Chapter 6

Replication

6.1 INTRODUCTION

This chapter deals with a detailed procedure of replication strategy and a reliability model. Initially, an introductory note is given on replication and later on the survey of the related replication strategies is given. In this chapter, we have proposed an availability model and fault recovery process. Further, the investigation on the case studies under normal fault exceptions is explored. Finally, we infer the results using simulation model.

Replication is the germination of any reliable p2pVoD system design. By replicating videos across multiple peers, p2pVoD applications can tolerate the loss of individual peer failures (Cohen and Shenker, 2002). This might achieve some level of reliability in the p2pVoD system, where most of the peers are unreliable (Golubchik et al, 2001), (Jack and Leung, 2002), (Yang et al, 2003), (Do et al, 2004). As the scope and scale of such system continues to grow, it is important to develop comprehensive design principles for the problems that are of fundamental importance to reliability engineering (Ranganathan et al, 2002), (Tokekar et al, 2005), (Poon et al, 2005). In this chapter, we have addressed one such problem of reliability engineering for our p2pVoD system. The problem is to maintain a reliable state of movies in our p2pVoD architecture, where the lifetime of the reliable state must substantially exceed the lifetime of the individual peers in the p2pVoD system.

In our p2pVoD architecture, replication is a crucial service because the individual peers are unreliable when compared to the dedicated streaming server. Moreover, the growth of the chain is directly proportional to the growth of the number of peers in the p2pVoD system. At any instant, the reliable state must be recovered not only from a single peer failure in the chain, but also from multiple simultaneous failures. This motivated us to study the reliable state problem for building highly available movies in the p2pVoD system. The replication technique is a key mechanism required to build such a reliable state of the movies. The peers in our p2pVoD architecture are likely to fail under the following circumstances.

1) The participating peer ceases without prior intimation (disappearance).
2) The behavior of the peer is unpredictable due to irregular joining of peers and inadvertently leaving the system (peer churn).
3) Higher peer churn rate leads to lower availability of the movies and frequent service interruption in the chain (incoherence).
4) An unexpected comeback of the participating peer in the midst of the ongoing chaining session (reentrance).
5) Participating peer permanently ceases their participation because of the software or hardware failure, or network communication failures etc. (malfunction).

In any of the above circumstances, the peer is presently not participating in the chain and the reliable state for that peer is transient. The p2pVoD system must ensure that the reliable state is recovered at least from other participating peers. To achieve appropriate reliable state of movies in the facade of dynamic peer churns is one of the major challenges of our p2pVoD architecture.

To encounter this challenge, we propose a new replication technique called M-Replication, wherein the word Movie is abbreviated as M. The proposed M-Replication technique ensures that the reliable state of the movies is maintained against individual or multiple failures of the peers. To maintain the long-term availability of the movies in the chain, M-Replication technique creates functional replicas along the ongoing video chaining session and non-functional replicas around the auxiliary video chaining session. In order to maintain the reliable state of the movies in the chain, M-Replication reactively repairs the lost replicas along and around the chains. Before we explicate about our proposed M-Replication technique, let us review some of the existing replication techniques that suits well for our p2pVoD architecture.

6.2 REPLICATION TECHNIQUES

In general, any replication technique must consider the following important problems.

1) Number of Replicas Problem.
2) Replica Placement Problem.
3) Update Control Problem.

The phrase “Number of Replicas Problem” indicates how many number of replicas are required for a movie to maximize its’ reliable state. The phrase “Replica Placement Problem” indicates in what proximity the replicas must be placed. The
phrase “Update Control Problem” indicates how much occurrences of control messages are required to repair the lost replicas during peer failures. Based on the above problems, four related replication techniques are reviewed in the next sub sections. For brevity, the above phrases are merged in the following sub sections, which suffice the knowledge of the replication technique without losing its intrinsic characteristics.

6.2.1 DYNAMIC RESOURCE REPLICATION
Rong (2008) has proposed a dynamic resource replication strategy. In this replication strategy, an active approach is taken by using video request rate as the primary metric to trigger the replication process. This process adaptively replicates videos into various peers based on the configurable properties. These configurable properties are buffer, bandwidth and user preferences. Thereby, the video request rate is calculated using the number of requests received for a specific video, which is divided by the elapsed time. This data is stored in a table that contains the total number of requests and their request rate of the videos. Whenever the total number of request for a video reaches a certain predefined threshold, the replication process is triggered for that specific video. In this replication process, the total number of replicas generated is the product of the video request rate and the aggressive factor. The latter is the indicator function used to generate greater number of replicas. The problem to generate number of replicas is easily solvable in this replication technique. But, the problem of placement and update control is more complicated because of peer churns.

6.2.2 SQUARE ROOT REPLICATION
Edith and Shenker (2002) have proposed a square root replication strategy. In this replication strategy, the number of replicas is calculated based on the square root of the request rate ratio and the popularity of the requested movie. During peer churns, the replicas are generated dynamically and placed proximately depending on the request rate ratio. Hence, the problem to generate number of replicas and the placement of replicas are done effectively. Though the availability level of the movies is high, the overhead caused during update control are at stake.

