(2.14) $x_1+x_2\geqslant a-m_1-m_2,\ x_2+x_3\geqslant b-m_2-m_3,\ x_1+x_3\geqslant c-m_1-m_3.$

The latter c.r.p.c.'s exists if and only if we have:

$$d\geqslant \max\Big\{a-m_1-m_2,\ b-m_2-m_3, c-m_1-m_3,\ \frac{a+b+c}{2}-\frac{m_1+m_2+m_3}{2}\Big\}. \tag{2.15}$$

Proof. The big coalition has no effects on the previous configurations of the (m,p)-bargaining set $\mathscr{M}$, because this coalition cannot be used for m-objection and p-counterobjections, since it contains all the players. So, we have to see in what conditions a c.r.p.c. $(x_1, x_2, x_3;\ 123)$ belongs to the new (m,p)-bargaining set. As it has to be coalitionally rational it is necessary that x_1, x_2, x_3 satisfy the inequalities:

$$x_1+x_2\geqslant a,\ x_2+x_3\geqslant b,\ x_1+x_3\geqslant c,$$

but, as $a\geqslant a-m_1-m_2$, $b\geqslant b-m_2-m_3$, $c\geqslant c-m_1-m_3$, we have the relations (2.14). On the other hand, if conditions (2.14) are satisfied, then there is no m-objections.

For instance, if the player 1 has a m-objection $(y_1, 0, y_3;\ 13, 2)$ against the player 2, we have $y_1>x_1+m_1$, $y_3\geqslant x_3+m_3$, $y_1+y_3\leqslant c$, therefore $x_1+x_3\geqslant c-m_1-m_3$. But, from the relations (2.14), we have $x_1+x_3\geqslant c-m_1-m_3$, so the player 1 cannot have any m-objections. In order that the inequalities (2.14) have at least one solution, the necessary and sufficient conditions are:

$$d\geqslant a-m_1-m_2,\ d\geqslant b-m_2-m_3,\ d\geqslant c-m_1-m_3 \text{ and}$$

$$d\geqslant \frac{a+b+c}{2}-\frac{m_1+m_2+m_3}{2}$$

i.e. relation (2.15).

Remark 2.15. If we take $m_i=0$, for all i, $i=1, 2, 3$, in Theorem 2.14 we obtain Theorem 5.1 in [1].

As one can see the conditions of (m,p)-stability are sufficiently complicated just in the case of a three-person game. In order to simplify them we shall introduce the notions of m-kernels. But this is for a future paper.

REFERENCES

1. Aumann, R. J. and Maschler, M. — *The bargaining set for cooperative games*. Annals of Mathematics, Study no. 52, Princeton University Press, Princeton. 1964, 443–476
2. Gulaşu, S. and Maliţa, M. — *Triade*. Editura ştiinţifică, Bucureşti, 1973, p. 101–135
3. Maschler, M. and Davis, M. — *The kernel of a cooperative game*. Naval Research Logistics Quarterly, Office of Naval Research, vol. 12, 1965

Received 26.IX 1981

Analele ştiinţifice ale Universităţii „Al. I. Cuza" din Iaşi
Supliment la tomul XXVIII, s. I a, 1982

ANALYSIS OF CERTAIN POSSIBILITY FOR PARALLEL TREATMENT OF THE ROMBERG QUADRATURE ALGORITM

BY
CONSTANTIN CHIRUŢĂ

I. **The general theory.** In order to compute approximatively the integral $\int_a^b f(x)dx$, Romberg gave a method of numerical quadrature, using the trapezoidal rule and accelerating the convergence. In his method, one builds a triangular table:

$A_{1,1}$
$A_{2,1}A_{2,2}$
$A_{3,1}A_{3,2}A_{3,3}$
$A_{4,1}A_{4,2}A_{4,3}A_{4,4}$
. . . .

such that the first column is obtained by using the generalized trapezoidal method for $m_k=2^{k-1}$, where the values of the stepsize are: $h_k=(b-a)/m_k$; $k=1, 2, 3, \ldots$; hence:

$$A_{k,1}=\frac{h_k}{2}\Big\{f(a)+f(b)+2\Big[\sum_{i=1}^{2^{k-1}-1}f(a+ih_k)\Big]\Big\}. \tag{1}$$

Therefore, $\int_a^b f(x)dx=A_{k,1}-\frac{b-a}{12}h_k^2 f''(\xi_k)$, where $\xi_k\in[a,b]$.

