TECHNICAL RESEARCH REPORT

IP Multicast via Satellite: A Survey

by Gun Akkor, Michael Hadjitheodosiou, John S. Baras

CSHCN TR 2002-35
(ISR TR 2002-69)

IP Multicast via Satellite: A Survey

Gun Akkor, Michael Hadjitheodosiou, and John S. Baras
Center for Satellite and Hybrid Communication Networks,
Institute for Systems Research, University of Maryland,
College Park, MD 20742, USA.
e-mail: {akkor, michalis, baras}@isr.umd.edu

Abstract

Many of the emerging applications in the Internet, such as tele-conferencing, distance-learning, distributed games, software updates, and distributed computing would benefit from multicast services. In many of these applications, there is a need to distribute information to many sites that are widely dispersed from each other. Communication satellites are a natural technology option and are extremely well suited for carrying such services. Despite the potential of satellite multicast, there exists little support for satellite IP multicast services. Both Internet Engineering and Internet Research Task Forces (IETF and IRTF) have been involved in a research effort to identify the design space for a general purpose reliable multicast protocol and standardize certain protocol components as *building blocks*. However, for satellite multicast services, several of these components have a different design space. In this paper, we attempt to provide an overview of the design space and the ways in which the network deployment and application requirements affect the solution space. We maintain a similar taxonomy to that of the IETF efforts, and identify which key components of a general multicast protocol are affected by two of the most common satellite network deployment scenarios. We also highlight some of the issues which we think are critical in the development of next generation satellite IP multicast services.

Index Terms

IP Multicast, Satellite Networks, Taxonomy

I. Introduction

Many of the emerging applications in the Internet, such as tele-conferencing, distance-learning, distributed games, software updates, and distributed computing would benefit from multicast services. IP multicast [1] is a networking technique, which allows a source to send data to multiple destinations simultaneously using a single transmit operation. Multicasting makes efficient use of network bandwidth over unicast transmission. Besides performance improvements, multicasting allows deployment of truly distributed and collaborative applications. For all these reasons, IP multicast services will play an important role in the next generation Internet.

In many broadband applications, there is a need to distribute information to many sites that are widely dispersed from each other. Communication satellites are a natural technology option and are extremely well suited for carrying such services. Also, a satellite-based infrastructure can, in many cases, be established to offer widespread service provision with greater ease and simplicity than an infrastructure based on terrestrial broadband links. Thus, the ability to service many users and to solve the expensive ”last-mile” issue without dedicating to each user cable, fiber, switching equipment, and ports, etc. makes satellites attractive for broadband communications. Generally, it would be a lot easier and faster to set up a satellite connection compared to other broadband solutions. At the same time, satellites are also an attractive option for interconnection of geographically distributed high-speed networks. Hence, while much of the broadband communication today is carried via terrestrial links, satellites will come to play a greater and more important role. Next-generation satellite communication systems utilizing higher frequency bands such as the Ka-band, spot-beam technology and on-board processing are currently under development. Ka-band is very desirable for satellite communication systems, because it offers abundant bandwidth. The use of spot-beam and on-board processing technologies enable the use of small, low-power, low-cost user terminals that offer two-way direct communication [2]–[5]. These systems are likely to play an important role in the global communication infrastructure.

Both Internet Engineering and Internet Research Task Forces (IETF and IRTF) have been involved in a research effort to identify the design space for a general purpose reliable multicast protocol and standardize certain protocol components as *building blocks*. However, it is arguably impossible to achieve a *one-size-fits-all* design recipe, due to the diverse range of multicast applications and the variety of multicast network topologies. Most of the recently proposed multicast protocols address the requirements of different types of applications and network topologies.

Despite the potential of satellite multicast, there exists little support for satellite IP multicast services. Since the Internet protocols have been designed without taking into account the inherent characteristics of the physical medium and the deployed network, efficient integration of the next-generation satellite systems with IP multicast services requires the study and adaptation of these protocols [6]. In this paper, we attempt to provide an overview of the design space and the ways in which the network deployment and application requirements affect the solution space in a satellite environment. An earlier survey of protocols, functions and mechanisms for multi-point communication can be found in [7]. [8] gives a taxonomy of multicast transport protocols and classifies them according to important common features. This work has been taken forward in the context of satellite networks in [9].

In this paper, we maintain a similar taxonomy to that in [8] and [9] because it is also consistent with the IETF standardization efforts [10]–[13]. However, we first discuss the characteristics of satellite networks and identify which key components of a general multicast protocol is affected by certain network deployment scenarios. In Section III, we present a survey of current alternatives on the design of multicast protocols and discuss their suitability to our network scenarios. In Section IV, we give recommendations on the desirable futures of protocols for satellite IP multicast. Section V concludes the paper.

II. Characteristics of Satellite Networks

Satellites systems are becoming an integral component of broadband network. They have several characteristics that are particularly attractive in this case, such as breadth of broadcast "reach", ubiquitous access, low-cost global coverage, large and most importantly flexible capacity. At the same time, characteristics such as the relatively long propagation delay and the complications of a wireless channel, constitute serious shortcomings. Finally, a few other characteristics such as on-board switching, regeneration and spot-beam technology offer opportunities and represent challenges that can transform some of the shortcomings into strengths.

In this section, we provide a description of the technology challenges that must be overcome to permit the successful and harmonious integration of IP multicast services and satellite networks.

A. Constraints Imposed by the Channel

Spectrum congestion of frequency bands (L, S, C, or Ku bands) that had been conventionally used for satellite services is leading to the use of higher frequency bands such as the Ka-band (20-30 GHz). Ka-band is very desirable for multimedia communications because it offers wider bandwidth. Such large bandwidth segments are unavailable at lower frequencies, such as Ku-band (12-18 GHz) and C-band (4-6 GHz), which were until recently the bands used for Fixed Satellite Service (FSS) communications. Most VSAT and DBS TV systems in operation today use portions of the Ku-band [14], while most Ka-band systems are scheduled to start offering customer service after the year 2002.

