New method for solving high accuracy surface modeling

CHEN Chuanfa[1], **YUE Tianxiang**[1], **LIU Hongtao**[2]

1. *Institute of Geographic Sciences and Natural Resources Research*, *Chinese Academy of Sciences*, *Beijing* 100101, *China*
2. *Jinan Institute of Engineering and surveying*, *Shandong Jinan* 250013, *China*

Abstract: High accuracy surface modelling (HASM) constructed based on the fundamental theorem of surface is more accurate than the classical methods, but the computational speed of HASM is proportional to the third power of the total number of grid cells in the computational domain. In order to decrease the computational cost and improve the accuracy of HASM, this paper employed a modified Gauss-Seidel (MGS) to solve HASM. The fact that MGS is more accurate and faster than GS is proved in terms of theorem. Gauss synthetic surface was employed to comparatively analyze the simulation errors and the computing time of MGS and GS. The numerical tests showed that under the same simulation accuracy, MGS is faster than GS, and the time difference between MGS and GS is approximately proportional to the second power of the total number of grid cells. Under the same outer or inner iterative cycles, MGS is more accurate than GS. The computing time of MGS is proportional to the first power of the total number of grid cells. Compared with the direct methods for solving HASM, MGS greatly shortens the computing time of HASM. SRTM3 (36°—37°N, 107°—108°E) of Dongzhi tableland located in Gansu province was employed as a real word example to validate the accuracy of HASM based on MGS. In the example, about 50% of SRTM3 was used as validation points, the others for DEM simulation. The results indicated that RMSE of HASM based on MGS is about 2.4, 1.8, 1.3, 2.7 times less than those of KRIGING, IDW, TIN and NEAREST.

Key words: GS iteration, surface simulation, accuracy, test analysis, interpolation

CLC number: TP751.1 **Document code:** A

Citation format: Chen C F, Yue T X and Liu H T. 2010. New method for solving high accuracy surface modeling. *Journal of Remote Sensing.* **14**(4): 742—750

1 INTRODUCTION

The computational process of high accuracy surface modelling (HASM) has been mature in terms of theory (Yue & Du, 2006). It can be divided into three processes including the coefficient matrix formulating of Gaussian equation, sampling equation formulating and solving the linear system of HASM. When the computational domain is regular and the sampling equation has one order truncation error, the solving method of HASM equations has an effect on its computational efficiency (Al-Kurdi & Kincaid, 2006, Yue *et al.*, 2007). Former researches indicated that the computational cost of the direct method for solving HASM is proportional to the third power of the total number of grid cells, which seriously influences its wide application (Yue *et al.*, 2007).

The iterative method has been accepted as an efficient method for solving huge linear system (Saad, 2003). One of those methods is Gauss-Seidel (GS), which only needs few save volume, especially for solving huge sparse one (Bramble & Pasciak, 1992; Ujević, 2006). But the convergence speed of GS is very low. So we used a modified GS method (MGS) for solving HASM. Gaussian synthetic surface was employed to compare the efficiency of MGS with GS. In the real world example, the SRTM3 of Dongzhi tableland was used to compare the performance of HASM based on MGS with those of the classical interpolation methods including IDW, TIN, KRIGING and NEAREST with the default parameters performed in ARCGIS 9.3.

2 MGS ITERATION

2.1 MGS formulation

Suppose HASM can be eventually transformed to solve a linear system, which is expressed as, $Sx^{k+1}=b^k$, where, S is a symmetric definitive matrix, $S\in R^{n\times n}$, $T\in R^n$, k is iterative times. The process of solving the linear system of HASM is named outer iteration, while updating the vector b is termed inner iteration. The detailed information about the formulation of MGS can be found in the paper (Yue & Du, 2005).

The component process of solving HASM based on GS is expressed as,

Received: 2009-06-01; **Accepted**: 2009-08-07

Foundation: China National Science Fund for Distinguished Young Scholars (No.40825003); National High-Tech R&D Program of China (No. 2006AA12Z219); National Key Technologies R&D Program of Ministry of Science and Technology of China (No. 2006BAC08B); Major Directivity Projects of Chinese Academy of Science (No. KZCX2-YW-429).

First author biography: CHEN Chuanfa (1982—), male, PhD student. His research interests include surface modelling and DEM uncertainty. He has published 6 papers. E-mail: chencf@lreis.ac.cn

$$\begin{aligned}&\text{for } k=0,1,2,\cdots \text{//iterative times}\\&\quad y_1=x_k \text{//the kth iteration}\\&\quad \text{for } i=1,2,\cdots,n\\&\qquad y_{i+1}=y_i+\alpha_i e_i\\&\quad \text{end for } i\\&\quad x_{k+1}=y_{n+1}\\&\quad \text{stopping criteria}\\&\text{end for } k\end{aligned} \tag{1}$$

where,

$$e_1=(1,0,0,\cdots,0)^{\mathrm{T}},e_2=(0,1,0,\cdots,0)^{\mathrm{T}},\cdots,e_n=(0,0,0,\cdots,1)^{\mathrm{T}} \tag{2}$$

$$\alpha_i=-\frac{p_i}{s_{ii}},\ p_i=(S_i,y_i)-b_i \tag{3}$$

If f has a second derivative at y_i, in terms of Taylor expansion, we have

$$\begin{aligned}&f(y_{i+1})=f(y_i+\alpha_i e_i)=f(y_i)+\alpha_i(Sy_i-b,e_i)+\\&\frac{1}{2}\alpha_i^2(Se_i,e_i)=f(y_i)+\alpha_i p_i+\frac{1}{2}\alpha_i^2 s_{ii}\end{aligned} \tag{4}$$

Based on Eq. (3) and Eq. (4), we can get

$$f(y_i+\alpha_i e_i)-f(y_i)=-\frac{1}{2}\frac{p_i^2}{s_{ii}}\leqslant 0 \tag{5}$$

i.e.

$$f(y_{i+1})-f(y_i)=-\frac{1}{2}\frac{p_i^2}{s_{ii}}\leqslant 0 \tag{6}$$

Eq. (6) indicates that GS can reach its convergence.