The elements of the columns 2, 3, 4,... are calculated by the formula:

$$A_{k,j}=\frac{4^{j-1}A_{k,j-1}-A_{k-1,j-1}}{4^{j-1}-1} \tag{2}$$

with $k=2, 3, 4, \ldots$ and $j=2, 3, \ldots, k$.

Taking into account the way in which the second column was built, one could expect to have a faster convergence to the exact value of the integral on the second column than on the first one. It is proved (see [4]) that, if the function f has derivatives of any order on the closed interval $[a,b]$, then a olumns of Romberg's table converge to $\int_a^b f(x)dx$, and, in some additional conditions (from the extrapolation procedure), every column converges faster than the previous one. In fact, one proves that the diagonal sequence is the fastest sequence which converges to $\int_0^b f(x)dx$ and practically these elements are often used until $|A_{n,n}-A_{n-1,n-1}|<\varepsilon$, where ε is the prescribed tolerance. In [5], Romberg's method is compared with other formulas of quadrature for finite intervals and integrands which can be well approximated by polynomials. The comparison is made specially with the formulas of Gauss type, because the formulas of Newton-Côtes type are surly eclipsed by Romberg's method from nearly all points of view. Romberg's method is superior to the methods of Gauss type in stability and elegance of calculations, but it is inferior in precision and, specially, in the amount of calculus.

II. **Ways to improve Romberg's method.** There are different ways to improve Romberg's method :

1. To reduce the number of evaluations of the integrand values for calculating the first column in Romberg's table one remarks that (see[3]) :

$$(3)\qquad \sum_{i=1}^{2^{k-1}-1} f\left(a+\frac{i}{2}h_{k-1}\right)=\sum_{i=1}^{2^{k-2}} f\left(a+\left(i-\frac{1}{2}\right)h_{k-1}\right)+\sum_{i=1}^{2^{k-2}-1} f(a+ih_{k-1}).$$

From this one gets :

$$(4)\qquad A_{k,1}=\frac{1}{2}\left[A_{k-1,1}+h_{k-1}\sum_{i=1}^{2^{k-2}} f\left(a+\left(i-\frac{1}{2}\right)h_{k-1}\right)\right]$$

for $k=2,3,4,\ldots$.

Hence, to calculate an element in the first column of Romberg's table, one can use the element previously calculated, and this reduces at about half the number of evaluations for the integrand. This is the algorithm always used in the programming of Romberg's method.

2. Another method to reduce the number of evaluations of the integrand is that in which, instead of halving the interval $[a,b]$ at every step, other criteria are used to subdivide the interval. We mention here two such criteria (see [2], [6]) :

$$(5)\qquad h_k=(b-a)/k \quad \text{for } k=1,2,3,\ldots \quad \text{and}$$

$$(6)\qquad h_1=b-a,\ h_2=(b-a)/2,\ h_3=(b-a)/3,\ h_k=(b-a)/m_k \text{ where } m_k=2m_{k-2} \text{ for } k=4,5,6,\ldots .$$

With (5), we have to perform fewer calculations than in the case $h_k=(b-a)/2^{k-1}$, but some problems of numerical stability arise. The best results related both to the amount of calculation and to the accuracy and numerical stability were obtained with (6).

3. The best method seems to be parallel processing.

III. **Parallel treatment of Romberg's quadrature algorithm**

We use for this formula (4) in order to obtain the elements in the first column of Romberg's table and the result will be taken from the diagonal of the table after testing the tolerance. The method asks for computing the elements row by row, hence in the order :

$A_{1,1}$; $A_{2,1}$; $A_{2,2}$; $A_{3,1}$; $A_{3,2}$; $A_{3,3}$; $A_{4,1}$; and so on.

