Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Epidemiology-based Task Assignment Algorithm
for Distributed Systems

Parth Brahmbhatt, Sergio G. Camorlinga

Abstract

Objective: Design task assignment algorithms based on the patterns of disease spread among the population.
Scope: Epidemiology studies spatiotemporal patterns of illness in populations and the factors affecting it. An epidemic emerges out of the population activities and environment. Task assignment is a common activity in many realms where sub-tasks are created, delegated, and collectively carried out to achieve the original task. Due to its complexity and context, task assignment can be a challenging activity that can result in limited outcomes. This research studies task assignment as an epidemic assigned to a distributed system. We have developed computational models to understand the outbreak of aerosol-borne diseases by using the agent-based modelling approach. Experiments are carried out to observe the patterns of emergence during the spread of disease among the individuals and get insights of their mechanisms. These mechanisms are used to design algorithms for task assignment on distributed systems.

Results: Understanding the emergent behaviour of diseases can provide the platform for the development of distributed algorithms that can be helpful in overcoming some of the challenges of task assignment in a distributed system.

Keywords: Agent Based Modelling; Distributed Algorithms; Task Assignment; Modelling; Computational Epidemiology; Epidemiology
1. Introduction

"An Epidemic is the occurrence, in a community or region, of cases of illness, specific health-related behaviour, or other health-related events clearly in excess of normal expectancy." Since the middle ages, epidemics has influenced significant historical events rather it be the plague in Roman times or the fall of Han Dynasty in the 3rd century in China. Moreover, the 1918 flu pandemic in U.S.A was responsible for more deaths than even the World War I itself. In last 50 years the world has seen the epidemics caused by HIV/ AIDS, SARS, influenza and many others. According to the World Health Organization website, every year around 15 million people die due to infectious diseases which eventually leads to epidemics. Since epidemics is a big challenge in front of human society, the interest in studying possible solutions to prevent or control any kind of epidemic outbreaks has been developed among the research community. Epidemiology is the formal branch of science that specially focuses on the study of emergence or spread of disease. Technically Epidemiology is "the study of the occurrence and distribution of health-related events, states, and processes in specified populations, including the study of the determinants influencing such processes, and the application of this knowledge to control relevant health problems." With the rise in globalization around the world, different factors contributing to the spread of disease or outbreaks have increased as never before. An outbreak of disease is not confined to just few regions or countries, today a virus can travel from one part of the world to other part within the matter of day. With the increase in disease factors the complexity in epidemiology have also increased. Epidemiologists and researchers have looked towards the field of Computational Epidemiology to counter this complexity. Traditionally the field of epidemiology has been widely limited to studying the past events, but with the help of Computational Epidemiology researchers have the power of predicting the future in unique ways. Computational epidemiology is an interdisciplinary field, which puts more emphasis on developing models and then using these models to understand the spatiotemporal patterns of illness among the population.

### Nomenclature

| Symbol | Description |
|--------|-------------|
| $c_j$ | Computation resource required by task “$j$” |
| $C_i$ | Computation resource free in processor “$i$” |
| $e_j$ | Execution time required for completion of task in ideal environment |
| $E_i$ | Execution time for completing the task “$t_j$” on processor “$p_i$” |
| HIST | Returns with the histogram of computation done. |
| ISO | Computers slips to isolation if value = 1 |
| KILL | Kill every package of the task by itself except the “HIST”. |
| $m_t$ | Memory required by task “$T$” |
| $M_p$ | Memory free in the processor “$P$” |
| $P$ | Processor |
| Ppltn | Initial population size. |
| $T$ | Task |

In this research, we have created a model for a severe outbreak of the Severe Acute Respiratory Syndrome (SARS) virus. On March 12, 2003 the World Health Organization (WHO) issued a global alert which described number of cases of a typical pneumonia in Vietnam, Hong Kong, China and Singapore. SARS is caused by the mutated virus of the Corona virus family. Due to its common cold like spreading capacity it created a situation of medical emergency over the world. The SARS virus was a big challenge for WHO to counter as at least 8096 people were affected in a short period of time out of which 21% were healthcare workers. We viewed the SARS virus as an emergent system property arising from the activities, interdependencies, relationships and internal/external environmental factors among others exiting in our society. Once, the emergence of SARS virus has been understood with computer models, the learning can be translated and applied to solve a number of problems in the field of distributed computing. Distributed systems is the collection of independent systems that appears as a single coherent system to its users. Since a distributed system has number of...
interconnected processors or nodes, task allocation is one interesting problem to solve. The problem is to find an effective and optimal way to allocate a given task among different processors to maximize the system reliability. Many researchers have taken a lot of interest in a path of solving this problem of task allocation. With this work, we have applied our learning from SARS as a complex adaptive system to design algorithms for task allocation in a distributed system. Other researchers have also taken an inspiration from nature to solve the problem of task allocation \(^8,^9\), but this work highly focuses on reduction of the overall cost of task allocation by understanding the inter-system behaviour and non-linearity of distributed system rather than just following conventional sequential approach.

