Calculation error estimation without assuming a theoretical solution in the numerical calculation of the 1D Poisson equation by IFDM

Tsugio Fukuchi (fkbandou@sunny.ocn.ne.jp)

Research Article

Keywords: Poisson equation, interpolation finite difference method (IFDM), high-accuracy calculation, estimation of calculation error, Newton's method

Posted Date: June 23rd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1748619/v2

Calculation error estimation without assuming a theoretical solution in the numerical calculation of the 1D Poisson equation by IFDM

Tsugio Fukuchi*, E-mail: fkbandou@sunny.ocn.ne.jp
*Tsubokura Ground Survey and Design Ltd., Fukushima 960-0113, Japan

The extreme high-accuracy calculation of the 1D Poisson equation by the interpolation finite difference method (IFDM) is possible. Numerical calculation errors have conventionally been evaluated by comparison with theoretical solutions. However, it is not always possible to obtain a theoretical solution. In addition, when trying to obtain numerical values from the theoretical solution, the exact numerical value may not be obtained because of inherent difficulties, that is, a theoretical solution equal to the exact numerical solution does not hold. In this paper, we focus on an ordered structure of the error calculated by the high-order accuracy finite difference (FD) scheme. This approach clarifies that in the numerical calculation of the 1D Poisson equation, which is the most basic ordinary differential equation, the error of the numerical calculation can be evaluated without comparison with the theoretical solution. Furthermore, in the numerical calculation by the IFDM, not only the discrete solution but also the continuous approximate solution is defined by piecewise interpolation polynomials.
Keywords: Poisson equation, interpolation finite difference method (IFDM), high-accuracy calculation, estimation of calculation error, Newton’s method

1. Introduction

Mathematically, innumerable higher-order differential equations can be considered, but there is a phenomenological fact that the basic equations of physical engineering are overwhelmingly second-order differential equations [1]. Among them, the Poisson equation (including the Laplace equation as a special form) has an important meaning as a partial differential equation that reflects a static physical phenomenon that does not depend on time.

In general, in the ideal numerical analysis by the finite difference method (FDM) regarding the Poisson equation, the following three characteristics are important: (i) being able to handle arbitrary boundary shapes, (ii) being able to perform high-accuracy calculations, and (iii) being able to perform high-speed calculations. Regarding the characteristic of (i), there is an approach from the Ghost fluid method [2-4]. Regarding the characteristic of (ii), there is an approach from the compact FDM [5-7]. The author also reported that the numerical calculation system by the interpolation finite difference method (IFDM) has the above three characteristics [8-16]. The second-order accuracy FD scheme has usually been employed in the numerical calculation of the Poisson equation. The accuracy of the FD

scheme is often sufficient. However, many studies of high-accuracy numerical analysis by high-order FD schemes have been reported. Furthermore, many studies have investigated high-accuracy numerical calculation methods based on the spectral method [17].

Numerical calculations are usually performed in double precision calculations. The number of displayed digits is 15. If the number of significant digits in the numerical calculation is 15, the final goal of the high-accuracy numerical calculation has been reached. Numerical analysis by the spectral method has potential. However, even in numerical calculations by the IFDM, such a calculation can be performed [16].

The discussion of numerical error is usually based on the assumption that the exact value is known. In the one-dimensional boundary value problem, the theoretical solution is often obtained, but in the multidimensional boundary value problem, it is usually difficult to obtain a theoretical solution. Even if a theoretical solution is obtained, if it is expressed in an infinite series, errors will occur due to quantification, and the accuracy may be lower than the numerical calculation itself [14]. In addition, in a kind of conformal mapping problem (or Cauchy-Riemann problem), the algorithm associated with the digitization of the theoretical solution becomes complicated, and considerable calculation time may be required [11]. Even in such a case, it is very convenient if there is an algorithm that can evaluate the numerical calculation error by numerical analysis. This problem is the theme of this paper. Such an argument seems to be a new attempt. In high-accuracy, numerical calculation by the IFDM, there is a well-ordered structure in the numerical calculation error, and for this reason, it is shown that the error of the numerical calculation can be evaluated even if the exact discrete solution of the numerical calculation is not known.

The paper directly related to this paper is reference [16], which details three types of high-order accuracy numerical schemes,

(i) Standard algebraic polynomial interpolation, SAPI(m) scheme,

(ii) Second-order-based interpolation, SOBI(m) scheme,

(iii) Compact interpolation finite difference, CIFD(m) scheme,

and illustrates how calculation errors occur. (m) shows the accuracy order of each scheme. The error evaluation method shown in this paper can be applied to any of the above three schemes, but it is limited here to numerical calculation by CIFD (m). Section 2 outlines the derivation of the high-order, accuracy difference scheme reported in the previous paper. In Section 3, the theme of this paper is described to focus on the well-ordered structure of calculation errors and to show that it is possible to evaluate errors without assuming an exact solution. In Section 3-1, as an example, the forcing term of the 1D Poisson equation is described as an aperiodic function, and in Section 3-2, it is described with a periodic function. Section 4 shows that even if there are several discrete output points, the entire solution is expressed as a continuous quantity by each of piecewise high-order interpolation polynomials that are almost equivalent to the theoretical solution itself. Section 5 outlines the content

of this paper.

2. Derivation of finite difference equations for high-order accuracy calculations

In this paper, the 1D Poisson equation is described as follows.

$$\nabla_{(1)}^2 u \equiv \frac{\partial^2 u}{\partial x^2} \equiv u_{xx} \equiv u_{2x} = f(x) \tag{1}$$

It is conventionally abbreviated that $\partial^2 u/\partial x^2 \equiv u_{xx}$, but since many higher-order derivative notations are required in this paper, it is expressed as $\partial^2 u/\partial x^2 \equiv u_{2x}$, and $f(x)$ is often abbreviated as f. Three schemes, SAPI(m), SOBI(m), and CIFD(m), are defined as high-order accuracy difference schemes by the IFDM in Eq. (1) [16]. Here, the explanation of SAPI(m) is omitted and starts from SOBI(m). The finite difference equation (FDE) by SOBI(m) in Eq. (1) is expressed as follows.

$$\delta_x^2 u_i + u_i(m) = f_i, m = 2, 4, \cdots, m_{max}, \delta_x^2 u_i \equiv \frac{u_{i-1} - 2u_i + u_{i+1}}{h^2} \tag{2}$$

$u_i(m)$ is an additional term to the second-order difference, meaning the higher-order difference correction term. When using the central difference in evenly spaced grid points, the accuracy order m is even [16]. There is no theoretical upper limit for m, but under double-precision calculation, when m becomes larger than a certain value, improvement in calculation accuracy cannot be expected. This de facto upper limit for m, m_{max} varies depending on the calculation conditions but is easily confirmed by several trial calculations.