The precedence graph will be (fig. 1) :

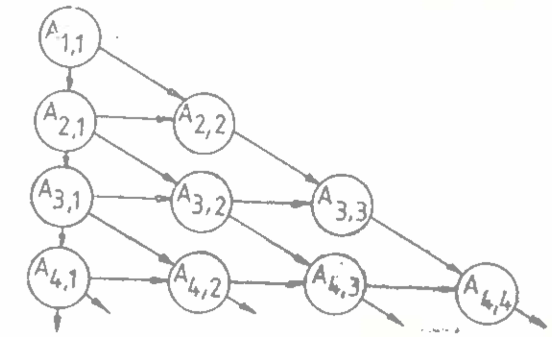

Fig. 1

The computing of the elements on the first column requires the longest time, even if one uses formula (4). This suggests the using of a multiprocessor system with two identical processors P_1 and P_2 such that one of them calculates only the values of the first column, and the second the other values of the table. A way of scheduling the two processors (nonpreemptive schedule), could be the following, if we take into account the precedence graph (fig. 2) :

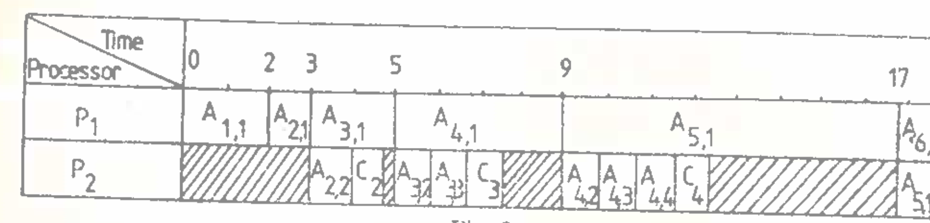

Fig. 2

Function	a	b	ε	Exact value with 8 decimals	Calculated value	Row in Romberg table	Necessary times in seconds	
							Serial	Parallel
$\frac{1}{1+x^4}$	0	1	10^{-6}	0,86697298	0,866973	6	0,75	0,72
$\ln x$	1	10	10^{-6}	14,02585092	14,0259	13	2,82	2,76
e^{-x^2}	0	5	10^{-6}	0,88622692	0,886227	7	0,78	0,75
$\cos x$	0	$\pi/2$	10^{-6}	1,00000000	1,00000	7	0,69	0,66

Fig. 3

Here we denote by C_i, $i=2, 3, 4, \ldots$, the stopping test at the end of the row number i, namely if $|A_{i,i}-A_{i-1,i-1}|<\varepsilon$ (if the test is accomplished then C_i also writes $A_{i,i}$), and the hatched part represents the free time of the processors. We take as time unit the necessary time to evaluate an integrand. For the schedule in fig. 2 we obtain the comparison table which contains the necessary times for serial and parallel computations of some definite integrals (fig. 3). The results are obtained by simulating the parallelism on the serial computer Felix C-256, using FORTRAN and ASSIRIS programs

Remark. We could use a multiprocessor system with two processors with different speed; the fastest processor will be used to compute the elements on the first column. We also could use the preemptive scheduling. However, in both cases, the results will be better, but more difficult to simulate.

REFERENCES

1. Blum, E. K. — *Numerical Analysis and Computation. Theory and Practice.* Addison-Wesely Publishing Company, 1972
2. Bulirsch, R. — *Bemerkungen zur Romberg Integration.* Numerische Mathematik 6, 1964, p. 6—16
3. Burden, L. R., Douglas, Faires, J., Reynolds, C. A. — *Numerical Analysis.* Prindle, Weber & Schmidt, Boston, Massachusetts, 1978
4. Henrici, P. — *Elements of Numerical Analysis.* John Wiley & Sons, Inc., 1964
5. Ralston, A.; Wilf, S. H. (edit. by) — *Mathematical Methods for Digital Computers,* vol. II, John Wiley & Sons, Inc., 1967
6. Stroud, A. H. — *Numerical Quadrature and Solution of Ordinary Differential Equations.* Springer-Verlag, New York, Heidelberg, Berlin, 1974

Received 6.VII.1981

Faculty of Mathematics
University of Iaşi
6600 Iaşi, R.S. Romania