

The Satellite Reliable Distribution Protocol (SRDP)

Franco Tommasi, Simone Molendini, and Andrea Tricco

Abstract – The Satellite Reliable Distribution Protocol (SRDP) is proposed and discussed. SRDP is designed for satellite-based data broadcasting and aims at transmitting reliably multicast via satellite Internet applications' data in an efficient and scalable way.

SRDP operates in three steps, called phases. In phase 1 the sender transmits a file through the satellite internet adding a reasonable degree of redundancy. In phase 2 the sender retransmits a set of new redundant packets to recover some packets that were lost during phase 1. Phase 3 allows each receiver to request all its missing packets by a TCP connection to the sender. In order to evaluate the performance of the SRDP, we have considered two different types of loss behaviours: random and burst. Our results show how to choose the amount of redundancy in order to optimize the performance of the SRDP.

Index terms: Reliable multicast, satellite, scalability

I. INTRODUCTION

The satellite is an almost ideal medium for wide-scale broadcasting: today satellites reach a number of households and businesses far beyond that of traditional broadcast methods [1]. Satellite broadcasting is not as expensive as one might think if the potential number of receivers and the covered geographic area are taken into account. The satellite then can today be used to transmit reliably bulk data like software products, off-line multimedia content, CD-ROM images and product catalogues. A protocol to allow Internet hosts to receive such data through satellites is then required.

Many reliable multicast protocols have been proposed until now and several comparisons exist [2-7], but the use of these protocols in the Internet has been lamented [8]. Since a "one-size-fits-all" solution to the reliable multicast problems does not seem to exist, in this document we propose and discuss the Satellite Reliable Distribution Protocol (SRDP), a protocol designed for satellite-based data broadcasting. From an architectural point of view, SRDP is an application and some of its messages are encapsulated in UDP, others in TCP.

A common weakness of other protocols supporting reliable multicast is that they do not carefully consider satellite networks' peculiarities. Satellite networks suffer from relatively high bit error rates compared with terrestrial fibre trunks. Moreover, the feedback channel could be unavailable for some receivers. SRDP is designed for the networks where the satellite is used as a means to distribute data to a group of receivers. From this point of view, the SRDP protocol aims at

transmitting reliable multicast information in an efficient and scalable way for the star-based topologies that are intrinsic to the satellite transmissions. In fact, SRDP can profit from feedback coming from receivers provided with an outbound connection but it can be used as well to transmit data to receivers provided only with an incoming satellite connection. For example, it would be meaningful to have SRDP transmissions without feedback if receivers are terminals that wait for SRDP services (an instance of such service is the updating of information panels).

The rest of the paper is organized as follows. Section II characterizes the reference topology and Section III provides a basic description of the SRDP. Section IV deals with the performance of SRDP in two contexts of random and burst losses presenting the results of simulations. Conclusions are drawn in Section V.

II. INTERNETWORKING WITH THE SRDP

In this section we analyse the reference topology and the design steps to give SRDP efficiency and fairness.

A. Reference topology and definitions

Satellites provide broadcast transmissions in the fullest sense, since their footprints can be made to cover large areas of the earth. In general terms, basic satellite services provide for simplex communication. Telecommunication and broadcasting services today are showing an increasing trend to add interactivity to traditional TV. There are several ways of implementing a feedback channel for satellite multicast services: commonly-proposed terrestrial return channels include PSTN (Public Switched Telephone Network), ISDN (Integrated Services Digital Network) and GSM (Global System for Mobile communication). Moreover, there is a huge worldwide interest in defining a satellite return channel (duplex transmission) and standards, as RCS (Return Channel via Satellite) [10], to use it already exist: among other benefits, satellite based return channels allow to cover rural or remote areas not covered by terrestrial services.

In our topology, see figure 1, we discuss the SRDP protocol with reference to its use in a "satellite internet", that is a connection of IP networks containing at least a satellite link. Such network might be (or not) connected to the Internet, in both instances using IP protocol.

receivers, based on the sender's estimate of the number of receivers, the available outgoing bandwidth etc.

