Contemporary Engineering Sciences, Vol. 7, 2014, no. 28, 1579 - 1588
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ces.2014.411214

Applying Volunteer MapReduce Platform to Assist Search and Rescue Operations

Bolatzhan Kumalakov, Darkhan Akhmed-Zaki

Faculty of Mechanics and Mathematics
al-Farabi Kazakh National University
al-Farabi ave. 71
Alamaty, Republic of Kazakhstan

Grzegorz Dobrowolski

Department of Computer Science
AGH University of Science and Technology
A. Mickiewicza 30
Cracow, Poland

Copyright © 2014 Bolatzhan Kumalakov, Darkhan Akhmed-Zaki and Grzegorz Dobrowolski. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Presented case study is based on application of the previously designed volunteer peer-to-peer MapReduce computing platform to assist evacuation of people who are disoriented in a forest that is affected by wildfire. Problem solution requires extensive use of features which are built into mobile devices in such a way, that computing not only become execution device, but also interface and data collector sensors. Nonetheless, despite being advanced to such a level every device in the infrastructure resumes its voluntarily mapper or reducer node features.

Mathematics Subject Classification: 68U01

Keywords: multi-agent systems, search and rescue operations, clustering

1 Introduction

In [1] we present a Peer-to-Peer MapReduce (P2P-MapReduce) platform that employs wide range of devices (from industrial server machines to handheld gadgets) in order to organize and execute volunteer high performance computing applications. It has already proven to effectively solve large numeric [2] problems, but in this piece of work we explore how features that are built into standard mobile devices (such as smartphones and tablet computers) could be exploited extensively in order to solve complex problems. Unlike in previous work in this case mobile devices do not just serve as computing nodes, but become active data collectors and user interfaces while maintaining mapper and reduccer responsibilities.

Presented case study is based on bulding an application that supports search and rescue (SnR) team operations in the forest affected by wildfire. In particular software solution should compute optimal evacuation points; help people (who have to be evacuated) to navigate to the appropriate evacuation point and meet the SnR team there. It must take into account emotional and physical condition of the people when suggesting evacuation points and SnR evacuation plan. In order to do so platform makes use of agent intelligence and negotiation protocols for planning.

2 Materials and Methods

SnR team structure and action planning framework is based on descriptions from [3].

In order to carry out the case study we employ following research method or algorithm: first, we define the problem and the solution software requirements. Second, we present an overview of existing SnR software tools and position our development against them. Finally, the case proceeds to solution design, implementation and test results.

Main investigation materials are the computing infrastructure that includes HP ProLiant 6-th generation server that is equipped with 16 Gb RAM and 4 core Intel Xeon processor, 3 Android OS based smartphones. They are connected via computer network, partially wired, and partially wi-fi. Wi-fi is broadcasted using TP-LINK TL-WR841N router. Software environment includes Windows OS, Java development kit 7, and Jade (Java Agent Development Framework) for Server side and standard Android 4.2 installment on smartphones with Jade-Leap container. All test results are saved to the MS Excel file for visualization and further presentation.

2.1 Problem and Software Requirements Definition

Consider following scenario: several wildfire sources are developing in the national park due to hot and dry climatic conditions. When first responders arrive SnR team discovers that several groups of people are disoriented in the forest and are not capable of navigating through the area (Figure 1). In order to locate and evacuate these people from the site SnR team employs special Wi-Fi equipment to cover forest area with the wireless network and instructs people to connect their handhold devices to it using loudspeakers.

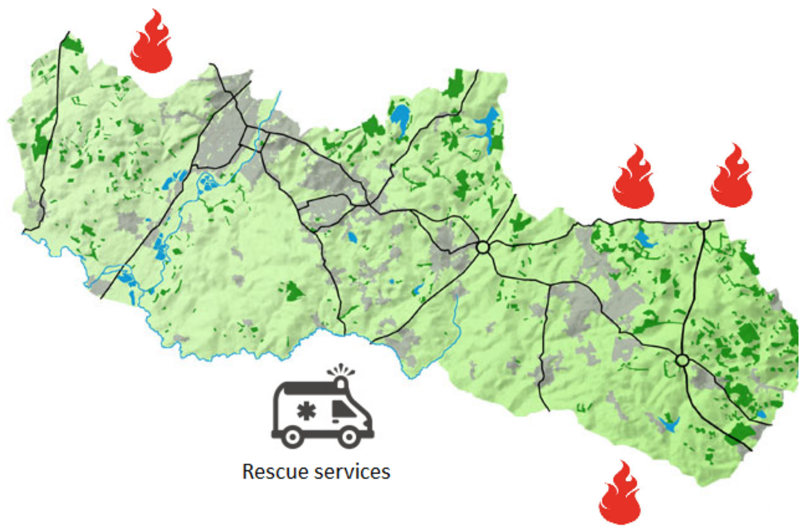

Figure 1: Figure visualizes a forest that is being affected by the wildfire. There is a group of first responders, among whom there is a Search and Rescue team that aims to locate and evacuate people threatened by hazardous conditions.

