2014 Sixth International Symposium on Parallel Architectures, Algorithms and Programming

A Parallel Implementation of Acoustic Propagation Model on Cloud Platform

Binbin Yu Simin Cheng Dan Wang Xichen Wu
Hongkun Mai
School of Computer Science
University of Science and Technology
Hefei, China
{yubb, chengsm, danewang, wxc422, maihk}@mail.ustc.edu.cn

Guangzhong Sun
School of Computer Science
University of Science and Technology
Hefei, China
gzsun@ustc.edu.cn

***Abstract*— Three dimensional wave propagation model of parabolic approximation type is widely used in exploring the ocean. For this application, FOR3D model is one of the mostly used models, for it takes the azimuthal coupling into consideration and the result gains a better precision. When the precision requirement is high, large scale computation task would be faced, which cannot be solved only in one computer or constrained time. In this paper, we propose a parallel method to decompose the computation task, which divide the original computation into small size. Then each processor get one piece of task and run the FOR3D independently. Our method was implemented on Windows Azure and the result shows that we almost gain a linear speed-up.**

***Keywords*—acoustic propagation model; FOR3D; parallel application; Windows Azure**

I. INTRODUCTION

As the ocean is almost "opaque" to light wave and electromagnetic wave, the acoustic is the only effective Information Carrier in the ocean [1], so people create a variety of underwater acoustic equipment for underwater detection, navigation, positioning, communication, remote sensing and remote control. On the civilian side, sonic can be used in fish detecting, exploring submarine topography and sediment seabed prospecting, etc. On the military side, sonic is mainly used in submarines, torpedoes, sonar and other aspects. As the ocean plays as the wave propagation channel, which is extremely complex and time-varying. So exploring how the acoustic propagates in ocean becomes very essential. It is clear that acoustic propagates in three dimensions—range, depth, azimuth. Then we can build a three dimensional wave propagation model and get the Helmholtz equation (shown in equation (1) in section Ⅱ).

Helmholtz equation is a high order pseudo-partial differential equation. In mathematics, such equation is hard to get an absolutely precise solution. So based on different approximate strategy, more than one kind of model is proposed to solve this problem, such as RMPE3D, FOR3D and so on. FOR3D is one of the mostly used models, which is proposed by Ding Lee et.al in [2]. FOR3D takes more factors into consideration about real environment. Solving such three-dimensional acoustic propagation problems is still difficult because the ocean must be modeled by a large number of environmental parameters. Besides, even if the problem is formulated elegantly by means of mathematical and physical theory, the implementation of such a solution into realistic computation results into a large-scale computation problem.

With the development of science and technology, more data now is captured and requires higher precision, which results in large scale computation task. The computation cannot be solved only in one computer or constrained time. So how to solve such problem becomes a more urgent topic. Nowadays, hardware has been developed in an unprecedented speed, so some scholars has proposed some ideas to solve this problem.

In this paper, we propose a parallel program for FOR3D. We validate our experiment on Windows Azure. The following part is organized as follows: we introduce the FOR3D model in detail in part Ⅱ. The main experiment and result is introduced in part Ⅲ. For the last part, we conclude our whole work and talks about the future work.

II. PRELIMINARIES

A. FOR3D[2, 3, 4]

By modeling the three dimensional acoustic propagation, we gain the following equation:

$$\frac{\partial^2 P}{\partial r^2}+\frac{1}{r}\frac{\partial P}{\partial r}+\frac{1}{r^2}\frac{\partial^2 P}{\partial \theta^2}+\frac{\partial^2 P}{\partial z^2}+k_0^2 n^2(r,\theta,z)P=0 \qquad (1)$$

Let $P=P(r,\theta,z)$ denotes the acoustic propagation in space, $n(r,\theta,z)=c_0/c(r,\theta,z)$ denotes the index of refraction.

Let $P(r,\theta,z)=\mu(r,\theta,z)\nu(r)$, (1) turns out to be like

$$\frac{\partial^2 \mu}{\partial r^2}+2ik_0\frac{\partial \mu}{\partial r}+\frac{1}{r^2}\frac{\partial^2 \mu}{\partial \theta^2}+\frac{\partial^2 P}{\partial z^2}+k_0^2[n^2(r,\theta,z)-1]\mu=0 \qquad (2)$$

Initial condition: $\mu(r_0,\theta,z)=\mu_0(\theta,z)$

Surface condition: $\mu(r,\theta,z)=\mu_s(r,\theta)$

Bottom condition: $\mu(r,\theta,z_B)=\mu(r,\theta)$

End condition: $\mu_w(r_w,\theta,z)=\mu_w(\theta,z)$

Then we define that:

$$X=n^2(r,\theta,z)-1+\frac{1}{k_0^2}\frac{\partial^2 \mu}{\partial z^2}, \qquad Y=\frac{1}{k_0^2 r^2}\frac{\partial^2 \mu}{\partial \theta^2}$$

We get equation (3)

$$\mu_r=ik_0(-1+\sqrt{1+X+Y})\mu \qquad (3)$$

2168-3034/14 $31.00 © 2014 IEEE
DOI 10.1109/PAAP.2014.57

B. Classifications

Before talking about how to solve equation (3), we classify the problem based on azimuthal variation:

3-D： If we consider both the θ dependence and θ coupling, the model is exactly described as (3).

