Commentary
Volume 13:6, 2022

Global Journal of Technology and Optimization

ISSN: 2229-8711 Open Access

Distributed Memory Computing's Potential and Limitations

Maurice Herlihy*

Department of Electrical and Computer Engineering, Carnegie Mellon University, Forbes Ave, Pittsburgh, Pennsylvania, USA

Description

Anything that can be calculated by a Turing machine is considered to be computable in sequential systems. Computability concerns take on a new hue in distributed systems when computations necessitate cooperation amongst numerous parties. There are numerous issues that are not solvable in this situation as well, but these limitations on computability are more related to the difficulty of making judgments in the face of uncertainty and less to the innate computational ability of individual individuals [1]. Each participant could construct the entire system state and the computation could proceed sequentially if the participants could reliably and instantly interact with one another. However, in any realistic model of distributed computing, each member initially only has a partial understanding of the system's overall state and the uncertainties brought on by errors and arbitrary time constraints [2,3].

The communication model, timing model, and failure model of a distributed system all play important roles in determining what can and cannot be calculated in that system. This paper examines significant computability results and describes the central role that topology plays in the distributed computability theory using the typical distributed system paradigm, where processes execute asynchronously, communicate by reading and writing shared memory, and fail by crashing. The paper also takes into account various additional presumptions that can be used to get around conclusions that are impossible. Failure detectors and antagonists are names for these presumptions. The article concludes by demonstrating a potent simulation technique that demonstrates how wait-freedom and t-resilience are identical from a computability standpoint. The algorithms produce a synthetic view when combined. Numerous of the examined tasks, including consensus, set agreement, approximate agreement, and loop agreement, can be characterised merely in terms of sets of potential input and output values, without needing to explain which procedure can be used to assign or select which value. A process can use another process's input or output value in a colourless job without going against the task's specifications. The renaming task is one instance of a task that is not colourless. Processes in this task begin with unique names derived from a big name space, and they must end with names derived from a smaller name space. In this case, a process cannot use the output name of another since the output names would not be distinctive [4,5].

For parallel computing, a heterogeneous network of computers can be utilised. In order to speed up the resolution of a single problem, the network is used as a parallel computer system. In this instance, the user offers a specialised parallel application created to effectively tackle the issue on the diverse computer network. The primary objective of the kind of utilisation is high performance. Like with conventional use, the user supplies both the application's code and the input data. The source code is typically transferred to the computers, where it is locally compiled, when each computer in the network has a distinct architecture. All the libraries required to create local executables should be available on every computer. Distributed computing may also make advantage of a heterogeneous computer network. When using parallel computing, the user's computer or any other networked single computer can be used to run the application. The only justification for using many computers is to speed up the application's execution. In contrast to parallel computing, distributed computing addresses circumstances in which a programme cannot be executed on the user's computer simply because all of the application's components are not present on this computer. One such circumstance is when a user cannot give all of the application's code because it is only available on distant computers. Due to the required resources, the user's machine might not be able to execute such a code component [3, 4].

The third group of topics is technical safeguards, which secure the whole information system of a health organization's network. The majority of security breaches occur via electronic media, through the usage of computers and other portable electronic devices, hence this subject is crucial to guaranteeing the organization's security. This topic includes security procedures for scanning for viruses, using firewalls and encryption, and authenticating information. Lemke came to the conclusion that encryption and firewalls were the most often used security measures. Antivirus software, chief information security officers, and cloud computing are a few other notable security measures that are also in use, though their implementation is budget-dependent. Technical safeguards, which secure a health organization's network's entire information system, make up the third category of subjects. This topic is essential to ensuring the security of the company since the majority of security breaches happen through electronic media, through the use of computers and other portable electronic devices. This subject covers security practises such as virus scanning, employing firewalls and encryption, and information authentication. Encryption and firewalls were found to be the most often deployed security methods, according to Lemke. Other significant security methods include cloud computing, chief information security officers, and antivirus software, albeit their deployment is budget-dependent [1, 2].

***Address for Correspondence:** Maurice Herlihy, Department of Electrical and Computer Engineering, Carnegie Mellon University, Forbes Ave, Pittsburgh, Pennsylvania, USA; E-mail: herlihy.maurice@up.ac.za*

Date of Submission: 09 June, 2022, Manuscript No. GJTO-22-71123; **Editor Assigned:** 11 June, 2022, PreQC No. P-71123; **Reviewed:** 16 June, 2022, QC No. Q-71123; **Revised:** 21 June, 2022, Manuscript No. R-71123; **Published:** 26 June, 2022, DOI: 10.37421/2229-8711.2022.13.300

Acknowledgement

None.

Conflict of Interest

The authors reported no potential conflict of interest.

References

1. Herlihy, Maurice, Sergio Rajsbaum and Michel Raynal. "Power and limits of distributed computing shared memory models." *Theoretical Com Sci* 509 (2013): 3-24.
2. Upadhyaya, Sujatha R. "Parallel approaches to machine learning-A comprehensive survey." *J Parallel Dist Com* 73 (2013): 284-292.
3. Guzek, Mateusz, Johnatan E. Pecero, Bernabé Dorronsoro and Pascal Bouvry. "Multi-objective evolutionary algorithms for energy-aware scheduling on distributed computing systems." *App Soft Com* 24 (2014): 432-446.
4. Keshta, Ismail and Ammar Odeh. "Security and privacy of electronic health records: Concerns and challenges." *Egypt Info J* 22 (2021): 177-183.

5. Chauhan, Avinash. "Parallel and distributed computing-A review." *Int J Dist Com Tech* 2 (2016): 35-46.

How to cite this article: Herlihy, Maurice. "Distributed Memory Computing's Potential and Limitations." Glob J Tech Optim 13 (2022): 300.