J. Appl. Environ. Biol. Sci., 4(9S)224-230, 2014

© 2014, TextRoad Publication

ISSN: 2090-4274
Journal of Applied Environmental and Biological Sciences
www.textroad.com

Role of Parallel Computation Technology in Developed Scientific Fields

Mehrdad Hashemi[1], Shabnam Ahari[2]

[1]Department of Applied Mathematics & Cybernetics, Baku State University, Baku, Azerbaijan
[2]Department of Computer Engineering, Islamic Azad University, Ahar branch

Received: September 12, 2014
Accepted: November 23, 2014

ABSTRACT

In this paper the role of distributed and parallel computing in different fields of science and engineering is discussed. Our purpose in this introductory article is highlighting the impacts of using multi processor environment for achieving more accuracy and less execution time and memory usage in solving different arithmetical and logical problems which potentially are considered talented for parallelism. Here we brought concrete and interesting examples from different fields of science which we can apply parallel and distributed structure in place of traditional sequential structures for solving them. Also a concrete applied problem related to oil extraction industries investigated here.
KEYWORDS: Parallel computation, multi processor systems, computational problem

1 INTRODUCTION

It may be you are one of those people for whom "fast" isn't fast enough. Today's workstations are about hundred times faster than those made just a decade ago, but the most of computational scientists and engineers need even more speed. They make great simplifications to the problems they are solving and still must wait hours, days or even weeks for their programs to finish running. Faster computers let yon tackle larger computations. Suppose you can afford to wait overnight for your program to produce a result. If your program suddenly ran 10 times faster, previously out-of-reach computations would now be within your grasp. You could produce in 15 hours an answer that previously required nearly a week to generate. Of course, you *could* simply wait for CPUs to get faster. In about five years CPUs will be 10 times faster than they are today (a consequence of Moore's Law). On the other hand, if you can afford to wait five years, you must not be in that much of a hurry! Parallel computing is a proven way to get higher performance now. Parallel computing is the use of a parallel computer to reduce the time needed to solve a single computational problem. Parallel computing is now considered a standard way for computational scientists and engineers to solve problems in areas as diverse as galactic evolution, climate modeling, aircraft design and molecular dynamics and etc. Many important scientific problems are so complex that solving them vianumerical simulation requires extraordinarily powerful computers. These complex problems, often called grand challenges for science, fall into several categories: Quantum chemistry, statistical mechanics, and relativistic physics, cosmology and astrophysics, computational fluid dynamics and turbulence, materials design and super conductivity, biology, pharmacology, genome sequencing, genetic engineering, protein folding, enzyme activity, and cell modeling, medicine, and modeling of human organs and bones, global weather and environmental modeling.

Applications that explore, query, analyze, visualize, and, in general, process very large scale data sets are known as *Data Intensive Applications*. Large scale data intensive computing plays an increasingly important role in many scientific activities and commercial applications, whether it involves data mining of commercial transactions, experimental data analysis and visualization, or intensive simulation such as climate modeling. By combining high performance computation, very large data storage, high bandwidth access, and high-speed local and wide area networking, data intensive computing enhances the technical capabilities and usefulness of most systems. The integration of parallel and distributed computational environments will produce major improvements in performance for both computing intensive and data intensive applications in the future. The purpose of this introductory article is to provide an overview of the main issues in parallel data intensive computing in scientific and commercial applications and to encourage the readers to go into the more in-depth articles later in this special field. Of course, we can't easily change all sequential solution methods existing for a computational problem into parallel version. There are some important factors that impact on performance of a parallel algorithm. For example, when we design a

*** Corresponding Author:** Mehdard Hashmi, Department of Applied Mathematics & Cybernetics, Baku State University, Baku, Azerbaijan. Email: mehan121@yahoo.com

parallel algorithm for a special problem we should note that any parallel construction can be created just in some situations like below: the focused problem containing a mathematical function that (f) is expensive to evaluate, as might be the case if each f evaluation requires the solution of an auxiliary problem, the number of equations (n) in the system is large, the interval of integration is long and or the Initial Value Problems (IVP) must be solved repeatedly, as happens in parameter fitting problems. Firstly, some examples of using parallel and distributed systems explained here [1]. Secondly, we would like to discuss about a concrete applies problem related to oil extraction industry that can be solved in parallel form. General information and other details will be explained in that section.

