INTERNATIONAL JOURNAL OF SATELLITE COMMUNICATIONS AND NETWORKING
Int. J. Satell. Commun. Network. 2011; **29**:163–184
Published online 9 April 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sat.959

On the fairness of return channel capacity allocation in DVB-RCS-based satellite networks

P. Chatziparaskevas*,†, G. Koltsidas and F.-N. Pavlidou

Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Panepistimioupolis, 54124 Thessaloniki, Greece

SUMMARY

Broadband satellite access (BSA) systems can form an alternative path for the provision of Internet access in areas with poor network infrastructure. The DVB-RCS standard introduced the specifications of an interaction channel for two-way BSA networks. In this study, a new dynamic scheduling strategy for the interaction channel of GEO satellite networks is proposed, evaluated and compared with a typical Round Robin scheme. The main idea of the proposed strategy is to change, prior to each allocation, the sequence according to which bandwidth is assigned to the satellite terminals. The new sequence is fully specified by a set of fairness indices, each one related to a unique terminal and updated after each allocation. Along with the examined scheduling strategies, two capacity request calculation techniques found in the literature are also evaluated and compared through a series of simulations. Copyright © 2010 John Wiley & Sons, Ltd.

Received 10 July 2008; Revised 1 October 2009; Accepted 2 October 2009

KEY WORDS: DVB-RCS; DCA; QoS; capacity allocation; capacity requests; fairness

1. INTRODUCTION

Interactivity is a key issue in modern telecommunication networks even in traditional distribution services like TV or radio [1]. Modern satellite services require two-way interactive connections. The DVB-RCS (Digital Video Broadcasting–Return Channel via Satellite) standard [2], published by the European Telecommunications Standards Institute, defines the air interface specification of the interaction channel of two-way broadband satellite access (BSA) systems. The standard defines the mechanisms for the provision of a full range of modern applications such as Internet browsing, e-mail, VoIP and video conference via satellite. The BSA network gives also the possibility of integrating satellite networks into the traditional terrestrial telecommunications infrastructure where the IP protocol is dominant. Moreover, as real time and multimedia applications are gaining popularity, it must be capable of providing end-to-end quality of service (QoS) for traffic with strict demands on jitter, delay and packet loss rate [3]. The DVB-RCS standard also defines specific mechanisms for QoS provision that can cooperate with the respective QoS architecture of IP networks.

Furthermore, given a limited return channel capacity, the link utilization must be optimized to serve as many users as possible. Dynamic capacity allocation (DCA) mechanisms defined in the DVB-RCS standard can ensure QoS provision for bursty traffic generated by the Return

*Correspondence to: P. Chatziparaskeyas, Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Panepistimioupolis, 54124 Thessaloniki, Greece.
†E-mail: panch@auth.gr

Contract/grant sponsor: European Satellite Communications Network of Excellence (SatNEx II)

Channel Satellite Terminals (RCSTs). In this study, we consider the CF-DAMA (Combined Free–Demand Assignment Multiple Access) protocol as the DCA mechanism that regulates return channel access. According to CF-DAMA, capacity is allocated primarily on a request basis and afterwards freely among the terminals. Forward channel traffic (towards the RCSTs), which tends to be smooth, is served by a static TDM scheme according to the DVB-S2 standard [4]. In order for the system to perform, capacity must be fairly allocated so that no end user is favoured, unless it has paid for extra fixed capacity. Moreover, capacity requests must be accurate to maximize link utilization and minimize the unused capacity.

This study therefore focuses on two issues. The first is the fairness of the capacity allocation process. We propose a new allocation algorithm that treats terminal capacity requests in an intuitively fair way. The criterion for a terminal to receive priority in bandwidth allocation is how well its requests were satisfied during the previous allocation process. Terminals with small request satisfaction degree should receive priority at the next allocation. The objective is to assign each time the maximum possible amount of the requested capacity without having terminals that monopolize the channel. To evaluate the performance of the proposed scheme, it is compared with the static round robin (RR) scheme used in [5] in terms of fairness and queuing delay. The second issue addressed is the capacity request calculation technique, which is important for the allocation process to perform. Two request schemes found in the literature are evaluated for both the allocation algorithms.

