International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 3 Issue 4, April-2014, pp: (278-282), Impact Factor: 1.252, Available online at: www.erpublications.com

Parallel Efficiency for Graph Search

Pooja Devi[1], Shalini Gupta[2], Sunita Choudhary[3]
[1,2]M.Tech. Scholar, Dept. of Information and Technology, Banasthali University, Jaipur
[3]Associate Professor, Dept. of Computer Science, Banasthali University, Jaipur

Abstract: Cloud computing is offering utility oriented IT services to users worldwide. It enables hosting of applications from consumer, scientific and business domains. However data centers hosting cloud computing applications consume huge amounts of energy, contributing to high operational costs and carbon footprints to the environment. With the increasingly ubiquitous nature of Social Networks and Social Cloud users are starting to explore new ways to interact with and exploit these developing paradigms. Social Networks are used to reflect real world relationships that allow users to share information and form connections between one another. Cloud services allow individuals and businesses to use software and hardware that are managed by third parties at remote locations. Examples of Cloud services include online file storage, social networking sites, webmail, and online business applications. The Social Graph in the Internet context is a graph that depicts personal relations of internet users. The social graph has been referred to as "the global mapping of everybody and how they're related". As we know that social graph have large amount of data. Accessing useful information from large amount of data is very difficult. So to avoid this problem Map Reduce processing paradigm has used. Map Reduce is a programming model for processing large data sets with a parallel, distributed algorithm on a cluster using parallel computing. Parallel computing is a process that simultaneously uses various computing resources to solve problems which has the advantages of speeding up program execution and saving cost. In this paper we are trying to simulate the concept of Map Reduce and to get good parallel efficiency ratio.

Keywords: Cloud Computing, Map Reduce, Parallel Computing, Parallel Efficiency, Social Networks.

Introduction

Cloud computing refers to applications and services offered over the Internet. These services are offered from data centers all over the world, which collectively are referred to as the "cloud." This metaphor represents the universal nature of the Internet. Examples of cloud computing include online backup services, social networking services, and personal data services such as Apple's Mobile. Cloud computing also includes online applications, such as those offered through Microsoft Online Services. Hardware services, such as redundant servers, mirrored websites, and Internet-based clusters are also examples of cloud computing [1]. Cloud computing has some characteristics i.e. on-demand self service, broad network access, resource pooling, rapid elasticity, pay per use. Cloud computing and social networking has intermingled in a variety of ways. Most obviously social networks can be hosted on cloud platforms or have scalable applications within the social networks. Social networks are networks of users connected through relationships such as friendship, following or otherwise. Through these relationships, users are able to share content amongst themselves. There are numerous existing social networking websites such as Orkut, Facebook, Linked.in, and Google+. On these sites, one of the greatest concerns has been the security and privacy of personal data. That is to control the personal information that is being shared to other users and social applications, as well as how information is being shared with third-parties. Social networking sites use in various fields i.e. business, marketing, entertainment and purely personal reason etc. It makes a social graph by connecting people to each other via social networking sites. Social graph is a set of vertices, or nodes, and edges that connect pairs of vertices. A drawing of a graph in which each person is represented by a dot called node and the friendship relationship is represented by a line called edge. The social graph in the Internet context is a graph that depicts personal relations of internet users. To search information in short time in Social Graph, the concept of parallel computing is widely used. Now parallel computing is the use of multiple processors or computers working together on a common task- Each processor works on its section of the problem and processors can exchange information. Parallel computing allows us to: Solve problems that don't fit on a single CPU and solve problems that can't be solved in a reasonable time. A scalable parallel implementation is one where: (a) the parallel efficiency remains constant as the size of data is increased along with a corresponding Increase in processors and (b) the parallel efficiency increases with the size of data for a fixed number of processors [2].