System task is to identify personal devices' GPS coordinates, assist rescue services by estimating optimal evacuation points and supplying map directions. In our initial work we suggest that device owners are grouped by geographical location and rescued by SnR team driving specialized vehicle. Nonetheless, in might be the case that some people are not able to proceed to the evacuation point because of injury or stressfulness of situation and resulting disorientation.

Algorithm 1 formalizes the high-level solution of the problem. Input data is stored as two layers that may be merged. Local map (layer one) here is supplied in a form of a graph, where roads and passes are represented by edges, and road/pedestrian crosses are represented by vertices. Second layer defines GPS coordinates of graph objects and is used to identify mobile device positioning in the area. Using such formalization we are able to define following tasks:

First, since pick-up path is the root that starts in one place, has to go through every evacuation point at least once, and return to the starting point we treat it as a traveling salesman problem and apply "branch and bound" algorithm to solve it. We do not claim that it is the best or in any way more preferable way of solving the problem in similar cases. However, it well suits defined case because there is no need for high precision of the result coordinates and the number of evacuation points is not likely to exceed 50.

```
Data: GPS coordinates of mobile devices and local map of the
      evacuation site
Result: Directions to evacuation points for mobile device users, and
        pick-up path for SnR vehicle
initialization;
read local map;
read mobile device coordinates;
define group centers;
while grouping condition is not met do
  | group mobile devices by geographical location;
end
define evacuation points and compute users' gathering pathes;
compute pick-up path for SnR vehicle;
```

Algorithm 1: Algorithm defines global system behavior given local area map and GPS coordinates of mobile devices.

Second, grouping mobile devices by location to identify optimal evacuation point is a clustering task. Nonetheless, it is not that straightforward. As defined earlier, some people might not be able to reach distance locations, thus, there i a need to extend clustering algorithm in such a way that it takes into account individual traveling capabilities. Algorithm that solves this problem is presented in subsection 2.3.

Taking into account the defined task SnR team should be equipped with the software tool that is capable of:

1. identifying handheld device location on a geographical map;

2. computing evacuation points and optimal pathes to them;

3. passing computed pathes to the mobile devices;

4. estimating shortest/fastest people evacuation root for SnR vehicles.

Let us also assume that SnR services are fully equipped with the necessary machinery (equipment) and our goal is to design the software solution that supports the required functionality.

2.2 Overview of the existing platforms

Most of the existing SnR platforms provide technology support to the rescue operation management, not the first responders. They include cascading web map services and Geographical Information Systems [4], and remote sensing [5] solutions. Both enable high level view and understanding of the disaster situation, region specifics regarding an incidents and available resources. In other words they support strategic decision making, realtime command and coordination of the available resources (such as machinery, people, etc.) at the top level.

Nonetheless, first responders are the people who carry out duties of on-site immediate help to victims and evacuation in case of an emergency. In [5] authors note that “the primary link in the chain of information exchanges that leads to making critical, perhaps lifesaving, decision.” However, there is a lack of technology (software) support for first responders. For instance, in [9] authors note that “during fire incidents, when the first responders arrive on site, they have very limited information about the building, occupants and/or the location of the hazard.” This lack of information could affect decision making and result in such problems as inadequate priority of SnR activities, inefficient positioning in the area SnR operation and others that might cause deaths and casualties.

In order to overcome these limitations number of authors propose their algorithms that may support SnR team operations. First, in [10] authors propose a protocol based on temporarily order routing algorithm, and apply emergency region concept in it. They argue it gives safe evacuation roots. [7] presents a piece of work that takes into account spread hazards (such as fire etc.) on static landscape, and base its work on the idea that people being evacuated should always be far away from the hazard. The LifeNet project [6] aims to crete an electronic “lifeline” instead of the physical lifeline that is used in SnR practice to guide fire-fighters in complex structural buildings. The electronic lifeline is designed as a wearable computing system and micro display to compute and display navigation guidance in the buildings.

In [8] authors combine a network of sensors with mobile robots and radio tags to provide an integrated view for situation awareness, guide fire-fighters to targets, and warn them of potential dangers. Although being an innovative and well designed concept, ir assumes that sensors and robots are deployed prior to organizing the network, thus, setting the system up may cause a delay in performing evacuation actions.