2. **APPLIED AND SCIENTIFIC EXAMPLES TALENTED FOR PARALLELISM**.

In many fields of science we can find hundreds applied and computational problems that they can be modeled in mathematical base. For example, most of such problems can be written in the form of linear on non linear system of equations. Also they can be written in the form of ordinary differential equations, either linear or non linear form. As proof, we can mention some examples from different areas of science that these problems can be found there more than usual. We would like to divide these examples into two basic and engineering categories. In the both categories we brought examples that scientists applied parallel structure for solving considered problems in order to access more accuracy, time saving and also optimal memory usage.

2.1. Basic sciences: About problems related to physics it will be interesting knowing that the natural parallel and distributed structures of beam physics problems allow the use of parallel and distributed computer systems. But the usual approaches based on traditional numerical methods demand using the resources of supercomputers. This leads to the impossibility of using such multiprocessing systems as computational clusters. In many research papers some examples of beam physics problems are discussed from the computational point of view using clustered systems [2]. Related to physics we can encounter with some other interesting problems that are talented for parallelism. Generally, we can say that any problem containing optimization aspect can be talented for parallelism. Our main discussed problem in this paper is related to optimization of results obtained from a mathematical problem. But specially about applying parallel computing in physics, the global optimization is playing an increasing role in physics, chemistry, and biophysical chemistry. One of the most important applications of global optimization is to find the global minima of the potential energy of molecules or molecular assemblies, such as crystals. The solution of this problem typically requires huge computational effort. Even the fastest processor available is not fast enough to carry out this kind of computation in real time for the problems of real interest, e.g., protein and crystal structure prediction. One way to circumvent this problem is to take advantage of massively parallel computing. You can find detailed information and real examples about this in [3].

Parallel computing also will likely play an important role in integrating spatial environmental models for large-scale systems. Inter visibility analysis with error simulation in a digital elevation model is used to illustrate an approach to developing parallel models, and to demonstrate some benefits of high-performance computing. Analyzing the structure of the application problem ensures an appropriate match between problem and parallel system implementation. Data communication is the most important computational issue in this application. Then we can strongly say that the role of multi processor environment and parallel computing strategy in aero cosmic and aerospace field is inevitable [4].

Advances in high performance computing are making it possible to calculate the properties of materials accurately and reliably from the fundamental laws of quantum mechanics. Many non-technical explanations are given of how the calculations are done, and of how massively parallel computing is already playing an important role in the field. The new possibilities that are opening up are illustrated by recent works on properties of materials under extreme conditions. The key importance of high performance computing for the future development of different fields of science is indicated in many articles and research reports related to using parallel computation in science and engineering [5].

The use of parallel computing is gaining increasing popularity in geographic information systems applications too. There exists a class of spatial analysis algorithms that are based on local computation and are single step, hence leading to simple and efficient parallel code. For another class of algorithms it is not possible to make any assumption about the locality of computation, for example when extracting complex or global terrain features, and a number of iteration may be necessary to satisfy a convergence criteria, giving rise to non-local iterative algorithms. An example is the algorithm to extract drainage basins from digital terrain models. Despite the increasing difficulties there is an interest in parallelizing non-local iterative algorithms. The number of applications that require parallel and high-performance computing techniques has diminished in recent years due to the continuing increasing in power of PCs, workstation and mono-processor systems. In natural fields of science as geographic topics using parallel computation strategy is developed too. However, Geographic information systems (GIS) still provide a resource-hungry application domain that can make good use of parallel techniques. There are many works which discuss about geographical systems for environmental and defensive applications and some of the algorithms and techniques we have deployed to deliver high-performance prototype systems that can deal with large data sets. GIS applications are often run operationally as part of decision support systems with both of a human interactive component as well as large scale batch or server-based components. Parallel computing technology embedded in a distributed system therefore provides an ideal and practical solution for multi-site organizations and especially government agencies who need to extract the best value from bulk geographic data. The distributed computing approaches which used to integrate bulk data and metadata sources and the grid computing techniques has been described in related works and reports. In situation like this embed parallel services in an operational infrastructure should be used. We describe some of the parallel techniques we have used: for data assimilation; for image and map data processing; for data cluster analysis; and for data mining. We also discuss issues related to emerging standards for data exchange and design issues for integrating together data in a distributed ownership system. We include a historical review of our work in this area over the last decade which leads us to believe parallel computing will continue to play an important role in GIS. We speculate on algorithmic and systems issues for the future [6, 7].