However, Ka-band has one major disadvantage. Rain and atmospheric attenuation present a significant challenge to transmission of signals at Ka-band frequencies [15]–[17], as the molecular water vapor absorption resonance frequency is located at the center of the band, at 22.3 GHz. The resulting signal fading causes not only just random bit errors but also longer bursty errors that can cause the transmission operation to be completely “off” for short periods of time, even though the channel would be of extremely high quality for most of the operation. Moreover, such extreme attenuation usually occurs in cells of small dimension (5-10 km in diameter) compared to the footprint of a narrow spot-beam (650 km in diameter at a geosynchronous orbit). Therefore, at any time instant, user stations experience very different channel conditions, leading to a *heterogeneity* in the reception quality. Continuing demand for additional bandwidth has forced commercial satellite system designers to consider even higher frequency bands, namely the so-called V-band (40-75 GHz). Some military satellite systems already operate in this frequency range. These higher frequencies offer additional challenges to the designer such as more severe multi-path fading and scattering of transmitted signals. Clearly, satellite channels suffer from higher error rates and bursty error patterns than terrestrial wireline networks, and extra coding protection is required to maintain ”fiber-like” quality of service (QoS).

Another set back for satellite networks is the long propagation delays associated with satellite channels. The physical distance of a communications satellite from the source and destination imposes a significant propagation delay on every transmission. This delay can introduce problems not just in real-time delay-sensitive applications but also adversely affect the performance of certain protocols, such as ATM or TCP/IP. In geostationary earth orbit (GEO) and medium earth orbit (MEO) systems, the propagation delay (260 ms and 100 ms, respectively) is much higher than in low earth orbit (LEO) systems (10 ms), but in LEO constellations the need to route a signal through multiple satellites imposes delay, too, and might also increase the variance of the delay.

Unreliability of the satellite channel, *long propagation delays* and the *heterogeneity* in the channel quality impose a significant challenge in the design of several reliable multicast protocol components. These challenges are discussed in detail in Section III.

B. Constraints Imposed by the Network Topology

We consider two of the most common topologies for multicast service support involving broadband satellite:

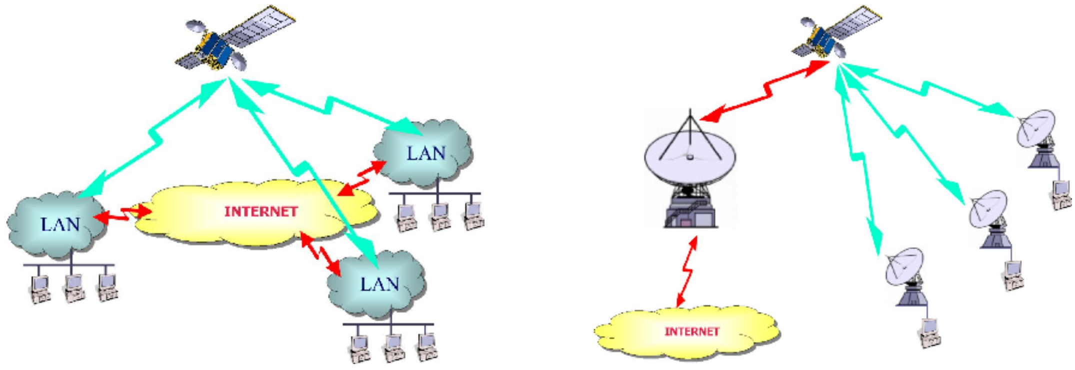
(i) Satellite networks can be deployed as a *backbone* for interconnection of geographically distributed high-speed local area networks (LAN). In this scenario, local area networks are connected to the satellite backbone through one or more gateway nodes that have satellite uplink capability (Fig. 1(a)). This network topology gives rise to a hierarchical structure: satellite and gateway nodes acting as an overlay network for the LAN(s). In general, LAN(s) may also have access to a terrestrial core network.

(ii) A *direct-to-home* (or *direct-to-business*) deployment, where the network consists of independent ground terminals with direct uni- or bi-directional connection to the satellite. In this scenario, network has a star topology and user terminals have no access to other networks. Ground terminals access the terrestrial core network through a gateway node located at the so called *network operations center* (NOC) (Fig. 1(b)).

Network topology is one of the most important constraints, because it strictly limits the mechanisms that can be used for reliability, flow and congestion control. The limitations and strengths of current design alternatives for a given network topology are discussed in the next section.

III. Taxonomy of Multicast Building Blocks

A reliable multicast protocol operates through inter-working of several different components. The topology of the deployed network and the requirements of the target application constrains



(a) Case I: Backbone Deployment (b) Case II: Direct-to-home Deployment

Fig. 1. Satellite Network Topologies

the design space of these components. In this section, we provide a taxonomy of the design features which are desirable for a reliable multicast protocol.

A. Loss Detection and Feedback

To provide reliability, a protocol needs to identify the packets which failed to reach a given destination. This is achieved through loss-notification (feedback) packets returned to the designated source(s) [1](e.g. sender) by the receivers. Traditionally, this feedback has been in one of the following forms:

- Positive Acknowledgments (ACK): Receivers return ACK packets to the designated source, indicating which packets have been received.
- Negative Acknowledgments (NACK): Receivers return NACK packets to the designated source, indicating only packets that are missing by a receiver.
- A hybrid approach (i.e. both ACK(s) and NACK(s)).

For multicast services, ACK-based loss-notification has been shown to lead to the *ACK implosion* problem. The problem arises when a large number of multicast receivers return an ACK packet for every packet they receive correctly, causing a serious network congestion around the links of the source. Another potential problem is that, the source is required to keep track

[1]We adopt the term *designated source* in place of the sender because, unlike the unicast communication, some multicast protocols use intermediate nodes to collect and aggregate feedback information.

of the size, and the current state of the reporting receiver-set, in order to identify the last data packet correctly received by all receivers. In a large scale multicast application, the memory and processing load becomes prohibitive.

NACK-based feedback alleviates some of the problems. The performance comparison study presented in [18] confirms that NACK-based multicast transport protocols deliver better performance than their ACK-based counterparts in wireline terrestrial networks. In NACK-based feedback, receivers detect missing packets by checking for gaps in the packet sequence numbers and report to the designated source. The source neither needs to know the size of the receiver-set at any point in time, nor keeps track of the current state of every receiver in its group. Also, the number of NACK packets is expected to be less than that of ACK packets at low error rates. A disadvantage is that, it is more difficult for the source to know when it can free the transmission buffers, since NACK packets may be lost in transmission.

