A RELAXATION METHOD FOR SOLVING ELLIPTIC DIFFERENCE EQUATIONS*

R. P. FEDORENKO

Moscow

(*Received* 20 *April* 1961)

§ 1

WHEN solving elliptic equations by the method of finite differences we have to deal with systems of linear algebraic equations, often of a very high order. Given a sufficiently high order of the system, the familiar iterative methods of solution of such systems are very slowly convergent. Numerous works have been devoted to methods of speeding up the convergence of the iterations. These speeding-up methods can be split provisionally into two groups. The first group includes methods which use the spectrum of the iterative operators; they are described in detail in text-books [1] and [2]. The second, rather indefinite group includes the so-called "relaxation" methods, that are based essentially on "intuition", the "computer's experience"; they are regarded as applicable for non-mechanical computation by a sufficiently experienced group of workers, but as little suited to being carried out on digital computers; it is usually suggested that the relaxation method can be extremely effective [3].

The present article describes a method that we have developed for iterative solution of elliptic difference equations, using an idea not unlike the relaxation method. The present method was put into practice on a digital computer and gave good results.

Le us take Poisson's equation in a rectangular domain:

$$\Delta u \equiv \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = D. \tag{1}$$

Certain boundary conditions are specified on the boundary of the domain.

We introduce into the domain a rectangular net with equal steps with respect to x and y: $\Delta x = \Delta y = 1$; the base-points of the net will be enumerated by pairs of numbers (m, n):

$$m = 0, 1, \ldots, M, M+1,$$
$$n = 0, 1, \ldots, N, N+1.$$

We pass in the usual way to a system of linear algebraic equations

$$(\Delta_{(r)} u)_{m,n} \equiv u_{m-1,n} + u_{m,n-1} - 4u_{m,n} + u_{m,n+1} + u_{m+1,n} = D_{m,n}. \tag{2}$$

This system is supplemented by the necessary number of "boundary conditions", which there is no need to write down. The familiar iterative process

$$u^{\nu+1}_{m,n} = u^{\nu}_{m,n} + \alpha(u^{\nu}_{m-1,n} + u^{\nu}_{m,n-1} - 4u^{\nu}_{m,n} + u^{\nu}_{m,n+1} + u^{\nu}_{m+1,n} - D_{m,n}) \tag{3}$$

is very slowly convergent for $0 < \alpha < \frac{1}{4}$. We shall try to use some special features of the convergence in order to speed it up.

Let $u_{m,n}$ be the strict solution of system (2); $u^{\nu}_{m,n}$ is the approximate solution obtained after ν iterations; $\mathbf{u}^{\nu}$ is the vector whose components are $u^{\nu}_{m,n}$; the iterative process may be written in the abbreviated form

$$\mathbf{u}^{\nu+1} = A_{\alpha}\mathbf{u}^{\nu} - \alpha\mathbf{D}.$$

* Zh. vych. mat. **1**: No. 5, 922–927, 1961.

A_α is the operator of passage from the ν-th iteration to the $(\nu+1)$-th. Let

$$z^\nu = \mathbf{u} - \mathbf{u}^\nu$$

be the error in the ν-th iteration.

Let $\mathbf{f}^{(\varphi, \psi)}$ be the eigenfunctions of the operator A_α, $\lambda_{\varphi, \psi}$ the corresponding eigenvalues (φ, ψ are integers).

Now

$$\mathbf{z}^\nu = \sum_{\varphi, \psi} c^\nu_{\varphi, \psi} \mathbf{f}^{(\varphi, \psi)} = \sum_{\varphi, \psi} c^0_{\varphi, \psi} \lambda^\nu_{\varphi, \psi} \mathbf{f}^{(\varphi, \psi)}. \tag{4}$$

We shall speak (very provisionally) of the eigenfunctions as "good" and "bad"; the good ones include those that are smooth on the net and have few changes of sign in the domain; the "bad" ones often change sign and oscillate rapidly. It is well known to every calculator that "bad" functions have no value from the point of view of a finite-difference method: a finite-difference solution that pretends to accuracy must be in essence composed of "good" functions. It is also well known that the greatest difficulties in iterative methods are bound up with quenching of the "good" components of the error of the initial approximation: annihilation of the "bad" components may be very rapid.

