数理解析研究所講究録
1198 巻 2001 年 154-160

On numerical methods for solving linear systems appearing in Infinite Precision Numerical Simulation.

Toshiki TAKEUCHI (竹内 敏己), Hideo SAKAGUCHI (坂口 秀雄),
Cheng-Hai JIN (金 成海) and Hitoshi IMAI (今井 仁司)

Department of Mathematics, Faculty of Engineering, University of Tokushima
(徳島大学 工学部)

1 Introduction

We presented already IPNS(Infinite-Precision Numerical Simulation) to PDE systems with smooth solutions[5]. Errors in numerical simulations originate from truncation errors in discretization and rounding errors. IPNS realizes arbitrary reduction of both errors. In IPNS the spectral method is used for the control of truncation errors. In particular, the spectral collocation method with the Chebyshev-Gauss-Lobatto points[1] is very useful. The order of the approximation can be controlled by the number of collocation points. Multiple precision arithmetic[7] is used for the control of rounding errors, and it is easily available by using the library on the net, e.g. http://www.lmu.edu/acad/personal/faculty/dmsmith2/FMLIB.html [10]. It has already been shown that IPNS is effective in inverse problems, free boundary problems, etc[4, 6, 9, 12].

On the other hand, IPNS needs huge computer resources. This is due to that IPNS is based on multiple precision arithmetic. High-performance solver is necessary to solve the linear system which is derived by applying the spectral collocation method. The coefficient matrix of this linear systems are non-symmetric and the condition numbers are large in general. The Gauss elimination method has been used to solve linear system until now. But, the Gauss elimination method need the great time of calculation and the large memory area. Then, we use the CGS[11] method in this paper. Preconditioning is important for the CGS method. We use the SOR method that iteration step is fixed as preconditioner.

2 Test problem

Here, we consider the following Cauchy problem[2].

Problem. Find $u(x,t)$ s.t.

$$\Delta u(x,y) = 0, \qquad \text{in} \quad (0,1)\times(0,1), \tag{2.1}$$

$$u(0,y) = 0, \qquad 0 \le y \le 1, \tag{2.2}$$

$$u(1,y) = 0, \qquad 0 \le y \le 1, \tag{2.3}$$

$$u(x,0) = 0, \qquad 0 \le x \le 1, \tag{2.4}$$

$$\frac{\partial u(x,0)}{\partial y} = \frac{1}{\pi}\sin(\pi x), \qquad 0 \le x \le 1. \tag{2.5}$$

The exact solution to Problem is

$$u(x,y) = \frac{1}{\pi^2}\sinh(\pi y)\sin(\pi x). \tag{2.6}$$

This is an inverse problem. Usual approaches are applications of the regularization and least square method or AI. We tried some methods[3, 8], however we were not satisfied. Here, we apply IPNS directly. We use the same order approximation in x and y directions for simplicity. N represents the order in spectral collocation method. The number of total collocation points is $(N+1)^2$. The coefficient matrix of the linear system which is derived by applying the spectral collocation method is $M \times M$ matrix, where $M = N(N-1)$. This is a sparse matrix but is not a band matrix. Fig. 1. shows the form of the coefficient matrix. Here, $*$ represents non-zero element.

$$\left(\begin{array}{cccccccccccccccccccc}
* & & & & * & & & & * & & & & * & & & & * & & & \\
& * & & & & * & & & & * & & & & * & & & & * & & \\
& & * & & & & * & & & & * & & & & * & & & & * & \\
& & & * & & & & * & & & & * & & & & * & & & & * \\
* & & & & * & * & * & * & * & & & & * & & & & * & & & \\
& * & & & * & * & * & * & & * & & & & * & & & & * & & \\
& & * & & * & * & * & * & & & * & & & & * & & & & * & \\
& & & * & * & * & * & * & & & & * & & & & * & & & & * \\
* & & & & * & & & & * & * & * & * & * & & & & * & & & \\
& * & & & & * & & & * & * & * & * & & * & & & & * & & \\
& & * & & & & * & & * & * & * & * & & & * & & & & * & \\
& & & * & & & & * & * & * & * & * & & & & * & & & & * \\
* & & & & * & & & & * & & & & * & * & * & * & * & & & \\
& * & & & & * & & & & * & & & * & * & * & * & & * & & \\
& & * & & & & * & & & & * & & * & * & * & * & & & * & \\
& & & * & & & & * & & & & * & * & * & * & * & & & & * \\
* & & & & * & & & & * & & & & * & & & & * & * & * & * \\
& * & & & & * & & & & * & & & & * & & & * & * & * & * \\
& & * & & & & * & & & & * & & & & * & & * & * & * & * \\
& & & * & & & & * & & & & * & & & & * & * & * & * & *
\end{array}\right)$$

Fig. 1. The form of the coefficient matrix.

