Availability measurement in Peer to Peer Network Management Systems

Ouldooz Baghban Karimi[1], Saleh Yousefi[2], Mahmood Fathy[1], Mojtaba Mazoochi[3]

1. *Computer Engineering Faculty, Iran University of Science and Technology; Email: ouldoozbk@comp.iust.ac.ir, mahfathy@iust.ac.ir*
2. *Computer Engineering Department, Urmia University; Email: s.yousefi@urmia.ac.ir*
3. *Iran Telecommunication Research Center; Email: mazoochi@itrc.ac.ir*

Abstract

Scalability, decentralized processing time and power and also eliminating the single point of failure are some of the benefits of using peer to peer systems for management purposes. There has been little effort on formulating or evaluating different important factors of network management in P2P management networks. As an important factor we've focused on availability in network management systems. In this paper, we propose our framework for networks management, formulate availability in this platform. Then we provide an algorithm that helps to increase data availability in P2P management networks.

1. Introduction

Right network management solutions empower network operators to provide new services with the information they've gained on their current services, maintain the service quality they already provide, and manage billing and usage that would finally lead to increasing profits. Carefully planned management solution would also help service users get better service, and administrators reorganize their workload. On the other hand, operators are always in a challenge to reduce network operating costs and provide new services at an increasing speed. These facts highlight the need for an effective, automated, network management solution.

SNMP, the most recognized and widely used solution in networks management, as well as other solutions for network management, suffer from different deficiencies. Security problems that lead to many devices close their ports for them are one of the problem types. Single point of failure, which may cause lack of management if the only server that we have fails to operate correctly, is another problem. These systems also lack scalability potential.

Scalability, decentralized processing time and power, eliminating the single point of failure, self-management and dynamic behavior of nodes within the network are some of the benefits of using peer to peer systems for management purposes.

One of the requirements of a management system is to provide reasonable availability of all management entities as well as all data objects available in a management system and make sure that minimal work is done in the event of failure of an object request. Researches have been conducted on availability in management networks [3].

Availability for a system is normally defined as the property of being reachable and useable upon demand by an authorized entity, i.e.,

$$A = \frac{MTTF}{MTTF + MTTR},$$

where MTTR and MTTF are respectively Mean Time to Repair and Mean Time to Failure. There are several availability measurements and availability increasing methods for P2P networks [4, 5, 6, 9, 10, and 11]. Peer availability, average system availability, data object availability and service availability are some of these factors. Peer availability is the probability that the peer is up at any point of time and responds appropriately to messages received. Average availability is the average of peer availabilities across all peers. Data object availability is the probability of receiving a positive response to an object request in the system. Service availability is the ability of the P2P system to satisfy client requests which is also can be considered as the total data object availability of the system.

All these availability measurements are defined and formulated to be used for file-sharing systems. But there is no formal method to measure availability in peer to peer management networks. It is even not clear that which parameters of availability are desirable in peer to peer management networks.

In this paper, we have formulated availability measurements in P2P management networks. We've discussed that which availability factors are

978-1-4244-2917-2/08/$25.00 ©2008 IEEE

important in the management P2P networks and how we can measure the important availability factors in the P2P management networks. We have used a special framework for proposal for P2P management networks and we have used this modeling to obtain the calculations we've obtained. Later, we show by numerical analysis the impact of each factor taken into account in our calculations and concluded that the design we've used have different benefits to be used for peer to peer management networks.

A comparison of time-based availability and presence-based availability have previously been done in [1] for file sharing systems. We used both of these calculations in our method to obtain measurements for p2p management networks. But the time-based availability, in our opinion, was over-simplified because the frequency access which normally is taken into account in workload and service availabilities are ignored. Data object availability is also simplified in the service availability terminology.

The rest of this paper is organized as follows: In section 2, we introduce under-study management In section 3, we discuss important availability factors in P2P management networks and numerate them using available measurements used to formulate and evaluate the availability in P2P networks. An algorithm to increase the critical data availability in the proposed framework is presented in section 4. In section 5, we present a numerical analysis on the research. Finally the paper is included in section 6 followed by some future works direction.