For a satellite multicast application, there are other potential problems related to loss detection and feedback. Due to high error rates of the satellite links and large receiver-sets, even NACK-based feedback may lead to an implosion problem. Moreover, feedback packets need to be transmitted through the shared satellite uplink, causing a waste of already scarce satellite resources.

Because of the aforementioned problems, existing multicast protocols couple their feedback scheme with supporting mechanisms such as feedback aggregation [19]–[22] and feedback suppression [23]–[25]. However, the suitability of these algorithms for satellite multicast services depends on the topology of the network, as we will further elaborate in Section III-C.

B. Loss Reduction

Fundamental component of a reliable transport protocol is packet recovery. Automatic Repeat Request (ARQ) is a well-known technique for this purpose. In ARQ, sources respond to loss notification reports by retransmitting the missing packets. In a satellite multicast application, pure-ARQ for packet recovery turns out to be inefficient for several reasons. Long propagation delays associated with satellite links make the delay incurred during this repeat-request process unacceptable for many delay sensitive applications (e.g. video streaming). The frequent and deep fades observed in satellite links result in high error rates and bursty error patterns. Therefore, even if the feedback mechanism is efficient in returning the loss information back to the sources, in a typical wide area satellite deployment, considerable network bandwidth and processing time would be spent by retransmitting the different packets lost at different receivers. Several early papers exist on use of ARQ in satellite multicast applications [26]–[29].

Forward error correction (FEC) coding is a well-known technique for protecting data against corruption [30]–[32]. FEC coding involves addition of redundant (parity) data to the original stream either at the physical bit level or at the packet level. In some FEC schemes, redundant data are added to the physical bit stream. This may improve the satellite link's effective bit error rate (BER), reducing the number of errors without an increase in the satellite transmitted power (EIRP). However, in the case of an IP multicast protocol, the network layers will detect corrupted packets and discard them or the transport layers can use packet authentication to discard corrupted packets. Therefore, the primary application of FEC codes to IP multicast protocols is as an erasure code [12], [33]. Erasure codes allow generation of n encoding packets from k original data packets. In such cases, the original k data packets can be reconstructed as long as k out of the transmitted n encoding packets are received correctly. Use of FEC erasure coding offers some solution to a number of problems.

One immediate benefit of packet level FEC coding is the reduction in the number of lost packets. This minimizes the number of feedback reports as well as the need for retransmissions, and improves scalability. FEC coding has an additional benefit for satellite multicast service. Unlike the unicast communication, where the retransmission exclusively benefits the receiver that made the request, in multicast, the protocol has the option to either unicast or multicast the retransmitted packet. Multicast of retransmitted packets would benefit multiple receivers in case of correlated loss. However, it may as well degrade channel utilization in case of uncorrelated loss. Because of the broadcast nature of the satellite channel, all packets are received by all members of the receiver-set. Transmitting parity packets in place of corrupted data packets significantly improves the channel utilization, since a single parity packet can repair loss of different packets at different receivers of the same session. Also, the protocol becomes more scalable in the number of receivers. The protocol does not need to know which packets have been lost by the receivers of the session but only the maximum number of packets lost by any receiver of the session. Therefore, the feedback from a single receiver has been reduced [34]–[37].

FEC is very effective at reducing the repair traffic per packet loss. However, it requires that the data to be sent is grouped into blocks, which can add to end-to-end latency. Another drawback of FEC is that more link capacity is used and that there is high cost of encoding and decoding of transmitted data [38]. It is important to note that FEC techniques trade link capacity to improve the probability of successful transmissions. Therefore, there has been a lot of effort to come up with adaptive schemes that dynamically adjust the amount of parity generated [39]–[44].

C. Feedback Suppression and Aggregation

Several existing multicast protocols adopt feedback suppression and feedback aggregation schemes to control the number and flow of feedback packets. Applicability of these methods in a satellite multicast application depends heavily on the architecture of the deployed network. Feedback aggregation is applicable in networks where a logical or physical hierarchy is possible. In a *backbone* scenario, it is possible for intermediate LAN routers to aggregate feedback packets from receiver nodes and to report only the aggregated feedback information to the satellite gateways. Gateway nodes can further aggregate reports from several routers to pass only a single feedback information back to the source. This aggregation towards the root of the hierarchy cuts back on the number of feedback reports and prevents receivers from contacting the source directly, enabling the protocol to scale over large set of receivers [19]–[22]. However, it requires additional functionality at the intermediate routers of the terrestrial network [45]. This raises questions about whether the router extensions can be standardized and automatically configured. In a *direct-to-home* deployment, feedback aggregation would not be possible because the star topology does not lend itself into a hierarchy.

Feedback suppression is applicable if NACK-based feedback reports are multicast to the network. This allows receivers to listen to the feedback reports and to suppress their own feedback reports if the received NACK report is for the same missing packet [23]–[25]. Coupled with random timers and statistical back-off, feedback suppression would, in an ideal setting, let only one feedback report to be returned to the sender for every missing packet. However, multicast forwarding of feedback reports must be scoped by use of *time-to-live* (TTL) fields to minimize the flooding of the network by feedback packets. It is difficult to estimate the right parameters for timers. Feedback suppression is not applicable for *direct-to-home* deployment since we assume there is no direct connection among receivers. Even though the satellite can relay the feedback packet by broadcasting it to all the receivers, the delay incurred during the coordination may be unacceptable in some applications.

D. Packet Recovery

In unicast communication, feedback reports are returned to sender and retransmissions are initiated by the sender. In order to avoid feedback implosion and traffic concentration at the sender and to reduce the packet recovery latency, several multicast protocols adopt *local recovery* methods [19], [22], [24], [25] . Local recovery allows designated nodes to buffer data packets, and initiate retransmissions on behalf of the sender. Therefore, receivers first try to recover missing packets through these nodes. If the missing packet can not be recovered through the use of designated nodes, then the retransmission request is forwarded to the sender.