D. Congestion control

Congestion collapse in today's Internet is mainly prevented by the congestion control schemes defined by TCP [12]. It is then important to be careful about the congestion-related issues raised by the use of UDP-based protocols. Particularly, UDP multicast applications have the potential to do more congestion-related damages to the Internet than UDP unicast ones. Some general-purpose reliable multicast protocols have been published before the impact of congestion caused by reliable multicast was fully appreciated. These cases were recently deprecated [8].

To analyse the problem of congestion in our topology let us consider it in four separate sub-paths: a) the Reaching path, b) the Satellite link, c) the Receiving path and d) the Return path.

a) The Reaching path. This is the most critical part of the network. Ss is available to a potentially large number of senders some of which are possibly hops away. There is therefore an obvious and classical risk of congestion connected to such transmissions. There are a number of possible solutions: TCP tunnelling, Explicit Congestion Notification, QoS mechanisms, to name a few. By adopting some of those mechanism it is possible to access Ss from a larger set of locations. On the other hand if no such mechanism is adopted, the use of Ss must be restricted to senders which are close to Ss and connected to it by well-provisioned links. Another problem to be considered is the amount of bandwidth to be assigned to each transmission. Since more than one sender can access Ss at the same time, an out-of-band method for coordinating and assigning available bandwidth is needed (for example the Service Scheduler designed in [13]).

b) The Satellite link. Since each Satellite link is used by only one Ss at the time the problem of congestion on such a link is automatically solved by the out-of-band method quoted in the last paragraph.

c) The Receiving path. First of all it must be noted that the present price of a satellite receiver is so affordable that the simplest way to avoid congestion problem is to attach directly a Receiver host to a dedicated Satellite receiver, thus avoiding all congestion problems. If a Satellite receiver is used to serve an entire LAN the risk of congestion is still low as normally the satellite bandwidth and the LAN bandwidth are magnitude orders away. A TCP tunnel from the Sr to each receiver Host is activated in the case that the LAN could be congested or in the case that the Satellite receiver is used to serve distant LANs.

d) The Return path. SRDP uses TCP on the Return channel so no special congestion issue is raised.

In conclusion, because of the peculiarities of the satellite internet we felt that no special congestion related mechanism should be included in SRDP. Congestion controlling methods such the ones quoted above must be used both in the cases of

satellite internets where the sender or the receivers are far from the satellite link and in the use with non-satellite internets.

III. DESCRIPTION OF THE SRDP PROTOCOL

SRDP operations are performed in three different phases. During phase 1, also called "Satellite phase", a file is transmitted through the satellite internet adding a reasonable degree of redundancy. During phase 2, also "Hybrid phase", a set of redundant packets are retransmitted to attempt the recovery of packets lost during phase 1. To this purpose, the sender periodically polls the receivers for feedback about the number of lost packets (Stat Req and Stat Rep messages). During phase 3, also "Terrestrial phase", each receiver is allowed to request all its missing packets by opening a TCP connection to the Sender. Phase 3 is then aimed at eliminating the remaining losses. These three phases are separated in time and identified by SRDP synchronization messages.

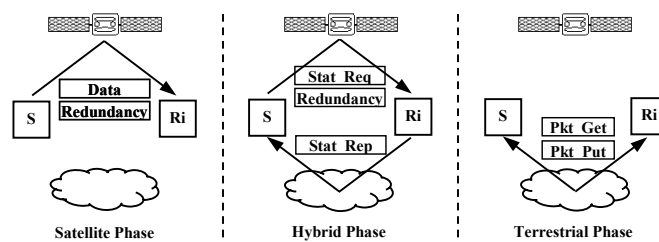


Figure 2. The Three phases.