1.1. Research Methodology

The data about the SARS outbreak (e.g. cases, death tolls, spread around globe) has been derived by triggering search queries on various online medical journals, Pubmed.gov, who.int \(^10\), cdc.gov \(^11\) etc. On completion of the data-collection phase, the analysis about factors causing the emergence of SARS virus in society has been carried out. These factors are then modelled using the agent-based modelling approach by using the NetLogo simulation software \(^14\). People are the agents in our epidemiology-based model and an artificial breed of population is randomly created. The person’s immunity, current health condition, and closeness with other agents in the environment have been modelled. This model uses the data and alerts which were issued by the WHO \(^10\) and the Centers of Disease Control and Prevention (CDC) \(^11\) during the SARS epidemic. Based on these data, several SARS stages have been inferred as shown in Table 1\(^10-13\).

| Stage       | Definition                              | Interpretation in our Model               |
|-------------|-----------------------------------------|------------------------------------------|
| Susceptible | People who have higher probability of getting affected | People with lower immunity or in very close contact with SARS patients |
| Latent      | People who have caught the virus but no symptoms have yet appeared. | First 7 days of disease |
| Infectious  | People who have been showing symptoms. | From 7 – 14 days of disease |
| Hospitalized| People who have been admitted to hospitals | After 9th day of disease depending on various different factors |
| Isolated    | People who have been sent to isolation, so that other people does not get infected | 0 = No Isolation  
1 = Isolation |
| Recovered   | People who have been recovered from the disease | When there are no traces of SARS in the person’s body |
| Dead        | People who have died due to the SARS infection | Once the person gets infected, his immune system never overcomes. |
| Immune      | People with better immunity at a certain point in time to get infected | If immunity of a person is strong, but there is a fair chance of immunity changing as time passes. |

The emergence of SARS virus is quite interesting, for instance it took less than 2 days for the SARS virus to migrate from China to Canada\(^12\) with important societal consequences. Due to this property of spreading fast,
studying SARS emergence by modeling and simulation can be helpful in developing an algorithm for task allocation in distributed systems. Task allocation is the system property that emerge within the distributed system following insights from the SARS pandemics. In this research we have followed an unorthodox approach of assuming the spread of SARS virus among the population as the assigned task to the system, and the number of patient getting infected or sick due to virus are interpreted as completion of the task.

1.2. Problem Description

This paper discusses about a novel way of allocating a given task over a distributed system. Traditionally, the task allocation has been a challenge in distributed systems. We have considered a system comprising of a set of n processors, $P = \{P_1, P_2, \ldots, P_n\}$; and a task $T$ that needs to be allocated over the system. The global task of "Software Update" is used to instantiate the epidemiology-based distributed algorithm. The “Software Update” task has been generated by one of the processors and the challenge is to update each and every other processor with the software update in an efficient, cheap and faster way. The optimal task assignment procedure over the system increases the system reliability by immediately disseminating the update. The task allocation procedure consists of inter-process communication, intra-process communication, message passing, and task execution processes which make it a very costly procedure. The main aim of this work is to reduce the overall cost of task allocation by reducing the cost (e.g. processing time, communication time, etc.) of the above mentioned individual procedures.

2. Model

2.1. Model description

Modelling a complex system helps us to point out important mechanisms underlying any kind of occurrence of a system. We utilized an agent-based model for this research that provides us with the capability to observe non-linear dependencies and understand the individual level contribution in the emergence of system properties. The agent-based modelling equips us with the capability to observe and understand the distributed interactions occurring among the different participants such as agent-agent interactions, agent-environment interactions etc. in the system. An emergent property is a surprising pattern (i.e. a system property) that is observed as a result of different distributed interactions in the system. Understanding emergence will give a better insight about the factors contributing the most in existence of the system.

Epidemics is a complex system, as there are many agents with unpredictable behavior, an environment formed of many controlled and uncontrolled parameters, and lots of distributed interactions.

We created a NetLogo model to characterize the pandemics system with environmental settings that are relevant to real-life settings. The modelling of SARS outbreak in a community has been carried out in this work. A small community of population varying from 1,000-10,000 is simulated. Instead of describing the series of occurrence based on statistical data of 2002 SARS epidemiology, our model tries to describe the behavior of SARS virus in any community. The model is developed mainly keeping the functional properties of SARS-Cov virus along with some properties and behavioral aspects of common people.