In networks where a logical or physical hierarchy is possible, intermediate routers can buffer the packets forwarded through them and initiate retransmission up on receiving a request. Coupled with feedback aggregation (Section III-C) *router-assisted* local recovery is shown to improve the scalability and the performance of reliable multicast protocols [37], [46]–[50].

In a *backbone* deployment, satellite gateways are in a very good position to assist in the local recovery together with other intermediate routers. However, as it is the case with feedback aggregation, *router-assisted* local recovery requires support from intermediate network elements and its availability depends on existence of router extensions [45], [51]. A generalization of *router-assisted* local recovery is possible, if receiver nodes are also allowed to respond to retransmission requests. In this case, feedback packets need to be multicast to the (sub)set of receivers. However, efficient scoping of feedback and repair packets must be implemented to avoid flooding of network. In a *direct-to-home* deployment, all receivers receive packets directly from the satellite and there are no intermediate routers between satellite and the receivers. Therefore, it is difficult to implement local recovery for this type of networks.

E. Congestion and Flow Control

The Internet congestion control has been achieved by the widespread use of TCP protocol. Multicast protocols need to have suitable techniques to avoid congestion and to determine a “fair” share of the available resources with respect to TCP network traffic. Therefore, most of the recent studies and proposals for congestion control of multicast traffic in the Internet try to be *TCP-friendly* by adopting the behavior of TCP congestion control algorithm as the standard of fairness.

TCP protocol uses a window-based congestion control algorithm, where the size of the congestion window determines the number of outstanding (i.e. not yet acknowledged) packets in the network. The window size is regulated in response to ACK packets returned to the sender. Lack of an ACK packet acts as a loss (and/or delay) indication causing the window size to be decreased, otherwise it is increased. However, it is difficult to extend window-based regulation to multicast communication. Applying a single window size to the whole session has been shown to restrict the throughput more severely than required by the bottleneck path [52] because, the throughput is dictated by the receiver with the longest round-trip delay (RTT) rather than the bottleneck receiver (i.e the receiver with the smallest throughput). Use of per packet ACK(s) for regulation of the session rate is another problem for multicast applications, because it causes, as in the case of receiver feedback for reliability, feedback implosion problem at the sender. Another fundamental problem in multicast congestion control is the *loss path multiplicity problem* [53]. Unlike unicast congestion control, where the sender regulates its transmission rate based on the loss indications from a single receiver, a multicast source receives loss indications from multiple receivers, reflecting diverse conditions in various parts of the network. For example, if a source receives multiple loss indications from a set of receivers that are behind the same bottleneck link and reduces its rate in response to each such loss indication, than it would be overcompansating for a single loss. Therefore, loss indications have to be appropriately combined when making a single rate control decision [54], [55]. The multicast source must regulate its rate according to the most congested path, or equivalently, according to the lossiest receiver in the multicast group.

Some of the previous proposals on the issues of error recovery and feedback implosion (Section III-A-Section III-C) can be incorporated into congestion control schemes. MTCP protocol [56] is based on a multi-level logical tree, which is used to aggregate ACK-based feedback packets, where the root is the sender and other nodes in the tree are receivers. Internal nodes, referred to as *sender's agents* (SA) monitor the congestion level of their children and compute the minimum bandwidth available from the sender to their children by maintaining a TCP-like congestion window. When sending an ACK to its parent, a SA includes in the ACK packet, a summary of the congestion level of its children. The sender regulates its rate based on its own summary. PGMCC [57], [58] uses a window-based control that mimics the TCP behavior, but runs it between the sender and the bottleneck receiver avoiding some of the problems addressed in [52]. The bottleneck receiver, referred to as the *acker*, is chosen dynamically based on receivers' loss rate reports embedded into NACK-based feedback reports. ACK-based feedback is only used between the *acker* and the sender for congestion control.

Difficulties in implementing window-based congestion control have lead to rate- and equation-based control protocols [59]–[61]. These protocols calculate the rate at the sender based on the long-term throughput equation for TCP protocol. The equation gives the expected rate of a TCP flow as a function of the steady-state loss event rate, round-trip time, and the packet size. Receivers measure their loss rate and include this information in the feedback packets returned to the sender. The sender uses the feedback messages to measure and estimate the round-trip time to the receivers and uses the rate equation to derive the acceptable transmission rate. However, estimation of round-trip time without causing excessive traffic at the sender presents a formidable task, and is the key component of equation-based control protocols.

Use of single rate congestion control algorithms for multicast services necessarily forces the sender to adjust its rate according to the bottleneck receiver. Heterogeneity of the receiver-set penalizes the receivers with higher throughputs. To avoid this problem, several authors have suggested layered organization of the transmitted data [62], [63]. In layered multicast transmission, sender distributes the data using layers with increasing bandwidth. Receivers add or drop layers based on the perceived bandwidth of the path to the sender. This allows each receiver to match its desired rate. In order to use layered transmission as a congestion control algorithm, however, receivers behind the same router have to act in a coordinated manner, because the action of a receiver dropping a layer (due to perceived congestion) would not be effective unless all the receivers sharing the bottleneck drop the same layer. Moreover, if a receiver causes congestion on the bottleneck link by adding a new layer, another might interpret the resulting losses as a consequence of its current level and drop a layer. Therefore, fairness of the protocol requires that all of them have a similar behavior [63]. As another point, a layered organization is only possible if the data to be transferred supports it. Layered organization is more effective for video and audio-conferencing applications, where data can be organized into layers of increasing quality [64]. However, for bulk data transfer, this is not possible. By using a proper arrangement, a similar approach could be used [65]–[67] if the data can be processed beforehand. For data generated on the fly, it becomes more complex to find a suitable data organization without wasting network capacity [68]. Layered transmission is less suitable for satellite multicast, since all receivers see the same effective link rate.

All current proposals try to achieve TCP compatibility because, TCP protocol is the *de facto* standard for end-to-end congestion control in the Internet. However, deployment of satellite networks either as an access network to Internet or a core network for global services poses new questions on the performance of TCP protocol over hybrid or pure satellite networks. TCP protocol, initially designed for terrestrial networks with low link error rates, assumes all loss indications are due congestion in the network (i.e overflow of router buffers). Therefore, sender decreases its rate each time a packet loss is detected. For satellite networks, this causes unnecessary performance degradation when the losses are due to link errors. Long propagation delays associated with satellite links are another performance concern for TCP protocol, since the congestion control window is regulated per round-trip time. These and similar issues are addressed by the research community in several occasions [69]–[71]. Therefore, proposals for multicast congestion control should take into account the characteristics of the satellite networks.