6.2.3 PROACTIVE REPLICATION
Sit et al (2006) have a proactive replication strategy. In this replication strategy, the generation of replicas is irrespective of the peer churns. The replicas are not generated in response to the failures, but periodically created at a fixed low rate. In this replication strategy, a Tempo is created to proactively maintain the replicas. Each peer
in Tempo operates under a bandwidth budget, which is specified by the peer’s user. A budget is a configurable bandwidth for the measurement of the required number of replicas. Therein, the peers cooperate and attempt to maximize the availability of movies by constantly creating new replicas at a fixed rate within their individual budget limitations. Tempo efficiently adjusts the replication level during the peer churns subjected to the bandwidth budget constraints. This ensures that each peer’s bandwidth usage for the proactive maintenance is partially predictable. On a whole, the number of replicas are generated based on the Tempo and the placement of these replicas are done proximately. But, due to the dynamicity of the peer churn rate, this prediction may diverge from the reality. Hence, Tempo cannot be previsioned. Moreover, the update control problem directly depends on the users’ cooperation.

6.2.4 OPTIMAL REPLICATION

Kangasharju et al. (2006) have proposed an optimal replication strategy. In this replication strategy, the optimal number of replicas are generated for each of the movie based on the video request rate and peer availability distribution. The popularity of the movie is known a priori. The peer availability probability is also known a priori. In order to achieve high hit probability, only the popular movies are replicated across the available peers. The degree of replication herewith is the function of the movie’s popularity. The value for the number of replicas is derived from a logarithmic replication rule, which is optimal. The placement of the replicas is fairly tight on the upper bound limit of the peer availability distribution. However, due to peer churns the peer availability distribution can fluctuate impulsively. Thereafter, the update control problem radically exchanges control message for recovery. Hence, this will create a major bandwidth clog in the network traffic.

Henceforth, we have addressed every important aspect that is related to the problems of the replication in our proposed M-Replication technique.

6.3 M-REPLICATION

First, we foresight the availability model for the movies, so that the replication strategies can be designed profoundly. Next, replication strategy is developed which is practical and tractable. The objective herein is to develop a replication strategy that can create optimal number of replicas in the dynamicity of impulsive peer churns. The replication strategy must also resolve the problem of generating an optimal number of replicas. The placement problem is instigated by a fault tolerant policy for faster
recovery. Finally, the problem of update control messages is regulated using recovery control message. Next, let us expound the M-Replication technique in the following sub sections.

6.3.1 AVAILABILITY MODEL
A movie is said to be available, when its video blocks are reachable within the proximity. A movie is said to be highly available, when its video blocks are reachable by certain level of redundancy. Eventually, support for the fault tolerance mechanism is needed. Therefore, the availability in our M-Replication technique is defined as the probability of contributing peer’s reachability by certain level of redundancy, wherein each peer has an independent and identical failure probability. Indeed, the perceived QoS of the receiving peers are much influenced by the availability level of movies. To define the availability level of a movie, we need to know the peer availability distribution $H(\cdot)$.

Now, let us derive an expression for the peer availability distribution. Let $P_{up}$ denote the duration of peer uptime and $P_{dn}$ denote the duration of peer downtime. In general, the peer availability $A_{up}$ is defined as

$$A_{up} = \frac{P_{up}}{P_{up} + P_{dn}}$$

and the peer unavailability $A_{dn}$ is defined as

$$A_{dn} = \frac{P_{dn}}{P_{up} + P_{dn}}$$

Therefore, the probability of peer availability distribution is defined as the exponential distribution with mean peer availability $x$ as

$$H(A_{up} > A_{dn}) = 1 - e^x$$

Thereby, the M-Replication strategy uses the peer availability distribution $H(A_{up} > A_{dn})$ to maximize the availability level of the movie.

6.3.2 M-REPLICATION STRATEGY
During chaining, if the chain belongs to the current streaming session then it is called an ongoing video chaining session. If the chain belongs to the supplementary session of an outgoing video chaining session then it is called an auxiliary video chaining session. We assume that peer failures in the chain are independent and identically distributed. This allows replicas to be placed orderly in peers along the ongoing video chaining session and around the auxiliary chaining session. In practice, the
independent and identically distributed assumption is realistic, which will be discussed in Section 6.3.4. For our p2pVoD architecture, we define a replication strategy to maximize the availability level of the movie. In this replication strategy, a replica is defined as a redundant video block that can be stored in the inward peer of the ongoing video chaining session or in the outward peer of the auxiliary video chaining session. The redundant video blocks that are placed in the inward peers along the ongoing video chaining session are called as functional replicas. The redundant video blocks that are placed in the outward peers along the auxiliary chaining session are called as non-functional replicas. The functional replicas are the copy of the chaining bridge image (Backward/Residual buffer) which had previously stored in the inward peers along an ongoing chaining session. The non-functional replicas are the copy of the chaining bridge image (Backward/Residual buffer) of an ongoing video chaining session, which is stored in the outward peers along the auxiliary chaining sessions. Note that, a single functional replica is stored in an ongoing video chaining session and more than one copy is stored in the auxiliary video chaining sessions.

The problem here is to estimate optimal number of replicas that should be placed in the peers along an ongoing video chaining session and around the auxiliary video chaining sessions.

In M-Replication strategy, the optimal solution is a combinatorial function of per peer availability and the number of replicas per popular movie, which can produce a mapping between individual replicas and individual peers. To calculate the optimal number of replicas, we need to know the peer availability distribution \( H(A_{up} > A_{dn}) \).

