Media-on-Demand Transmission Schemes

Extract from
Draft Ph. D. Thesis of
Dipl.-Inf. Boris Nikolaus, Germany

2. November 2007

Contents

List of Figures

List of Tables

Chapter 3

Media Transmission Schemes

IN the previous chapter, an introduction to media transmission systems has been presented, covering the structure and operation of sender and receiver systems, an analysis of content types and encodings of transmitted media streams and an examination of different network types which can be used for media transmissions. This chapter focuses on one single (but important) part of a media transmission system: the **media transmission scheme**.

The media transmission scheme of a media transmission system describes how a media stream is transferred from a sender system to receiver systems. Currently, a wide range of transmission schemes have been proposed, and the goal of this chapter is to provide an overview about their functioning and basic approaches, their differences and commonalities and their advantages and drawbacks.

The main goal thereby is not to present a detailed and complete description of every transmission scheme but to show the basic idea behind each of them. This understanding of the differences between the schemes is particularly important to comprehend the evaluation and comparison process of the transmission schemes which concludes each of the sections describing transmission schemes.

Besides the textual (and sometimes mathematically supported) descriptions, diagrams are commonly added for better understanding, showing the internal structure of the transmission scheme or presenting an example transmission in a graphical notation. The used diagram types are introduced in the subsections where they are used for the first time, in particular

- channel-oriented diagrams in section 3.3.1 on page 11,
- stream-oriented diagrams in section 3.3.3 on page 14,
- enumerating transmission schedules in section 3.7.2 on page 50,
- tabular transmission schedules in section 3.7.2 on page 50,
- rectangular transmission schedules in section 3.7.2 on page 50,
- tree transmission schedules in section 3.7.2 on page 50.

Before starting with the presentation of existing transmission schemes, the following section 3.1 provides the mathematical background for an efficiency analysis of the required bandwidth of the schemes which is used to rate the schemes. In the six sections following this one, many different schemes are presented:

Starting with some traditional media transmission schemes in section 3.2, this chapter continues with a group of reactive media-on-demand transmission schemes in section 3.3 and strides ahead to the — most interesting — pro-active media-on-demand transmission schemes, which are presented in sections 3.4 to 3.7. Finally, section 3.8 presents the last group of transmission schemes which combine reactive and pro-active approaches.

In the last section of this chapter, a short summary and overall comparison of the presented transmission schemes and enhancements is given.

3.1 Efficiency of Media Transmission Schemes

Each of the following sections conclude with a comparison of the proposed transmission schemes. The most important issue for a comparison of different schemes is the **transmission efficiency**, i. e. the bandwidth consumption for a specific setup (e. g. the required sender bandwidth for a given maximum playback delay in case of pro-active schemes or for a given maximum playback delay and for a specific client request pattern in case of reactive schemes) compared to the theoretical lower limit of bandwidth which is needed to serve the same setup.

Bandwidth requirements of a transmission scheme depend very much on the prevailing conditions: While the media bit rate influences the transmission bandwidth requirements linearly and thus can be ignored in an analysis of the transmission scheme, the efficiency of a particular transmission scheme is mainly influenced by parameters like playback delay and the request pattern. For these reasons, typical request patterns for media-on-demand transmissions are examined in the next subsection.

For a comparison of transmission schemes, an efficiency value is used which can be visualized for different preconditions in charts at the end of each following section. This efficiency value η is calculated by

$$\eta = \frac{B^-}{B} \tag{3.1}$$

where B is the average bandwidth consumption of the transmission scheme and B^- is the minimum required bandwidth for the same setup.

As $B \geq B^- > 0$ is always fulfilled, the resulting value range of the efficiency is $0 < \eta \leq 1$, where $\eta = 1$ means a perfect transmission scheme in respect to its average bandwidth consumption.

This comparison introduces a question which has first been formulated in [EVZ99a] for reactive transmission schemes and in [ES01] for pro-active schemes: *What is the minimum required bandwidth to serve all requests of a given media-on-demand setup?* In subsections 3.1.2 and 3.1.3, this problem is discussed and solutions for different preconditions and types of transmission schemes are presented.

3.1.1 Request Pattern for Media-on-Demand Transmissions

Requests for media-on-demand transmissions are in mostly any case not predictable and thus have to be simulated through random processes when evaluating transmission schemes. When request patterns for media streams are analyzed, typically two probability distributions have to be examined:

The first distribution describes how customers' requests are spread over the provider's media assortment. It depends very much from the topicality and popularity of the media streams as well as from advertisements or sales discounts, but also from time, date, season, weather or competitive offers [LV93].

Interestingly, if the probabilities are not artificially impaired too much by the provider, the distribution is nearly identical in mostly any case: Media streams in media-on-demand systems are requested approximately according to the Zipf distribution with a parameter $s \approx 0.729$ [DPK94, DSS94] where the Zipf distribution is defined through the probability function [KRB06a, Wik06f]

$$P_s[X = n] = \frac{n^{-s}}{\sum_{i=1}^{N} i^{-s}} \tag{3.2}$$

In this formula, X is the variable of the random process, N is the total number of media streams in the media assortment, $1 \leq n \leq N$ is the rank of the examined media stream and s is a parameter of the distribution. The resulting distribution is shown in figure 3.1 for different numbers of media streams.

In the context of media-on-demand, the most important fact about this distribution is that most requests are performed for a very low number of media streams: For example in case of an assortment consisting of 1 000 media streams, about 36.2 % of all requests are sent for the most popular 5 % of the streams. Table 3.1 gives further examples for this distribution.

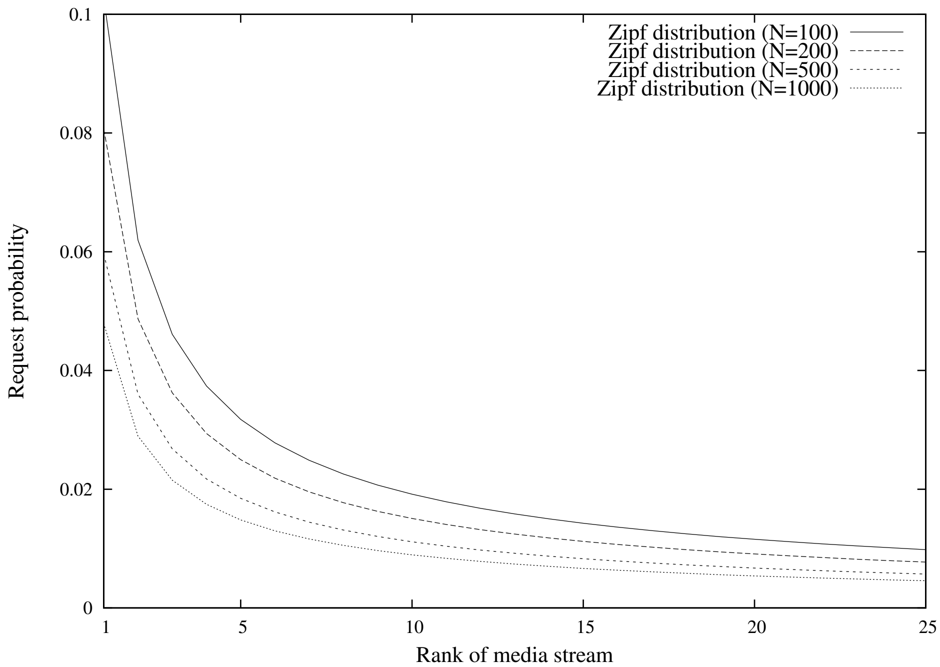


Figure 3.1: Zipf distribution with parameter 0.729 for media stream popularity

The other distribution of relevance describes how requests for a single media stream are distributed along time. Although the request rate for a single media stream varies

Number of requested media streams	Fraction of requested media streams	Fraction of requests
1	0.1 %	4.8 %
2	0.2 %	7.6 %
5	0.5 %	13.1 %
10	1.0 %	18.4 %
20	2.0 %	25.1 %
50	5.0 %	36.2 %
100	10.0 %	46.7 %
200	20.0 %	59.4 %
500	50.0 %	80.3 %

Table 3.1: Request frequencies for the most popular parts of the media stream assortment for a assortment of 1 000 media streams

over time (which may even change the popularity ranking of the media stream and shows up in the Zipf distribution), short term observations show that requests are Poisson distributed, a result which shows a high independence of request times of different customers[1].

The Poisson distribution [KRB06b, Wik06e] is parameterized by a single parameter λ and is defined by the following probability function

$$P_\lambda[X = n] = \frac{\lambda^n \cdot e^{-\lambda}}{n!} \tag{3.3}$$

where X is again the variable of the random process, n is the number of requests during a predefined period and e is the base of the natural logarithm ($e \approx 2.718\,281\,828$). Figure 3.2 shows some examples for Poisson distributions using different parameters.

3.1.2 Minimum Required Bandwidth for Reactive Transmission Schemes

For reactive transmission schemes and a limited receiver bandwidth of R times the (constant[2]) media bit rate b, the authors of [EVZ99a, EVZ01] presented the following formula assuming Poisson distributed requests of parameter λ:

$$B^- \approx b \cdot \eta_R \cdot \ln\left(1 + \frac{\lambda \cdot d}{\eta_R}\right) \tag{3.4}$$

where η_R is the positive real constant which satisfies the following equation:

$$\eta_R \cdot \left(1 - \left(\frac{\eta_R}{\eta_R + 1}\right)^R\right) = 1 \tag{3.5}$$

[1] Other distributions may occur if the transmission is time-bounded, e. g. most people would join the eight o'clock news at the same time for custom reasons or join a football transmission curious nearly without delay.

[2] Bandwidth variations are typically ignored when calculating the bandwidth requirements of a particular transmission scheme. See section **??** for impacts of variable-bit-rate media streams to transmission schemes and the efficiency loss which has to be expected when the transmission scheme does not support variable-bit-rate media streams natively.

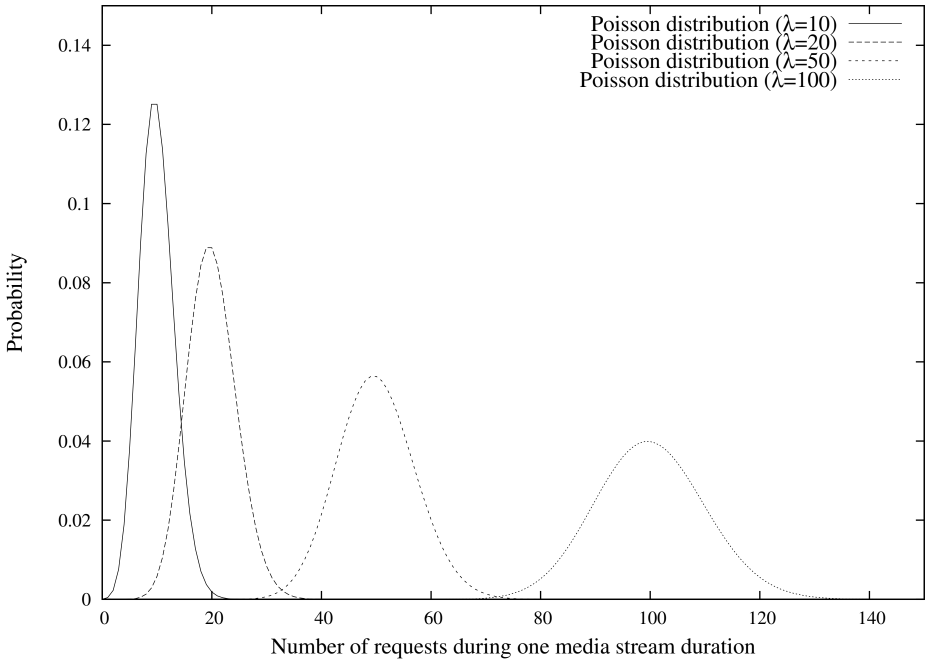


Figure 3.2: Poisson distribution for media stream requests

This formula has been obtained by assuming that infinitely small segments are transmitted on channels at the media bit rate and calculating the transmission frequency of each segment from probabilities. Table 3.2 shows the value of η_R and $\frac{B^-}{b}$ for some values of R and λ.

R	η_R	$\frac{B^-}{b}$ for $\lambda = 10$	$\frac{B^-}{b}$ for $\lambda = 100$	$\frac{B^-}{b}$ for $\lambda = 1\,000$
2.0	1.618	3.190	6.699	10.401
3.0	1.191	2.669	5.292	8.023
4.0	1.078	2.512	4.895	7.367
5.0	1.035	2.450	4.742	7.116
∞	1.000	2.398	4.615	6.909

Table 3.2: Value of η_R and the bandwidth requirements of the sender system $\frac{B^-}{b}$ for different values of receiver bandwidth R and request frequencies λ

If the segments are transmitted on channels at a bandwidth below the media bit rate instead, the required bandwidth at the sender system can even be reduced. The authors of the above equation provided in [EVZ01] the result if the channel bandwidth is reduced to $0 < C < 1$ times the media bit rate:

$$B^- \approx b \cdot \eta_{R,C} \cdot \ln\left(1 + \frac{\lambda \cdot d}{\eta_{R,C}}\right) \tag{3.6}$$

where $\eta_{R,C}$ is the positive real constant which satisfies the following equation:

$$\eta_{R,C} \cdot \left(1 - \left(\frac{\eta_{R,C}}{\eta_{R,C} + C}\right)^{\frac{R}{C}}\right) = 1 \tag{3.7}$$

If C tends to zero, the bandwidth at the sender system is minimized. The value of $\eta_{R,C}$ converges in this case to the constant $\eta_{R,\epsilon}$ which satisfies the following equation:

$$\eta_{R,\epsilon} \cdot \left(1 - e^{-\frac{R}{\eta_{R,\epsilon}}}\right) = 1 \tag{3.8}$$

Table 3.3 shows the value of $\eta_{R,\epsilon}$ and $\frac{B^-}{b}$ for some values of R.

R	$\eta_{R,\epsilon}$	$\frac{B^-}{b}$ for $\lambda = 10$	$\frac{B^-}{b}$ for $\lambda = 100$	$\frac{B^-}{b}$ for $\lambda = 1\,000$
1.0	∞	10.000	100.000	1 000.000
1.2	3.188	4.527	11.085	18.335
1.5	1.716	3.296	7.004	10.929
2.0	1.255	2.753	5.510	8.386
2.5	1.120	2.571	5.044	7.613
3.0	1.063	2.491	4.843	7.281
4.0	1.020	2.428	4.688	7.028
5.0	1.007	2.408	4.641	6.950
∞	1.000	2.398	4.615	6.909

Table 3.3: Value of $\eta_{R,\epsilon}$ and the bandwidth requirement $\frac{B^-}{b}$ of the sender system for different values of receiver bandwidth R and request frequencies λ

3.1.3 Minimum Required Bandwidth for Pro-Active Transmission Schemes

For pro-active transmission schemes, the minimum required bandwidth has been examined several times if no restriction is put on the recipient bandwidth, e. g. by the authors of [ES02] which provided the following formula for the lower bound of required bandwidth:

$$B^- \approx b \cdot \left(\ln\left(1 + \frac{d}{W^+}\right) + O\left(\frac{\delta}{W^+}\right)\right) \tag{3.9}$$

where B^- is the lower bound for the required bandwidth for the transmission, b is the bit rate of the media stream, W^+ is the (maximum) playback waiting time, d is the duration of the media stream and δ is the duration of each segment of the media stream.

In the more detailed article [ES01] of the same authors, this lower bound has been calculated exactly as:

$$B^- = b \cdot \left(\Psi_0\left(\frac{d + W^+}{\delta}\right) - \Psi_0\left(\frac{W^+}{\delta}\right)\right) \tag{3.10}$$

where Ψ_0 is the digamma function [Wal01a, Wei06] which fulfills $\Psi_0(x) = \gamma + \sum_{i=1}^{x-1} \frac{1}{i}$ where $\gamma \approx 0.577\,215\,665$ is the Euler-Mascheroni constant [Wal01b]. As this formula is based on information theoretic arguments, the bandwidth of no pro-active on-demand transmission scheme can fall below it. But as shown later (see subsections 3.6.4 and 3.7.1) it is possible for a pro-active transmission scheme to reach this lower bound.

For huge values of x, Ψ_0 can be approximated by $\Psi_0(x) \approx \ln x$ which gives the result of equation (3.9). But the above formula also allowed the authors to derive a more exact approximation using a better approximation for Ψ_0:

$$B^- \approx b \cdot \left(\ln\left(1 + \frac{d}{W^+}\right) + \frac{1}{2\frac{W^+}{\delta}\left(1 + \frac{W^+}{d}\right)} + O\left(\frac{\delta^2}{W^{+2}}\right) \right) \tag{3.11}$$

In this thesis, always the exact formula of equation (3.10) is used to compute the efficiency of pro-active transmission schemes. Only when estimating the effects of some enhancements, the approximations are used.

Unfortunately, no lower limit has been given yet in literature when the receiver system is not able to receive all transmission channels at once. Therefore the following model is used to calculate the lower limit:

- The media stream is split into equally sized segments.
- Each segment is transmitted round-robin on its own channel.
- The receiver system joins the channels starting with the first one, up to the bandwidth limit of the receiver system.
- When the receiver has got all data of a channel, the channel is released and the regained bandwidth is used to continue joining of further channels.
- All segments are transmitted using the lowest possible bandwidth. The bandwidth of each channels has to reflect the playback time of the segment and the joining delay of receiver systems for this channel.
- To find the bandwidth limit, the number of segments into which the media stream is split is increased infinitely.

The limit where this algorithm converges to can also be retrieved by moving to continuous functions. This yields to the following definitions:

1. Calculate the position in the media stream until which the receiver system can join the reception immediately:
$$t_0 = W^+ \cdot (e^R - 1) \tag{3.12}$$
2. Let Ft be the bandwidth characteristics of the media stream transmission at time t. Then for $t \leq t_0$, let Ft be
$$Ft = \frac{b}{W^+ + t} \tag{3.13}$$
3. Let $F(t)$ be an antiderivative of Ft. Therefrom follows for $t \leq t_0$ that $F(t)$ equals
$$F(t) = b \cdot \ln(W^+ + t) + C \tag{3.14}$$
with some constant C.

4. For $t > t_0$, let Ft and its antiderivative $F(t)$ be the solution of the following differential equation:

$$F(t) = b \cdot R + F\left(t - \frac{R}{Ft}\right) \tag{3.15}$$

Then the required bandwidth for the transmission of the media stream from t_1 to t_2 is

$$B^{-} = \int_{t_1}^{t_2} Ft\,dt = F(t_2) - F(t_1) \tag{3.16}$$

Unfortunately, there is no closed formula for equation (3.15) available so it must be solved numerically.

Although no prove for the optimality of this model is given here, these approaches (which converge to the same results) can be seen as a straightforward extension of the Polyharmonic Broadcasting scheme which has been proven to be optimal (see section 3.6.4).

Figure 3.3 shows the so gained minimum bandwidth requirements at the sender system for different receiver bandwidths. The most interesting fact about the resulting bandwidth requirement is that the influence of the receiver bandwidth is much minor than expected: Even for a very short playback delay of $\frac{1}{7200}$ of the media stream duration (which equals 1 second for a two hour media stream), the bandwidth requirements for a transmission which considers a three channel receiver bandwidth capacity is less than 4 % higher than the bandwidth requirement without receiver bandwidth restriction. Because of this result, lowering the receiver system bandwidth requirements is studied for the Generalized Greedy Broadcasting scheme in section **??**.

3.2 Traditional Media Transmission Schemes

Before introducing "advanced" media-on-demand transmission schemes, this subsection presents some traditional media transmission schemes. The term "traditional" thereby is used to indicate that these schemes have been invented long time ago and provide simple and (partially) trivial solutions for media stream transmissions. Some of these systems do not even offer "on-demand" support, and others provide on-demand services using non-technical solutions. But for completeness and as these schemes are the origins of the media-on-demand idea, a short overview of these transmission schemes is given in this section.

3.2.1 Single Broadcast

Since the invention of radio and television broadcasting (in 1893 and 1925, respectively [Wik06d, Wik06b]), broadcasting organizations established and started to transmit media streams (traditionally news, music and movies) regularly. The schedule which is used by the broadcasting companies is usually very irregular: For a viewer it is mostly impossible to predict the schedule of a specific transmission. Nevertheless, single broadcasts are very popular today and are used broadly.

To make the transmissions more predictable for viewers, several solutions have been established:

- A TV/radio guide lists the schedule for the next week(s).

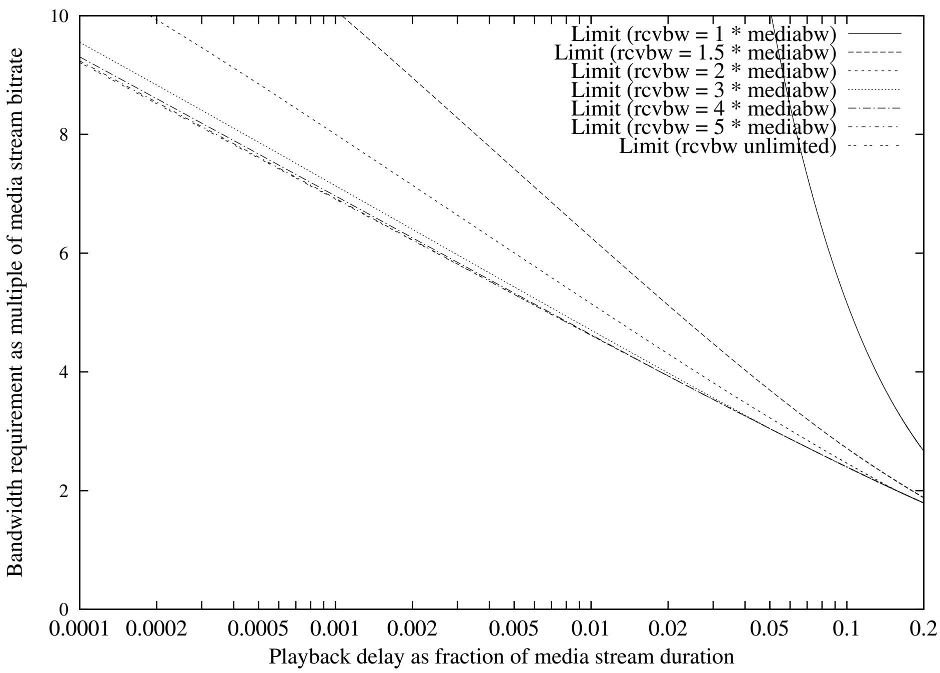


Figure 3.3: Minimum required bandwidth at the sender system for different receiver bandwidth capacities

- Some transmissions are broadcast at regular times (e. g. many radio stations transmit news every full hour, and episodes of TV series are transmitted daily or weekly at the same time).
- Special events (e. g. sport events, concerts) are transmitted without delay (live) or with minor delay (near-live).
- For request programs, viewers can vote for their favorite movie which is then transmitted at a preallocated slot in the schedule. Similarly, many radio stations allow their hearers to send their music wishes by telephone or mail.

But howsoever, for a viewer it is nearly impossible to determine when a specific movie will be transmitted the next time, so this transmission scheme cannot be classified as "on-demand".

3.2.2 Personal Tape Archive

Keeping a personal tape archive is a very simple form of traditional media-on-demand: It allows to request a media from a given assortment within a relative short time (depending only on the cataloging and organization of the archive). Besides of the space consumption of a personal video assortment, the main disadvantage of this type of media-on-demand lies in the availability of media streams in the archive: Each media stream has to be bought or recorded from another source (e. g. a previous broadcast transmission) before it can be viewed, so the assortment typically does not contain the newest media streams and is acquainted within short time.

3.2.3 Video Rental Stores and Media Libraries

An alternative to personal tape archives are commercial or public ones: Video rental stores or media libraries. In contrast to personal tape archives, their assortment does not require any space at home, is usually up-to-date and contains many new movies. The main disadvantage of these solutions are that it takes some time and (sometimes) self conquest to fetch a movie and some luck that the wanted movie is available. Additionally, the movie has to be returned afterwards.

3.2.4 Complete Preloading of Media Streams

Complete Preloading of media streams means that streams are transmitted to the receiver system in advance (i. e. before a playback request is received) and stored on the receiver system until they are requested. Although this media transmission scheme has the best efficiency conceivable, it does not provide a media-on-demand access in the strict sense as the media reception is independent from the media request and the media stream is not received concurrently to the media playback.

Complete preloading has a lot of disadvantages: Firstly, the receiver system must have enough storage to save a whole copy of all media streams at once which imposes high costs for receiver systems (or limits the size of the media assortment badly). Secondly, the receiver system has to stay tuned on to the preloading transmission permanently to receive and store all pushed streams.

Instead of preloading all media streams, it is also possible that the customer performs a selection some time before the playback is requested [WMS06]. The receiver system can then tune to exactly the selected media streams and store them until the playback is requested.

While this reduces the storage requirements of the receiver system, it does not allow any spontaneous requests for media streams which have not been requested in advance.

3.2.5 Summary of Traditional Media Transmission Schemes

Traditional media transmission schemes do not provide a media-on-demand service in the strict sense: The transmission schemes do not provide the media stream on request, use physical delivery, require huge storage devices or a preselection of the transmitted media streams. Ignoring these problems, Selective and Complete Preloading provide the best approximation to media-on-demand.

3.3 Reactive Media-on-Demand Transmission Schemes

After the presentation of these very simple, traditional media transmission schemes in the last section, this section advances to more modern solutions which provide real media-on-demand service.

In this section, several reactive media-on-demand transmission schemes are described. The first two of them are still very simple and transmit the media streams as a whole, but the following five schemes use partial transmissions to lower the bandwidth requirements.

The transmission scheme presented in the last subsection of this section differs from the above ones completely in the type of media distribution because the media stream is forwarded between recipient systems.

3.3.1 Point-to-Point Transmissions

For Point-to-Point transmissions (which are sometimes referred to as **streaming** [Wik06c] in the context of media transmissions or **vodcasting** [Wik06a] if based on Really Simple Syndication (RSS) feed technology), the media stream is transmitted to the receiver system just-in-time on request, using a one-to-one (Point-to-Point) transport connection. Point-to-Point transmissions allow the highest level of interactivity, as any request from the receiver system can be handled at the sender system without influencing other receivers. But on the other side, using one dedicated channel per active receiver system causes high bandwidth requirements for the sender system as the sender system must be able to attend all active recipients at a time. Besides of the high bandwidth requirements, this transmission scheme does not guarantee a maximum playback delay (except for the case that enough resources are available to service all potential receiver systems concurrently).

For managing the available sender system resources, a simple scheduling strategy can be used, e. g. incoming requests may be appended to a first-in-first-out (FIFO) queue and removed from the queue whenever resources are available.

To visualize this type of transmission scheme, a **channel-based diagram** is used: In channel-based diagrams, the transmitted part of the media stream of all channels are shown using bars, which are arranged in rows, one below the other. If the media stream has been cut into several segments, the bars may be labeled with indexes to name the segments they carry. If channels of different or varying bandwidth are used, the thickness of the bars is used to indicate the bandwidth.