In recent years, in our world the problems related to water and oil products transmission are become as the one of most vital and critical aspects of any government and society. Absolutely, using parallel computation technology in this field can help developing and improving this section. Most of problems in mentioned area can be modeled and changed into mathematical and numerical problems and we can applied parallel strategies in this situation. Efficient numerical tools taking advantage of the ever increasing power of high-performance computers, become key elements in the fields of energy supply and transportation, not only from a purely scientific point of view, but also at the design stage in industry. Indeed, flow phenomena that occur in or around the industrial applications such as gas turbines or aircraft are still not mastered. In fact, most Computational Fluid Dynamics (CFD) predictions produced today focus on reduced or simplified versions of the real systems and are usually solved with a steady state assumption. By investigating in related researches it can be shown how recent developments of CFD codes and parallel computer architectures can help overcoming this barrier. With this new environment, new scientific and technological challenges can be addressed provided that thousands of computing cores are efficiently used in parallel. Strategies of modern flow solvers are discussed with particular emphases on mesh-partitioning, load balancing and communication. These concepts are used in two CFD codes developed by CERFACS: a multi-block structured code dedicated to aircrafts and turbo-machinery as well as an unstructured code for gas turbine flow predictions. Leading edge computations obtained with these high-end massively parallel CFD codes are illustrated and discussed in the context of aircrafts, turbo-machinery and gas turbine applications. In general, it can be said that future developments of CFD and high-end computers are proposed to provide leading edge tools and end applications with strong industrial implications at the design stage of the next generation of aircraft and gas turbines [8].

2.2. Engineering fields: Newly, a parallel optical processor for implementing arithmetic operations is presented. The presented processor addresses some of the architectural complexities of the past parallel optical processors. In some papers nontrivial computational problems have been adapted to the presented processor. As the special topic in parallel computing technology the optical processor for performing arithmetic operations in parallel systems is developed. The implementation of arithmetic operations makes it possible to perform various computational tasks. As case studies the bounded subset sum problem in parallel has been solved and parallel execution testing has been performed. The processor uses two-dimensional optoelectronic planes for both performing logic operations and storing data, which eliminates the need for transferring data from electronic devices in each step of the computation. The presented processor seems easier to realize than most of the past parallel optical processors due to its simpler and more compact architecture, while staying powerful enough to carry out computations from diverse applications [9].

Absolutely, for implementation of any parallel algorithm in a parallel environment we need to a parallel programming language to test its performance and superiority in comparing with traditional sequential version. Parallel programming is programming in a language that allows you to explicitly indicate how different portions of the computation may be executed concurrently by different processors. Existing programming languages have some facilities for parallel processing. But some of these languages are stronger than the others and have instructions or commands which help programmers to implement parallel operations. A lot of research has been invested in the development of compiler technology that would allow ordinary Fortran 77 or C programs to be translated into codes that would execute with good efficiency on parallel computers with large numbers of processors. This is very difficult problem and while many experimental parallelizing compilers have been developed, at the present time commercial systems are still in their infancy. The alternative is for you to write your own parallel programs.

In object-oriented programming (OOP) languages, the ability to encapsulate software concerns of the dominant decomposition in objects is the key to reaching high modularity and loss of complexity in large scale designs. However, distributed-memory parallelism tends to break modularity, encapsulation, and the functional independence of objects, since parallel computations cannot be encapsulated in individual objects, which reside in a single address space. For reconciling object-orientation and distributed-memory parallelism, OOPP (Object-Oriented Parallel Programming) is introduced, a style of OOP where objects are distributed by default. As an extension of C++, a widespread language in HPC, the PObC++ language has been designed and prototyped, incorporating the ideas of OOPP [10].

