Fabled Boundary Condition Forecast Algorithm in Multigrid Domain Decomposition Parallel Computing

Guo Qingping[1] Yakup Paker[2] Zhang Shesheng[1]

Dennis Parkinson[2] Wei Jialin[1]

[1]*Wuhan Transportation University, Wuhan 430063, P. R. China*

[2]*Queen Mary & West field College, University of London.E1 4NS U.K.*

qpguo@public.wh.hb.cn guo@dcs.qmw.ac.uk paker@dcs.qmw.ac.uk

Abstract: We propose a fabled boundary condition forecast algorithm for multi grid parallel computing, and derive a forecast function formula in this paper. Numerical results of one and two-dimension boundary condition problems obtained with the algorithm in a PVM network computing environment show that the algorithm has high parallel efficiency.

Keywords: Multi-Grid, Domain Decomposition, Boundary Condition Forecast

1.Introduction

In recent years, several new multi grid parallel algorithms have been proposed in the literatures [1,2,3]. Guo Qingping et al (1998)[1] considered the domain decomposition method for cyclical temperature in ceramic/metal composites, and showed that MGP algorithm can obtain good numerical computing speedup in networking computing environment. Xu Zhengquan (1996) [2] discussed domain decomposition method of multi-grid distributed computing, and pointed out that a good initial value is very important when the equation is solved using iterative method. Iterative methods are used often in MGP, and amount of computing and communication increase as the number of iterative increases. Louis (1982)[3] detailed the polynomial iterative method which may speed up iterative speed, and showed that better initial value may be obtained by using his theory.

According to the theory given in reference [3], we decompose a numerical domain into several sub-domains. Boundaries between sub-domains are defined as a fabled boundary, and a mathematical model to forecast the boundary value at the fabled boundaries is proposed. Some numerical examples are given to show how to use our algorithm.

Full multi grid method is one of efficient multi grid method ([4] 1986 H. Hoppe). Combining the fabled boundary condition forecast method with the FMGM we can obtain a high efficient parallel fabled boundary condition forecast full-multigrid method.

2. Construction of boundary forecast function

Many boundary problems may be written as :

$L^{Ù}u\ (x) = f^{Ù}(x)\quad x\quad Ù$ (1)

$L^{Ã}u\ (x) = f^{Ã}(x)\quad x\quad Ã$

The Ù is a define domain of unknown function u (x),Ã is a boundary of Ù, L is a differential operator. Suppose$Ù_i$ is ist grid onÙ and $Ã_i$ a corresponding boundary by using difference method, Eq (1) may be written as

$L_i^{Ùi}u_i(x) = f^{Ùi}(x)$ x   Ù   (2)

$L_i^{Ãi}u_i(x) = f^{Ãi}(x)$ x   Ã

We decompose domain Ù into several sub-domain $Ù_i$, i =1,2,..,m, $Ã_i$ is boundary of sub-domain $Ù_{i,}$ so that the fabled boundary $Ã_{in}$ is

$Ã_{in}=Ã_1+Ã_2+...+Ã_m–Ã$

Suppose $u_i(x_{in})$ are the numerical value of function u while x_{in} $Ã_{i\,n}$, then $\{u_i(x_{in})\}$ is a sequence of fabled boundary points. By using of $u_1(x_{in}),u_2(x_{in}),...,u_m(x_{in})$, we construct a subspace

$R_n = span\{u_1(x_{in}),u_2(x_{in}),...,u_m(x_{in})\}$

Assume that G is a mapping of R_n to space R, and

$v_{n+1}=G(Y)$ Y R_n (3)

then it is possible to take v_{n+1} as forecast initial value on the fabled boundary for (n +1) th grid, and v_{n+1} should satisfy condition:

$v_{n+1}-u(x_{in})$ < $u_n- u(x_{in})$ x_{in} Ù (4)

here $u(x_{in})$ is function value at point x_{in}. We propose some models to forecast v_n as following. In those models, symbol $v_{i,j}$ is used to denote the fabled boundary value of jth model on the ith grid.

Model 1. One step forecast function

$v_{n+1,1}(x_{in}) =u_n(x_{in})$

Model 2. Two step forecast function

$v_{n+1,2}=u_{n-2}+3(u_n-u_{n-1})$

Theorem 1. If n   , u_n $u(x_{in})$, then if n   , $v_{n+1,1}$ $u(x_{in})$, $v_{n+1,2}$ $u(x_{in})$.

Proof. When n   , u_n $u(x_{in})$, so that if n   ,u_n-u_{n-1} $u(x_{in})-u(x_{in})=0$

Therefore if n   , we obtain

$v_{n+1,1}=u_n$ $u(x_{in})$ $v_{n+1,2}=u_{n-2}+3(u_n-u_{n-1})$ $u(x_{in})$

#.