The parameter transmission-rate and spread-ratio are used in the model to understand the significance of these properties on the emergence of the pandemics. The interventional-parameters like isolation helps us to understand the behavior of SARS virus with and without any foreign intervention to resist its spread. In this model we have mainly focused on keeping the agent as well as environmental properties as random as we can, this helps in understanding the efficiency of SARS virus in a very uncontrolled and unpredictable environment. The agent properties like immunity and day-of-infection play an important role in understanding the significance of individual properties in the spread of SARS virus as well. Moreover, the different phases of the disease are taken into consideration, for instance the patient in latent phase cannot spread the disease, but the immunity of the patient might increase or decrease. In our model each NetLogo tick, which is a unit of simulation time, represents 1 hour. We simulated the model for a period of 272 days in our experiments, which was the original span of 2002 SARS epidemics, to give us a reasonable period of time to observe the emergence of the pandemics.
3. Experimental Results

For each set of experiments, we have considered four main factors: (a) Infected patients inserted during the model initialization (b) Transmission-rate of disease (c) Isolation: On/Off (d) Spread-ratio: Average, Slow or Fast. To observe the significance of these factors in the emergence of SARS virus in society, we have followed a sensitivity analysis method of changing each factor at a time to see what factor affects the most. Moreover, during the experimental phase of study our focus was to identify the inter-dependency among the factors.

In our experiments we observed that if the intervention procedure of Isolating patients is not followed, then the number of infected patients increases substantially. Thus we came to conclusion that Isolation of patient is an important factor to control the spread of disease. Figure 1 denotes the results when the Isolation intervention is considered, while Figure 2 shows the results when the Isolation intervention is not considered.

Fig 1. Isolation Intervention
Fig 2. No Isolation Intervention

4. Epidemiological-based Task Assignment Algorithm

Following the experimental results obtained from the modeling and simulation of the SARS epidemics, we translated learnt insights into a distributed system algorithm for the task assignment problem. We consider the emergence of the pandemics as a global task to be allocated in a distributed system.

4.1. Assumptions

Let us assume that a processor $p_i$ has a “Software Update” package, and the task is to update the software of each and every processor in the network. To allocate the task, the following assumptions in Table 2 are considered:

| Property                        | Assumption          |
|---------------------------------|---------------------|
| Network type                    | Closed and Secured  |
| Peer Trust factor               | Very high           |
| Task                            | Software Update     |
| Processing required for task    | Pre-defined         |
| Memory required for task        | Pre-defined         |
| Execution time                  | Pre-defined         |
| Inter-task communication cost   | 0                   |
| Frequency of communication      | Frequent            |

4.2. Proposed Epidemiology-based Task Assignment Algorithm

The following algorithm outlines how we the “Epidemiology-based task assignment” works. One assumption is that a seed task “$t_j$” is initialized on one of the processors “$p_i$”. The seed task “$t_j$” starts the pandemics. The algorithm is divided into two main procedures: (a) Disseminate_task and (b) Execute_task. Both procedures act as two independent threads executed on the processor simultaneously. The thread of Disseminate_task is controlled by
the external flag “ISO”. If ISO = ON, then the processor kills the thread of “Disseminate_task” stopping the spread of the task at that processor. CLOSE_CONTACT is the function that calculates the trust factor between the two peers in network. The trust factor among the peers is decided by formulate (1).

ALGORITHM:

INPUT : Processors $P = \{p_i\}_{i=1}^{N}$, // $N$ = number of processors in the system
Processors capacities $\{C_i, M_i\}_{i=1}^{N}$
Execution time of the component of task on processor $p_i$: $\{E_i\}_{i=1}^{N}$
Task $T = \{t_i\}_{i=1}^{M}$, // $M$ = number of task components
Task component requirements: $\{c_j, m_j, e_j\}_{j=1}^{M}$

OUTPUT RESULT: Task allocated to every node over the network

PROCEDURE: Initialize_task[] // Step starts the seed task
$p_i \leftarrow \text{INITIALIZE}[t_i]$ // Initialize the seed task on processor $p_i$
$P\text{pltn} \leftarrow \sum_{i=1}^{N} p_i$ // Counts the total number of nodes in network

PROCEDURE: Disseminate_task [] //Thread1: Spreading of task over the network
for $i = 1, 2, 3, \ldots, N$ do
  for each close_contacts of $p_i$ do
    List $\leftarrow \text{SELECT q-of [close_contacts]}$ // List of nodes in close contact of processor “$i$” is created
    $p_i \rightarrow \{p_i\}_{i=1}^{q}$ // The package containing task “$t_j$” is transferred
  end
end

