Available online at www.sciencedirect.com
ScienceDirect
Procedia Engineering 89 (2014) 488 – 494
Procedia Engineering
www.elsevier.com/locate/procedia

16th Conference on Water Distribution System Analysis, WDSA 2014

Cloud-Based Decision Making in Water Distribution Systems

I. Montalvo Arango[a,*], J.S. Izquierdo [b], E.O.G. Campbell [b], R. Pérez-García[b]

[a]3S Consult GmbH, Albtalstraße 13, 76137, Karlsruhe, Germany
[b]Group Fluing-IMM, Polytechnic University of Valencia, Spain

Abstract

The evolution of technologies of informatics and telecommunications has improved significantly the way water distribution systems can be managed. The use of quasi real-time information is becoming a desired standard in many water utilities. Nevertheless, tools for water distribution system analysis and for supporting decision making haven´t been integrated with the same speed. This research uses cloud computing capabilities for supporting analysis and decision making in water distribution systems. It includes the development of evolutionary algorithms ready to run in a cloud environment for searching the best possible decisions considering a set of objectives. The evaluation of solutions profits from the scalability of the cloud for making faster the solution search process.

Peer-review under responsibility of the Organizing Committee of WDSA 2014

Keywords: cloud computing; water distribution system analysis; decision making; multi-objective optimization

1. Introduction

Cloud computing has been gaining ground during the last years in different applications areas. It is hard to find a short and single definition of cloud computing because it involves several interrelated parts. Typical definitions consider it as a model to make servers, applications and other IT resources available over the internet [4]. For many users it is just practically related to a "place" thanks to the phrase "in the cloud". Nevertheless, information have been saved in external servers and remote services have been also ran "long" before start talking about cloud computing. More than a place or resources over the internet cloud computing is a new approach to conceive distributed

* Corresponding author.
E-mail address: imontalvo@ingeniousware.net

Peer-review under responsibility of the Organizing Committee of WDSA 2014
doi:10.1016/j.proeng.2014.11.241

applications, both from software and hardware perspectives, based on accessibility, scalability, elasticity and reliability. There are five essential characteristics that can summarize what cloud computing is [5]:

- On-demand self-service: A consumer can unilaterally provision computing capabilities without requiring human interaction.
- Broad network access: Capabilities are available over the network and can be used by heterogeneous client platforms.
- Resource pooling: The provider´s computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.
- Rapid elasticity: Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand.
- Measured device: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g. storage, processing, bandwidth, and etc.)

A generalized adoption of cloud based solutions in the water industry has still a long way to go. Nevertheless, several interesting applications has been already started and published [2, 3]. Clouds bring out a wide range of benefits including configurable computing resources, economic savings and service flexibility. However, security and privacy concerns are shown to be the primary obstacles to a wide adoption [6]. Water companies can select the approaches that better suit their requirements among four different deployment models [5]:

- Private cloud: The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers.
- Community cloud: Cloud infrastructure provisioned for exclusive use by a specific community of consumers from organizations that have to share concerns.
- Public Cloud: Infrastructure provisioned for open use by the general public.
- Hybrid cloud: Composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability.

The hybrid cloud approach seems to have several advantages with respect to the other approaches for the case of the water industry because of the flexibility it provides to keep information inside the proprietary IT infrastructure of companies while also using some public infrastructure. As service models there are three options available [5]:

- Software as a Service (SaaS): Use of the provider´s applications running on a cloud infrastructure.
- Platform as a Service (PaaS): Capability provided to the consumer to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services and tools supported by the provider.
- Infrastructure as a Service (IaaS): The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and runs arbitrary software, which can include operating systems and applications.

This work will be focused on a platform as a service solution intended to be run in Windows Azure, the cloud solution of Microsoft.