Sometimes it is desirable to show a playback together with the transmission. In this case the received parts of the transmission are shaded and an additional bar (per visualized playback) below the time axis shows the time of playback. The time of the recipient request is shown in front of the bar using an arrow head.

Figure 3.4 shows a channel-based diagram for three Point-to-Point transmissions, together with the playbacks of the recipients.

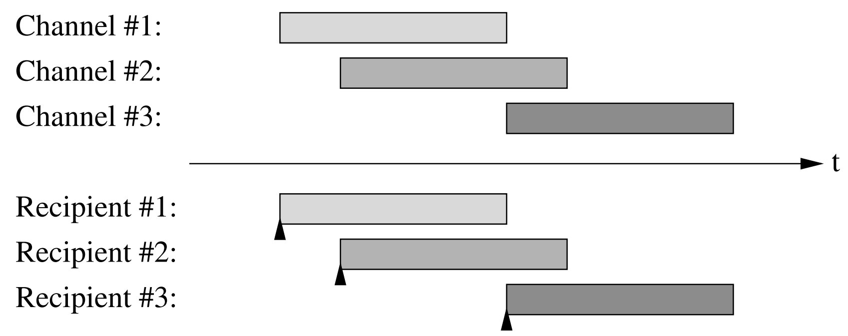


Figure 3.4: Diagram of a Point-to-Point transmission.

3.3.2 Batching

Batching is a very old and intuitive way to lower the bandwidth requirements of the Point-to-Point transmission scheme by using one-to-many transmissions. In Batching systems, several requests from receiver systems for the same media stream will be served at once in a batch, using only a single transmission.

As it is a seldom coincidence that several receiver systems ask simultaneously for the same media stream, receiver requests have to be collected for some time, delaying the playback for the first receivers. For making the decision when to start a transmission, several different strategies exist, always trying to make a compromise between fairness and the number of served recipients. To find a balance between fairness and the number of served recipients is very important for Batching schemes [GH90, BCMM00, CT99, PCL98a]: If the system weights fairness to high, it serves requests too inefficient and therefore decreases the number of served recipients. On the other side, if media streams with many outstanding requests are preferred to media streams with few requests, the consumers of the latter streams have to wait too long and may renege the service at all.

The most popular Batching algorithms for media-on-demand are:

- **First-come-first-served (FCFS)** [DSS94]: Requests are inserted into one waiting queue. Whenever a channel becomes available, the longest waiting request is removed from the queue and served (together with all other requests for the same media stream). This Batch algorithm is the fairest one the maximum playback delay is independent from the requested media stream.
- **Delayed First-come-first-served (Delayed FCFS)**: Same as FCFS, but requests are never served immediately but always after a short delay, so the available bandwidth at the sender system is not exhausted as fast as when using FCFS.
- **Maximum-queue-length (MQL)** [DSS94]: For each media stream, a separate queue is maintained. When a channel becomes available, the queue with the most requests is searched and these requests are served. This algorithm has a much lower average service time, but handles requests for seldom media streams very unfair as these customers may have to wait a long time for their request being served. In case of high utilization of the sender system, requests for seldom media streams may even never get served.
- **Maximum-factored-queue-length (MFQL)** [AWY01]: Whenever a channel becomes available, the media stream with the highest factored queue length is scheduled where the factors of the queues are the reciprocal of the square root of the request frequency of the media streams. This strategy depicts a compromise between MQL (which does not consider that viewers may renege because of too long waiting times) and FCFS (which does not take queue lengths into account).
- **Prioritized Queuing**: FCFS-like Batching, but requests can additionally be prioritized to pass other requests with low priority. For example, requests of premium customers can be assigned a higher priority to provide lower waiting times to them.
- **Combinations** of a Batching scheme with *n* Staggered Broadcasting transmissions (e. g. FCFS-*n*, MQL-*n* [DSS94] or MFQL-*n*): The hottest *n* media streams are transmitted using Staggered Broadcasting (see section 3.4.2) and the remaining media streams are served using another algorithm. As a consequence, the *n* hottest media streams have a fixed maximum playback delay while the remaining media streams can be handled as desired.
- **Combinations** of a Batching scheme with Single Broadcasting: As an alternative to queueing requests which cannot be fulfilled immediately, it is possible to

allow the consumer to switch onto the last requested transmission of the media stream. If the consumer chooses this option, he misses the beginning of the media stream but he is not required to wait until bandwidth becomes available for a complete new transmission.

Besides of these ones, many schemes from other Batching systems may also be used for media-on-demand systems.

Batching schemes can lower the bandwidth requirements a lot but they also reduce the interactivity which can be granted to each consumer compared to Point-to-Point transmissions as any navigation would affect all consumers of the whole batch. This decoupling of media-on-demand sender and receiver systems is typical for media-on-demand systems which serve several consumers with a single transmission and is indispensable if the interacting user is not isolated from the remaining ones (or moved to another group [LLG97]) and if the interactivity cannot be performed solely on the media-on-demand receiver system (e. g. backward navigation can be provided by the receiver system if it has enough storage to save one copy of the media stream).

Figure 3.5 shows a transmission diagram for a Batching transmission (e. g. Delayed FCFS), together with the request times and the playback of three recipient.

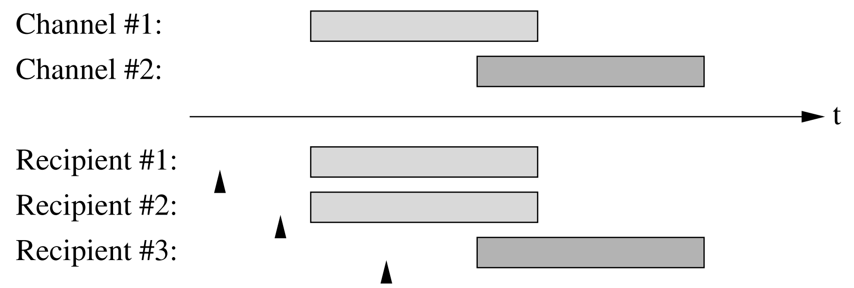


Figure 3.5: Diagram of a Batching transmission.

3.3.3 Adaptive Piggy-Backing

While Batching benefits from one-to-many transmissions by allowing several recipients to listen to one media stream, the Adaptive Piggy-Backing transmission scheme [GLM95, GLM96, AWY96a] introduced a completely new approach to reduce the bandwidth in media-on-demand transmissions: The basic idea of the Adaptive Piggy-Backing transmission scheme is that a recipient can *benefit from other ongoing transmissions* of the same media stream.

To bring two media stream transmissions to the same time of transmission without interrupting the older one, the display rates of the transmissions are modified for the Adaptive Piggy-Backing transmission scheme. A bandwidth gain is achieved this way because the channel of one of the transmissions can be terminated and used for the next request when the streams have been merged.

The authors of the article describing the Adaptive Piggy-Backing transmission scheme also stated that

> [. . .] the video editing industry have assured [them] that alterations of the actual display rate in the 2-3 % range or expansion and contraction

(through interpolation) in the 8 % range can be accomplished without being noticeable to the viewer.

But even if one media stream is expanded by 8 % and the next following one contracted by 8 %, two request can only be merged if the time between their arrivals is shorter than 16 % of the media duration. For example for a two hour media stream, the requests can only be merged if the second request arrives within 19.2 minutes after the first one. And the bandwidth reduction is very small, even when the offset of the second request is as near as 9.6 minutes, only the half media stream transmission can be left out.

Another drawback of this approach is that this creates additional requirements to the display system: Assuming a transmission which includes a visual stream, the display system must be able to adapt its frame rate to the display rate which has been determined by the Adaptive Piggy-Backing transmission scheme. If this is not supported by the display system, frames have to be inserted into or deleted from the transmitted stream for presentation to adjust the display rate. These changes cause irritating judders and thereby reduce the media quality significantly during presentation.

In [GLM95, GLM96], four different algorithms for the selection of the display rate adjustment are proposed:

- **Odd-Even Reduction Policy:** The media stream for the first request is adjusted to the lowest possible bit rate, the next request (within a given window) uses the maximum bit rate until the stream merges. If the next request arrives very late (e. g. at a time so it cannot be merged) or if a third request arrives, the algorithm is restarted, serving the request again using the lowest possible bit rate.
- **Simple Merging:** For the first stream, the lowest bit rate is used, for all further streams within a fixed window the maximum bit rate is used.
- **Greedy Policy:** Same as odd-even policy, but after a merge or when reaching the window when no other stream can merge to it, the algorithm is applied again to this stream. This way, merged streams can be merged to other merged ones.
- **Limited Merging:** Merging is restricted to a predefined first part of the media, so the whole media needs not to be available using several bit rates on the sender system.

For the Adaptive Piggy-Backing transmission scheme, a new type of diagram is needed as the merging process is not clearly visible in the channel-based type of diagram which has been used to illustrate the previously described transmission schemes. In **stream-oriented diagrams**, each stream transmission is represented by a diagonal line. The bit rate of the transmission is shown by the slope of the line, and a premature termination of a channel can be identified by the fact that the line does not reach the height of the end stream marker at the left, vertical axis. Horizontal, dashed lines are sometimes added at the premature termination of a stream to show which stream is used to continue the playback. (For Adaptive Piggy-Backing, the continuation stream is met by the terminating stream, so this line is not necessary here.)

Figure 3.6 shows a transmission diagram for the Adaptive Piggy-Backing transmission scheme, illustrating the playback of three recipients. The transmission of the second recipient is merged to the first one in this example.

3.3.4 Patching

Patching [HCS98b] uses the same idea as Adaptive Piggy-Backing to reduce the bandwidth of the sender system: Recipients benefit from other ongoing transmissions of the

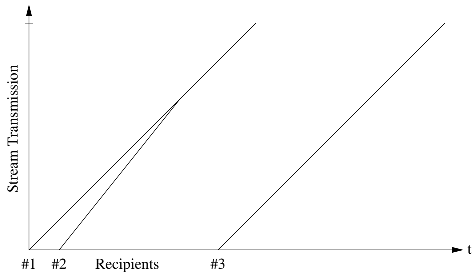


Figure 3.6: Diagram of an Adaptive Piggy-Backing transmission.

same media stream. But instead of modifying the display rate of transmissions, Patching requires that the clients receive data from another transmission while the actual transmission is received and played.

Therefore two types of transmissions are possible for the Patching scheme: Firstly, the sender system can transmit a media stream in its whole, as it has been done in the Batching scheme. The first request for a media stream is always served using such a **full transmission**. But secondly, if a request arrives a short time after a full transmission has been started, the sender system can decide to send only the "missed" part to the recipient. The recipient has to listen to both the new **patch transmission** and the ongoing full transmission in this case, buffering the data of the full transmission and "patching" the parts together when playing.

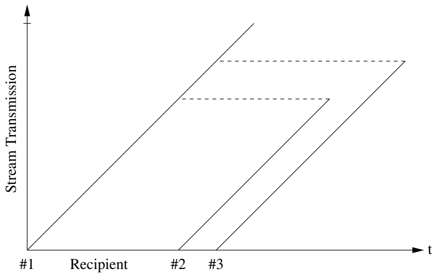
While full transmissions must be sent using one-to-many transmissions, the patch transmission can use either one-to-one transmissions (if the recipient request is to be served immediate), or one-to-many transmissions (if several recipients are served with the patch transmission, e. g. if Batching is used to collect several requests).

As above already mentioned, the Patching scheme introduces two new requirements to the receiver system compared to the Batching scheme:

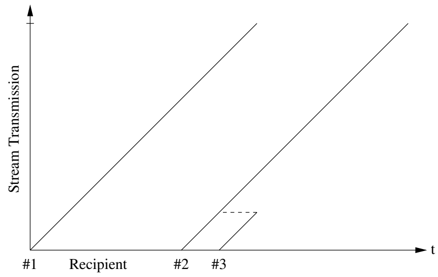
- The receiver system must be able two receive two transmissions at the same time, so the bandwidth requirement of the receiver system is twice the (maximum) media stream bit rate.
- The receiver system must have a buffer where the data of the full transmission is stored while the patch stream is played.

The efficiency of the Patching scheme depends very much on the decision when a full transmission is to be started for an incoming request and when a patch stream is used instead. Unfortunately, using patch transmissions instead of full transmissions does not always result in a lower bandwidth as illustrated in figure 3.7: In figure 3.7a, both transmissions #2 and #3 merge to transmission #1, resulting in a total of 2.5 media stream transmissions, whereas in figure 3.7b only transmission #3 merges to the previous full transmission #2 and the needed bandwidth totals to $\frac{13}{6} \approx 2.17$ media stream transmissions.

Another problem arises if the buffer size of the recipient system is not large enough to hold half of the media stream. (It is not possible that the receiver system has to buffer more than half of the media stream as only two transmissions are received and one of them is played immediately.) In [HCS98b, CHV99], some algorithms for the merge decision have been proposed for this case:



(a) Both streams #2 and #3 patch to the full transmission #1



(b) Only the last stream #3 patch to the preceding full transmission #2

Figure 3.7: Two different ways for the transmission of three media streams using patching.

- **Greedy Patching**: If the client buffer is too small for the missed part, the last part of the full transmission is buffered and the patch has to transmit the remainder.
- **Grace Patching**: If the client buffer is too small for the missed part, a new full transmission is scheduled.
- **Patching with Fixed Window**: If the client request arrives within a fixed window after a full transmission, only the patch is transmitted, otherwise a new full transmission is started. For this algorithm, the client buffer must be large enough to store a part of the media stream of the size of the fixed window.
- **Optimal Patching** [CHV99, GT01, GT99]: Same as patching with fixed window, but the window size is calculated using the request arrival rate of the recipients. Assuming a Poisson distribution of the client requests, this algorithm calculates the window size so that the average bandwidth is minimized.

Figure 3.8 shows the four different approaches how a receiver request is handled if it arrives so late that the missed part cannot be cached by the client. To emphasize the differences, a very small buffer has been chosen.

The authors of [BNL04] provided an algorithm of complexity $O(n^2)$ for calculating the optimal merge decisions for a set of request times. Although this algorithm cannot be used in a practical environment (as the request times are normally not known in advance), it is possible to use this algorithm to examine the theoretical limit which can be reached by the Patching transmission scheme at all.

3.3.5 Tapping

Tapping [CL97, SGRT99] can be seen as an improved version of Patching[3]. While the Patching scheme only differentiated two types of transmissions — full transmissions and patch transmissions — the Tapping transmission scheme uses three classes: **original streams** for full transmissions, **full tap streams** for patch transmissions and **partial tap streams** as new class of transmission.

[3]Strictly speaking, Tapping has been published before Patching, so Patching has to be looked at as a simplified version of Tapping. But for better understanding, Patching has been presented before Tapping in this thesis.

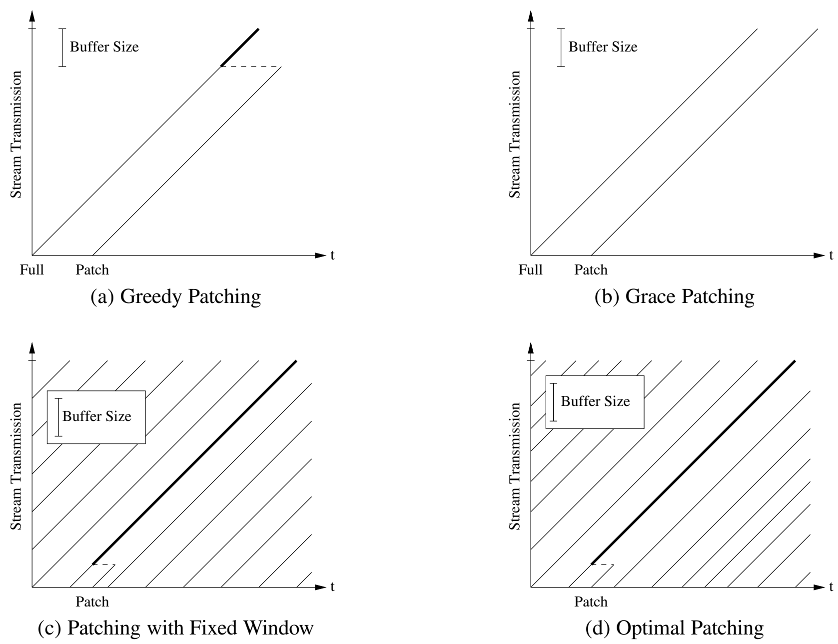


Figure 3.8: Four ways how a request (Patch) is handled which arrives too late to merge with the previous full transmission.

Partial tap streams are used when a stream should be merged to another transmission but the buffer of the receiver system is too small to save the missed part of the transmission. For partial tap streams, the buffer of the receiver system is first filled with the data of the ongoing original transmission until it is full. When the data of the tap stream reaches the buffered position, the transmission on the tap stream can be interrupted and the receiver system can play the buffered data. While playing data from the buffer and therefore emptying it, the receiver system again listens for new data from the original transmission and saves it into the buffer. When the data of the buffer has been played, the stream is again continued at the tap stream, until the position in the buffer is reached and the procedure is repeated.

If the receiver system can receive more than two streams at once, extended versions of Tapping [CL99, CLP00] can be used:

- **Extra tapping**: The receiver system can tap to any ongoing stream transmission, not only to the original transmission.
- **Stream stacking**: When the receiver system has space left in its buffer and another channel is not used at the moment (e. g. a partial tap stream), the order of the transmissions may be rearranged, using the suspended other channel for the transmission. This does not reduce the total bandwidth to transmit but allows to release channels earlier.

Figure 3.9 shows a (greedy) Patching transmission compared to an original Tapping transmission, a Tapping transmission using extra tapping and a Tapping transmission using stream stacking.

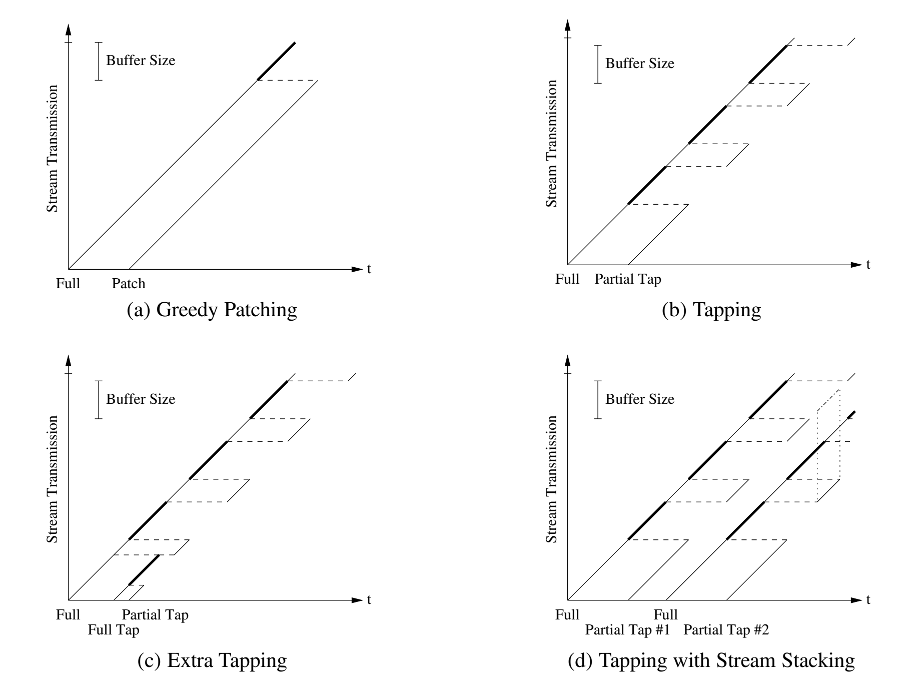


Figure 3.9: Comparison of Patching and Tapping.

The idea of Tapping has also been published from other authors under the terms **Generalized Buffer Reuse Patching** or **Optimized Patching** [SGRT99]. Additionally, in [Pâr01b], the authors improved the performance of Tapping by preloading a part of the media stream to the receiver system some time before the stream is requested. The effects of partial preloading are examined in detail in section **??**.

3.3.6 Merging

Merging [EVZ99b, EVZ99a, EVZ01] is the most generalized version of Patching and Tapping: In the Merging scheme, receiver systems can receive data from any stream, not only full transmissions.

This improves the performance, but at the same time it increases the complexity at the sender side: The sender system must not only decide if a full or patch transmission should be used for a request, all merge decisions have to be evaluated.

Merging is a very efficient transmission scheme for media on demand, and several strategies which streams to merge with which other stream have been proposed:

- **Earliest Reachable Merge Target (ERMT)** [EVZ99b, EVZ99c]: For the Earliest Reachable Merge Target strategy, the first possible merge target is selected when the merging process of all (yet) existing streams is simulated.
- **Simple Reachable Merge Target (SRMT)** [EVZ99b, EVZ99c]: As the decision for Earliest Reachable Merge Target is very complicated to calculate in large systems, this strategy uses a more simple one: A merge target is only selected

if the merge time of the merge target with its immediately preceding stream lies behind the start time of the examined stream.

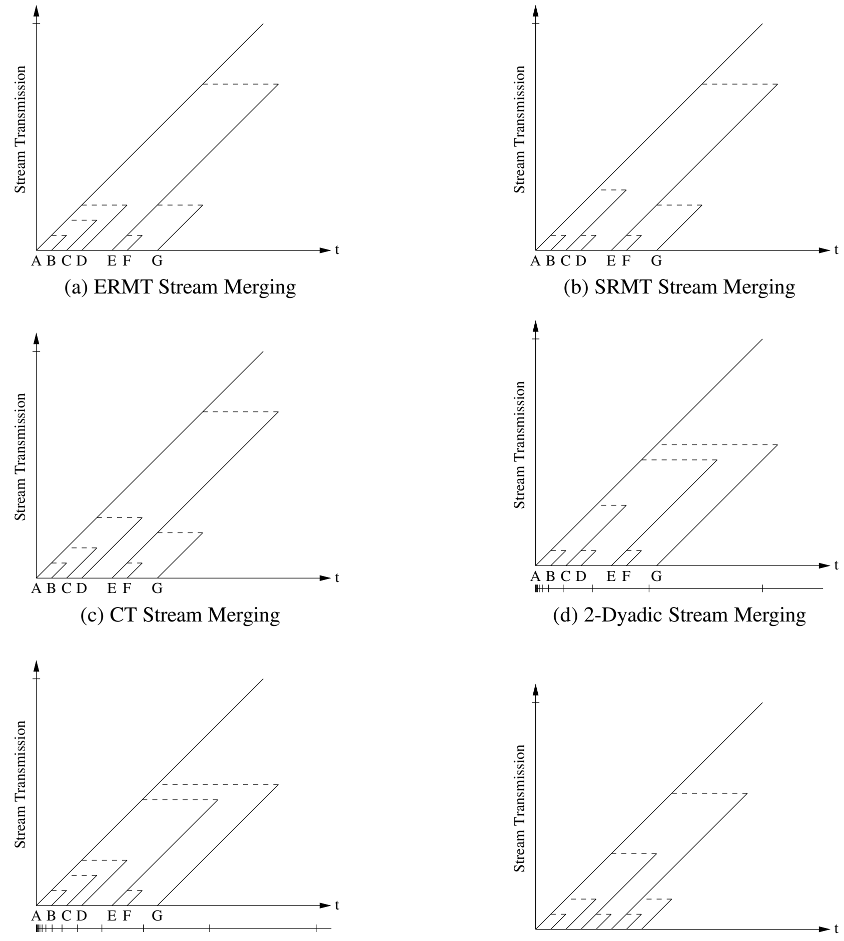
- **Closest Target (CT)** [EVZ99b, EVZ99c]: Another simplification of the Earliest Reachable Merge Target strategy is to always merge to the closest preceding stream which still exists and restart the search when the chosen stream terminates as it merged to another stream.
- **Dyadic Stream Merging (2-Dyadic)** [CJM02]: For the Dyadic Stream Merging strategy, the interval of a full playback is divided into dyadic intervals of parameter 2 (i. e. into intervals which are obtained by splitting the interval into two halves and continuing splitting the first half interval recursively). The first request of the second half starts a new full transmission, therefore no requests of the second half have to be considered here. For requests of the first half, the first request in each of the dyadic intervals is merged directly to the full transmission. For the remaining request, the intervals, starting at these requests until the end of their dyadic interval are again split into dyadic intervals and requests in them are merged in the same manner, continuing until all requests have been processed.
- **Dyadic Stream Merging with golden ratio (ϕ-Dyadic)** [BNGLT02]: This strategy uses the same merging algorithm as the Dyadic Stream Merging strategy, but instead of halving intervals, they are split according to the golden ratio $\frac{1+\sqrt{5}}{2}$ [BS91]. The authors of the Dyadic Stream Merging strategy have proven that this is the optimal merging strategy for Poisson distributed request arrivals if the requests are not known in advance.
- **Optimal stream forest calculation** [LLG98, BNL04]: For a given list of request times it is possible to calculate the optimal merging combinations in $O(n^2)$ using dynamic programming. This approach cannot be used in real environments as the list of request times is not known in advance.
- **Optimal off-line merging trees** [BNGL03]: For a pro-active transmission scheme (sometimes called off-line transmission scheme), it is possible to compute merging trees in advance. The optimal merging trees in this case are Fibonacci trees, i. e. trees which are defined by a recursive, Fibonacci-like function:

$$S(i) = \begin{cases} \text{the single node } i & \text{if } i \leq 2 \\ \text{the tree obtained by adding } S(i-2) \text{ as last} \\ \text{child to the root node of } S(i-1) & \text{if } i > 3 \end{cases} \tag{3.17}$$

Figure 3.10 shows an example how one request pattern is handled differently by the ERMT, SRMT, CT, 2-Dyadic, ϕ-Dyadic and optimal off-line merging strategies.

3.3.7 Virtual Batching

Virtual Batching (sometimes also referred to as Chaining) [SHH97, SHT97] differs from the above transmission schemes by requiring that each receiver system can send a media stream to another receiver system. Virtual Batching uses this capability to create chains of receivers, where each receiver passes the media stream data to another receiver with a delay (which is realized by buffering the data before passing it). If a new recipient requests the media stream before the buffer of the last recipient in the chain was exhausted, the new recipient can simply be added to the end of the chain, otherwise the receiver system becomes the first system in a new chain.



(a) ERMT Stream Merging

(b) SRMT Stream Merging

(c) CT Stream Merging

(d) 2-Dyadic Stream Merging

(e) ϕ-Dyadic Stream Merging

(f) Optimal Off-line Stream Merging

Figure 3.10: Comparison of different Stream Merging strategies.

