General Methodology for Converting a Sequential Evolutionary Algorithm into Parallel Algorithm with MPI to Water Design Networks

Marco Antonio Cruz-Chávez, E.Y. Avila-Melgar, Sergio A. Serna Barquera, Fredy Juárez–Pérez

CIICAp, Universidad Autónoma del Estado de Morelos
Av. Universidad 1001. Col. Chamilpa, C.P. 62209. Cuernavaca, Morelos, México
{mcruz,erikay,aserna, juarezfredy}@uaem.mx

Abstract

This paper presents a general methodology for the conversion of the sequential evolutionary algorithms into parallel evolutionary algorithms using MPI. The methodology is showed by applying it to a combinatorial optimization problem well-known as Water Distribution Networks. Also a comparison of two parallelization approaches is done, showing the advantages and disadvantages of each approach. The implementation of this parallel algorithm reduces the necessary time to obtain good solutions for NP-Complete problems as Water Distribution Networks.

1. Introduction

Because of their stochastic nature there is no guarantee that the global optimum will be found using Genetic Algorithms, although the number of applications suggests a good rate of success in identifying good solutions [1].

Experimental research shows Genetic Algorithms can be effectively used for finding good solutions in “reasonable” computation times. Genetic algorithms are acknowledged to be inherently parallel.

Parallel computation has arisen to decrease the time spent for the algorithms searching for solutions. The parallelization of an algorithm is realized to address difficult problems [2] of realistic scale in which large problem instances executed by sequential algorithms would last many years searching for the optimal solution.

The main goal of parallelizing an algorithm is to decrease the sequential computing time by distributing processing among available processors. Parallelizing is based in the cooperation concept which implies having many processors working to improve the quality of the solution to solve a problem. Currently, parallelizing algorithms is an alternative when trying to solve optimization complex problems in which the entire solution space is so large. For example NP-Complete problems, such as Water Distribution Networks Design Problem.

The water distribution networks design belongs to a wide group of problems. Finding the optimum solution to these problems is extremely complex, even sometimes impossible [3]. These problems have been thoroughly studied in the last few decades, by diverse authors. In [4] the authors propose several methods and techniques in order to deal with this type of problem using theoretical models. But when attempts are made to solve these problems for real instances, it becomes increasingly complex to find the optimum solution.

The water distribution problem is classified as a complex optimization NP-hard problem [5]. Basically, it consists of finding the most efficient way to supply water to consumers, within given constraints. The problem can be solved from different stages: design, operation, rehabilitation and maintenance.

Given the network layout and pressure requirements to satisfy the users water demands, the water distribution network design problem consist in sizing the basic network components, like pipes, valves, pumps, reservoirs and so on. The principal goal is minimizing the overall costs by finding the least-cost design configuration of water distribution networks fulfilling the commercial and hydraulic restrictions.

Commercial pipes are important elements to carry the water from reservoirs to consumers. Valves are used to keep pressures constant and, pumps are necessary if the water distribution technique is by pumping.

Finally reservoirs are necessary distribute water to consumers.

Bhave [6] classifies water networks according their topology. Generally, water networks can be branched or looped.

A branched network is a tree-like network and has no loops. Usually it has one source node, one or more intermediate nodes and it has more than one sink. Frequently branched networks are used to provide supply in small rural communities, horticultural greenhouse. In real life, the principal problem that branched networks present is ruptures in pipes, causing loss of service in several points in the network. This happens because there is only one path between one point and another. In a looped network, the interruption of service due to ruptures in pipes happens less frequently. The design of these networks allows water to arrive to its destination via several trajectories. For this reason a break in some pipe does not usually gravely affect other points in the network. The efficient design of looped networks is a much more complex problem than the design of branched ones, but their greater reliability can compensate for the increase in cost when closing some loops [7].

Water Distribution Networks Design helps to the network to have a good performance. However it does not guarantee efficient water supply to consumers, satisfying their requirements and, it does not guarantee manage water resources properly.