Let us consider the element

$$z_{i+1}=y_i+h_i+\gamma_i q_i \tag{7}$$

where, $h_i=-\frac{p_i}{s_{ii}}e_i$, $\gamma_i\in R,q_i\in R^n$. In terms of Taylor expansion, we have,

$$\begin{aligned}f(y_i+h_i+\gamma_i q_i)&=f(y_i)+(f'(y_i),h_i+\gamma_i q_i)+\\&\frac{1}{2}(S(h_i+\gamma_i q_i),h_i+\gamma_i q_i)\\&=f(y_i)+(Sy_i-b,h_i)+\gamma_i(Sy_i-b,q_i)+\\&\frac{1}{2}(Sh_i,h_i)+\gamma_i(Sq_i,h_i)+\frac{1}{2}\gamma_i^2(Sq_i,q_i)\\&=f(y_{i+1})+\gamma_i[(Sy_i-b,q_i)+(Sq_i,h_i)]+\\&\frac{1}{2}\gamma_i^2(Sq_i,q_i)\end{aligned} \tag{8}$$

We now define the function,

$$g(\gamma)=\gamma_i[(Sy_i-b,q_i)+(Sq_i,h_i)]+\frac{1}{2}\gamma_i^2(Sq_i,q_i) \tag{9}$$

Such that, $g'(\gamma)=(Sy_i-b,q_i)+(Sq_i,h_i)+\gamma_i(Sq_i,q_i)$, $g''(\gamma)=(Sq_i,q_i)\geqslant 0$ indicating that g(γ) has its minimum. From the equation g′(γ)=0, we get

$$\gamma_i=-\frac{(Sy_i-b,q_i)+(Sq_i,h_i)}{(Sq_i,q_i)} \tag{10}$$

From Eq. (9) and Eq. (10), we have

$$\begin{aligned}g(\gamma_i)&=-\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)}+\\&\frac{1}{2}\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)^2}(Sq_i,q_i)\\&=-\frac{1}{2}\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)}\leqslant 0\end{aligned} \tag{11}$$

From Eq. (8) and Eq. (11), we get

$$\begin{aligned}f(z_{i+1})-f(y_i)&=f(y_{i+1})-f(y_i)\\&-\frac{1}{2}\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)}\end{aligned} \tag{12}$$

Eq. (12) indicates that the element z_{i+1} gives better reduction of f than the element y_{i+1}. The pre-code of MGS for solving HASM can be expressed as,

$$\begin{aligned}&\text{for } k=1,2,\cdots \text{//iterative times}\\&\quad z_1=x_k;\text{//the kth iteration}\\&\quad \text{for } i=1,2,\cdots,n\\&\qquad z_{i+1}=z_i+h_i+\gamma_i q_i;\\&\quad \text{end for } i\\&\quad x_{k+1}=z_{n+1};\\&\quad \text{stopping criteria}\\&\text{end for } k\end{aligned} \tag{13}$$

where,

$$\begin{aligned}&h_i=-\frac{\overline{p}_i}{s_{ii}}e_i,\overline{p}_i=(S_i,z_i)-b_i,\\&\gamma_i=-\frac{(Sy_i-b,q_i)+(Sq_i,h_i)}{(Sq_i,q_i)}\end{aligned} \tag{14}$$

From Eq. (1) and Eq. (13), we can get that MGS updates two components of the approximate solution of GS at each iterative cycle.

2.2 *q* determination

From the theory of MGS, we can see that q is an n-order vector. In this paper,

$$q_i=e_j,\ (i\neq j),\ \text{so } z_{i+1}=z_i+h_i+\gamma_i e_j \tag{15}$$

where,

$$h_i=-\frac{\overline{p}_i}{s_{ii}}e_i,\overline{p}_i=(S_i,z_i)-b_i \tag{16}$$

$$\begin{aligned}\gamma_i&=-\frac{(Sz_i-b,e_j)+(Se_j,h_i)}{(Se_j,e_j)}\\&=-\frac{\overline{p}_j-\frac{\overline{p}_i}{s_{ii}}s_{ij}}{s_{jj}}=-\frac{\overline{p}_j}{s_{jj}}+\overline{p}_i\frac{s_{ij}}{s_{ii}s_{jj}}\end{aligned} \tag{17}$$

where,
$$\overline{p}_j=(S_j,z_i)-b_j. \tag{18}$$

We can choose j in different way. In this paper, we choose j=i−1, (i=2, 3,⋯,n), $j=n,(i=1)$,

$$\begin{aligned}&z_i=z_{i-1}+h_{i-1}+\gamma_{i-1}e_{i-2},\\&h_{i-1}=-\frac{\overline{p}_{i-1}}{s_{i-1,i-1}}e_{i-1},(i=1,2,\cdots,n)\end{aligned} \tag{19}$$

where, $\gamma_0=\gamma_n,e_0=e_n,e_{-1}=e_{n-1},h_0=h_n,\overline{p}_0=\overline{p}_n,s_{00}=s_{nn}$.

From Eq. (18) and Eq. (19), we get that

$$\begin{aligned}\overline{p}_j&=(S_j,z_i)-b_j=(S_{i-1},z_{i-1}+h_{i-1}+\gamma_{i-1}e_{i-2})-b_{i-1}\\&=(S_{i-1},z_{i-1})-b_{i-1}-\frac{\overline{p}_{i-1}}{s_{i-1,i-1}}(S_{i-1},e_{i-1})+\gamma_{i-1}(S_{i-1},e_{i-2})\\&=\overline{p}_{i-1}-\frac{\overline{p}_{i-1}}{s_{i-1,i-1}}s_{i-1,i-1}+\gamma_{i-1}s_{i-1,i-2}=\gamma_{i-1}s_{i-1,i-2}\end{aligned} \tag{20}$$

where, $s_{0,-1}=s_{n,n-1},s_{1,0}=s_{1,n}$.