2. Cloud-based water distribution system analysis

During the research that has been supporting this paper it was found a reference to a parallel version of EPANET [7] named PEPANET [8]. This new version was intended to be run not in the cloud but in the grid, it is specifically based on Message Passing Interface (MPI). There are some differences compared to cloud technology that could be seen perfectly in [9]. A PhD thesis was also found related to high performance hydraulic simulations [10], in this case, a software solution named IRMA was deployed on PACA Grid; a Cloud Infrastructure maintained by the French region “Provence-Alpes-Côte d’Azur”. A preference for cloud computing in order to provide a flexible and portable infrastructure for end-users is also mentioned in [11] when referring to the use of parallel frameworks for supporting sensor placement in water distribution systems.

Writing cloud-based software solutions for supporting decision making in water distribution system is a challenging task. For desktop applications it is very extended the use of the EPANET toolkit [7] for running hydraulic simulations. A singular difficulty for a cloud-based solution arises because of the limitation to use the well-known EPANET toolkit [7] when a good level of scalability is desired. During the development of this work a new version of the EPANET toolkit was writing in C#. This new version was named WaterCT and is stored in a private repository at http://bitbucket.org. Researchers interested in testing the code can contact the authors. As a first step, the structure of the existent EPANET toolkit was tried to be maintained. Later, the code was refactored to make it closer to the object oriented programming philosophy. This way several instances of the analysis engine could be created on demand for satisfying scalability requirements.

When running algorithms for decision making, several researchers have been dealing with evolutionary algorithms. In this context a relative high number of evaluations of an objective function aimed to evaluate decisions are required. These evaluations normally needs to run at least one simulation scenario of the network under decision. The amount of computational effort is significantly increased when long period simulations are required and many simulation scenarios need to be evaluated. Evaluating the reliability of the network after the analysis of consequences of single pipe breaks is a typical example. In this context, the availability of a scalable cloud-based solution for running network simulations is without doubts a great advantage for reducing computation time. Applications for decision making can also be extended to an online context where the integration of quasi real time measurements and the needs of fast calculation are strongly required.

3. Distributed evolutionary algorithms

Recently there have been several efforts to exploit the capacity resources available in the internet for Evolutionary Computation [12, 13, 14]. During this research a new version of Agent Swarm Optimization (ASO) [15] was designed to be run in Windows Azure. ASO is an agent-based framework that combines the strengths of multiple evolutionary algorithms and other techniques to solve optimization problems using a multi-objective approach. The main characteristics of ASO that were also used in the cloud-based version can be summarized as follows:

- New leadership concept different than the classic leadership in Particle Swarm Optimization: Swarms select as leader the closest particle to the so-called utopia point in the objective space. The utopia point is defined as the point in the objective space whose components give the best values for every objective. As this point is not known a priori, it is used a dynamic approximation, termed singular point, which is updated with the best values found so far during the evolution of the algorithm
- Normalization: Each objective may be expressed in different units and it is necessary to make some regularization for evaluating distances in the objective space. Once a regularization mechanism has been enforced, to establish the distance between any two objective vectors the Euclidean distance between them is calculated. The worst and best objective values are not usually known a priori; they are updated while the solution space is explored.
- Pareto Front enrichment: It is not easy to find a general heuristic rule for deciding which part of the Pareto front should be more closely represented and how much detail the representation of the Pareto front should contain. Those decisions are strongly dependent on the people solvin the problem and on the problem itself. In ASO users has the possibility to select the zones of the Pareto front where a higher level of details is desired.
- Dynamic population: With a fix population number it would be impossible to properly described a Pareto front with any desired level of detail. That’s why swarms are designed to clone agents when the density of points in the Pareto front should be increased. In case multiple solutions get dominated during the search process, swarms can also reduce its population dynamically.

- Human computer interaction: Specialists are able to interact with algorithms during runtime for both adding new swarms in zones of interest and proposing new potential solutions to existing swarms. The participation of several human agents with different perspectives on a problem is very close to what happens in the practice of engineering decision making, where politicians, economists, engineers, and other actors are involved in final decisions. The idea of incorporating user experience into the search process is a step forward in the development of computer-aided design.
- Swarm hierarchies: A hierarchical organization of swarms help to solve the task of evaluating efficiently if a new solution belongs to a Pareto front or not. Swarms first check if the solution in question belongs to its own Pareto subset. If the solution is not dominated then the swarm asks asynchronously is superior to check if the solution is dominated or not. The process is repeated at every hierarchical level if the solution is found to be non-dominated; in that case all swarms involved in the checking process will have the information of the new non-dominated solution.