But MPI (Message Passing Interface) is a standard specification for message passing libraries. Libraries meeting the standard are available on virtually every parallel computer system. Free libraries are also available in case you want to run MPI on a network of workstations or a parallel computer built out of commodity component (PCs and switches). If you develop programs using MPI, you will be able to reuse them when you get access to a newer, faster parallel computer.

In electrical engineering field the parallel computation can play an inevitable role. Because electrical circuits are used in any field of industry and daily life more and more. Of course, by using parallel technology we will encounter by many inventions and innovations in mentioned field. Transmission and distribution of electricity involve technical as well as Non-Technical Losses (NTLs). Illegal consumption of electricity constitutes a major portion of the NTL at distribution feeder level. Considering the severity and devastating effects of the problem, illegal consumption of electricity has to be detected instantly in real-time. To this end, possibility and role of High Performance Computing (HPC) algorithms in detection of illegal consumers has become an important problem. We can find many methods and solutions that designs and implements an encoding procedure to simplify and modify customer energy consumption data for quicker analysis without compromising the quality or uniqueness of the data. Parallelization of overall customer classification process is exists too. Suggested different parallelized algorithms have resulted in appreciable form [11].

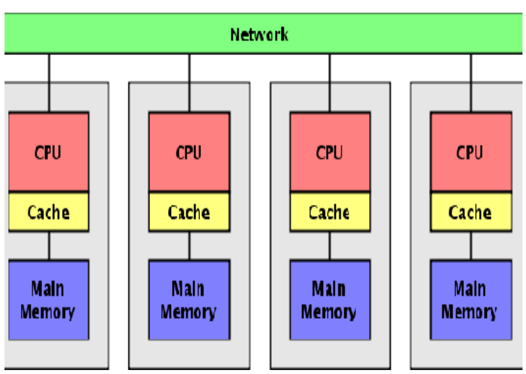

Figure 1. Multi processor architecture scheme

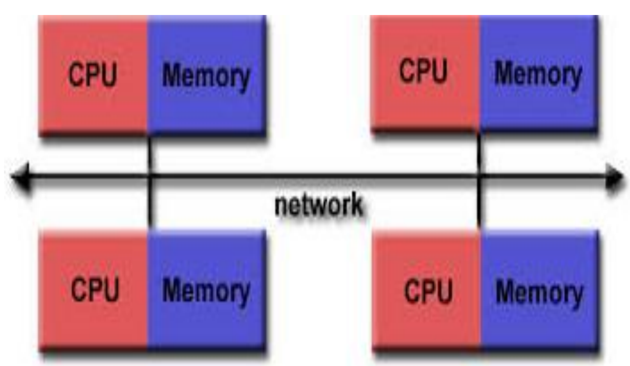

Figure 2. Multi computer architecture scheme

Finally, we would like to discuss about a concrete computational problem related to oil extraction industry which is named gas lift process. Here our main focus is on the parallelization of Gas-lift process. There are many scientific works focused on mathematical solving for oil extraction problems and specially related to gas-lift method. For more information you can see [12-14]. But in fact, there wasn't a concrete and comprehensive mathematical model for mentioned problem and even existing methods and algorithms can't reach to acceptable accuracy and fastness. In our research we are focusing the results arises from [15-17]. The problems of motion of fluids, gases and gas–liquid mixtures in pipes related to gas-lift oil recovery are mathematically formulated as Ordinary Differential Equations (ODE)s based on initial value problems

(IVP)s and it solved through Iterative Runge Kutta (IRK). We parallelized the traditional and sequential solution method and increase accuracy or obtained results using parallel computing environment. Figure 3 give a comprehensive image of gas lift oil well.

