

Preference based Resource Allocation and Sharing via Social Networks through Feedback Integration

Hima K S

*Department of Computer Science and Engineering
Nehru College of Engineering and Research Centre
Thrissur, India*

Avanish Kumar Singh

*Department of Computer Science and Engineering
Nehru College of Engineering and Research Centre
Thrissur, India*

Abstract

Resource allocation and sharing is one of the core foci of economic research. Whenever one group of people has access to, or can provide certain resources, and another group is interested in consuming these resources, mechanisms can facilitate such an exchange. Such mechanisms can include monetary based exchange, exchange based on credits, bilateral exchange etc. Here in order to facilitate this sharing mechanism, the pre-established trust formed through friend relationships within a Social network is used to form a dynamic cloud-like platform, where the sharing of resources occurs through “friend” relationship. The users can have preference with whom they need to share their resources. Preference mechanism is provided through feedback integration where the preference is calculated automatically, unlike the traditional method where it is provided manually, which is difficult in larger user scenario. Feedback is provided by the already existing users. For the matchmaking between consumers and providers, Deferred acceptance algorithm is used which provides a stable match among the users. Feedback Integration mechanism is used to provide ranking between friends initially which reduces the load in providing the ranks manually especially in case of large number of users.

Keywords: Preference, Social Networks, Feedback, Allocation

I. INTRODUCTION

This paper solves the issue of trust and accountability in cloud by bringing in a new concept called “Social Compute Cloud”¹. A social cloud is a dynamic cloud-like sharing platform utilizing the concept of social network. A social network is a digitally encoded form of socially connected peers which have social relationships among its members. The problem of trust can be solved by including social network into this dynamic platform. Social networking sites such as Facebook, twitter, Google+ etc. plays an important role in everyday life. Here social networking is used as a medium for sharing and allocating resources because of its widespread use and size. Social networks are chosen as a platform for sharing due to its vast importance today. As per Pew Internet Project’s research related to social networking, from the year 2005 to 2014 there is an increase from 8 percent to 74 percent regarding the percentage of adults who use social networking sites while accessing the internet. Also as of September 2014, 71% of online adults use Facebook.

The concept of social cloud is motivated by the vision that the socially connected peers could access the resources which they are in need of but which they are not in possession of. So the concept of social compute cloud brings the idea that the resources are shared using a social network platform.

The main thing to be noted here is that these resources are only shared and there are no monetary benefits associated with it. In the earlier papers of Social cloud² there was a virtual credit model, but this paper of Social Compute Cloud¹ does not tell about any of the selling resources but only sharing of resources through a Social Clearing House. The main benefit associated with this model is that the users can have preferences with whom they can share their resources. These preferences are derived automatically through the feedback mechanism³ obtained from the neighboring peers. To support user preferences several algorithms are considered for the matching of users i.e. the consumers and providers.

Today the importance of social networks has increased. So people are using these social networks for different types of collaborations like scientific portals. There is Social Volunteer Computing⁴, proposed by McMahan and Milenkovic, which is used for the peer to peer storage, and here also there is an underlying social relationship between the customers and the providers, but unlike Social Compute Cloud there is no bilateral exchange of information in this.

There is an also off-site storage system like FriendBox⁵, a friend to friend system which provides secure and private storage system. Cooperative backup systems like FriendStore⁶ is proposed where the users are given choice to what nodes are required for them for the backup.

There are several algorithms that are considered in the preference matching. C. Haas, S. Caton, and C. Weinhardt⁷ proposed several algorithms and various incentives for the social cloud. Here the authors investigate how the resources are allocated effectively and also the performance of various algorithms that they have considered. The authors also considered that the Social

Compute Cloud can be implemented by extending Seattle⁸. Seattle is an open source peer-peer educational platform which can be used to implement virtualization using its lightweight virtualization layer.

II. SYSTEM MODEL

A social compute cloud is provided to enable access to resources which are contributed by the socially connected peers. Here the resources are owned and consumed by the members of the social community. By this the consumers are able to access the resources that are contributed by the providers. Through this the consumers can execute applications on compute resources that are provided by the providers.

The architecture of social compute cloud is given in Figure No. 1. There are three main sections in the architecture. They are the Social Cloud Platform, Technical adapter and the Socio economic model.

A. Social Compute Cloud Platform

There should be a platform which should coordinate the basic functionality of the system. The Social Cloud Platform is the main part of the architecture. It contains the social clearing house, the middleware, the databases and the matching mechanisms.

The Social Clearing House is the central part of the system. It contains mainly two databases: one for maintaining the details regarding the resources and the other which contains details regarding the users. The details regarding the users can mean the social ties and other preference details. It also contains the details regarding with whom the users can share data.