The rest of the article is organized as follows: in Section 2, we describe the basic BSA network architecture along with the MF-TDMA (Multi Frequency–Time Division Multiple Access) scheme, the CF-DAMA protocol and the network Differentiated Services (DiffServ) mechanisms. In Section 3, we present the satellite terminal architecture that was employed for the simulation of the BSA network. Section 4 focuses on the capacity request calculation methods, while Section 5 is dedicated to the proposed DCA algorithm and its mechanisms. In Section 6, we discuss details regarding the simulation, such as the traffic models used and the traffic Scenarios simulated. In addition, simulation results for the performance metrics are illustrated and analysed. Finally, we state some general conclusions on the issues investigated as well as some ideas about future work in this field.

2. BSA SYSTEM

We consider the network architecture proposed in [5], which comprises a population of RCSTs, a communications satellite, a traffic Gateway (GW) and a Network Control Centre (NCC) situated on the ground, as show in Figure 1. Small local area networks (LANs) are connected to the RCSTs and generate IP traffic. Such small networks can be found in residential, corporate or university premises. The forward channel carries data from the GW to the RCSTs and

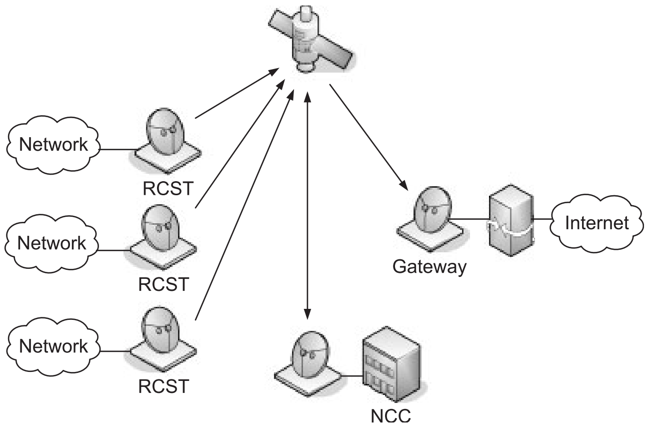

Figure 1. The DVB-RCS network architecture considered in this study.

DOI: 10.1002/sat

signalling information from the NCC to the RCSTs. The traffic from the RCSTs to the GW is transmitted through the return channel.

The DVB-RCS standard defines the MF-TDMA as the multiple access scheme. This combined-access scheme is a variant of TDMA that allows a group of RCSTs to communicate with a GW using a set of carrier frequencies. Capacity is organized in frames consisting of a number of time slots on a number of carriers. A number of consecutive frames make up a superframe, the duration of which equals to the capacity allocation period. Frames may be fixed or dynamic, i.e. their composition varies over time. Examples of fixed frame compositions are provided in the DVB-RCS guidelines document [6]. In this study, we consider the case of fixed frames with eight carriers and a duration of 26.5 ms. Each carrier is divided into 26 traffic slot durations, 24 of which are used for transmitting traffic (TRF) bursts that carry useful data, and the remaining 2 can be used for transmitting 4 synchronization (SYNC) bursts that carry control information. Each burst is surrounded by a guard time that allows for RCST power switch-off transient and system timing errors. Each TRF burst carries one ATM cell (53 bytes) resulting in a maximum transmission rate per carrier of 384 kbits/s and a total uplink capacity of 3072 kbits/s.

The CF-DAMA protocol specifies that return channel capacity is assigned to the RCSTs in response to capacity requests sent periodically in SYNC bursts. If all requests are satisfied and there is still capacity left, it is allocated freely among the RCSTs. The request–allocation process is subject to a delay called scheduling lag. The scheduling lag (Equation (1)) includes the propagation times on the forward link ($\mathrm{Tp}_{\mathrm{fwd}}$) and return link ($\mathrm{Tp}_{\mathrm{rtn}}$), the processing times in the RCSTs and the NCC as well as a latency introduced to ensure that all the RCSTs have properly received the Terminal Burst Time Plan (TBTP).

$$\text{Scheduling lag} = \mathrm{Tp}_{\mathrm{fwd}} + \mathrm{Tp}_{\mathrm{rtn}} + \text{Processing time} + \text{Safe frame period} \tag{1}$$

The TBTP message is the NCC response to the terminal requests. It contains the allocation plan of frame slots to the RCSTs for the next superframe and is periodically transmitted every superframe via the forward channel. The TBTP transmission is subject to the same propagation delay ($\mathrm{Tp}_{\mathrm{fwd}} = \mathrm{Tp}_{\mathrm{rtn}}$) experienced by capacity requests on their way to the NCC.