In a *direct-to-home* type deployment scenario, there exists a flow control problem rather than a congestion problem, since the satellite uplink/downlink bandwidth is fixed and has to be shared amount all active connections. Therefore, the sources have to throttle their transmission rates not in response to loss indications (in this scenario they are due to link errors), but according to the backlog of buffers at the satellite terminals.

F. Other Issues

There are several other issues that need to be addressed in providing satellite IP multicast services. *Security* is particularly important, since many satellite terminals share access to the received signal. Multicast services add more complexity to several security components such as the key distribution, key revocation, source and receiver authentication. The primary issue constraining the design space is related to receiver-set scaling. Key distribution and revocation may be significantly affected by receiver-set scaling. However, multicast services may actually benefit from the single-hop star topology of the satellite network and the broadcast nature of the channel. Another issue that the security model must take into account is the role of third-parties. Many multicast protocols rely on intermediate nodes for retransmissions, feedback aggregation and suppression. In particular, it must be clear whether such third-parties are trusted or untrusted. Satellites may act as trusted parties since they are protected from physical tampering, however this would require on-board implementation of additional memory and processing capabilities. A taxonomy and discussion on security for IP multicast can be found in [7], [10], [72]. [9] briefly addresses the vulnerability of reliable multicast protocols to various malicious attacks.

Group management is another component that must be considered in implementing multicast services. A multicast application may implement implicit or explicit group membership management. In the former case, the protocol does not require knowledge of group membership. This implicit group management model fits well with the way IP multicast works, where senders do not keep track of the receiver-set. However, explicit group membership information may be required by other components of the protocol to implement data reliability, flow control and security. The security algorithms may require membership information for implementing access control, key distribution, key revocation and receiver authentication. Flow control algorithms may remove a member that consistently reports losses to safeguard transmission to other receivers. For example, for satellite communication, group members having transmission outages due to rain may be removed from the group.

Another key issue that we choose not to address in this paper is the *multicast routing* and construction of the multicast distribution tree. Several papers address the issue and a survey on the subject can be found in [7], [73], [74]. Satellite networks have a unique structure that may prove to be beneficial in terms of multicast routing. Many inter-domain multicast routing protocols construct the distribution tree by making use of designated central nodes [75]–[77]. The single-hop star topology of the core network and the broadcast nature of the channel (Fig. 1(b)) place the satellite in an advantageous position for construction and maintenance of the multicast tree. An overlay satellite network (Fig. 1(a)) is advantageous in constructing intra-domain multicast distribution trees. However, it is not clear how existing protocols may be integrated for seamless operation of hybrid terrestrial-satellite networks.

IV. Future Challenges and Research Issues

In this section, we outline some of the critical issues in the development of next generation satellite IP multicast services. Some of the problems outlined in this section have already been identified as the key problems for multicast services by the research community as well as the

IETF. However, in the case of satellite multicast services, these problems have either different roots or different solution spaces. Therefore, in this section, we revisit these questions and try to provide design guidelines.

A. Bulk Feedback

In the wireline terrestrial networks, feedback from multicast receivers is problematic for two reasons. First, the flow of feedback packets from the multicast receivers, which are located typically at the leaves of the multicast distribution tree, to the multicast sender, which is the root of the tree, causes network traffic concentration around links of the sender. Secondly, sender is required to process a large number of feedback packets, which may be prohibitive in some cases. However, for satellite IP multicast services, there are other aspects to this problem.

In satellite multicast networks, all receivers receive packets directly from the satellite and there are no intermediate routers between the satellite and the receivers. In other words, the receivers are only single hop away from the satellite and there is no physical hierarchy between the satellite and the receivers. The multicast schemes that are based on physical or logical hierarchy for aggregation of receiver feedback or local recovery can not be applied in this network topology. As a result, the *feedback implosion* problem becomes very challenging.

The problem is further aggravated by the fact that the return channel (uplink) is a shared medium and the resources are scarce. This problem does not exist in terrestrial wireline networks and is overlooked in satellite multicast scenarios. Assigning a separate return channel to every multicast receiver would result in the waste of resources and certainly would not scale. Considering that (i) feedback information may contain redundant information (due to correlations among the loss pattern of receivers) and, (ii) most multicast algorithms only need to track the behavior of a subset of receivers with the worst case channel conditions, the challenge is to design efficient algorithms to select and filter-out information from multiple receivers to allow only the most relevant feedback information to be conveyed to the source using as few channels as possible. In the ideal case, a single unified feedback information, as it is in the unicast transmission, would be transmitted using only a single uplink channel per multicast session. However, it is not feasible for multicast users to communicate through the satellite channel to facilitate coordination among the receiver-set, since this would cause more overhead than conveying the feedback information. Hence, any such algorithm needs to rely on statistical timers and probabilistic behavior. Therefore, there is a three-way trade-off between the size of the receiver-set, number of channels reserved for feedback transmission and the timeliness of the feedback information.

B. Integrated FEC

In wireline terrestrial networks, packet losses are attributed to buffer overflows in the intermediate routers, where as in satellite networks, packets are lost due to impairments of the satellite channel. IEFT proposals recommend integration of the FEC schemes at the transport protocol level in order to aid in the multicast retransmission. The broadcast nature of the satellite channels boosts the benefits of using FEC coding at the transport level. However, transport layer application of the FEC coding requires processing blocks of packets together. This may have implications in terms of the delay since the transport protocol needs to wait for multiple packets before decoding a packet and passing it to higher layers. Moreover, encoding of large packets would have a considerable computational load on the sources. Therefore, there are applications of FEC coding at the data link layer for generating redundant bits to protect individual packets against corruption. This has the effect of reducing the effective bit error rate of channel. The key challenge is to allow cross-layering all the way from physical layer up to transport layer. Because, unlike the wireline networks, where the physical layer provides a stable low-error rate medium, channel state in satellite scenario is highly variable and countermeasures at the lower levels impact the performance of the transport layer.