PROCEDURE: Execute_task [] //Thread2: Execution of Task component on processor
for $i = 1, 2, 3, \ldots, q$ do
  if $c_j < C_i$ and $m_j < M_i$ then // Checking if required processing and memory are available
    $\text{EXECUTE}[t_i]$ // Execution time of Task component on processor “$i$”
    $\text{INCREMENT}[E_i]$ // Checks if the Execution of task on processor is greater than
    While $(E_i > \frac{1}{2}e_j)$ do // processing required for completion of task component
      if ISO = false then
        $\text{CONTINUE}[\text{Disseminate_task}]$
        $\text{CONTINUE}[\text{Execute_task}]$
      end
      else
        $\text{STOP}[\text{Disseminate_task}]$
        $\text{CONTINUE}[\text{Execute_task}]$
      end
    end
  else
    $\text{HIST}[C_i, M_i]$ // HIST function monitors the processor “$p_i$”
  end
  if EXECUTION NOT COMPLETED then // Error handling mechanism
    $\text{ROLLBACK TO LAST COMPLETED CHECKPOINT}$
    $\text{KILL}[t_i]$ // Error handling mechanism
    $\text{ANALYZE}[\text{HIST}]$
  end
end
FUNCTION: close_contacts

\[
\text{TRUST\_FACTOR} = \frac{\text{Total number of transactions between two processors in a defined time period}}{\text{Total number of transactions by a processor with other processors in the network}}
\] (1)

5. Discussions

In order to tackle the challenge of task assignment, the most appealing epidemic property is the speed at which the disease spreads among the community. This property leads to the rise of the epidemic. We are interested in the way that disease starts from a single person but goes further to spread throughout the community, within days. We try to inherit such properties from epidemic to design an algorithm for task assignment services in a distributed system.

In the majority of cases, the complex system of epidemics is the result of continuous interaction among people within same geographical areas. Thus one can say that epidemics is a peer-to-peer (P2P) system, where each peer/node is a person who might get infected by the virus causing disease. The virus can be interpreted as a necessary task which is needed to be assigned to every node in the network. The close contact among the people in epidemics can be interpreted as a trust factor among the different nodes in the distributed system. Since the agents follow random motion in the epidemic model, a susceptible person might come within a 3-meter radius or close contact of an infected person. This is translated as a transaction in the form of messages or other communications among the nodes of the distributed system. One node can have a number of transactions with several other nodes in the system. Since the nodes based on trust factor w.r.t. to a given node “i” are listed, the package containing the task is randomly transferred from the node “i” to a susceptible node. This activity is similar to the transfer of the virus from an infected to the susceptible person.

Once the virus penetrates into the body of a susceptible person, it starts reproducing itself and affecting the immunity of the body. In a distributed system the immunity for nodes is the free computing and free memory capacity available in the node. The available memory and computing capacity varies based on the circumstances. Hence as suggested in the algorithm, if the required parameters match, the processing starts which can be related to the infected phase of SARS. Whereas in some cases, due to a lack of processing and memory available, the package will be inactive. This is the latent phase for that node, where the package is present in the system but it is waiting for required computing capacity and memory space to activate.

The next stage from latent phase in SARS is the infected stage, where the infected person is very contagious. We replicate this scenario with the “Execute_task” thread, where the node starts spreading the task to other peers in its trusted network. Since each node has its own independent properties, non-linear effects on the dissemination of the disease have been observed in the systems.

A control factor that came as a boon in disguise during the 2002 epidemic of SARS was sending patients to complete Isolation. For the P2P system the same feature of sending the nodes to Isolation acts as an important aspect of the algorithm. If the “Isolation” is ON, it means that a peer cannot share the package with any other peers to its close contact in the network. Using the “Isolation” factor in this model, the speed of task assignment can be controlled. If the “Isolation” is OFF, it means that the peer continues “Disseminate_task” along with “Execute_task” thread.

Task assignment is one of the biggest problem in the field of distributed system. Today, in the world of fast-technology where every node is mobile, task assignment among the whole system is even more difficult and chaotic than before. Due to this chaos, the probability of bottlenecks in the system is high, which make the system more susceptible to failure. We are in the constant quest of more efficient and faster algorithms for task assignment, we believe that CAS-based algorithm can serve the purpose.

Having no centralized controller and giving every entity power of taking decisions, make the system more realistic and faster to react to service demands. Moreover, as every node behaves freely in the system, the efficiency of the system increases and the probability of traditional problems like bottlenecks, slow response rate etc. is
reduced. Furthermore, the emergent pattern of execution of the task assignment make the system more secure as no one can predict how the system behaves.