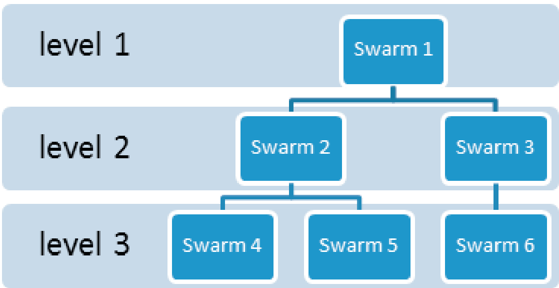

Fig. 1. Hierarchy of swarms.

- Rule-driven agents: New agents can be endowed with specific, problem-dependent behavioral rules purposely designed to heuristically approach the solution process to the problem in hand. These rules try to mimic the judgment of a human expert when approaching the solution of the problem in hand. These rules can be defined a priori and can also be the result of applying data mining and rule generation techniques to visited regions of the solution space.

The main difference among this version and the previous version of ASO is the cloud-based scalability capacity. This capacity gives a high flexibility to users for deciding how many agents are preferred to be running when finding solutions for a problem. This capacity is virtually only limited by the economy.

4. Using the Windows Azure Infrastructure

Windows Azure is a broad set of technologies for supporting cloud computing. Basically it is software running on hardware located in Microsoft data centers around the world. Services provided by Windows Azure include several ways to run applications, technologies for storing, managing and analyzing data, ways to connect applications, among others. This work will make references just to three of these important aspects in Windows Azure used to develop our software solution: The execution models, the data management and the messaging.

4.1. Execution models

The execution modes of cloud solutions mentioned in the first section of this paper have also their own expression in Windows Azure. The approach commonly called infrastructure as a service is represented by Virtual Machines, which are on demand and are paid by hour. In cases when all that is needed is a simple website, an option named precisely Website is designed to do the work. For the purpose of these research neither the mentioned Virtual Machines nor the Website option were used. Instead, it was used something called cloud services. It is the original technology for running apps in Windows Azure and it represents what is called Platform-As-A-Service (P-A-A-S). Cloud services are focused on applications and not on the platform they run on. No virtual machine needs to be explicitly created or managed; they are automatically created and managed by Windows Azure. Under Cloud Services two main type of Virtual Machines are known: Web Roles and Worker Roles. Windows Azure is told how many of those Virtual Machines are required. It starts them, it maintains them, it manages them; that's a platform as a service.

The Cloud Service developed in this work uses three Virtual Machines interconnected among them:

- A Web Role: It is used as the interface to interact with users. Allows defining the problem to be solved. Include solutions from users to be evaluated. Create swarms (population based algorithms) for solving a problem. Allow the selection of Pareto front zones under interest.
- Worker Role (Agent Swarms): Explore and exploit the solution space. Put solutions on the Queue to be evaluated. Create the Pareto Front. Learn from evaluation feedback.
- Worker Role (Solutions Evaluator): Evaluate solutions Send evaluations result back to the swarm to improve its behavior

4.2. Data management

Three main options are available in the Azure platform for data management: SQL Database, Table Storage and Blob Storage. The software implementation accompanying this work has only used Table Storage and Blob Storage. SQL Database is great for relational data but when possible and when a good scalability is also desired, the use of Table Storage is preferred instead. Table storage is simpler, is faster and is cheaper. The other option used, Blob Storage, is intended to store binary large objects. In this work it was mainly used for backing up data.