Graph Creation for Social Networks

A social network is a structure of entities interconnected through a variety of relations. These entities are typically referred to as "users". The relationships between these users have a number of different names across different social networks such as friends, or followers. Through these relationships users share messages and media amongst themselves. There exist a number of online social networking websites such as the popular Facebook, LinkedIn and Twitter. These social networking sites have well over 100 million active members. With such a great number of users using these services, social networks present an interesting area of study in a variety of ways. Here Graph creation provides the facility to search user interest by using graph simulator. This graph simulator provide some facilities like-

- How to create nodes
- Clear all nodes
- Delete a particular node
- How to change size, shape and color of nodes
- How to connect nodes.

This section show that how to connect nodes with single node and after that connected nodes connect with other nodes and finally connect initial node and final node at third level. In other way, make a graph where one node connected to other node and other node is connected to more number of other nodes and find out the common interest between one node and other more nodes in very efficient and fast manner. Here node act as a people and connection between nodes make an edge. Figure 1 shows that initially one person (N1) have two friends.

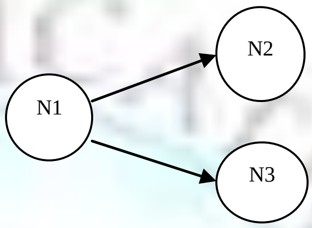

Figure 1. : **Initial level** **1st level**

Now one people (N1) has two friends (N2 and N3) but node N2 and N3 also have two friends (N4, N5 and N6, N7). This is shown by figure 2.

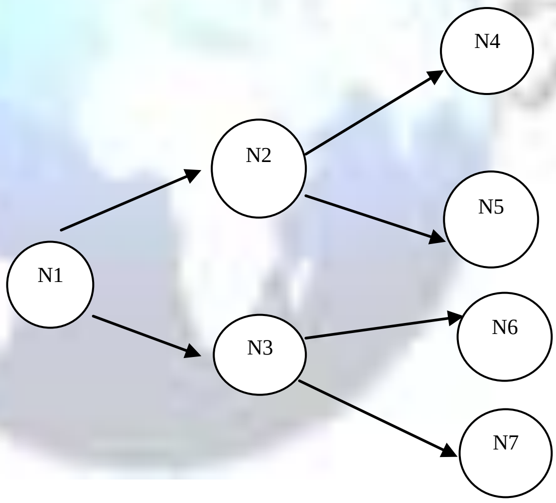

Figure 2: **Initial level** **2nd level** **3rd level**

Above figures shows that these are the very small graph but now a day's social networking sites frequently used by the many other peoples so that number of contacts will increase day by day and also accessing will become very difficult. In this paper initially assume 20 nodes and each have 2 friends and at 3rd level each have 2 friends so we have 400 total number of friends to search this is very difficult and searching process become very complex and take more time. To solve this problem Map Reduce concept has used. In this paper we also focus on database that have all information about the people (Nodes) like- name, job, gender, organization etc and show all details on the screen like- name, city, gender, job, organization and relationship on a node by fetching the information from the database and also show wanted node information when we select that particular node.

Implementation of Map Reduce

Social Networking sites like Facebook has huge amount of information about people like what they are interested in, books they read, restaurants they like and more. From this huge amount of information, any one may have different interest of search. For example:

International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 3 Issue 4, April-2014, pp: (278-282), Impact Factor: 1.252, Available online at: www.erpublications.com

- Therefore users face many problems to search a common interest like hobby, reading interest.
- User may be interested in finding a person having same birth city or some specific organization.

In facebook, one people have many hundred of friends and these hundreds of friends also have many friends. In these contacts, there may be persons sharing common things with user. To search such person from huge list is difficult task. In this paper, we are trying to simulate the concept of Map Reduce that search a common interest from huge amount of information and try to get good parallel efficiency ratio. Take a similar example like-Search results draw from any information a Page, place, group, app, game, or person has shared with connections or the public. The first version of Graph Search focuses on people, photos, places, and interests, but will expand to include data from posts, status updates, and third-party apps. Search results are unique to each person as no two social graphs are the same. Also, results are based not just on relationships, but rather strength of the relationships.