The DVB-RCS standard defines five capacity categories that can serve traffic with various requirements:

- *Continuous rate assignment* (CRA) is the rate capacity negotiated directly between the terminal and the NCC at the beginning of a connection and assigned in full throughout the entire connection.
- *Rate-based dynamic capacity* (RBDC) is the rate capacity assigned after explicit requests in terms of slots per frame. Each request overrides all the previous requests from the same terminal and is subject to a maximum guaranteed rate limit negotiated between the terminal and the NCC. In this study, the maximum number of RBDC slots is considered equal for all the RCSTs and calculated by Equation (2):

  $$\mathrm{RBDC}_{\max} = \left\lfloor \frac{n_{\mathrm{TRF}} - n_{\mathrm{CRA}}}{N} \right\rfloor \tag{2}$$

  where n_{TRF} is the number of traffic slots per frame, n_{CRA} the number of CRA slots per frame and N the number of terminals logged in the system. In case a terminal request exceeds the value of $\mathrm{RBDC}_{\max}$, it is assigned $\mathrm{RBDC}_{\max}$ slots, while the amount in excess is added to the volume-based dynamic capacity (VBDC) capacity required. RBDC requests are valid for a certain period of time. If this time elapses without any new request transmitted by the corresponding RCST, it is no longer assigned RBDC slots.
- *VBDC* is the volume capacity assigned after explicit requests, which are cumulative, meaning that new requests are added to the old ones. The cumulated capacity is decremented by the amount of VBDC slots assigned during each allocation process.
- *Absolute volume-based dynamic capacity* (AVBDC) is also volume capacity and differs from VBDC in that new AVBDC requests override the previous ones that come from the same terminal. This capacity category was not used in this study.

DOI: 10.1002/sat

Table I. DiffServ PHBs—DVB-RCS capacity categories mapping.

	CRA	RBDC	VBDC	FCA
DE			100%	
AF		70%	30%	
EF	80%	20%		

- *Free capacity assignment* (FCA) involves the remaining capacity after the satisfaction of all the terminal requests. No traffic should be exclusively mapped to FCA, as its availability is highly variable. Typically, it is bonus capacity granted to the terminals and can be used as supplement to improve traffic delay and jitter.

CRA and maximum RBDC capacity are guaranteed for every RCST. Thus, Relation (3) should be satisfied during every capacity allocation round:

$$\sum \mathrm{CRA} + \mathrm{RBDC}_{\max} \leqslant \mathrm{Total\_return\_channel\_capacity} \tag{3}$$

To provide QoS for IP traffic, the BSA network needs to be compatible with the DiffServ architecture. The DiffServ architecture [7] defines mechanisms that classify and mark packets belonging to a specific class of traffic. Packets are then forwarded according to priority policies called per-hop behaviours (PHBs) that define the packet forwarding properties associated with its class of traffic. There are three main PHBs:

- *Default PHB* (DE) which is typically best-effort traffic.
- *Expedited forwarding* (EF) PHB which is used for low-loss, low-latency and low-jitter traffic generated mainly by real-time services.
- *Assured forwarding* (AF) behaviour group which defines four separate AF classes. Within each class, packets are given a-drop precedence (high, medium or low). The AF PHB group is suitable for TCA (traffic conditioning agreement) compliant traffic with very small drop probability.

The DiffServ architecture is implemented in the BSA network by mapping the IP PHBs to DVB-RCS capacity categories. In this study, for simplicity reasons, a static mapping technique was implemented, according to which each DVB-RCS capacity category serves a certain percentage of each PHB traffic. Table I presents the mapping percentages used in this study.

