Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 115 (2017) 846–852

Procedia Computer Science

www.elsevier.com/locate/procedia

7th International Conference on Advances in Computing & Communications, ICACC-2017, 22-24 August 2017, Cochin, India

Task Based Resource Scheduling in IoT Environment for Disaster Management

J Sathish Kumar[1,*], Mukesh A Zaveri[2], Meghavi Choksi[3]

[1,2,3] *Computer Engineering Department, Sardar Vallabhbhai National Institute of Technology, Surat-395007, India*

Abstract

Poor resource management may lead to loss of many lives and affects the economy. During the disasters, communication networks are destroyed and re-establishment of dynamic network is a challenging issue. However, Internet of Things is an advanced technology which enables communication anytime, anywhere and anyplace. In such situations, IoT may provide available information of resources that can be scheduled for different tasks. In this paper, we address resource scheduling issue and exploited bankers algorithm ensuring the optimum utilization of resources. Further, we evaluated our approach with execution time and fairness of resource allocation for utility.

Peer-review under responsibility of the scientific committee of the 7th International Conference on Advances in Computing & Communications.

Keywords: Resource Scheduling; Disaster Management; Internet of Things; Graph Theory;

1. Introduction

Natural disasters such as floods, tsunamis and earthquakes, strikes the different parts of the world atleast every year with a great loss of humanity. During 2015, the number of disasters were reported by United Nations Office for Disaster Risk Reduction (UNISDR) is 346. This lead to death of 22,773 lives, 98.6 million people were affected and US$66.5 billions of economic damage [1]. Exclusively in India, from 2011-2015, 38 million people were affected with different types of disasters and 29 million dollars of economic damage occurred [2]. One of the major reasons for such a great loss is unavailability of real time network communication, lack of task management and resource management. To minimize the impact of such disasters, technological support is a way out for addressing these problems. Internet of Things (IoT) is such an influential upcoming technology that functions based on Internet [3, 4, 6]. Hence, IoT can help in a great extent in establishment of dynamic IP enabled network.

* Corresponding author. Tel.: +91-7041668913
E-mail address: ds14co001@coed.svnit.ac.in

1877-0509
Peer-review under responsibility of the scientific committee of the 7th International Conference on Advances in Computing & Communications
10.1016/j.procs.2017.09.167

In the available literature of disaster management [1, 2, 5, 9], majorly the challenges are divided the four phases. Namely, the mitigation [15] phase and preparedness phase are during the period of before the disaster occurs. In the mitigation phase various issues like public education, critical infrastructure protection and infrastructure improvement and information campaigns are addressed. Volunteer management, community preparedness, emergency response plan and material management are addressed in the preparedness phase. Likewise the response phase has to be dealt immediately after the short duration of the disaster. Issues like resource scheduling, resource allocation, victim management and plan implementation, situation awareness, call_take_dispatch and etc are addressed in this phase. Whereas, the recovery phase may extent to long duration of time after the disaster. In this phase, issues like damage assessment, procurement, public information and insurance claim are considered. The glance of the four phases is as shown in the Fig 1.

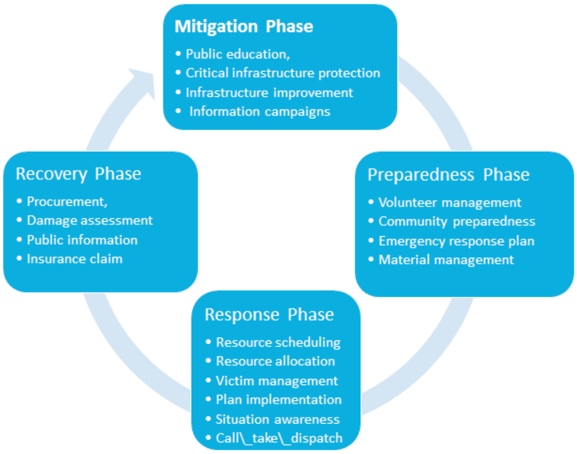

Fig. 1. Various Phases of Disaster Management

Resource scheduling in the response phase is very critical after post disaster management that need to address the tasks on urgent basis. Resource scheduling can be executed in administration level and on field level. However, on field execution of resource scheduling is crucial that directly involves in rescuing, recovering and saving people lives. To handle this critical situation for real time response, various tasks should be handled properly. Different tasks require various resources. These resources should be scheduled in such a way that all tasks can be accomplished in a particular time frame so that loss can be minimized. Therefore, by considering these circumstances, we formulate the problem and address the critical response for resource scheduling by exploiting the bankers algorithm [12, 13].