We derive the second-order central difference. The following equations are obtained by Taylor expansion.

$$u(x+h) = u(x) + hu_x + \frac{h^2}{2!}u_{2x} + \frac{h^3}{3!}u_{3x} + \frac{h^4}{4!}u_{4x} + \cdots \tag{3}$$

$$u(x-h) = u(x) - hu_x + \frac{h^2}{2!}u_{2x} - \frac{h^3}{3!}u_{3x} + \frac{h^4}{4!}u_{4x} + \cdots \tag{4}$$

The following second-order difference scheme is obtained by adding both equations:

$$u_{2x} = \frac{u(x+h) - 2u(x) + u(x-h)}{h^2} + O(h^2) \tag{5a}$$

$$O(h^2) = -2\frac{h^2}{4!}u_{4x} - 2\frac{h^4}{6!}u_{6x} - 2\frac{h^6}{8!}u_{8x} - 2\frac{h^8}{10!}u_{10x} \cdots \tag{5b}$$

We change the above expression to the FD scheme expression and obtain the following expression.

$$u_{i,2x} = \delta_x^2 u_i + O_i(h^2) \tag{6a}$$

$$O_i(h^2) = -2\frac{h^2}{4!}u_{i,4x} - 2\frac{h^4}{6!}u_{i,6x} - 2\frac{h^6}{8!}u_{i,8x} - 2\frac{h^8}{10!}u_{i,10x}\cdots \tag{6b}$$

When $O_i(h^2)$ is disregarded in Eq. (6), Eq. (2) becomes $\delta_x^2 u_i = f_i$ with $u_i(2) = 0$, resulting in the usual second-order accuracy FDE. When the first term on the right-hand side is adopted in Eq. (6b), Eq. (2) obtains the following expression.

$$\delta_x^2 u_i + u_i(4) = f_i, u_i(4) = -2\frac{h^2}{4!}u_{i,4x} \tag{7}$$

As previously described, up to the 10th order accuracy, $u_i(m)$ can be summarized as follows.

$$u_i(2) = 0 \tag{8a}$$
$$u_i(4) = -2(h^2/4!)u_{i,4x} \tag{8b}$$
$$u_i(6) = -2(h^2/4!)u_{i,4x} - 2(h^4/6!)u_{i,6x} \tag{8c}$$
$$u_i(8) = -2(h^2/4!)u_{i,4x} - 2(h^4/6!)u_{i,6x} - 2(h^6/8!)u_{i,8x} \tag{8d}$$
$$u_i(10) = -2(h^2/4!)u_{i,4x} - 2(h^4/6!)u_{i,6x} - 2(h^6/8!)u_{i,8x} - 2(h^8/10!)u_{i,10x} \tag{8e}$$

The method of calculation based on this equation was defined as SOBI(m) [16]. To calculate with this method, $u_{i,4x}, u_{i,6x}, \cdots, u_{i,mx}$ must be defined. For details, refer to a previous paper [16]. This method is considered to be theoretically equivalent to the SAPI(m), but in the high-order accuracy calculation, the digit-loss error is reduced by the SOBI(m) calculation, and a higher accuracy calculation is expected. In this paper, the calculation by these methods is omitted.

We are currently considering the Poisson equation. From the characteristics of the Poisson equation, the expression $u_{i,nx} = f_{i,(n-2)x}$ is generally obtained, and Eq. (8) is modified as follows.

$$\tau_i(2) = 0 \tag{9a}$$
$$\tau_i(4) = -2(h^2/4!)f_{i,2x} \tag{9b}$$
$$\tau_i(6) = -2(h^2/4!)f_{i,2x} - 2(h^4/6!)f_{i,4x} \tag{9c}$$
$$\tau_i(8) = -2(h^2/4!)f_{i,2x} - 2(h^4/6!)f_{i,4x} - 2(h^6/8!)f_{i,6x} \tag{9d}$$
$$\tau_i(10) = -2(h^2/4!)f_{i,2x} - 2(h^4/6!)f_{i,4x} - 2(h^6/8!)f_{i,6x} - 2(h^8/10!)f_{i,8x} \tag{9e}$$

In this case, Eq. (2) is expressed as follows by changing the additional term $u_i(m)$ to $\tau_i(m)$.

$$\delta_x^2 u_i + \tau_i(m) = f_i, m = 2, 4, \cdots, m_{max} \tag{10}$$

This change in notation is very important. The area $[x_{i-1} - x_{i+1}]$ where the second-order FD scheme is defined is referred to as the compact area. For example, in Eq. (8b), $u_{i,4x}$ must be determined by setting a fourth-degree interpolation polynomial. Since there are 5 calculation points, the definition points extend beyond the compact area. The same condition holds for others, and the local coordinate origin of the interpolation polynomial must be changed depending on the calculation point [16]. However, when Eq. (9) is applied, the interpolation polynomial can be processed in the compact area in any case, and its local origin coincides with the calculation point. Furthermore, the degree of the interpolation polynomial is also m-2. If $f(x)$ is an analytic function that is easy to differentiate, it is possible to use the analytical values for $f_{i,2nx}, n = 1,2,\cdots$. However, there are cases where differentiation is not easy with complicated analytic functions. Therefore, to create a general-purpose system, we have to use the method that uses an interpolation polynomial.

For $\tau_i(4)$ in Eq. (9b), we set the following three points over the compact area:

$$p_{i-1}(x_{i-1}, f_{i-1}), p_i(x_i, f_i), p_{i+1}(x_{i+1}, f_{i+1})$$

We obtain the following quadratic interpolation equation with the calculation point as the coordinate origin:

$$f = b_0 + b_1 x + b_2 x^2 \tag{11}$$

The equation is easily calculated and defined as $f_{i,2x} = 2!\, b_2 + O_i(h^2)$; then, $O_i(h^2)$ is negligible and becomes $\tau_i(4) = -2b_2(2!/4!)h^2$. For $\tau_i(6)$, we obtain the following quartic interpolation curve with the calculation point as the coordinate origin:

$$f = b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 \tag{12}$$

It holds that $f_{i,2x} = 2!\, b_2 + O_i((h/2)^4)$ and $f_{i,4x} = 4!\, b_4 + O_i((h/2)^2)$; therefore, the equation becomes $\tau_i(6) = -2b_2(2!/4!)h^2 - 2b_4(4!/6!)h^4$. In the same manner, the whole is expressed as follows:

$$\tau_i(2) = 0 \tag{13a}$$

$$\tau_i(4) = -2b_2h^2/(3 \cdot 4) \tag{13b}$$

$$\tau_i(6) = -2b_2h^2/(3 \cdot 4) - 2b_4h^4/(5 \cdot 6) \tag{13c}$$

$$\tau_i(8) = -2b_2h^2/(3 \cdot 4) - 2b_4h^4/(5 \cdot 6) - 2b_6h^6/(7 \cdot 8) \tag{13d}$$

$$\tau_i(10) = -2b_2h^2/(3 \cdot 4) - 2b_4h^4/(5 \cdot 6) - 2b_6h^6/(7 \cdot 8) - 2b_8h^8/(9 \cdot 10) \tag{13e}$$