There should be a middleware in order to implement virtualization. It should provide various sandboxing mechanisms. Seattle is mentioned in the model as a middleware to access the resources. The lightweight virtualization can be provided easily using the Seattle platform. Actually Seattle is an open peer to peer virtualization platform.

A socio technical adapter provides access to the users' social graph. Through this the social ties or the graph of a particular user can be determined. Once the social ties of the user are being interpreted the social clearing house checks for the preferences associated with a particular user. A preference module is used for this.

The matching mechanisms provide the necessary allocation of resources for the customers.

B. Technical Adapter

To enable sharing of resources there should be access to users' social network. For this purpose a technical adapter is used. Here Facebook is considered and as an access to users' profile, an authentication using the API's are considered such as Facebook Connect. Once the social graph is obtained the question is how to interpret the social ties. The social tie interpretation include how to provide and with whom to share the resources. There are many methods and any one can be adopted. It can be by 1) manually ranking the friends numerically, 2) adopt some deduction methods to calculate the preference or 3) decide using some mathematical models.

There are many advantages and disadvantages for these methods. The method mentioned initially is very easy and can be done manually. But the problem arises when the social graph is very large. In such cases it will be difficult to rank each friend manually. The other methods will be convenient for large data sets. But such methods will have more computational overhead.

Another important concept to be considered is the type of preference that exists. The preference profile consists of the ranks that a user has towards another user. So in such a case,

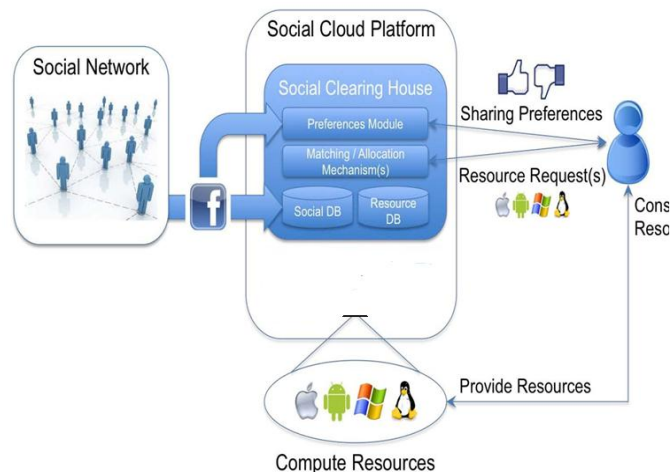


Fig. 1: Social Compute Cloud Model

There can be many choices. Preference is represented through ranked ordered lists. A preference profile for user A is said to be complete if preference ranking is given for every other users. It is incomplete if user A leaves at least a single another user unranked

C. Socio Economic Model

This part of the model shows the allocation process of resources. Once the preferences are captured the request for the resources and the availability of resources is to be checked. All these are done in the Social Clearing House i.e. the central part of the system. The overhead associated with this model is the storage of the necessary information associated with it. When the supply and demand are captured certain constraints are specified such as stability, fairness and the allocation process is done according to that. This model concentrates only on non-monetary benefits. The requests are being captured and the allocation is done using certain algorithms which is discussed in the next section. The allocation is done on one-one basis and this paper does not focus on many to many or one to many basis.

III. PREFERENCE RANKING USING FEEDBACK INTEGRATION

Preferences can be mentioned manually or can be captured leveraging some methods. Several algorithms are used for the preference based matching. Feedback method is used here in the decision making of preference rankings, in order to avoid being matched to untrustworthy users (or provides not likely to deliver their service). In the real world scenario the preference may change over time. So there will be difficulty in ranking people with whom we have not interacted so far. So there is a need for trust networks. These trust networks incorporate the feedback of other users in addition to own experiences about transactions with other users. By using trust networks and the corresponding trust values, users might be able to improve the accuracy of their preference rankings over time. There are four options associated with feedback provision: not giving any feedback, giving truthful feedback, giving partial feedback (i.e. either a limited number of users give feedback or feedback is not given at every time step), giving untruthful (malicious) feed-back. When aggregating feedback it is useful to note that a single value is only received by the peer which is the rank and the method used for calculating this value is not revealed. Ranking is calculated using the feedbacks that are given by the peers. The feedback is calculated using the positive comments and negative comments. If there are no comments about friends then the user can give the ranking manually. Certain functions are used to calculate the rank using the negative and positive comments. Total count of the negative comments and positive comments for a particular user is calculated. Then its fraction against the total word count is calculated. Based on the positive comments and the negative comments the rank is generated automatically. If the positive comments are higher than its corresponding integer value is taken and if that integer value is lesser then lesser priority ranking is given. The reverse scenario takes place in the case of negative comments. After taking the integer values if that integer value is lesser then greater priority value is given.