In the following chapters, we introduce the results by using various solvers.

3 Gauss elimination method

In this section the numerical results by the Gauss elimination method with 120 digit numbers are shown. The maximum errors, CPU times and the used memory are shown in Table 1. The amount of operations is proportional to $M^3(N^6)$ and the amount of use of memory is proportional to $M^2(N^4)$. In this paper, all numerical calculation are executed by Compaq AlphaServer ES40 (CPU : 21264–666MHz, memory : 4GB).

Table 1. Numerical results by the Gauss elimination method.

N	M	Maximum error	CPU time(s)	Memory(MB)
10	90	1.25×10^{-7}	0.8	5
20	380	2.86×10^{-19}	64	34
30	870	1.49×10^{-30}	813	160
40	1560	1.13×10^{-44}	4811	507
50	2450	7.22×10^{-58}	18879	1243
60	3540	8.09×10^{-74}	57836	2600

4 SOR method

In this section the SOR method for Cauchy problem is considered. We calculate the optimum relaxation parameters ω and the spectral radiuses $\rho(\omega)$ to the iterations matrices of SOR method by numerically. The results when $10 \leq N \leq 30$ are shown in Table 2. When N is even number, the value of ω is more than 1. In these case, the SOR method are not converged. When N is odd number, the spectral radius to the optimum relaxation parameter is approximately 1. It shows that the convergence of the SOR method is very slow. Therefore, the SOR method doesn't suit this problem.

Table 2. The Optimum relaxation parameters ω and the spectral radiuses $\rho(\omega)$.

N	ω	$\rho(\omega)$	N	ω	$\rho(\omega)$
11	1.5746315721	0.9990287280	21	1.6983954897	0.9999997536
13	1.6086960381	0.9997967442	23	1.7122827084	0.9999999642
15	1.6374743368	0.9999696146	25	1.7263674194	0.9999999935
17	1.6620194319	0.9999907450	27	1.7402953657	0.9999999985
19	1.6831462737	0.9999990088	29	1.7524578040	0.9999999998

Fig. 2. shows the profiles of the spectral radiuses in N=10, 11, 20 and 21.