SRDP has been designed to be used with the satellite networks where an important goal is to reduce the waste of resources. In these environments the time that is dedicated to the transmission is often limited and it constrains the total number of redundant packets that can be added. The typical process of bandwidth purchase can be outlined as follows. A certain amount of data packets K must be transmitted. If the channel bandwidth available is B (packets/s), one has to decide a priori (based on experience and/or on the results of our simulations) the amount of redundancy packets, H , to add to the data. The time slot to be purchased will then at least be $(K+H)/B$ seconds. For this reason, SRDP assumes the parameter H as fixed.

In order to understand how the SRDP protocol uses redundancy, let H be the number of redundant packets that are available to the sender, and K be the number of original packets obtained from the file. The choice of the H parameter influences the performances of the protocol. If H is large, the reliability increases but the goodput of the protocol diminishes. Instead, if H is small, the protocol has a better performance in terms of goodput but the reliability is reduced.

We also define the following relation: $H = H1 + H2$, where $H1$ and $H2$ are the quantities of redundancy that are added in phases 1 and 2 respectively. For instance, if $H2 = 0$ the behaviour of the SRDP protocol is equal to that of the protocols that use unidirectional FEC techniques. Instead, if $H1 = 0$ the protocol relies on an ARQ mechanism only. The results of the simulations, presented in the following sections, aim at helping us in the right choice of these parameters. It

Following the SFS algorithm the sender will send a sequence of Stat_Req messages until it receives a number of replies equal to or larger than E, or the parameter ND reaches the value 1. In this latter case the sender will receive a Stat_Rep message from all the receivers endowed with a return channel. In other words, the purpose of the SFS algorithm is to dynamically adjust the value of the parameter ND in order to limit the number of Stat_Rep messages received and to avoid the implosion problems. Such algorithm allows to quickly converge with a $O(\log(RM))$ complexity. It is important to note that during a SRDP transmission the number of receiving members is not supposed to increase because no one can join the SRDP session after it has just started.

B.3 Repair Strategy

After collecting the Stat_Rep messages as adequately filtered by the SFS algorithm, the sender has to identify the sequence of redundant packets that will be transmitted in phase 2. Several algorithms can be adopted, we will propose a simple algorithm in the following. In order to determine the number of redundant packets to transmit, the following definitions are given:

- $h1$ is the number of redundant packets transmitted in phase 1 for each TG;
- $h2[i]$ is the counter of redundant packets transmitted in phase 2 for the i -th TG;
- Packets_Count is the counter of redundant packets transmitted for all the TGs;
- Statq[i] is the number of lost packets for the i -th TG ($1 \leq q \leq E$) as derived from each of the E Stat_Rep messages;
- $\max_vector[i] = \text{Max}_q(\text{Statq}[i]); 1 \leq q \leq E; \forall i \in [1, N];$

The algorithm goes as follows:

- 1) $h2[i]$ is set to 0, $\forall i \in [1, N]$.
- 2) Packets_Count is set to 0.
- 3) TGs are scanned cyclically starting from the first ($i = 1$).
- 4) At each step a redundant packet is transmitted if ($h2[i] < \max_vector[i]$ AND $\max_vector[i] > h1$). If a redundant packet is transmitted, Packets_Count and $h2[i]$ are increased by 1.
- 5) Then the next TG is considered ($i = i + 1$) and the above operation is performed until $h2[i]$ reaches $\max_vector[i]$ or Packets_Count reaches H2.
- 6) If the algorithm stops before Packets_Count reaches H2 then the sender consumes the remaining redundancy ($H2 - \text{Packets_Count}$) cyclically sending a new redundant packet for each TG.

The sender could use other more sophisticated algorithms to exploit the similitude between Stat_Rep vectors and to optimise the number of receivers that can benefit from the new

sequence of redundant packets. But it must be kept in mind that such algorithms increase further the load on the sender.

C. Phase 3: Terrestrial Phase

If a receiver has not correctly received all the TGs, after the satellite phase and the hybrid phase, the only chance for it to receive the missing packets is to use its return channel. During this phase receivers request all missing packets to the sender and receive them in a client/server manner using the Pkt_Get and Pkt_Put messages.