A search for "People who like the things I like" might yield a set of results that look something like the image below. Since that particular search results in over 1,000 people it might be worth filtering. Using the column on the right, we can filter by categories like current city and organization.

Figure 3

The following figure shows Map Reduce concept. In this concept we take different number of Mapper and Reducer to solve above problem [4]. When we apply number of Mapper and Reducer on nodes then we find a cluster wise result so that graph searching can easily do. This can be explained by 5 Mappers and 3 Reducers example:

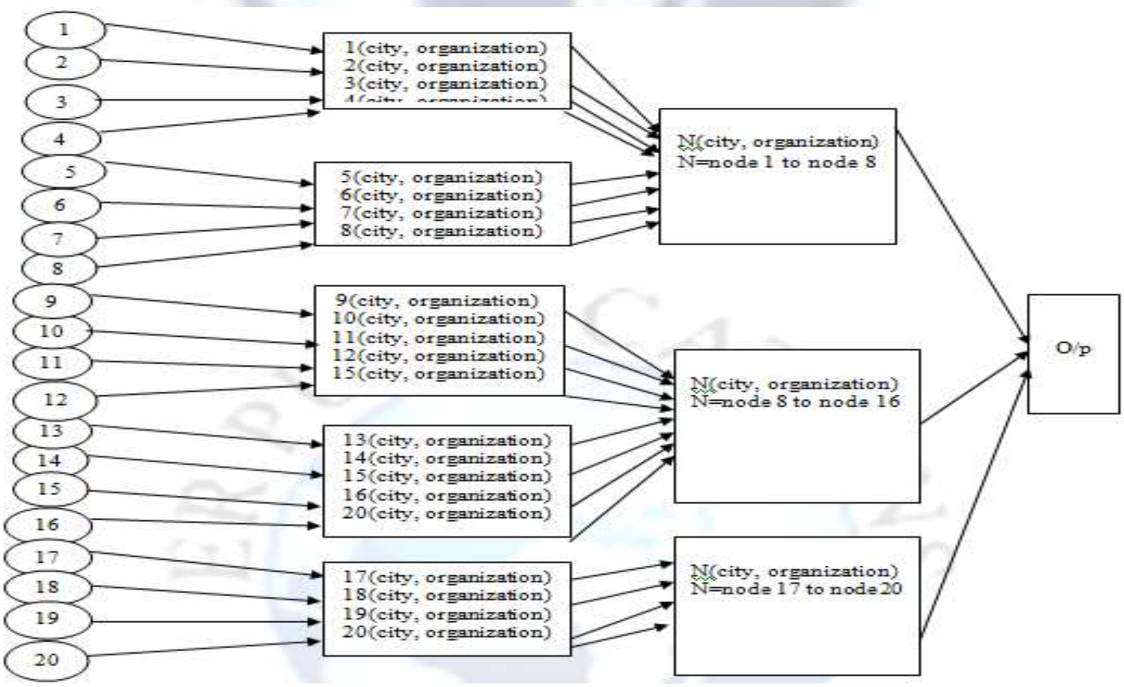

Figure 4

Here we have an example of 15 nodes for implementation. In these following figures we use one mapper, two mappers and one reducer on a 3rd level of graph and we can also increase number of mapper and reducer and calculate a good parallel efficiency ratio. These figures show the actual implementation of Map Reduce with different number of mapper and reducer.