Now, we can randomly choose \( y \) blocks of replicas such that the probability \( Pr \) can be estimated to ensure that sufficient replicas are available at the time when video blocks are accessed. To estimate the probability \( Pr \), \( h \) chaining sessions are randomly picked around the auxiliary chaining sessions inclusive of an ongoing video chaining session. The probability of choosing exactly \( y \) chaining sessions follows a binominal distribution with mean peer distribution availability. This implies that any chaining session chosen randomly has a mean availability of \( A_{up} \) and variance of \( A_{up} (1 - A_{up}) \). Therefore, if \( Y \) is a random variable for the number of chaining sessions available then the probability of actual number of chaining session available is given by
We characterize the available level of the movie for our replication strategy. The objective here is to optimize the number of replicas for maintaining a desired level of video blocks availability and to calculate the replication factor $r$ for a movie. Suppose, we assume $r$ copies of replicas must be placed in different chaining sessions then at least one of those $y$ chaining session must be available to recover the video blocks. Using Equation 6.1, the probability that one or more chaining session available is

$$\Pr(Y = 0) = (1 - A_{up})^h$$

Hence,

$$\Pr(Y \geq 1) = 1 - (1 - A_{up})^h$$

Since, $Y$ follows a binominal distribution, the availability level of a movie is defined as the probability $\mathcal{A}$ that has one or more available chaining sessions as

$$\mathcal{A} = \Pr(Y \geq 1) = 1 - (1 - A_{up})^h$$

$$1 - \mathcal{A} = (1 - A_{up})^h$$

$$\log(1 - \mathcal{A}) = h \cdot \log(1 - A_{up})$$

$$h = \frac{\log(1 - \mathcal{A})}{\log(1 - A_{up})}$$

(6.2)

Thereby, the value of $h$ chaining sessions is assigned to the value of replication factor $r$. The optimal estimation of the replication factor $r$ can be validated with a simple example. Suppose, if we want the highest availability level of the movie (0.999) with mean peer availability of 0.5 then according to the Equation 6.2, the replication factor $r$ is 10. In this case, the functional replica is 1, which creates a copy of its own chaining peer’s buffer image along the ongoing video chaining session. The copies of remaining 9 non-functional replicas are the copy of the chaining peer’s buffer image of the functional replica, which is streamed from the video source to the auxiliary video chaining sessions. Similarly, if we want a lower availability level of the movie (0.995, 0.95, 0.9, and 0.8) with mean peer availability of 0.5 then according to the Equation 6.2, the replication factor $r$ is 8, 5, 3, and 2 respectively.

6.3.3 STATE INFORMATION TABLE

Whenever the playback of the movie has been completed, it is simply assumed that peer gracefully leaves the ongoing video chaining session. However, this assumption is frail because of the following circumstances; peer disappearances, peer churns,
incoherencies, reentrances and malfunctions. Therefore, we investigate in totality these circumstances for faster recovery.

The problem of peer churn can be resolved by using a state information table. In a video chaining session or in the auxiliary video chaining sessions, a peer must send the State Information (SI) table along the chaining session periodically. The SI table contains peer identifiers and its mean availability, such that the succeeding peer that follows a preceding peer will have their ascendants’ state information. The interval to update SI can be either a synchronized self-clock or triggered upon an arrival of a modified SI from the preceding peers. Whenever, a peer receives a modified SI from the preceding peers then it updates its SI table. In an ongoing chaining session, suppose, if the preceding peer do not respond then after the expiry of the countdown timer of the succeeding peer, and based on its SI table, it determines to which ascendant to connect for continuing in the video chaining session.

If a peer wants to leave an ongoing video chaining session in the midst then a “leave” message will be sent to all its descendants. Thereafter, the SI tuple of the departed peer will be removed from the SI table of all its descendants in the chain. However, if a peer fails in the midst then after the expiry of the countdown timer the corresponding peer’s SI tuple will be removed from the SI table of all its descendants. Whenever, a SI tuple is removed from the SI table of a peer then the corresponding peer of the descendants will launch a fault recovery process to its ascendants along the chaining session. After a successful fault recovery process, the chaining session resumes into normal deportment. Suppose, if there is no ascendants to recover from the fault along the chaining session then the recovery information is sent to the video source. Thereby, the video source recovers from the fault by persuading an auxiliary video chaining session for the ongoing video chaining session.

6.3.4 FAULT RECOVERY PROCESS
Let us understand the fault recovery process with a simple illustration for clarity. Let us consider at least 5 peers in each of the chaining session. Each of these chaining sessions corresponds to cardinality of one-to-one mapping of movies. The Backward/Residual bridge specified for a peer in the chaining session is same as M-Chaining technique that was discussed in Chapter 5. Henceforth, the placements of replicas are done using M-Chaining technique. Each of the preceding peers sends its SI to the succeeding peers periodically. Let us consider at least three movies in this
illustration. Let $M_1, M_2, M_3$ denote the respective movies. Thereafter, there will be only three unique chaining sessions.