The most of research realized has been focused on water distribution network design. However, water distribution network operation is important to use the available hydraulic resources properly: water distribution for all consumers and avoiding over-exploitation of reservoirs. This work is focused in water distribution network operation.

This paper is organized as follows. Section 2 presents a description of the principal problem to convert an existent sequential evolutionary algorithm into a parallel evolutionary algorithm to the water distribution problem. Section 3 presents a state-of-the-art survey for the Water Distribution Network. Section 4 presents a methodology for the conversion of the sequential evolutionary algorithms into parallel evolutionary algorithms using MPI. Section 5 asserts general conclusions of this work.

2. Water Distribution Network Problem

2.1 Problem Description

Water Distribution Network operation basically consist of providing the most efficient supply to consumers from sources. The problem consists of finding a program that carries out an efficient distribution to all the users of the network. The service distribution should be carried out in the least possible time, to minimize energetic costs caused by pumping, while satisfying the requirements of the users, principally pressures according to their demands.

Actually, the best supply from sources to consumers must be constant and uninterrupted. However, to date it has not been implemented in some countries because of the energetic high costs and availability of hydraulic resources.

Alternatively, Cruz-Chavez et al. [8] propose an approach to schedule the network operation guarantying efficient network operation and minimizing the energetic costs caused by the water distribution. The mathematical formulation of the water distribution network distribution has been set up in [8]. In the mathematical model, the objective function consists of minimizing the distribution cost of water. The model constraints are: the resource capacity, the physical design of the network, and the availability of the resources.

2.2 Genetic Algorithm

Genetic algorithms are stochastic search procedures based on the evolutionary mechanisms of natural selection and genetics. Genetic algorithms theory was proposed by Holland [9] and developed by Goldberg [10].

A population, in genetic algorithms is composed by a set of individuals. Each individual in a population is represented by a set of parameters that describes a solution. Each solution is codified into a chromosome structure to represent the analogy with the characters strings found in DNA in evolutionary natural selection. Traditionally, genetic algorithms used alphabetic or binary representation. Also, classic operators like selection, crossover and mutation techniques were used, Fig 1.

Genetic Algorithms have been applied successfully in different fields. They also have been applied for the water distribution network problem and they suggest a good rate of success in identifying good solutions. However, for large instances hours, days, or even years would be needed for finding optimal solution.

Because of the computation times could be very long, computation parallel has arisen. The objective is to decrease the long times employed searching for solutions. This paper presents a methodology for the conversion of the sequential evolutionary algorithms into parallel evolutionary algorithms as applied for the water distribution network problem.

The complexity of converting a sequential algorithm into parallel algorithm is principally communication among processors. The sequential algorithm is implemented in C language and contains complex dynamic data structures, Fig. 2. The structures must be sent and received in the processors iteratively in each generation of the genetic algorithm.