Hence, the data acquired for the various tasks to be performed at different locations are to be clustered. These places are identified using location estimation algorithms in IoT that is proposed in [11]. Further, the task management can be area wise or region wise. Hence, using clustering approach for efficient communications and connectivity of the devices in IoT network, which is proposed in [8, 10]. Assuming that dynamic network has been established after the disasters, real time communications in the network can be dealt efficiently. However, scheduling the available resources for different tasks for disaster management has been identified as an non-trivial problem which we are addressing in this paper. Also, we carried out the experimental simulation results with execution time, stability in the allocation and fairness for the utilization of resources.

Rest of the paper is presented as follows. The problem description and algorithm are detailed in section 2. Resource scheduling algorithm and corresponding complexity analysis are described in section 3. In section 4, simulation results are presented and conclusion in section 5.

2. Problem Description

In this section, the problem description is described with the help of graph theory which is detailed as follows. Let us assume that we have t tasks and r resources. The tasks can be represented as t which can further defined as a set of sub tasks, say $t_1, t_2...t_i..t_m$. Likewise, the resources are represented as r, that can further divided into set of resources defined as $r_1, r_2...r_j..r_n$ with k similar type of instances. Now, the scheduling is represented as a directed graph $G = (V, E)$, where V is a set of vertices that indicates the tasks and resources, $E \subseteq \{\{t, r\} | t, r \in V, t \neq r\}$ defines the potential allocation edges. For example, let us assume that 3 tasks has to be addressed and 4 resources are available with multiple instances as shown in the Fig 2. Resource r_1 has 1 instance, r_2 has 2 instances, r_3 has 1 instance and r_4 has 3 instances.

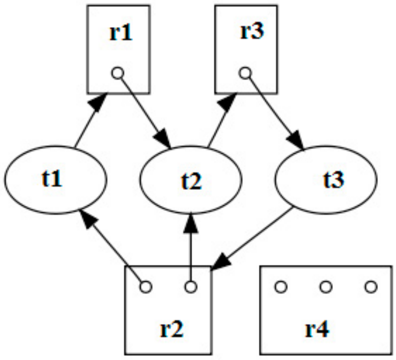

Fig. 2. Example graph representing tasks and resources

In the context of disaster, the tasks can be determined as establishing communication network, providing medical treatment to the people, rescue and recovery operations and many more. Suppose, if these tasks should be addressed in all the places where disaster takes place, resources like fire engines, volunteers, military force, ambulance and medical help are required. However, the resources allocated by considering many factors such as the distance between the disaster place and resources, traffic considerations, road maintenance etc. Now, Fig 2 depicts the demand of the same resources for different tasks i.e., t_1, t_2 demands r_1 but r_1 is already allotted to t_2. At the same time, t_2, t_3 demands r_3 but r_3 is already allotted to t_3. Hence, these kind of improper scheduling may leads to loss of many lives instead of saving them. Therefore, to address such stability in the allocation of scheduling the tasks and resources we explored the available literature and utilized the bankers algorithm for efficient resource scheduling [12, 13]. This approach verifies the safe allocation before scheduling the resources for tasks and brings the stability in the network. The algorithm and corresponding details are described in the next section.

3. Proposed Resource Scheduling Algorithm

Often disaster tasks may occur in a time overlapped way. This scenario leads to requests of all the resources simultaneously by all the tasks in the respective locations. However, if demand of the resources are more than the requested and without proper scheduling of the resources to the tasks certainly results in chaotic situations. Therefore, efficient resource scheduling with safe sequence by addressing various tasks is critical and an important problem. In this regard, for the proper scheduling of the tasks with resources we propose the resource scheduling approach based on bankers algorithm. Certain assumptions have been taken into consideration before the description of the algorithm that are detailed as follows.

The bankers algorithm is a resource scheduling and deadlock avoidance algorithm used in operating systems. It works in such a way that by verifying for safety in the scheduling of all the resources by predetermined fashion. Then, for possible tasks the algorithm makes a verification in many states before making a decision whether allocation should be allowed to continue or not. So, for bankers algorithm works by having the knowledge of each resource demanded by each tasks, each resource is currently allocated to each tasks and each resource in the network currently available for further allocation [12, 13].