Using receiver systems as further senders reduces the bandwidth requirements of the sender system a lot, but induces several problems:

- At first, the receiver systems must be able to send the media stream to another receiver system. This is probably the biggest problem as the back-channel of the receiver system often does not provide enough bandwidth (if available anyhow).
- The receiver system must provide means for resource reservations: It has to provide buffer space for delaying the media stream in the chain and it has to reserve bandwidth for sending the media stream to the next system in the chain.
- The media transmission system is more error-prone as it requires proper working of receiver systems. For example, switching a receiver system off or a power failure at one of the receiver system breaks the chain and causes an abortion of the stream transmission at all systems in the tail of the chain.
- All receiver systems in the chain must be trusted that they send the data un-

modified. Probably, the media stream has to be signed using an asymmetrical algorithm, so the recipient systems can check if the media stream has not been altered.

- It may be undesirable that a receiver system can detect its successor system in the chain for privacy reasons.

3.3.8 Peer-to-Peer Networks

A generalization of Virtual Batching is provided by Peer-to-Peer Networks: In a Peer-to-Peer Network, the sender and receiver roles are merged to a single peer role, so every node can consume data and provide data at the same time.

Similar to some advanced media-on-demand transmission schemes, todays Peer-to-Peer Networks split files into several small parts which are retrieved separately and finally concatenated. In contrast to advanced media-on-demand transmission schemes where the splitting of media streams into segments is used to lower the total bandwidth consumption for a media stream transmission to several recipients, the reason for splitting them in Peer-to-Peer Networks only serves the purpose to lower the bandwidth requirements of the sending peers[4]. The total bandwidth consumption for media stream transmissions using Peer-to-Peer Networks will still be the same as for Point-to-Point Transmissions because no multicasting transmissions are used.

Similar to Virtual Batching, Peer-to-Peer Networks have several backdraws if used for media-on-demand:

- The peer must be able to send the media stream to another peer. In contrast to Virtual Batching, the peer must not support sending the media stream at full media bit rate.
- The peer must provide means for resource reservations: As for Virtual Batching, buffer space and bandwidth for sending media stream segments must be reserved.
- The Peer-to-Peer Network will fail to work if the total media bandwidth of all active requests exceeds the total transmission bandwidth of all peers. As peers in Peer-to-Peer Networks have often only a low upstream bandwidth (e. g. using ADSL), the upstream bandwidth of several peers is required for the transmission of one media stream. This disallows to use Peer-to-Peer Networks for media-on-demand when it is possible that a high ratio of the peers is requesting a media stream concurrently.
- The peer must provide real-time semantics so each segment is available in due time when needed for playback. Peer-to-peer software which is currently used for file distribution does not provide any timing constraints for transmissions.
- The last three bullets of the list of drawbacks of Virtual Batching (more error-prone, security considerations, privacy concerns) apply to Peer-to-Peer Networks, too.

3.3.9 Comparison of Reactive Transmission Schemes

A comparison of the above reactive transmission schemes is very difficult as the schemes differ in many parameters, e. g.:

[4]The peers of Peer-to-Peer Networks have often only a low upstream bandwidth (e. g. using ADSL), so several peers are combined in a Peer-to-Peer Network to transmit the file in an acceptable time.

- averagely needed bandwidth of the sender system,
- maximum of needed bandwidth of the sender system,
- disposable bandwidth of the sender system,
- maximum bandwidth capacity of the receiver system,
- maximum storage requirements of the receiver system,
- playback request pattern,
- average playback delay,
- maximum playback delay,

To get an overview of the efficiency, the following assumptions are made for the forthcoming comparison:

- Playback requests are Poisson distributed with parameter λ.
- Receiver systems have enough storage to buffer any necessary amount of media data.
- The transmission of a single media stream is examined.

The first of this assumptions can be accepted as long as requests from different receiver systems are independent from each other (see section 3.1.1). The second assumption can be validated by observing the steadily falling costs of memory and hard disk storage: Memory is nowadays cheaper than EUR 0.065 per megabyte [Pri05a], and hard disks are available for EUR 0.33 per gigabyte [Pri05b], making tight restrictions on storage requirements nearly obsolete. The restriction to a single media stream is valid as long as all media transmissions are independent from each other.

Using these assumptions, the following schemes can be omitted in the comparison:

- FCFS Batching, MQL Batching and MFQL Batching only differ in the way how they handle requests for different media streams. Because of restriction three, these Batching schemes give equal results in this comparison.
- Prioritized Queuing Batching assumes different classes of recipients. This favors some requests by penalizing other ones, but the overall performance is lowered.
- The Adaptive Piggy-Backing transmission scheme modifies the display rate of the media stream to accomplish merges. As this violates the real-time constraint of media-on-demand transmission schemes (see section **??**), the Adaptive Piggy-Backing scheme is excluded here.
- The different Patching schemes assume a limited receiver storage. Due to restriction two, only Patching with sufficient large storage space and an optimal patching window (calculated according to [EVZ99a, EVZ01] from the average request rate) is used.
- The Tapping schemes are equivalent to Patching if no limit on the storage size of the recipient systems is imposed.
- Virtual Batching and Peer-to-Peer Networks are omitted as they use several recipients as additional senders.

To compare the remaining transmission schemes, they are evaluated using simulated, Poisson distributed client requests of different request rates. Each simulation is run 50 times and the observed interval is chosen each time to contain 5 000 requests to get meaningful results.

For transmission schemes which provide immediate service, the efficiency of the transmission scheme is retrieved by calculating the minimum required bandwidth using the formula given in section 3.1.2 and dividing this value by the average bandwidth of the simulations as described in 3.1. Figure 3.11 shows the results for request rates between 1 and 1 000 request per media stream duration. The results of the Optimal Merge Decision Patching algorithm (see section 3.3.4) have been included in this graph although it requires knowledge of the request times in advance. As this knowledge gives the algorithm an advantage to transmission schemes which have to react on random request patterns, it has a higher efficiency. For low request rates, its "efficiency" is even higher than 100 % as the efficiency calculation assumes a random Poisson process for the distribution of requests without knowledge of future requests.

For the Point-to-Point transmission scheme, a receiver bandwidth equal to the media bit rate has been asserted. Under this precondition, the Point-to-Point transmission scheme has an efficiency of 100 % as it is the optimal (and apart from channel permutations the only) solution which provides immediate service if the receiver systems cannot receive more than the media bit rate. To prove that the Point-to-Point transmission scheme looses its attractiveness if the receiver bandwidth is not limited to the media stream bit rate, the efficiency of the Point-to-Point transmission scheme has been added a second time to the graph for the case that the receiver system is able to receive twice the media bit rate.

Besides of this ones, it is clearly visible that the ERMT Merging algorithm has the highest efficiency of all reactive transmission schemes. Another important point about reactive transmission schemes is that they are less efficient for high media request rates.

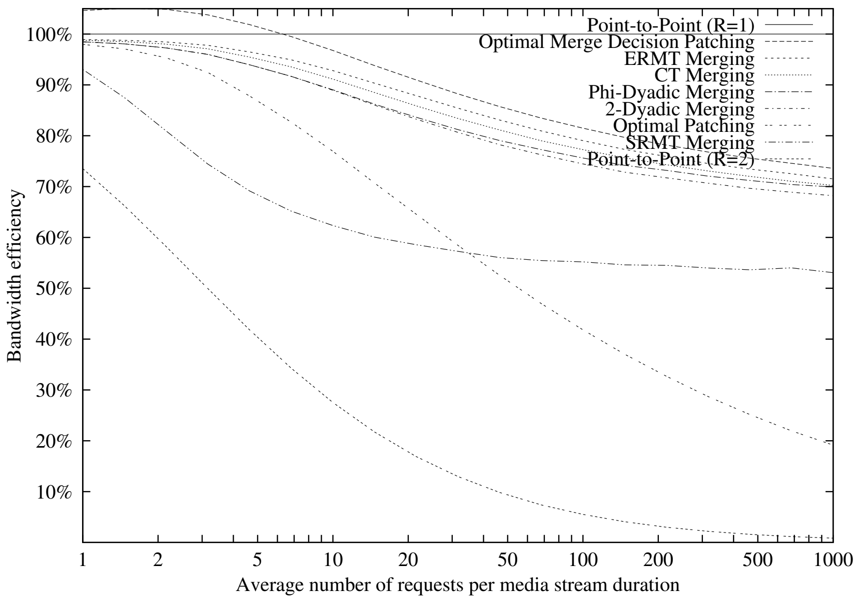


Figure 3.11: Comparison of efficiency of reactive transmission schemes with immediate service

For the Batching transmission scheme which serves requests delayed, the bandwidth requirements are measured for a maximum playback delay of $\frac{1}{120}$ of the media stream duration (which corresponds to a playback delay of one minute for a two hour media stream) and compared to the theoretical minimum to calculate the efficiency of this scheme. Figure 3.12 shows the result of this analysis. As for the Point-to-Point transmission scheme, the Batching transmission scheme requires only a receiver bandwidth equal to the media bit rate and has an efficiency of nearly 100 % under this precondition. As above, the efficiency of the Batching scheme for a receiver bandwidth of twice the media bit rate has been added to the graph to make it comparable with other transmission schemes if this precondition is not necessary.

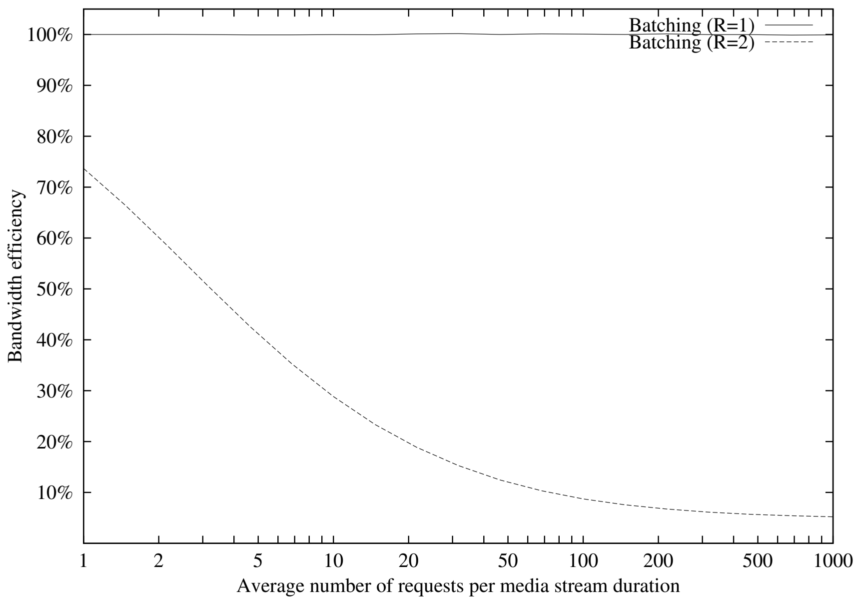


Figure 3.12: Efficiency of reactive transmission schemes with delayed service

3.4 Non-Segmenting Pro-Active Transmission Schemes

In the last section, reactive transmission schemes have been examined. Starting with this section and extending up to section 3.7, the large group of pro-active transmission schemes is examined. In contrast to reactive schemes which transfer media streams (immediately or with some delay) as a result of individual request from consumers, pro-active transmission schemes transmit media streams continuously, independent of any requests. This has several advantages over reactive schemes:

- Playback requests must not be forwarded from the receiver system to the sender system, it is actually possible to drop any communication from the receiver system to the sender system.
- Resource requirements (bandwidth, memory) and service guarantees (playback delay) can be calculated independent of the number of concurrent transmissions and the pattern of client requests.

- The needed server bandwidth for highly requested media streams is lower if a pro-active scheme is used.
- The transmission schedule has not to be created dynamically but can be computed in advance which lowers the real-time requirements of the sender system and removes some complexity.

The main disadvantage of pro-active transmission schemes is that they cannot save bandwidth when no recipient listens to the transmission. Solutions to this problem are addressed in more detail in sections 3.8 and **??**.

The remainder of this section describes two very old and simple pro-active transmission schemes. These transmission schemes always transmit media streams as a whole and are generally only interesting because of their simplicity. Several more advanced schemes are presented in the sections after this one in three groups, sorted by their functioning.

3.4.1 Round-Robin Broadcasting

A simple type of media-on-demand is to transmit one (or several) media streams per transmission channel permanently in a round-robin manner. This type of media-on-demand is very popular in small, ancient media-on-demand systems as sometimes still be found in hotels. Its main advantages are low system requirements: Besides of accounting, the media-on-demand receiver system has no other requirements than a normal TV system, and at the sender side, usual video cassette recorders with automatic rewind or endless tapes can be used. The main disadvantage of Round-Robin Broadcasting is the high playback delay: In worst case, a consumer has to wait nearly one whole media playback duration until the media stream is started again.

Figure 3.13 shows a transmission diagram for a Round-Robin Broadcasting transmission, together with the media playback by one recipient.

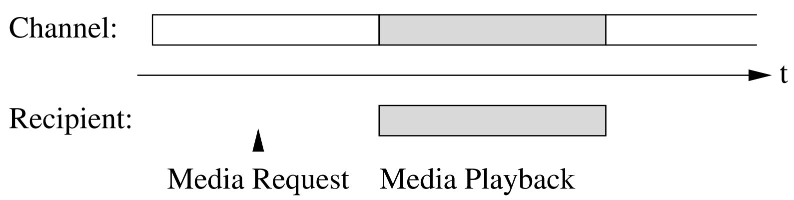


Figure 3.13: Diagram of a Round-Robin Broadcasting transmission.

3.4.2 Staggered Broadcasting

An improvement to the Round-Robin Broadcasting scheme presented above are Staggered Broadcasting transmissions (also known as Baseline transmissions) [PLE03]: For Staggered Broadcasting transmissions, each media stream is transmitted several times in parallel, using different phase shifts for each transmission. As a result, the playback delay is reduces to a fraction equivalent to the reciprocal of the number of used channels.

Another improvement of this transmission scheme is the possibility to navigate within the media stream: By changing to another transmission of the same media stream, the receiver system can jump within the media stream in a limited way (fixed jump interactivity).

Although the improvement over Round-Robin Broadcasting is very high, this transmission scheme requires a lot of sender bandwidth, so the number of transmitted media streams is typically low.

Figure 3.14 shows a transmission diagram for a transmission using three channels for the Staggered Broadcasting scheme, together with the playback by one recipient.

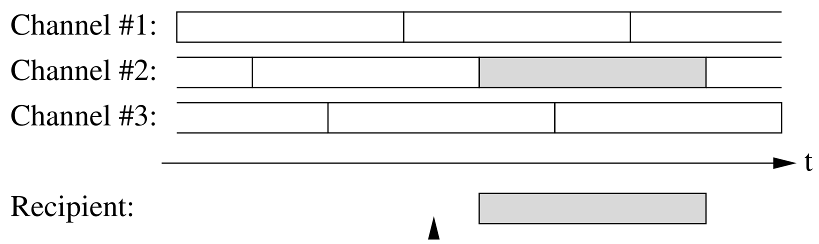


Figure 3.14: Diagram of a staggered transmission.

3.4.3 Comparison of Non-Segmenting Pro-Active Transmission Schemes

Round-Robin Broadcasting transmissions and the Staggered Broadcasting transmission scheme are two very simple but widely used versions for pro-active media-on-demand. Their main advantages lie in the low requirements for the receiver systems and their simplicity for both sender and receiver systems. But for short playback delays, they are practically of no use:

- For the Round-Robin Broadcasting scheme, no modification of the playback delay is possible, the worst case playback delay is always equal to the media stream duration.
- For the Staggered Broadcasting scheme, the required bandwidth is inversely proportional to the playback delay, i. e. the bandwidth at the sender system increases very fast if the playback delay becomes small.

The efficiency of the Round-Robin Broadcasting scheme is 100 % for the single set of supported parameters (playback delay equals stream duration), but as no modification of the playback delay is supported, the use of this transmission scheme is very limited.

Calculating the efficiency for the Staggered Broadcasting scheme depends very much on the prerequisites: Assuming that the receiver system can only receive one channel at media stream bit rate, it is not possible to create a pro-active transmission scheme with lower bandwidth requirements than the Staggered Broadcasting scheme provides, i. e. the efficiency η is 100 %.

But if the receiver systems can receive more than a single channel at media stream bit rate, the efficiency decreases very fast for short playback delays. For example, if a playback delay of $\frac{1}{120}$ of the media stream duration is desired (which equals a playback delay of one minute for a two hour media stream) and the receiver system is able to receive six times the media bit rate (which is the bandwidth needed by an advanced transmission scheme presented later for the same setup), the efficiency of the Staggered Broadcasting scheme is only $\eta = \frac{\Psi_0\left(\frac{1+\frac{1}{120}}{\frac{1}{7200}}\right) - \Psi_0\left(\frac{\frac{1}{120}}{\frac{1}{7200}}\right)}{120} = \frac{\Psi_0(7260)-\Psi_0(60)}{120} \approx 4.0\,\%$[5].

3.5 Size-Based Segmenting Transmission Schemes

The last section introduced the idea of pro-active transmission schemes with two very inefficient approaches. Starting with this section and continued up to section 3.7, a group of advanced, much more efficient, pro-active media-on-demand transmission schemes is examined.

One common fact for this group of schemes is that the media stream is not transmitted sequentially any more. Instead, the media stream is split into several segments, which are broadcast according to a fixed or calculated **transmission schedule** by the sender system. On the receiver system side, any received segments have to be saved for these schemes and must be reassembled when they are needed for playback.

The main advantage of this segmentation procedure is that bandwidth can be saved: As receiver systems do not need the later segments of a media streams until all preceding segments have been played, these segments can be transmitted less frequently.

Three different ways how segmented transmissions are performed have evolved:

- **Size-based approach:** All segments are transmitted in parallel at the same time using the same bandwidth, but the segment sizes are increasing. This way, the size of a segment determines the frequency for the transmission of the data of the segment, and increasing the size for later segments of a stream therefore reduces the transmission frequency of these segments.
- **Bandwidth-based approach:** The media stream is split into segments of equal size, which are transmitted in parallel, using decreasing bandwidths. Using this approach, the bandwidth determines the frequency for the segment transmissions. As the bandwidth for later segments of a media stream is decreased, their transmission frequency is lowered.
- **Frequency-based approach:** The media stream is split into segments of equal size, which are transmitted using the same bandwidth for all segments but only at increasing periodic intervals. As the segment period determines the frequency for the segment transmission in this approach, the frequency for the transmission of later segments of a media stream is decreased by increasing the periods of these segments.

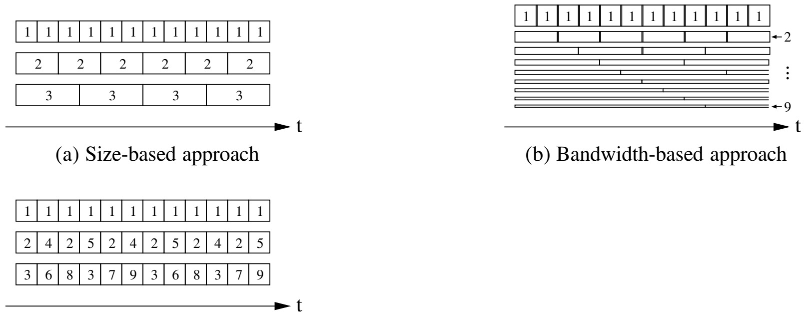
Figure 3.15 illustrates these three different approaches.

The remainder of this section describes transmission schemes which use the size-based approach, while transmission schemes of the bandwidth-based and frequency-based approaches are described in the next two sections 3.6 and 3.7, resp.

3.5.1 Pyramid Broadcasting

The Pyramid Broadcasting scheme [VI96] was the first published transmission scheme which used the segmenting approach. Although its origins lie rather in TV network

[5]This calculation is based on equation (3.10) using a slot interval of $\delta = \frac{d}{7200}$ which equals segments of one second in case of a two hour media stream.



(a) Size-based approach

(b) Bandwidth-based approach

(c) Frequency-based approach

Figure 3.15: Three different approaches for segmented media-on-demand transmissions

based transmissions, it can be used on any other multicast or broadcast capable network.

For the Pyramid Broadcasting scheme, each media stream is split into segments of geometrically increasing size, i. e. the size of the i-th segment is α^{i-1} times the size of the first segment. The parameter α thereby is chosen by the Pyramid Broadcasting Scheme in such a way that the maximum playback delay is minimized. As the maximum playback delay W^+ can be calculated from α and the number of channels N (which is the same as the number of segments n for the Pyramid Broadcasting scheme) by

$$W^+ = d \cdot \frac{\alpha - 1}{\alpha \cdot (\alpha^N - 1)} \tag{3.18}$$

it is only possible to calculate the optimum values for α by applying a numerical minimum search. Table 3.4 shows the results of this calculation for two to eight channels.

Number of channels	α
2	1.7105
3	2.0556
4	2.2397
5	2.3485
6	2.4183
7	2.4664
8	2.5012
∞	$e \approx 2.7182$

Table 3.4: Optimal values for the parameter α of the Pyramid Broadcasting scheme

For the Pyramid Broadcasting scheme, all media streams are transmitted on the same set of channels, thus the total available bandwidth of the sender system is divided into as many channels as segments are used in each media stream (which assumes that all media streams have the same duration).

To receive a media stream, the receiver system has to join the first channel and has to wait for the next transmission of the first segment of the desired media stream. The segment data is then received into a buffer and the playback is started. While playing the segment, the receiver system joins the next channel, waits for the beginning of the next segment and appends it to the buffer. This procedure is continued until all segments have been received and played.

Figure 3.16 shows a media transmission using the Pyramid Broadcasting scheme for three channels and two media streams (one consisting of segments 1 to 3, the other of segments A to C). Please note that the channel bandwidth is much higher than the media bit rate for the Pyramid Broadcasting scheme to work.

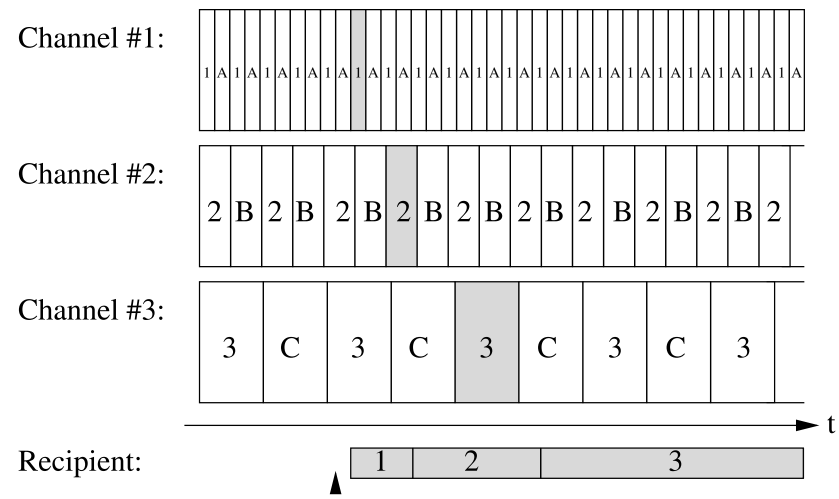


Figure 3.16: Pyramid Broadcasting

3.5.2 Permutation-Based Pyramid Broadcasting

Permutation-based Pyramid Broadcasting [AWY96b] is a direct successor of the Pyramid Broadcasting scheme. The goal of the authors of the Permutation-based Pyramid Broadcasting scheme was to reduce the high bandwidth and disk throughput requirements of the Pyramid Broadcasting scheme at the receiver systems.

Its basic architecture is the same as for Pyramid Broadcasting, but instead of transmitting the segments as single pieces at high bandwidth, they are split into subsegments and interleaved with subsegments of other media streams. By this means, the authors of the Permutation-based Pyramid Broadcasting scheme achieved that the receiver systems do not have to receive each segment as a whole any more but only subsegments at periodic intervals.

Another difference between Permutation-based Pyramid Broadcasting and Pyramid Broadcasting are the formulas which are used to calculate the parameter α of the transmission scheme, i. e. the growth factor of the segments. But as the authors of [HS97] have already shown, this modification degrades the performance of Permutation-based Pyramid Broadcasting below that of Pyramid Broadcasting.

3.5.3 Skyscraper Broadcasting

Skyscraper Broadcasting [HS97, HS00] uses the same size-based approach as Pyramid Broadcasting, but introduces several improvements. Firstly, only one media stream is considered for each transmission, i. e. different media streams are sent independently using different channels. This not only removes some unnecessary complexity from the transmission scheme, it also reduces the bandwidth requirements of the receiver systems.

Another progress is that the authors of the Skyscraper Broadcasting scheme reduced the analysis of size-based transmission schemes to the examination and comparison of different **broadcasting series**, where each broadcasting series simply describes the (relative) sizes of the segments of the examined transmission scheme. For example, the broadcasting series of Pyramid Broadcasting is the geometric series $1, \alpha, \alpha^2, \alpha^3, \ldots$ because the size of the i-th segment is α^{i-1} times the size of the first segment for this transmission scheme.

For Skyscraper Broadcasting, a new broadcasting series has been introduced which is recursively defined by the following function:

$$S(i) = \begin{cases} 1 & \text{if } i = 1, \\ 2 & \text{if } i = 2 \text{ or } i = 3, \\ 2 \cdot S(i-1) + 1 & \text{if } i \bmod 4 = 0, \\ S(i-1) & \text{if } i \bmod 4 = 1, \\ 2 \cdot S(i-1) + 2 & \text{if } i \bmod 4 = 2, \\ S(i-1) & \text{if } i \bmod 4 = 3 \end{cases} \tag{3.19}$$

The broadcasting series generated by this function starts with

$$1, 2, 2, 5, 5, 12, 12, 25, 25, 52, 52, 105, 105, 212, 212, 425, \ldots \tag{3.20}$$

and results in shorter playback delays than the geometric series of Pyramid Broadcasting.

Another advantage of Skyscraper Broadcasting is that the broadcasting series can be adjusted to the receiver systems storage size: If an upper bound is applied to the elements of the broadcasting series, the required storage of the receiver system can be limited (at the cost of a longer playback delay or higher sender bandwidth requirements). In section **??**, this approach is revived in the context of a frequency-based transmission scheme.

Skyscraper Broadcasting uses channels which have the same bandwidth as the media stream to transmit[6] instead of high bandwidth channels which are required for Pyramid Broadcasting and Permutation-based Pyramid Broadcasting. Of these channels, the receiver systems must be able to receive two channels simultaneously for Skyscraper Broadcasting transmissions, but as the channel bandwidth is much lower than for the Pyramid Broadcasting scheme, the bandwidth requirements have been much relaxed for the receiver systems.

Figure 3.17 shows a media transmission using the Skyscraper Broadcasting scheme for five channels.

[6] Skyscraper Broadcasting does not support media streams which use a variable-bit-rate encoding.