From Eq. (17) and Eq. (20), we get,

$$\gamma_i = -\gamma_{i-1}\frac{S_{i-1,i-2}}{S_{i-1,i-1}} + \overline{p}_i \frac{S_{i,i-1}}{S_{i-1,i-1}S_{ii}} \tag{21}$$

3 GAUSSIAN SYNTHETIC SURFACE SIMULATION

In this paper, we employed Gaussian synthetic surface (Fig.1) to validate the efficiency of MGS and GS for solving HASM. The formulation of Gaussian synthetic surface is expressed as,

$$f(x,y) = 3(1-x)^2 \mathrm{e}^{-x^2-(y+1)^2} - 10(x/5 - x^3 - y^5)\mathrm{e}^{-x^2-y^2} - \mathrm{e}^{-(x+1)^2-y^2}/3$$

The computational domain is [−3,3]×[−3,3], −6.5510<*f*(*x*, *y*) <8.1062.

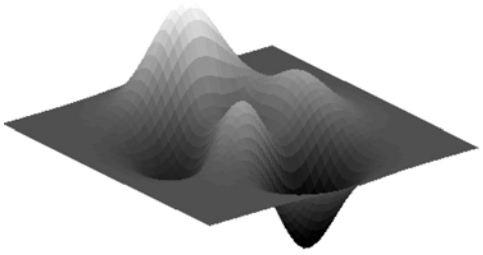

Fig. 1 Gaussian synthetic surface

The first test is that we fix sampling interval (*m*=4), inner accuracy tolerance ($\max_{i,j}(f_{i,j}^n - f_{i,j}^{n+1}) < 10^{-7}$) and outer iterative times (1 times), and change the number of sampling points in the computational domain to compare the computational efficiency of MGS with GS. From Table 1 and Fig.2, we can get that the computational time difference of MGS and GS is proportional to the second power of the total number of grid cells, which indicates that the bigger the computational domain, the more efficiency of MGS is. The relationship between the time difference of MGS and GS and the number of grid points is expressed as,

$$t = 2.2336 - 3.5292\times10^{-7}\mathrm{gn} + 6.472\times10^{-13}\mathrm{gn}^2 \quad (R^2 = 0.9925) \tag{22}$$

where, *t* is the time difference of the two methods, gn is the number of grid cells.

From Table 1, we can get that the inner iteration of MGS is always smaller than that of GS. Although the iterative times difference of the two methods becomes smaller with the number of grid cells increasing, the time difference is still increaseing, which is due to the increasing time of each iterative cycle.

The second test is that we fix the sampling interval (*m*=4), number of grid cells (1001×1001) and outer iterations (5 times),

Table 1 Efficiency comparison between MGS and GS

gn	CPU time/S		Time difference	Inner iteration	
	GS	MGS	GS–MGS	GS	MGS
101×101	0.3997	0.3952	0.0045	160	96
301×301	3.4381	3.1122	0.3259	131	81
501×501	9.5758	8.5352	1.0406	116	72
1001×1001	38.5613	34.0659	4.4955	90	56
2001×2001	157.3256	136.6111	20.7145	73	46
3001×3001	355.0508	310.8424	44.2083	63	40
4001×4001	984.5033	820.0605	164.4428	56	26

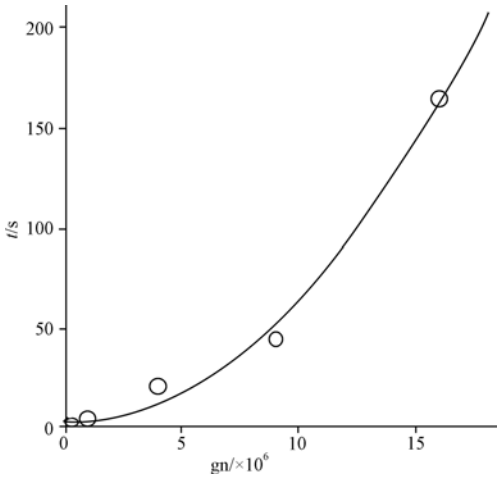

Fig. 2 Regression curve of time difference against total number of grid cells

and change the inner iterations to compare the RMSEs of the two methods. The test results are shown in Table 2.

From Table 2, we get that when the inner iteration is small, MGS can obtain higher accuracy than GS. In this test, when the inner iterative times are 320, GS can also reach its convergence.

The third test is that we fix the sampling interval (*m*=4), the number of sampling points (1001×1001) and inner iterations (50 times), and change the outer iterations to compare the RMSEs of the two methods. The results are shown in Table 3.

From Table 3 we can get that the lesser the outer iterations, the more accuracy of MGS is. With the increasing of outer iterations, both GS and MGS can reach the same convergence.

In the fourth test, we fix the sampling interval (*m*=4), inner and outer iterations (10 times and 5 times), and change the number of grid cells to validate the computing time of MGS. The results are shown in Table 4.

From Table 4 and Fig.3, we can get that the computing time of MGS is proportional to the first power of the total number of grid cells in the computational domain. The regression relationship between the number of grid cells and the computing time can be expressed as,

Table 2 RMSE comparison between MGS and GS under different inner iterations

Inner iterations	GS/×10^{-3}	MGS/×10^{-3}	GS-MGS/×10^{-3}
5	6.4517	3.9587	2.4390
10	3.4696	2.0536	1.4160
20	1.8455	1.3144	0.5311
40	1.2548	1.1298	0.1250
80	1.1217	1.1122	0.0090
160	1.1120	1.1119	0.0001
320	1.1119	1.1119	0

Table 3 RMSE comparison between MGS and GS under different outer iterations

Outer iteration	GS/×10^{-5}	MGS/×10^{-5}	GS–MGS/×10^{-5}
2	15.524	15.178	0.3460
4	6.6014	6.4578	0.1436
8	3.4646	3.4426	0.0220
10	3.0331	3.0203	0.0128
20	2.1085	2.1065	0.0020
25	1.8870	1.8865	0.0005
28	1.7848	1.7848	0

Table 4 MGS CPU time under different number of grid cells

Number of grid cells	CPU time/s
101×101	0.289764
201×201	0.805835
401×401	3.253838
801×801	13.038501
1601×1601	52.220574
3201×3201	208.628482

$$t = 2.036\times10^{-5}\,\mathrm{gn} + 0.010\ (R^2 = 1) \quad (23)$$

Where, t is the computing time, gn is the number of grid cells.