2. Management Framework

There have been some framework proposals for P2P management networks [7, 8]. We considered a framework for our peer to peer management network for better performance. In this section, we are going to explore the management framework to be used in the rest of this paper. For now, take this framework as a proposal framework to obtain formulas, but later in this paper, we'd show that this framework has many benefits to be used as a framework for peer to peer management networks.

We need to handle peer grouping in our system to achieve the framework we need. We assume that our management network works on a structured peer to peer overlay.

We also handle some kind of grouping in this P2P network as ruled by the following rules: TLMs are Top level managers. Top level managers are responsible for interaction with user and execution of top management tasks such as making high level decisions that may affect the whole system. TLMs are allowed to have interface with the user and MLMs. MLMs are Mid-Level managers who can have interface with TLMs and LLMs. MLMs could be delegated by TLMs to do management tasks. LLMs are Low Level managers who can acquire information form the managed objects and send them to MLMs We assume that LLMs are not able to decide about the network state but they also can be delegated by MLMs to do management tasks, they are also data collection peers of the management networks.

TLMs, MLMs and TLMs are used to build up groups in their own levels. These groupings are done saving the group members as a list in the representative of the group and a backup of it on the backup representative like what is done for locality management in [11]. This representative is the one responsible to pass the information to the higher level. E.g. if we are in the LLM, the representative is to pass the information acquired from the network to the MLM requesting it and if we are in an MLM group, the representative of the MLM group is responsible to pass the information requested to the TLM needing it. The backup representative is used to eliminate the single point of failure in each grouping we've done.

Our main goal in a management system is to gain information form the system, decide upon the information collected from the system and then to apply the changes to the system based on the decisions made upon the information. Information could only be collected from the managed objects by the LLMs. LLMs maintain a filtering on the information acquired and send them to the MLMs. MLMs could send them to TLMs if the user has requested them or if the decisions is to be made in the TLMs. MLMs can also work on a management by delegation method, i.e. they can be delegated by the TLMs to do some management tasks. In this particular event there is no need for them to filter and forward the information to the TLMs.

We have an algorithm to maintain the grouping that we have for our P2P management network. Initial set up of the groups that we need is done based on the locality information as well as device selection criteria. We do not discuss the grouping and formation of the groups' criteria in this paper, because it is more complicated to be discussed together with availability information together. You can find more information on this part of the task in [6].

3. Availability Formulation

To reach the best performance in a management system like one defined above, different kinds of availabilities would be important for us for different phases of action. These availabilities may seem very similar to those typically used in P2P networks, but there are some special considerations in their calculations because of their special purpose of use:

Manager availability: Availability of management peers is important for the management network when we need that specific peer to carry our special control task for us. The calculation is similar to peer availability in normal P2P networks. We name this availability in the management networks as the *manager availability*. Manager availability in an NMS system, using time-based availability formulas would simply be:

$$A_{man}(i) = A_p(i) = \frac{\text{Up Time}}{\text{Total Obsevation Time}} = \frac{\sum_{num(s)} t_s(i)}{T}$$

$t_s(i)$ is the duration of session of peer i.

In [1] presence-based availability is defined as the availability of a peer or a set of peers from the point of view of one or more observers. The authors of the paper have claimed that presence-based availability would give us a better measure of the system availability since it considers the dependency between nodes requesting and the peers possessing the data object required that eliminates the problem of their distributed up-times effect on the availability of the system.

For calculation of k-of-n availability (availability in the group of n peers that any k number of peers are sufficient to assume the system available) [1] proposes using sets of peers rather than individual ones and considers that only a subset of that set may be needed. S_k is used to denote the subset of peers including k nodes. $O_s(t)$ is the online function which is 1 when all k nodes in the S_k are online and 0 otherwise. Based on the assumptions above [1] defines the presence-based availability of peer as the 1-of-1 availability. N is the total number of machines in the P2P system; M is number of unavailable machines in time t. So we can conclude that presence-based manager availability in a system is as follows:

$$A_p(i) = \frac{\int_{t=0}^{T} O_i(t)(N - M(t) - 1)dt}{\int_{t=0}^{T} (N - M(t) - O_i(t))dt}$$