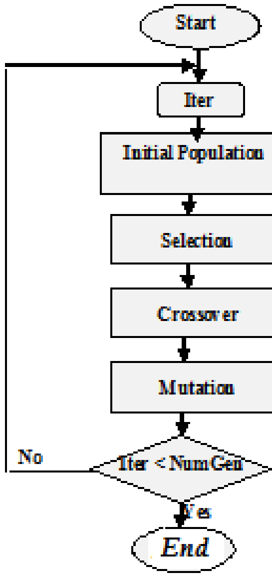

Fig. 1. Traditionally Evolutionary Algorithm

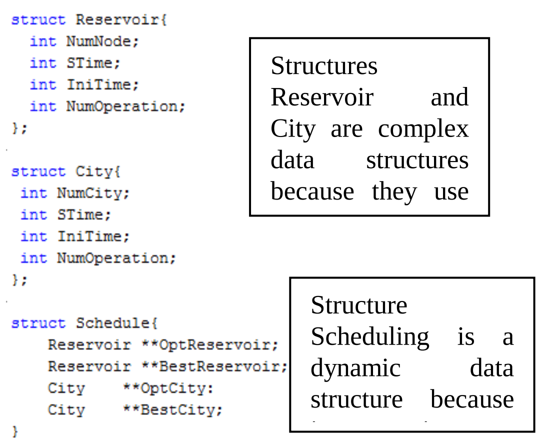

Fig. 2. Dynamic complex data structures

Dynamic data structures keep the information organized. But they use pointer to refer other data types and they are not recognized in parallel computing using Message Passing Interface (MPI).
Dynamic data-type transfer is an issue on all distributed systems, and if a message passage library provides means to transfer complex data-types, few provide their dynamic data-type counterpart.

3. State-of-the-art survey for the Water Distribution Network

Water Distribution Network Problem has been widely studied for many researchers. According to the presented characteristics, for this paper, the problem can be divided in two stages: classic and modern. In the classic stage the first attempts to solve this problem were based on Linear Programming techniques. Alperovits and Shamir [11] define the problem of optimizing the design of a water distribution system by sizing its components and setting operational decisions for pumps and valves under a number of loading conditions. Constrains are that demands are to be met and pressures at selected nodes in the network are to be within specified limits. Pipe diameters are stated as decision variables. They are based on the split-pipe variables. It means that each link could be divided into two or more different pipe sizes. Alperovits and Shamir [11] proposed a linear programming gradient method. The method used by Alperovits has been adapted and improved by Quindry [12], Goulter et al. [7], Fujiwara et al. [13], Kessler et al. [14], Gupta et al. [5], Eiger et al. [15], by using other solution methods, Table 1. Costs for the solutions are showed in Fig 3.

Table 1. Split-pipe Water Distribution Networks Design Problem

Date	Author	Method	Cost
1977	Alperovits et al.	Gradient	497525
1979	Quindry et al.	Gradient	441522
1986	Goulter et al.	Gradient	435015
1987	Fujiwara et al.	Quasi-Newton	415271
1989	Kessler et al.	Gradient	417500
1990	Loganathan et al.	Heuristic	412931
1993	Gupta et al.	Fletcher-Powell	407625
1994	Eiger et al.	Branch and Bound	402352
1995	Loganathan et al.	Heuristic	403561
1997	Varma et al.	Successive Quadratic Programming	441310

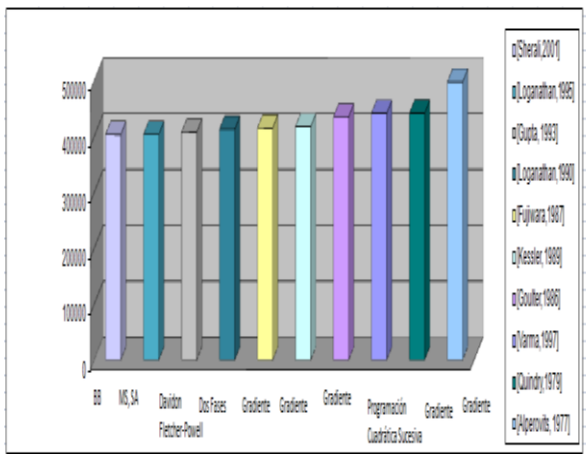

Fig. 3. Costs in Classic Stage for WDND

In last few decades, called in this paper modern stage, the water distribution network has gradually been modified.

Currently the water distribution network problem has been formulated as a non linear programming problem. Pipe diameters are stated as discrete decision variables.

Different heuristics have been proposed to deal with water distribution networks design. Even when heuristics does not guarantee that the global optimum will be found, the number of applications suggests a good rate of success in identifying good solutions, Table 2.