IV. PREFERENCE MATCHING ALGORITHM

There are several algorithms that give stable matches based on certain preference structures⁶. The Figure No. 2 shows the matching scenario. Depending on the quality guarantees that an algorithm offers, they can be distinguished into exact, approximate and heuristic algorithms. Exact algorithms yield the optimal solution for a given scenario, and approximate algorithms guarantee that the solution quality is within a certain bound of the optimal solution. Heuristic algorithms, in general, do not provide such a quality bound, yet have other advantages such as the flexibility to consider multiple simultaneous objectives. Here Deferred Acceptance Algorithm is considered which comes under exact algorithm.

Gale and Shapley introduced this algorithm. This algorithm works for preferences that are without ties. So if there exist a tie then the tie has to be broken before giving as the actual input. The tie is broken according to the rule that FCFS. If there is an unmatched user of side 1 who has not proposed to all acceptable users in their preference list it is proposed to the partner that is highest rank in the list. If the partner is currently unmatched then that pair is matched and allocated. Also if the proposed partner is currently matched to a lower rank then it is temporarily detached and then matched to the side 1 partner. It can be easily adapted to cope with incomplete preferences, and indifferences are usually handled by breaking the ties first before applying the DA in its standard formulation. The disadvantage of this algorithm is that in case of incomplete preferences the DA does not guarantee a stable matching.

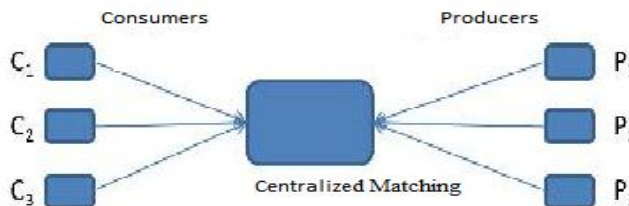


Fig. 2: Matchmaking process

V. CONCLUSIONS AND FUTURE DIRECTIONS

Allocation and sharing of resources can be done in many ways. But the way to do it using the trust formed by individuals in a social network is being shown here. Here a dynamic cloud-like platform is being created and utilized for sharing of resources. The users are given freedom to decide with whom they wish to share their resources by giving preferences. Preference-based

matching is a suitable approach to allocate resources on a non-monetary basis. The algorithm considered here focus on stable match. Here the preference is automatically allotted, therefore the burden for users is very less when compared to rank the users manually especially when there are large number of friends to be ranked. Also to increase the trust in automatic ranking mechanism, feedback is collected from existing users regarding another user and this feedback is integrated to obtain an appropriate rank. The feedback collected from friends do not involve complex mechanism, it uses simple commenting method. Altogether this system is user-friendly and do not involve complex methodology. Hence it is easy to use and can surely be used as a platform for sharing of resources.

ACKNOWLEDGMENT

We thank God almighty for giving us the strength and blessing us with his grace. We also thank the faculty members and students of our college for their valuable support. We also express our heartfelt thanks to Simon Caton, Christian Haas, Kyle Chard, Kris Bubendorfer and Omer Rana, whose work inspired us to put forward this paper.

REFERENCES

- [1] Caton S, Haas C, Chard K, Bubendorfer K, Rana O, "A Social Compute Cloud: Allocating and Sharing Infrastructure Resources via Social Networks," IEEE Transactions on Services Computing, 2014 Sept., 7(3):pp. 359 - 72.
- [2] Chard K, Caton S, Rana O, and Bubendorfer K, "Social Cloud: Cloud Computing in Social Networks," IEEE 3rd International Conference on Cloud Computing (CLOUD), 2010, pp. 99-106.
- [3] Haas C, Petri I, Rana O, "Improving Resource matchmaking through Feedback Integration," 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing.
- [4] McMahon A and Milenkovic V, "Social volunteer computing,"
- [5] Journal on Systemics, Cybernetics and Informatics (JSCI), 9(4): pp. 34 – 38, 2011.
- [6] Gracia-Tinedo R, M. Sanchez´-Artigas M, Moreno-Martinez A, and Garcia-Lopez P, "Friendbox: A hybrid f2f personal storage application," IEEE 5th International Conference on Cloud Computing (CLOUD) 2012, pp. 131-138.
- [7] Tran D. N, Chiang F, and Li J, "Friendstore: Cooperative online backup using trusted nodes," Proceedings of the 1st International Workshop on Social Network Systems (SocialNet'08), Glasgow, Scot-land, 2008.
- [8] Haas C, Caton S, and Weinhardt C, "Engineering incentives in social clouds," Proceedings of the 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2011)., 2011, p. 572575.
- [9] Cappos J, Beschastnikh I, Krishnamurthy A, and A. T., "Seattle: A platform for educational cloud computing," The 40th Technical Symposium of the ACM Special Interest Group for Computer Science Education (SIGCSE '09)., Chattanooga, TN USA, 2009.