And if we use the k-of-n availability, we could gain grouped manager availability for a group of n peers out of which k are sufficient for full functionality which would lead us to:

$$A_k^n = \frac{\int_{t=0}^{T} O_k(t)(N - M(t) - n)dt}{\int_{t=0}^{T} (N - M(t) - O_k(t))dt}$$

Critical Data Availabilit*y:* Data object availability is important for us when we need to obtain certain information from the system. Since most of the time this data acquisitions may be associated with a critical status, one of the most important measurements should be done on some specific critical data objects' availability. We name this availability for management networks the *Critical Data Availability.*

Availability of a special block of data (Data Object) in peer-to-peer systems using m out of n erasure codes in a time-based calculation [2] is formulated as:

$$A(Object) = P_o = \sum_{i=0}^{n-m} \frac{\binom{M}{i}\binom{N-M}{n-i}}{\binom{N}{n}}$$

P_o is the probability that an object is available, n is total number of fragments; m is number of fragments needed for reconstruction; N is the total number of machines in the P2P system; M is number of currently unavailable machines and X is the total number of data objects in the system.

We consider that in our management network, each critical block of data is erasure coded in each level of our management system. In this system, most of the decision makings are done in the TLM layer. So when we need a critical data to make a decision, we have to obtain it from a managed object which is done in the LLM, then pass it to the MLM and then give it to the TLM. Using erasure coding at each level, availability of a critical block of data, using time-based calculations would be a multiplication of availability at each level:

$$A_{CD} = \prod_{g \in \{LLM, MLM, TLM\}} \sum_{i=0}^{n-m} \frac{\binom{M_g}{i}\binom{N_g - M_g}{n-i}}{\binom{N_g}{n}} \times \prod_{l \in \{l_1, l_2\}} A_{Link}(l)$$

X is the total number of data objects in the system.

Presence-based availability of a critical data in NMSs is quite challenging. Since we need all the information to be available to the user from the users' perspective and the user would only acquire information from the TLMs, we will need to consider the presence-based availability of the system and other parameters from the TLMs point of view. And if we use the m-of-n availability, equal to the erasure coding parameters we use for our data objects we would obtain the A_{CD} which would be:

$$A_{CD} = A_m^n = \prod_{g \in \{LLM, MLM, TLM\}} \frac{\int_{t=0}^{T} O_m(t)(N_g - M_g(t) - n)dt}{\int_{t=0}^{T} (N_g - M_g(t) - O_m(t))dt}$$

One of the main important things to consider in the availability calculations in the P2P management networks is that, in contrast with file-sharing peer to peer systems, most of the management data is unimportant to us if we could not gain them in the right time. So the way we have to use for increasing our data object availability is for sure not the replication or even erasure coding. Instead, we have to use the right way to send our data to the managers needing them to ensure the right data on right time. We have proposed an scheme for this in section three.

Procedure availability: Procedure availability is considered to be the availability of a management procedure on a specific peer which is also available. If we consider each manager peer which is up has no problem in the management software, the manager availability and the procedure availability would be the same in the network. But if we consider a peer could be up with a problem on its management software which may be silent on the time of a request and may show itself afterwards, procedure and peer availability will become two different problems. Procedure availability would be of main importance in the security investigations of network management systems.

Normal Data Availability: Despite of critical data which should be erasure coded, management data is time-sensitive, so we usually do not have enough time to backup, make replicas or erasure code them to prevent their loss. In fact, if a normal block of management data does not reach the target manager at the needed time it is totally useless and it does not need any backup.