These expressions have a kind of regularity, and a general-purpose system can be easily created using

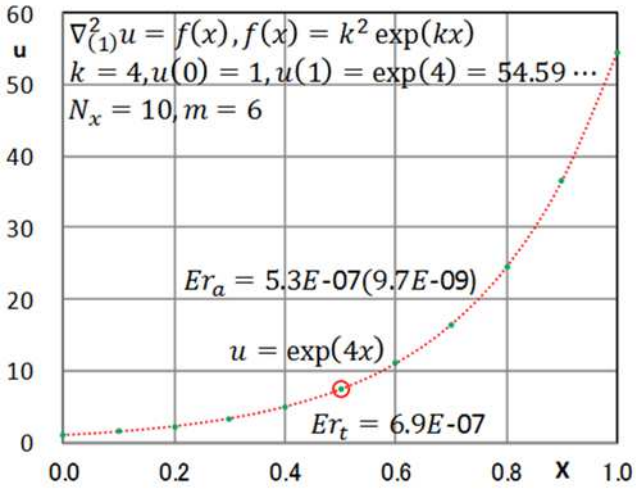


Fig. 1 Calculation example with aperiodic function, $N_x = 10, m = 6$

m of $\tau_i(m)$ as a parameter. The scheme calculated in this way is also a form of the SOBI scheme but is especially referred to as the compact IFD (CIFD) scheme.

$\tau_i(m)$ changes depending on the calculation point. The polynomial for f_i of degree m-2 may be calculated each time with the simultaneous equations, but the Vandermonde matrix, VM(m), is isomorphic at all calculation points in this case. Once the inverse matrix IVM(m) is calculated, the polynomial can be easily calculated from IVM(m) [16]. The latter should be adopted when creating a general-purpose system.

The numerical solution is obtained at the fastest speed by the direct method using the tridiagonal matrix algorithm TDMA-direct, and no matter how large the division number N_x is, the calculation is instantly completed regardless of the division number N_x [13].

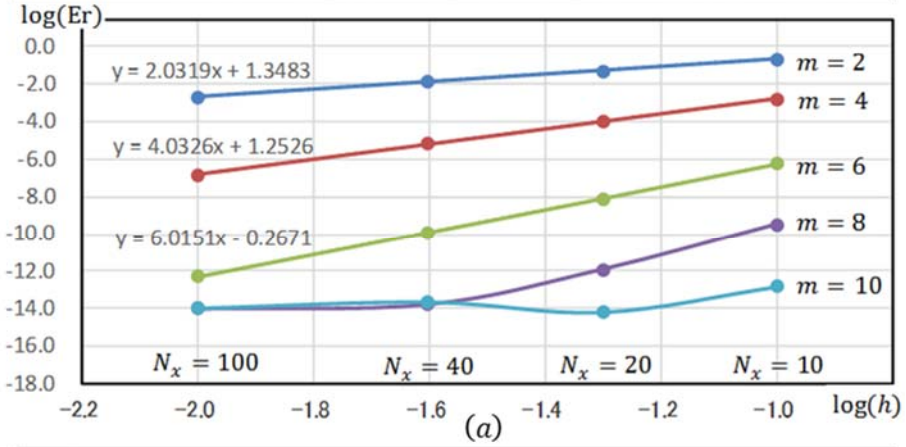
3. Ordered structure of numerical calculation error

3-1 When $f(x)$ is an aperiodic function

Figure 1 shows an example of the numerical calculation of $f(x) = k^2\exp(kx), k = 4$. The calculation conditions are shown in the figure. The division number is $N_x = 10$ and $h = 0.1$. The theoretical solution is $u = \exp(4x)$. In the problems addressed in this paper, the numerical solution and exact solution cannot be visually distinguished. Therefore, only the former is shown. In the calculation with $(N_x, m) = (10, 6)$, the overall mean error is $Er_a = 5.3E$-07 (=5.3×10^{-7}). The overall mean error is defined as $Er_a = \sum_{i=1}^{N_x-1}|u_i - U_i|/(N_x - 1)\ (> 0)$. $U_i = U(x_i)$ is an exact solution.

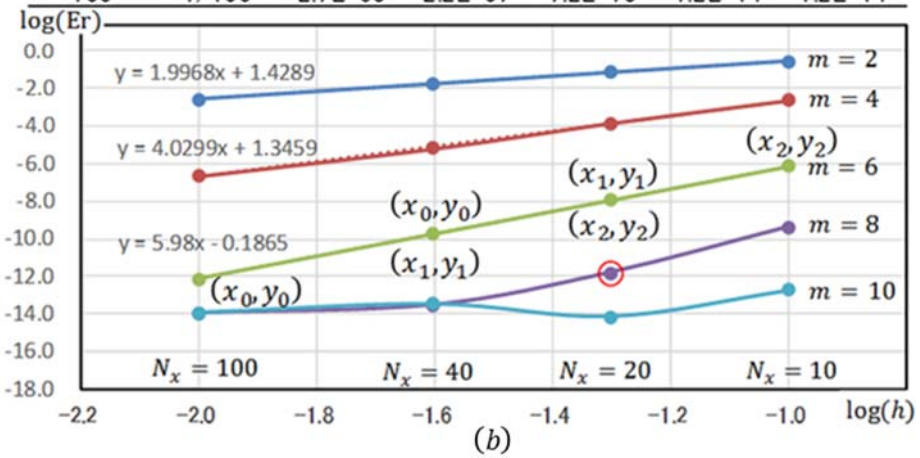
To properly evaluate the numerical calculation error, individual errors are added and averaged, so it may be effective to divide the above error by the maximum absolute variable value in the domain, which is defined as a relative error. Next, the error becomes $Er'_a = 5.3E$-07/exp(4)=9.7E-09. It is