4.3. Messaging

Virtual machines are not alone living in isolation. They need to communicate each other. Inside one single virtual machine, several parts of an application could be running and also needing to get communicated to coordinate their actions. A typical example of that communication is the collaboration among swarms to determine the solutions belonging to the Pareto front when solving multi-objective optimization problems as mentioned in section 3 of this document. Windows Azure provides two technologies for supporting this kind of communications: Azure Queues and Service Bus. Service Bus is a little more complex and it can be used not only for making possible the communication among work roles but also for connecting applications running in the cloud with applications running in on premise servers. Queues are simpler and it was the option preferred in the context of this work. Basically it was used to allow the communication between the web roles and the worker roles. The web role handled the interaction with users, once a problem was defined; users were able to start running algorithms for finding solutions. The web role communicates the problem to be solved to the corresponding work role in charge of the swarm execution. It also communicates which algorithms should be executed inside the swarms in order to solve the problem. Once the different agents of active swarms start executing, they will need to evaluate the solutions they are exploring. The evaluation of solutions happens in another work role; it implies a communication from the work role of swarms to the work role of evaluating solutions. After solutions are evaluated, a feedback of that evaluation should be given back to the agents in order to improve and continue their search. In this case, a communication happens from the work roles evaluating solutions back to the work roles running swarms.

4.4. Tests

Note that when running applications either work roles or web roles can be added if needed. The idea behind the application developed is to provide the scalability of the cloud whenever needed. The limitation in this case is practically determined by the amount of money available to pay for the running infrastructure. For testing purposes, just three virtual machines were used for running two decision making examples: the small case of the well-known Hanoi network and the design of the water network of San José de las Lajas, a small town in Cuba.

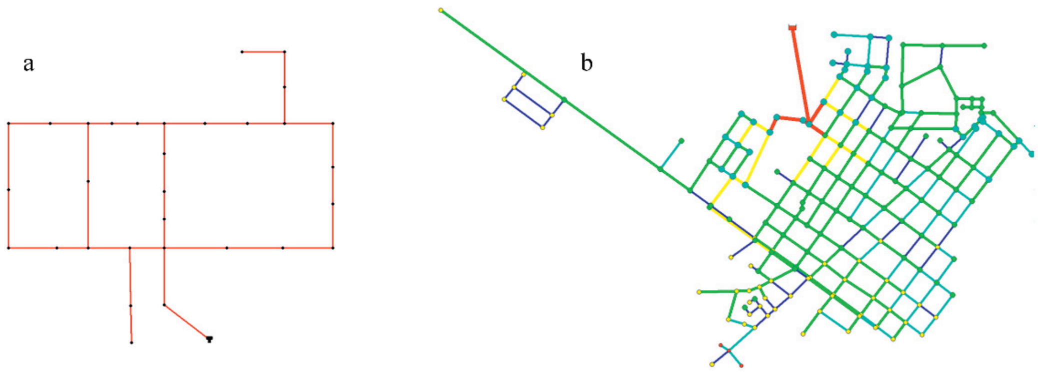

Fig. 2. (a) Hanoi Network; (b) San José Network.

Test executions provided expected results both from the perspective of the hydraulic simulation of solutions and the perspective of the functionality of Agent Swarm Optimization. The evaluation of the efficiency gained because of the scalability will be done in a posterior phase of these research considering long period simulations in more complex problems. Tests on this phase of the research were only intended for a basic proof of concepts.

5. Conclusions

In this paper we have presented an introductory approach of using cloud computing for supporting decision making processes in water distribution system. Presented ideas were based on the use of Windows azure and other Microsoft technologies. One of the main contributions during the research was the development of a version of the EPANET toolkit totally rewritten in C#, using dynamic memory and the object oriented programming paradigm. The code was written with the intention of being easily scalable in Windows Azure. Nevertheless, it could be potentially used on desktop applications if desired. Additionally, a new version of Agent Swarm Optimization was written to be run on the cloud, taking advantages of the Cloud Services that could be provided from Windows Azure. The combination of both cloud-based water distribution analysis engine and multi-objective optimization platform, open a new path for developing applications for supporting decision making in the water industry. Twenty years ago it was great to run single simulations on a desktop computer to know how a water network will work under some specific scenarios. Today's challenges requires the analysis of a high number of scenarios, sometimes integrated with quasi real time measurements and with a strong expectation to get results very quickly in a quasi-online context. Contaminant source identification, online calibration of models and anomalies and event detection in water distribution system are some of the applications that can take great advantage of the use of cloud based solutions for decision making. The adoption of these technologies is still slow in the water industry but it is a question of time and a question of realizing the great things that could be done under a well-controlled security risk level.