C. Flow Control and Congestion

There is an effort in the research community to provide a ubiquitous congestion control algorithm for hybrid satellite-terrestrial networks. This requires careful modification of the TCP protocol to match the characteristics of the satellite channels. Another approach is to split a connection at the terrestrial-satellite network interface, and to run TCP between the end nodes and the satellite network while running a customized algorithm in the core network. The latter approach has several advantages in practice. It allows satellite provider to optimize the traffic flow inside the core network to achieve high utilization. Also, it makes it possible to push the congestion to the edges of the satellite network and let the TCP protocol's congestion control mechanism to take care of the congestion in the terrestrial portion of the network. Inside the core network, the problem is reduced to a flow control problem rather than a congestion control problem since the satellite bandwidth is fixed and is to be shared among all the active connections.

For multicast services, the challenge then is to determine the flow rate of the session. In the wireline terrestrial networks, links in different locations of the network are shared by different flows. Hence they experience different congestion levels and do not have the same capacity. A multicast session either adjusts its rate proportional to the capacity of the receiver behind the bottleneck link or uses transmission rate layering. Transmission rate layering allows receivers to use one of a set of pre-defined rates which matches the network capacity of their location. However, the latter approach is not appropriate for satellite networks. Due to the single-hop star topology of the satellite core network, all receivers see the same link speed. Heterogeneity of the receiver-set, in this context, is because of the difference between the channel quality of different receivers rather than the congestion. The rate of the transmission must be adjusted to achieve the target transmission quality at the receiver with the worst channel quality. Also, different receivers may demand different quality levels. Therefore, the challenge is to deal with heterogeneous receivers and to satisfy their quality requirements fairly. Some of the quality measures such as packet loss rate, packet delay and delay jitter are perceived at higher protocols layers while the performance is effected by the low level physical phenomena. Therefore, higher protocol layers should interact with the lower levels to achieve performance improvements.

D. On-board Buffering and Processing

Implementing on-board processing and buffering has several merits that would improve the performance of satellite services, including but not limited to IP multicast. For satellites operating around 8 GHz and above, on-board buffering of messages and automatic repeat request (ARQ) can be used to minimize the effects of rain outages [16]. This is particularly advantageous at higher frequencies and at lower elevation angles, both of which significantly increase the excess path loss due to rain. The benefits are more pronounced in case of multicast services. By performing ARQ on-board the satellite, it is possible to minimize the number of feedback packets that needs to be propagated back to the source. This would improve both the bandwidth efficiency of the multicast service and the transmission delay.

On-board processing allows regeneration, duplication and encoding of packets. Implementing on-board processing would benefit multicast services by allowing adaptive FEC coding and parity (re)transmission. It also allows isolating the uplink and downlink which would be beneficial in a multicast scenario since, unicast uplink communication between the source and satellite is likely to have different parameters than the satellite downlink. For example, at the downlink, more redundancy may be needed to accommodate the requirements of the receiver with the worst channel quality than the uplink channel. Isolation of the uplink and downlink, in this case, would allow a higher code rate at the uplink, and a lower rate code at the downlink. Coupled with on-board buffering, it makes it possible to dynamically change the transmission parameters of the downlink as the multicast membership changes, while keeping the uplink communication the same.

V. Conclusion

In this paper, we have presented a taxonomy and survey of various design alternatives for supporting multicast services and discussed how the design space of some of the common protocol components are constrained in the context of satellite networks. Our classification is based on the IETF *building blocks* for multicast protocols, but highlights which key components of a general multicast protocol are affected by the two of the most common satellite network deployment scenarios.

We also outlined some of the issues, which we think are critical in the development of next generation satellite IP multicast services. Some of these problems, such as feedback suppression and integration of FEC coding at the transport or link layer, have counterparts in terrestrial networks, but they have to be addressed separately while taking into consideration the unique characteristics of satellite networks. We believe that efficient solutions to these problems and development of new technologies such as on-board processing and buffering would demonstrate the true value of satellite networks in the global communication infrastructure.

References

[1] S. Deering, “Host extensions for IP multicast,” IETF Request for Comments, August 1989, RFC 1112.

[2] E. J. Fitzpatrick, “SPACEWAY system summary,” *Space Communications*, vol. 13, pp. 7–23, 1995.

[3] F. Gargione, T. Iida, F. Valdoni, and F. Vatalaro, “Services, technologies, and systems at Ka band and beyond: A survey,” *IEEE J. Select. Areas Commun.*, vol. 17, no. 2, pp. 133–144, February 1999.

[4] “HUGHES SPACEWAY System,” www.spaceway.com.

[5] “HUGHES DIRECWAY System,” www.direcway.com.

[6] F. Filali and W. Dabbous, "Issues on IP multicast service behaviour over the next generation satellite-terrestrial hybrid networks," in *Proc. of Computer and Communications*, 2001, pp. 417–424.

[7] C. Diot, W. Dabbous, and J. Crowcroft, "Multipoint communication: A survey of protocols, functions and mechanisms," *IEEE J. Select. Areas Commun.*, vol. 15, no. 3, pp. 277–290, 1997.

[8] K. Obraczka, "Multicast transport protocols: A survey and taxonomy," *IEEE Commun. Mag.*, January 1998.

[9] M. W. Koyabe and G. Fairhurst, "Reliable multicast via satellite: A comparison survey and taxonomy," *Int. J. Sat. Comm.*, vol. 19, no. 1, pp. 3–23, 2001.

[10] M. Handley, S. Floyd, B. Whetten, R. Kermode, L. Vicisano, and M. Luby, "The reliable multicast design space for bulk data transfer," IETF Request for Comments, August 2000, RFC2887.

[11] B. Whetten, L. Vicisano, R. Kermode, M. Handley, S. Floyd, and M. Luby, "Reliable multicast transport building blocks for one-to-many bulk-data transfer," IETF Request for Comments, January 2001, RFC3048.

[12] M. Luby, L. Vicisano, J. Gemmell, L. Rizzo, M. Handley, and J. Crowcroft, "Forward error correction building blocks," Internet Draft, February 2002, work in progress, draft-ietf-rmt-bb-fec-06.txt.