Let $C_1, C_2, C_3$ denote the respective chaining sessions for the corresponding movie $M_1, M_2, M_3$. Let us consider movie $M_1$ for the sake of this illustration. Let us presume that the desired availability level of movie $M_1$ be 0.9 with mean peer availability of 0.5. According to the Equation 6.2, the replication factor $r$ is 3. Therefore, one functional replica must be stored in video session $C_1^1$ along with the ongoing video chaining session $C_1$, and 2 non-functional replicas must be stored in auxiliary video chaining sessions $C_1^2$ and $C_1^3$ along with the ongoing video chaining sessions of $C_2$ and $C_3$ respectively. Note that, the replication factor will not be affected by the variations in the peer availability distribution as long as the mean remains the same along the chaining session.

![Figure 6.1: Snapshot of the Ongoing Chaining Sessions at Some Moment.](image)

Let us guesstimate the snapshot of the ongoing chaining sessions at some point of streaming timeline as shown in the Figure 6.1. In this illustration, it can be precisely visualized the flow of video blocks along the video chaining session and around the auxiliary video chaining sessions. It can also be observed that each of the peers along the chaining session sends its SI to the descendants. For better insight of the problem, the chaining bridges of these peers are shown in Figure 6.2. Note that, video source herein can be the streaming server or super peer or a normal peer, which apparently depends on the context. Each of the ongoing video chaining session has the video block numbers of their respective movies, which are represented as single-lined box in the figure. The shaded box represents the current playback video block in that
The replicas are represented in double-lined box. The ellipsis represents the unused chaining buffers. Note that, the video block numbers of functional replicas are stored along the ongoing video chaining session \( C1 \) and, the video block numbers of non-functional replicas are stored along the ongoing video chaining session \( C2 \) and \( C3 \).

![Figure 6.2: Video Block Numbers those are stored in Peers along the Chaining Session.](image)
Let us understand the following case studies under normal fault exceptions.

a) Suppose Peer2 departures normally in the midst of the ongoing video chaining session \( C_1 \). Before Peer2 departures, it should send a “leave” message to all its descendants (Peer3, Peer4, and Peer5) along the chaining session \( C_1 \). Upon arrival of the message, the descendants should remove Peer2’s tuple entry in its SI table. In this illustration, Peer3 was a succeeding peer and it was following the preceding Peer2 for the video blocks. Now, Peer3 will initiate a control message to Peer1 after searching for the predecessor of the preceding Peer2 as shown in Figure 6.3(a). In this case, the control message will contain the next playback video block number 88 (shown in Figure 6.2). According to the status quo of the replicated video block numbers stored in Peer1, the required video block number 88 is found in Peer1. Thereon, the Peer1 starts streaming...
video block number 88 onwards to Peer3 as shown in the Figure 6.3(b). Note that, in the event of reordering of peers in the chaining session, the group fingerprint-IDS in the video blocks are not affected. However, there is pricing negotiations between the receiving peer and the sending peer. This remark is taken for the current case as well as for the following case studies.

b) Suppose, if both Peer3 and Peer4 departures unexpectedly without any notice. Wherein, Peer5 was a succeeding peer and it was following the preceding Peer4 for the video blocks. After the expiry of the countdown timer, Peer5 initiates a control message to Peer3, after searching for the predecessor of the preceding Peer4. Obviously, in this case Peer3 was also departed earlier without any notice. Again, Peer5 initiates another control message to Peer1 with the required video block number 77 (shown in Figure 6.2) as shown in the Figure 6.4. According to the status quo of the replicated video block numbers stored in Peer1, the required video block number 77 is not found in Peer1. Therefore, the request from Peer5 is rejected by Peer1 courteously. Eventually, Peer5 will send the control message to the video source for fault recovery.

![Figure 6.4: Recovery Process of Peer5 after Failures of Peer3 and Peer4.](image-url)

c) Suppose, if the video source receives a control message to recover from the fault from any of the peers in an ongoing video chaining sessions. Depending on the required video block number, a non-functional replica for that block is chosen. In this illustration, for the second case study, the choice can be between Peer8 and Peer13 along the auxiliary video chaining sessions $C_1^2$ and $C_1^3$ respectively. Since, the required video block number 77 is found both in
Peer8 and Peer13 (shown in Figure 6.2). Note that the selection of the preceding peer should not increase the network traffic rate or should not overload the peer, or both. Because, in practice many peers might be selected for fault recovery. As per this analogy, suppose if Peer8 is selected as the preceding peer for Peer5 then Peer8 starts streaming video block number 77 onwards to Peer5, thereby recovering from the fault as shown in Figure 6.5.

Figure 6.5: Recovery Process of Peer5 with Peer8.

Figure 6.6: Exemplified Recovery Process of Peer14 with Peer12.

d) Suppose, if any of the peers along the ongoing video chaining sessions \( C1, C2, C3 \) fails then the analogy of the case studies (a), (b), and (c) can be
induced to recover from the fault. An instance of such failure of Peer13 and recovery of Peer14 is exemplified in Figure 6.6.

6.4 RELIABILITY MODEL

Two important parameters are idealized in M-Replication technique to ensure long-term availability level of the movie. The parameters are number of video chaining sessions and the repair rate. A repair mechanism here is accomplished by probing a Repair-Control Message (RM) along the ongoing video chaining sessions for retaining the previous state of the failed chain. A chain can be repaired upon any of the following failures such as disappearance, churn, incoherence, reentrant, and malfunction. Therefore, a steady state of the video chaining session must be maintained for the long-term availability of the movie. Hence, the metric lifetime of the state is used for optimizing the chaining sessions. To optimize the chaining sessions, the parameters must be tuned to maximize the lifetime of the state. To maximize the lifetime of the state we must resolve the following computations.