Theorem 2. Let us denote u(s) as u_s in which the s is an integer variable. If

$u(s) = u_{n-1} + C_1$ $– n + 1) + C_2$ $– n + 1)^2$ (5)

then $u(n+1) = u_{n-2}+3(u_n – u_{n-1})$

where the C_1 and C_2 are some kinds of constant.

Proof. From the conditions of the theorem, we may derive formulas for C_1 and C_2:

$C_1 = 0.5\ (u_n-u_{n-2})$, $C_2 =0.5\ (u_n-2u_{n-1}+u_{n-2})$

so that

$u\ (n+1) = u_{n-1}+2C_1+4C_2 = u_{n-1}+u_n-u_{n-2}+2(u_n-2u_{n-1}+u_{n-2})$

$= u_{n-2}+3(u_n-u_{n-1})$

#.

Eq.(5) is a polynomial of degree 2. From Eq.(5), the model 2 can be derive when s=n+1, and the maximal value of Eq (5) is

$u_{max} = u_{n-1} - 0.25C_1^2/C_2$

Model 3. Maximal value forecast function $v_{n+1,3}$

$v_{n+1,3} = u_{n-1} - 0.25C_1^2/C_2$ (6)

Theorem 3. If $\{u_n\}$ is a non decreasing sequence and has a limit value, and

$u_{n+1}-u(x_{in}) < u_n-u(x_{in})$.

then $v_{n+1,3}>v_{n+1,2}$

Proof. Because $\{u_n\}$ is the non decreasing sequence and has a limit value, and

$u_{n+1}-u(x_{in}) < u_n-u(x_{in})$.

then

$C_1 = 0.5 (u_n-u_{n-2}) > 0$

$C_2 = 0.5 (u_n-2u_{n-1}+u_{n-2}) =0.5 [u_n-u_{n-1} - (u_{n-1}-u_{n-2})] < 0$

so that the limit value of Eq.(5) is the maximal value of Eq.(5), and

$v_{n+1,3} = u_{max} > v_{n+1,2}$

#.

Obviously it is easy to proof theorem 4.

Theorem 4. If $\{u_n\}$ is a decreasing sequence and has a limit value, and

$u_{n+1}-u(x_{in}) < u_n-u(x_{in})$.

then $v_{n+1,3} < v_{n+1,2}$

3. Fabled Boundary Condition Forecast Algorithm with Maximal Values

Using model 3, value of function u at fabled boundary$\tilde{A}_{in}$ in domain decomposition can be forecasted. Numerical procedure could be described as following.

(1).Calculate function value u_i on corresponding grids by few sweeps, there i = 1,2,3. The ith grid is coarser then (i+1)th grid. So that initiative fabled boundary values on each sub-domain at the first three grids are obtained.

(2).According to model 3, calculate $v_{4,3}$;

(3).Taking $v_{4,3}$ as fabled boundary value at fourth grid, interpolating the current approximation of function u_3 onto the next finer grid to get initiative value of u_4. Calculate the function value u_4 in each sub-domain by iterative method respectively;

(4). Interpolate the current approximation of function u_4 onto the next finer grid to get initiative value of u_5.

(5).Calculate value of u_5 by few times of globe iteration among whole domain. This step needs sub-domain boundary value exchange, and the fabled boundary value at each sub-domain can be calculated out. This step is used to smooth forecast error of fabled boundary value.

(6). If $(u_5-u_3)(u_3-u_1)>0$, and $u_5-u_3 < u_3-u_1|$, then with data $(1,u_1)$, $(3,u_3)$, $(5,u_5)$ calculate $v_{6,3}$ by Eq.(5) and (6), there u_1, u_3 and u_5 replacing u_{n-2}, u_{n-1} and u_n respectively. Otherwise let $v_{4,3}=0.5(v_{3,3}+v_{4,3})$, and go to step 3. Value of u_1, u_3 and u_5 are all obtained by global domain iteration.

(7). If u_5-u_3 <å, å is an required error bound, then function value u_5 is a required result, and the calculation ends. Otherwise go to next step;

(8).u_3 u_1,u_4 u_2,u_5 u_3,$v_{6,3}$ $v_{4,3}$, go to step 3.

According to the step (3) of the numerical procedure, fabled boundary value need not change on same grid in each pair of sub-domains, so that we need not communicate between sub-domains during iteration. From step (2) and step (6), we can forecast next finer grid function value at fabled boundary by the value of three coarser grids.

4. Case study

Example 1. One dimension boundary problem.