[13] B. Adamson, C. Bormann, M. Handley, and J. Macker, "Nack-oriented reliable multicast protocol building blocks," IETF Draft, November 2001, work in progress, draft-ietf-rmt-bb-norm-03.txt.

[14] R. A. Nelson, "V-band: Expansion of the spectrum frontier," *Via Satellite*, February 1998.

[15] E. H. Satorius and L. H. Tong, "Analysis of rain compensation algorithm for K/Ka-band communications," *International Journal of Satellite Communications*, vol. 14, no. 3, 1996.

[16] M. J. Willis, "Fade countermeasures and VSAT systems for $20/30$ Ghz," Ph.D. dissertation, University of Surrey, 1993.

[17] F. Davarian, "Ka-band propagation research using ACTS," *International Journal of Satellite Communications*, vol. 14, no. 3, 1996.

[18] D. Towsley, J. Kurose, and S. Pingali, "A comparison of sender-initiated and receiver-initiated reliable multicast protocols," *IEEE J. Select. Areas Commun.*, vol. 15, no. 3, pp. 398–406, April 1997.

[19] R. Yavatkar, J. Griffioen, and M. Sudan, "A reliable dissemination protocol for interactive collaborative applications," University of Kentucky, Tech. Rep., 1995.

[20] J. C. Lin and S. Paul, "RMTP: A reliable multicast transport protocol," in *Proc. of IEEE INFOCOM*, March 1996, pp. 1414–1424.

[21] S. Paul, K. K. Sabnani, and D. M. Kristol, "Reliable multicast transport protocol (RMTP)," *IEEE J. Select. Areas Commun.*, vol. 15, no. 3, pp. 407–421, April 1997.

[22] D. Chiu, S. Hurst, M. Kadansky, and J. Wesley, "TRAM: A tree-based reliable multicast protocol," Sun Microsystems Labs, Tech. Rep. SML-TR-98-66, July 1998.

[23] J. P. Macker and W. K. Dang, "The multicast dissemination protocol," Naval Research Laboratory, Tech. Rep., 1996.

[24] S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang, "A reliable multicast framework for light-weight sessions and application level framing," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 407–421, 1997.

[25] T. Speakman, D. Farinacci, S. Lin, and A. Tweedly, "Pragmatic general multicast (PGM) transport protocol specification," Internet Draft, June 1999, work in progress, draft-speakman-pgm-spec-03.txt.

[26] S. B. Calo and M. C. Easton, "A broadcast protocol for file transfer to multiple sites," *IEEE Trans. Commun.*, vol. 29, pp. 1701–1707, November 1981.

[27] K. Mase, T. Takenaka, H. Yamamato, and M. Shinohara, "Go-back-N ARQ schemes for point-to-multipoint communications," *IEEE Trans. Commun.*, vol. 31, pp. 583–589, April 1983.

[28] I. S. Gopal and J. M. Jaffe, "Point-to-multipoint communication over broadcast links," *IEEE Trans. Commun.*, vol. 32, pp. 1034–1044, September 1984.

[29] K. Sabnani and M. Schwartz, "Multidestination protocols for satellite broadcast channels," *IEEE Trans. Commun.*, vol. 33, pp. 232–240, March 1985.

[30] R. E. Blahut, *Theory and Practice of Error Control Codes*. MA: Addison Wesley, 1984.

[31] A. J. MacAuley, "Reliable broadband communication using a burst erasure correcting code," in *Proc. of ACM SIGCOMM*, Philadelphia, PA, September 1990.

[32] L. Rizzo, "Effective erasure codes for reliable computer communication protocols," *ACM Computer Communication Review*, vol. 27, no. 2, pp. 24–36, April 1997.

[33] M. Luby, L. Vicisano, J. Gemmell, L. Rizzo, M. Handley, and J. Crowcroft, "The use of forward error correction in reliable mutlicast," Internet Draft, February 2002, work in progress, draft-ietf-rmt-info-fec-02.txt.

[34] J. Nonnenmacher, E. W. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," *IEEE/ACM Trans. Networking*, vol. 6, no. 4, pp. 349–361, August 1998.

[35] L. Rizzo and L. Vicisano, "A reliable multicast data distribution protocol based on software FEC techniques," in *Proc. of HPCS*, 1997.

[36] D. Rubenstein, S. Kasera, D. Towsley, and J. Kurose, "Improving reliable multicast using active parity encoding services," in *Proc. of IEEE INFOCOM*, 1999.

[37] I. Rhee, S. R. Joshi, M. Lee, S. Muthukrishnan, and V. Ozdemir, "Layered multicast recovery," in *Proc. of IEEE INFOCOM*, vol. 2, 2000, pp. 805–813.

[38] L. Rizzo, "On the feasibility of software FEC," DEIT, Tech. Rep., January 1997, http://www.iet.unipi.it/ luigi/softfec.ps.

[39] H. Linder, I. Miloucheva, and H. D. Clausen, "A FEC based multicast transport protocol for multimedia applications in satellite environments," in *Proc. of IPCCC*, 1997, pp. 419–425.

[40] D. E. Gossink and J. P. Macker, "Reliable multicast and integrated parity retranmission with channel estimation considerations," in *Proc. of IEEE GLOBECOM*, vol. 6, 1998, pp. 3608–3613.

[41] J. Yoon, A. Bestavroz, and I. Matta, "Adaptive reliable multicast," in *Proc. of IEEE ICC*, September 1999, pp. 1542–1546.

[42] S. Payne, "Reliable multicast via satellite," Master's thesis, University of Maryland, College Park, MD, 1999.

[43] S. Cho, A. Goulart, I. F. Akyildiz, and N. Jayant, "An adaptive FEC with QoS provisioning for real-time traffic in LEO satellite networks," in *Proc. of IEEE ICC*, June 2001, pp. 2938–2942.

[44] K. Manousakis, "Reliable multicasting based on air caching for flat heirarchy networks," Master's thesis, University of Maryland, College Park, MD, 2002.