NMS availability: Service availability, or average availability among all peers in a network, is important for us when we want to know the overall system status at a specific time. We identify it in management networks as the NMS availability. NMS availability is the average manager availability in a system. In a time-based calculation, it would be:

$$A_{NMS} = \frac{1}{N}\sum_{i=1}^{N} A_{man}(i)$$

In presence-based availability, the NMS availability would be equal to the all number of groups in the network with n peers that k of them is sufficient for full functionality:

$$A_{NMS} = A_{Sys}^{k} = \frac{(N-n)!\,n!}{N!}\sum_{\{k-of-n\}} A_k^n$$

In other words, if we consider all data on such groups, availability of all CDs in the network would result in the NMS availability: (X: number of all data objects in the system, NCD is number of Critical Data(CD) available in the system)

$$A_{NMS} = \frac{1}{X}\left(\sum_{i=1}^{NCD} A_{CD}(i) + \sum_{i=1}^{X-NCD} A_{ND}(i)\right)$$

Configuration availability: Service availability is important for us to measure how many of our data acquisitions from the network were successful. Then decide on the grouping of the system to increase the service availability. In other words, service availability gives the feedback to the management network how to reconfigure itself to achieve the best performance possible in each phase of action. We identify this availability in the management networks as the *Configuration Availability*.

Service availability in time-based calculations is formulated as:

$$A_{Service} = \frac{1}{X}\sum_{i=0}^{X} A_{Object}(i)F(i)$$

Or

$$A_{Service} = \sum_{j=k}^{N-M} \binom{N-M}{j} A_{Sys}{}^{j}(1-A_{Sys})^{N-M-j}$$

So, time-based configuration availability could be defined as:

$$A_{Conf} = \frac{1}{X}\sum_{i=0}^{X} A_{CD}(i)F_{\Pr ocedure}(i)$$

But the difference with the system availability in the normal P2P network is in that every CD is a new one in the P2P management network so we cannot compute the frequency of access to that exact item; instead we use the frequency of access to that procedure which gives us the CD. We can also use the formula below for configuration availability:

$$A_{Conf} = \prod_{g\in\{LLM,MLM,TLM\}} \sum_{j=k}^{Ng-Mg} \binom{N_g - M_g}{j} \times A_{NMS}{}^{j}(1-A_{NMS})^{Ng-Mg-j} \times \sum_{l\in\{l_1,l_2,l_3\}} A_{Link}(l)$$

And the presence-based configuration availability would be:

$$A_{Conf} = \prod_{g\in\{LLM,MLM,TLM\}} \sum_{i=1}^{X} \frac{(N_g - n)!\,n!}{N\,!} A_{CD}$$

Link Availability: We also need the *Link Availability* in the management network to be the availability of a link between an upper level management entity and its descendant managers to acquire their data. The link is not a physical network link between the peers; instead it represents an overlay network link among the peers. The link availability in overlay network are considered to be high due to the characteristics of such networks, but in the event that grouping avoids us from achieving a peer form its descendants, it may come to help us.

4. Proposed Algorithm

In this section we propose an algorithm to increase the data availability in P2P management networks avoiding data replicas and erasure coding.

So our problem is how to measure the availability of a normal block of data in P2P management networks and what should we do to increase this availability.

Our algorithm begins working after the groups is been formed. The code executes on the representative of each of the groups. The algorithm monitors all the peers available in the P2P management network group. It maintains a list of the peers available in that specific group. It runs a monitoring algorithm on all peers on a periodic fashion. For monitoring a kind of ping operation is done in the application level. Then it increases the score it maintains for each of the nodes it has in its list. The list is sorted based on the availability score calculated for each node and the most available peer on the top of the list is nominated for the backup representative node. After the representative node is been selected and we have a backup of all we need on backup representative, the same algorithm is began to run on the backup representative and after each period of monitoring the representative and the backup representative began to exchange information.

When they exchange information on the list of representative and backup representative, if other two nodes be on the top of this list, they will change their place with representative and backup by copying all needed information and algorithm code into them. The simple pseudo code of the algorithm is shown in Fig. 1.

```
While (representative==potential representative)
On (timeout)
{
       For (i=0; i<number of nodes in the group; i++)
      {
            If the node is available
                        Availability [peer[i]] ++;
            Else
                        Availability [peer[i]] --;
       }
       Sort the list based on Availability

       Exchange list info with backup representative

       Select the first node on the Availability list as the
       potential representative

       If (Availability [representative]>= Availability
       [potential representative])

       Potential representative= representative;
}
Copy List and algorithm code + representative
information to the new representative
```