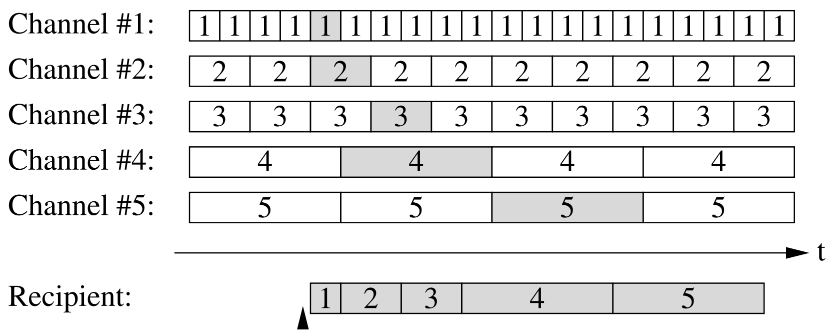


Figure 3.17: Skyscraper Broadcasting

3.5.4 Mayan Temple Broadcasting

The Mayan Temple Broadcasting scheme [PLM99] generalizes the Skyscraper Broadcasting scheme in mainly two points: Firstly, it does not use a fixed broadcasting series but a function which calculates the size of the segments from the cumulative bandwidth function of the media stream[7]. The major advantage of this approach is that the Mayan-Temple Broadcasting scheme can support variable-bit-rate media streams. As a consequence, the channel bandwidth can be selected independently from the media stream bit rate, even for constant-bit-rate media streams.

Secondly, the Mayan Temple Broadcasting scheme supports a mechanism called **partial preloading**. Partial preloading means that a part (sensibly the beginning) of the media stream is transmitted to the recipient system before the client requests the media stream. For a media-on-demand system this means that the beginnings of all media streams have to be transmitted periodically on an additional channel and have to be saved by the recipient systems for later playback. The advantages and drawbacks of partial preloading will be reviewed in section **??**.

The formula for the duration δ_i of segment $i \geq 2$ of the Mayan Temple Broadcasting scheme is given as:

$$\delta_i = f^{-1}\left(f\left(W^+ + \sum_{j=2}^{i-1}\delta_j\right) + \beta \cdot \left(W^+ + \sum_{j=2}^{i-1}\delta_j\right)\right) - W^+ + \sum_{j=2}^{i-1}\delta_j \tag{3.21}$$

where f is the cumulative bandwidth function (see section **??**), f^{-1} is the inverse cumulative bandwidth function (see section **??**) and β is the channel bandwidth. This formula can be reduced to

$$\delta_i = W^+ \cdot \sum_{j=1}^{i-1}\left(\frac{\beta}{\beta_1}\right)^j \cdot \binom{i-2}{j-1} \tag{3.22}$$

if all segments have the same average bit rate β_1 (which is esp. true for constant-bit-rate media streams)[8].

[7] The cumulative bandwidth function describes the characteristics of the bit rate variations of a variable-bit-rate media stream. A definition of the cumulative bandwidth function is presented in section **??**.

[8] Please take care when reading the article about the Mayan Temple Broadcasting scheme [PLM99] as in their formula the binomial coefficient is accidentally missing.

For a constant-bit-rate media stream and channels at media stream bit rate, this formula generates the following broadcasting series:

$$(1), 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1\,024, 2\,048, 4\,096, 8\,192, 16\,384, \ldots \tag{3.23}$$

The first segment from this series is typically transmitted in advance (i. e. preloaded) for this transmission scheme (denoted by the parenthesis in the broadcasting series) but may also be transmitted on a separate channel if preloading is not used.

3.5.5 Client-Centric Approach

The Client-Centric Approach transmission scheme [HCS98a] has its origins at the Skyscraper Broadcasting scheme, but it allows to specify the number of channels R which a receiver system can join simultaneously. It uses another broadcasting series, which is defined by the following function[9]:

$$\begin{aligned} S_R(i) &= \begin{cases} 1 & \text{if } i = 1, \\ 2 \cdot S(i-1) & \text{if } i \bmod R \neq 1, \\ S(i-1) & \text{if } i \bmod R = 1 \end{cases} \\ &= 2^{i - \lceil \frac{i}{R} \rceil} \end{aligned} \tag{3.24}$$

For $R = 2, \ldots, 5$, the broadcasting series of this scheme are

$$\begin{aligned} R = 2: &\quad 1, 2, 2, 4, 4, 8, 8, 16, 16, 32, 32, 64, 64, 128, 128, 256, \ldots \\ R = 3: &\quad 1, 2, 4, 4, 8, 16, 16, 32, 64, 64, 128, 256, 256, 512, 1\,024, 1\,024, \ldots \\ R = 4: &\quad 1, 2, 4, 8, 8, 16, 32, 64, 64, 128, 256, 512, 512, 1\,024, 2\,048, 4\,096, \ldots \\ R = 5: &\quad 1, 2, 4, 8, 16, 16, 32, 64, 128, 256, 256, 512, 1\,024, 2\,048, 4\,096, 4\,096, \ldots \end{aligned} \tag{3.25}$$

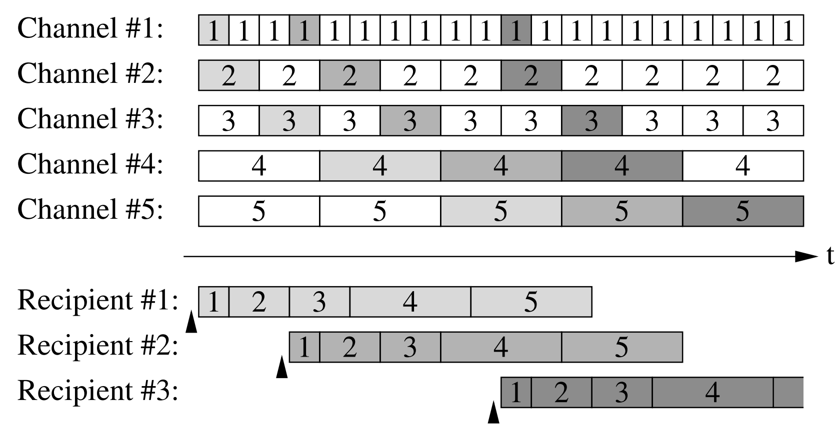


Figure 3.18: Client-Centric Approach

[9] The recursive version of the formula in [HCS98a] is unfortunately wrong, but the closed form is correct.

3.5.6 Greedy Disk-Conserving Broadcasting

The Greedy Disk-Conserving Broadcasting scheme [GKT02] is another successor of the Skyscraper Broadcasting scheme. Similar to the Client-Centric Approach, it allows to create a broadcasting series for receiver systems which can only receive a limited number of R channels simultaneously. The used broadcasting series is defined by the following function for $R = 2$:

$$S_2(i) = \begin{cases} 2^{i-1} & \text{if } i \leq 3, \\ 4 & \text{if } i = 4, \\ 10 & \text{if } i = 5 \text{ or } i = 6, \\ 24 & \text{if } i = 7 \text{ or } i = 8, \\ 5 \cdot S_2(i-4) & \text{if } i \geq 9 \end{cases} \tag{3.26}$$

and the following one for $R \geq 3$:

$$S_R(i) = \begin{cases} 2^{i-1} & \text{if } i \leq R+1, \\ \left\lfloor \frac{\sum_{k=i-R}^{i-1} S_R(k)}{S_R(i-R)} \right\rfloor \cdot S_R(i-R) & \text{if } i > R+1 \end{cases} \tag{3.27}$$

For $R = 2, \ldots, 5$, this gives the following broadcasting series:

$$\begin{aligned} R = 2: &\quad 1,2,4,4,10,10,24,24,50,50,120,120,250,250,600,600,\ldots \\ R = 3: &\quad 1,2,4,8,14,24,40,70,120,200,350,600,1\,000,1\,750,3\,000,5\,000,\ldots \\ R = 4: &\quad 1,2,4,8,16,30,56,104,192,360,672,1\,248,2\,304,4\,320,8\,064,14\,976,\ldots \\ R = 5: &\quad 1,2,4,8,16,32,62,120,232,448,864,1\,674,3\,240,6\,264,12\,096,23\,328,\ldots \end{aligned} \tag{3.28}$$

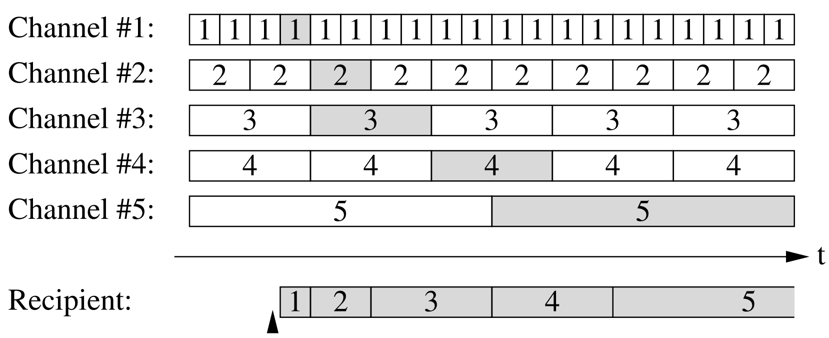


Figure 3.19: Greedy Disk-Conserving Broadcasting

3.5.7 Greedy Equal Bandwidth Broadcasting

With the Greedy Equal Bandwidth Broadcasting scheme [HNvB99], the authors wanted to get the most efficient transmission scheme of the size-based, segmenting group. Therefore, the authors of this scheme proved that their scheme uses the best bandwidth to channel division (which is an equal bandwidth division) and best segment sizes for the size-based class of broadcasting schemes. To achieve this maximum of efficiency, the Greedy Equal Bandwidth Broadcasting requires that the receiver systems can join all used transmission channels at once.

One major advantage of the Greedy Equal Bandwidth Broadcasting scheme to the previously presented schemes is that the playback delay can be specified independently of the segment sizes. This way, the number of segments can be increased without changing the playback delay, only the channel bandwidth, the number of channels and the size of the segments change when the number of segments is modified. This idea is reviewed again in section **??**.

Another change to the above described transmission schemes is the starting point of data reception: Instead of waiting for the beginning of the next segment after joining a channel, the receiver systems have to receive any data immediately. By starting the reception immediately (an idea which has been proposed for the Polyharmonic Broadcasting scheme which is presented in section 3.6.4), the broadcasting scheme becomes more efficient because segments can be played at the receiver systems immediately after their transmission period has elapsed, even if the channel bandwidth is lower than the media bit rate. (If the receiver system awaits the beginning of the segment instead, the segment cannot be played if it is received concurrently to its playback and if the channel bandwidth is lower than the media bit rate because the receiver system would run out of data.) Again, this idea is analyzed in more detail later in this thesis in section **??**.

The broadcasting series of the Greedy Equal Bandwidth Broadcasting scheme is defined by the following function:

$$S(i) = \left(\frac{d}{W^+} + 1\right)^{\frac{i-1}{n}} \tag{3.29}$$

where d is the total length of the media stream and W^+ is the desired maximum playback delay.

If the playback delay is set to the playback duration of the first segment (for better comparison with the previous schemes), this formula gives the following broadcasting series:

$$1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1\,024, 2\,048, 4\,096, 8\,192, 16\,384, 32\,768, \ldots \tag{3.30}$$

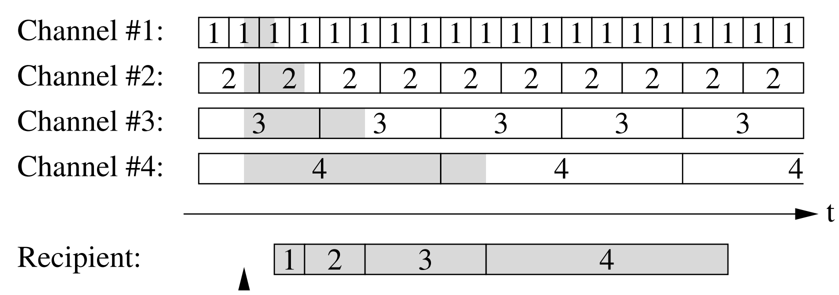


Figure 3.20: Greedy Equal Bandwidth Broadcasting

3.5.8 Fibonacci Broadcasting

Fibonacci Broadcasting [YK03] is a transmission scheme which uses the Fibonacci series (leaving out the first element) as broadcasting series, so its broadcasting series is defined by the following function:

$$S(i) = \begin{cases} 1 & \text{if } i = 1, \\ 2 & \text{if } i = 2, \\ S(i-2) + S(i-1) & \text{if } i \geq 3 \end{cases} \tag{3.31}$$

which gives the following broadcasting series:

$$1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1\,597, \ldots \tag{3.32}$$

Using Fibonacci Broadcasting, the receiver systems only have to receive two channels at a time.

Similar to the Greedy Equal Bandwidth Broadcasting scheme, this transmission scheme uses the efficiency advantage that is gained if the receiver systems receive data from the channels immediately after joining them instead of waiting for the next segment beginnings.

Figure 3.21 shows an example transmission using the Fibonacci Broadcasting scheme.

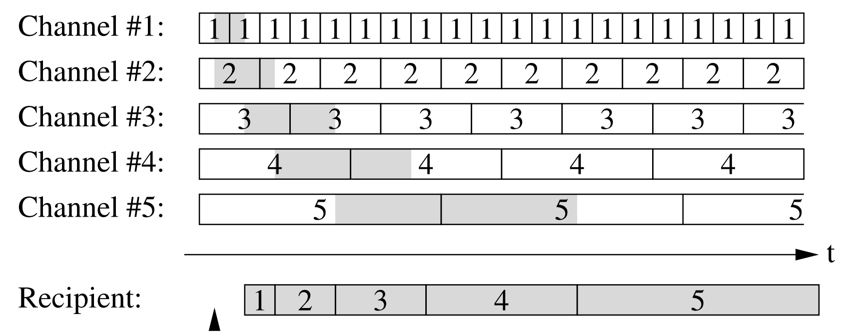


Figure 3.21: Fibonacci Broadcasting

3.5.9 Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting

The Reliable Periodic Broadcasting scheme [MEVSS01], which has also been published independently by the authors of the Fibonacci Broadcasting scheme some time later as Generalized Fibonacci Broadcasting scheme [YK03], can be seen as a successor of the Fibonacci Broadcasting scheme. Compared to the Fibonacci Broadcasting scheme, it has been generalized in such a way that the channel bandwidth and the number of channels a receiver system can receive simultaneously can be configured.

The broadcasting series of the Reliable Periodic Broadcasting scheme/Generalized

Fibonacci Broadcasting scheme can be determined from the following formula:

$$S(i) = \begin{cases} 1 + \dfrac{\sum\limits_{k=1}^{i-1} S(k)}{g} & \text{if } i \leq R, \\ \dfrac{\sum\limits_{k=i-R}^{i-1} S(k)}{g} & \text{if } i > R \end{cases} \tag{3.33}$$

where g is a parameter which specifies the bandwidth of the transmission channels, i. e. the bandwidth of the channels is $\frac{1}{g}$ times the media bit rate.

For $g = 1$ (for better comparison), this gives the following broadcasting series:

$$\begin{array}{ll} R=2: & 1,2,3,5,8,13,21,34,55,89,144,233,377,610,987,1\,597,2\,584,\ldots \\ R=3: & 1,2,4,7,13,24,44,81,149,274,504,927,1\,705,3\,136,5\,768,10\,609,\ldots \\ R=4: & 1,2,4,8,15,29,56,108,208,401,773,1\,490,2\,872,5\,536,10\,671,20\,569,\ldots \\ R=5: & 1,2,4,8,16,31,61,120,236,464,912,1\,793,3\,525,6\,930,13\,624,26\,784,\ldots \end{array} \tag{3.34}$$

As one can see, $g = 1$ and $R = 2$ gives the broadcasting series of the Fibonacci Broadcasting scheme and $g = 1$ and $R = \infty$ gives the broadcasting series of the Greedy Equal Bandwidth Broadcasting scheme.

Again, this transmission scheme requires that the receiver systems receive data from the channels immediately after joining them.

Figure 3.22 shows an example transmission using the Reliable Periodic Broadcasting scheme/Generalized Fibonacci Broadcasting scheme using channels of half the media bit rate and a receiver bandwidth of three channels ($g = 2$, $R = 3$).

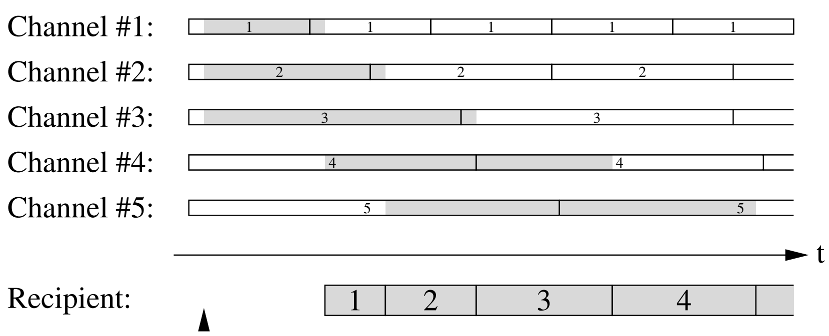


Figure 3.22: Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting

3.5.10 Comparison of Size-Based Segmenting Transmission Schemes

Comparing all size-based transmission schemes with each other is very difficult as the transmission schemes differ in several points:

- The number of channels which have to be received simultaneously by the client systems are not the same for all transmission schemes: For Pyramid Broadcasting and Permutation-based Pyramid Broadcasting, just one channel has to be received at once, for Skyscraper Broadcasting and Fibonacci Broadcasting, two channels must be received in parallel, for Mayan Temple Broadcasting and Greedy Equal Bandwidth Broadcasting, all channels have to be received simultaneously and using the Client-Centric Approach, the Greedy Disk-Conserving Broadcasting or the Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting, it is possible to specify the number of receiver channels.

- For the Pyramid Broadcasting and the Permutation-based Pyramid Broadcasting scheme, the bandwidth of the channels is higher than the media bit rate while all other transmission schemes allow to use a channel bandwidth equal to the media bit rate. The Mayan Temple Broadcasting scheme and the Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting scheme even allow to specify a channel bandwidth independent of the media stream bit rate.
- The Mayan Temple Broadcasting scheme supports variable-bit-rate media streams, all other transmission schemes only support constant-bit-rate media streams.
- The Skyscraper Broadcasting scheme and the Mayan Temple Broadcasting scheme allow to specify a limit for the required storage of the receiver system.
- The Mayan Temple Broadcasting scheme supports partial preloading.
- The Greedy Equal Bandwidth Broadcasting scheme allows to define the playback delay independent of the size of the first segment.

To get an overview of the efficiency of all size-based transmission schemes, most of these features have to be ignored and only a few setups are examined: In a first setup, the receiver system can only receive two channels equal to the media bit rate, while in a second setup, no limit is imposed on the receiver bandwidth. To include Pyramid Broadcasting and Permutation-based Pyramid Broadcasting in this comparison, a third setup is examined where the receiver system is allowed to receive a channel of a bandwidth higher than the media bit rate[10]. In all setups, the receiver storage is pretended to be sufficiently large and neither partial preloading nor variable-bit-rate media streams are used in any of the setups.

Figures 3.23 to 3.25 show the results for each of the three setups, resp. The sawtooth-like shape of the curves is a result of the low granularity of the playback delays which are supported by all schemes of the size-based class: If the bandwidth is increased slowly and the transmission scheme cannot benefit from this (as it still provides the same playback delay as before), the calculated efficiency decreases. Each time when a bandwidth is reached which allows to switch to a shorter playback delay, the efficiency increases immediately and produces a sawtooth in the graph.

For the two-channel setup (see figure 3.23), the Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting scheme performs best if it is used with a parameter $g \geq 2$. The reason for this is that the Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting scheme allows to increase the number of segments independent from the sender and receiver bandwidth which allows to approximate the transmission frequency of the transmitted parts more exactly near to the theoretically needed frequency.

[10] The channel bandwidth of Pyramid Broadcasting and Permutation-based Pyramid Broadcasting varies depending on the total bandwidth and the calculated α-parameter. As the efficiency values of both the Pyramid Broadcasting scheme and the Permutation-based Pyramid Broadcasting scheme are very poor, this is not examined in more detail in this thesis.

If the receiver system is able to receive the complete transmission at once (see figure 3.24), the Greedy Equal Bandwidth Broadcasting scheme and the Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting scheme produce nearly the same results (if the channel bandwidth of the Greedy Equal Bandwidth Broadcasting scheme is set to $\frac{1}{g}$ of the media stream bit rate where g is the parameter of the Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting scheme) except for the fact that the Greedy Equal Bandwidth Broadcasting scheme does not produce as high sawtooth-like shapes as the other scheme because the playback delay can be modified independent of segment sizes. The Greedy Disk Conserving Bandwidth scheme and the Client Centric Approach deliver the same results as the Reliable Periodic Broadcasting/Generalized Fibonacci Broadcasting scheme with $g = 1$.

From figure 3.25 it is also clearly visible that the Pyramid Broadcasting scheme and the Permutation-based Pyramid Broadcasting scheme perform much worse than the other size-based schemes.

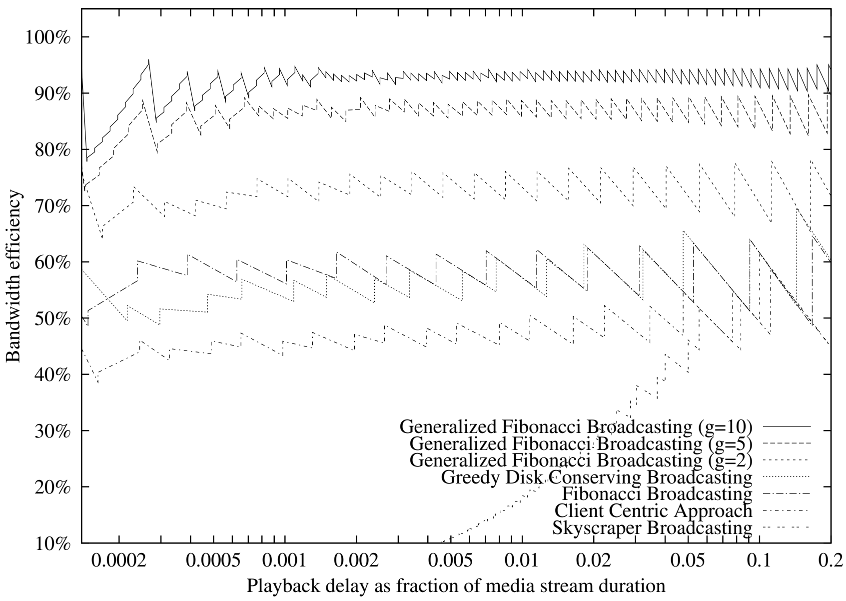


Figure 3.23: Efficiency of size-based schemes when only twice the media bit rate can be received

3.6 Bandwidth-Based Segmenting Transmission Schemes

In the previous section, the size-based transmission schemes improved the efficiency of the media-on-demand transmissions by increasing the size of later segments and therefore reducing the transmission frequency of these segments. The bandwidth-based transmission schemes use another approach to achieve a similar effect: Similar to the size-based transmission schemes, they use one channel for the transmission of each segment, but instead of increasing the segment sizes for later segments, segments of

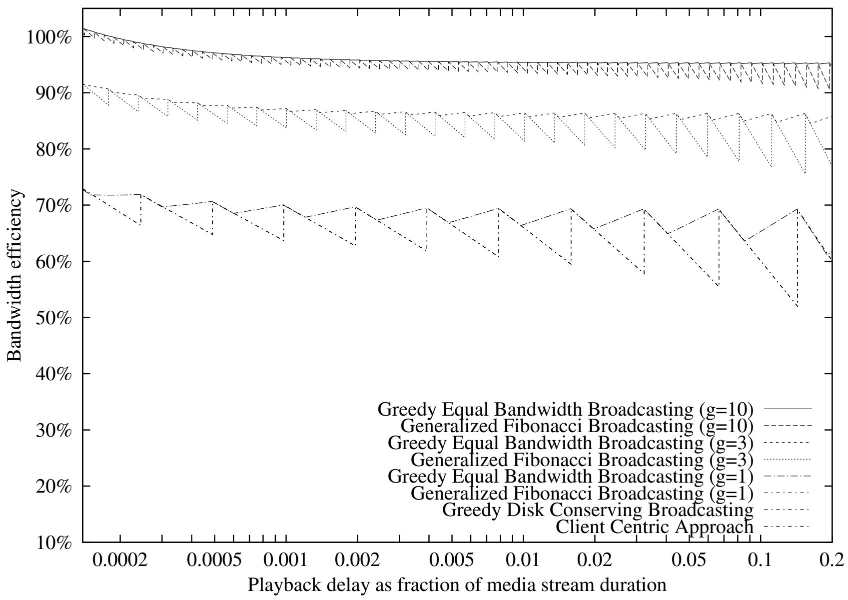


Figure 3.24: Efficiency of size-based schemes when the whole transmission can be received at once

equal size are used and the bandwidth of the channels is decreased. The result of this bandwidth modification is that later segments take longer to be transmitted, so their transmission frequency is lowered this way.

The remainder of this section describes transmission schemes which benefit from this approach. The first published scheme of this class is especially interesting as it supplies some mathematical background for the calculation of a lower bound for on-demand transmissions, although the scheme is not functional [PCL98b]. It is followed by a series of transmission schemes which correct this problem in different ways or focus on some other issues.

3.6.1 Harmonic Broadcasting

The Harmonic Broadcasting scheme [JT97a] uses segments of equal size and a very simple formula for the bandwidth of the channels on which the segments are transmitted:

$$B_i = \frac{b}{i} \tag{3.35}$$

Figure 3.26 shows an example transmission using the Harmonic Broadcasting scheme.

The idea behind this simple formula is that segment i can be transmitted at one i-th of the media bit rate because the receiver system has i times the duration of the first segment to receive segment #i before it is needed for playback.