$u'' = -100x^2 x$ Ù=[0,1]

$u(0) = u(1) = 0$

The exact solution of this equation is $u = 25x(1-x^3)/3$

In order to explain our numerical method simply, we decompose domain Ù into two sub-domain Ù$_1$=[0, 0.5] and Ù$_2$=[0.5, 1], and take two networked computers to calculate function u on each sub-domain respectively, the nth computer only calculates function value u on nth sub-domain, n =1,2. The parallel software environment used is the PVM. In this simple example, the fabled boundary is at point x = 0.5. Let the step length of the coarsest grid (1st grid) is h = 0.5, the 2nd grid h = 0.25, the 3rd grid h=0.125, etc. At fabled boundary x = 0.5 let initial function value $u_0=0$. Using model 1, we calculate the function value u1 on first grid by few iteration, then interpolate u1 into second grid and calculate function value u2 on second grid, etc. We obtained $u_1(x) = 3.125$, $u_2(x) = 3.515$ etc at the point x=0.5. By using model 3, after few iterations on the first three grids we forecast the function value $v_{4,3} = 3.63745$ at fabled boundary x=0.5.Then calculate function value of u on 4'th grid by solving two sub-domain relevant independent boundary condition problem. Then interpolate the fourth grid on to the fifth to get initiate values and forecast the fabled boundary by our formula, calculate u value at all points on the 5th grid by normal domain decomposition method with few iterations. On the fifth grid there is necessary for data exchange between two sub-domains to smooth boundary forecast error. We obtain $u_5(0.5) = 3.641841$.Similarly we forecast the function value $v_{6,3} = 3.644253$ at fabled boundary x=0.5 on the sixth grid, solving two sub-domain relevant independent boundary condition problem etc until error bounds conditionå=0.0001 between two subsequent iterations has been satisfied. In this example, we obtain the satisfied numerical value of u on 8th grid.

General speaking we calculate unknown u by normal parallel computing on decomposed sub-domain with fabled boundary data exchange on all odd number grids, and solving relevant independent boundary condition problems with forecast fabled boundary values for all even number grids. By this way we reduce nearly half communication costs.

Denote number of computers as N. We decompose domain [0, 1] into sub-domain $[x_{k-1},x_k]$, k=1,2,...,N. Let one computer only calculate function value of u on one sub-domain. Table 1 shows the parallel calculate efficiency data of model 1 and model 3.When N>2,the efficiency of model 3 is 12 higher than that one of model 1.

Table 1. The parallel calculate efficiency

	1	**2**	**3**	**4**
Model 1	100	92.19	84.31	81.13
Model 3	92.19	96.31	93.82	91.42

Example 2. Two dimension no-line boundary problem

$u_{xx}+u_{yy}-u^2=-50x(x+y)-625(x+y-x^2-y^2)^2/9$ (x,y)   Ù=[0,1]   [0,1]

u(0,y)=u(1,y)=u(x,0)=u(x,1)=0

We decompose domain Ù into sub-domain

$Ù^i$={(x,y)   (i-1)/N   x   i/N,0   y   1,i=1,2,...,N,}

The fabled boundary is

$Ã^{in}$ ={(x,y)   x=i/N,0<y<1,i=1,2,...,N,}

Let one computer only calculate value of function u on one sub-domain. Step of 1st grid is h=1/N. The step of 2nd grid is h = 0.5/N.The step of 3rd grid is h=0.25/N etc. Table 2 shows the parallel calculate efficiency data of model1 and model 3. When N>2, the efficiency of model 3 is 12.2 higher than the one of model 1.

Table 2. The parallel calculate efficiency

	1	**2**	**3**	**4**
Model 1	100	93.03	85.26	82.72
Model 3	100	97.31	94.41	92.03

5. Further Improvement

In the proposed algorithm we transform a boundary problem into several relevant independent sub-domain boundary problems on all even number grid. Therefore it is nature and efficient to solve those boundary problems with the V-cycle multi grid method (MGM) without any extra communication between sub-domains. We name it as modified parallel full multi grid method with fabled boundary forecast.

6.Conclusion

Model 1 is a linear forecast method. Model 2 is a polynomial forecast method, but model 3 is a polynomial maximal forecast method. The model 3 may decrease the amount of communication between sub-domains. The examples of one and two-dimension problems show that our fabled boundary initial value forecast method is efficient in parallel multi grid computing.

Reference

[1] Guo Qingping. Parallel computing using Domain Decomposition for Cyclical Temperatures in Ceramic/Metal Composites,EuroPVM/MPI98, Liverpool UK, 1998.

[2] Xu Zhengquan. Domain decomposition method of multi grid distributed computing, Journal of Numerical and Compution.Vol.1.1996.p1-5.

[3] Louis. A. Hageman & David M. Young. Applied Iterative Methods. New York. 1982 p50-197.

[4] H. Hoppe and H. Mühlenbein, Parallel adaptive full-multigrid methods on message-based multiprocessors, Parallel Computing, No 3, North-Holland. 1986 p269-287