N_x	h	2	4	6	8	10
10	1/10	2.1E-01	1.7E-03	5.3E-07	3.1E-10	1.5E-13
20	1/20	5.0E-02	1.0E-04	8.0E-09	1.2E-12	5.9E-15
40	1/40	1.2E-02	6.1E-06	1.2E-10	1.7E-14	2.2E-14
100	1/100	1.9E-03	1.6E-07	5.1E-13	9.9E-15	9.9E-15



(a)

N_x	h	2	4	6	8	10
10	1/10	2.7E-01	2.2E-03	6.9E-07	4.1E-10	1.9E-13
20	1/20	6.8E-02	1.4E-04	1.1E-08	1.6E-12	7.1E-15
40	1/40	1.7E-02	6.1E-06	1.7E-10	3.0E-14	3.6E-14
100	1/100	2.7E-03	2.2E-07	7.2E-13	1.2E-14	1.2E-14



(b)

Fig. 2 Calculation example of aperiodic function, log-log error diagram, CIFD(m), (a) overall mean error, (b) individual error at designated point ($x = 0.5$)

understood that the significant figures in the calculation are approximately 8 digits. At this time, an individual error occurs at each calculation point. The error at the calculation point, $x = 0.5$, is $Er_t = u_5 - U_5 = 6.9E\text{-}07$. This error indicates a deviation from the theoretical value U_5; its absolute value is adopted in the log-log error diagram, which will be subsequently described. Er_a and Er_t have the same order of error.

Figure 2 shows a log-log error diagram as a result of calculation, including other cases. Figure 2(a) shows the overall mean error, which is expected to be $Er_{(m)} = C_{(m)}h^m$. Taking the common logarithm

of each side, this error becomes $\log(Er_{(m)}) = m\log(h) + \log(C_{(m)})$, which is usually a straight line in log-log representations. We define this line as an (actual) accuracy power (AP-) line. In CIFD(2), the AP-line is $\log(Er_{(2)}) = \alpha_{(2)}\log(h) + \beta_{(2)} = 2.0319\log(h) + 1.3483$ (the subscript (m) is often omitted). The reason the line becomes $\alpha_{(2)} = 2 + \epsilon$ (where ϵ is a small positive value) is the influence of higher-order terms with truncation errors [14]. In the error diagram, we generally express $\log(Er_{(m)}) = \alpha_{(m)}\log(h) + \beta_{(m)}$. We define $\alpha_{(m)}$ as the accuracy power, which is generally expressed as $\alpha_{(m)} = m + \epsilon(> 0)$. If the number of divisions is multiplied by n, the error is expressed as $Er_{(m)}{}' = C_{(m)}(h/n)^{\alpha_{(m)}} = Er_{(m)}/n^{\alpha_{(m)}}$. We define this error as the accuracy power law. When defining an approximate value, $\alpha_{(m)} = m$ may be assumed. We define $\beta_{(m)}$ as the basic accuracy factor. Assuming $h = 0.1$, $\log(Er_{(m)}) = \beta_{(m)} - \alpha_{(m)}$, the error becomes $10^{\beta_{(m)} - \alpha_{(m)}}$. We define this as the basic error. The AP line is clearly defined in CIFD(m), where m = 2, 4, and 6, but the AP line is not clearly defined in CIFD(m), where m = 8 and 10. This phenomenon occurs because it is a double precision calculation, and the AP line is always defined in the calculation where the effective digit can be arbitrarily specified [15,16].

Figure 2(b) shows a log-log error diagram of the individual error at $x = 0.5$. Individual errors occur in both positive values and negative values corresponding to each calculation, but absolute values are used for drawing. Fig. 2(a) and Fig. 2(b) are similar. Usually, error evaluation is premised on the existence of a theoretical solution. However, from the theoretical feature of individual error, the error can be evaluated without assuming the theoretical solution. This is the theme of this paper. In the AP line with CIFD (6) of Fig. 2 (b), the coordinate values of the following three points

$$p_0(x_0, y_0), p_1(x_1, y_1), p_2(x_2, y_2)$$

are defined in log-log coordinates. First, let us consider the case of $x_0 = \log(1/40)$. For a "strict" AP line to hold, the following equation must hold.

$$\frac{y_2 - y_0}{x_2 - x_0} = \frac{y_1 - y_0}{x_1 - x_0}, x_0 = \log(1/40), x_1 = \log(1/20), x_2 = \log(1/10) \tag{14}$$

Referring to Fig. 3, the following two cases can be considered:

Case (1): Fig. 3(a) shows the case where the exact value v exists above.

Case (2): Fig. 3(b) shows the case where the exact value v exists below.

u_0, u_1, and u_2 are the values determined by numerical calculation. In the range in which the AP line holds, it is limited to both cases, that is, v does not occur between u_0 and u_2. In both cases of Fig. 3, Eq. (14) is expressed as follows.

$$\frac{\log|v - u_2| - \log|v - u_0|}{x_2 - x_0} = \frac{\log|v - u_1| - \log|v - u_0|}{x_1 - x_0} \tag{15}$$

The following function is defined to obtain the true value v by Newton's method.

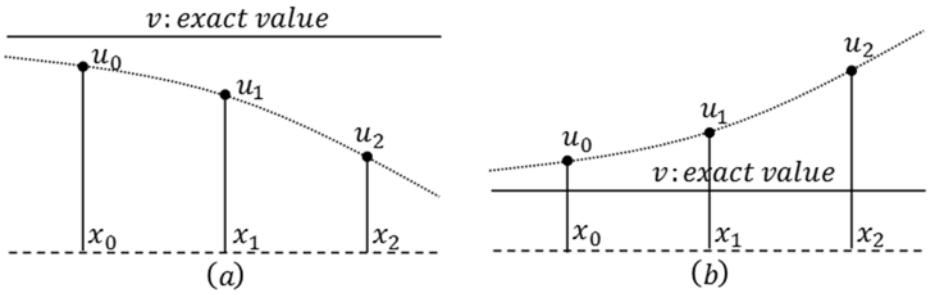


Fig. 3 Conceptual diagram of error-two patterns of deviation from the exact value

$$y \equiv f(v) = \frac{\log|v-u_2| - \log\ |v-u_0|}{x_2 - x_0} - \frac{\log|v-u_1| - \log|v-u_0|}{x_1 - x_0} \tag{16}$$

To adopt Newton's method, the first derivative value must be determined; in both cases, it is expressed as follows.

$$y' = \frac{u_2 - u_0}{(v-u_2)(v-u_0)(x_2 - x_0)\ln(10)} - \frac{u_1 - u_0}{(v-u_1)(v-u_0)(x_1 - x_0)\ln(10)} \tag{17}$$

where ln is the natural logarithm. In this study, the initial value by Newton's method is $v_0 = u_0 + \epsilon(= 10^{-15})$ in Fig. 3(a) and $v_0 = u_0 - \epsilon(= 10^{-15})$ in Fig. 3(b).