1) Calculation of lifetime of a state on certain degree of replication and certain rate of repair.

2) Calculation of resource constraints for maintaining the steady state.

3) Tuning the values based on the above calculations.

6.4.1 BASIC MODEL

The video chaining session consists of some partisan number of peers on which the replicas are stored. These peers participate in a video chaining session for the duration of $A_{up}$. We assume that $A_{up}$ is an exponential distribution with mean departure of $\frac{1}{\lambda}$.

Thus the probability of the peer availability is defined as

$$Pr\{A_{up} > t\} = e^{-\lambda t}$$

Note that, $A_{up}$ is an independent and identically distributed for all peers along the video chaining session. Recall the definition of reliable state from the Section 6.1 that the lifetime of the reliable state must substantially exceed the lifetime of the individual peers in the p2pVoD system. Any reliable state replicated on a peer is available for the duration of its participation along the video chaining session. When a peer departs without any notice, we assume that its state is lost. For the durability of the video chaining session, we need to maintain certain reliable state (S) for that
video chaining session. Therefore, this state must be replicated on multiple peers along the video chaining session.

Let us consider a state for the number of chains \( S \) be \( r \), where the value of \( r \) is determined using Equation 6.2. Over the period of time, many peers might depart from the video chaining session, which can decrease the number of replicas present in the peers. To balance the state, the video chaining session must be repaired by creating new replicas to account for the lost replicas. Therefore, the repair mechanism must first detect the lost replicas and then it must create new replicas by copying the redundant lost replicas to another peer along the video chaining session. Let \( A_{dn} \) denote the time duration for the repair mechanism. We assume that \( A_{dn} \) is an exponentially distributed with the mean repair rate of \( \frac{1}{\mu} \). Thus, the probability of repair rate is defined as

\[
Pr\{A_{dn} > t\} = e^{-\mu t}
\]

The normalized repair rate \( \gamma \) is defined as the ratio between the rate of departure and the rate of repair. Therefore, the repair ratio is defined as

\[
\gamma = \frac{\mu}{\lambda}
\]

The normalized repair rate represents the balance between the rate of losing replicas and the rate of creating new replicas. Note that, the use of exponential model for both peer participation and replica repair is a common practice in the reliability engineering. Since, the memoryless property of the exponential distribution follows a Markov Process hence the reliable state of M-Replication is mathematically tractable.

### 6.4.2 MARKOV CHAIN

The above basic model is reduced to a Markov Chain. During repair mechanism, the reliable state has \( k \) number of video chaining sessions. The remaining \( (r - k) \) video chaining sessions are being repaired. For maintaining the availability level of a movie in the Markov Chain, there can be \((r + 1)\) possible states. If there are \( k \) number of video chaining sessions then the state is in \( k \). In state \( k \), any one of the \( k \) video chains can fail, such that the transition occurs to state \((k - 1)\), or one of the remaining \((r - k)\) video chains being repaired, such that the transition occurs to state \((k + 1)\). This is a Continuous Time Markov Chain (CTMC), which is depicted pictorially in Figure 6.7, and its possible transition probabilities are functions of \( \lambda, \mu, k \) as follows.
Figure 6.7: CTMC representing the Reliable State of the Video Chaining Session.

For state 1

\[ P_{1,0} = 1 \]
\[ P_{1,2} = \frac{(r - 1)\mu}{\lambda + (r - 1)\mu} \]

For state \( k \)

\[ P_{k,k-1} = \frac{k\lambda}{k\lambda + (r - k)\mu} \]
\[ P_{k,k+1} = \frac{(r - k)\mu}{k\lambda + (r - k)\mu} \]

For state \( r - 1 \)

\[ P_{r-1,r-2} = \frac{(r - 1)\lambda}{(r - 1)\lambda + \mu} \]
\[ P_{r-1,r} = \frac{\lambda}{(r - 1)\lambda + \mu} \]

For state \( r \)

\[ P_{r,r-1} = 0 \]

Note that, state 0 is an absorbing state beyond which the recovery is impossible.

Now, we derive an expression to calculate the expected lifetime of the reliable state \( S \). Let \( t_s \) denote the expected lifetime of the reliable state which is actually the average time it takes to reach state 0, starting from state \( r \). We measure this by using the concept of an \textit{epoch}. An epoch herein is defined as a particular period of time of noteworthy. At the start of the epoch, the state is in \( r \). At the end of the epoch, the state either reaches to 0 or returns to \( r \) by starting a new epoch. The video chaining session goes through number of epochs and in each epoch it returns to state \( r \), but not the last epoch. In last epoch, the video chaining session goes to state 0. Formally, the expected lifetime of the video chaining session \( t_s \) is defined as the product of expected number of epoch \( n_e \) and the expected duration \( t_e \) in each epoch. Note that in
each epoch, starting in state \(r\), upon the failure, it makes a transition to \((r - 1)\). The following derivation proceeds in two steps for the calculation of the expected lifetime of the video chaining session.

6.4.3 NUMBER OF EPOCHS

Let \(Q_k\) be the probability that the state of the video chaining session reaches 0, before state \(r\), starting in state \(k\). From the transition probability, it clearly indicates that \(Q_0 = P_{1,0} = 1\) and \(Q_r = P_{r,r-1} = 0\).