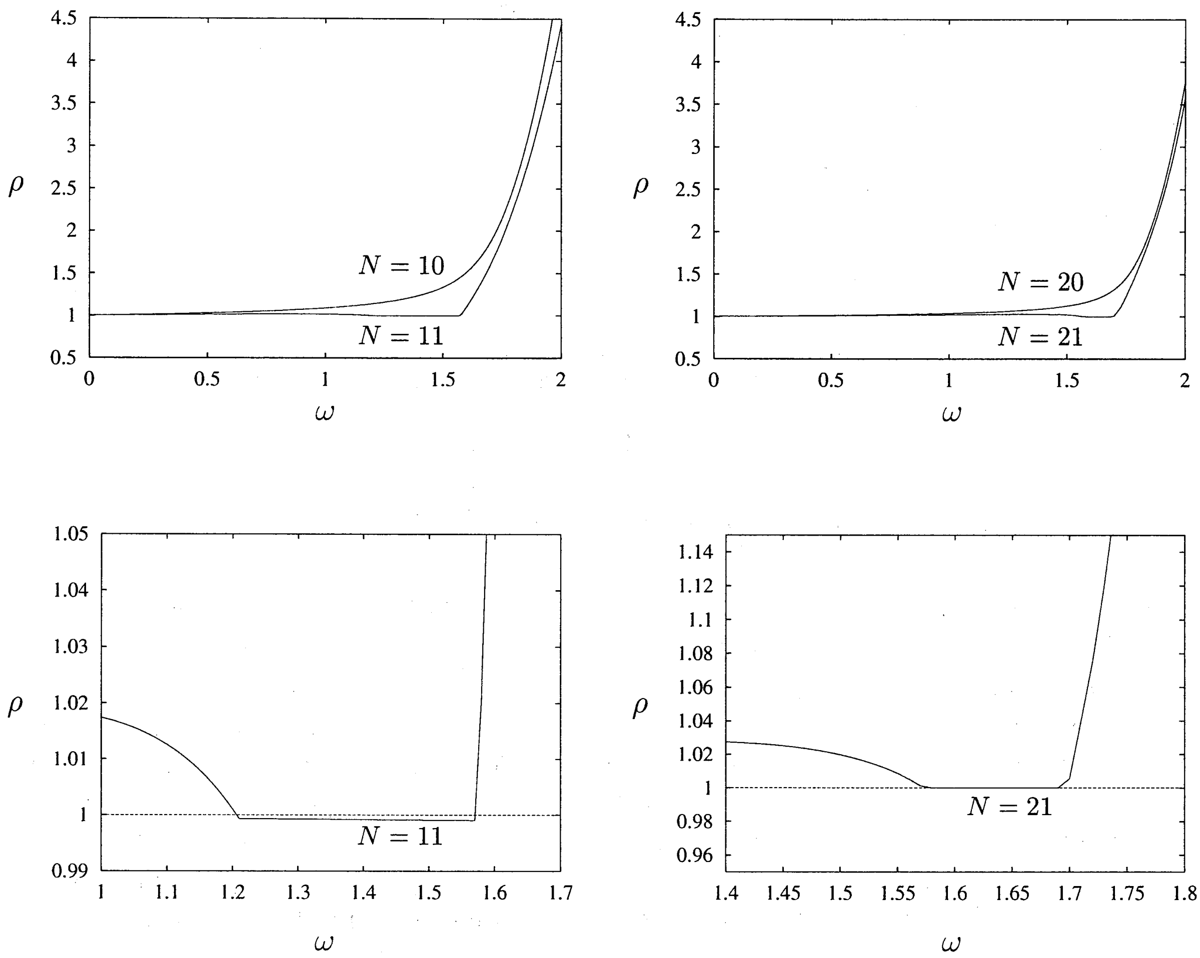

Fig. 2. The profiles of the spectral radiuses.

5 CGS method

In this section some numerical results by the CGS method are shown.

First, Fig. 3 shows the results obtained with unpreconditioned CGS method. In this case, $N = 20$ and 60 digit numbers have been used. But the CGS method is not converged.

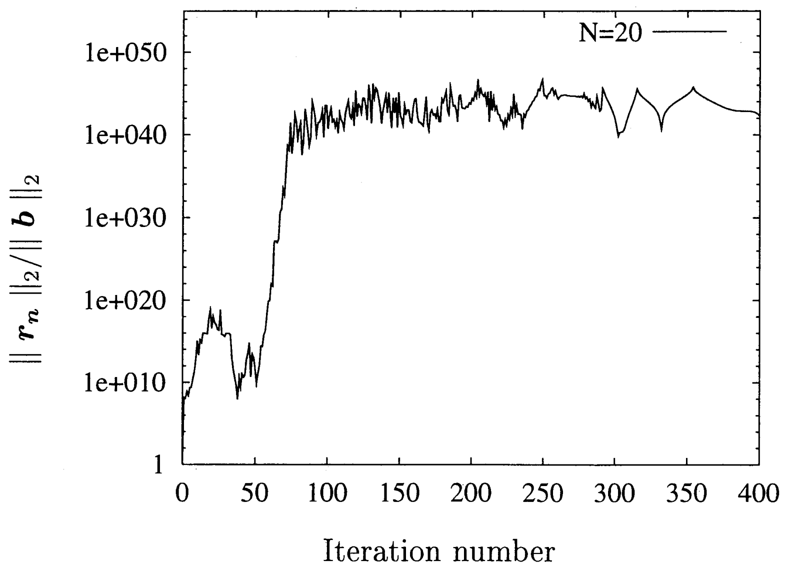

Fig. 3. Unpreconditioned CGS method

Next, we show the results by PCGS method. The SOR method was used as preconditioner. The iteration step of the SOR method is fixed. The iteration numbers of the CGS method to the relaxation parameters when the number of iteration for the SOR method are 1 and 5 are shown in Fig. 4. Here, $N = 20$ and 60 digit numbers have been used. Iterations were stopped when $\| \boldsymbol{r_n} \|_2 / \| \boldsymbol{b} \|_2 \leq 10^{-60}$. The vertical axis represents the iteration numbers of the CGS method. The CGS method is converged for various values of the relaxation parameter ω. When the relaxation parameter ω is 1.3~1.4, it seem that the iteration numbers are least. On the other hand, the SOR method is not converged when $N = 20$. Therefore, the optimal relaxation parameter ω in original SOR method is not important.