[45] P. Radoslavov, C. Papadopoulos, R. Govindan, and D. Estrin, "A comparison of application-level and router-assisted hierarchical schemes for reliable multicast," in *Proc. of IEEE INFOCOM*, 2001, pp. 229–238.

[46] S. K. Kasera, J. Kurose, and D. Towsley, "A comparison of server-based and receiver-based local recovery approaches for reliable multicast," CMPSCI, Tech. Rep., December 1997.

[47] J. Nonnenmacher, M. Lacher, M. Jung, E. W. Biersack, and G. Carle, "How bad is reliable multicast without local recovery ?" in *Proc. of IEEE INFOCOM*, vol. 3, 1998, pp. 972–979.

[48] S. K. Kumar, "Scalable reliable multicast in wide area networks," Ph.D. dissertation, University of Massachussets Amherst, Amherst, MA, September 1999.

[49] H. Jianhua, "Performance evaluation on the satellite based reliable multicast with and without local recovery," in *Proc. of ECUMN*, 2000, pp. 311–318.

[50] S. K. Kumar, S. Bhattacharyya, M. Keaton, D. Kiwior, J. Kurose, D. Towsley, and S. Zabele, "Scalable fair reliable multicast using active services," *IEEE Network*, January 2000.

[51] D. Rubenstein, N. F. Maxemchuk, and D. Shur, "A centralized, tree-based approach to network repair service for multicast streaming media," AT&T, Tech. Rep., November 1999.

[52] J. S. Golestani and K. K. Sabnani, "Fundamental observations on multicast congestion control in the Internet," in *Proc. of IEEE INFOCOM*, 1999.

[53] S. Bhattacharyya, D. Towsley, and J. Kurose, "The loss multiplicity problem in multicast congestion control," in *Proc. of IEEE INFOCOM*, 1999.

[54] ——, "Design and analysis of loss indication filters for multicast congestion control," CMPSCI, Tech. Rep., February 2000.

[55] P. Thapliyal, J. Li, and S. Kalyanaraman, "LE-SBCC: Loss-event oriented source-based multicast congestion control," *Multimedia Tools and Applications Journal*, vol. 17, no. 2-3, 2002.

[56] I. Rhee, N. Balaguru, and G. N. Rouskas, "MTCP: Scalable TCP-like congestion control for reliable multicast," in *Proc. of IEEE INFOCOM*, 1999.

[57] L. Rizzo, "PGMCC: A TCP-friendly single-rate multicast congestion control scheme," in *Proc. of ACM SIGCOMM*, 2000.

[58] L. Rizzo, G. Iannaccone, L. Vicisano, and M. Handley, "PGMCC single rate multicast congestion control: protocol specification," Internet Draft, June 2002, work in progress, draft-ietf-rmt-bb-pgmcc-01.txt.

[59] J. Widmer and M. Handley, "Extending equation-based congestion control to multicast applications," in *Proc. in ACM SIGCOMM*, 2001.

[60] ——, "TCP-friendly multicast congestion control: protocol specification," Internet Draft, November 2001, work in progress, draft-ietf-rmt-bb-tfmcc-00.txt.

[61] J. P. Macker and R. B. Adamson, "A TCP-friendly, rate-based mechanism for NACK-oriented reliable multicast congestion control," in *Proc. of IEEE GLOBECOM*, vol. 3, 2001, pp. 1620–1625.

[62] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in *Proc. of ACM SIGCOMM*, 1996.

[63] L. Vicisano, J. Crowcroft, and L. Rizzo, "TCP-like congestion control for layered multicast data transfer," in *Proc. of IEEE INFOCOM*, 1998.

[64] X. Li, S. Paul, and M. Ammar, "Layered video multicast with retransmissions: Evaluation of hierarchical rate control," in *Proc. of IEEE INFOCOM*, 1998.

[65] L. Vicisano, "Notes on a cumulative layered organization of data packets across multiple streams with different rates," UCL Computer Science Research Note, 1998, work in progess.

[66] V. Roca, "Packet scheduling for heterogeneous multicast transmissions," in *Proc. of PHSN*, 1999.

[67] M. J. Donahoo, M. H. Ammar, and E. W. Zegura, "Multiple channel multicast scheduling for scalable bulk-data transport," in *Proc. of IEEE INFOCOM*, vol. 2, 1999, pp. 847–855.

[68] V. Roca, "Analysis of several scheduling algorithms for the heterogeneous multicast distribution of data flows generated on the fly," September 1999, work in progress.

[69] V. G. Bharadwaj, J. S. Baras, and N. P. Butts, "Internet service via broadband satellite networks," Center for Satellite and Hybrid Communications, Tech. Rep., 1999.

[70] I. F. Akyildiz, G. Morabito, and S. Palazzo, "TCP-Peach: A new congestion control scheme for satellite IP networks," *IEEE/ACM Trans. Networking*, vol. 9, no. 3, pp. 307–321, June 2001.

[71] X. Zhou, X. Liu, and J. S. Baras, “Flow control at satellite gateways,” Center for Satellite and Hybrid Communication Networks, Tech. Rep., 2002.

[72] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas, “Multicast security: A taxonomy and efficient constructions,” in *Proc. of IEEE INFOCOM*, vol. 2, 1999, pp. 708–716.

[73] M. Ramalho, “Intra- and inter-domain multicast routing protocols: a survey and taxonomy,” *IEEE Communications Survey and Tutorials Online Magazine*, vol. 3, no. 1, 2000, http://www.comsoc.org/livepubs/surveys/index.html.

[74] V. Roca, L. Costa, R. Vida, A. Dracinschi, and S. Fdida, “A survey of multicast technologies,” Work in progress, September 2000, version 1.3.

[75] A. J. Ballardie, P. Francis, and J. Crowcroft, “Core based trees,” in *Proc. ACM SIGCOMM*, 1993.

[76] T. Ballardie, B. Cain, and Z. Zhang, “Core based trees multicast routing,” IETF Draft, March 1998, work in progress, draft-ietf-idmr-cbt-spec-v3-00.txt.

[77] B. Fenner, M. Handley, H. Holbrook, and I. Kouvelas, “Protocol independent multicast - sparse mode: Protocol specification,” IETF Draft, July 2001, work in progress, draft-ietf-pim-sm-v2-new-03.txt.