Unlike in presented solutions we propose to use general propose handheld devices (smartphones) of the people being evacuated. Moreover, our platform only requires installing one application, thus, takes considerably less time to setup and run the infrastructure. While this particular case study does not involve estimating spread of hazardous substances our platform has proven to support this functionality in our previous work.

2.3 Solution design

In order to identify optimal evacuation points we first group mobile device users and identify locations that SnR vehicle is able to reach. Second, we supply every device with the direction to its own gathering point. At this point following questions have to be addressed:

1. how many people are going to be evacuated from every single point? It helps to identify the capacity of the vehicle to be sent.
2. is everyone in shape to rich the gathering point? There might be groups of people, who are not able to walk required distances due to their emotional or health conditions.

Answers to these questions can only be obtained from mobile device owners. We propose that their answers are standardized and used by intellectual agents to negotiate gathering point.

Let us formalize the task as follows: $S = \{s_1, s_2...s_n\}$ is the set of all handheld devices controlled by intellectual agents. $R = \{r_1, r_2...r_m\}$ is the set of all possible evacuation points that can be reached by SnR vehicle, which performs evacuation. If selected as an evacuation point, every $r \in R$ gets an agent associated with it. At this point we assume that device owners are able to walk at least some distance. If not, they remain at the same location while evacuation actions is performed and SnR forces approach them individually.

$Z = \{z_1, z_2...z_t\}$ is a set of clusters' shadow centers that is used by an algorithm to group devices before assigning evacuation point r_i. If algorithm execution associates device s_i with evacuation cluster center z_j, we denote it as $s_i \rightarrow z_j$. It is worth pointing out that z_j is eventually set to point r_j where the people are going to be picked up. As a result, $s_i \rightarrow z_j$ equals $s_i \rightarrow r_j$, and this is important at a later stage.

We also define distance function ρ between two objects s_i and r_j as (1).

$$\rho(s_i, r_j) = \sqrt{(x_{s_i} - x_{r_j})^2 + (y_{s_i} - y_{r_j})^2} \quad (1)$$

Algorithm 2 presents solution design, which is in fact a modified verison of k-means with additional features that make it suitable for the case study. In particular it takes into account specifics of evacuation area, and secondly makes correction on objects' psychological and physical state (in this case every object represents mobile device owners).

Second *while loop* in algorithm 2 corrects evacuation point position with respect to people conditions. It is done through agent negotiation, where

Data: GPS coordinates of mobile devices, local map and coordinates of evacuation points
Result: Every device is associated with its evacuation point
initialization;
read local map and evacuation points R;
read coordinates of mobile devices S;
define class centers Z;
while *MaxIteration is not reached and* Z *change* **do**
    for every $s \in S$ compute
    $s \to Z_j$ where $\rho^*(s_i, Z_j) = min\{\rho(s_i, Z_1), \rho(s_i, Z_2), ...\rho(s_i, Z_n)\}$;
    update coordinates for all $z_k \in Z$
    $X_{z_k} = (x_{z_k} + x_{s_m})/2, Y_{z_k} = (y_{z_k} + y_{s_m})/2, Z_{z_k} = (z_{z_k} + z_{s_m})/2$
end
compute $\rho^*(Z_j, R_t)$ and assign $Z_j = R_t$;
while $\phi(s)$ *not null* **do**
    if $\omega(s) > 0$ **then**
        Z' = shift(Z);
    end
end

Algorithm 2: Modified version of the k-means algorithm that suits application specifics of the defined case study.

communication variables are: $\phi(s)$ is the shift request from mobile agent s, $\omega(s)$ decision to shoft towards particular agent that is defined as:

$$\omega(s) = \begin{cases} 1 & \text{shift half a distance towards agent, s if he is the only one} \\ 0.5 & \text{shift quarter a distance towards agent s, if his reason has highest value} \\ 0 & \text{otherwise} \end{cases}$$

All the values here are discrete and are computer strictly in accordance with the definition.

3 Results and Discussion

Designed solution was implemented using Jade platform and tested in two main dimensions: first, we tested application performance by comparing P2P-MapReduce implementation against sequential program; second, we measured how different would the results be with and without agent negotiation (i.e. if solution does not take into account conditions of the people being evacuated).

Figure 2 visualizes the data, obtained while testing sequential and P2P-MapReduce solutions. When these results are compared we can see that if the number of objects does not exceed 150, P2P-MapReduce platform shows poorer performance. On a larger number of objects it performance does exceed sequential implementation.

This is due to the expenses of network communication and agent negotiations. Nonetheless, with an increase in the number of objects this latency impact decreases.