The above formula assures that the segment start is received when the playback time of the segment is reached and that the segment has been completely received when its playback time ends. But unfortunately, this does not ensure that the segment can be played as is illustrated in figure 3.27, a problem which results from the fact that

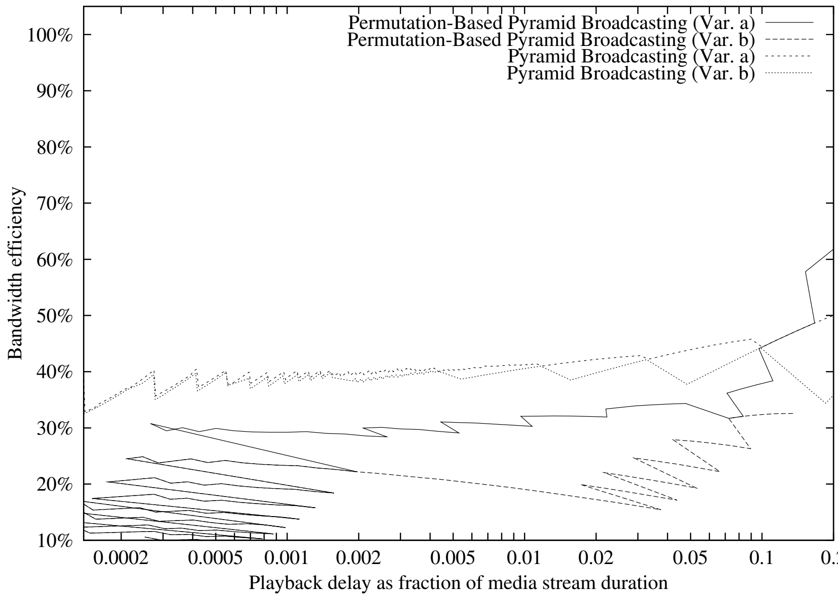


Figure 3.25: Efficiency of size-based schemes where the receiver has to receive two high-bandwidth channels

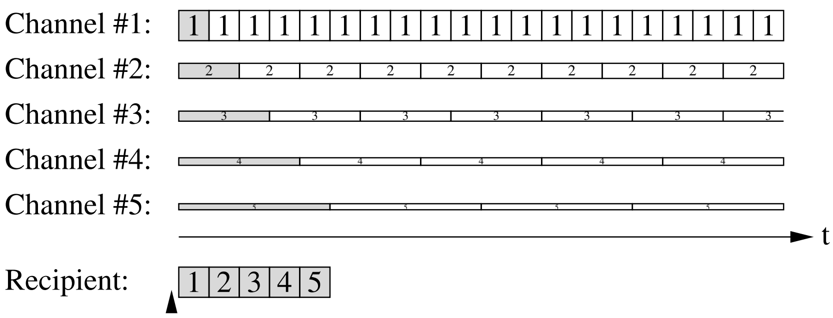


Figure 3.26: Harmonic Broadcasting

the channel bandwidth is lower than the media bit rate. A simple way to correct this problem is to double the playback waiting time at the receiver side, but this decreases the efficiency of the transmission scheme a lot. In the following subsections, other solutions are proposed.

Although this means that the transmission scheme is non-functional, it marks a very important step in the evolution of transmission schemes as it introduced a new and very efficient class of transmission schemes.

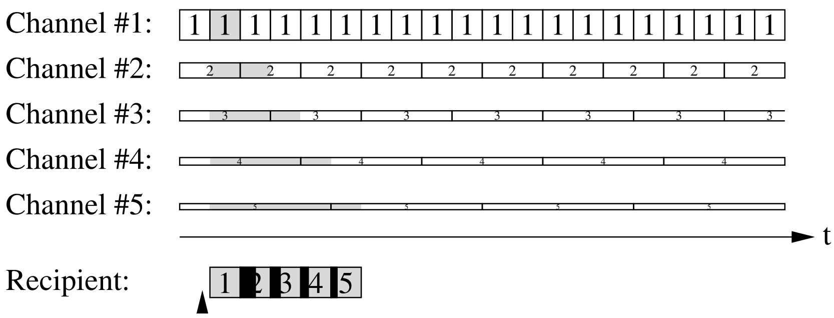


Figure 3.27: Example for failure of Harmonic Broadcasting: The blackened parts at the recipient are missing when the media stream is played

3.6.2 Cautious Harmonic Broadcasting

The authors of [PCL98b] not only discovered that the Harmonic Broadcasting Protocol is not working in the proposed version, they also provided three broadcasting schemes as solutions which are presented in this and the following two subsections.

The Cautious Harmonic Broadcasting [PCL98b] scheme is one of their solutions. For this transmission scheme, the media stream is cut into segments of equal size, but the channel assignment is changed slightly if compared to the Harmonic Broadcasting scheme: On the first channel, the first segment is transmitted at full bandwidth. The second channel alternately transmits the second and third segment at full bandwidth (which is — strictly speaking — against the rules of bandwidth based transmission schemes but these two segments form the only exception). Channels three to $n-1$ are used to transmit the remaining segments four to n at decreasing bandwidth, using one $(i-1)$-th of the media stream bit rate as bandwidth for the transmission of the $(i+1)$-th segment on the i-th channel.

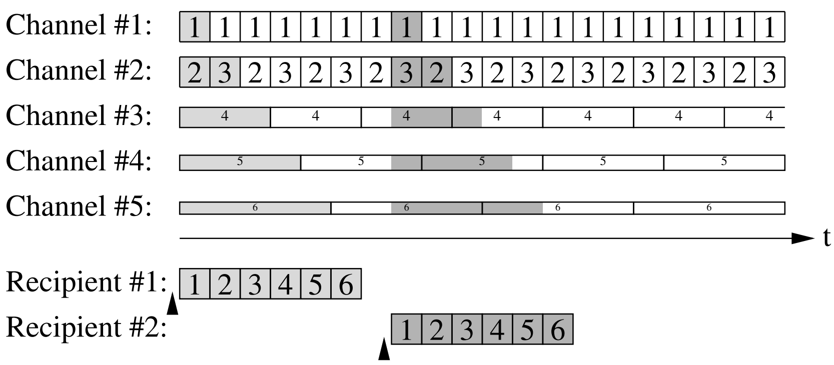


Figure 3.28: Cautious Harmonic Broadcasting

3.6.3 Quasi-Harmonic Broadcasting

The Quasi-Harmonic Broadcasting [PCL98b] scheme is another solution to solve the problems of the Harmonic Broadcasting Protocol.

For the Quasi-Harmonic Broadcasting scheme, the media stream is not only cut into segments of equal size, all segments except the first one are further cut into smaller subsegments and transmitted using a complex scheme:

The first segment is simply transmitted on the first channel at the full media stream bit rate. But channel $i \geq 2$ transmits all subsegments of the segment i at $\frac{m}{i \cdot m - 1}$ times the media stream bit rate in the order specified by the series

$$\left(\begin{cases} i \cdot (t \bmod m) + \left\lfloor \frac{t \bmod im}{m} \right\rfloor & \text{if } t \bmod m \neq m-1, \\ (t \bmod (i-1)) + 1 & \text{if } t \bmod m = m-1 \end{cases} \right)_{t \in \{0,1,2,\ldots,i \cdot m - 1\}} \tag{3.36}$$

where $m \geq 2$ is a parameter. The advantage of the Quasi-Harmonic Broadcasting scheme is that it converges to the same bandwidth requirements as the non-functional Harmonic Broadcasting scheme if m tends towards infinity. Figure 3.29 shows an example transmission for the Quasi-Harmonic Broadcasting scheme.

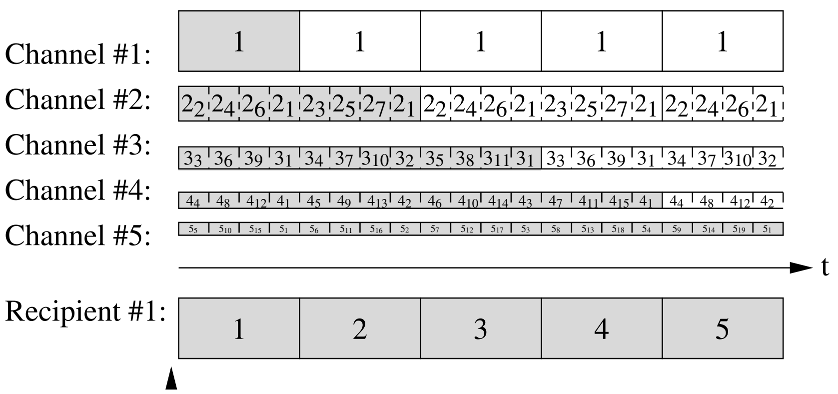


Figure 3.29: Quasi-Harmonic Broadcasting with parameter $m = 4$

3.6.4 Polyharmonic Broadcasting

The Polyharmonic Broadcasting [PCL98c] scheme uses another trick to get around the error of Harmonic Broadcasting: Instead of starting the playback when the beginning of the first segment is received (and therefore the next slot interval boundary is reached), Polyharmonic Broadcasting uses the fixed wait policy which already has been presented in section 3.5.7 for the Greedy Equal Bandwidth Broadcasting scheme, i. e. segments are received immediately after joining the channel (instead of waiting for the next segment boundary) and the playback is started a fixed time after the receiver system started reception (and not at the next segment boundary). This way, Polyharmonic Broadcasting assures that each segment has been completely received when its playback is started.

As the bandwidth division and segment mapping are the same as of Harmonic Broadcasting, Polyharmonic Broadcasting reaches the bandwidth promises of Harmonic Broadcasting. The only difference is that the minimum playback delay equals to the maximum playback delay for the Polyharmonic Broadcasting scheme due to the fixed wait policy, most other broadcasting schemes (without fixed wait policy) have a minimum playback delay of zero.

Another extension of the Polyharmonic Broadcasting scheme is that the playback delay can be selected as an arbitrary multiple of the segment size (shown as parameter *m* in figure 3.35 at the end of this section). This allows to reduce the segment size while keeping the playback delay constant and thereby reducing the total bandwidth requirement. The consequences of this idea are examined in more detail in sections **??** and **??**.

The authors of [ES01, ES02] proved that Polyharmonic Broadcasting has the lowest possible bandwidth requirements for a given segment size, so its efficiency equals 100 % for a given segment size. Figure 3.30 shows an example transmission for the Polyharmonic Broadcasting scheme.

Polyharmonic Broadcasting is also able to transmit variable-bit-rate media streams: In this case, the duration for the transmission of a segment is determined from its playback time (or decode time if this is lower) and the bandwidth of the channel of the segment is calculated according to the size of the segment [Hu01].

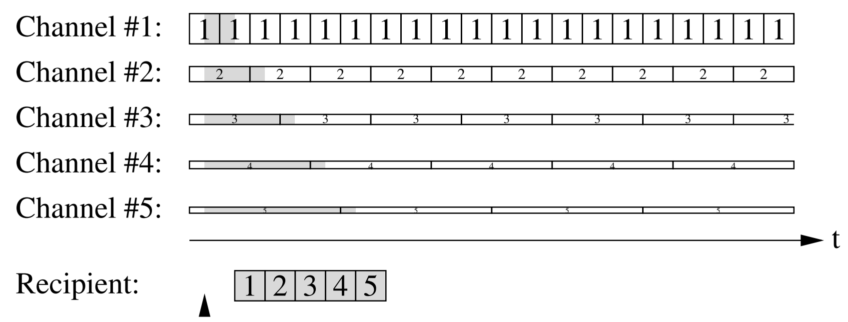


Figure 3.30: Polyharmonic Broadcasting

3.6.5 Tailored Transmissions

The Tailored Transmission scheme [BM99] can be seen as an extension of the Polyharmonic Broadcasting scheme. It uses the same basic idea to circumvent the problems of Harmonic Broadcasting and allows to decouple the playback delay from the slot interval but introduced more flexibility at the same time:

- The transmission scheme supports variable-bit-rate media streams efficiently. The underlying idea of the Tailored Transmission scheme for supporting variable-bit-rate media streams is to use different bandwidths for each segment which are chosen in such a way that the segments are transmitted at the lowest possible period. This idea can indeed be used for any of the transmission schemes of the bandwidth-based class although the Tailored Transmission scheme is the only one which describes this approach. Further information on variable-bit-rate transmissions is given in section **??**.
- The transmission scheme allows to select all but one of the design parameters media duration length, playback delay, receiver bandwidth, transmission band-

width and receiver storage requirements and throughput to create a matching transmission scheme.

3.6.6 Staircase Broadcasting

Staircase Broadcasting [JT97b] is another transmission scheme of the class of bandwidth-based transmission schemes. In contrary to the above ones, it is based on a binary splitting of the bandwidth, i. e. segments $\#2^k$ to $\#2^{k+1}-1$ are transmitted at $\frac{1}{2^k}$ of the media bit rate ($k \geq 0$). As a result, the channels of the transmission can be aggregated into channels at media bit rate.

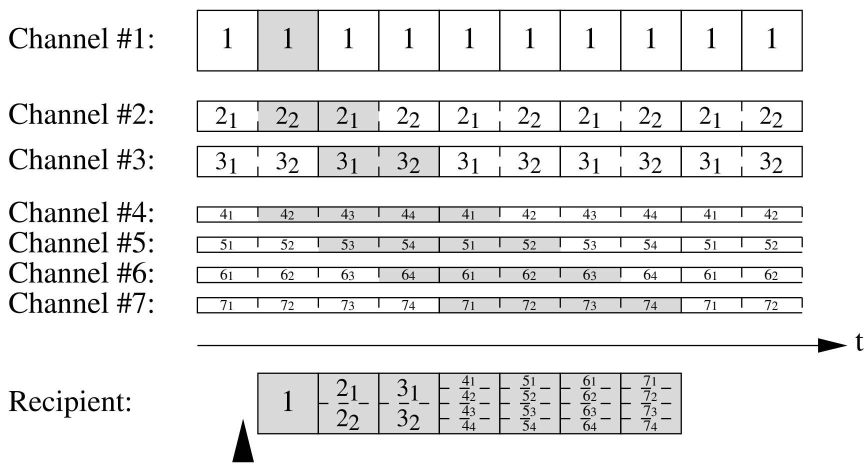


Figure 3.31: Staircase Broadcasting

To circumvent the problems of Harmonic Broadcasting, the authors of the Staircase Broadcasting scheme used another type of subsegment splitting: Instead of splitting a segment into subsegments of same duration to transmit them on low bandwidth channels (**vertical partitioning**), they split the segment transmission into streams of same bandwidth (**horizontal partitioning**). This allows to play a segment even when the first subsegment is just received — which was the problem of Harmonic Broadcasting. Figure 3.32 shows this difference for a two segment transmission.

Although this splitting technique seems to be a completely new idea, it has already been proposed similarly in a previous transmission scheme. To understand this, it is important to realize that subsegments of same bandwidth are produced by interleaving small parts of the segments into the subsegments. But this introduces an additional, small playback delay as the first part of the first subsegment has to be played at full speed while it is received with lower bandwidth. This problem can be solved if the first parts of the first subsegments of the segments are transmitted more often. Exactly this approach is used in the Quasi-Harmonic Broadcasting scheme, where the parameter m describes in how many parts a subsegment is split.

3.6.7 Seamless Staircase Broadcasting

The Seamless Staircase Broadcasting scheme [TCS04] has a very similar structure as the Staircase Broadcasting scheme which has been proposed in the previous subsection.

(a) Subsegment of same duration produce problems at Harmonic Broadcasting

(b) Subsegments of same bandwidth can be used for instead

Figure 3.32: Comparison of different sub-segmentation schemes

Similar to the Staircase Broadcasting scheme, the segments are transmitted for this transmission scheme in such a way that the channels can be aggregated into channels at media bit rate.

The main difference between the Seamless Staircase Broadcasting scheme and the Staircase Broadcasting scheme is that the Seamless Staircase Broadcasting scheme does not use the binary horizontal partitioning scheme for the first three channels. Instead, the following schedule is used:

- The first segment is transmitted at media bit rate.
- The second and third segment are transmitted at media bit rate alternately.
- Segment $\#3 \cdot 2^k + 1$ to $\#3 \cdot 2^{k+1}$ are transmitted at $\frac{1}{3 \cdot 2^k}$ of the media bit rate ($k \geq 0$).

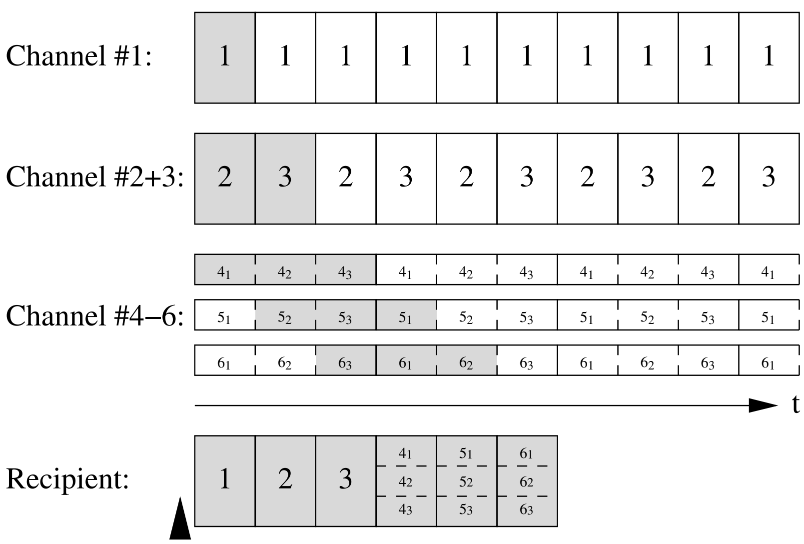


Figure 3.33: Seamless Staircase Broadcasting

Although this transmission scheme is less efficient than the original Staircase Broadcasting scheme, the Seamless Staircase Broadcasting scheme can be regarded as an enhanced version of the former one as it supports a **seamless bandwidth change**: Often the media-on-demand providers want be able to change the bandwidth assignments to media streams to increase or decrease the playback delay dynamically As ongoing transmissions should not be disrupted, the only chance to perform these changes for most transmission schemes is to start the transmission with the new parameters in parallel to the ongoing one and “fading out” the old transmission. The disadvantage of this procedure is that both transmissions consume bandwidth during the transition time.

With the Seamless Staircase Broadcasting scheme, the bandwidth can be increased dynamically (in steps of the media bit rate): Each time the bandwidth is increased by the media bit rate, the playback delay halves a short time thereafter[11]. It is even possible to double the playback delay at a later time up to the initial value, thereby releasing bandwidth after some transition time.

Figure 3.34 shows an example where bandwidth is dynamically added, including an ongoing playback (recipient #1 in the figure) which continues uninterrupted after the bandwidth change and a playback (recipient #2) which is served after the bandwidth has been changed. Seamless bandwidth changes are re-examined in section **??** as an extension to the newly proposed Generalized Greedy Broadcasting scheme.

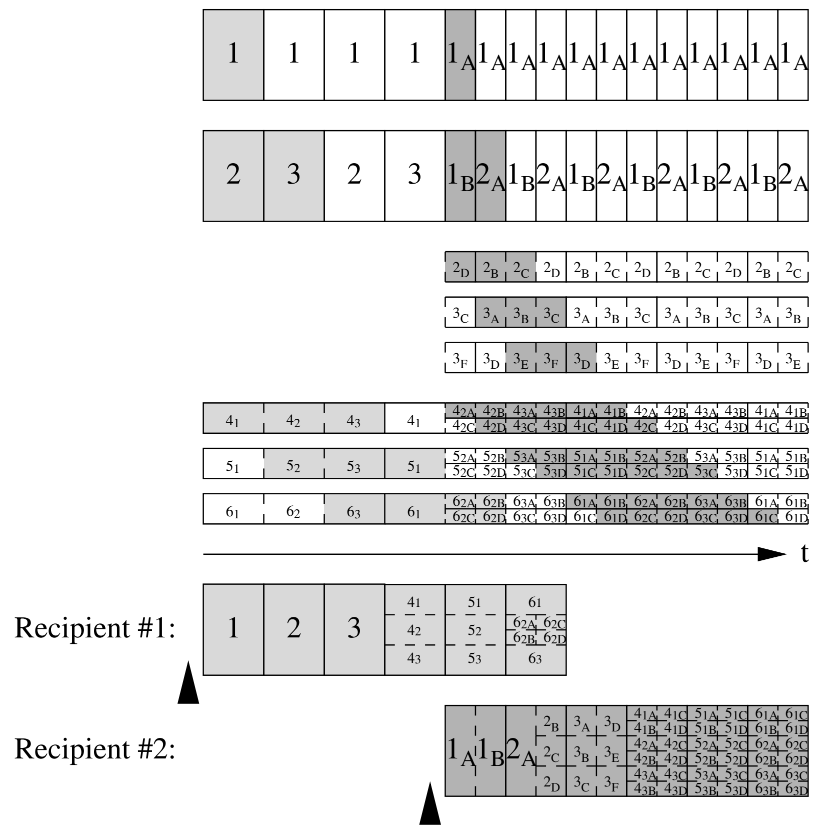


Figure 3.34: Seamless Staircase Broadcasting

[11] Unfortunately, the Seamless Staircase Broadcasting scheme cannot serve playback requests while this transition takes place, a fact which has not been mentioned by the authors of the transmission scheme.

3.6.8 Comparison of Bandwidth-Based Segmenting Transmission Schemes

Comparing all bandwidth-based segmenting transmission schemes with each other imposes similar difficulties as the comparison of the size-based segmenting transmissions schemes:

- The Quasi-Harmonic Broadcasting scheme, the Polyharmonic Broadcasting scheme and the Tailored Transmission scheme support a parameter[12], which, if set to a high value, increases the efficiency. For the Polyharmonic Broadcasting scheme and the Tailored Transmission scheme, the efficiency even tends towards 100 % if the parameter is increased. (But as this also increases the number of segments, calculations are stopped when the transmission scheme uses more segments than the lower bound bandwidth calculation assumed, otherwise the efficiency of these transmission schemes would advance beyond 100 %.)
- The Polyharmonic Broadcast scheme and the Tailored Transmission scheme also supports variable-bit-rate media streams and allows to restrict the receiver bandwidth, receiver storage size and throughput.
- The Seamless Staircase Broadcasting scheme supports seamless bandwidth changes for ongoing transmissions.
- For the Polyharmonic Broadcasting scheme and the Tailored Transmission scheme, the minimum playback delay equals the maximum playback delay, for the other transmission schemes of the bandwidth based group, the minimum playback delay is zero.

To compare the bandwidth-based transmission schemes, the following assumptions have been made:

- The available bandwidth of receiver systems is at least as high as the aggregate bandwidth of a whole transmission.
- The storage space available at the receiver systems and the throughput of it are high enough to cope with the transmission.
- A constant-bit-rate media stream is transmitted.
- Only the maximum playback delay is taken into account when the efficiency is calculated.

Under these preconditions, figure 3.35 shows the efficiency of the bandwidth-based transmission schemes. (The Harmonic Broadcasting scheme has been omitted as it is not working correctly, see section 3.6.1.) It is clearly visible, that the Polyharmonic Broadcasting scheme and the Tailored Transmission scheme (which produce identical results under the above preconditions) outperform the other bandwidth-based transmission schemes.

[12] In the original article of the Tailored Transmission scheme, the ratio of playback delay to segment size has not been named. As the Tailored Transmission scheme can be seen as an extension of the Polyharmonic Broadcasting scheme, the same parameter definition $m = \frac{W^+}{\delta}$ is used in this thesis.

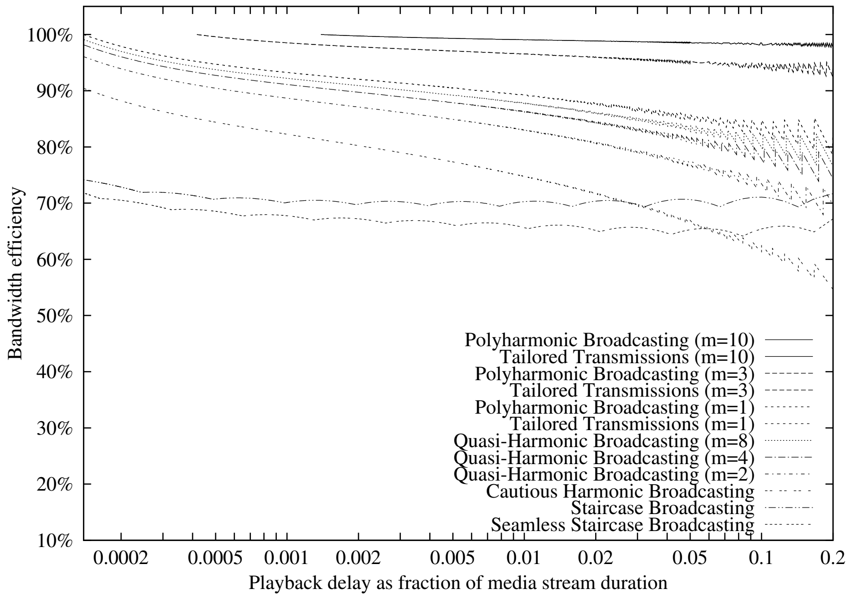


Figure 3.35: Efficiency of bandwidth-based schemes

3.7 Frequency-Based Segmenting Transmission Schemes

The third class of pro-active media-on-demand transmission schemes uses channels of equal bandwidth where segments of equal size are transmitted. Instead of using segments of increasing size or decreasing bandwidth, the frequency-based transmission schemes transmits the segments at decreasing frequency. As the average bandwidth which is used for the transmission of the segments decreases for the later segments, bandwidth is saved.

The main advantage of this procedure is that all segments are of equal size and that all channels use the same bandwidth. But to perform a successful frequency-based transmission, the segments have to be arranged on the channels in such a way that no two segments of one channel have to be transmitted at the same time. As the calculation of the optimal arrangement is NP-hard (this problem is a variant of the bin-packing problem [Wik06g], see [BNBNS98] for a prove), only non-perfect solutions are available for large transmission schedules, so the transmission schemes which are presented in this section mainly differ in the algorithm they use for arranging the segments.

3.7.1 Harmonic Equal-Bandwidth Broadcasting

Although never published separately, this broadcasting scheme is referenced several times when it comes to frequency-based transmission schemes. The reason therefore is that it is the most efficient frequency-based transmission scheme: It has an efficiency of 100 %.

The Harmonic Equal-Bandwidth Broadcasting scheme uses a single channel for each segment, each at the media stream bit rate, and the i-th segment is transmitted every $i + \left\lfloor \frac{W^+}{\delta} \right\rfloor - 1$ slot intervals on its channel. (Similar to the Polyharmonic Broadcasting scheme, it is possible to use a playback delay independent from the segment size to increase the efficiency.) Due to its similarities with the Polyharmonic Broadcasting scheme, the Harmonic Equal-Bandwidth Broadcasting scheme reaches the same efficiency as the Polyharmonic Broadcasting scheme, but without the need for a fixed wait policy.

Unfortunately, the bandwidth of this transmission scheme is extremely varying: At the beginning of the transmission ($t = 0$), only one channel transmits a segment, but at time offset $t = \delta \cdot \operatorname*{lcm}_{1 \leq i \leq n} i$ all n channels are in use[13] which makes the transmission scheme unusable in any practical environment. Figure 3.36 shows an example for five channels.

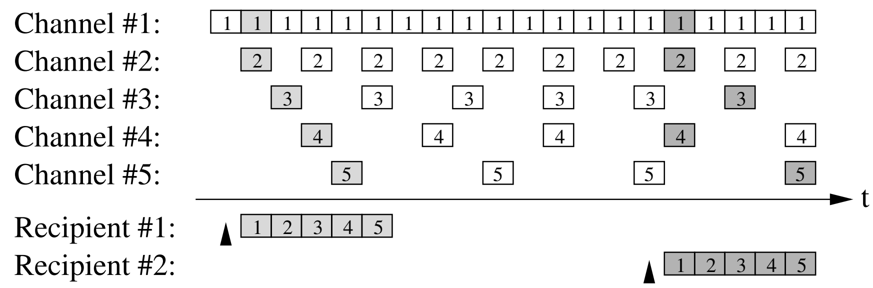


Figure 3.36: Harmonic Equal-Bandwidth Broadcasting

3.7.2 Fast Broadcasting

The Fast Broadcasting transmission scheme [JT98] was the first scheme which has been published for the frequency-based class and started a new, very efficient and useful type of transmission schemes. It uses a simple, binary scheme for the segment arrangement as illustrated in figure 3.37.