In former researches, the sources of computing time of the classical HASM were mainly from the differential equation simulation, inverse matrix computation, matrix multiplication, and linear system solution (Yue *et al*., 2004). The DEM construction in Dafosi Shaanxi province indicated that the computing time is about 10 hours to simulate 1500×1500 grid cells with the classical HASM (Song & Yue, 2009). In this paper, we only used 208.63 seconds to simulate 3201×3201 grid cells with HASM based on MGS. The computing time of MGS decreases two order of magnitude compared with the direct methods for solving HASM.

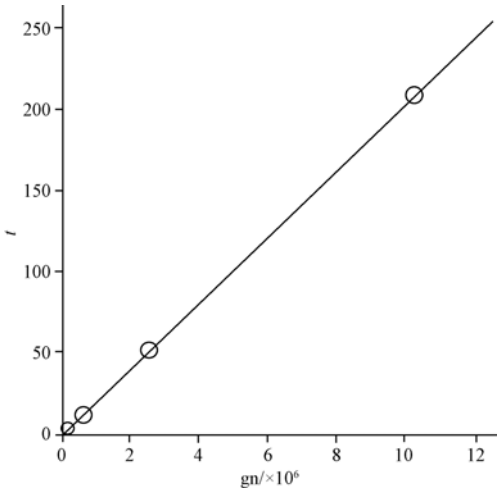

Fig. 3 MGS regression curve of computation time against total number of grid cells

4 REAL WORLD EXAMPLE

Dongzhi tableland (36°—37°N, 107°—108°E) was em ployedas a real world example to compare the performance of HASM based on MGS with those of the classical interpolation methods including IDW, TIN, KRIGING and NEAREST with the default parameters performed in ARCGIS 9.3. Dongzhi tableland was located in Gansu province, in the middle of loess plateau, north of jin river, south of malian river. The Dongzhi tableland was incised to be a fragmented landform. The complex landform is very suitable for DEM construction test. The source of the data is from the SRTM3 with the resolution of 90 m. In this study area, the number of grid cells is about 1201×1201, half of which were randomly selected for DEM construction, the others for DEM accuracy validation.

The simulation results of the DEM are shown in Table 5. From Table 5, we can obtain that the RMSEs of KRIGING, IDW, TIN, NEAREAST are 2.4, 1.8, 1.3 and 2.7 times as much as that of HASM.

Table 5 RMSE comparison among different methods

Method	HASM	KRIGING	IDW	TIN	NEAREST
RMSE /m	9.8	24.0	17.2	12.9	26.8
Ratio	1.0	2.4	1.8	1.3	2.7

5 CONCLUSIONS

Former researches indicated that the computational cost of the classical HASM seriously influences its widespread application. In this paper, MGS was employed to solve HASM. Firstly, we proved the efficiency of MGS in terms of theory; secondly we compared the computational accuracy of MGS with that of GS based on Gaussian synthetic surface. The results indicated MGS is more accurate than GS. Dongzhi tableland in Gansu province as a real world example showed that HASM based on MGS is more accurate than the classical interpolation methods including IDW, TIN, KRIGING and NEAREST.

The pre-smoother and post-soother of classical Multi-grid method is GS. In this paper, we proved MGS is more efficient than GS, so MGS can take place of GS as a smoother. HASM has much potential in computing speed improvement. An adaptive method adapts the finite-difference mesh to place more grid points in regions where high resolution is needed (Berger, 1989), while using fewer grid points in regions where a coarser mesh is sufficient to adequately resolve the resolution, which can save much space and computational cost (Piquet & Vasseur, 1998; Liu, 1995). So the future effort is toward developing an adaptive method of HASM.

REFERENCES

Al-Kurdi A and Kincaid D R. 2006. LU-decomposition with iterative refinement for solving sparse linear systems. *Journal of Computational and Applied Mathematics*, **185**(2): 391—403

Berger M J. 1989. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*, **82**: 64—84

Bramble J H and Pasciak J E. 1992. The analysis of smoothers for multigrid algorithms. *Mathematics of Computation*, **58**(198): 467—488

Brandt A. 1977. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, **31**(138): 333—390

Chen C F, Yue T X, Du Z P and Lu Y M. 2010. DEM construction based on HASM and related error analysis. *Journal of Remote Sensing*, **14**(1): 80—84

Chen C F, Du Z P and Yue T X. 2010. A research of SRTM void-filling based on high accuracy surface modelling. *Journal of Geodesy and Geodynamics*, **30**(1): 126—129

Davis T J. 2006. Direct Methods for Sparse Linear Systems. Philadelphia: SIAM

Liu C Q. 1995. Multi-Grid Method and its Application in Fluid Dynamics. Beijing: Tsinghua Publisher

Piquet J and Vasseur X. 1998. Multigrid preconditioned Krylov subspace methods for three-dimensional numerical solutions of the incompressible Navier–Stokes equations. *Numerical Algorithms*, **17**: 1—32

Saad Y. 2003. Iterative Methods for Sparse Linear Systems. Philadelphia: Society for Industrial Mathematics

Ujević N. 2006. A new iterative method for solving linear systems. *Applied Mathematics and Computation*, **179**(2): 725—730

Yue T X and Du Z P. 2006. Comparison analysis between HASM and the classical methods. *Progress in Natural Science*, **16**(8): 986— 991

Yue T X , Du Z P, Song D J and Gong Y. 2007. The accuracy loss and solution methods of HASM applications. *Progress in Nature*, **17**(5): 624—632

Yue T X, Du Z P and Song D J. 2007. A new method of surface modelling and its application to DEM construction. *Geomorphology*, **91**(1—2): 161—172

高精度曲面模型解算改进的 Gauss-Seidel 法

陈传法[1], 岳天祥[1], 刘洪涛[2]

1. 中国科学院 地理科学与资源研究所, 北京 100101
2. 济南市勘察测绘研究院, 山东 济南 250013

摘 要: 为了降低 HASM 的时间复杂度, 采用一种改进 Gauss-Seidel(GS)算法(MGS)解算 HASM 方程组。首先, 从理论上分析了 MGS 算法收敛速度快于 GS 算法, 然后以高斯合成曲面作为研究对象, 用四组模拟试验表明, 相同的网格数、达到相同的计算精度, MGS 算法计算时间小于 GS 算法, 且两种算法时间差与模拟区域网格数呈二次线性相关; 固定网格数, 使用相同的内迭代或者外迭代次数, MGS 算法精度高于 GS 算法, 但增加内迭代或者外迭代次数, GS 算法同样收敛; MGS 算法计算时间与网格数呈线性相关。MGS 算法能够有效解决 HASM 模拟大区域的计算时间瓶颈, 提高 HASM 运算速度。以甘肃省董志塬某测区 SRTM3 作为研究对象, 基于 MGS 的 HASM 用于模拟 DEM 表明, HASM 精度要高于传统的插值方法。