For modeling resource scheduling strategy in disaster management, we define four matrices. These matrices are related to demand, allocation, availability and requested resources to accomplish a particular task. if Demand[i,j] = k, then task t_i may request at most k instances of resource type r_j. If Allocation[i,j] = k, then task t_i is currently allocated k instances of resource type r_j. If Available[j] = k, then there are k instances of resource r_j. If Requested[i,j] = k, then t_i may need k more instances of resource type r_j to complete the task. The approach should identify the requirements of the tasks and verify if the requested resources are available or not as given in the equation 1.

$$\sum_{i=1}^{m}\sum_{j=1}^{n} Requested_{ij} \leq \sum_{j=1}^{k} Available_j \tag{1}$$

where $Requested_{ij}$ is the i^{th} task requesting j^{th} resource out of m tasks, n resources and $Available_j$ is the available resources of n with k types. Further, the number of resources instances contributed by each resource type must satisfy the criteria as shown in the equation 2.

$$\sum_{i=1}^{n}\sum_{j=1}^{k} r_{ij} \leq \sum_{i=1}^{m}\sum_{j=1}^{n} Demand_{ij} \tag{2}$$

where r_{ij} indicates the n number of resources with k types and $Demand_{ij}$ indicates the m tasks with n resources. With the above assumptions and constrains, the bankers proposed algorithm for the resource scheduling of various tasks can be addressed by the following algorithm.

Algorithm 1: Resource Scheduling Algorithm

```
Input: t tasks and r resources with k instances ;
Output: Scheduling of resources with respective tasks ;
/* Currently j available resources */ ;
Available [j] ;
/* Task t_i is currently allocated k instances of resource type r_j */ ;
Allocation [i, j] ;
/* Task t_i may request at most k instances of resource type r_j */ ;
Demand [i j] ;
while ∀i ∈ t_i do
    if Requested[ij] ≥ Demand[ij] then
        Go for next task, since task t_i has exceed the maximum claim
    end
    else if Requested[ij] ≥ Available[j] then
        Go for next task, since task has to wait because unavailability of resources
    end
    /* Verifying the availability for the requested task i */ ;
    Available [j] = Available [j] - Requested [i j] ;
    /* Allocating the requesting j resources for task i */ ;
    Allocation [i j] = Allocation [i j] + Requested [i j] ;
    /* Decreasing the demand after allocating the requesting resources*/ ;
    Demand [i j] = Demand [i j] - Requested [i j] ;
    /* Adding them to pool for scheduling of remaining resources */ ;
    Available [j] = Available [j] + Allocated [i j]
end
return schedule and execute the tasks t_i
```

It is important to analyze the proposed algorithm in terms of computational complexity for critical time analysis and response. The proposed algorithm time complexity is $O(m^2n)$ with atmost m possible tasks and n resources. However, the brute force algorithm takes $O(m!/(m-n)!)$ since it goes through each enumeration to verify whether the allocation is safe or not. For m tasks and n resources, the number of enumerations is ${}^{m}P_n$ which is equal to $(m!/(m-n)!)$. Hence, the proposed algorithm using bankers strategy gives better results than brute force approach which is FCFS (First come First Serve).

4. Simulation Results

Without proper mapping to real time environment of identifying the tasks and resources, it is difficult to validate our approach. Therefore, we made efforts to represent our approach in Google maps [16]. Hence, the proposed approach is shown in real time allocation using google maps in the city of Surat, Gujarat as shown in Fig 3. To verify, we considered 3 types of resources say r_1, r_2, r_3 with 10, 5, 7 instances respectively. Also, we considered 5 tasks to be carried out, say t_1, t_2, t_3 t_4 and t_5. Let us say the available resources with each task, say t_1 has (0 1 0), t_2 has (2 0 0), t_3 has (3 0 2), t_4 has (2 1 1) and t_5 has (0 0 2). The demand of each task can be given as, say t_1 demands (7 5 3), t_2 for (3 2 2), t_3 for (9 0 2), t_4 for (2 2 2) and t_5 for (4 3 3). Now, our scheduling algorithm results in executing of the tasks in sequence of (t_2, t_4, t_1, t_3 and t_5) respectively, for safe and stable allocation of resources in the network. Therefore, our approach assures that all the tasks are addressed with proper utilization of all resources.