EF PHB requires guaranteed bandwidth availability to support low-loss, low-latency and low-jitter traffic. To meet these requirements, it cannot be served entirely on a request basis undergoing the scheduling lag, hence it is mapped mainly to CRA. In this study, EF traffic was not used because fixed capacity assignment is out of the scope of the study. The AF PHB group is mapped to RBDC by 70% and VBDC by 30%, as it has no delay and jitter constraints. It is not fully mapped to RBDC because this capacity category provides a certain bandwidth guarantee, and this would lead to a very simplified system configuration. DE traffic is served totally by VBDC, which is the least reliable request-based capacity. FCA does not serve explicitly any PHB, although it can be a useful supplement in case of unexpected congestion. The mapping of PHBs to DVB-RCS capacity categories is performed by specific RCST mechanisms described in the following section.

3. RCST ARCHITECTURE

RCSTs are responsible for IP traffic classification, IP packet segmentation into ATM cells, cell classification and finally implementation of a specific queuing discipline. An RCST architecture for QoS provisioning is proposed in [5] and comprises six main blocks as shown in Figure 2:

- *Traffic selector* classifies IP packets into the appropriate IP queues.
- *IP queues* contain IP packets before they are segmented into ATM cells.

DOI: 10.1002/sat

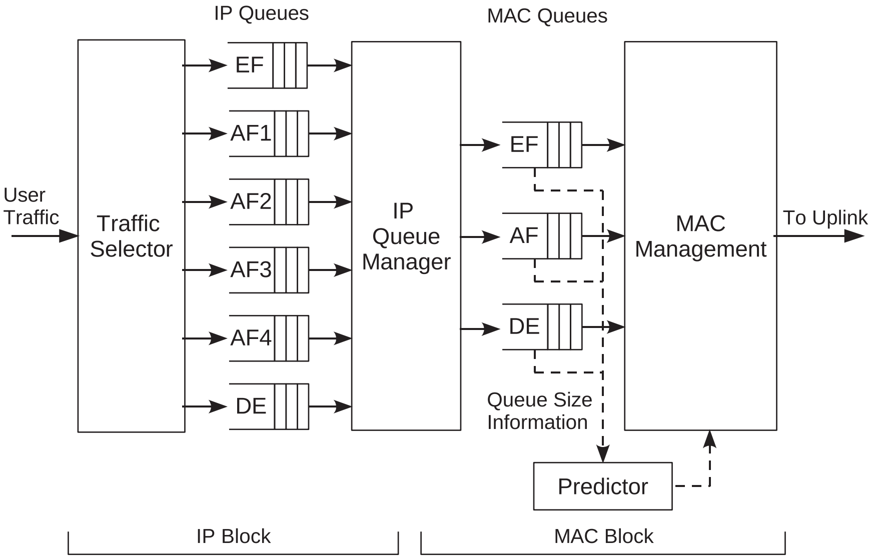

Figure 2. RCSTs' DiffServ architecture and DCA mechanisms.

- *IP queue manager* segments IP packets into fixed-sized ATM cells according to the AAL5 layer (ATM Adaptation Layer 5) [8], and classifies them into the appropriate medium access control (MAC) queue.
- *MAC queues* contain ATM cells prior to their transmission in the return link. In this study, only one MAC queue was used for the AF PHB group, as AF traffic was considered homogeneous with no further distinction in precedence levels.
- *MAC management* module computes capacity requests based on buffer size and employs a queuing policy to determine which MAC queue will transmit next. In this study, we used the priority queuing discipline, according to which higher priority queues transmit next, unless they are empty.
- *Predictor* estimates future traffic according to incoming rate and queue size measurements taken regularly. Prediction is necessary in request calculation to estimate the amount of traffic that will arrive from the time the request is sent until the requested capacity is allocated, i.e. during the scheduling lag.

The following section focuses on how capacity requests are calculated based on measurements of the MAC queue size and taking into account the QoS requirements of the traffic served by each queue.

