extended Interval Arithmetic in IEEE Floating-Point Environment, Interval Computations 4, 100-129, (1994)

[12] Schwandt, H.: An Interval Arithmetic Approach for the Construction of an Almost Globally Convergent Method for the Solution of Nonlinear Poisson Equation on the Unit Square, SIAM Journal of Scientific and Statistical Computing 5 (2), 427-452, (1984)

[13] Schwandt, H.: Almost Globally Convergent Interval Methods for Discretizations of Nonlinear Elliptic Partial Differential Equations, SIAM Journal on Numerical Analysis 23 (2), 304-324, (1986)

[14] Schwandt, H.: The Solution of Nonlinear Elliptic Dirichlet Problems on Rectangles by Almost Globally Convergent Interval Methods, SIAM Journal on Scientific and Statistical Computing 6 (3), 617-638, (1985)

Received December, 2012