For \(0 < k < r\), \(Q_k\) satisfies the following recurrence equation

\[
Q_k = \frac{k\lambda}{k\lambda + (r-k)\mu} Q_{k-1} + \frac{(r-k)\mu}{k\lambda + (r-k)\mu} Q_{k+1}
\]

\[
Q_k = P_{k,k-1} Q_{k-1} + P_{k,k+1} Q_{k+1}
\]

Applying \(P_{k,k-1} + P_{k,k+1} = 1\) and proceeding, we get

\[
(P_{k,k-1} + P_{k,k+1})Q_k = P_{k,k-1} Q_{k-1} + P_{k,k+1} Q_{k+1}
\]

\[
(Q_{k-1} - Q_k) = \frac{P_{k,k+1}}{P_{k,k-1}} (Q_k - Q_{k-1})
\]

Now, solving \(P_{k,k-1}\) and \(P_{k,k+1}\) yields to

\[
(Q_{k-1} - Q_k) = \frac{(r-k)}{k} \gamma^2 (Q_k - Q_{k-1})
\]

where \(\gamma = \frac{\mu}{\lambda}\)

Let \(Q^*\) denote the probability that the video chaining session reaches state 0, before state \(r\), starting in state \((r - 1)\). Repeatedly applying the recurrence Equation 6.3, we obtain

\[
(Q_{r-1} - Q_r) = Q^*
\]

\[
(Q_{r-2} - Q_{r-1}) = \frac{1}{(r-1)} \gamma (Q_{r-1} - Q_r)
\]

\[
= \frac{1}{(r-1)} \gamma Q^*
\]

\[
(Q_{r-3} - Q_{r-2}) = \frac{2}{(r-2)} \gamma (Q_{r-2} - Q_{r-1})
\]

\[
= \frac{2}{r-2} \gamma \left( \frac{1}{(r-1)} \gamma Q^* \right)
\]

\[
= \frac{1}{(r-1)^2} \gamma^2 Q^*
\]

Then proceeding inductively,
where $Q_0 - Q_r = 1 - 0 = 1$

$$1 = \sum_{k=0}^{k=r-1} (Q_{r-k-1} - Q_{r-k})$$

$$1 = \sum_{k=0}^{k=r-1} \left( \frac{1}{(r-1)} \gamma^k Q^* \right)$$

$$1 = Q^* \sum_{k=0}^{k=r-1} \left( \frac{1}{(r-1)} \gamma^k \right)$$

$$Q^* = \frac{1}{\sum_{k=0}^{k=r-1} \left( \frac{1}{(r-1)} \gamma^k \right)} \quad (6.4)$$

Therefore, the expected expression for the number of epochs is

$$n_e = \frac{1}{Q^*} \quad (6.5)$$

### 6.4.4 DURATION OF EPOCH

Let $T_k$ be the expected time that elapses before the state of the video chaining session reaches to either state 0 or state $r$, starting in state $k$. In state 0 or state $r$ the expected time will be 0. Therefore, $T_0 = 0$ and $T_r = 0$.

For $0 < k < r$, $T_k$ satisfies the following recurrence equation

$$T_k = \frac{k\lambda}{k\lambda + (r-k)\mu} T_{k-1} + \frac{(r-k)\mu}{k\lambda + (r-k)\mu} T_{k+1} + \frac{1}{k\lambda + (r-k)\mu}$$

$$T_k = P_{k,k-1} T_{k-1} + P_{k,k+1} T_{k+1} + t_k$$

Applying $P_{k,k-1} + P_{k,k+1} = 1$ and proceeding, we get

$$(P_{k,k-1} + P_{k,k+1}) T_k = P_{k,k-1} T_{k-1} + P_{k,k+1} T_{k+1} + t_k$$

$$(T_{k-1} - T_k) = \frac{P_{k,k+1}}{P_{k,k-1}} (T_k - T_{k+1}) - \frac{t_k}{P_{k,k-1}}$$

Now, solving $P_{k,k-1}, P_{k,k+1}$ and $t_k$ yields to

$$T_{k-1} - T_k = \frac{r-k}{k\gamma} (T_k - T_{k+1}) - \frac{1}{k\lambda} \quad (6.6)$$

where $\gamma = \frac{\mu}{\lambda}$.
Let $T^*$ denote the expected time that elapses before the state of the video chaining session reaches either state 0 or state $r$, starting in state $(r - 1)$. Repeatedly applying the recurrence Equation 6.6, we obtain

$$(T_{r-1} - T_r) = T^*$$

$$(T_{r-2} - T_{r-1}) = \frac{1}{(r-1)} \gamma (T_{r-1} - T_r) - \frac{1}{(r-1)\lambda}$$

$$= \frac{1}{(r-1)} \gamma T^* - \frac{1}{(r-1)\lambda}$$

$$(T_{r-3} - T_{r-2}) = \frac{2}{(r-2)} \gamma (T_{r-2} - T_{r-1}) - \frac{1}{(r-2)\lambda}$$

$$= \frac{2}{(r-2)} \gamma \left( \frac{1}{(r-1)} \gamma T^* - \frac{1}{(r-1)\lambda} \right) - \frac{1}{(r-2)\lambda}$$

$$= \frac{1}{(r-2)} \gamma^2 T^* - \frac{2\gamma}{(r-2)\lambda} - \frac{1}{(r-2)\lambda}$$