The calculation results are shown in Table 1(c). v_t is the exact solution, the true value, indicated by 15 significant digits. v_c is a convergent solution determined by Newton's method. The convergence condition by Newton's method is $|v_n - v_{n-1}| \leq \epsilon(= 10^{-15})$. In this case, the number of convergence iteration calculations is $n\ = 10$. Other calculations are similar. Er_t is the error with respect to the true value. $v_c - v_t$=7.1E-13 is shown at the bottom. In this case, the estimated value expected by the numerical calculation has an error on the order of 10^{-13}. The error is 1/100 or smaller than the true error of u_0. Er_c is the error for v_c. Note that Er_t and Er_c are equivalent in the double digit display of the error. The error is evaluated without assuming the theoretical value.

The above findings are obtained via numerical error analysis by Newton's method, but it is possible to directly estimate the approximate value of the error. Er_s is a calculated value of $u_1 - u_0$ and $u_2 - u_0$. In this case, the error evaluation of u_0 is impossible, but each error of u_1 and u_2 is the same result in the double digit comparison compared with Er_t.

When it is calculated as $N_x \times 2^n, n = 1,2,3\cdots$, the numerical calculation absolute error decreases as n increases. The case of Er_s shows that the approximate value of the error of N_x and $2N_x$ can be grasped by comparing the numerical calculation result of $4N_x$.

Table 1(c) also shows the calculated values of α and β, which are the estimated values of the AP line. It is confirmed that there is not much difference from the AP-line, α, and β of CIFD (6)

Table 1 Analysis table of individual error; refer to Fig. 2(b)

$(a)\ m=2$					
h	v_t	7.38905609893065	Er_t	Er_c	Er_s
1/40	u_0	7.40605594722765	1.7E−02	1.7E−02	***
1/20	u_1	7.45695364420706	6.8E−02	6.8E−02	5.1E−02
1/10	u_2	7.65902636146620	2.7E−01	2.7E−01	2.5E−01
***	v_c	7.38891967991964	−1.4E−04	α=1.989	β=1.421
$(b)\ m=4$					
h	v_t	7.38905609893065	Er_t	Er_c	Er_s
1/40	u_0	7.38904759812962	−8.5E−06	−8.5E−06	***
1/20	u_1	7.38892024781491	−1.4E−04	−1.4E−04	−1.3E−04
1/10	u_2	7.38689277589765	−2.2E−03	−2.2E−03	−2.2E−03
***	v_c	7.38905613342685	3.4E−08	α=3.993	β=1.328
$(c)\ m=6$					
h	v_t	7.38905609893065	Er_t	Er_c	Er_s
1/40	u_0	7.38905609876202	−1.7E−10	−1.7E−10	***
1/20	u_1	7.38905608815057	−1.1E−08	−1.1E−08	−1.1E−08
1/10	u_2	7.38905541260940	−6.9E−07	−6.9E−07	−6.9E−07
***	v_c	7.38905609893136	7.1E−13	α=5.992	β=-0.171
$(d)\ m=8$					
h	v_t	7.38905609893065	Er_t	Er_c	Er_s
1/22	u_0	7.38905609892988	−7.7E−13	−7.0E−13	***
1/20	u_1	7.38905609892906	−1.6E−12	−1.5E−12	−8.2E−13
1/18	u_2	7.38905609892696	−3.7E−12	−3.6E−12	−2.9E−12
***	v_c	7.38905609893058	−7.0E−14	α=8.218	β=-1.125

in Fig. 2. The case of CIFD (6) has been explained above, but the same investigation was performed for CIFD (m), where m = 2 and 4. The results are shown in Tables 1(a) and (b). As with CIFD (6), the expected results are obtained. It is understood that the estimated value v_c becomes more accurate as the m value becomes larger.

Next, we consider the case of $x_0 = \log(1/100)$ in Fig. 2. The results are shown in Table 2. It is observed that the accuracy is improved as a whole compared with the results in Table 1. Focusing again on the AP line of m = 6, it is assumed that the four points are exactly positioned on a straight line. The derived $\alpha_{(6)}, \beta_{(6)}$ and v_c must also be equivalent. However, this is not the case. Even if the AP line appears to be a straight line, it is a regression line, and each data point in the log-log expression is not strictly defined by a single line. We intuitively obtain the reasonable conclusion that a group with numerous divisions can obtain a more accurate estimation than a group with a small number of divisions.

In Fig. 2(b), m = 8, and the AP line is not shown because the calculation of $N_x = 40$ is the VE0 calculation. In this case, Table 1(d) shows the results of estimating the error from the calculated values of $N_x = 22, 20, 18$ to confirm the calculation accuracy near $N_x = 20$. At m = 2, 4, and 6, the error

evaluations of Er_t and Er_c were identical in the double digit display. However, in this case, the evaluations were slightly different. It can be said that Er_s is effective as a reference value for error. Let us define the method here as the three-point method for error estimation.

Here, we focused on the center point. (The condition for evaluating the error at the center point is that N_x must be even.) The above method can be applied to locations other than the center point, as shown in the next section. However, since Figs. 2(a) and (b) are similar, a similar error occurs as an order of error at other calculation points.