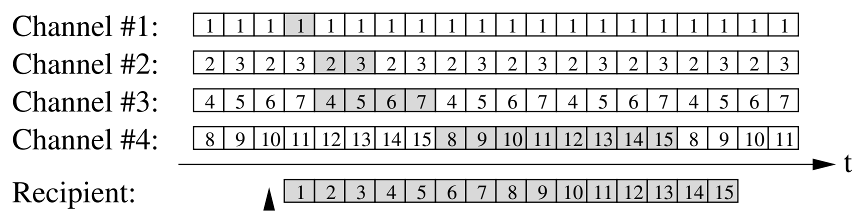


Figure 3.37: Fast Broadcasting

To write a frequency-based transmission scheme down, several different notations are used in the literature. The following are the most important ones:

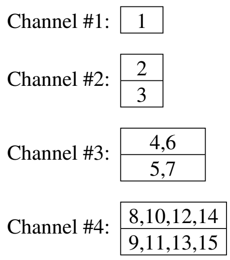
[13] The lcm operator denotes the least common multiple.

- As frequency-based transmission schemes are periodic, the most simple method for showing a frequency-based transmission scheme is to enumerate the segments of one period for each channel. Sadly, the period becomes very large for some transmission schemes, e. g. the period of one channel of the Greedy Broadcasting scheme using five channels is 5 880 slot intervals, and for the same transmission scheme using eight channels, it is about $91 \cdot 10^{15}$ slot intervals. For this reason, the enumeration is only sensible for a fraction of the frequency-based schemes.
- For the New Pagoda Broadcasting scheme [Pâr99], a rectangular representation has been invented. As the New Pagoda Broadcasting scheme splits the available bandwidth of each channel twice, it uses two dimensions for the representation. The first split of a channel into a number of subsets from which segments are transmitted round-robin is shown by dividing a rectangle into the appropriate number of rows, and for each row, the contained segments are enumerated for one period. As the segments of each row in the New Pagoda Broadcasting scheme are consecutive, this rectangular representation describes the transmission schemes of the New Pagoda Broadcasting very well, but it cannot be used efficiently for other, more complex transmission schemes, for the same reasons as above.
- A universal representation can be obtained by creating a table containing the period, the phase shift and the channel number of each segment. This table allows to decide when the sender has to transmit a segment in a very simple way: If t counts the slot intervals since the beginning of the transmission, segment #i has to be transmitted whenever $t \bmod \pi_i = \phi_i$ where π_i and ϕ_i denote the period and phase shift of segment #i, resp. Unfortunately, this table is very confusing and should better be used by machines than humans. Another disadvantage of the table representation is that errors in the transmission scheme are very hard to detect.
- For the Greedy Broadcasting Protocol, the authors developed a tree-based representation for transmission schedules, which not only allows a short, concise, and visual notation of transmission schemes but also enables the development of simple, graph-based algorithms for the creation, manipulation and examination of transmission schemes. (In section **??**, such an algorithm is presented.) This tree-based representation is based on the fact that the frequency-based transmission schemes use time-multiplexing for dividing the available bandwidth of a channel (or of the resulting sub-channel of a preceding time-multiplexing) into smaller parts until they are used to transmit segments. Therefore, the tree-based representation uses trees with the following characteristics:
  - Each channel is represented by one tree.
  - Each segment is represented by a leaf node in the tree which corresponds to its channel.
  - Each time-multiplexing is represented by a non-leaf node in the tree, dividing the bandwidth to its child nodes by transmitting them round-robin. Thus the number of child nodes of this node gives the multiplicity of the multiplexing process and the order of the child nodes specifies the transmission order of segments from the appropriate subtrees.

Figure 3.38 shows these four diagram types for the above Fast Broadcasting Transmission for comparison. Due to its compactness and clearness, tree-based diagrams are used in most cases to illustrate frequency-based transmission schemes from here on.

Channel #1: 1
Channel #2: 2, 3
Channel #3: 4, 5, 6, 7
Channel #4: 8, 9, 10, 11, 12, 13, 14, 15

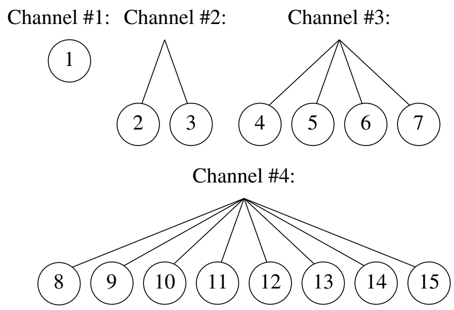
(a) Enumerating representation



(b) Rectangular representation

Segment #	Channel #	Period	Phase Shift
1	1	1	0
2	2	2	0
3	2	2	1
4	3	4	0
5	3	4	1
6	3	4	2
7	3	4	3
8	4	8	0
9	4	8	1
...	...	...	...

(c) Tabular representation



(d) Tree representation

Figure 3.38: Comparison of different representations for a frequency-based transmission scheme

3.7.3 Seamless Fast Broadcasting

The Seamless Fast Broadcasting scheme [THYL+01] is actually nearly identical to the Fast Broadcasting scheme (the segments have just been shifted to another startup position on the channels). But similar to the Seamless Staircase Broadcasting scheme (see section 3.6.7), the Seamless Fast Broadcasting scheme supports bandwidth changes for ongoing transmissions.

Figure 3.39 shows the process of a seamless channel addition for the Seamless Fast Broadcasting scheme.

3.7.4 Pagoda Broadcasting

As the Fast Broadcasting transmission scheme only performs with poor efficiency, the authors of the Pagoda Broadcasting scheme [YP01] increase it by using another transmission schedule: Instead of a simple binary scheme, Pagoda Broadcasting uses a scheme which has been manually optimized for the first channels while it follows mostly a simple scheme for the latter ones. Figure 3.40 shows an example for four channels and figure 3.41 shows the tree representation of the used transmission schedule.

In [PCL99b], the authors extended the Pagoda Broadcasting scheme in two topics: Firstly, the transmission scheme for an even number of channels is improved slightly

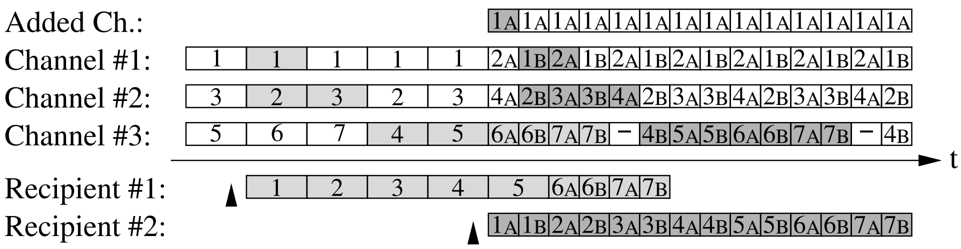


Figure 3.39: Example for a channel addition in an ongoing transmission using the Seamless Fast Broadcasting scheme

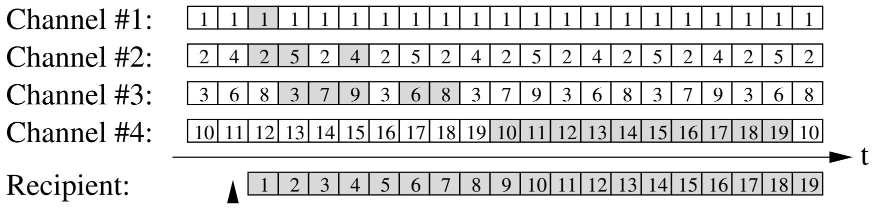


Figure 3.40: Pagoda Broadcasting

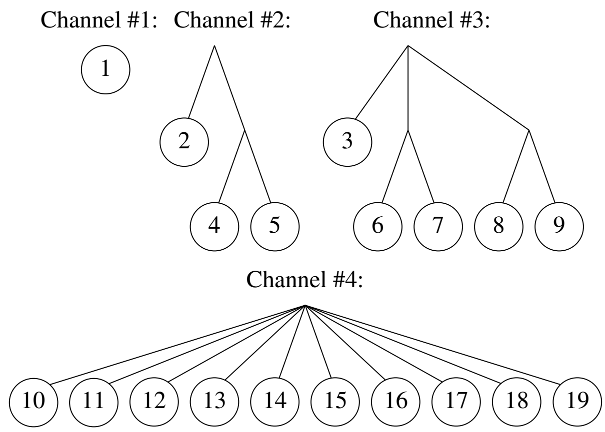


Figure 3.41: Tree representation of a Pagoda Broadcasting transmission scheme

and secondly, the authors suggest that the last channel may be shared across several media stream transmissions if it is only partially used — a topic which is reviewed in section **??**.

3.7.5 New Pagoda Broadcasting

Some time after the publication of the Pagoda Broadcasting scheme, the authors of the Pagoda Broadcasting scheme improved their transmission scheme with the New Pagoda Broadcasting scheme [Pâr99]. Thereby they used another, hand-optimized segment-to-channel mapping for the first channels and a better algorithm for the latter channels, which is based on the rectangular representation for transmission schemes which they have introduced (see section 3.7.2 for a description of this representation). Figure 3.42 shows an example for four channels, figure 3.43 shows the rectangular and tree representation of the used transmission schedule.

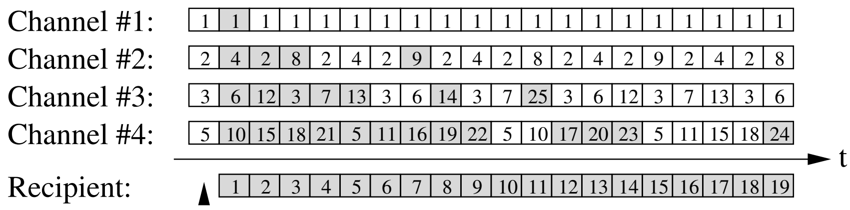


Figure 3.42: New Pagoda Broadcasting

3.7.6 Fixed Delay Pagoda Broadcasting

An extension to the New Pagoda Broadcasting scheme presented above is the Fixed Delay Pagoda Broadcasting scheme [Pâr01a]. In contrary to the Pagoda and New Pagoda Broadcasting scheme, it is possible to specify an arbitrary playback delay (in multiples of the slot interval) for this transmission scheme. If this playback delay is longer than the one slot interval playback delay of Pagoda/New Pagoda broadcasting, the created transmission schedule requires much less bandwidth. Figure 3.44 shows an example for two channels and a (maximum) playback delay of four slot intervals and figure 3.45 shows the rectangular and tree representation of the used transmission schedule.

Additionally, the Fixed Delay Pagoda Broadcasting scheme allows to define a limit for the receiver bandwidth to create a transmission schedule which does not exceed this limit.

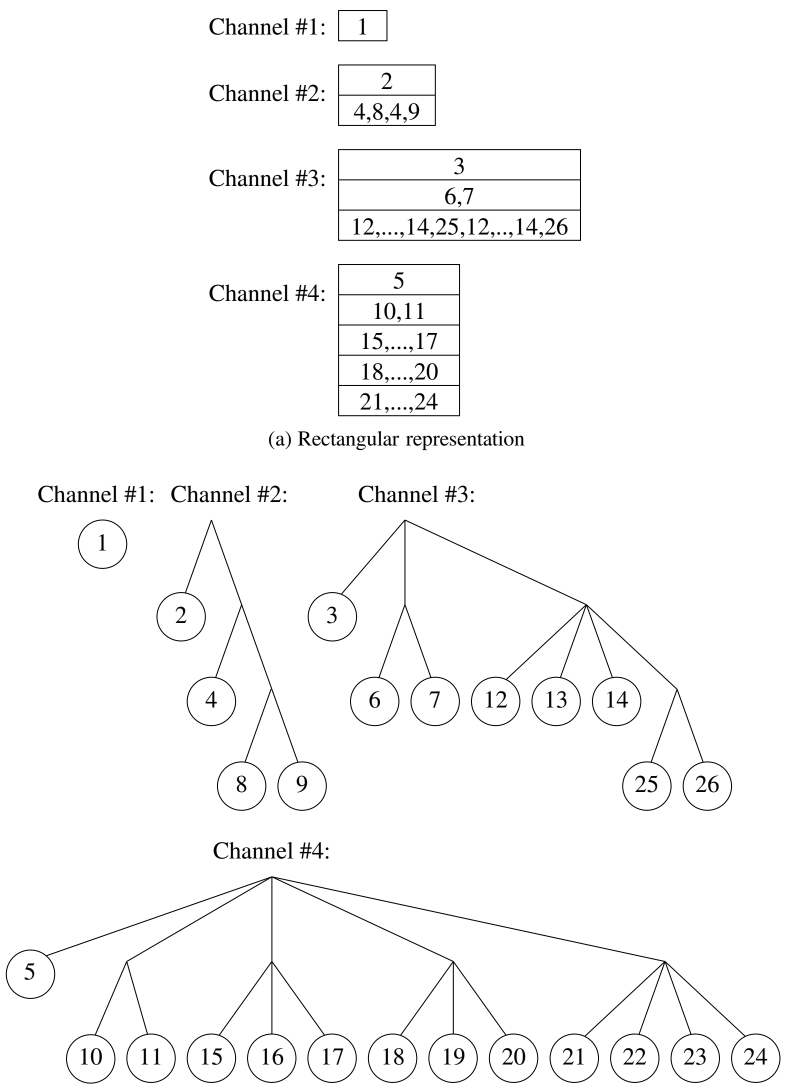
In sections **??** and **??**, the effects of the playback delay and decoupling the playback delay from the slot interval are examined in detail, in section **??** the consequences of a limited receiver bandwidth is studied again.

3.7.7 Variable Bandwidth Broadcasting

The Variable Bandwidth Broadcasting scheme [PL03] is an enhanced version of the New Pagoda Broadcasting scheme: It uses a similar layout and reaches an efficiency similar to the Fixed Delay Pagoda Broadcasting scheme with $W^+ = \delta$. Figure 3.46 shows an example of a Variable Bandwidth Broadcasting transmission.

The most interesting point about the Variable Bandwidth Broadcasting scheme is that it supports dynamic bandwidth changes, similar to the Seamless Staircase Broadcasting scheme and the Seamless Fast Broadcasting scheme.

The authors of the Variable Bandwidth Broadcasting scheme also analyzed one of the conditions which is necessary for supporting dynamic bandwidth changes: When a channel is added and the playback delay is decreased from one to a half slot interval, all segments which had been scheduled with their *maximum required period* are sent too infrequently in the new transmission scheme. Therefore these segments have to be moved to a new position (typically by moving all these segments together by one



(a) Rectangular representation

(b) Tree representation

Figure 3.43: Two different representations of a New Pagoda transmission scheme

position, i. e. by moving the first segment to the new channel, the second to the old position of the first segment and so on). The authors also proposed the idea to artificially decrease the period by one for the latter segments to prevent that they have to be moved.

Unfortunately, the authors missed to recognize a second condition: When a segment is moved forward in time, it is possible that this interrupts ongoing playbacks. Figure 3.47 shows an example where this segment moving causes a hole in a playback. This approach (together with a solution for the moving problem) is revised in section **??**.

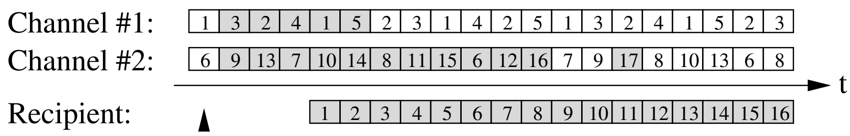


Figure 3.44: Fixed Delay Pagoda Broadcasting

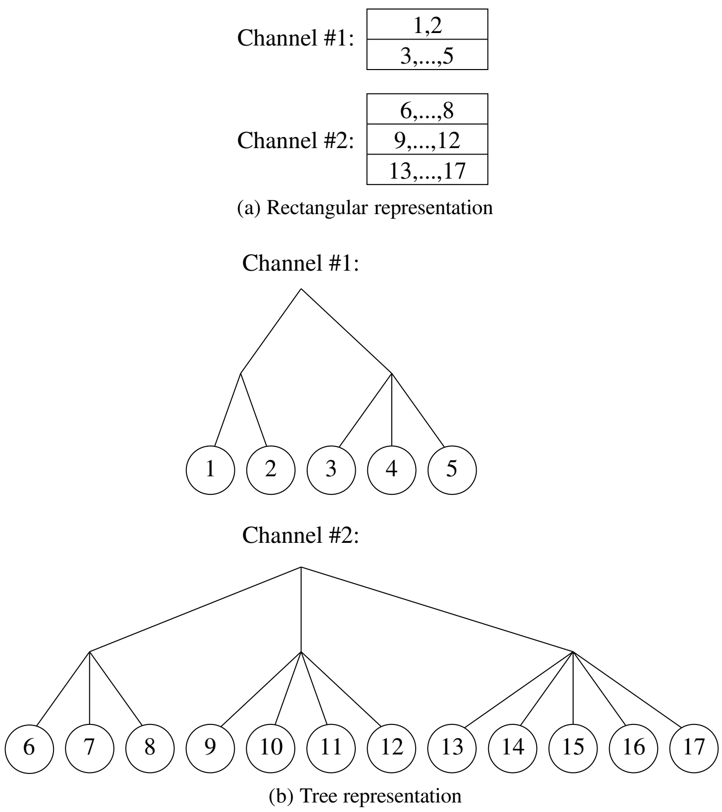


Figure 3.45: Two different representations of a Fixed Delay Pagoda transmission scheme

3.7.8 Greedy Broadcasting/Recursive Frequency Splitting

The next transmission scheme has been developed independently by two groups at nearly the same time as Greedy Broadcasting scheme [BNL02, BNL03, CST00] and Recursive Frequency Splitting scheme [TYC02].

While the Recursive Frequency Splitting scheme uses a more mathematical description of the algorithm, the authors of the Greedy Broadcasting scheme introduced the tree-based representation for transmission schemes and developed a tree-based algorithm. This algorithm inserts the segments one-by-one as nodes into the transmission trees at a position which is selected by a simple heuristic:

1. Primarily, the segment is inserted in such a way that the segment is transmitted

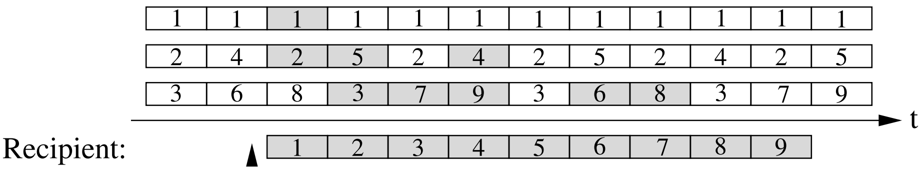


Figure 3.46: Variable Bandwidth Broadcasting

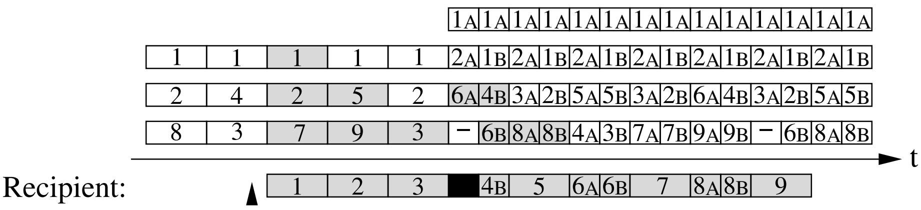


Figure 3.47: Ongoing transmissions may be interrupted when a channel is added using the Variable Bandwidth Broadcasting scheme

as often as necessary but as infrequently as possible, so the least bandwidth is wasted.

2. If the bandwidth loss for several nodes is equal, the node with the highest transmission period is selected, leaving the nodes with low periods for the remaining segments.

3. If this node has a transmission period low enough to carry more than the segment to be inserted, it is split into several parts of equal period, leaving the remaining nodes for other segments. In the tree representation, these splits are represented by non-leaf nodes, while the inserted segments are the leaf-nodes.

Figure 3.48 shows an example for four channels and figure 3.49 shows the tree representation of the used transmission schedule. This transmission scheme will be reviewed extensively in chapter **??** as its full potential has not been tapped yet.

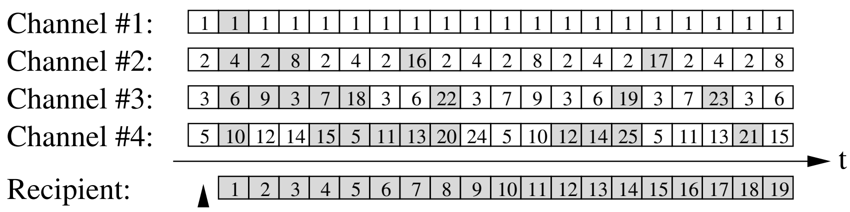


Figure 3.48: Greedy Broadcasting/Recursive Frequency Splitting

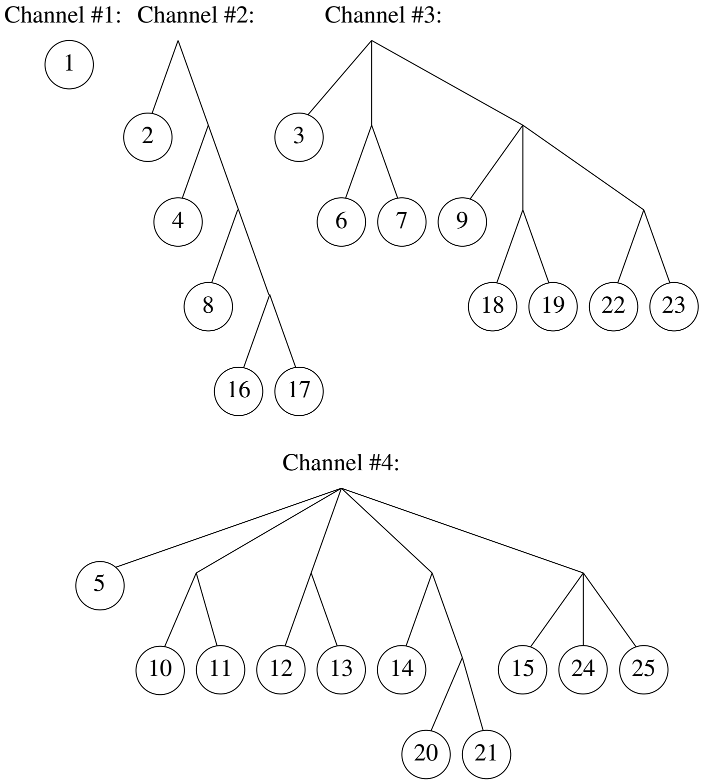


Figure 3.49: Tree representation of a Greedy Broadcasting/Recursive Frequency Splitting transmission scheme

3.7.9 Fuzzycast

Fuzzycast [JWX02] is based on the Harmonic Equal-Bandwidth Broadcasting scheme, but removes its high bandwidth requirements using a heuristic algorithm: When it encounters a slot interval which contains too many segments, it decides to move the highest segment(s) to another slot interval, either forward or backward in time. Unfortunately, moving segments has a drawback: If segments are transmitted earlier this way, the bandwidth is increased and the transmission scheme looses some efficiency. If segments are transmitted later, the playback delay has to be increased, which lowers the efficiency, too.

A major disadvantage of the Fuzzycast transmission scheme is that its bandwidth requirement or playback delay cannot be guaranteed for a permanent transmission: As the moving process is heuristic, there is no way to predict how much free slots are available in the neighborhood of a specific segment position (assuming that the granted bandwidth is not so high that nearly no moving is required). As a consequence, the Fuzzycast scheme has an unpredictable bandwidth consumption or playback delay because the bandwidth can only be examined if the playback is simulated until end.

One interesting improvement to reduce the number of segment moves has been mentioned by the authors of the Fuzzycast Broadcasting scheme: If several transmissions are grouped together, the probability to find a free slot at the destined position or in its near neighborhood is much higher than for a single transmission. Therefore the sender bandwidth can be reduced by this grouping for each of the transmissions. The not mentioned drawback of this grouping is that the receiver bandwidth must be higher when one transmission uses free slots of another one.

3.7.10 Dual Broadcasting

The Dual Broadcasting scheme [PCL99a] is a transmission scheme which supports receiver systems without and with local storage. To support both systems in a heterogene environment at the same time, a transmission scheme which requires no storage (e. g. Round-Robin transmissions or Staggered Broadcasting) and one which uses storage at the recipient systems (e. g. Fast Broadcasting or Pagoda Broadcasting) have to be used at the same time.

But a simple combination of these two transmission schemes is a waste of bandwidth as the recipient systems which are equipped with storage could benefit from the no-storage transmissions. The authors of the Dual Broadcasting scheme therefore developed a transmission scheme where the storage-based transmission relies on the no-storage transmission.

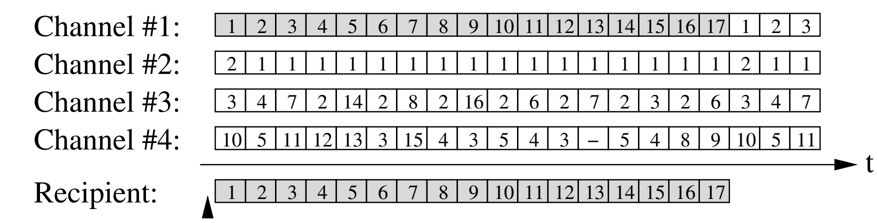


Figure 3.50: Dual Broadcasting

Unfortunately, the Dual Broadcasting scheme is not a pure frequency-based transmission scheme as it transmits the segments at non-constant frequency. As a consequence, transmission schemes according to this scheme cannot be created as simple as the other proposed schemes. Even the authors of the Dual Broadcasting scheme only presented two schedules for this scheme (one for three and one for four channels) without providing an algorithm for general construction.

Another thing to note here is that the Fast Broadcasting scheme (see section 3.7.2) already supports a reception without storage with a maximum playback delay of one media stream duration, but the Dual Broadcasting scheme outperforms this.

3.7.11 Comparison of Frequency-Based Segmenting Transmission Schemes

Comparing the frequency-based segmenting transmission schemes is a much simpler task than comparing size-based or bandwidth-based schemes as the differences between the various frequency-based schemes are much less significant. Besides of the layout of the segments on the channels, the transmission schemes differ in the following points:

- For the Fixed Delay Pagoda Broadcasting scheme and the Harmonic Equal-Bandwidth Broadcasting scheme, the playback delay can be defined independent of the segment size.

- The Fixed Delay Pagoda Broadcasting scheme supports receiver systems with limited receiver bandwidth or storage size.
- The Fast Broadcasting scheme and the Dual Broadcasting scheme support receiver systems without storage.
- The Seamless Fast Broadcasting scheme and the Variable Bandwidth Broadcasting scheme support seamless bandwidth changes for ongoing transmissions.