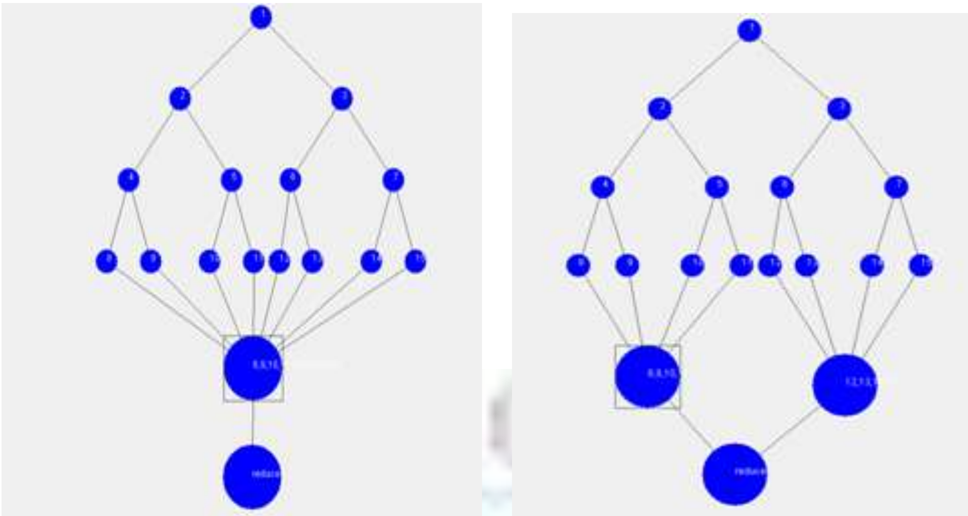

Figure 5: Map Reduce

Calculate Parallel Efficiency for Map Reduce

Parallel efficiency is impacted by overheads such as synchronization and communication costs, or load imbalance. The Map Reduce master process is able to balance load efficiently if the number of map and reduce operations are significantly larger than the number of processors. For the purposes of our analysis we assume a general computational task, on a volume of data D which takes wD time on a uniprocessor including the time spent reading data from disk performing computations and writing it back to disk (i.e. we assume that computational complexity is linear in the size of data). Let c be the time spent reading one unit of data from disk. Further, let us assume that our computational task can be decomposed into map and reduce stages as follows: First cmD computations are performed in the map stage, producing σD data as output. Next the reduce stage performs crσD computations on the output of the map stage, producing σμD data as the final result. Finally, we assume that our decomposition into a map and reduce stages introduces no additional overheads when run on a single processor, such as having to write intermediate results to disk, and so on.

$$wD = cD + cmD + cr\sigma D + c\sigma\mu D$$

Now consider running the decomposed computation on P processors that serve as both mappers and reducers in respective phases of a Map Reduce based parallel implementation [3]. As compared to the single processor case the additional overhead in a parallel Map Reduce implementation is between the map and reduce phases where each mapper writes to its local disk followed by each reducer remotely reading from the local disk of each mapper. For the purposes of our analysis we shall assume that the time spent reading a word from a remote disk is also c, i.e. the same as for a local read. Each mapper produces approximately σD/P data that is written to a local disk (unlike in the uniprocessor case), which takes cσD/P time. Next, after the map phase, each reducer needs to read its partition of data from each of the P mappers, with approximately one Pth of the data at each mapper by each reducer, i.e. σD/P2. The entire exchange can be executed in P steps, with each reducer r reading from mapper r + i mod r in step i. Thus the transfer time is cσD/P2 × P = cσD/P. The total overhead in the parallel implementation because of intermediate disk writes and reads is therefore 2cσD/P. We can now compute the parallel efficiency of the Map Reduce implementation as :

$$€mr = \frac{wd}{p\left(\frac{wd}{p}+\frac{2c\sigma D}{p}\right)} = 1/(1+2c\sigma/w)$$

Volume of data is D and time spent reading one unit of data on one disk is c. The map phase produces *mP* partial counts, so *σ = mP/D,* where m is number of distinct nodes.

After calculating Parallel Efficiency on 15 nodes result will display in form of table and graph.