We shall take $\alpha \sim \frac{1}{8}$ in (3); this choice of α leads to the fact that, after a fairly small number of iterations, the error $\mathbf{z}^\nu$ will consist of "good" eigenfunctions $\mathbf{f}^{(\varphi, \psi)}$; the same is true for the discrepancy

$$\delta^\nu_{m, n} = (\Delta_{(r)} u^\nu)_{m, n} - D_{m, n}. \tag{5}$$

We shall use the following method to annihilate the "good" components of the error. We introduce into the domain an auxiliary net, the step of which is q times greater than the step of the original net; we shall assume that the base-points (k, l) of the auxiliary net coincide with the base-points of the original net (this is not in general necessary). We shall suppose for simplicity that the boundary base-points of the auxiliary net fall on the boundary of the domain (we shall subsequently renounce this condition also).

We shall solve in this auxiliary net the equation

$$(\Delta_{(r)} v^\nu)_{k.l} = q^2 \delta^\nu_{k, l}; \tag{6}$$

$\delta^\nu_{k, l}$ is equal to $\delta^\nu_{m, n}$ at the base-point of the original net which coincides with the base-point (k, l) of the auxiliary net.

The "boundary conditions" for system (6) correspond to homogeneous boundary conditions for system (2). Since the number of base-points of the auxiliary net is much smaller than the number of base-points of the original net, the computer operating time necessary for this can be reckoned insignificant. Moreover, there is no need for us to solve equation (6) too accurately. We now interpolate the function $v^\nu_{k, l}$ from the base-points of the auxiliary net to the base-points of the original (say linearly with respect to x and with respect to y) and subtract the function $v^\nu_{m, n}$ thus obtained from $u^\nu_{m, n}$.

Notice that, if we estimate the quality of the ν-th iteration from the discrepancy $\boldsymbol{\delta}^\nu$, by taking say the norm

$$\|\boldsymbol{\delta}^\nu\| = \sum_{m, n} |\delta^\nu_{m, n}|,$$

the approximate solution $\mathbf{u}^\nu - \mathbf{v}^\nu$ will be even worse than $\mathbf{u}^\nu$; the discrepancy for it roughly doubles. This is bound up with the fact that, given the method selected for interpolation of the functions from the base-points of the auxiliary to the base-points of the original net, the discrepancy for the function $\mathbf{u}^\nu - \mathbf{v}^\nu$ only changes at the base-points (m, n) which lie on the co-ordinate lines through the base-points of the auxiliary net. But the discrepancy (and hence the error $\mathbf{z}$) now consists in essence of "poor" eigenfunctions, which can be rapidly quenched by a fairly small number of iterations in accordance with (3); the process is then repeated until the necessary accuracy is obtained (in practice, until the norm of the discrepancy becomes fairly small).

The effect achieved by this means can be estimated reasonably well. It is easily seen that, when finding the function $\delta^{\nu}_{k,l}$, we take no account of the component of $\delta^{\nu}_{m,n}$ that vanishes at all the base-points of the auxiliary net; this component of the error will only be quenched as a result of the basic iterative process in accordance with (3), and the convergence is determined by the corresponding eigen number*. Thus the number q must not be chosen too large, or too small, so that the solution of system (6) does not turn into a problem requiring too much time (it is often a case of bearing in mind the auxiliary memory of the computer, which is necessary for the solution of auxiliary equation (6)). This remark suggests a fairly simple means of increasing the effectiveness of the proposed method: the position of the auxiliary with respect to the original net must be changed periodically; this entails some extra complication in the programme, but does not require an auxiliary memory.

§ 2

We shall now describe the effect obtained in an actual computation by the proposed method It should be mentioned that this method was developed in connection with the numerical solution of a non-stationary problem of the hydrodynamics of an incompressible fluid; here, we are required to solve a Poisson difference equation at each step in time. Moreover, the values of $u_{m,n}$ and $D_{m,n}$ occupied almost all the working store of the computer, so that employment of the most effective methods for speeding up the convergence of the iterations (e.g. the method proposed by A. A. Abramov [4]), which entail storage of the results from previous stages of the iterative process, would necessitate conversion to a slower form of computer memory, which is naturally undesirable.

The following may also be mentioned as regards practical realization of the proposed method. Instead of iterations in accordance with (3), a Seidel iterative process was taken as basic:

$$u^{\nu+1}_{m,n} = \frac{1}{4}\,(u^{\nu+1}_{m-1,n}+u^{\nu+1}_{m,n-1}+u^{\nu}_{m+1,n}+u^{\nu}_{m,n+1}-D_{m,n}). \tag{7}$$

Employment of (7) instead of (3) is more convenient for two reasons: firstly, as indicated in [5], the convergence of the Seidel process is twice as fast as the convergence of the process using (3) with $\alpha = \frac{1}{4}$. Secondly, computation using (7) takes up roughly twice as little time on the computer as when using (3).