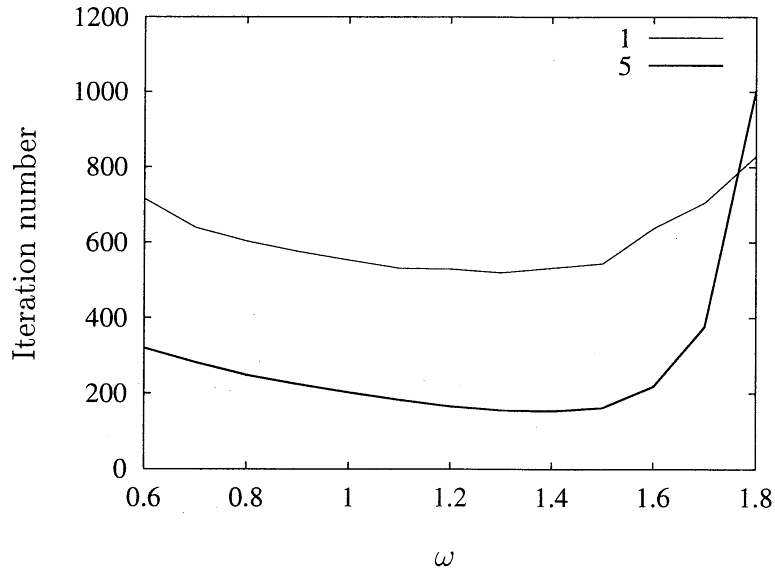

Fig. 4. Iteration numbers to relaxation parameters ω

Fig. 5 shows that the iteration number and the amount of operations of the PCGS method to the iteration step of the SOR method with $\omega = 1.3$ and 60 digit numbers. Iterations were stopped when $\| \boldsymbol{r_n} \|_2 / \| \boldsymbol{b} \|_2 \leq 10^{-60}$. The horizontal axis represents the number of iterations for SOR method. The vertical axis represents the number of iterations for CGS method and the number of matrix-multiplications for the CGS method and the SOR method in the left figure and right figure, respectively. The mount of operations of the PCGS is least when the number of iterations for SOR method is 3.

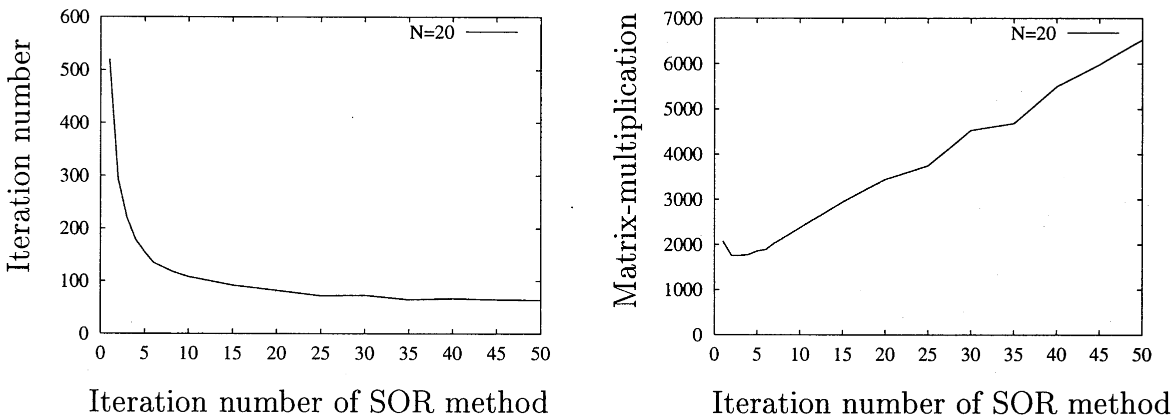

Fig. 5. Relationship between CGS and SOR

Fig. 6 shows the convergence behavior of PCGS method when $N = 60$ and digit number is 400. In preconditioner SOR method, we take the parameter $\omega = 1.4$ and fixed the number of iterations is 3. Iterations were stopped when $\| \boldsymbol{r_n} \|_2 / \| \boldsymbol{b} \|_2 \leq 10^{-90}$. In this case our method needs only 300MB memories, but the Guass elimination method needs 2600MB.