Table 2. Water Distribution Networks Design Problem based on discrete variables

Date	Authors	Method	Cost
1997	Savic y Walters	Genetic Algorithm	419000
1998	Abebe et al.	Global Optimization	422000
1998	Abebe et al.	Genetic Algorithm	424000
1998	Abebe et al.	Controlled Random Search	439000
1999	Montesinos et al.	Genetic Algorithm	456000
1999	Cunha y Sousa	Simulated Annealing	419000
2001	Geem et al.	Harmony Search	419000
2003	Eusuff y Lansey	Shuffled Frog Leaping Algorithm	419000
2003	Matías	Genetic Algorithm	419000
2006	Reca et al.	Genetic Algorithm	419000

Costs obtained in non programming linear approach can be seen in Fig 4. Decision variables are discrete.

Generally, topology is looped, solution methods are based on metaheuristics and, the network is fed by gravity technique to supply consumers.

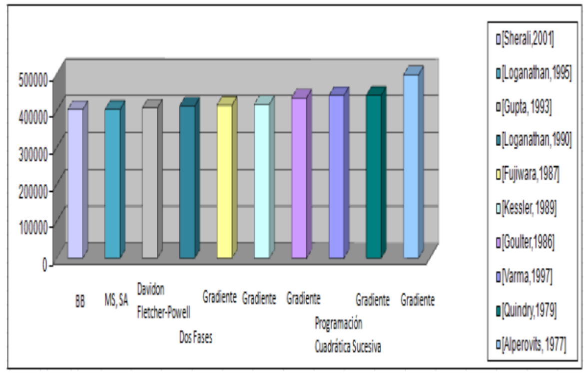

Fig. 4. Costs in Modern Stage for WDND

Some important differences between classic and modern stages are related with topology. Some researchers use additional components in networks, such as pumps, valves [6, 7, 8, 9, 10, 11, 16] while others suppose an ideal network model [17]. In addition, some authors deals with branched networks while others deal with lopped networks. Finally, some authors use gravity technique while others use pumping technique to feed the network

4. Methodology for Functional Parallelization of Evolutionary Algorithm

Functional parallelization consists of distributing tasks among the number of available processors.

A Master-Slave paradigm is used to keep the sequence of the sequential algorithm [18]. The master processor centralizes the population and it manages selection and replacements of individuals.

Master sends subpopulation to the slaves. Slaves receive subpopulations to evaluate them and then slaves return subpopulations to master.

The proposed methodology, for functional parallelization of evolutionary algorithm as applied to the water distribution networks consist of the next steps:

1. Given a sequential solution algorithm, the first step of parallelization is the analysis of the algorithm to identify the functions that require the most computational resources. Also, it is necessary to identify coupling functions to determine if they can work independent.
2. An exhaustive analysis should be realized on the sequential algorithm to identify cohesion and coupling of the code. If the code is highly coupled, communications among processors would be constantly, to send and receive data that must be modified in different functions. It is known that

constant communications in distributed computing really affect the throughput of the network.

It is very recommendable to avoid excessive communications among processors.

3. Data that will be sent to processors should be analyzed. The existence of dynamic data structures benefits because keeps the information organized and, information can be send together. However the library MPI, used to parallelize the algorithm, does not recognize dynamic data types. It is necessary to define a process equivalent to serialize and de-serialize. [19] Propose a methodology to convert manually a C language data type into MPI equivalent data type. Also a tool called Automap is proposed to convert data types automatically to MPI.
4. Once it is defined the possible division of tasks and data, it is important to identify the available processors. According to the available processors a manually or dynamic assignment can be done trying to have a load balance. The load balance lets processors to finish the assigned task almost at the same time, avoiding the processors have idle time.
5. Another important aspect to consider when parallelizing evolutionary algorithms is a synchronization process. It is because of the number of iterations that have to be carried out. The master processor has to get the data processed for the slaves each iteration and, using it, the master has to create a new population. Then the master sends new population to slaves. For the synchronization some techniques can be used: semaphores, barriers, or blocking communication defined in MPI library.
6. Finally, it is necessary to use some instructions of the MPI environment, Table 3.