关键词: GS 迭代, 曲面模拟, 精度, 试验分析, 插值

中图分类号: TP751.1 **文献标识码:** A

引用格式: 陈传法, 岳天祥, 刘洪涛. 2010. 高精度曲面模型解算改进的 Gauss-Seidel 法. 遥感学报, **14**(4): 742—750
Chen C F, Yue T X and Liu H T. 2010. New method for solving high accuracy surface modeling. *Journal of Remote Sensing*. **14**(4): 742—750

1 引 言

自主研发的高精度曲面建模(HASM)方法在理论上已经趋于成熟(岳天祥 & 杜正平,2006; 陈传法等, 2010a)。HASM 计算可分为 3 个过程: Gauss 方程系数矩阵的生成、采样方程的建立和代数方程组的求解。当求算的区域为矩形区域, 采样方程为一阶截断时, 代数方程组的求解方法影响 HASM 的计算效率(Al-Kurdi & Kincaid, 2006; 岳天祥等, 2007)。直接法解算 HASM 的计算复杂度与模拟区域网格数的三次方呈正比(Davis, 2006), 庞大的计算量严重制约其推广使用(Yue 等, 2007)。

目前, 迭代法为解算方程组的有效方法(Saad, 2003), 其中, Gauss-Seidel(GS)迭代法具有存储空间少的优点被广泛使用(Bramble & Pasciak, 1992; Ujević, 2006), 但 GS 算法解算 HASM 收敛速度较慢。为此, 本文采用一种改进的 GS 迭代算法(Modified Gauss-Seidel, MGS)解算 HASM 方程组。以高斯合成曲面作为研究对象, 验证 MGS 算法的收敛性; 以甘肃省董志塬某测区 SRTM3 为研究对象, 验证基于 MGS 的 HASM 的计算精度。

2 MGS 迭代法

2.1 MGS 原理

设 HASM 模拟最终转换为求解线性方程组 $Sx^{k+1}=b^k$, 具体推导过程可见参考文献陈传法等, (2010b)。其中系数矩阵 S 为对称正定矩阵, $S\in R^{n\times n}$, $T\in R^n$, k 为迭代次数。更新 b 的过程为外迭代, 解算方程的过程为内迭代。

基于 GS 算法解算 HASM 的分量迭代形式为:

for $k=0,1,2,\cdots$ //表示迭代次数
  $y_1=x_k$ //第k次迭代
  for $i=1,2,\cdots,n$
    $y_{i+1}=y_i+\alpha_i e_i$
  end for i

收稿日期: 2009-06-01; 修订日期: 2009-08-07

基金项目: 国家杰出青年科学基金(编号: 40825003), 国家高新技术发展计划(编号: 2006AA12Z219), 国家科技支撑计划课题(编号: 2006BAC08B)和中国科学院知识创新工程重要方向项目(编号: kzcx2-yw-429)。

第一作者简介: 陈传法(1982—), 男, 山东沂源人。中国科学院地理科学与资源研究所博士生。目前研究方向: 曲面建模和 DEM。E-mail: chencf@lreis.ac.cn。

$$\begin{aligned}&x_{k+1}=y_{n+1}\\&\text{stopping criteria}\\&\text{end for } k\end{aligned}\tag{1}$$

其中,

$$\begin{aligned}&e_1=(1,0,0,\cdots,0)^{\mathrm{T}},e_2=(0,1,0,\cdots,0)^{\mathrm{T}},\\&\cdots,e_n=(0,0,0,\cdots,1)^{\mathrm{T}}\end{aligned}\tag{2}$$

$$\alpha_i=-\frac{p_i}{s_{ii}},p_i=(S_i,y_i)-b_i\tag{3}$$

设 $f(y_{i+1})$在点 y_i 处二阶可导, 则 $f(y_{i+1})$在 y_i 点处泰勒级数展开得:

$$\begin{aligned}f(y_{i+1})&=f(y_i+\alpha_ie_i)=f(y_i)+\alpha_i(Sy_i-b,e_i)+\\&\frac{1}{2}\alpha_i^2(Se_i,e_i)=f(y_i)+\alpha_ip_i+\frac{1}{2}\alpha_i^2s_{ii}\end{aligned}\tag{4}$$

由式(3)和式(4)得:

$$f(y_i+\alpha_ie_i)-f(y_i)=-\frac{1}{2}\frac{p_i^2}{s_{ii}}\leqslant 0\tag{5}$$

即

$$f(y_{i+1})-f(y_i)=-\frac{1}{2}\frac{p_i^2}{s_{ii}}\leqslant 0\tag{6}$$

表明 GS 算法能收敛。

令

$$z_{i+1}=y_i+h_i+\gamma_iq_i\tag{7}$$

其中, $h_i=-\frac{p_i}{s_{ii}}e_i$, $\gamma_i\in R,q_i\in R^n$, 其余符号意义同(3)。

设函数 $f(z_{i+1})$在点 y_i 处二阶可导, 则函数 $f(z_{i+1})$在该点泰勒级数展开得:

$$\begin{aligned}f(y_i+h_i+\gamma_iq_i)&=f(y_i)+(f'(y_i),h_i+\gamma_iq_i)+\\&\frac{1}{2}(S(h_i+\gamma_iq_i),h_i+\gamma_iq_i)\\&=f(y_i)+(Sy_i-b,h_i)+\gamma_i(Sy_i-b,q_i)+\frac{1}{2}(Sh_i,h_i)+\\&\gamma_i(Sq_i,h_i)+\frac{1}{2}\gamma_i^2(Sq_i,q_i)\\&=f(y_{i+1})+\gamma_i[(Sy_i-b,q_i)+(Sq_i,h_i)]+\\&\frac{1}{2}\gamma_i^2(Sq_i,q_i)\end{aligned}\tag{8}$$