When choosing the right-hand sides for the auxiliary system (6) it is better to take the mean values of the discrepancy at the base-points of the original net, close to the relevant base-point of the auxiliary net, and not the value of $\delta^{\nu}_{m,n}$ at some base-point of the original net.

It is self-evident that the base-points of the auxiliary net need not necessarily coincide with base-points of the original net; the boundary of the auxiliary domain need not necessarily fall along the boundary of the domain; it is only necessary to take boundary conditions for system (6) such that the function, interpolated from the base-points of the auxiliary to the base-points of the original net, satisfies on the boundary of the domain the homogeneous boundary conditions of system (2).

On comparing the proposed method with the methods of speeding up the convergence of the iterations that are based on the properties of the spectra of the iterative operators, it must be observed that the proposed method requires substantially greater outlay in programming, and in this lies its defect. On the other hand, it has an important advantage: its application and effectiveness do not depend on what we know about the spectrum of the iterative operator. Although the effectiveness of the proposed method is in fact closely linked with the properties of the spectrum of the basic iterative operator, the connection is of a different kind to that for methods using e.g. Chebyshev polynomials, or for the method of "over-relaxation" (see [5]), the decisive factor in

* It is clear that we should keep in view the "best" of the eigenfunctions that vanish at the base-points of the auxiliary net, i.e. the one that vanishes only at these base-points (assuming, of course, that such an eigenfunction exists).

the effectiveness of these latter being the closeness to 1 of the eigenvalue of the iterative operator. In our method this eigenvalue does not play such an important role, since it is associated with a "good" eigenfunction.

Our method was used to solve the 1st and 2nd boundary problems for equation (2) on a 39×47 net (including boundary points); the step of the auxiliary net was chosen five times greater than the original step. In this case the outlay in computer time entailed by a single solution of system (6) and correction of the approximate solution was roughly equal to the time outlay on 1·5–2 iterations in accordance with (7); we chose as initial approximation for the iterations a function equal to zero at interior points of the net and to specified values (in the case of the 1st boundary problem) at boundary points. The speed of convergence γ of the iterations was determined from the decrease of the norm of the discrepancy:

$$\|\delta^\nu\| = \sum_{m,n} |\delta^\nu_{m,n}|, \quad \gamma = \lim_{\nu\to\infty} \frac{1}{\nu} \ln \frac{\|\delta^\nu\|}{\|\delta^0\|}.$$

It is obviously more correct to estimate the convergence of the iterations by the number

$$\gamma_{eff} = \lim_{\nu\to\infty} (1/\nu_{eff}) \ln \|\delta^\nu\| / \|\delta^0\|,$$

where ν_{eff} is the number of iterations in accordance with (7) plus the operating time of the computer entailed in the use of the auxiliary net (the time expended in one Seidel iteration is taken as the unit of time). Numerical experiments showed that, depending on how many Seidel iterations fitted in with one application of the auxiliary net, γ varied between the limits of $-0{\cdot}20$ and $-0{\cdot}22$; as regards γ_{eff}, this was fairly stable:

$$\gamma_{eff} \sim -0{\cdot}18.$$

To diminish the discrepancy of the initial approximation by a factor of 1000 required as much computer operating time as 37–40 iterations in accordance with (7). If the effect obtained is compared with the results of [5], where a 10^3 times diminution of the discrepancy was obtained after 92 iterations in accordance with the "over-relaxation" method on a 50×50 net (in the case of the first boundary problem for equation (2)), it appears that our method is in this case 3–3·5 times less time-consuming than Young's method (see [5])* (Young's method had maximum efficiency here, due to the precise knowledge of the bounds of the Seidel iteration operator spectrum).

The table shows that diminution of the norm of the discrepancy occurs from iteration to iteration in the case when 7 Seidel iterations alternate with an application of the auxiliary net. It is clear from the Table that the iterations are fairly uniformly convergent: γ and γ_{eff} are established almost at once. Similar results are obtained with different ratios between the number of iterations and the number of applications of the auxiliary net.