Similar to the comparison charts for size- and bandwidth-based transmission schemes, it is assumed that the receiver systems provide an unlimited amount of bandwidth and storage to receive the transmission. Additionally, the ability to change the playback delay dynamically is ignored here.

Figure 3.51 shows the efficiency of the frequency-based transmission schemes if the slot duration equals the playback delay. As expected, the (practically unusable) Harmonic Equal-Bandwidth Broadcasting scheme provides the highest efficiency of all these transmission schemes, nearby followed by the Greedy Broadcasting/Recursive Frequency Splitting scheme. But the Harmonic Equal-Bandwidth Broadcasting scheme is still far away from the theoretical lower bound (i. e. from an efficiency $\eta = 1$) if the segment size is bound to the playback delay, esp. for long playback delays.

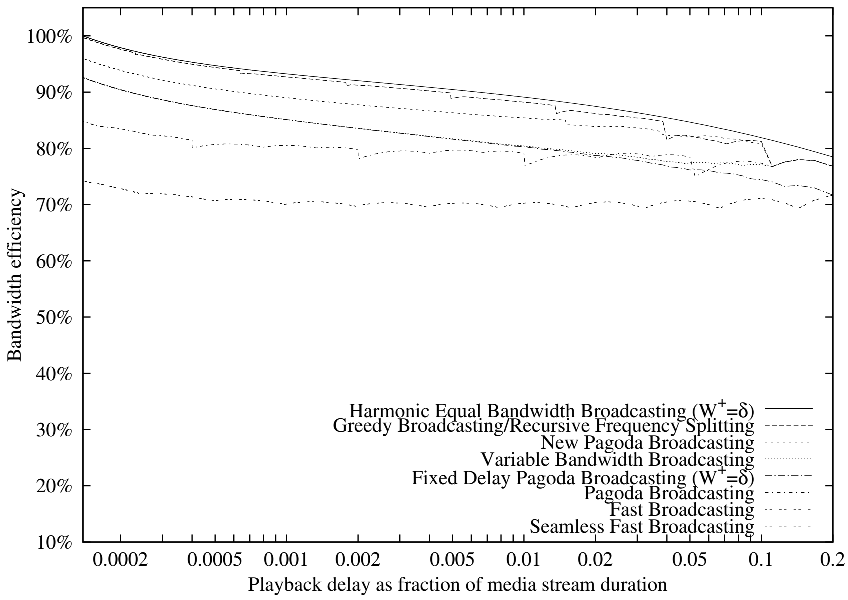


Figure 3.51: Efficiency of frequency-based schemes if the playback delay equals the slot interval

If the transmission scheme supports to select the playback delay independent from the segment size, the segment size can be decreased while keeping the playback delay constant. Figure 3.52 shows the result of this procedure for the two transmission schemes which support this functionality. In this case, both transmission schemes operate much better, the Harmonic Equal-Bandwidth Broadcasting scheme even operates at 100 %.

The Dual Broadcasting scheme has been omitted from these comparisons as no algorithm for generation of large Dual Broadcasting transmission schedules is available.

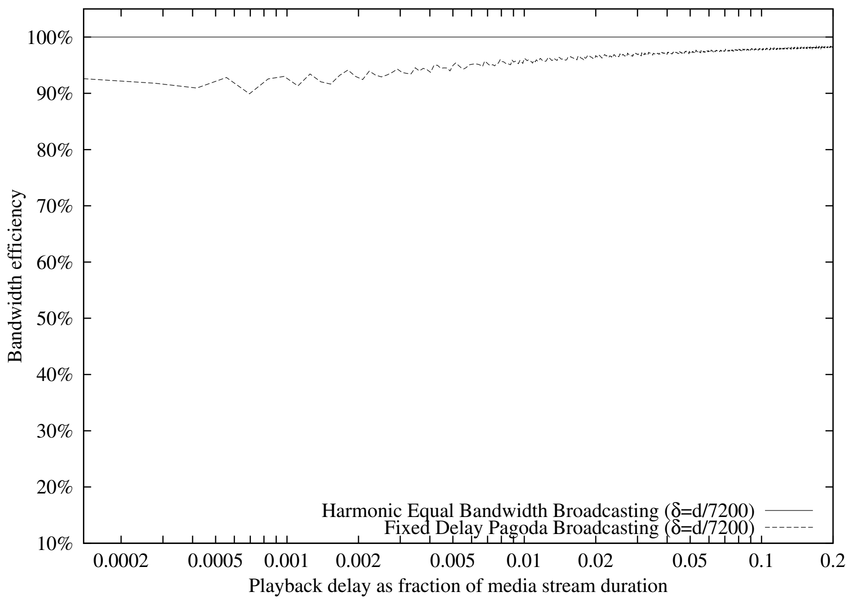


Figure 3.52: Efficiency of frequency-based schemes for which the slot interval can be decreased independent of the playback delay

3.8 Reactive-Pro-Active-Hybrid Transmission Schemes

The above presented transmission schemes are either reactive or pro-active by design. In this section, **reactive-pro-active-hybrid transmission schemes** are presented which use combinations of reactive and pro-active design patterns.

One way to combine reactive and pro-active transmission schemes is to send part of the media stream using a reactive transmission scheme and the remainder using a pro-active scheme. The reason for combining a reactive and a pro-active transmission scheme this way is that the most bandwidth of a media transmission is required for the transmission of the first segments which are received only by a few recipients, thus the most bandwidth is wasted by the transmission of the first part of the media stream if only few recipients are listening to the transmission. If a reactive transmission scheme is used for the first part of the media stream, this bandwidth can be saved, providing at the same time a short playback delay (or even immediate access, depending only on the reactive scheme).

Another way to apply reactive behavior to a pro-active transmission scheme is to use playback request to evaluate which segment transmissions are not needed by any recipient and omitting the transmission of these segments in a pro-active transmission schedule.

3.8.1 Unified Video-on-Demand Broadcasting

The Unified Video-on-Demand Broadcasting scheme [Lee99] is a combination of the Staggered Broadcasting scheme (see section 3.4.2 with Point-to-Point transmissions (see section 3.3.1: As the playback delay of the Staggered Broadcasting scheme is very high, the Unified Video-on-Demand Broadcasting scheme uses Point-to-Point transmissions to transmit the beginning of the media stream to recipients which called in at an ongoing transmission. That way, the Unified Video-on-Demand Broadcasting scheme provides immediate access without the need to transmit the whole media stream to the recipients.

3.8.2 Batching Unified Video-on-Demand Broadcasting

Similar to the Unified Video-on-Demand Broadcasting scheme of the last subsection, the Batching Unified Video-on-Demand Broadcasting scheme [LL00] benefits from the Staggered Broadcasting scheme. But instead of using Point-to-Point transmissions for the beginning of the media streams, this scheme broadcasts the beginning at times determined by a Batching scheme (see section 3.3.2). Consequentially, the Batching Unified Video-on-Demand Broadcasting scheme does not offer immediate access any more but still reduces the playback delay of the Staggered Broadcasting scheme.

3.8.3 Reactive Broadcasting

Similar to the above two transmission schemes, the Reactive Broadcasting scheme [PCL00a] is actually a combination of two transmission schemes: A reactive transmission is used for the transmission of the first segment of the media stream and a pro-active transmission scheme is used for the remaining segments. But instead of using Staggered Broadcasting combined with Point-to-Point Transmissions or Batching, the authors of [PCL00a] suggested to use Tapping (see section 3.3.5) as reactive scheme and a Pagoda-like scheme (see section 3.7.4) as pro-active scheme.

3.8.4 Dynamic Skyscraper Broadcasting

The Dynamic Skyscraper Broadcasting scheme [EV98] makes two modifications to the Skyscraper Broadcasting scheme presented in section 3.5.3. Firstly, it divides subsequent segment transmissions up to a specific length to **transmission clusters** and allows that transmission clusters can be dropped if no recipient is listening to the appropriate transmission. As they suggest to use the same transmission cluster sizes for all media streams, this spare bandwidth can simply be reused for other media stream transmissions. Figure 3.53 illustrates the use of transmission clusters.

Secondly, they proposed three new broadcasting sequences:

$$\begin{aligned} A:&\quad 1,2,2,4,4,8,8,16,16,32,32,64,64,128,128,\ldots \\ B:&\quad 1,2,2,6,6,12,12,24,24,48,48,96,96,192,192,\ldots \\ C:&\quad 1,2,2,6,6,12,12,36,36,72,72,216,216,432,432,\ldots \end{aligned} \tag{3.37}$$

These broadcasting sequences avoid conflicts and holes in transmission clusters which would occur if the original broadcasting sequences are used. Broadcasting sequences B and C additionally require that the receiver system is able to receive three channels at a time, while for sequence A two channels still suffice.

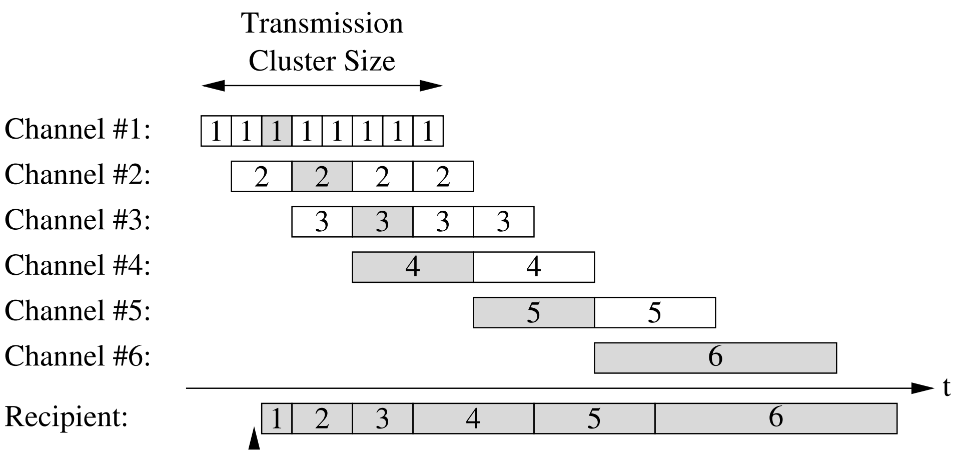


Figure 3.53: Dynamic Skyscraper Broadcasting

3.8.5 Partitioned Dynamic Skyscraper Broadcasting

The Partitioned Dynamic Skyscraper Broadcasting scheme [EFV99] is a variant of the Dynamic Skyscraper Broadcasting scheme presented above. Instead of using a single sender, the Partitioned Dynamic Skyscraper Broadcasting scheme assumes a hierarchical setup, consisting out of a global sender system and a set of regional sender systems. Therefore the Partitioned Dynamic Skyscraper Broadcasting scheme requires that the receiver systems is able to receive data simultaneously from the global sender system and from its next regional sender system (doubling the bandwidth requirement for the clients and assuming a network which benefits from regional sender systems).

The Partitioned Dynamic Skyscraper Broadcasting scheme uses this setup by transmitting the first segments of a media stream using regional sender systems and sending the latter segments from the global sender system. This reduces the global network utilization as the most bandwidth is required for the first segments which are only transmitted regionally. Figure 3.54 shows an example for a Partitioned Dynamic Skyscraper Broadcasting transmission.

3.8.6 Universal Broadcasting

The Universal Broadcasting scheme [PCL00b] is based on the Fast Broadcasting scheme (see section 3.7.2), but it uses two modifications to save bandwidth if only few recipients are requesting the media:

1. Only required segments are sent at the latest possible time. To ascertain which segments are required, this transmission scheme is based on requests from recipient systems to determine at which slot intervals recipients have started a playback.
2. The segments of each channel may be shifted if no ongoing playback is interfered thereby.

In section **??**, the idea behind this approach is reviewed in a generalized version.

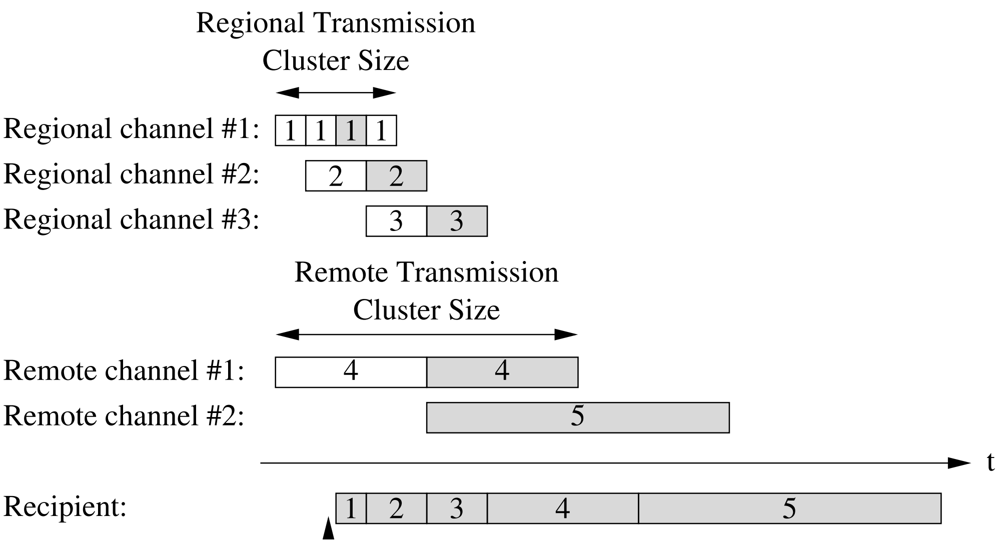


Figure 3.54: Partitioned Dynamic Skyscraper Broadcasting

Using these two modifications, the Universal Broadcasting scheme needs no bandwidth if no playbacks are active and it only needs one transmission channel if only one request is active. As the authors have examined, the Universal Broadcasting scheme needs less bandwidth than the (pro-active) New Pagoda transmission scheme for up to 55 requests per hour if a two hour media stream is transmitted in 127 segments. For higher request rates, the Universal Broadcasting scheme converges to the Fast Broadcasting scheme.

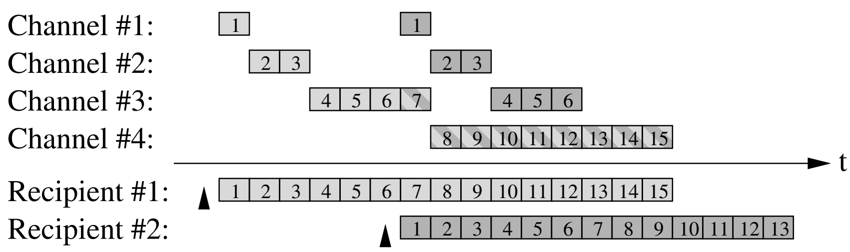


Figure 3.55: Universal Broadcasting

Another thing to note here is that it is possible to apply the first of the above mentioned modifications — suppression of segment transmissions which are not needed by any receiver system — to any segmenting transmission scheme. The second modification — shifting of segments in channels — can only by applied to transmission schemes which transmit all segments of a channel with the same frequency, thus it can only be applied to the Fast Broadcasting scheme of the frequency-based group of transmission schemes. (By theory, it would also be possible to apply it to transmission schemes of e. g. the size-based group as they only transmit a single segment per channel, but as their efficiency is below that of Fast Broadcasting this is not studied here in more detail.)

3.8.7 Channel-Based Heuristic Distribution

The Channel-Based Heuristic Distribution scheme [ZP02] is the result of a further development of the Universal Broadcasting scheme. It produces similar results as the Universal Broadcasting scheme, but it has been extended to support increased playback delays. Figure 3.56 shows an example for this transmission scheme with a playback delay of two slot intervals.

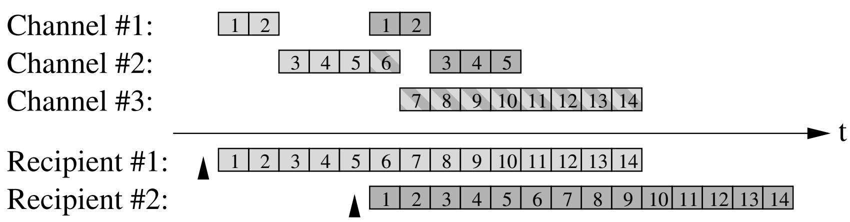


Figure 3.56: Channel-Based Heuristic Distribution

3.8.8 Comparison of Reactive-Pro-Active-Hybrid Transmission Schemes

This section provided mainly two different approaches how pro-active transmission schemes can be used in a reactive environment. Firstly, two transmission schemes can be combined: A reactive transmission scheme can be used for the beginning of the media stream and a pro-active transmission scheme for the remainder. This idea has been used by the Unified Video-on-Demand Broadcasting scheme, the Batching Unified Video-on-Demand Broadcasting scheme, the Reactive Broadcasting scheme and the Dual Broadcasting scheme.

Alternatively, it is possible to use solely a pro-active transmission scheme, omitting transmissions which are not needed by any recipient. This approach is followed in the Dynamic Skyscraper Broadcasting scheme, the Partitioned Dynamic Skyscraper Broadcasting scheme, the Universal Broadcasting scheme and the Channel-based Heuristic Broadcasting scheme.

Additional ideas which have been proposed in these transmission schemes include combining segments to transmission clusters (to manage unused bandwidth more simple), dividing the transmission into a global and several regional parts (to reduce global bandwidth consumption by frequent transmissions of media stream beginnings) and shifting of segments in channels (to lower bandwidth consumption).

For a comparison of hybrid transmission schemes, the scheme which provides immediate service (the Reactive Broadcasting scheme) and the schemes which require a playback delay after requests (the Unified Video-on-Demand Broadcasting, the Batching Unified Video-on-Demand Broadcasting, the Dynamic Skyscraper Broadcasting, the Partitioned Dynamic Skyscraper Broadcasting, the Universal Broadcasting and the Channel-based Heuristic Distribution scheme) have to be examined separately, similar to the comparison of reactive transmission schemes in section 3.3.9.

The Reactive Broadcasting scheme, the only proposed hybrid transmission scheme with immediate service, shows a poor efficiency for very low request rates as shown in figure 3.57, a result of the permanent transmission of the latter part of the media stream. For higher request rates, the efficiency stays nearly constant around the one of the used pro-active Pagoda Broadcasting scheme.

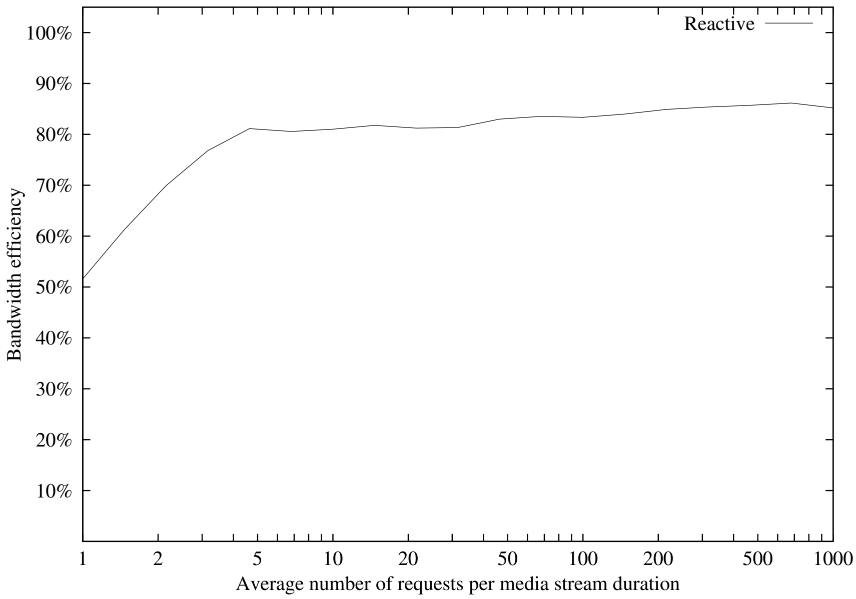


Figure 3.57: Efficiency of hybrid transmission schemes with immediate service

Of the transmission schemes which require a playback delay, the Channel-Based Heuristic Distribution scheme provides the best efficiency for all request rates. Figure 3.58 shows the efficiency of these transmission schemes with a playback delay of $\frac{1}{120}$ of the media stream duration.

3.9 Summary of Media Transmission Schemes

In this chapter, many different approaches for transmission schemes have been presented. Generally, they can be divided into two groups: The reactive and hybrid transmission schemes which rely on playback requests from the receiver systems and the pro-active transmission schemes which transmit the media streams independent of any requests.

The first group can be further divided into schemes which deliver a media stream immediate on request and schemes which use a playback delay to reduce bandwidth requirements. Of the schemes which provide immediate service, the ERMT Merging scheme provides the highest efficiency for few to medium requested streams but also requires a lot of calculation for each playback request. The CT Merging and the ϕ-Dyadic Merging schemes are not yet as efficient as the ERMT Merging scheme but are much more simple to implement. Only for high request rates (more than approximately 150 requests per media stream duration), the Reactive Broadcasting scheme

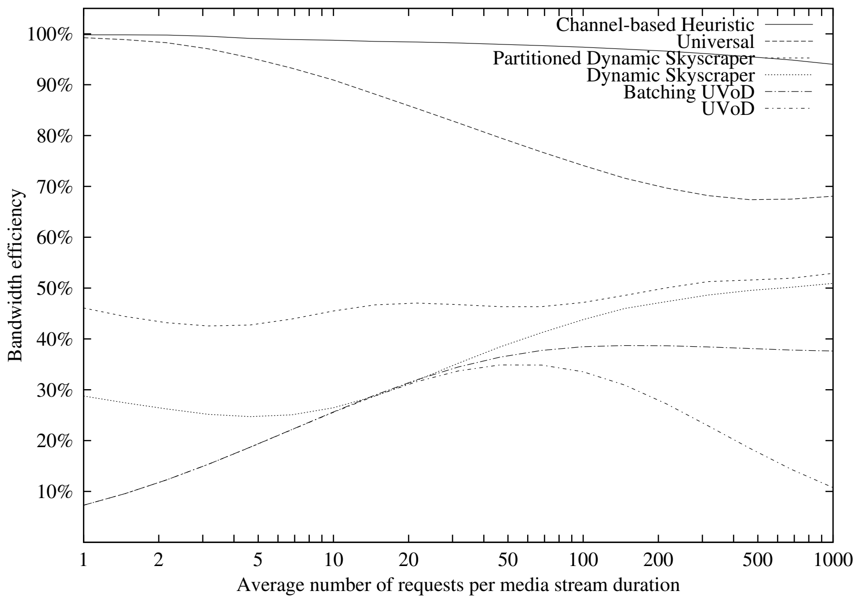


Figure 3.58: Comparison of efficiency of hybrid transmission schemes with delayed service

performs better than the ERMT Merging scheme. The reason therefore is that the Reactive Broadcasting scheme requires a permanent (pro-active) transmission of the latter part of the media stream which seems to be profitable only if the media stream is requested often enough.

Of the reactive and hybrid group of transmission schemes which require a playback delay, the Channel-Based Heuristic Broadcasting scheme performs best.

The segmenting pro-active transmission schemes can be subdivided into three groups based on the mechanism which is used to reduce the sender bandwidth consumption for a transmission: Size-based transmission schemes use increasing segment sizes at equal bandwidth, bandwidth-based schemes transmit segments of equal size in channels of decreasing bandwidth and frequency-based schemes send segments of equal size on channels of equal bandwidth at decreasing frequency. Although all this three groups reach the same goal (transmitting the latter parts of a media stream less frequently than the former parts), they perform very different:

The size-based group is probably the most simple but at the same time the least efficient one: The reason for this efficiency loss is that the later segments of a media stream become very large which prevents a fine-grained scheduling.

The bandwidth-based schemes provide nearly ideal theoretical conditions: The efficiency can be increased to reach the theoretical limit and many enhancements (e. g. adaption to variable-bit-rate media streams, support for receiver system constraints and seamless bandwidth changes) have been proposed for these schemes. Unfortunately, these transmission schemes perform very poor in practical environments for several reasons:

- If many segments are used (which is necessary to achieve a high efficiency), the bandwidth for the later segment transmissions becomes very low. Effectively, the media bit rate of the last segments is so low that it is practically impossible to transmit data at these bit rates[14]. Therefore many channels have to be aggregated into channels of higher bandwidth which imposes several additional problems (e. g. handling of fractions in bit rates, identification of aggregation boundaries for splitting into subsegments at the receiver side and increased damage in case of data losses if one aggregated packet is lost). Additionally, every segment is transmitted at a different bit-rate which makes any calculations more complex.

- The bandwidth-based schemes are based on the fact that part of each segment is transmitted at every slot interval. This means that even a short transmission failure causes holes in all segments if it cannot be repaired. For example for video streams, a loss of all data of one slot interval of a transmission would cause decoding problems in every segment of the whole media stream.
- Another consequence of the idea that part of each segment is transmitted at every slot interval is that both the sender and the receiver systems need to access all segments at each slot interval. This increases the hardware requirements for receiver and sender systems dramatically: To save the data of one slot interval to the corresponding segments on a hard disk, the disk must support high-speed random access which makes these systems very expensive[15]. Advanced caching strategies may be used to reduce the number of disk accesses by increasing the needed amount of memory.

The frequency-based schemes do not have any of these problems: All segments have the same size and are transmitted at the same bandwidth, only few segments have to be accessed each slot interval and data losses only affect a few segments. When comparing the frequency-based schemes, the Fixed-Delay Pagoda Broadcasting scheme provides the highest efficiency, nearby followed by the Greedy Broadcasting/Recursive Frequency Splitting scheme. Unfortunately, these schemes do not provide the same efficiency as the bandwidth-based schemes.

Even some of the less efficient transmission schemes qualify for a more precise examination because some of them provide worthwhile enhancements, e. g. dynamic bandwidth changes for ongoing transmissions or support for variable-bit-rate transmissions.

Unfortunately, there is no efficient transmission scheme which provides a high efficiency and which supports all these enhancements at once. For this reason, a new transmission scheme is proposed in the next chapter which benefits from several perceptions of this chapter and which incorporates many approaches of the examined transmission schemes.

[14] If a two-hour media stream of 2 Mbit/s is transmitted with a playback delay of 60 seconds and a segment size equal to one second, the last segment would have to be sent at approximately 275.5 bit/s.

[15] For example, during the first slot interval of a two hour media stream with one second playback delay, parts of 7 200 segments are transmitted. This would require 7 200 disk accesses per second and yields seek times of 138 µs (typical seek times of todays disks are around 9 ms). Saving the received data linearly onto disk and assembling it at playback time does not help either, the disk seeks are required at assembly time this way.

Appendix B

Nomenclature

IN this thesis, symbol names in equations have been used uniformly wherever possible. The following table provides a reference of the used symbol, its meaning and unit. For scalar values, the unit is displayed as 1, for functions, domain and codomain are separated by an arrow (→).

