# AN APPROACH PRIORITIZING THE CAUSAL FACTORS OF LARGE SCALED DATA USING SOFT COMPUTING: A CASE STUDY

Jyoti Prakash MISHRA [1], Zdzislaw POLKOWSKI [2], Sambit Kumar MISHRA [3]

[1] Gandhi Institute for Education and Technology, Banatangi, Bhubaneswar, affiliated to Biju Patnaik University of Technology, Rourkela, Odisha, India, E-mail: jpmishra@gietbbsr.com
[2] WSG University, Bydgoszcz, Poland, E-mail: zdzislaw.polkowski@byd.pl
[3] Gandhi Institute for Education and Technology, Banatangi, Bhubaneswar, affiliated to Biju Patnaik University of Technology, Rourkela, Odisha, India, E-mail: sambitmishra@gietbbsr.com

**Abstract** *In general situation, the high intensive tasks linked to computation can be provisioned either through dedicated servers or can be properly filtered in virtual platforms. The major constraint in such situation can be associated with obtaining decision in process initiated as well as in the cost of data transmission preserving security. Sometimes some specific issues are required to be resolved during utilization of Internet of Things in specified applications expecting feasible solutions. Often it has been observed that the traditional computing mechanisms linked with the devices like routers equipped with specific infrastructures as well as services may not be adequate for implementation due to lack of flexibilities. In such situation, it may be difficult for data acquisition and processing. In fact, this complexity can be due to constrain in operations linked to computational resources especially in distributed environments. Sometimes also it is required to focus on specific data retrieved from different IoT distributed components linked to virtual machines. Accordingly, the techniques should be enabled on proper accumulation of data with accurate prediction prioritizing the causal factors and data sharing mechanisms. Though it is equally important to handle large scaled data related to issues of multi domain applications, it is essential to enhance the modularity, flexibility as well as scalability of the data and to maintain the optimal accuracy. Also to address these issues, specific computational approaches especially ant colony optimization technique can be the support to make commonalities and obtain close association of the resources with the relevant data. The implementation mechanism in virtual machines also supports integration of complex data and provisions privacy with security.*

***Key words****:* Distributed resources, Virtualization, Scalability, Query term, Pheromone.

**JEL Classification Codes: C8**

## INTRODUCTION

Of course there are ample of mechanisms to obtain the computing resources getting closer to the data resources. Sometimes applying the fog computing techniques or edge computing mechanisms, the enhancement can be possible with minimal cost. Also the implementation in the virtual platform integrates the complex coordinated applications to obtain optimality while deploying distributed resources irrespective to the stability as well as security. In fact, it has been observed that the solution obtained implementing the relevant mechanisms integrating specific cloud services can be more provisioned as compared to the simulations within other distributed computational methods. Indeed, the large scaled data being provisioned with the Internet of Things can be effectively analyzed especially with desired applied techniques and also sometimes processing all these data may not be easy or sufficient as it is required with security with low latency. In such situation, it is essential to adopt intelligent solutions to support the mechanism and help in this process of obtaining suitable architecture to satisfy all the requirements prioritized by the complex Internet of Things applications. These applications can be provisioned with the interrelated tags with radio-frequency identification, sensors, and actuators having the commonalities towards achieving the goal. Based on some specific aspects linked to signified applications, sensors as well as actuators may be involved in obtaining solutions focusing the real time data. In such cases, the services can be more latency sensitive and can tend to optimality based on the generated data. The mechanisms linked to virtual machines as well as data centers can be provisioned with data accumulated from the deployed sensors with specific protection to the latency sensitivity of the optimality. Somehow the resources linked to virtual machines may be in a position to manage complex tasks to avoid the inconsistencies while prioritizing the data from IoT applications. While adopting specific applications on IoT, some constraints like specific bandwidth, range specification with intermittent connectivity can be also be considered. In fact, bandwidth can be measured based on the amount of data to be transmitted per unit of time and of course it can be affected through many factors like accumulation and transmission of voluminous data linked to each device and deployment status of the devices. Accordingly, the size of data provisioning with the suitable protocol in the form of packets should be compared for similarity to observe authenticity. Of course symmetrical transmission cannot be observed always during transmission of data. But based on layered approach especially implementing the internet of things mechanisms, it can be possible to link the physical devices with suitable connectivity. The reliability as well as timely transmission of data is more important. Based on edge computation, the main objective can be to emphasize on size and structure of data with its further linkages. Finally, the data should be properly handled as well as processed based on layered approach.

## LITERATURE SURVEY

X. Lu et al.[1] in their work focused on the communication mechanisms within the machines and observed that to some extent the remote procedure calls based on Java sockets may face constraints linked to big data applications. Of course they made the provision towards enabling the optimal performance.

M. Isard et al.[2] during their research could able to retrieve the vital information associated with large scaled data particular implementing specific data processing models and focusing on open source implementation. In fact, it was intended towards simplicity, transparent fault tolerance as well as scalability.

J. Lofstead et al.[3] in their work focused on HPC systems, as large scaled data applications could be linked concurrently with the similar sharable storage systems. As such, the interference of I/O system might be the constraint on the performance of the applications. Also the applications on these data might face high latencies while performing I/O due to the necessary data transfers between the parallel file system and computation nodes.