Table 3. Instructions for MPI parallelization

FUNCTION	DESCRIPTION
MPI_Init	Initializes MPI environment
MPI_Finalize	Finalizes MPI environment
MPI_Bcast	Send data to processors
MPI_Gather	Get data from processors.
MPI_Scatter	Distribute Data to processors.
MPI_Comm_rank	Get processors identifiers
MPI_Comm_size	Gets the number of processors
MPI_Send	Sends data messages
MPI_Recv	Receive data messages
MPI_Wtime	Returns time in seconds
MPI_Get_processor_name	Returns the name of current processor.

For functional parallelization of evolutionary algorithm it was used the Master-Slave approach. For this work, the approach consists of a central node, called master. It distributes data and tasks to the slaves. The algorithm works with a unique population. The population is divided almost proportionally according to the available processors. Each slave works with a subpopulation applying the mutation operator and then returns the results to the master processor.

The master gets the results sent by slaves, it applies selection and crossover operators and, it sends to the slaves a new population to work with, Fig 5.

The principal difference between two approaches used for this work is the communication.

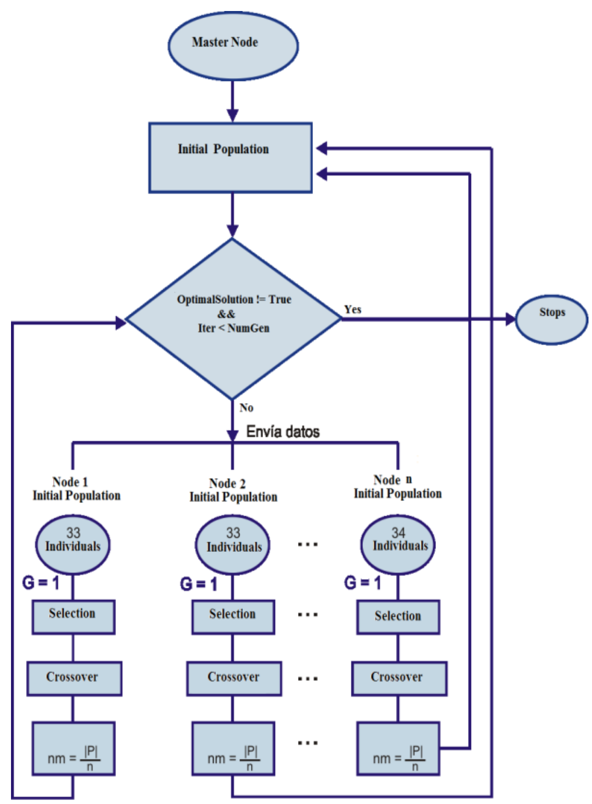

Fig 5. Functional Parallelization with Communication

The functional parallelization implements a constant communication among master and slaves. Also cooperation is needed to find best solutions iteratively. However, the principal overcoming of the functional implementation can be seen when a node fails or stops. The master stands waiting for the results. Master cannot continue with the next iteration of the program. Some additional mechanism to deal with nodes fails have to be implemented but it is beyond of the reach of this paper. The principal advantage of simple parallelization is the absence of dependences among processors. Each processor can work independent. Each node can find optimal solution. A node fails does not affect the other nodes because they work independent. Converting a sequential algorithm into a parallel algorithm requires an exhaustive knowledge of sequential algorithm. Additionally, it is important to assure consistently high level of performance when transferring data among processors.

6. Conclusions

Reducing execution time consuming is one of the advantages of parallel computing. Parallel computing is acknowledged as essential tool to deal with complex problems of combinatory optimization, in which the resources of only one computer to solve a NP-Complete problem would not be enough or would spend years searching for optimal solution.

Principal problematic of parallelizing a sequential algorithm was related with the dynamic data structures. It can be concluded that approach of parallelizing, present vantages and disadvantages, but according to the problem structure one of them can be chosen. Proposed methodology can help to convert a sequential evolutionary algorithm into a parallel evolutionary algorithm. For this work it was carried out a conversion of sequential algorithm into parallel algorithm as applied to water distribution network. A future work to give continuity to this research would be showing experimental results with the methodology as applied to water distribution network design.