The protocol must take into account two important effects that could reduce the performance of the sender in an unacceptable way. First of all some of the receivers might ask for a large number of packets, for example if the receiver's hardware is inadequate. SRDP is inherently a satellite protocol and this phase must be confined to the retransmission of just a little number of packets to a reduced number of members of the original receivers' community. The SRDP sender, which has the session manager role in the protocol reduces then the group's membership eliminating the worst of such receivers. That is, the sender limits the maximum number of packets that can be demanded (Max_Pkt). Beyond such threshold the receiver cannot benefit from this phase. A second effect that may have an impact on the sender's performance is the race between receivers, that immediately after the phase 2, compete to obtain the sender's resources. To reduce this effect the sender uses the spread factor parameter (SF). The sender manages the access defining a temporal value SF, proportional to the expected number of connections that are estimated by the use of the SFS algorithm. Such value is broadcasted together with Max_Pkt and Pkt_Conn (see below) inside the SRDP End Message (see table 1). A receiver must wait, before trying to connect the sender, for a time randomly chosen into the range $[0, SF]$. Moreover, a single connection must not request more than a fixed number of packets per connection (Pkt_Conn) in order to limit the connection time for each receiver. Each receiver must perform the following steps when accessing the sender:

- 1) if the number of missing packets is greater than Max_Pkt, then the algorithm stops and the reception fails;
- 2) waits randomly for a time chosen in the interval $[0, SF]$, then opens the connection;
- 3) sends a request for a maximum of Pkt_Conn packets;
- 4) if more packets are needed, iterates step 2, otherwise ends.

The sender is the session manager of all the phases. Nonetheless, communication in this phase is a receiver-initiated one because each receiver is the initiator of the connection. A receiver may even decide to connect the sender hours or days after the end of the satellite transmission; a sender may keep the file for several days: in any case, SRDP does not guarantee the receiver that the sender is able to correctly answer the request.

D. Message Types

Table 1 lists the message types currently defined in SRDP. The Begin message is the first message transmitted by the

equal to 1. The interleaving spreads the burst losses over more transmission groups partially transforming the burst errors in random losses. Therefore it is not useful to interleave for the purposes of this section.

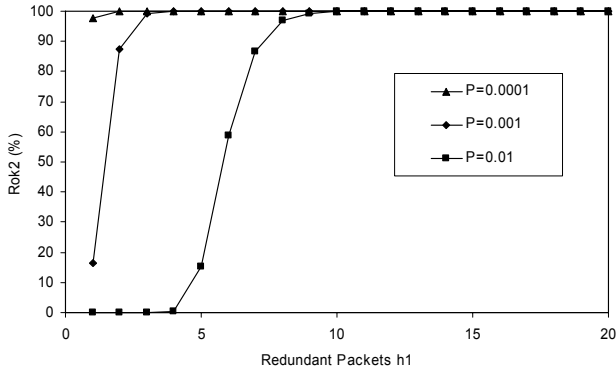


Figure 3. R_{OK2} as a function of $h1$ and P . $H2=0$, $R = 100$ and $M = 50\%$ of R .

The simulation whose results are shown in figure 3 investigates R_{OK2} as a function of the quantity of redundancy packets per TG, $h1$, and of the packet error rate P . R_{OK2} increases as $h1$ increases: for instance, with P is equal to 0.01, R_{OK2} is equal to zero when the number of redundant packets is lower than 4, while it reaches 100% (“complete reliability”, see section II B) for $h1 \geq 10$.