Figure 1: Pseudo code for the proposed algorithm

5. Numerical Study

For the correctness and usefulness of what we have presented in this paper, we conducted a handful of numerical analysis using MatLab. We considered the critical data availability for the first tests we run. With different numbers of m and n between 0 and 16 we found out the relevancy between the erasure coding parameter and availability for both time-based and presence-based calculations. Amazingly, time-based calculations gave us more analytical and easier to understand numbers in results. As shown in Fig. 1, time-based curve analysis shows us that the best availability could be gained when n is bigger than 10 and m is less than 8 we would gain a good availability of the system which in minimal value is equal to 99.01% when 40% of the total machines in the system are down.

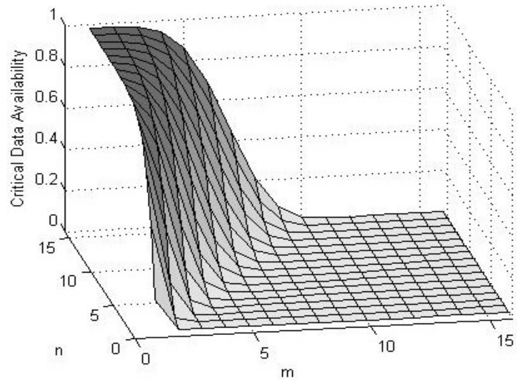

Figure 2: Time-based Availability for Critical data, with n and m varying between 0 and 16.

On the other hand, according to Fig. 3 ,on the same configuration, we could just gain 89.27% availability in the presence-based availability calculations, when n was 4 and m was 1 which means full redundancy. The reason for this result is that we considered that n in the m/n erasure coding is equal to the n in k-of-n presence based availability would gain us the critical data which is erasure coded on n machines.

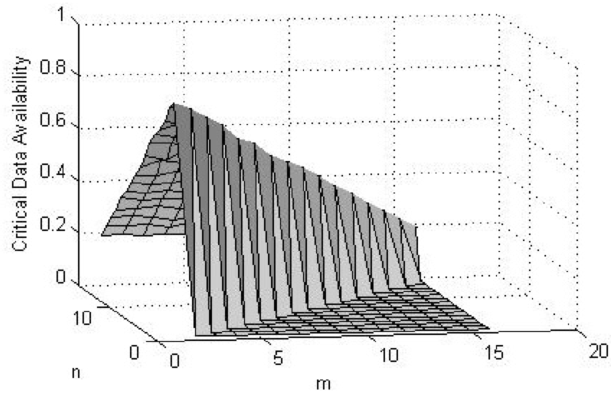

Figure 3: Presence-based Availability for Critical data, with n and m varying between 0 and 16.

Lets consider the results would be used on two different phases of the system, first, when we are to design our management system and second, when we are to manage our system having the data. Now consider the results we'd taken from our numerical analysis. Since our numerical analysis on the time-based calculation is increasing in the point that we have a full 15 redundancies, it shows us there is no dependency between the absence of machines and the erasure coding parameters, but as we get the best results in the presence-based availability on m less than 4 and our m in erasure coding is also the m used in grouping to obtain the online function. The difference in the two cases is that the presence-based availability obliges all m machines to be online to get the availability required but time-based gives us no info that those machines that are available, carry the info we want or not. It seems that best combination of the calculations to be used is to use time-based availabilities in design, which gives us better numerical for design of the system and when we make our system online, use the presence-based availabilities to acquire more accurate information about our system.

Fig. 4 shows time-based and presence-based NMS availabilities. As can be seen in the picture, time-based NMS availability is just a average of peer availabilities in the system. But presence-based availability considers manager communication. So, in lower percentages of the peer availability the presence-based availability falls because of the lower communication probability among peers in different layers.