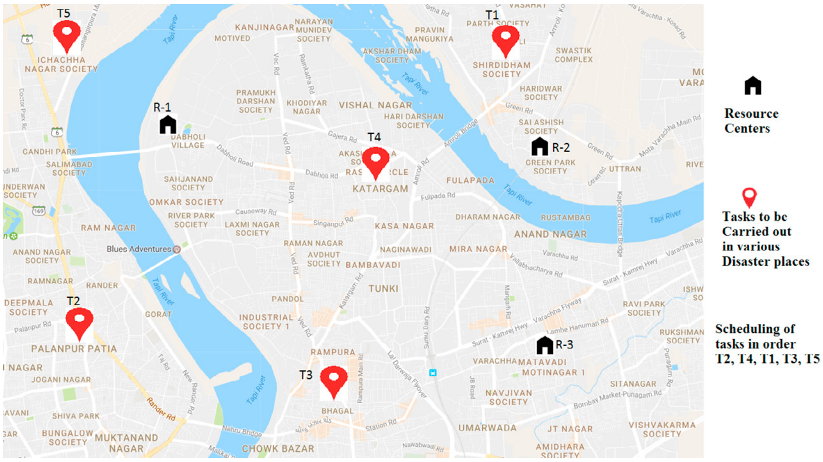

Fig. 3. Real time scheduling shown in Google Maps

Also, the simulation results for scheduling tasks and resources is evaluated in terms execution time and fairness in the allocation resources that determines the utilization. By First Come First Serve (FCFS) approach which works in the fashion of brute force and our proposed approach with respect to execution of time with different number of tasks are compared and is shown in Fig 4. The proposed approach out performs the FCFS in terms of bringing stability and safe in the scheduling of resources and tasks. Both the approaches were evaluated with 3 resources of (10,5,7) and (5,5,5) instances. Although both approaches have almost same execution time but as number of tasks increases the our proposed approach execution time is giving better results. The number of tasks is considered up to 10, as an example to demonstrate our approach is working well but it can extended with higher number of tasks. Hence, our proposed approach results in better scheduling that leads to better stability.

The proposed approach is devised in such a way that complete utilization of the resources are carried out. For each task the resources were allocated with complete fairness. Jain et al. [14], proposed to measure the fairness in terms of quantity which is given in the following equation 3.

$$f(X) = \left[\sum_{i=1}^{n} x_i\right]^2 / n \times \sum_{i=1}^{n} x_i^2 \qquad (3)$$

where $0 \leq f(X) \leq 1$ is fairness measure of resource allocation and $X = (x_1, x_2, ..., x_n)$ implies the allocated resources, n is the number of resources and activities and x_i is the amount of resource allocated to individuals $i = 1, 2, ..., n$. A large value of $f(X)$ represents fairer resource allocation from the system perspective. The corresponding results is shown in Fig 5. Also, by deduction we can say greater the value of fairness implies better the stability, that is fairness is directly proportion to stability.

Hence, when compared with FCFS and our proposed approach, the scheduling and allocation of the resources in FCFS is not good for the tasks from 5 onwards. But, whereas in our approach the allocation is stable even in the

Fig. 4. Comparative analysis of task execution time

number tasks are increasing which is shown in Fig 5. Since, FCFS approach couldn't able to perform the allocation under the same environment where our proposed approach is carried out, we couldn't able to compare the fairness allocation for the larger inputs. Because, IoT assumes huge number of devices are going to take part, it reasons out that the proposed approach of resource allocation performs well. Also, it is well suited for the applications of disaster management where the rescue, recovery operations are critical.