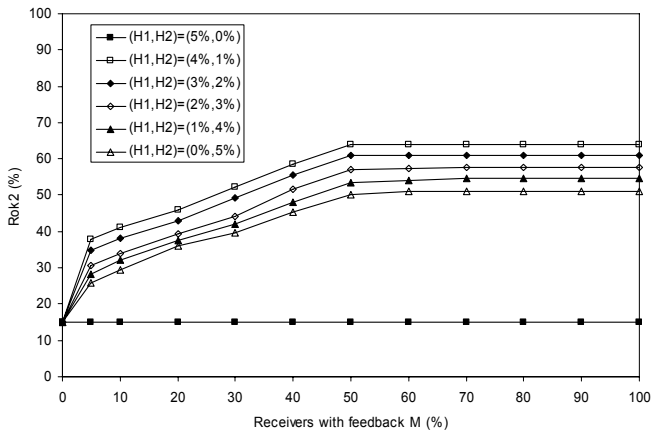


Figure 4. R_{OK2} as a function of M . $H=5\%$, $R = 100$, $P = 0.01$.

The purpose of the simulation whose results are shown in figure 4 has been to examine the role that the percentage of receivers endowed with a Return path M has on the reliability of the protocol. When the protocol uses only phase 1, corresponding to the curve (5%,0%) in the figure, R_{OK2} has a constant value (the packet level FEC used in phase 1 is independent from the feedback availability). Instead, the other curves show that R_{OK2} increases with the percentage of receivers M . It can be observed that, if M exceeds the 50%, the curves are levelling off near the maximum value. This behaviour suggests that to benefit from phase 2 it is not necessary that return channels are available for more than 50% of the receivers (and this value has been then used in the following simulations).

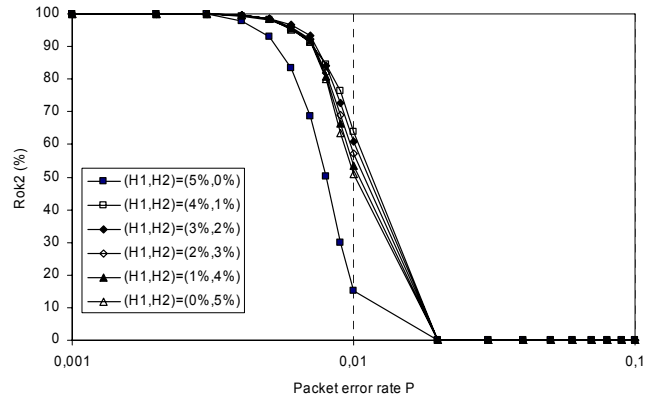


Figure 5 R_{OK2} as a function of P . $H=5\%$, $R = 100$, $M = 50\%$ of R .

In figure 5 we analyse the effects that different packets error rates P have on the performance of our protocol. The rain causes an attenuation between 8dB and 10dB for frequencies between 12GHz and 14GHz, increasing the bit error probability b by a factor between 10 and 200 [1]. Figure 5 shows R_{OK2} as a function of the packet error rate P for different values of the couple $(H1, H2)$: 6 curves are presented for each possible couple of entire percentage values obtained by a fixed value of H ($H=5\%$). For all figures, the values of the parameters H , $H1$, $H2$ are expressed in terms of the percentage of the total number of data packets. For example, the couple (5%,0%) means that the protocol is only using phase 1, on the contrary, when $(H1, H2)$ is equal to (0%,5%) SRDP is using all the redundancy in phase 2.

First of all, figure 5 easily shows that for each curve R_{OK2} decreases when the packets error rate P increases. It can be seen that all curves have the same threshold behaviour and that the chosen couples of values $(H1, H2)$ are the critical ones (as could be derived from figure 3). Comparing the curves (5%,0%) and (4%,1%) in the figure, one can observe that the latter one has a better behaviour. That is, even if only 50 percent of the receivers has a return channel, all the receivers benefit from the phase 2 redundancy.