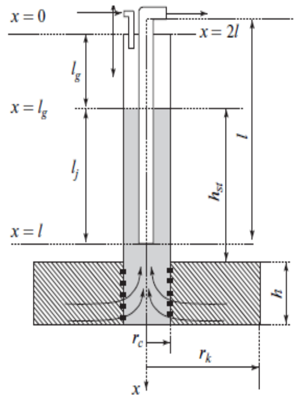

Figure 3 – Gas lift system applied on an oil well

The main purpose of this process is lifting oil from well by injecting gas into it. In fact, this process is one of EOR methods for more extracting of oil from wells. But this is so important finding the optimal amount of injected gas for lifting maximum amount of gas-oil mixture. As it seems, this is completely an optimization problem. We solved the equation gotten from mathematical model of this process in the form of non linear ODEs. Equations arise from this problem is as system (1) shown below:

$$Q' = \frac{2a_1\rho_1 F_1 Q^2}{{c_1}^2{\rho_1}^2{F_1}^2 - Q^2}, x \in [0, l-0] \; Q(0) = Q_0 \; (1)$$

$$Q' = \frac{2a_2\rho_2 F_2 Q^2}{{c_2}^2{\rho_2}^2{F_2}^2 - Q^2}, \quad x \in [l+0, 2l], \qquad Q(l+0) = \gamma Q(l-0) + (-\delta_3(Q(l-0) - \delta_2)^2 + \delta_1)\overline{Q}$$

Where coefficients used in the equation are as:

a_1=0.1008449644636985 ρ_1=0.717, f_1=0.006021647718768237 C_1=331,

a_2= -89.77276884123468 ρ_2=700, f_2= 0.004185386812745002 C_2=850.

γ=1, δ_1=0.1, δ_1=0.1, δ_1=0.02 and $\overline{Q}$ = 100.

But this solution is for some amount from given interval as IVPs. Using parallel computing we can solve mentioned equation for much more initial values from given interval in order to get much more results and increasing accuracy of solution. Increasing accuracy can be happened by increasing number of amounts from given interval and also it can be happened by decreasing the amount of a coefficient used in IRK. But by using one processor computer it can be executed in hours and days. In this purpose we apply parallel computing methods for solving mentioned problem. Numerical results and charts obtained from parallel and sequential algorithms and comparison of them proofed that our suggested algorithm can be more accurate and of course more fast.

3. **CONCLUSION**

With the industry-wide switch to multiprocessor and multicomputer architectures, parallel computing has become the only venue in sight for continued growth in application performance. In order for the performance of an application to grow with future generations of hardware, a significant portion of its computation must be done with scalable parallel algorithms. It is therefore important to develop and deploy as many scalable parallel algorithms as possible. This paper takes a critical look at the major challenges involved in the development of scalable parallel algorithms and points to needs for compiler tool innovations to help address these challenges. About parallel programming language which is the most important tool for implementation and comparison of designed parallel algorithm with traditional sequential version, we briefly can say that the PObC++ language implements the concept of Object-Oriented Parallel Programming (OOPP). OOPP reconciles distributed-memory parallel programming with OO programming principles. OOPP separates concerns about inter-object and inter-process communication. OOPP makes it possible the encapsulation of distributed-memory parallel computations in objects. Performance of PObC++ programs is almost similar to the performance of C++/MPI programs. Finally, as a general result of this paper, Parallel computing has been an enormous success. This is despite many arguments against it. Parallelism has become main stream, in the multi core chip, the GPU, and the internet data center running MapReduce. In our study field large-scale scientific computing, parallelism now reigns triumphant. About the gas lift process we investigated parallel solving of the concrete and computational mathematical problem related to oil extraction industry which is named the gas lift method. As mentioned above, the mathematical model for this problem exists and scientists have suggested some sequential traditional solutions for that. Now we can say that, about problems like this it absolutely will be better to use parallel computations for solving them. Specially, where we need to optimize obtained results for getting accuracy and optimal amount from a massive composed of results. In parallel version we save noticeable execution time and memory usage. Also parallel solution method allowed us to obtain more accurate answers and optimize the determined problem easily in comparison with sequential one by computing the equation for the large number of initial values from given interval. Another proof for accuracy of the parallel solution method in comparing with sequential method is that we can change the factor of accuracy through the method RK.