Our study has some limitations, the data collected during the SARS analysis might not be completely realistic as many patients in remote areas might not have been registered on governmental records, which were used to create our SARS models. Although we consider a variety of factors in the spread of the disease, there can be other social factors (e.g. personal decisions) that may also prove to be relevant for the spread of the disease. This paper focused on the modeling of the SARS pandemic and the identification of relevant factors and dependencies, which can be used in the design of a task assignment algorithm for a distributed system. Future work will implement the algorithm and evaluate its performance.

6. Conclusion

We have leverage the existing similarity between disease epidemics and distributed system services. The paper evaluates several factors on the SARS pandemics from a CAS perspective. This is very useful because it provides several insights and inspiration used to develop an algorithm for the task assignment problem in a distributed system. The number of people getting infected or dying of the disease were interpreted as task completed, while the people not getting affected were represented as incompetence of the algorithm. The model was designed so as to find out the factors contributing to the rise of disease and further exploiting these factors to get better efficiency from the system. As the end result of this model we have designed the algorithm for distributed system we call it “Epidemiology based Task Assignment algorithm”.

Acknowledgements

The authors hereby express their gratitude to the funding organizations NSERC Discovery Development Grant DDG-2015-00005, the UoFw FGS Grant, the UoFw Research Office Special Grant and the UoFw ACS Department for providing resources and infrastructure for our research.

References

1. Porta M., Greelands S., Last JM, ed. A Dictionary of Epidemiology. Oxford University Press. New York, 1995. Page: 93-95
2. Brauer F., Van Den Driessche, P. and Wu, J. editors. Mathematical Epidemiology. Lecture Notes in Mathematics. Springer Verlag-Berlin Heidelberg 1945.
3. Top ten causes of death. World Health Organization Official Website; Geneva-Switzerland.[Last-accessed: May 2016] Link: http://www.who.int/mediacentre/factsheets/fs310/en/
4. Marathe M. and Vullikanti A. 2013, Computational epidemiology. Commun. ACM56, 7 (July 2013), Page: 88-96. doi: http://dx.doi.org/10.1145/2483852.2483871.
5. Booth CM, Matukas LM, Tomlinson GA et al. Clinical features and short-term outcomes of 144 patients with SARS in greater Toronto area. JAMA 2003; 289(1):2801-2809. doi: http://dx.doi.org/10.1001/jama.289.21.JOC30885.
6. WHO summary of portable SARS cases with onset of illness from November 1 2002 to 31 July 2003. World Health Organization Official Website; Geneva. [Last accessed: April 2016].Link: http://www.who.int/csr/sars/country/table2004_04_21/en/print.html
7. Andrew T., Steen M; Distributed systems: Principles and Paradigms. Edition 2. Pearson Education. New Jersey 2007. Page: 2-3
8. Scholz and E. Harbeck, "Task assignment for distributed computing," Advances in Parallel and Distributed Computing, 1997. Proceedings, Shanghai, 1997, pp. 270-277. doi: 10.1109/APDC.1997.574043.
9. Jian Lin, A.M.K. Cheng, R. Kumar, "Real-Time Task Assignment in Heterogeneous Distributed Systems with Rechargeable Batteries", AINA, 2009, 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA) 2009, pp. 82-89, doi:10.1109/AINA.2009.41
10. WHO summary of portable SARS cases with onset of illness from November 1 2002 to 31 July 2003. World Health Organization Official Website; Geneva. [Last accessed: April 2016].Link: http://www.who.int/csr/sars/en/
11. Centers for Disease Control and Prevention. Atlanta-USA. [Last Accessed: May 2016] Link: http://www.cdc.gov/sars/about/fs-sars.html
12. WHO summary of portable SARS cases with onset of illness from November 1 2002 to 31 July 2003. World Health Organization Official Website; Geneva. [Last accessed: April 2016].Link: http://www.who.int/csr/sars/country/en/
13. Chowell, G., Castillo-Chavez, C., Fenimore, P. W., Kribs-Zaleta, C. M., Arriola, L., & Hyman, J. M. (2004). Model Parameters and Outbreak Control for SARS. Emerging Infectious Diseases, 10(7), 1258–1263. http://doi.org/10.3201/eid1007.030647.
14. Wilensky, Uri, and William Rand. 2015. An Introduction to Agent-based Modeling: Modeling Natural, Social, and Engineered Complex Systems with Netlogo. MIT Press. https://mitpress.mit.edu/books/introduction-agent-based-modeling.