Table 2 Analysis table of individual error; refer to Fig. 2(b)

$(a)\ m=2$					
h	v_t	7.38905609893065	Er_t	Er_c	Er_s
1/100	u_0	7.39177721709330	2.7E−03	2.7E−03	***
1/40	u_1	7.40605594722765	1.7E−02	1.7E−02	1.4E−02
1/20	u_2	7.45695364420706	6.8E−02	6.8E−02	6.5E−02
***	v_c	7.38904985247626	−6.2E−06	α=1.997	β=1.431
$(b)\ m=4$					
h	v_t	7.38905609893065	Er_t	Er_c	Er_s
1/100	u_0	7.38905588123759	−2.2E−07	−2.2E−07	***
1/40	u_1	7.38904759812962	−8.5E−06	−8.5E−06	−8.3E−06
1/20	u_2	7.38892024781491	−1.4E−04	−1.4E−04	−1.4E−04
***	v_c	7.38905609921702	2.9E−10	α=3.998	β=1.335
$(c)\ m=6$					
h	v_t	7.38905609893065	Er_t	Er_c	Er_s
1/100	u_0	7.38905609892993	−7.2E−13	−6.9E−13	***
1/40	u_1	7.38905609876202	−1.7E−10	−1.7E−10	−1.7E−10
1/20	u_2	7.38905608815057	−1.1E−08	−1.1E−08	−1.1E−08
***	v_c	7.38905609893062	−3.0E−14	α=5.998	β=−0.164

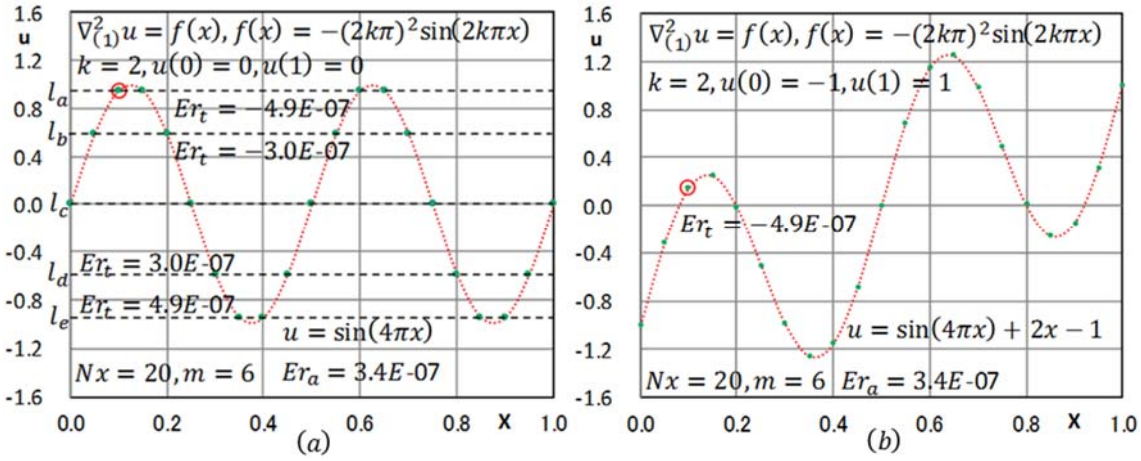
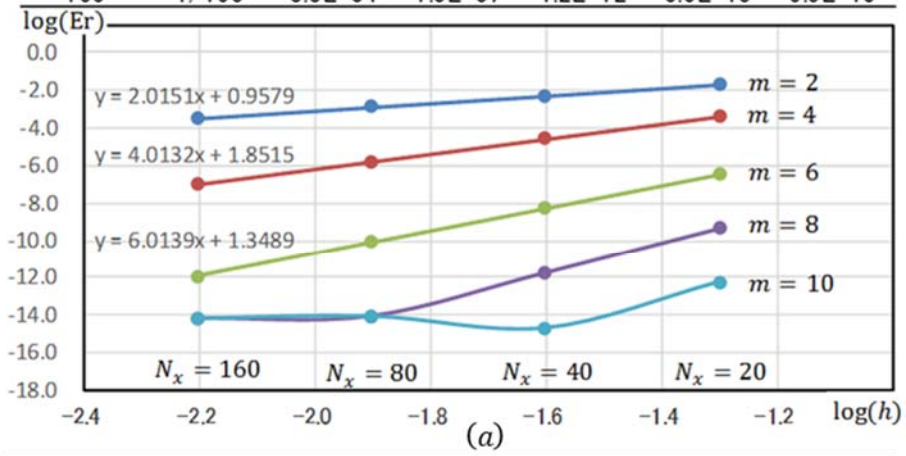


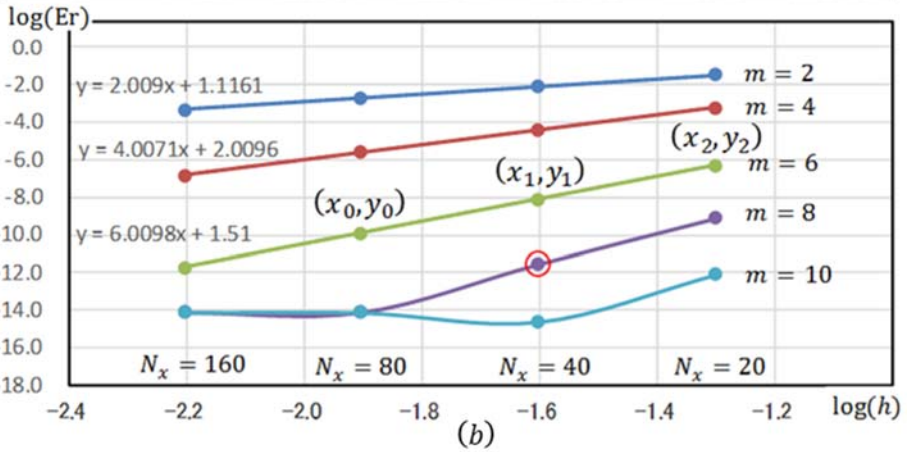
Fig. 4 Calculation examples with different Dirichlet boundary conditions, $N_x = 20, m = 6$

N_x	h	2	4	6	8	10
20	1/20	2.2E-02	4.3E-04	3.4E-07	4.9E-10	5.5E-13
40	1/40	5.4E-03	2.6E-05	5.2E-09	1.9E-12	2.1E-15
80	1/80	1.3E-03	1.6E-06	8.0E-11	8.6E-15	8.6E-15
160	1/160	3.3E-04	1.0E-07	1.2E-12	6.6E-15	6.5E-15



(a)

N_x	h	2	4	6	8	10
20	1/20	3.2E-02	6.3E-04	4.9E-07	7.2E-10	8.0E-13
40	1/40	7.9E-03	3.9E-05	7.6E-09	2.8E-12	2.4E-15
80	1/80	2.0E-03	2.4E-06	1.2E-10	7.7E-15	7.7E-15
160	1/160	4.9E-04	1.5E-07	1.8E-12	7.7E-15	7.7E-15



(b)

Fig. 5 Calculation example of periodic function, Log-log error diagram, CIFD(m), (a) overall mean error, (b) individual error at designated point ($x = 0.1$)