N*2-D：If the θ coupling is negligible that we can ignore in computation, (3) can be expressed like

$$\mu_r = ik_0(-1+\sqrt{1+X})\mu \tag{4}$$

2-D：If θ coupling and θ is negligible enough that we can ignore in computation, (3) can be expressed as:

$$\mu_r = ik_0(-1+\sqrt{1+X'^2}) \tag{5}$$

X' is defined as:

$$X' = n^2(r,z) - 1 + \frac{1}{k_0^2}\frac{\partial^2\mu}{\partial z^2}$$

Compared to 3-D, N*2-D do not take θ coupling into consideration. The N*2-D [5] terminology was introduced to solve the solution of (uncoupled) 2-D propagation problems in N vertical planes. Finally, if bothθ coupling and θ dependence are negligible, the problem is 2-D problem.

Using the approximation,

$$\sqrt{1+X+Y} \cong \frac{1+p_1X+p_2Y}{1+q_1X+q_2Y} \cong 1+\frac{1}{2}X-\frac{1}{8}X^2+\frac{1}{2}Y \tag{6}$$

Then we can get the numerical solution to (4)

$$\mu(r+\Delta r,\theta,z) = \left(\frac{1+\left(\frac{1}{4}+\frac{\delta}{4}\right)X}{1+\left(\frac{1}{4}-\frac{\delta}{4}\right)X}\right)\left(\frac{1+\frac{\delta}{4}Y}{1-\frac{\delta}{4}Y}\right)\mu(r,\theta,z) \tag{7}$$

Equation (7) can be written into another formula:

$$PQ\mu^{j+1} = P^*Q^*\mu^j \tag{8}$$

In order to solve (8), write $Q\mu^{j+1} = \omega^{j+1}$, then we use the following two-step to solve the above equation (8):

First step: $\left[1+\left(1+\left(\frac{1}{4}-\frac{\delta}{4}\right)X\right)\right]\omega^{j+1} = [1+\left(\frac{1}{4}+\frac{\delta}{4}\right)X](1+\frac{\delta}{4}Y)\mu^j$

Second step: $\left(1-\frac{\delta}{4}Y\right)\mu^{j+1} = \omega^j$

To get the element of P and Q, substitute the definitions of X and Y.

For parallel consideration, supposing that there is no energy exchange in adjacent sectors, we can compute the equations separately by a section. As shown above, the 3-D terminology is so computational, and 2-D results in less accuracy. So in this paper, we focus only on N*2-D terminology.

If we analyze the FOR3D in detail, we find that the whole process of solving three dimensional problem consists of several independent tri-diagonal solves equation (2) in azimuthal direction, which is computational. Fig. 1 shows a three dimensional view of one sector, In order to enhance the computation speed, we parallel compute these solves. That is to say, we first parallel compute the solution atr, parallel compute the solution at r + Δr, then parallel compute at r + 2Δr, until reach the expected receiver range.

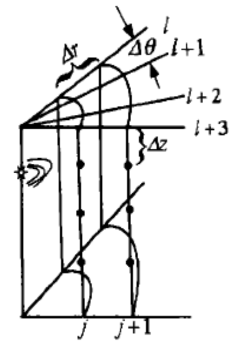

Fig. 1：3D view of one sector [2]

Suppose that there is no energy exchange [6] between adjacent sectors, the computation of each sector is independent from the others, which means that there is no data exchange between processors. So we divide the computation task into small pieces for multiple processors and gain speed-up.