令

$$g(\gamma)=\gamma_i[(Sy_i-b,q_i)+(Sq_i,h_i)]+\frac{1}{2}\gamma_i^2(Sq_i,q_i)\tag{9}$$

则 $g'(\gamma)=(Sy_i-b,q_i)+(Sq_i,h_i)+\gamma_i(Sq_i,q_i)$, $g''(\gamma)=(Sq_i,q_i)\geqslant 0$, 表明函数 $g(\gamma)$ 有极小值。

令 $g'(\gamma)$=0 得:

$$\gamma_i=-\frac{(Sy_i-b,q_i)+(Sq_i,h_i)}{(Sq_i,q_i)}\tag{10}$$

故

$$\begin{aligned}g(\gamma_i)&=-\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)}+\\&\frac{1}{2}\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)^2}(Sq_i,q_i)\\&=-\frac{1}{2}\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)}\leqslant 0\end{aligned}\tag{11}$$

由式(8)和式(11)得:

$$\begin{aligned}f(z_{i+1})-f(y_i)&=f(y_{i+1})-f(y_i)-\\&\frac{1}{2}\frac{[(Sy_i-b,q_i)+(Sq_i,h_i)]^2}{(Sq_i,q_i)}\end{aligned}\tag{12}$$

由式(12)表明, z_{i+1} 相比 y_{i+1} 使函数 f 更接近极小值。

MGS 伪代码为:

$$\begin{aligned}&\text{for } k=1,2,\cdots \text{//表示迭代次数}\\&z_1=x_k;\text{//第}k\text{次迭代}\\&\text{for } i=1,2,\cdots,n\\&z_{i+1}=z_i+h_i+\gamma_iq_i;\\&\text{end for } i\\&x_{k+1}=z_{n+1};\\&\text{stopping criteria}\\&\text{end for } k\end{aligned}\tag{13}$$

其中,

$$\begin{aligned}&h_i=-\frac{\overline{p}_i}{s_{ii}}e_i,\overline{p}_i=(S_i,z_i)-b_i,\\&\gamma_i=-\frac{(Sy_i-b,q_i)+(Sq_i,h_i)}{(Sq_i,q_i)}\end{aligned}\tag{14}$$

由式(1)和式(13)可见, MGS 每次迭代除了完成 GS 更新α_ie_i外, 还对 GS 模拟结果进一步更新λ_iq_i, 使模拟结果收敛速度更快。

2.2 q 的确定

由 MGS 原理可见, q_i 可以选取任意的 n 维向量。本文使 q_i=e_j, (i≠j), 因此

$$z_{i+1}=z_i+h_i+\gamma_ie_j\tag{15}$$

其中,

$$h_i=-\frac{\overline{p}_i}{s_{ii}}e_i,\overline{p}_i=(S_i,z_i)-b_i\tag{16}$$

$$\begin{aligned}\gamma_i&=-\frac{(Sz_i-b,e_j)+(Se_j,h_i)}{(Se_j,e_j)}=-\frac{\overline{p}_j-\frac{\overline{p}_i}{s_{ii}}s_{ij}}{s_{jj}}\\&=-\frac{\overline{p}_j}{s_{jj}}+\overline{p}_i\frac{s_{ij}}{s_{ii}s_{jj}}\end{aligned}\tag{17}$$

其中,

$$\overline{p}_j=(S_j,z_i)-b_j\tag{18}$$

j 可以有多种方式计算, 本文采用: j=i–1, (i=2,

3,⋯,n), j=n, (i=1)

$$z_i = z_{i-1} + h_{i-1} + \gamma_{i-1}e_{i-2},$$
$$h_{i-1} = -\frac{\overline{p}_{i-1}}{s_{i-1,i-1}}e_{i-1},(i=1,2,\cdots,n) \tag{19}$$

其中,

$$\gamma_0 = \gamma_n, e_0 = e_n, e_{-1} = e_{n-1}, h_0 = h_n, \overline{p}_0 = \overline{p}_n, s_{00} = s_{nn}$$

由式(18)和式(19)得:

$$\overline{p}_j = (S_j, z_i) - b_j = (S_{i-1}, z_{i-1} + h_{i-1} + \gamma_{i-1}e_{i-2}) - b_{i-1}$$
$$= (S_{i-1}, z_{i-1}) - b_{i-1} - \frac{\overline{p}_{i-1}}{s_{i-1,i-1}}(S_{i-1}, e_{i-1}) + \gamma_{i-1}(S_{i-1}, e_{i-2})$$
$$= \overline{p}_{i-1} - \frac{\overline{p}_{i-1}}{s_{i-1,i-1}}s_{i-1,i-1} + \gamma_{i-1}s_{i-1,i-2} = \gamma_{i-1}s_{i-1,i-2} \tag{20}$$

式中, $s_{0,-1}$=$s_{n,n-1}$,$s_{1,0}$=$s_{1,n}$。

由式(17)和式(20)得:

$$\gamma_i = -\gamma_{i-1}\frac{S_{i-1,i-2}}{S_{i-1,i-1}} + \overline{p}_i\frac{S_{i,i-1}}{S_{i-1,i-1}S_{ii}} \tag{21}$$

3 高斯合成曲面模拟

以高斯合成曲面为研究对象, 设计了 4 组试验验证 MGS 和 GS 解算 HASM 的效率, 即分别改变模拟区域网格点数, 内、外迭代次数比较 MGS 算法和 GS 算法, 固定内外迭代次数研究 MGS 算法计算时间与网格数的关系。高斯合成曲面(图 1)的数学表达式为:

$$f(x,y) = 3(1-x)^2\mathrm{e}^{-x^2-(y+1)^2} -$$
$$10(x/5 - x^3 - y^5)\mathrm{e}^{-x^2-y^2} - \mathrm{e}^{-(x+1)^2-y^2}/3$$