Then proceeding inductively,

$$(T_{r-k-1} - T_{r-k}) = \frac{1}{(r-k)} \gamma^k T^* - \frac{k\gamma^{k-1}}{(r-k)\lambda} - \frac{1}{(r-k)\lambda}$$

$$(T_0 - T_r) = \sum_{k=0}^{k=r-1} (T_{r-k-1} - T_{r-k})$$

where $(T_0 - T_r) = 0 - 0 = 0$.

$$0 = \sum_{k=0}^{k=r-1} (T_{r-k-1} - T_{r-k})$$

$$0 = \sum_{k=0}^{k=r-1} \left( \frac{1}{(r-k)} \gamma^k T^* - \frac{k\gamma^{k-1}}{(r-k)\lambda} - \frac{1}{(r-k)\lambda} \right)$$

(6.7)

Substituting Equation 6.4 in Equation 6.7, we obtain

$$0 = \frac{T^*}{Q^*} - \sum_{k=0}^{k=r-1} \frac{k\gamma^{k-1}}{(r-k)\lambda} - \frac{1}{(r-k)\lambda}$$

$$\frac{T^*}{Q^*} = \sum_{k=0}^{k=r-1} \frac{k\gamma^{k-1}}{(r-k)\lambda} - \frac{1}{(r-k)\lambda}$$

$$T^* = Q^* \left( \sum_{k=0}^{k=r-1} \frac{k\gamma^{k-1}}{(r-k)\lambda} - \frac{1}{(r-k)\lambda} \right)$$

Therefore, the expected value for each of the epoch duration is given by

$$t_e = T^*$$

(6.8)
6.4.5 EXPECTED LIFETIME OF THE RELIABLE STATE

The expected lifetime of the video chaining session $t_s$ is defined as the product of the expected number of epoch $n_e$ and the expected duration $t_e$ in each epoch.

$$t_s = n_e \cdot t_e$$  \hfill (6.9)

Substituting Equation 6.5 and 6.8 in Equation 6.9, we obtain

$$t_s = \frac{T^*}{Q^*}$$

$$Q^* \left( \sum_{k=0}^{r-1} \frac{k y^{k-1}}{r^{r-1} k!} - \frac{1}{(r-k) \lambda} \right)$$

Therefore,

$$t_s = \frac{1}{\lambda} \sum_{k=0}^{r-1} \frac{k y^{k-1}}{r^{r-1} k!} - \frac{1}{(r-k) \lambda}$$  \hfill (6.10)

6.5 RESOURCE CONSTRAINTS

It is elucidated from Equation 6.10 that the expected lifetime $t_s$ of the video chaining session increases with both $r$ and $\gamma$. In order to maximize $t_s$, we must increase both $r$ and $\gamma$ as large as possible. Therefore, we consider four important resource constraints to increase both $r$ and $\gamma$. The constraints considered here are the repair buffer, repair latency, the repair bandwidth, and minimum number of replicas.

6.5.1 REPAIR BUFFER

The number of replicas $r$ is limited by the total available chaining buffers along the video chaining sessions. Therefore, the constraint for the chaining buffer is given by

$$r \leq r_{max}$$

where $r_{max}$ is an upper bound limit on the number of replicas along the video chaining sessions.

6.5.2 REPAIR LATENCY

The repair of replicas is accomplished by probing control messages along the video chaining session. Overhead considerations may limit the frequency of these probes. The limitation is associated with the average latency that occurs in detecting the lost replica and creating a new replica. This has direct effect on the peer down time $A_{dn}$. Therefore, the constraint for the repair rate is given by

$$\gamma \leq \gamma_{max}$$
where $\gamma_{\text{max}}$ is an upper bound limit on the normalized repair rate along the video chaining sessions.

6.5.3 REPAIR BANDWIDTH

The bandwidth used for creating new replicas is an important constraint. The repair mechanism cannot use more than the available average bandwidth in the chaining session. The effect of the bandwidth constraint on $r$ and $\gamma$ is as follows. Recall from Section 6.4.1, that the probability of peer availability is $A_{up}$, after which the peer departures. After the departure of the peer, the repair mechanism creates new replicas after a time of $A_{dn}$ duration. Therefore, a new replica will be created after an average time of

$$E[A_{up} + A_{dn}] = E[A_{up}] + E[A_{dn}] = \frac{1}{\lambda} + \frac{1}{\mu}$$

If the size of the video block is $b$ then this will incur an additional overhead of $b$ terms per streaming unit. Therefore, the constraint for the bandwidth is given by

$$\frac{rb}{\left( \frac{1}{\lambda} + \frac{1}{\mu} \right)} \leq b_{\text{max}}$$

where $b_{\text{max}}$ is an upper bound limit on the repair bandwidth along the video chaining sessions.