No. of Mappers	No. of Reducers	Time	Volume of Data	Efficient ratio
1	1	2 min	10 Mb	89.72%
2	1	2 min	10 Mb	90.22%
4	1	2 min	10 Mb	91.09%
4	2	2 min	10 Mb	92.44%

Figure 6

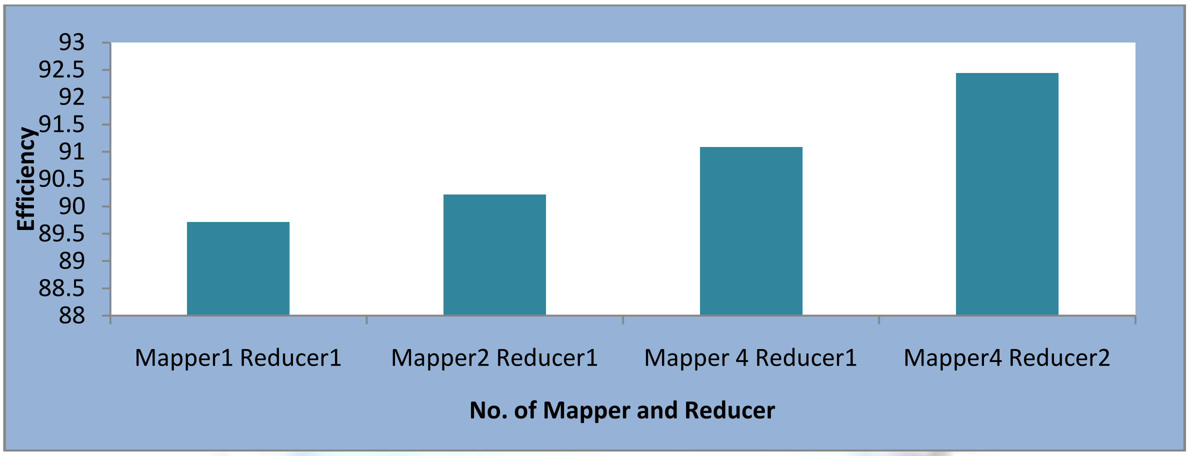

Figure 7 Parallel Efficiency graph

Acknowledgment

We gratefully acknowledge to the Dr. Sunita Choudhary mam who have contributed towards development of this paper.

Conclusion

From all experiments we can come to the following conclusions on performing parallel computing using cloud computing concept. Cloud Computing work well for most pleasingly-parallel problems. Their support for handling large data sets, the concept of moving computation to data. Cloud computing and social networks are two of the more powerful movements in the web 2.0 space. So the potential of social media and the cloud integrating is compelling to say the least. Need customer service applications that try to capture the crowd sourced pools of knowledge floating across the internet from sites like Google, Facebook and Amazon, and then use this information to better equip commercial customer service operations with useful knowledge. Social networks are becoming much more than an online gathering of friends; Facebook and Twitter are becoming destinations for ideation, e-commerce and marketing. It's of no surprise that companies want an easy and simple way to capture all of the information that is relevant to their businesses and then leverage this knowledge to improve customer service. In this paper we are simulating the concept of Map Reduce and try to get good parallel efficiency ratio. By comparing parallel efficiency with different number of processors we can easily find out how many number of Mapper and Reducer will provide a good efficient result. Finally we observe that this work may become very helpful to find out people throughout world with their more or less information.
.

References

[1]. F. Fabris, Introduction to Cloud Computing Architecture, Sun Microsystems, Inc.
[2]. G. C. Fox. Solving Problems on Concurrent Processors. Prentice Hall, 1988.
[3]. G. Shroff, ENTERPRISE CLOUD COMPUTING TECHNOLOGY, ARCHITECTURE, APPLICATIONS. Published in the United States of America by Cambridge University Press New York.
[4]. J. Dean and S. Ghemawat. Map Reduce: Simplified data processing on large clusters. In Proceedings of the 6th Symp. on Operating Systems Design and Implementation, 2004.
[5]. K. Chard, S. Caton, y. Omer Rana, z. Kris Bubendorfer, Social Cloud: Cloud Computing in Social Networks School of Engineering and Computer Science, Victoria University of Wellington, New Zealand.