3-2. When $f(x)$ is a periodic function

Figure 4 (a) shows an example when $f(x)$ is a periodic function: $f(x) = -(2k\pi)^2 \sin(2k\pi x)$. The general solution is $u = \sin(k\pi x) + c_0 + c_1 x$, but Fig. 4(a) is calculated by specifying the boundary condition as $u(0) = u(1) = 0$. By specifying $u(0) = -1, u(1) = 1$, Fig. 4(b) is obtained. Since the calculation error is determined regardless of the boundary conditions, the examination results for Fig. 4(a) are sufficiently general. In this case, if the center point is specified, the error becomes zero

Table 3 Analysis table of individual error; refer to Fig. 4(b)

$(a)\ m=2$					
h	v_t	0.951056516295154	Er_t	Er_c	Er_s
1/80	u_0	0.953014462778702	2.0E−03	1.9E−03	***
1/40	u_1	0.958917394960781	7.9E−03	7.8E−03	5.9E−03
1/20	u_2	0.982972442829758	3.2E−02	3.2E−02	3.0E−02
***	v_c	0.951094873407014	3.8E−05	α=2.027	β=1.140
$(b)\ m=4$					
h	v_t	0.951056516295154	Er_t	Er_c	Er_s
1/80	u_0	0.951058931199761	2.4E−06	2.4E−06	***
1/40	u_1	0.951095268645021	3.9E−05	3.9E−05	3.6E−05
1/20	u_2	0.951683937566720	6.3E−04	6.3E−04	6.3E−04
***	v_c	0.951056540588354	2.4E−08	α=4.018	β=2.024
$(c)\ m=6$					
h	v_t	0.951056516295154	Er_t	Er_c	Er_s
1/80	u_0	0.951056516176914	−1.2E−10	−1.2E−10	***
1/40	u_1	0.951056508703413	−7.6E−09	−7.6E−09	−7.5E−09
1/20	u_2	0.951056024050242	−4.9E−07	−4.9E−07	−4.9E−07
***	v_c	0.951056516293962	−1.2E−12	α=6.019	β=1.523
$(d)\ m=8$					
h	v_t	0.951056516295154	Er_t	Er_c	Er_s
1/50	u_0	0.951056516295620	4.7E−13	4.6E−13	***
1/40	u_1	0.951056516297931	2.8E−12	2.8E−12	2.3E−12
1/30	u_2	0.951056516323027	2.8E−11	2.8E−11	2.7E−11
***	v_c	0.951056516295157	3.0E−15	α=8.218	β=1.292

regardless of the number of divisions, so error analysis cannot be performed. When we specify the error evaluation point as $x = 0.1$, the number of adopted divisions is $N_x = 10n, n = 1,2,\cdots$. When this number is calculated by specifying $N_x = 20, x = 0.1$, it becomes $Er_t = -4.9E$-07. At this time, all the errors on l_a (dotted horizontal line) are equivalent. On l_b, the error is $Er_t = -3.0E$-07; on l_c, and all errors are zero. For l_d, $Er_t = 3.0E$-07; and for l_e, $Er_t = 4.9E$-07. The overall mean error is $Er_a = 3.4E$-07.

Figure 5(a) shows the log-log diagram of the overall mean error, and Fig. 5(b) shows the log-log diagram when x = 0.1 is specified. Table 3 is an error analysis diagram calculated in the same way as Table 1, and it is observed that the equivalence of Er_t and Er_c almost completely holds in this case.

4. Interpolation of CIFD(m) calculation

In numerical calculation by the IFDM, not only is the calculation result output at each discrete point, but also the solution is defined as a continuous quantity by piecewise interpolation polynomials. In SAPI (m) and SOBI(m), there is no problem because the interpolation polynomial related to the

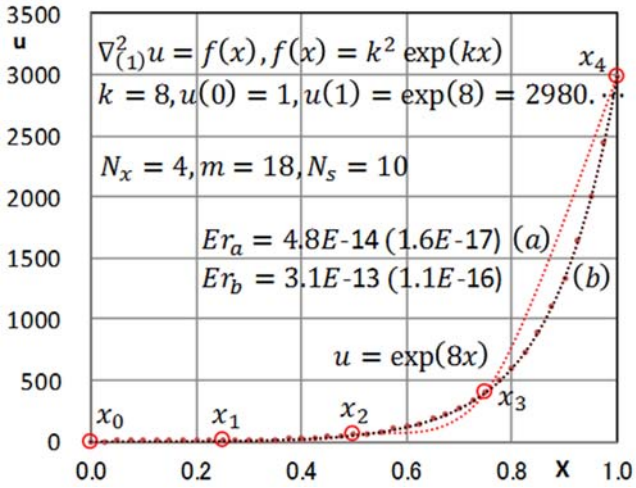


Fig. 6 Discrete point calculation (a) by CIFD (18) and display of the interpolation curve (b)

variable u is directly defined. However, in CIFD(m), it is necessary to separately prepare a simple routine to derive the interpolation polynomial related to the variable u.

Figure 6 shows the results when $f(x) = k^2\exp(kx), k = 8$ and is calculated by $N_x = 4, m = m_{max} = 18$. In this case, Er_a = 4.8E-14(1.6E-17), and of course, it is the VE0 calculation. The cubic spline curve in this case is shown, but it is not usable. Now, let us assume that the calculation is performed by m=4. Equation (11) has already been defined with the calculation point as the coordinate origin. The second-order integral would generally be expressed as:

$$\iint f\,dxdx \equiv u = c_0 + c_1x + \frac{b_0}{1\times 2}x^2 + \frac{b_1}{2\times 3}x^3 + \frac{b_2}{3\times 4}x^4 \tag{18}$$

Here, by definition, $c_0 = u_i$. In addition, the following relationship must be established.

$$u_{i+1} = u_i + u_{i,x}h + \frac{b_0}{1\times 2}h^2 + \frac{b_1}{2\times 3}h^3 + \frac{b_2}{3\times 4}h^4 \tag{19}$$

$$u_{i-1} = u_i - u_{i,x}h + \frac{b_0}{1\times 2}h^2 - \frac{b_1}{2\times 3}h^3 + \frac{b_2}{3\times 4}h^4 \tag{20}$$

From both equations, the following conclusions can be obtained.

$$u_{i,x} = \frac{u_{i+1} - u_{i-1}}{2h} - \frac{b_1}{2\times 3}h^3 \tag{21}$$

u_{i+1} and u_{i-1} are the values determined by numerical calculation. Equalizing the right-hand side c_1 of Eq. (18) and Eq. (21), $c_1 = u_{i,x}$, and the quaternary interpolation equation in the compact area is determined.