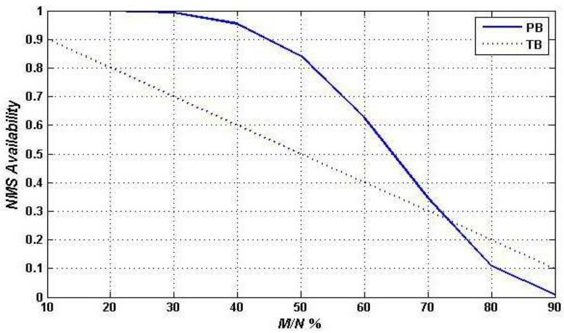

Figure 4: NMS availability

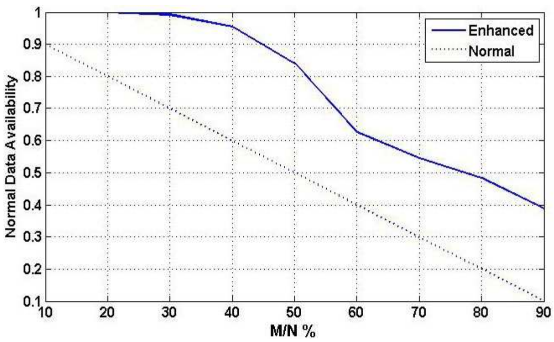

Figure 5: Normal Data Availability with and without the proposed enhancement algorithm

Fig. 5 shows normal data availability before and after running algorithm in our network. N is considered to be 1000 in this simulation. Normal data availability is equal to NMS availability when algorithm is not run on the system, because it is totally dependant on peer availability, so when a peer goes down all data associated with it is lost. When algorithm is run, our list data and representatives data is preserved, so we see an enhancement in the normal data availability.

6. Conclusion and Future Work

In this paper we presented a framework and a numerical analysis for availability on peer to peer management networks. First we defined the different entity availabilities which are important for us when we study such networks. Then we presented formulas for the calculation of such availabilities.

We conclude that time-based availability calculations would be more helpful when one is to design a management network, but while the system is online and working, presence-based availability would help to estimate the status of the network and make management decisions.

For the future work, we propose the numerical analysis of all availability calculations to the parameters they have to conclude an overall conclusion on the design of a network management system.

7. References

[1] R. J. Dunn, J. Zahorjan, S. D. Gribble, H. M. Levy, "Presence-Based Availability and P2P Systems", Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing, 2005.

[2] H. Weatherspoon, J. D. Kubiatowicz, "*Erasure Coding vs. Replication: A Quantitative Comparison*", Lecture Notes In Computer Science; Vol. 2429, Pages: pp: 328 – 338, 2002.

[3] S. Prabhu , R. Venkat, "High Availability for Network Management Applications", IEEE Proceedings on Second International Conference on Availability, Reliability and Security, 2007.

[4] G. Da Costa, C. Marchand, O. Richard, J-M Vincent, "Resource Availability for P2P Systems", IEEE Proceedings on the 20th International Conference on Advanced Information Networking and Applications, 2006.

[5] M.K. Ramanathan, "Increasing Object Availability in P2P Systems", IEEE Proceedings of the 18th International Parallel and Distributed Processing Symposium (IPDPS'04), 2004

[6] J. Chu, K. Labonte, B. N. Levin, "Availability and Locality Measurements of P2P File Systems", Proceedings of SPIE, vol. 4868, pp. 310--321, 2002.

[7] A. Panisson, D. M da Rosa, C. Melchiors, L.Z Granville, M. J. B. Almeida, L. M. R. Tarouco, "Designing the Architecture of P2P-Based Network Management Systems", Proceedings of the 11th IEEE Symposium on Computers and Communications, 2006.

[8] L. Z. Granville, D. M. da Rosa, A. Panisson, C. Melchiors, M. J. B. Almeida, and L. M. R. Tarouco, "Managing Computer Networks Using Peer-to-Peer Technologies", IEEE Communications Magazine, 2005.

[9] D. Figueiredo, J. Shapiro, D. Towsley, "Incentives to Promote Availability in Peer-to-Peer Anonymity Systems", Proceedings of IEEE ICNP, 2005.

[10] J. Tian, Z. Yang,Y. Dai, "A Data Placement Scheme with Time-Related Model for P2P Storages", Proceedings of IEEE P2P, 2007

[11] P. Knezevic, "Managing and Recovering High Data Availability in a DHT under Churn", IEEE Proceedings on CollaborateCom 2006.