研究区域为[−3,3]×[−3,3],

$-6.5510 < f(x,y) < 8.1062$。

试验 1 为固定采样间隔(m=4)、内迭代停机精度($\max_{i,j}(f_{i,j}^n - f_{i,j}^{n+1}) < 10^{-7}$)和外迭代次数(1 次), 改变模拟区域网格点数比较 MGS、GS 的计算时间(CPU 时间)以及停止迭代时内迭代次数。由表 1 和图 2 表明, MGS 算法计算时间要小于 GS 算法, 且 MGS 计算时间与 GS 的差值随着网格数的增多呈较好的二次线性相关, 即网格数越多, MGS 算法的优势越明显, 两种算法的时间差值与网格数的关系为:

$$t = 2.2336 - 3.5292\times10^{-7}\mathrm{gn} + 6.472\times10^{-13}\mathrm{gn}^2 \quad (R^2 = 0.9925) \tag{22}$$

式中, t 为计算时间差值, gn 为网格数目。

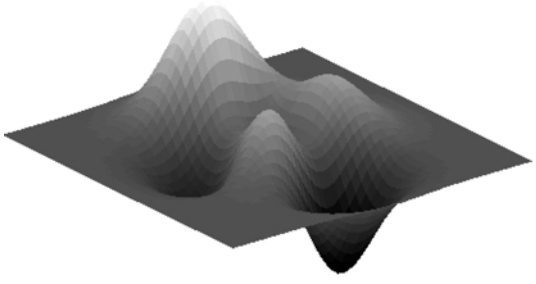

图 1 高斯合成曲面

表 1 MGS 和 GS 计算效率比较

网格数目 gn	CPU 时间/s		差值 t/s	内迭代次数	
	GS	MGS	GS−MGS	GS	MGS
101×101	0.3997	0.3952	0.0045	160	96
301×301	3.4381	3.1122	0.3259	131	81
501×501	9.5758	8.5352	1.0406	116	72
1001×1001	38.5613	34.0659	4.4955	90	56
2001×2001	157.3256	136.6111	20.7145	73	46
3001×3001	355.0508	310.8424	44.2083	63	40
4001×4001	984.5033	820.0605	164.4428	56	26

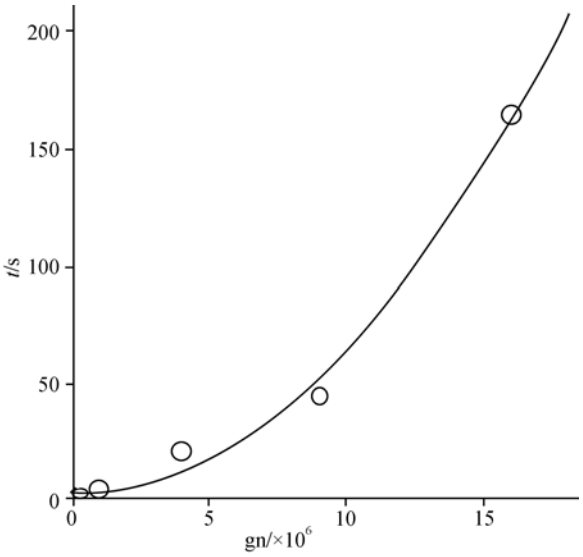

图 2 GS 和 MGS 时间差值与网格数回归曲线

比较内迭代次数可见, MGS 内迭代次数始终小于 GS。尽管随着网格数的增多 MGS 和 GS 迭代次数差值在减少, 但由于网格数越多, 每次迭代时间就越多, 因此两者时间差始终在增加。

试验 2 为固定采样间隔(m=4), 模拟区域网格数(1001×1001)和外迭代次数(5 次), 改变内迭代次数研究两种方法计算中误差。试验结果如表 2。

表 2 不同内迭代次数, MGS 和 GS 计算中误差比较

内迭代次数	GS/×10^{-3}	MGS/×10^{-3}	GS−MGS/×10^{-3}
5	6.4517	3.9587	2.4390
10	3.4696	2.0536	1.4160
20	1.8455	1.3144	0.5311
40	1.2548	1.1298	0.1250
80	1.1217	1.1122	0.0090
160	1.1120	1.1119	0.0001
320	1.1119	1.1119	0

由表 2 表明, 与 GS 相比, 当内迭代次数较少时, MGS 能够达到较高的精度, 且收敛速度快于 GS。在该试验中, 当内迭代次数为 320 时, GS 也得到稳定解, 表明只要增加方程组的内迭代次数, GS 方法同样能得到稳定解。

试验 3 为固定采样间隔(m=4), 模拟区域网格数(1001×1001)和内迭代次数(50 次), 改变外迭代次数验证 MGS 和 GS 的计算中误差。计算结果如表 3。由表 3 可见, 外迭代次数越少, MGS 相比 GS 精度越高, 表明 MGS 收敛速度要快于 GS, 但随着外迭代次数的增加, GS 和 MGS 均收敛于相同的值。

试验 4 为固定采样间隔(m=4), 内外迭代次数(内迭代为 10 次、外迭代为 5 次), 改变模拟区域的网格数研究 MGS 的计算时间。试验结果如表 4。

由表 4 和图 3 可见, MGS 计算时间与网格数呈非常好的线性相关关系, 其回归方程为:

$$t=2.036\times 10^{-5}\mathrm{gn}+0.010\ (R^2=1) \tag{23}$$

式中, t 为计算时间, gn 为网格数目。

表 3 不同外迭代次数, **MGS** 和 **GS** 中误差比较

外迭代次数	GS/$\times10^{-5}$	MGS/$\times10^{-5}$	GS−MGS/$\times10^{-5}$
2	15.524	15.178	0.3460
4	6.6014	6.4578	0.1436
8	3.4646	3.4426	0.0220
10	3.0331	3.0203	0.0128
20	2.1085	2.1065	0.0020
25	1.8870	1.8865	0.0005
28	1.7848	1.7848	0

表 4 **MGS** 计算时间与模拟区域网格数比较

网格数	CPU 时间/s
101×101	0.289764
201×201	0.805835
401×401	3.253838
801×801	13.038501
1601×1601	52.220574
3201×3201	208.628482