It is easy to generalize the above argument with the accuracy order m. Equation (18) is generalized and expressed as follows.

$$u = u_i + u_{i,x}x + \sum_{n=0}^{m-2} \frac{b_n}{(n+1)(n+2)} x^{n+2} \tag{22}$$

Equation (21) is generalized and expressed as follows.

$$u_{i,x} = \frac{u_{i+1} - u_{i-1}}{2h} - \sum_{n=1,3,\cdots}^{m-3} \frac{b_n}{(n+1)(n+2)} h^{n+2} \tag{23}$$

In this way, the piecewise continuous interpolation polynomial $P_i(m)$ is determined. A similar argument can be made by starting with the Taylor series (Maclaurin series) below.

$$u(x) = u_i + xu_{i,x} + \frac{x^2}{2!}u_{i,2x} + \frac{x^3}{3!}u_{i,3x} + \frac{x^4}{4!}u_{i,4x} + \cdots \tag{24}$$

Since it is a Poisson equation, the following equation holds.

$$u(x) = u_i + xu_{i,x} + \frac{x^2}{2!}f_i + \frac{x^3}{3!}f_{i,x} + \frac{x^4}{4!}f_{i,2x} + \cdots \tag{25}$$

The following argument process is identical to the above process.

Curve (b) in Fig. 6 shows the results of dividing h into N_s =10 equal parts. When $x_0 < x < x_1$, the value of $P_1(m)$ is output. When $x_1 < x < x_2$, the mean value of the value of $P_1(m)$ and the value of $P_2(m)$ is output. The overall mean error was Er_b = 3.1E-13(1.1E-16), and almost all of the calculated individual values become zero error, that is, it is the VE0 calculation.

The subdivision number N_s can be arbitrarily designated. At that time, if N_s increases to a certain extent, the value of Er_b hardly changes. For example, when $N_s = 20$, Er_b = 3.3E-13(1.1E-16).

From experience, when an analytic function is expressed by a polynomial, the Runge phenomenon may occur. It is not possible to clearly define the boundary between the two. However, the exponential function is not associated with the Runge phenomenon. In such a case, the VE0 calculation can be easily performed by reducing N_x and increasing m, as in the case described here. The value of N_s does not affect the calculation accuracy.

5. Conclusion

A new phase has arrived in the numerical calculation by the FDM, that is, in the numerical calculation by the IFDM, the VE0 calculation of the 1D Poisson equation became possible. Calculation examples are limited. However, this method can be applied by a general method as long as the forcing

term is an analytic function. There is no need to distinguish between a periodic function and an aperiodic function. In this paper, we focused on the ordered structure of numerical calculation errors and clarified that numerical calculation errors can be evaluated without assuming theoretical solutions. Presently, the above studies are limited to one-dimensional problems, but it is also necessary to consider their effectiveness in multidimensional problems. This issue will be investigated in the near future.

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] P.M. Morse and H. Feshbach, 2011, Methods of Theoretical Physics.
http://people.physics.tamu.edu/pope/methods_pages2011.pdf

[2] F. Gibou, R. P. Fedkiw, L. T. Cheng, and M. Kang, "A second-order-accurate symmetric discretization of the Poisson equation on irregular domains," Journal of Computational Physics, 176(1), 205–227 (2002).

[3] F. Gibou, R. P. Fedkiw, "A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem," J. Comput. Phys. 202, 577–601 (2005).

[4] Y. T. Ng, H. Chen, C. Min, and F. Gibou, "Guidelines for poisson solvers on irregular domains with Dirichlet boundary conditions using the ghost fluid method," J. Sci. Comput. 41, 300–320 (2009).

[5] R. S. Hirsh, "Higher order accurate difference solutions of fluid mechanics problems by a compact differencing technique," J. Comput. Phys. 19, 90–109 (1975).

[6] S. K. Lele, "Compact finite difference schemes with spectral-like resolution," J. Comput. Phys. 103, 16–42 (1992).

[7] A. J. Harfash and H. A. Jalob, "Sixth and fourth order compact finite difference schemes for two and three dimension Poisson equation with two methods to derive these schemes," Basrah J. Sci. A 24(2), 1–20 (2006).

[8] T. Fukuchi, "Numerical calculation of fully-developed laminar flows in arbitrary cross-sections using finite-difference method," AIP Advances 1(4), 042109 (2011).
http://dx.doi.org/10.1063/1.3652881

[9] T. Fukuchi, "Numerical stability analysis and rapid algorithm for calculations of fully developed laminar flow through ducts using time-marching method," AIP Advances 3, 032101 (2013).
http://dx.doi.org/10.1063/1.4794500.

[10] T. Fukuchi, "Finite difference method and algebraic polynomial interpolation for numerically solving Poisson's equation over arbitrary domains," AIP Advances 4, 060701 (2014).
http://dx.doi.org/10.1063/1.4885555.

[11] T. Fukuchi, “Numerical analyses of steady-state seepage problems using the interpolation finite difference method, soils and foundations,” The Japanese Geotechnical Society 54(6), 608–626 (2016). http://dx.doi.org/10.1016/j.sandf.2016.07.003

[12] T. Fukuchi, “New high-precision empirical methods for predicting the seepage discharges and free surface locations of earth dams validated by numerical analyses using the IFDM, Soils and Foundations,” The Japanese Geotechnical Society 58, 427–445 (2018). https://doi.org/10.1016/j.sandf.2018.02.011

[13] T. Fukuchi, “High-order accurate and high-speed calculation system of 1D Laplace and Poisson equations,” AIP Advances 9, 055312 (2019). https://doi.org/10.1063/1.5096395

[14] T. Fukuchi, “Higher order difference numerical analyses of a 2D Poisson equation by the interpolation finite difference method and calculation error evaluation,” AIP Advances 10, 125009 (2020) https://doi.org/10.1063/5.0018915

[15] T. Fukuchi, “Characteristic features of error in high-order difference calculation of 1D Poisson equation and unlimited high-accurate calculation under multi-precision calculation,” Mathematics and Computers in Simulation 190 (2021) 303–328, https://doi.org/10.1016/j.matcom.2021.05.011

[16] T. Fukuchi, “A whole high-accuracy numerical calculation system for the 1D Poisson equation by the interpolation finite difference method”, (2022), preprint, https://doi.org/10.21203/rs.3.rs-1739207/v1

[17] J. P. Boyd, Chebyshev and Fourier Spectral Methods, 2nd ed. (Dover Publications, Inc. 2001).