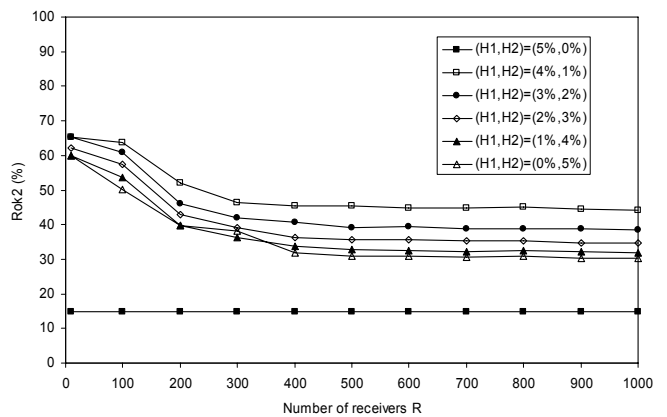


Figure 6. R_{OK2} as a function of R . $H=5\%$, $M = 50\%$, $P = 0.01$.

Figure 6 shows R_{OK2} as the function of the number of the receivers R for different values of the couple $(H1, H2)$. Like

- [6] D. Towsley, J. Kurose, and S. Pingali, "A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols", *IEEE/JSAC*, 15(3), 398-406, 1997.
- [7] M. W. Koyabe, G. Fairhurst, "Reliable Multicast via Satellite: A Comparison Survey and Taxonomy", *International Journal of Satellite Communications (IJSC)*, Vol:24(1), 21-26, 2001.
- [8] RFC2357 A. Mankin, A. Romanow, S. Bradner, V. Paxson. "IETF Criteria for Evaluating Reliable Multicast Transport and Application Protocols". June 1998.
- [9] K. Sollins, "The TFTP Protocol (Revision 2)", IETF RFC1350, July 1992.
- [10] ETSI EN 301 790, "Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems", V1.2.2, European Standard (Telecommunications series), December 2000.
- [11] L. Rizzo, "Effective erasure codes for reliable computer communication protocols". *ACM SIGCOMM Computer Communication Review*, vol.27, no.2, p. 24-36, April 1997.
- [12] V. Jacobson, 'Congestion Avoidance and Control', in *SIGCOMM '88: Communications Architectures and Protocols*, Stanford, USA, 18, 314 - 329, 1988.
- [13] European Space Agency (ESA) MODUS project, <http://telecom.estec.esa.nl/artes/artes3/fileincludes/projects/162ssi/ssi.cfm>.
- [14] J. Nonnenmacher, E. W. Biersack, D Towsley "Parity-Based Loss Recovery For Reliable Multicast Transmission". *IEEE/ACM Transaction on Networking*, 6(4): 349-361, August 1998.
- [15] UCB/LBNL/VINT. Network Simulator – ns2, <http://wwwmash.cs.berkeley.edu/ns/>.
- [16] L. Kleinrock, "Theory, Volume 1, Queueing Systems", John Wiley & Sons, 1975.
- [17] Clio srl. Available: <http://www.clio.it>
- [18] IANA, Internet Assigned Numbers Authority, <http://www.iana.org>.



Francesco Tommasi graduated from University of Pisa, Pisa, Italy, in electronic engineering in 1984. In the same year he joined CSELT, Turin, Italy, where he became involved in image processing. In 1988 he joined the Istituto per la Ricerca Scientifica e Tecnologica, Trento, Italy, where he made researches in image motion analysis. He has been head of software development at Cigraph, Venice, Italy. He is now associate professor at the Engineering Faculty of the University of Lecce, Italy. His present research interests include networking and the applications of satellite to distance learning.



Simone Molendini received the M.S. and Ph.D. degrees in computer engineering from the University of Lecce, Lecce, Italy, in 1999 and 2004, respectively. Currently, he is a Research Assistant in the Department of Innovation Engineering, University of Lecce. His currently research interests are in Internet signaling, mobility and satellite applications. Active in several IETF working groups, he is co-author of the RFC2961 - "RSVP Refresh Overhead Reduction Extensions", year 2001.



Andrea Tricco received the M.S. and Ph.D. degrees in computer engineering from the University of Lecce, Lecce, Italy, in 2001 and 2005, respectively. Currently, he is a Research Assistant in the Department of Innovation Engineering, University of Lecce. His currently research interests are in Internet signaling, mobility and satellite applications.