The result of application of our method is practically the same for the second boundary problem. It may further be mentioned that, if the iterations are taken only in accordance with (7), the asymptotic speed of convergence in the case of equation (2) in a rectangular domain can be evaluated (see [5]) and turns out in our case as

$$\gamma = -0{\cdot}00572.$$

Our method is thus over 30 times as efficient as Seidel's.

We also carried out computations using the auxiliary net in two positions. The convergence of the iterations is characterized here by the numbers $\gamma = -0{\cdot}42$, $\gamma_{eff} = -0{\cdot}275$.

Our method has been described as relaxational for the following reason. In the literature, a relaxation method is usually described thus: the calculator, having obtained an approximate

* It should be borne in mind that the basic iterative formula in [5] for the case in question:

$$u^{\nu+1}_{m,n} = (1-\omega)\, u^\nu_{m,n} + \frac{1}{4}\,\omega\,(u^{\nu+1}_{m-1,n} + u^{\nu+1}_{m,n-1} + u^\nu_{m+1,n} + u^\nu_{m,n+1} - D_{m,n})$$

requires about twice as long a time at a point as formula (7).

ν	$\|\delta^\nu\|$	$\|\delta^{\nu+1}\|/\|\delta^\nu\|$	ν	$\|\delta^\nu\|$	$\|\delta^{\nu+1}\|/\|\delta^\nu\|$	ν	$\|\delta^\nu\|$	$\|\delta^{\nu+1}\|/\|\delta^\nu\|$
1	1680	0·775	25	6·11	0·835	49	0·0405	0·87
2	1132	0·795	26	5·09	—	50	0·0325	0·8
3	895	0·835	27	4·30	0·845	51	0·0237	0·73
4	748	0·865	28	3·69	0.86	52	0·0186	0·785
5	646	0·885	29	2·77	0·75	53	0·0151	0·810
6	572	0·90	30	1·98	0·715	54	0·0125	0·825
7	516	0·745	31	1·57	0·79	55	0·0104	0·83
8	384	0·64	32	1·30	0·83	56	0·0088	0·845
9	245	0·735	33	1·11	0·855	57	0·0067	0·755
10	180	0·78	34	0·964	0·87	58	0·0047	0·705
11	140	0·83	35	0·842	0·87	59	0·0038	0·805
12	116	0·86	36	0·654	0·775	60	0·0032	0·840
13	99·7	0·87	37	0·459	0·705	61	0·0027	0·845
14	86·6	0·73	38	0·361	0·79	62	0·0023	0·85
15	62·6	0·69	39	0·293	0·81	63	0·0020	0·87
16	43·2	0·795	40	0·243	0·83	64	0·00161	0·815
17	34·2	0·83	41	0·204	0·84	65	0·00119	0·74
18	28·3	0·86	42	0·173	0·85	66	0·00094	0·79
19	24·2	0·87	43	0·134	0·78	67	0·00077	0·82
20	21·1	0·88	44	0·0978	0·66	68	0·00064	0·83
21	18·6	0·77	45	0·0776	0.79	69	0·00054	0·845
22	14·3	0·695	46	0·0643	0·83	70	0·00045	0·83
23	9·94	0·76	47	0·0544	0·85			
24	7·56	0·81	48	0·0467	0·855			

solution of the system and having computed its discrepancy, changes the approximate solution in such a way that the discrepancy is made to vanish where it is large without at the same time being increased at other points; naturally, this procedure is the more successful, the more experienced the calculator. Evidently, the "experience" of the calculator here consists in skill in solving "by eye" a system with a discrepancy as right-hand side. We have tried to transfer this operation to the digital computer.

Translated by D. E. BROWN

REFERENCES

1. **BEREZIN, I. S.** and **ZHIDKOV, N. P.**, Metody vychislenii. (Computational methods.) Vol. II. Fizmatgiz, Moscow, 1960.
2. **SAUL'EV, V. K.**, Integrirovanie uravnenii parabolicheskogo tipa metodom setok. (Integration of equations of the parabolic type by a net method.) Fizmatgiz, Moscow, 1960.
3. **MILNE, W. E.**, The numerical solution of differential equations. Wiley, N. Y. 1953.
4. **ABRAMOV, A. A.**, Dokl. Akad. Nauk SSSR **74**: 1051–1052, 1950.
5. **YOUNG, D.**, Iterative methods for solving partial difference equations of elliptic type. Trans. Amer. Math. Soc. **76**: 92–111, 1954.