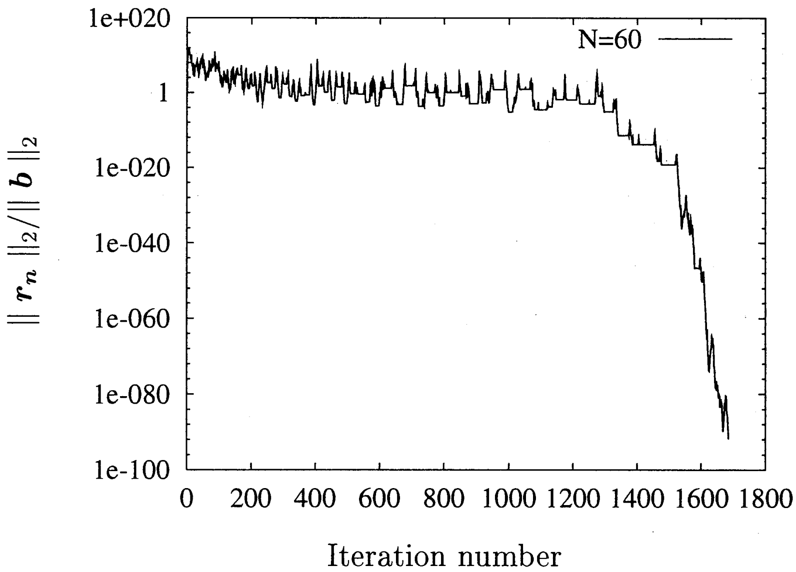

Fig. 6. Convergence history for PCGS when N=60

6 Conclusion

In this paper, the Gauss elimination method, the SOR method and the PCGS method are applied to the linear system which is derived by IPNS to Cauchy problem. The PCGS method with the SOR method that iteration step is fixed as preconditioner is effective. The memory could be substantially saved compared with the case to have used the Gauss elimination method.

Acknowledgements

This work is partially supported by Grant-in-Aid for Scientific Research(No. 10354001) and by CCSE of Japan Atomic Energy Research Institute.

References

[1] C. Canuto, M. Y. Hussaini, A. Quarteroni, T.A. Zang, *Spectral Methods in Fluid Dynamics*, Springer-Verlag, New York, 1988.

[2] H. Han and H. -J. Reinhardt, Some Stability Estimates for Cauchy Problems for Elliptic Equations, J. Inv. Ill-Posed Problems, **5**(5), 1997, pp.1–18.

[3] H. Imai, Application of the Fuzzy Theory and Spectral Collocation Methods to an Ill-Posed Shape Design Problem With a Free Boundary, in "Inverse Problems in Mechanics," Eds. S. Saigal and L.G. Olson, 186, pp.103-107, The American Society of Mechanical Engineers, 1994.

[4] H. Imai and T. Takeuchi, Application of the Infinite-Precision Numerical Simulation to an Inverse Problem, Proceeding of 1998-Workshop on MHD Computations "Study on Numerical Methods Related to Plasma Confinement", NIFS-PROC-40, 1999, pp.38–47.

[5] H. Imai, T. Takeuchi and M. Kushida, On Numerical Simulation of Partial Differential Equations in Infinite Precision, *Advances in Mathematical Sciences and Applications,* **9**(2), 1999, pp.1007–1016.

[6] H. Imai, T. Takeuchi, S. S. Shanta and N. Ishimura, On Numerical Computation of Lyapunov Exponents of Attractors in Free Boundary Problems, Proceeding of 1999-Workshop on MHD Computations "Study on Numerical Methods Related to Plasma Confinement", NIFS-PROC-46, 2000, pp.21–29.

[7] D. E. Knuth, *The Art of Computer Programming*, Addison-Wesley, 1981.

[8] A. Sasamoto, H. Imai and H. Kawarada, A Practical Method for an Ill-Conditioned Optimal Shape Design of a Vessel in Which Plasma Is Confined, in "Inverse Problems in Engineering Sciences", Eds. M. Yamaguti et al., pp.120-125, Springer-Verlag, 1991.

[9] S. S. Shanta, T. Takeuchi, H. Imai, M. Kushida, Numerical Computation of Attractors in Free Boundary Problems, Advances in Mathematical Sciences and Applications, **11**(2), 2001. (in press)

[10] D. M. Smith, A FORTRAN Package For Floating-Point Multiple-Precision Arithmetic, Transactions on Mathematical Software **17**, 1991, pp.273–283.

[11] P. Sonneveld, CGS, A Fast Lanczos-type Solver for Nonsymmetric Linear Systems, SIAM J. Sci. Stat. Comput., **10**, 1989, pp.26–52.

[12] Tarmizi, T. Takeuchi, H. Imai, Numerical Simulation of One-Dimensional Free Boundary Problems in Infinite Precision, Advances in Mathematical Sciences and Applications, **10**(2), 2000. (in press)