M. Zaharia et al[4] in their work have prioritized on resilient distributed datasets distributed across the cluster. In fact it has a provision to permit the users to apply coarse-grained transformations on these input partitions. Somehow these transformations may be linked with MapReduce-like operations. Considering the specific applications on distributed storage system, to some portions it permits to cache the operations in the memory and iterate quite efficiently.

S. Venkataraman et al.[5] during their research focused on optimization of large scaled data towards sustainment of high data locality and to minimize the I/O latency with contribution on application performance. Somehow, the similar mechanism may not be suitable for HPC systems.

C. S. Kruse et al.[6] in their work tried to accumulate the data and to process implementing the machine learning concepts. The mechanisms have been implemented to reveal hidden patterns linked to data as well as to track users. It was intended to achieve reliable and accurate results for general IoT applications.

M. Chiang et al.[7] in their research focused on fog computing applicable on local area networks and probably linked to large size servers and sensor devices. In fact these computational architectures could be employed to achieve services with greater availability, lower latency, as well as location awareness

M. Hartmann et al.[8] in their work focused on the smart healthcare domains incurring high cost for data transmission that covers a limited area associated with mobile edge computing architectures during transformations. In fact, they discussed on the trends in the advancement of edge IoT-based smart healthcare frameworks, including systems that employ edge computing for functional processes.

M. S. Hossain et al.[9] in their work tried to combine the machine and deep learning mechanisms with IoT architecture to boost big data processing ability. In fact, it is more powerful for managing these categorized data.

M. S. Hossain et al.[10] in their work focused on the issues linked to large scaled data. The mechanisms linked to IoT systems can implement blockchain to maintain data privacy. Also it can be helpful in deployment of critical services in IoT applications.

## IMPLEMENTATION USING ANT COLONY OPTIMIZATION TECHNIQUES

It is true that the ant colony optimization technique is well applied to large scale data and simultaneous it is also sequenced to prioritize the time as well as space complexity. In fact for instance, the search initiation linked to localized area can be a great support to enhance its performance, even if ignoring the heuristic information linked to the pheromone values. As it is inspired by the behavior of ants in general, it usually enables to traverse the optimal or near optimal solution exploiting the pheromones associated with other ants. Generally, the ants can be used to focus on various positions to obtain the desired solutions. In this respect, each ant can proceed to next move as per the schedule and updated pheromone values. This can be a continual process till obtaining the near optimal or optimal solution.

## IMPLEMENTATION WITH PSEUDO-CODES USING ANT COLONY OPTIMIZATION TECHNIQUES

```
        while (i!=null)
    for i = 1 to VM( VM= number of virtual machines)
    while (PEi(Processing elements)!=VM)
      select PEi for ith  move;
     end while
   globally update pheromone levels;
   end for
    end while
```

In this application it is intended to prioritize the ant colony optimization technique to focus on the linear space complexity of pheromones and implementation of pheromone information in local search path.

**Algorithm**

Step 1: Initialize the pheromone level

Step 2: for j = 1 to n ( iteration up to maximum)

Step 3: for i = 1 to $PE_i$

Step 4: for t = 1 to max_ant

Step 5: Perform selection of $ant_t$ to succeeding edge

Step 6: Update $ant_t$ with corresponding pheromone value and traverse to next move

Step 7: if i = n

improve $ant_t$ opting local search technique

end if

end for

end for

Step 8: Update gbest value pheromone level of corresponding ant

Step 9: Globally update pheromone values implementing gbest tour

Step 10: Update corresponding candidate solutions

end for

## EXPERIMENTAL ANALYSIS

In general, the ant colony mechanisms are quite easy towards implementation in a wide range but the performance is required to be monitored consistently as the large scaled data are involved. The algorithm implemented in this application initiates the search mechanism and responsible for minimizing the complexities in the pheromone levels. The maximum size of processing elements in this situation is with 55 and the optimum size of virtual machines in this aspect is clubbed with 60. As shown in figure-1, the process of execution of queries is of course dependent on the size of the processing elements. Accordingly, the size of the processing elements is directly proportional to the query execution time. Also as reflected in figure-2, the query response time is also quite dependent on the deployed virtual machines.

**Table 1. Processing elements with estimated query execution time**

| Sl.No. | No. of Processing elements | Query execution time(m.sec.) |
|---|---|---|
| 1 | 25 | 0.13 |
| 2 | 35 | 0.29 |
| 3 | 45 | 0.34 |
| 4 | 55 | 0.47 |