7. References

1. Savic, D. A., and G. A. Walters (1997), Genetic algorithms for least-cost design of water distribution networks, J. Water Resour. Plann. Manage., 123(2), 67–76
2. Blazewicz, J., Handbook on Parallel and Distributed Processing, In International Handbooks on Information Systems, Springer, 2000.
3. Iglesias P.L., Martínez F.J., Fuertes V.S., Pérez García R., Algoritmo genético modificado para diseño de redes de abastecimiento de agua, IV SEREA - Seminario Hispano Brasileño sobre Sistemas de Abastecimiento Urbano de Água, Brasil, 10 de Noviembre 2004.
4. Liberatore S., Sechi G.M., Zuddas P., Water Distribution Systems Optimization by Metaheuristic Approach, Advances in Water Supply Management, 265-272, Balkema publisher, Lisse, The Netherlands, 2003
5. Gupta, I., Bassin, J.K., Gupta A., Khanna P., Optimization of Water Distribution System, Environmental Software 8, 1993, pp. 101-113.
6. Bhave, P.R.(1991) Analysis of Flow in Water Distribution Networks, Ed. Technomic Publishing Company, Lancaster, Pennsylvania (EEUU).
7. Goulter, I. C., B. M. Lussier, and D. R. Morgan (1986), Implications of head loss path choice in the optimization of water distribution networks, Water Resources Research, 22(5), 819– 822.
8. Cruz-Chávez M. A., Empirical Transformation of Job Shop Scheduling Problem to the Hydraulic Networks Problem in a Water Distribution System, Congreso de Electrónica, Robótica y Mecánica Automotriz 2009.
9. Holland, J. H.(1975), Adaptation in natural and artificial systems. MIT Press, Cambridge, Mass.
10. Goldberg, D.E.(1989). Genetic algorithms in search, optimization and machine learning. Addison –Wesley Publishing Co., Reading, Mass.
11. Alperovits, E., and U. Shamir (1977), Design of optimal water distribution systems, Resources Research, 13(6), 885– 900.
12. Quindry, G. E., E. D. Brill, J. C. Liebman, and A. R. Robinson (1979), Comment on ‘‘Design of optimal water distribution system’’ by E. Alperovits and U. Shamir, Water Resources Research, 15(6), 1651–1654
13. Fujiwara, O., and D. B. Khang (1991), Correction to ‘‘A two-phase decomposition method for optimal design of looped water distribution networks,’’Water Resources Research, 27(5), 985– 986.
14. Kessler, A., and U. Shamir (1989), Analysis of the linear programming gradient method for optimal design of water supply networks, Water Resources Research, 25(7), 1469–1480.
15. Eiger, G., U. Shamir, and A. Ben-Tal (1994),” Optimal design of water distribution networks”, Water Resour. Res., 30(9), 2637– 2646.
16. Loganathan, G. V., H. D. Sherali, and M. P. Shah (1990), A two-phase network design heuristic for minimum cost water distribution system under a reliability constraint, Eng. Optim., 15(4), 311– 336.
17. Baños Navarro R., Meta heurísticas Híbridas para Optimización Mono-objetivo y Multi-objetivo, Tesis Doctoral, Almería, España, Diciembre 2006. Disponible en: www.ace.ual.es/~rbanos/CV.html, última fecha de acceso 12 Mayo de 2009.
18. Cantu-Paz, E., A Survey of Parallel Genetic Algorithms, Technical Report IlliGAL 97003, University of Illinois at Urbana-Champaign, 1997.
19. Michel, M. and Devaney, J. E. 2000. A Generalized Approach for Transferring Data-Types with Arbitrary Communication Libraries. In Proceedings of the Seventh international Conference on Parallel and Distributed Systems: Workshops (July 04 - 07, 2000). ICPADS. IEEE Computer Society, Washington, DC, 1983.