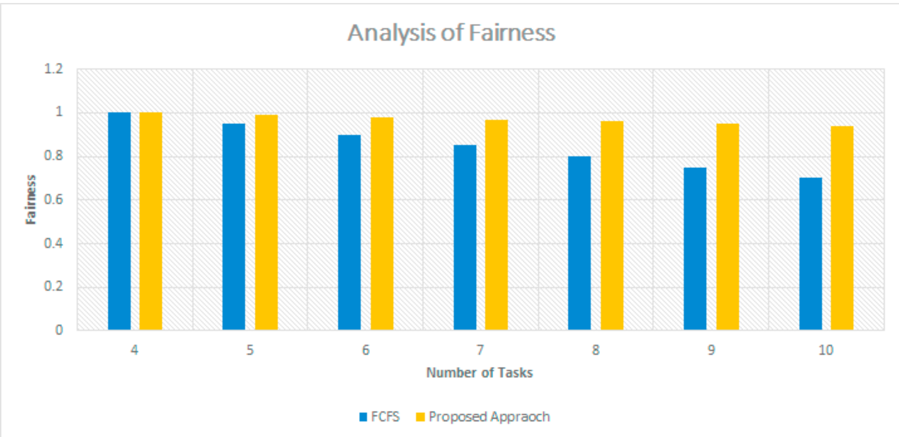

Fig. 5. Comparison of Fairness analysis in allocation

5. Conclusion

Resource scheduling and task management is critical during disaster scenarios. The role of IoT in the establishment of communication network in such cases helps in efficient mapping of resources to the network entities. Also, having the knowledge of the availability of resources and tasks assist to determine the scheduling and allocation efficiently using our proposed approach in order to bring the stability in the network. The proposed approach is evaluated in terms of fairness and execution time, which shows better results than FCFS brute force approach.

Acknowledgments

This work is supported by the Ministry of Electronics and Information Technology (MeitY), funded by Ministry of Human Resource Development (MHRD), Government of India (Grant No. 13(4)/2016-CC&BT).

References

[1] Wahlstrom, M and Guha-Sapir, D. "The Human Cost of Weather-Related Disasters 1995-2015.", Geneva, Switzerland: UNISDR. *Available online at https://www.unisdr.org/2015/climatechange/COP21_WeatherDisastersReport_2015_FINAL.pdf*, 2015.

[2] Data collection Survey For Disaster Prevention In India. *Available online at: http://open_jicareport.jica.go.jp/pdf/12245155.pdf* 2015.

[3] Lee, Gyu Myoung and Crespi, Noel and Choi, Jun Kyun and Boussard, Matthieu. "Internet of Things, Evolution of Telecommunication Services." *Springer* 2013; 257–282.

[4] Gubbi, Jayavardhana and Buyya, Rajkumar and Marusic, Slaven and Palaniswami, Marimuthu. "Internet of Things (IoT): A vision, architectural elements, and future directions." *Journal of Future Generation Computer Systems* 2013; **29** (7): 1645–1660.

[5] Muaafa, Mohammed and Concho, Ana Lisbeth and Ramirez-Marquez, Jose. "Emergency Resource Allocation for Disaster Response: An Evolutionary Approach." 2014.

[6] Yang, Lili and Yang, Shuang-Hua and Plotnick, L. "How the internet of things technology enhances emergency response operations." *Technological Forecasting and Social Change* 2013; **80** (9): 1854–1867.

[7] Kondaveti, Russell and Ganz, Aura. "Decision support system for resource allocation in disaster management." in Proceedings of Annual International Conference of Engineering in Medicine and Biology Society (EMBC), *IEEE* 2009; 3425–3428.

[8] Kumar, J Sathish and Zaveri, Mukesh A. "Clustering for collaborative processing in IoT network."in Proceedings of the Second International Conference on IoT in Urban Space, *ACM* 2016; 95–97.

[9] Pearce, Laurie. "Disaster management and community planning, and public participation: how to achieve sustainable hazard mitigation." *Natural hazards* 2003; **28**: 211–228.

[10] Kumar, J Sathish and Zaveri, Mukesh A. "Hierarchical Clustering for Dynamic and Heterogeneous Internet of Things." *Procedia Computer Science* 2016; **93**: 276–282.

[11] Pandey, Saurabh K and Zaveri, Mukesh A. "Localization for collaborative processing in the internet of things framework." in Proceedings of the Second International Conference on IoT in Urban Space, *ACM* 2016; 108–110.

[12] Lawley, Mark and Reveliotis, Spyros and Ferreira, Placid. "The application and evaluation of banker's algorithm for deadlock-free buffer space allocation in flexible manufacturing systems." *International Journal of Flexible Manufacturing Systems* 1998; **10** (1): 73–100.

[13] Madduri, Hari and Finkel, Raphael. "Extension of the Banker's algorithm for resource allocation in a distributed operating system.", *Information Processing Letters* 1984; **19** (1): 1–8.

[14] R. Jain, D. Chiu, and W. Hawe. "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems." *Digital Equipment Corporation, Technical Report DEC-TR-301* 1984, **38**.

[15] Arora, Hina and Raghu, TS and Vinze, Ajay. "Resource allocation for demand surge mitigation during disaster response." *Decision Support Systems* 2010; **50**: 304–315.

[16] Svennerberg, Gabriel. "Beginning Google Maps API 3." *Apress* 2010.