C. Windows Azure[7]

Windows azure is a cloud computing platform and infrastructure, created by Microsoft, for building, deploying and managing applications and services through a global network of

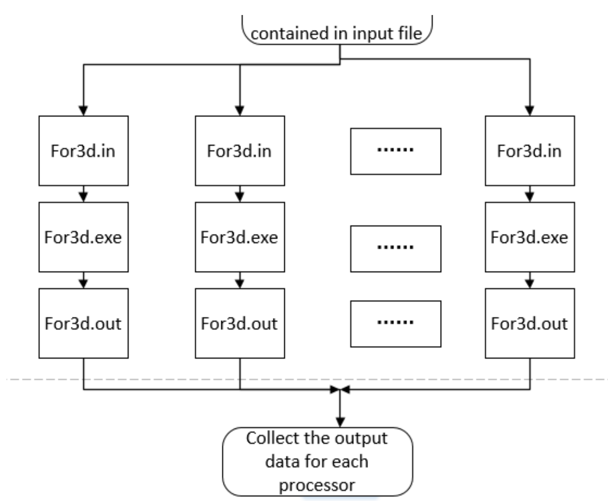

Fig. 2: the process of parallel FOR3D

Microsoft-managed datacenters. It provides both PaaS and IaaS services and supports many different programming languages, tools and frameworks, including both Microsoft-specific and third-party software and systems.

III. EXPERIMENT

A. PARALLEL FOR3D

In the following part, we propose a parallel FOR3D based on the traditional program [6, 7]. Fig. 2 shows the process of how the parallel FOR3D is organized. First, divide the data into several part (almost equal), then each processor get a piece of input data and computation task. Each processor do the computation independently almost without data exchanging.

B. DATA PARTITION

Denote n as number of solutions and m to be the number of processors. There are two ways to divide the data. One way is to crossly divide the data. In this way, processor $i(i \leq m)$ get the data for solution $j(j \leq n)$, i and j satisfy that j mod m =i-1. The other way is to divide successively. In this way, for processor 1 to n%m, each processor get the data from solution $i * \lceil n/m \rceil$ to $i * \lceil m/n \rceil$, and for processors greater than n%m, each processor get the data for from solution $n\%m * \lceil m/n \rceil + 1$ to $n\%m * \lceil m/n \rceil + \lfloor m/n \rfloor * (i - n\%m)$. The first method is distinct and easy, but we need to run the program several times. The latter one is also not so hard and each processor only need to run the program once. In our experiment, we choose the latter one for data partition, with no overlapping data among processors.

Fig. 3 shows the top view of cylindrical wedge for acoustic propagation. We also have the rule that divide the data for adjacent solutions into one sub-input file.

C. THE EXPERIMENT IMPLEMENTATION ON WINDOWS AZURE[8, 9, 10]

Windows Azure is a representative cloud computing system. It provides a windows-based cloud computing environment for running applications and storing data on servers in Microsoft data centers. It's important to note that the latency on Windows Azure is not negligible. But for our

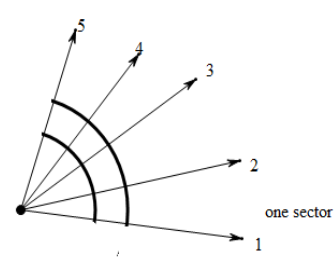

Fig. 3: top view of cylindrical wedge

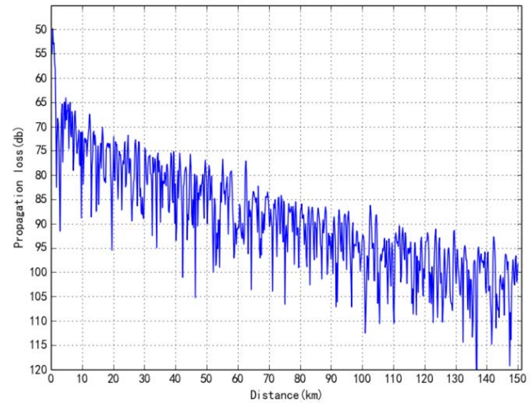

Fig. 4: Result comparison

experiment, there is almost no communications among processors. So we choose Windows Azure for computation, not the high performance clusters with low latency.

Our experiment is implemented on Windows Azure with multiple processors, 56G memory (RAM). Set the max range 150km, iteration step to 20m, and frequency to 100Hz and 50Hz. Fig. 4 shows the propagation loss in 90° direction performed on Windows Azure. And the result properly fit the real situation. The result shows that there is no obvious difference between the result for parallel FOR3D and traditional FOR3D. This means that the parallel FOR3D can reach the precision requirement.

Then we run our experiment with different number of processors to quantify the speed-up. Table Ⅰ shows the relationship between program running time and the number of processors. The more processors we use, the fewer running time we get.

TABLE Ⅰ. The computation time(s) with different processors

FRQ \ np	1	2	3	4	5	6	7	8
50	609.8	327.4	227.2	174.7	143.5	118.1	107.3	92.0
100	730.4	389.7	268.3	203.7	167.4	138.1	124.5	106.3

Fig. 5 directly illustrate the relationship between processors and computation time speed-up, with frequency of 50Hz and 100Hz respectively. The result shows that parallel FOR3D almost gain a linear speed-up. Then we visualize the acoustic propagation loss from source to receiver, Fig. 6 shows the acoustic field reconstruction result from 0° to 360°, the change of color from blue to red shows the propagation loss.