4. CAPACITY REQUEST CALCULATION

Capacity requests issued by the terminals to the NCC need to be accurate to avoid requesting excess capacity and thus wasting network resources. They also need to be predictive so that the assigned capacity can serve the traffic received during the scheduling lag, as stated in the previous section. Henceforth, we describe the two capacity request mechanisms examined in this study.

4.1. Capacity requests based on mapping IP service classes to DVB-RCS capacity categories

In the first request scheme (RS1), capacity needed for each PHB is initially mapped to the DVB-RCS capacity categories. The amount of capacity needed for each category is then transmitted to the NCC. In RS1, only RBDC and VBDC capacity is assigned in response to terminal requests.

DOI: 10.1002/sat

According to Table I, RBDC capacity serves 70% of AF and 20% of EF traffic. The required AF capacity can be calculated by Equation (4) in slots per frame [9].

$$C_{\mathrm{AF}}(t) = R_{\mathrm{AF}}(t) + \frac{Q_{\mathrm{AF}}(t)}{T_s} \tag{4}$$

where R_{AF} is the average rate of incoming AF traffic, $Q_{\mathrm{AF}}(t)$ the current AF MAC queue size in cells and T_s a constant time period. The concept of Formula (4) is that the required AF capacity equals the current average incoming AF traffic rate (R_{AF}) plus the capacity needed to empty the AF buffer in time T_s. R_{AF} is calculated via a sliding window technique and equals the number of cell arrivals during the window period divided by the period itself. In this study, we used a sliding window with a length of 0.265 s (10 frames), which in [10] was considered the optimal length. The time period T_s is equal to the request submission period so that the term $Q_{\mathrm{AF}}(t)/T_s$ represents the transmission rate required to empty the AF buffer till the next request calculation. According to Table I, the requested RBDC slots per frame will be:

$$\mathrm{RBDC}_{\mathrm{req}} = (0.7C_{\mathrm{AF}} + 0.2C_{\mathrm{EF}}) \cdot \mathrm{Frame\_duration} \tag{5}$$

VBDC requests serve 30% of AF traffic and all DE traffic. The required VBDC slots for the next allocation period are calculated by the following formula:

$$\mathrm{VBDC}_{\mathrm{req}} = \begin{cases} Q_{\mathrm{DE}}(n) - Q_{\mathrm{DE}}(n-1) + 0.3C_{\mathrm{AF}} \cdot \mathrm{AP} & \text{if } Q_{\mathrm{DE}}(n) > Q_{\mathrm{DE}}(n-1) \\ 0.3C_{\mathrm{AF}} \cdot \mathrm{AP} & \text{if } Q_{\mathrm{DE}}(n) \leqslant Q_{\mathrm{DE}}(n-1) \end{cases} \tag{6}$$

where $Q_{\mathrm{DE}}(n)$ is the current size of the DE MAC queue, $Q_{\mathrm{DE}}(n-1)$ the DE MAC queue size at the previous request calculation and AP the capacity allocation period. Typically, the VBDC slots required by an RCST are equal to the increase in the DE buffer size since the last request, plus 30% of the required AF rate. AF rate capacity (in slots/s) is transformed into volume capacity (slots) by multiplying C_{AF} with AP. This product yields the number of slots that need to be assigned during the next AP to reach the required AF rate. The NCC keeps a counter of the VBDC slots requested by each terminal. Each time a VBDC slot is assigned, the counter is decremented, whereas each time a new VBDC request is received, it is incremented by the amount of slots requested. Apart from RBDC and VBDC, one CRA slot per frame is assigned to each terminal to ensure at least one transmission during a frame.

4.2. Capacity requests directly for each IP service class

In the second request scheme (RS2), terminals explicitly request capacity for each IP service class (in our case AF and DE) rather than mapping the traffic to the DVB-RCS capacity categories. These requests are based on incoming rate and queue size measurements, along with the QoS requirements of the traffic each queue serves. The amount of the requested capacity ensures that the probability of the queue length Q exceeding a specified threshold n ($\Pr\{Q>n\}$) remains under a certain value. This value is called outage or QoS violation probability (P_{out}) and is determined by the QoS requirements of the corresponding traffic.