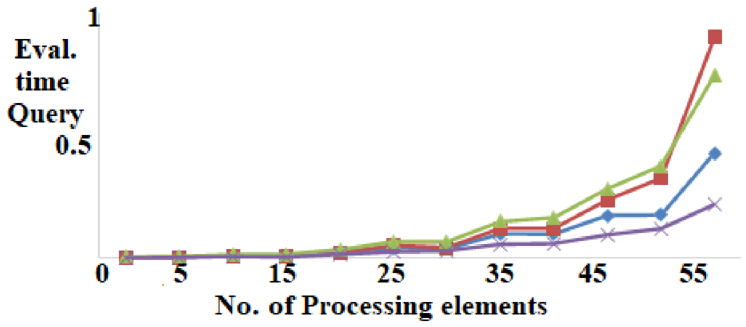


**Figure 1. Processing elements with estimated query execution time**

**Table 2. Deployed virtual machines with estimated time based on query response**

| Sl. No. | Deployed virtual machines | Estimated time based on query response(m.sec.) |
|---|---|---|
| 1 | 11 | 0.16 |
| 2 | 22 | 0.19 |
| 3 | 33 | 0.24 |
| 4 | 44 | 0.37 |

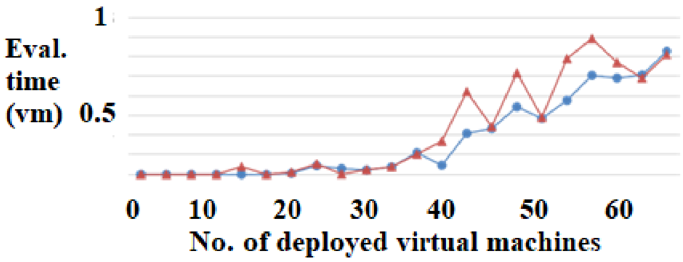


**Figure 2. Deployed virtual machines with estimated time based on query response**

## DISCUSSION AND FUTURE DIRECTION

During implementation using MATLAB-13, it is observed that total accumulation cost is imposed on virtual machines towards scheduling of tasks. The scheduling of tasks can be measured through the provisioned schedulers. Of course for each scheduler, the processing as well as data transfer mechanism linked to the virtual machines may not be unique. Accordingly, prioritizing the processing elements, the efficiency of the virtual machines can be enhanced. In this application, the firefly technique is applied to focus on the response time and CPU time of each and every task during the deployment of virtual machines. As soon as the tasks are scheduled with the processing elements, these can be again needed to be reallocated based on the response on the virtual machines.

## CONCLUSION

It is obvious that the scheduling of tasks during execution are uncommon associated with computational resources provisioned with virtual machines. Also it has been observed the enhancement of response time while association with large scaled application of data. Though it is not so easy to compute the performance, therefore prioritization is needed on each and every processing element during their dependencies. In this work, it is more focused on the scheduling mechanisms implementing the ant colony optimization algorithm. In fact, it has been prioritized on the efficiency estimated in terms of cost analysis of assigned tasks and performance of virtual machines. Also it is seen that application on large scaled data is confined to the response along with the task execution time.

## REFERENCES


1. X. Lu, N. S. Islam, M. Wasi-Ur-Rahman, J. Jose, H. Subramoni, H. Wang, and D. K. Panda, "*High-performance design of Hadoop RPC with RDMA over InfiniBand*", in International Conference on Parallel Processing, IEEE, 2013, pp. 641–650.
2. M. Isard, M. Budiu, Y. Yu, A. Birrell, and D. Fetterly, "*Dryad: distributed data-parallel programs from sequential building blocks*", in Special Interest Group on Operating Systems Review, ACM, vol. 41, 2007, pp. 59–72.
3. J. Lofstead, F. Zheng, Q. Liu, S. Klasky, R. Oldfield, T. Kordenbrock, K. Schwan, and M. Wolf, "*Managing variability in the I/O performance of petascale storage systems*", in International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE, 2010, pp. 1–12.
4. M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica, "*Resilient distributed datasets: a fault-tolerant abstraction for in-memory cluster computing*", in International Conference on Networked Systems Design and Implementation, USENIX, 2012, pp. 15–28.
5. S. Venkataraman, A. Panda, G. Ananthanarayanan, M. J. Franklin, and I. Stoica, "*The power of choice in data-aware cluster scheduling*", in International Conference on Operating Systems Design and Implementation, USENIX Association, 2014, pp. 301–316.
6. C. S. Kruse, R. Goswamy, Y. Raval, and S. Marawi, ''*Challenges and opportunities of big data in health care: A systematic review*'' JMIR Med. Informat., vol. 4, no. 4, p. 38, 2016.
7. M. Chiang and T. Zhang, ''Fog and IoT: An overview of research opportunities,'' IEEE Internet Things J., vol. 3, no. 6, pp. 854–864, Dec. 2016.
8. M. Hartmann, U. S. Hashmi, and A. Imran, ''*Edge computing in smart health care systems: Review, challenges, and research directions*'' Trans. Emerg. Telecommun Technol., early access, Aug. 2019, Art. no. e3710, doi: 10.1002/ett.3710.
9. M. S. Hossain, S. U. Amin, M. Alsulaiman, and G. Muhammad, ''*Applying deep learning for epilepsy seizure detection and brain mapping visualization*'' ACM Trans. Multimedia Comput., Commun., Appl., vol. 15, no. 1s, pp. 1–17, Feb. 2019.
10. M. S. Hossain and G. Muhammad, ''*Emotion-aware connected healthcare big data towards 5G*'' IEEE Internet Things J., vol. 5, no. 4, pp. 2399–2406, Aug. 2018.