REFERENCES

1. Cannataro, M., Talia, D., Srimani, P. K. (2002). Parallel data intensive computing in scientific and commercial applications. *Parallel Computing*. Vol.28, No.5, pp. 673-704.
2. Andrianov, S. (2004). Role of parallel and distributed computing in beam physics. Proceedings of the Sixth International Conference on Charged Particle Optics. Journal name: *Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*. V. 519, N. 1–2, pp. 37-41.
3. Jooyoung, L., Jaroslaw, P., Cezary, C. and others. (2000) Efficient parallel algorithms in global optimization of potential energy functions for peptides, proteins, and crystals. *Computer Physics Communications*, V. 128, N. 1-2, pp. 399-411.
4. Mills, K., Fox, G., Heimbach, R. (1992). Implementing an inter visibility analysis model on a parallel computing system. *Computers & Geosciences*. V. 18, N. 8, pp. 1047–1054.
5. Gillan, M. J. (1994). First-principles calculations on materials using massively parallel computing. *Future Generation Computer Systems*. V. 10, N. 2–3, pp. 213–222.
6. Hawick, K. A., Coddington, P. D., James, H. A. (2003). Distributed frameworks and parallel algorithms for processing large-scale geographic data. *Parallel Computing*. V. 29, N. 10. pp. 1297–1333.
7. CLEMATIS, A., CODA, A., SPAGNUOLO, M. (1998) DEVELOPING NON-LOCAL ITERATIVE PARALLEL ALGORITHMS FOR GIS ON A WORKSTATION NETWORK. *PARALLEL AND DISTRIBUTED PROCESSING, 1998. PDP '98. PROCEEDINGS OF THE SIXTH EUROMICRO WORKSHOP ON*. MADRID, PP.250 – 256.
8. Gicquel, L. Y. M., Gourdain, N., Boussuge, J. F., Deniau, H., Staffelbach, G., Wolf, P., Poinsot, T. (2011). High performance parallel computing of flows in complex geometries. *Comptes Rendus Mécanique*. V. 339, N. 2–3, pp. 104–124.
9. Jalili, S., Gholami, R. A. (2013). A parallel optical implementation of arithmetic operations. *Optics & Laser Technology*. V. 49, pp. 173–182.

10. Gurgel, P. E., Carvalho Junior, F. H. (2014). An object-oriented parallel programming language for distributed-memory parallel computing platforms. *Science of Computer Programming*. V. 80, N. A, pp. 65–90.
11. Depuru, S. S. S. R., Wang, L., Devabhaktuni, V., Green, R. C.(2013). High performance computing for detection of electricity theft.*International Journal of Electrical Power & Energy Systems*. V. 47, pp. 21–30.
12. *Yagubov M.A., Zamanova R.B.*A Formula for the Gradient of a Functional and a Theorem on Existence of Optimal Control in a Gas Dynamics Problem. *Appl. Comput. Math.*, 2013, V. 12, N. **1**, pp. 97-102.
13. Mukhtarova N. S., Ismailov N. A. Algorithm to solution of the optimization problem with periodic condition and boundary control. *TWMS J. Pure Appl. Math.*, 2014, V. 5, N. 1, pp. 130-137.
14. Askerov I. M., Ismailov N. A. Asymptotic method for solution of the optimization problem with periodic boundary conditions and control in gas lift process, *TWMS J. Pure Appl. Math.* 2013, V. 4, N. 2, pp. 237-241.
15. *Aliev Fikret. A., IsmailovN.A.* Inverse Problem to Determine the Hydraulic Resistance Coefficient in the Gas Lift Process, *Appl. Comput. Math.* 2013, V. 12, N. 3, pp. 306-313.
16. Askerov, I., Ismailov N. Asymptotic method for solution of the optimization problem with periodic boundary conditions and control in gas lift process, 2013. *TWMS J. Pure Appl. Math.*, V. 4, N. 2, pp. 237-241.
17. *Aliev F.A., Ismailov N.A., Temirbekova L.N.* Extremal Solution of the Problem of the Choice of Optimum Modes for Gas Lift Process. *Appl. Comput. Math.* 2012, V. 11, N. 3, pp. 348-357.