Acknowledgements

The project is supported by the German Federal Ministry of Education and Research (BMBF; project: 13N12180) and by the French Agence Nationale de la Recherche (ANR; project: ANR-11-SECU-006).

References

[1] G. von Laszewski, L. Wang, F.Wang, G.C. Fox, G.K. Mahinthakumar, Threat Detection in an Urban Water Distirbution systems with Simulations conducted in grids and Clouds, in Proc. The Second International Conference on Parallel, Distirbuted,Grid and cloud Computing for Engineering, Ajaccio, Corsica, France, 2011.

[2] L. Wang, D. Chen, W. Liu, Y. Ma, Y. Wu, Z. Deng, Parallel Simulation of Threat Management for Urban Water Distribution Systems with MapReduce in Clouds, Computing in Science and Engineering, 30 July 2012, http://doi.ieeecomputersociety.org/10.1109/MCSE.2012.89, IEEE computer Society Digital Library. IEEE Computer Society,

[3] Z.Y. Wu and M. Khaliefa, Cloud Computing for High Performance Optimization of Water Distribution Systems, in Proc. World Environmental and Water Resources Congress, Albuquerque, New Mexico, 2012, pp. 679-686.

[4] R. B. Hurwitz, M. Kaufman, F, Halper, Cloud Computing for Dummies, 2009.

[5] P. Mell, T., Grance, The NIST Definition of Cloud Computing, National Institute of Standards and Technology, U.S. Department of Commerce, Special Publication 800-145, 2011.

[6] I.M. Khalil, A. Khreishah, M. Azeem, Cloud Computing Security: A Survey, Computers 3 (2014) 1-35.

[7] L.A. Rossman, EPANET 2 User`s Manual, Cincinati (IN), USA, Environmental Protection Agency, 2000.

[8] S. Sreepathi, K. Mahinthakumar, E. Zechman, R. Ranjithan, D. Brill, X. Ma, G. Von Laszewski, Cyberinfraestructure for Contamination Source Characterization in Water Distribution Systems, in Proc. Computational Science – ICCS 2007, 7th International Conference Beijing, China, 2007.

[9] S. M. Hashemi, A. K. Bardsiri, Cloud Computing Vs. Grid Computing, ARPN J. of Systems and Software, 2 (5) 2012.

[10] G.P.Pezzi, High Performance Hydraulic Simulations on the Grid using Java and ProActive, PhD thesis directed by Denis Caromel, University of Nice-Sophiea Antipolis, 2011.

[11] W. E. Hart and R. Murray, Review of Sensor Placement Strategies for Contamination Warning Systems in Drinking Water Distribution Systems, J. of Water Resources Planning and Management, 136(6) (2010) 611-619.

[12] M.-S. Jian, T.-Y. Chou, A real-world-like evolutionary algorithm on the cloud-computing environment, in: Proc. 14th International Conference on Advanced Communication Technology (ICACT), 2012, pp 881-886.

[13] P Fazenda, J. McDermott, U.-M. O'Reilly, A library to run evolutionary algorithms in the cloud using MapReduce, Applications of Evolutionary Computation, Lecture Notes in Computer Science , 7248 (2012) 416-425.

[14] W.-P. Lee, Y.-T. Hsiao, W.-C. Hwang, Designing a parallel evolutionary algorithm for inferring gene networks on the cloud computing environment, BMC Systems Biology 8(5) (2014), DOI: 10.1186/1752-0509-8-5

[15] I. Montalvo, J. Izquierdo, J. Jerrera, R. Pérez-García, Water supply system computer-aided design by Agent Swarm Optimization. Computer-Aided Civil and Infrastructure Engineering, 29(6) (2014) 433-448