From the result we can see that if there are enough number of processors, the computation time can reduced to constrained time.

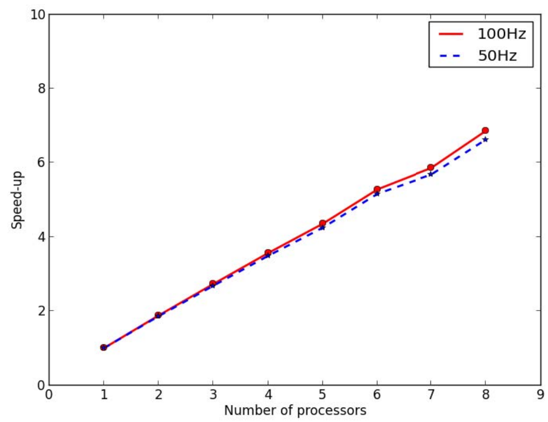

Fig. 5: Speed-up with different processors, with frequency of 50Hz and 100Hz respectively.

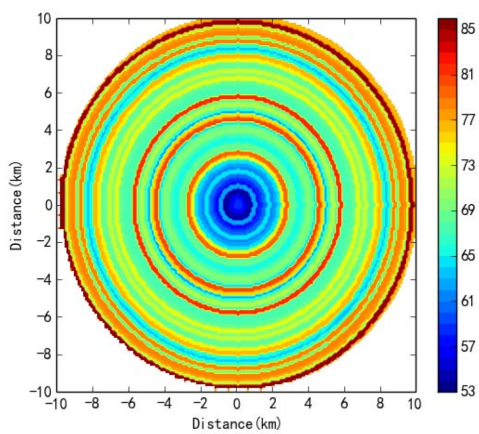

Fig. 6: Acoustic field reconstruction result

IV. CONCLUSION AND FUTURE WORK

Sonic plays a very important part in exploring the ocean and FOR3D is one of the mostly used model in this field. In this paper, we propose a parallel FOR3D version, which can not only satisfy the precision requirement, but also gain a linear speed-up. This means that if we get enough processors, the computation can be done in certain time.

In this paper, we focus on how to build a parallel model for FOR3D, so a simple seabed topography with same slope for each sector is considered. But in reality, seabed topography is a great factor in deciding how much calculation we do and may turn into a bottleneck for this model. At the same time, although the result for our work gain a linear speed-up, but we still cannot ignore the data distribution time. So in the future, more complicated seabed topography and data distribution time will be considered.

V. ACKNOWLEDGEMENT

This work is supported by Natural Science Foundation of China (No. 61033009 and No. 61303047). This work is supported by Anhui Provincial Natural Science Foundation (No.1208085QF106). We would also like to thank the Microsoft Research Asia for supporting us to use the Windows Azure computing resource in our experiment, and the Third Institute of State Oceanic Administration for providing us technical support.

REFERENCES

[1] Lu Xiaoting, Zhang Lin. "Summarization of modeling underwater acoustic propagation". Ocean Technology, 2011, 29(4): 48-53.

[2] Ding Lee, Martin H.Schultz, "Numerical ocean acousticpropagation in three dimensions". Singapare: World Scientific, 1995.

[3] Lee D, Botseas G, Siegmann W L. "Examination of three - dimensional effects using a propagation model with azimuth - coupling capability (FOR3D)". The Journal of the Acoustical Society of America, 1992, 91(6): 3192-3202.

[4] Lee D, Resasco D. C, Schultz M. H, et al. "A parallel 3-D parabolic wave equation solver". Singapore, World Scientific, 1994.

[5] J.S Perkins, R. N. Baer. "An approximation to the three dimensional parabolic equation method for acoustic propagation". Acoust. Soc. Am 72 515-522, 1982

[6] Fan Peiqin, Da Lianglong. "The parallel computing of underwater acoustic propagation in three dimensions". 2009 National Academic Conference and underwater acoustics branch of underwater acoustics reelection Conference Proceedings 2009.

[7] D. Chappell. "Introduction the Windows Azure platform". http://go.microsoft.com/?linkid=9752185, 2010.

[8] FOR3D model: http://oalib.hlsresearch.com/PE/for3d/

[9] Chen Guoliang, An Hong, Chen Ling, Zheng Qilong, Shan jiulong. "Practice of Parallel Algorithms". Higher Education Press, 2004.

[10] Da Lianglong. Modeling and Application of Underwater Acoustic Environmental Effect. Science Press. 2012.