4 实例验证

实例选择甘肃省董志塬某区域(36°—37°N, 107°—108°E)。董志塬地处黄土高原中部, 南临泾河, 北以马莲河的支流蔡家庙沟和蒲河一级支流小黑河为界, 地处马莲河、蒲河两大河流之间。长期的严重水土流失, 把董志塬切割的支离破碎, 千沟万壑, 复杂的地形非常适合 DEM 模拟研究。原始数据来源于 90m 分辨率的 SRTM3, 该测区内 SRTM3 文件包括高程点数为 1201×1201。其中, 选择 50%数据用于精度检验, 其余 50%用于 DEM 模拟。除了基于 MGS 的 HASM 模拟 DEM 外, ARCGIS 9.3 中传统的插值方法如: 克里格法(KRIGING)、反距离权重法(IDW)、线性插值(TIN)、最邻近法(NEAREST)也用于模拟 DEM, 各插值方法使用的参数为 ARCGIS 9.3 中的默认值。各种方法的模拟精度如表 5。

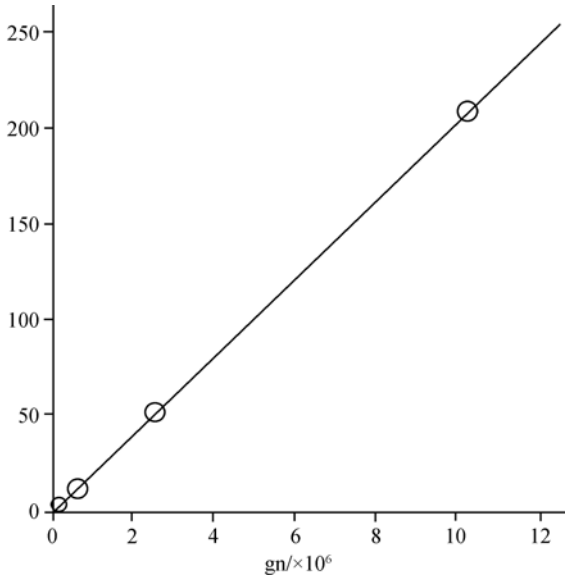

图 3 MGS 计算时间和网格数的回归曲线

由表 5 可见, 基于 MGS 的 HASM 插值精度要高于传统的方法。其中, KRIGING, IDW, TIN, NEAREST 的 RMSE 分别是 HASM 的 2.4, 1.8, 1.3 和 2.7 倍。

表 5 各种方法的模拟 **RMSE** 比较

方法	HASM	KRIGING	IDW	TIN	NEAREST
误差/m	9.8	24.0	17.2	12.9	26.8
比值	1.0	2.4	1.8	1.3	2.7

注: 比值表示传统方法 RMSE 与 HASM 比值

5 结 论

以往研究表明, HASM 计算量问题严重制约其推广使用。采用 MGS 解算 HASM, 首先从理论角度表明了 MGS 算法的收敛速度快于 GS 算法, 然后以高斯合成曲面为研究对象, 采用 4 组试验表明, MGS 算法优于 GS 算法, 且 MGS 算法的时间与模拟区域的网格数呈线性相关, 比直接法解算 HASM 时间复杂度降低两个数量级。以甘肃董志塬某测区 SRTM3 为研究对象表明, 基于 MGS 的 HASM 模拟 DEM 精度高于传统的方法。

HASM 在计算速度方面仍然有很大提升潜力。

自适应算法(Brandt,1977)根据模拟区域的地形复杂度自动调整网格分辨率, 即在地形复杂区域用细网格模拟, 在平坦区域用粗网格模拟(Berger,1989), 从而在保证模拟精度的同时提高计算速度(刘超群,1995; Piquet & Vasseur,1998)。因此本文下一步工作为发展自适应 HASM。

REFERENCES

Al-Kurdi A and Kincaid D R. 2006. LU-decomposition with iterative refinement for solving sparse linear systems. *Journal of Computational and Applied Mathematics*, **185**(2): 391—403

Berger M J. 1989. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*, **82**: 64—84

Bramble J H and Pasciak J E. 1992. The analysis of smoothers for multigrid algorithms. *Mathematics of Computation*, **58**(198): 467—488

Brandt A. 1977. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, **31**(138): 333—390

Chen C F, Du Z P and Yue T X. 2010a. A research of SRTM void-filling based on high accuracy surface modelling. *Journal of Geodesy and Geodynamics*, **30**(1):126－129

Chen C F, Yue T X, Du Z P and Lu Y M. 2010b. DEM construction based on HASM and related error analysis. *Journal of Remote Sensing*, **14**(1): 80－84

Davis T J. 2006. Direct Methods for Sparse Linear Systems. Philadelphia: SIAM

Liu C Q. 1995. Multi-Grid Method and its Application in Fluid Dynamics. Beijing: Tsinghua Publisher

Piquet J and Vasseur X. 1998. Multigrid preconditioned Krylov subspace methods for three-dimensional numerical solutions of the incompressible Navier–Stokes equations. *Numerical Algorithms*, **17**: 1—32

Saad Y. 2003. Iterative Methods for Sparse Linear Systems. Philadelphia: Society for Industrial Mathematics

Ujević N. 2006. A new iterative method for solving linear systems. *Applied Mathematics and Computation*, **179**(2): 725—730

Yue T X and Du Z P. 2006. Comparison analysis between HASM and the classical methods. *Progress in Natural Science*, **16**(8):986—991

Yue T X , Du Z P, Song D J and Gong Y. 2007. The accuracy loss and solution methods of HASM applications. *Progress in Nature*, **17**(5):624—632

Yue T X, Du Z P and Song D J. 2007. A new method of surface modelling and its application to DEM construction. *Geomorphology*, **91**(1—2): 161—172

附中文参考文献

陈传法, 杜正平, 岳天祥. 2010a. 基于高精度曲面建模方法的 SRTM 空缺插值填补.大地测量与地球动力学, **30**(1): 126－129

陈传法, 岳天祥, 杜正平, 卢毅敏. 2010b. 基于高精度曲面模型的 DEM 构建与误差分析. 遥感学报, **14**(1): 85－89

刘超群. 1995. 多重网格法及其在计算流体力学中的应用. 北京: 清华大学出版社

岳天祥, 杜正平. 2006. 高精度曲面建模与经典模型的比较分析. 自然科学进展, **16**(8): 986－991

岳天祥, 杜正平, 宋敦江, 龚云. 2007. HASM 应用中的精度损失问题和解决方案. 自然科学进展, **17**(5): 624—632