Symbol	Description	Unit
b	bandwidth of media stream (only applicable for constant-bit-rate media streams)	$\frac{bytes}{s}$
d	total duration of media stream	s
$f(t)$	cumulative bandwidth function, $f(t)$ is the position of the first byte in the media stream which is played at time t after beginning of playback	$s \rightarrow bytes$
$f^{-1}(p)$	inverse cumulative bandwidth function, $f^{-1}(p)$ is the time stamp in the media stream when the position p is reached	$bytes \rightarrow s$
n	number of segments of the media stream	1
s	size of media stream	$bytes$
B	total sender bandwidth of the transmission	$\frac{bytes}{s}$
B^{-}	theoretical lower bound for B	$\frac{bytes}{s}$
B°	average total sender bandwidth of the transmission	$\frac{bytes}{s}$
B_i	bandwidth of channel #i	$\frac{bytes}{s}$
C	fraction for lowering channel bandwidth	1
$F(t)$	playback function of the receiver system, based on the utilized playback delay; $F(t)$ is the position in the media stream which is played at time t	$s \rightarrow bytes$
$F^{-}(t)$	playback function of the receiver system, based on the minimum playback delay	$s \rightarrow bytes$
$F^{+}(t)$	playback function of the receiver system, based on the maximum playback delay	$s \rightarrow bytes$
$F^{-1}(p)$	inverse playback function, based on utilized playback delay	$bytes \rightarrow s$
$F^{--1}(p)$	inverse playback function, based on minimum playback delay	$bytes \rightarrow s$
$F^{+-1}(p)$	inverse playback function, based on maximum playback delay	$bytes \rightarrow s$

Symbol	Description	Unit
J	protocol specific latency for leaving and joining a channel	s
J_i	joining latency of channel #i relative to beginning of reception	s
L	Prime split limit; if $\left\lfloor \frac{\pi_i^+}{\Pi_N} \right\rfloor \geq L \cdot \Pi_N + 1$, a prime split of a node is prohibited	1
N	number of channels of the transmission	1
P	size of preloaded amount of media stream	$bytes$
$M^+(t)$	worst case storage requirement function, $M^+(t)$ gives the worst case storage requirement of a playback at time t	$s \rightarrow bytes$
P_i	total period of channel #i	1
P_i^+	imposed limit on P_i	1
$Q_{i,N}$	quality value for scheduling segment #i at node N, used to find the best node in the scheduling algorithm	1
$Q_{i,N}^{I}$	first of the quality values which are combined to $Q_{i,N}$	1
$Q_{i,N}^{II}$	second of the quality values which are combined to $Q_{i,N}$	1
$Q_{i,N}^{III}$	third of the quality values which are combined to $Q_{i,N}$	1
R	number of channels the receiver system can receive simultaneously	1
S	broadcasting series of size-based transmission schemes	$1 \rightarrow 1$
$T_{\text{playbackdelay}}(t)$	transmission function for additional playback delay	$s \rightarrow s$
$T_{\text{partialpreloading}}(p)$	transmission function for partial preloading	$bytes \rightarrow bytes$
$T_{\text{breakinsertion}}(t)$	transmission function for break insertion	$s \rightarrow s$
W^-	minimum possible playback delay for the transmission	s
W^+	maximum possible playback delay for the transmission	s
W°	average playback delay for the transmission	s
W^{II}	weight for weighting of $Q_{i,N}^{II}$ in $Q_{i,N}$	1
W^{III}	weight for weighting of $Q_{i,N}^{III}$ in $Q_{i,N}$	1
β	bandwidth of each segment transmission/bandwidth of each channel, $\beta = \frac{\sigma}{\delta}$	$\frac{bytes}{s}$
β_i	bandwidth of segment #i in the schedule	$\frac{bytes}{s}$
δ	slot interval = transmission duration of one segment	s
δ_i	duration of a segment #i	s
κ_i	channel number which is used for segment #i in the schedule; symbol taken from the Greek word καvάλι	1
π_i	utilized period of segment #i in the schedule; symbol taken from the Greek word περίοδος	1
π_i^+	maximum period of segment #i which is needed in the schedule for proper playback	1
σ	size of segments	$bytes$
σ_i	size of segment #i	$bytes$
τ_i	playback time of segment #i	s
ϕ_i	phase shift of segment #i, $0 \leq \phi_i < \pi_i$; symbol taken from the Greek word φάση	1
K_N	channel number which is used for node #i in the schedule	1

Symbol	Description	Unit
Π_N	utilized period of node $\#N$ in the schedule	1
Φ_N	phase shift of node $\#N$, $0 \leq \Phi_i < \Pi_i$	1
η	efficiency of a transmission schedule, $0 < \eta \leq 1$	1

The following mathematical symbols, functions and operators have been used throughout this thesis:

Operator	Description
∞	infinity
$\lfloor\rfloor$	truncation to next integer (towards $-\infty$)
$\lceil\rceil$	rounding up to next integer (towards $+\infty$)
$\wedge$	logical and
$\vee$	logical or
div	integer division, $a \text{ div } b = \left\lfloor \frac{a}{b} \right\rfloor$
gcd	greatest common divisor
lcm	least common multiple
mod	integer modulo, $a \bmod b = a - b \cdot \left\lfloor \frac{a}{b} \right\rfloor$

Bibliography

[AWY96a] C. C. Aggarwal, J. L. Wolf, and P. S. Yu, "On optimal piggyback merging policies for video-on-demand systems," in *Proc. ACM Int. Conf. on Measurement and Modeling of Computer Systems (SIGMETRICS)*, Philadelphia, PA, USA, May 1996, pp. 200–209.

[AWY96b] C. C. Aggarwal, J. L. Wolf, and P. S. Yu, "A permutation-based pyramid broadcasting scheme for video-on-demand systems," in *Proc. IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS)*, Tokyo, Japan, June 1996, pp. 118–126.

[AWY01] C. C. Aggarwal, J. L. Wolf, and P. S. Yu, "The maximum factor queue length batching scheme for video-on-demand systems," *IEEE Trans. on Computers*, vol. 50, no. 2, pp. 97–110, 2001.

[BCMM00] G. Boggia, P. Camarda, P. L. Mazzeo, and M. Mongiello, "Multicast distribution for multimedia on demand services," in *Proc. Int. Conf. on Communication Technology (ICCT)*, Beijing, China, Aug. 2000, pp. 1236–1243.

[BM99] Y. Birk and R. Mondri, "Tailored transmissions for efficient near-video-on-demand service," in *Proc. IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS)*, Florence, Italy, June 1999, pp. 226–231.

[BNBNS98] A. Bar-Noy, R. Bhatia, J. Naor, and B. Schieber, "Minimizing service and operation costs of periodic scheduling," in *Proc. ACM/SIAM Symposium on Discrete Algorithms (SODA)*, San Francisco, CA, USA, Jan. 1998, pp. 11–20.

[BNGL03] A. Bar-Noy, J. Goshi, and R. E. Ladner, "Off-line and on-line guaranteed start-up delay for media-on-demand with stream merging," in *Proc. ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, San Diego, CA, USA, June 2003, pp. 164–173.

[BNGLT02] A. Bar-Noy, J. Goshi, R. E. Ladner, and K. Tam, "Comparison of stream merging algorithms for media-on-demand," in *Proc. Conf. on Multimedia Computing and Networking (MMCN)*, San Jose, CA, USA, Jan. 2002, pp. 115–129.

[BNL02] A. Bar-Noy and R. E. Ladner, "Windows scheduling problems for broadcast systems," in *Proc. ACM/SIAM Symposium on Discrete Algorithms (SODA)*, San Francisco, CA, USA, Jan. 2002, pp. 433–442.

[BNL03] A. Bar-Noy and R. E. Ladner, “Windows scheduling problems for broadcast systems,” *SIAM Journal on Computing*, vol. 32, no. 4, pp. 1091–1113, 2003.

[BNL04] A. Bar-Noy and R. E. Ladner, “Efficient algorithms for optimal stream merging for media-on-demand,” *SIAM Journal on Computing*, vol. 33, no. 5, pp. 1011–1034, 2004.

[BS91] I. N. Bronštejn and K. A. Semendjaev, “Taschenbuch der Mathematik,” 25th ed. Stuttgart/Leipzig, Germany: B. G. Verlagsgesellschaft, 1991, p. 115.

[CHV99] Y. Cai, K. A. Hua, and K. Vu, “Optimizing patching performance,” in *Proc. Conf. on Multimedia Computing and Networking (MMCN)*, San Jose, CA, USA, Jan. 1999, pp. 204–215.

[CJM02] E. G. Coffman, jr, P. Jelenković, and P. Momčilović, “The dyadic stream merging algorithm,” *ACM Journal on Algorithms*, vol. 43, no. 1, pp. 120–137, 2002.

[CL97] S. W. Carter and D. D. E. Long, “Improving video-on-demand server efficiency through stream tapping,” in *Proc. IEEE Int. Conf. on Computer Communications and Networks (ICCCN)*, Las Vegas, NV, USA, Sept. 1997, pp. 200–207.

[CL99] S. W. Carter and D. D. E. Long, “Improving bandwidth efficiency on video-on-demand servers,” *Int. Journal of Computer and Telecommunications Networking*, vol. 31, no. 1–2, pp. 111–123, 1999.

[CLP00] S. W. Carter, D. D. E. Long, and J.-F. Pâris, “An efficient implementation of interactive video-on-demand,” in *Proc. Int. Symposium on Modeling, Analysis, and Simulation of Computer Systems (MASCOTS)*, San Francisco, CA, USA, Aug. 2000, pp. 172–179.

[CST00] C.-H. Chang, J.-P. Sheu, and Y.-C. Tseng, “A recursive frequency-splitting scheme for broadcasting hot video in VOD service,” unpublished slides, Department of CSIE, National Central University, Taiwan, 2000.

[CT99] S.-H. G. Chan and F. A. Tobagi, “On achieving profit in providing near video-on-demand services,” in *Proc. Int. Conf. on Communications (ICC)*, Vancouver, Canada, June 1999, pp. 988–994.

[DPK94] A. L. Drapeau, D. A. Patterson, and R. H. Katz, “Toward workload characterization of video server and digital library applications,” *ACM SIGMETRICS Performance Evaluation Review*, vol. 22, no. 1, pp. 274–275, 1994.

[DSS94] A. Dan, D. Sitaram, and P. Shahabuddin, “Scheduling policies for an on-demand video server with batching,” in *Proc. Annual ACM Int. Conf. on Multimedia (ACM MM)*, San Francisco, CA, USA, Oct. 1994, pp. 15–23.

[EFV99] D. L. Eager, M. C. Ferris, and M. K. Vernon, “Optimized regional caching for on-demand data delivery,” in *Proc. Conf. on Multimedia Computing and Networking (MMCN)*, San Jose, CA, USA, Jan. 1999, pp. 301–316.

[ES01] L. Engebretsen and M. Sudan. "Harmonic broadcasting is optimal assuming constant bit rate." July 2001. [Online]. Available: http://www.nada.kth.se/~enge/papers/Harmonic.pdf

[ES02] L. Engebretsen and M. Sudan, "Harmonic broadcasting is optimal," in *Proc. ACM/SIAM Symposium on Discrete Algorithms (SODA)*, San Francisco, CA, USA, Jan. 2002, pp. 431–432.

[EV98] D. L. Eager and M. K. Vernon, "Dynamic skyscraper broadcasts for video-on-demand," in *Proc. Workshop on Multimedia Information Systems (MIS)*, Istanbul, Turkey, Sept. 1998, pp. 18–32.

[EVZ99a] D. Eager, M. Vernon, and J. Zahorjan, "Minimizing bandwidth requirements for on-demand data delivery," in *Proc. Workshop on Multimedia Information Systems (MIS)*, Indian Wells, CA, USA, Oct. 1999, pp. 80–87.

[EVZ99b] D. Eager, M. Vernon, and J. Zahorjan, "Optimal and efficient merging schedules for video-on-demand servers," in *Proc. Annual ACM Int. Conf. on Multimedia (ACM MM)*, Orlando, FL, USA, Nov. 1999, pp. 199–202.

[EVZ99c] D. Eager, M. Vernon, and J. Zahorjan, "Optimal and efficient merging schedules for video-on-demand servers," University of Washington, Tech. Rep. CSE TR #99-08-01, 1999, unpublished report, thanks to Mr. Eager for sending me a draft version.

[EVZ01] D. Eager, M. Vernon, and J. Zahorjan, "Minimizing bandwidth requirements for on-demand data delivery," *IEEE Trans. on Knowledge and Data Engineering*, vol. 13, no. 5, pp. 742–757, 2001.

[GH90] A. D. Gelman and S. Halfin, "Analysis of resource sharing in information providing services," in *Proc. IEEE Global Telecommunications Conf. (GLOBECOM)*, San Diego, CA, USA, Dec. 1990, pp. 312–316.

[GKT02] L. Gao, J. Kurose, and D. Towsley, "Efficient schemes for broadcasting popular videos," *ACM Multimedia Systems Journal*, vol. 8, no. 4, pp. 284–294, 2002.

[GLM95] L. Golubchik, J. C. S. Lui, and R. R. Muntz, "Reducing I/O demand in video-on-demand storage servers," in *Proc. ACM Int. Conf. on Measurement and Modeling of Computer Systems (SIGMETRICS)*, Ottawa, Ontario, Canada, May 1995, pp. 25–36.

[GLM96] L. Golubchik, J. C. S. Lui, and R. R. Muntz, "Adaptive piggybacking: A novel technique for data sharing in video-on-demand storage servers," *ACM Multimedia Systems Journal*, vol. 4, no. 3, pp. 140–155, 1996.

[GT99] L. Gao and D. Towsley, "Supplying instantaneous video-on-demand services using controlled multicast," in *Proc. IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS)*, Florence, Italy, June 1999, pp. 117–121.

[GT01] L. Gao and D. Towsley, "Threshold-based multicast for continuous media delivery," *IEEE Trans. on Multimedia*, vol. 3, no. 4, pp. 405–414, 2001.

[HCS98a] K. A. Hua, Y. Cai, and S. Sheu, "Exploiting client bandwidth for more efficient video broadcast," in *Proc. IEEE Int. Conf. on Computer Communications and Networks (ICCCN)*, Lafayette, LA, USA, Oct. 1998, pp. 848–856.

[HCS98b] K. A. Hua, Y. Cai, and S. Sheu, "Patching: A multicast technique for true video-on-demand services," in *Proc. Annual ACM Int. Conf. on Multimedia (ACM MM)*, Bristol, UK, Sept. 1998, pp. 191–200.

[HNvB99] A. Hu, I. Nikolaidis, and P. van Beek, "On the design of efficient video-on-demand broadcast schedules," in *Proc. Int. Symposium on Modeling, Analysis, and Simulation of Computer Systems (MASCOTS)*, Maryland, MD, USA, Oct. 1999, pp. 262–269.

[HS97] K. A. Hua and S. Sheu, "Skyscraper broadcasting: A new broadcasting scheme for metropolitan video-on-demand systems," in *Proc. ACM Conf. on Applications, Technologies, Architectures and Protocols for Computer Communications (SIGCOMM)*, Cannes, France, Oct. 1997, pp. 89–100.

[HS00] K. A. Hua and S. Sheu, "An efficient periodic broadcast technique for digital video libraries," *ACM Journal of Multimedia Tools and Applications*, vol. 10, no. 2–3, pp. 157–177, 2000.

[Hu01] A. Hu, "Video-on-demand broadcasting protocols: A comprehensive study," in *Proc. Annual Joint Conf. of the IEEE Computer and Communcations Societies (INFOCOM)*, Anchorage, AK, USA, Apr. 2001, pp. 508–517.

[JT97a] L.-S. Juhn and L.-M. Tseng, "Harmonic broadcasting for video-on-demand service," *IEEE Trans. on Broadcasting*, vol. 43, no. 3, pp. 268–271, 1997.

[JT97b] L.-S. Juhn and L.-M. Tseng, "Staircase data broadcasting and receiving scheme for hot video service," *IEEE Trans. on Consumer Electronics*, vol. 43, no. 4, pp. 1110–1117, 1997.

[JT98] L.-S. Juhn and L.-M. Tseng, "Fast data broadcasting and receiving scheme for popular video service," *IEEE Trans. on Broadcasting*, vol. 44, no. 1, pp. 100–105, 1998.

[JWX02] R. Janakiraman, M. Waldvogel, and L. Xu, "Fuzzycast: Efficient video-on-demand over multicast," in *Proc. Annual Joint Conf. of the IEEE Computer and Communcations Societies (INFOCOM)*, New York, NY, USA, June 2002, pp. 920–929.

[KRB06a] S. Kotz, C. B. Read, and D. L. Banks, Eds., "Encyclopedia of statistical sciences," 2nd ed. Wiley-Interscience, 2006, vol. 15, pp. 9223–9224.

[KRB06b] S. Kotz, C. B. Read, and D. L. Banks, Eds., "Encyclopedia of statistical sciences," 2nd ed. Wiley-Interscience, 2006, vol. 9, pp. 6190–6197.

[Lee99] J. Y. B. Lee, "UVoD: A unified architecture for video-on-demand services," *IEEE Communication Letters*, vol. 3, pp. 277–279, 1999.

[LL00] V. C. H. Lee and J. Y. B. Lee, "Improving UVoD system efficiency with batching," in *Proc. Int. Conf. on Software, Telecommunications and Computer Networks (SoftCOM)*, Venice, Oct. 2000.

[LLG97] M. Y. Y. Leung, J. C. S. Lui, and L. Golubchik, "Buffer and I/O resource pre-allocation for implementing batching and buffering techniques for video-on-demand systems," in *Proc. IEEE Int. Conf. on Data Engineering (ICDE)*, Birmingham, UK, Apr. 1997, pp. 344–353.

[LLG98] S.-W. Lau, J. C. S. Lui, and L. Golubchik, "Merging video streams in a multimedia storage server: Complexity and heuristics," *ACM Multimedia Systems Journal*, vol. 6, no. 1, pp. 29–42, 1998.

[LV93] T. D. C. Little and D. Venkatesh, "Prospects for interactive video-on-demand," *IEEE Multimedia*, vol. 1, no. 3, pp. 14–24, 1993.

[MEVSS01] A. Mahanti, D. L. Eager, M. K. Vernon, and D. J. Sundaram-Stukel, "Scalable on-demand media streaming with packet loss recovery," in *Proc. ACM Conf. on Applications, Technologies, Architectures and Protocols for Computer Communications (SIGCOMM)*, San Diego, CA, USA, Aug. 2001, pp. 97–108.

[PCL98a] J.-F. Pâris, S. R. Carter, and D. D. E. Long, "Bandwidth allocation issues in near video on demand services," in *Proc. on Multimedia Technology and Applications Conf. (MTAC)*, Anaheim, CA, USA, Sept. 1998, pp. 196–200.

[PCL98b] J.-F. Pâris, S. W. Carter, and D. D. E. Long, "Efficient broadcasting protocols for video on demand," in *Proc. Int. Symposium on Modeling, Analysis, and Simulation of Computer Systems (MASCOTS)*, Montreal, Canada, July 1998, pp. 127–132.

[PCL98c] J.-F. Pâris, S. W. Carter, and D. D. E. Long, "A low bandwidth broadcasting protocol for video on demand," in *Proc. IEEE Int. Conf. on Computer Communications and Networks (ICCCN)*, Lafayette, LA, USA, Oct. 1998, pp. 690–697.

[PCL99a] J.-F. Pâris, S. W. Carter, and D. D. E. Long, "Combining pay-per-view and video-on-demand services," in *Proc. Int. Symposium on Modeling, Analysis, and Simulation of Computer Systems (MASCOTS)*, College Park, MD, USA, Oct. 1999, pp. 270–276.

[PCL99b] J.-F. Pâris, S. W. Carter, and D. D. E. Long, "A hybrid broadcasting protocol for video on demand," in *Proc. Conf. on Multimedia Computing and Networking (MMCN)*, San Jose, CA, USA, Jan. 1999, pp. 317–326.

[PCL00a] J.-F. Pâris, S. W. Carter, and D. D. E. Long, "A reactive broadcasting protocol for video on demand," in *Proc. Conf. on Multimedia Computing and Networking (MMCN)*, San Jose, CA, USA, Jan. 2000, pp. 216–223.

[PCL00b] J.-F. Pâris, S. W. Carter, and D. D. E. Long, "A universal distribution protocol for video-on-demand," in *Proc. IEEE Int. Conf. on Multimedia and Expo (ICME)*, New York City, NY, USA, July 2000, pp. 49–52.

[PL03] J.-F. Pâris and D. D. E. Long, "A variable bandwidth broadcasting protocol for video-on-demand," in *Proc. Conf. on Multimedia Computing and Networking (MMCN)*, Santa Clara, CA, USA, Jan. 2003, pp. 209–219.

[PLE03] A. Papagiannis, D. Lioupis, and S. Egglezos, "A scalable, low-cost VoD server with multicast support," in *Proc. Int. Conf. on. Information Technology, Research and Education (ITRE)*, Newark, NJ, USA, Aug. 2003, pp. 464–468.

[PLM99] J.-F. Pâris, D. D. E. Long, and P. E. Mantey, "Zero-delay broadcasting protocols for video-on-demand," in *Proc. Annual ACM Int. Conf. on Multimedia (ACM MM)*, Orlando, FL, USA, Oct./Nov. 1999, pp. 189–197.

[Pri05a] "PQU Corporation 512MB PC2100 DDR DIMM Memory." PriceGrabber.com. July 2005. [Online]. Available: http://www.pricegrabber.com/search_getprod.php/masterid=683285

[Pri05b] "Seagate Barracuda 160GB hard drive." PriceGrabber.com. July 2005. [Online]. Available: http://www.pricegrabber.com/search_getprod.php/masterid=683285

[Pâr99] J.-F. Pâris, "A simple low-bandwidth broadcasting protocol for video on demand," in *Proc. IEEE Int. Conf. on Computer Communications and Networks (ICCCN)*, Boston-Natick, MA, USA, Oct. 1999, pp. 118–123.

[Pâr01a] J.-F. Pâris, "A fixed-delay broadcasting protocol for video-on-demand," in *Proc. IEEE Int. Conf. on Computer Communications and Networks (ICCCN)*, Scottsdale, AZ, USA, Oct. 2001, pp. 418–423.

[Pâr01b] J.-F. Pâris, "A stream tapping protocol with partial preloading," in *Proc. Int. Symposium on Modeling, Analysis, and Simulation of Computer Systems (MASCOTS)*, Cincinnati, OH, USA, Aug. 2001, pp. 423–430.

[SGRT99] S. Sen, L. Gao, J. Rexford, and D. Towslew, "Optimal patching schemes for efficient multimedia streaming," in *Proc. ACM Int. Workshop on Network and Operation System Support for Digital Audio and Video (NOSSDAV)*, Basking Ridge, NJ, USA, June 1999, pp. 455–463.

[SHH97] S. Sheu, K. A. Hua, and T.-H. Hu, "Virtual batching: A new scheduling technique for video-on-demand servers," in *Proc. IEEE Int. Conf. on Database Systems for Advanced Applications (DASFAA)*, Melbourne, Australia, Apr. 1997, pp. 481–490.

[SHT97] S. Sheu, K. A. Hua, and W. Tavanapong, "Chaining: A generalized batching technique for video-on-demand systems," in *Proc. IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS)*, Ottawa, Canada, June 1997, pp. 110–117.

[TCS04] Y.-C. Tseng, Y.-C. Chueh, and J.-P. Sheu, "Seamless channel transition for the staircase video broadcasting scheme," *IEEE/ACM Trans. on Networking*, vol. 12, no. 3, pp. 559–571, 2004.

[THYL+01] Y.-C. Tseng, C.-M. Hsieh, M.-H. Yang, W.-H. Liao, and J.-P. Sheu, "Data broadcasting and seamless channel transition for highly-demanded videos," *IEEE Trans. on Communications*, vol. 49, no. 5, pp. 863–874, 2001.

[TYC02] Y.-C. Tseng, M.-H. Yang, and C.-H. Chang, "A recursive frequency-splitting scheme for broadcasting hot videos in VoD service," *IEEE Trans. on Communications*, vol. 50, no. 8, pp. 1348–1355, 2002.

[VI96] S. Viswanathan and T. Imielinski, "Metropolitan area video-on-demand service using pyramid broadcasting," *ACM Multimedia Systems Journal*, vol. 4, no. 4, pp. 197–208, 1996.

[Wal01a] G. Walz, Ed., "Lexikon der mathematik." Heidelberg/Berlin, Germany: Spektrum Akademischer Verlag, 2001, vol. 2, p. 96.

[Wal01b] G. Walz, Ed., "Lexikon der mathematik." Heidelberg/Berlin, Germany: Spektrum Akademischer Verlag, 2001, vol. 2, p. 99.

[Wei06] E. W. Weisstein. "Digamma function." MathWorld — A Wolfram Web Resource. 1999–2006. [Online]. Available: http://mathworld.wolfram.com/DigammaFunction.html

[Wik06a] "Vodcasting." Universität Trier, Medienwissenschaften Department. 2006. [Online]. Available: http://wellman.uni-trier.de/index.php?title=Vodcasting

[Wik06b] "Radio." Wikipedia. 2001–2006. [Online]. Available: http://en.wikipedia.org/wiki/Radio

[Wik06c] "Streaming media." Wikipedia. 2001–2006. [Online]. Available: http://en.wikipedia.org/wiki/Streaming_media

[Wik06d] "Television." Wikipedia. 2001–2006. [Online]. Available: http://en.wikipedia.org/wiki/Television

[Wik06e] "Poisson distribution." Wikipedia. 2002–2006. [Online]. Available: http://en.wikipedia.org/wiki/Poisson_distribution

[Wik06f] "Zipf's law." Wikipedia. 2002–2006. [Online]. Available: http://en.wikipedia.org/wiki/Zipf%27s_law

[Wik06g] "Bin packing problem." Wikipedia. 2003–2006. [Online]. Available: http://en.wikipedia.org/wiki/Bin_packing

[WMS06] M.-Y. Wu, S. Ma, and W. Shu, "Scheduled video delivery — a scalable on-demand video delivery scheme," *IEEE Trans. on Multimedia*, vol. 8, no. 1, pp. 179–187, 2006.

[YK03] E. M. Yan and T. Kameda, "An efficient VOD broadcasting scheme with user bandwidth limit," in *Proc. Conf. on Multimedia Computing and Networking (MMCN)*, Santa Clara, CA, USA, Jan. 2003, pp. 200–208.

[YP01] P.-F. You and J.-F. Pâris, “A better dynamic broadcasting protocol for video-on-demand,” in *Proc. IEEE Int. Performance, Computing and Communication Conf. (IPCCC)*, Phoenix, AZ, USA, Apr. 2001, pp. 84–89.

[ZP02] Q. Zhang and J.-F. Pâris, “A channel-based heuristic distribution protocol for video-on-demand,” in *Proc. IEEE Int. Conf. on Multimedia and Expo (ICME)*, Lausanne, Switzerland, Aug. 2002, pp. 245–248.