6.5.4 MINIMUM REPLICA

There exists a tradeoff between $r$ and $\gamma$ in maximizing $t_s$. Thus, either one of the two trading policies can be adopted to make an optimal replication strategy. The first trading policy is called as Max-Repair and the second trading policy is called as Max-Replica. If a max-repair policy is adopted then the value of $\gamma$ is large and the value of $r$ is small correspondingly. If a max-replica policy is adopted then the value of $\gamma$ is small and the value of $r$ is large correspondingly. We adopt max-replica policy in our M-Replication technique, because the p2pVoD applications are very sensitive towards delay. Note that, the max-replica has larger value for $r$ and has smaller value for $\gamma$. Therefore, the constraint for the minimum number of replicas is given by

$$r_{\text{min}} = \frac{b_{\text{max}} \left( 1 + \frac{1}{\gamma} \right)}{\lambda b}$$

where $r_{\text{min}}$ is an lower bound limit on the number of replicas along the video chaining sessions.
6.6 SIMULATION

The focus of the simulation is to give a fairer comparison among various replication techniques that can be used in our p2pVoD architecture. Hence, we have simulated our proposed M-Replication technique with other replication techniques such as Dynamic Resource Replication, Square Root Replication, Proactive Replication and Optimal Replication, which was briefed in Section 6.2. The topology used in the simulation is similar to our p2pVoD architecture. The parameters and its default values of the simulation are given in the Table 1.1.

The simulated model was evaluated for several trials. The result shown is an average of all simulation trials, which was carried out in all the clusters. The results of the simulation were evaluated with different success playback probability within the cluster. The success playback probability for a movie with length \( l \) is defined as the probability of successful reception of the video blocks throughout the entire duration of the movie. We have also evaluated the mean life time of the video chaining sessions to observe the availability level of movies in the cluster.

Figure 6.8 shows the generation of the replication factor \( r \) with respect to mean peer availability \( H(x) \) and mean video availability \( A \). It can be observed that the convex of the quadrilateral is increased linearly with respect to mean peer availability.
and mean video availability. These variations indicate that the M-Replication strategy adapts dynamically by replicating more number of replicas for the most popular movies and less number of replicas for the least popular movies.

Figure 6.9 shows the success playback probability of movies with respect to Poisson arrival rate. Initially, the arrival rate begins with a less number of requests and gradually increases with mean peer availability. As the arrival rate increases exponentially with respect to mean peer availability, the success playback probability is also coherently increased. It can also be observed from the Figure 6.9 that the replication techniques require minimum of 0.4 mean peer availability to assert the success playback probability. As it can also be observed that our proposed M-Replication technique achieves the greater success playback probability of 0.03% more than the optimal replication technique, 0.05% more than the proactive replication technique, 0.15% more than the square root replication technique and 0.23% more than the dynamic resource replication technique.

It can be observed from Figure 6.10 that the mean lifetime $t_s$ increases with both the number of replicas $r$ and the normalized repair rate $\gamma$ (gamma). Therefore, in order to maximize $t_s$, the M-Replication technique uses the Max-Replica policy. It can be analytically analyzed from the Figure 6.10 (a) that as the number of replicas
(a) Mean lifetime of the Replicas with respect to Number of Replicas.

(b) Mean Lifetime of the Replicas with respect to Gamma.

Figure 6.10: Mean Lifetime of the Replicas with respect to Number of Replicas and Gamma.
Figure 6.11: Average Buffer Utilization in Peers.

(a) Average Bandwidth Utilization in Peers.
increases the mean lifetime of the replicas are also increased concurrently. Meanwhile during failure of the peers, the repair mechanism of the M-Replication technique minimizes the mean peer unavailability by reactively repairing the lost replicas. Therefore, it can be observed from Figure 6.10 (b) that the increasing value of the normalized repair rate $\gamma$ with respect to the number of replicas influences the mean lifetime of the replicas. Eventually, our M-Replication technique adapts dynamically the normalized repair rate with respect to the number of replicas.

The impact of the bandwidth and buffer utilization can be observed from the Figure 6.11 and Figure 6.12 for M-Replication, Dynamic Resource Replication, Square Root Replication, Proactive Replication and Optimal Replication to our p2pVoD architecture, which shows effective results. It can be clearly observed from the Figure 6.11 that the impact of buffer utilization reflects only after 10 replicas. These replication techniques started utilizing the additional buffer space only after the increase of mean arrival rate of peers, thereby increasing the replication factor. It can be also observed that M-Replication technique uses lesser buffer space than other replication techniques.

Note that, the impact of bandwidth utilization by the peers involved in the chaining sessions should be inversely proportional to the bandwidth utilization at the
streaming server. It can be observed from the Figure 6.12(a) that the average bandwidth utilization in peers increases linearly with respect to the amount of replication factor.

It is worthwhile to note that in Figure 6.12(b), the peers using M-Replication technique, the virtue of bandwidth utilization in peers is inversely proportional to the bandwidth utilization at the streaming server. Meanwhile, in peers using other replication techniques, the bandwidth utilization in peers is directly proportional to the bandwidth utilization at the streaming server.

6.7 SUMMARY

In this chapter, an M-Replication technique was proposed for our p2pVoD system, which ensured the reliable state of the movies against individual or multiple failures of peers. To maintain a long term availability of the movies in the video chain, our proposed M-Replication technique creates functional replicas along the ongoing video chain session and, non-functional replicas around the auxiliary video chain session. We have developed an availability model and a reliability model for our proposed M-Replication technique. The result of the simulation shows that our M-Replication technique uses lesser bandwidth and buffer for the replication strategy than other replication techniques. In the next chapter, we will discuss about transmission mechanism and its impact on chaining session during VCR operations.