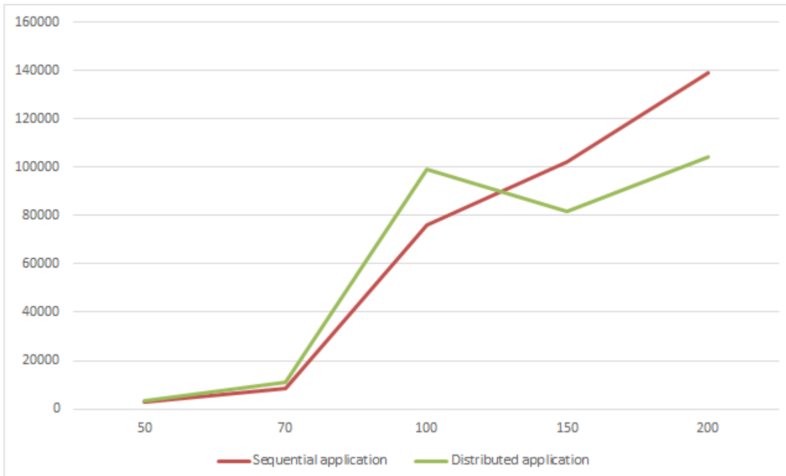

Figure 2: Figure presents testing results for sequential and P2P-MapReduce implementation of the program.

Second test case is based on measuring and comparing walking distance that physically and psychologically affected people had to walk before they get to evacuation point (figure 3). Compared solutions here are one with and the other without agent neotiation protocol.

From the figure we can see that affected people have to walk on average 26% distance less.

4 Conclusions

Paper presents case study where P2P-MapReduce platform is applied to solve SnR assistance problem. In order to do so it makes extensive use of mobile devices' build in functionality to provide user interface and collect input data. Core of the solution is the extended version of the k-means algorithm that employs agent negotiation protocol to adjust its results with respect to peoples' physical and emotional conditions.

As a result we have explored how mobile device features could be extensively used for computational proposes in P2P-MapReduce platform, collect and analyze empirical data on the newly designed platform performance, and solve complex case study using novel approach.

Figure 3: Figure presents testing results for solutions with and eithout agent negotioation protocol implemented.

Future research is going to concentrate on further complicating platform application scenarios both from technological and application environment stands.

Acknowledgements. Presented results were obtained under research grant from the Ministry of education and science of the Republic of Kazakhstan "Organizing distributed and cloud computations to solve resource-demanding tasks."

References

[1] D. Ahmed-Zaki, G. Dobrowolski and B. Kumalakov, Peer-to-peer mapreduce platform, *ICAART,* **2** (2013), 565 - 570.

[2] D. Ahmed-Zaki and B. Kumalakov, Solving complex iterative tasks using intellectual load distribution and mpi, *Bulletin of the National Engineering Academy of the Republic of Kazakhstan,* **2(52)** (2014), 25 - 31.

[3] G. Foo, *Emergency Response and Serach and Rescue. An Operational Book,* Dimersar, London, 2009.

[4] J. Hutton and A. Melihen, Emergency response remote sensing evolves in the wake of experience, *Photogrammetric engineering and remote sensing,* **72(9)** (2006), 977-981.

[5] M. Klann, Tactical navigation support for firefighters: The lifenet adhoc sensor-network and wearable system, M. Klann, editors, Mobile

Response, *Lecture Notes in Computer Science,* **5424** (2009), 41-56. http://dx.doi.org/10.1007/978-3-642-00440-7_5

[6] M. Vasardani and D. Flewelling, Cascading web map services and gis web servers for emergency response management, *Geomatica,* **59(4)** (2005), 435444.

[7] S.H. Yang and P. Frederick, Safetynet a wireless sensor network for fire protection and emergency responses, *Measurement and Control,* **39(7)** (2006), 218 219. http://dx.doi.org/10.1177/002029400603900704

[8] S. Sawyer, A. Tapia, L. Pesheck and J. Davenport, Mobility and the first responder, *Commun. ACM,* **47(3)** (2004), 62-65. http://dx.doi.org/10.1145/971617.971647

[9] V. Kumar, D. Rus and S. Singh, Robot and sensor networks for first responders, *IEEE Pervasive Computing,* **3(4)** (2004), 24-33. http://dx.doi.org/10.1109/mprv.2004.17

[10] Y. Tseng, M. Pan and Y. Tsai, Wireless sensor networks for emergency navigation, *IEEE Computer,* **39(7)** (2006), 55-62. http://dx.doi.org/10.1109/mc.2006.248

Received: November 11; Published: November 27, 2014