As MMPP (Markov-modulated Poisson process) and PMPP (Pareto-modulated Poisson process) traffic sources are used in the simulations (Section 6.1), each RCST forms an MMPP/G/1 or PMPP/G/1 queuing system, respectively. Assuming that each source remains in the same state during the request–allocation cycle, the system can be considered as M/G/1 [10]. Moreover, if a constant queue service rate is assumed, the M/G/1 becomes M/D/1. In this case, the requested rate capacity for each queue is given by the following equation:

$$C = \lambda \frac{e^{\gamma} - 1}{\gamma} \tag{7}$$

where

$$\gamma = -\frac{\ln(\Pr\{Q>n\})}{n+1}$$

Q is the queue size and λ the average incoming traffic rate in slots/s, calculated using the sliding window technique described in Section 4.1. According to [10], the outage probability is set to

DOI: 10.1002/sat

10^{-3}, and the threshold buffer sizes are $n_{\mathrm{AF}} = 115$ cells and $n_{\mathrm{DE}} = 150$ cells. Apart from requesting capacity for each service class, RS2 does not specify any fixed capacity assignment.

5. CAPACITY ALLOCATION

5.1. RCST data list and its update algorithm

The capacity allocation procedure, which is performed in the NCC, comprises two steps: first, the capacity requests that arrived at the NCC during the previous allocation period are processed. Then the NCC assigns time slots on the available carriers to the terminals according to the amount of slots they requested.

To fulfil the first step, the NCC keeps an incoming request queue where new capacity requests are pushed. It also keeps a record of each terminal logged in the system in a list called RCST data list. Each list entry contains several fields depending on the request scheme used. For the RS1 scheme, an RCST entry may contain the following fields:

- Terminal id
- Request satisfaction ratio (RSR), Equation (8)
- CRA slots per frame
- RBDC slots per frame requested
- Cumulated VBDC slots requested
- RBDC request expiration timer
- Maximum RBDC slots
- RBDC request validity period

For the RS2 scheme, an RCST entry should contain at least the following information:

- Terminal id
- RSR, Equation (9)
- EF slots requested
- AF slots requested
- DE slots requested

During the first step of the allocation procedure, an algorithm checks all the new requests that have arrived in the request queue and accordingly updates the RCST data list. This algorithm is different for each request scheme. The algorithm pseudocode [5] adapted for the RS1 request scheme is illustrated bellow:

```
For each request in the incoming request queue
      Get the RCST that sent this request

      if(there is no entry for this RCST in the RCST data list)
            Create a new entry in the list for this RCST and initialize it
      else
            Search through the RCST data list and find this RCST's entry

      If(RBDC slots requested)
            Reset the RBDC request expiration timer for this RCST to the RBDC
            request validity period
            if(RBDC request amount $\leqslant$ Maximum RBDC)
                  Overwrite the previous RBDC request amount for this RCST with
                  the new amount
            else
                  Overwrite the previous RBDC request amount for this RCST with
                  the Maximum RBDC
                  Add the portion of RBDC request amount in excess of the Maximum
                  RBDC to the cumulated VBDC request amount
```

DOI: 10.1002/sat

```
      If(VBDC slots requested)
            Add the VBDC request amount to the cumulated VBDC request amount
```

For the RS2 request scheme, the algorithm simply updates the values of the requested EF, AF and DE capacities for each RCST entry:

```
For each request in the incoming request queue
       Get the RCST that sent this request

       if(there is no entry for this RCST in the RCST data list)
            Create a new entry in the list for this RCST and initialize it
       else
            Search through the RCST data list and find this RCST's entry

       if(EF slots requested)
            Update EF slots requested by this RCST
       if (AF slots requested)
            Update AF slots requested by this RCST
       if (DE slots requested)
            Update DE slots requested by this RCST
```

5.2. Slot allocation algorithm

The second step of the allocation procedure involves the algorithm that assigns frame slots to the terminals according to the updated RCST data list. Iuoras *et al.* [5] used an RR allocation scheme wherein slots are assigned consecutively to each RCST entry. The first RCSTs to receive slots tend to have their requests satisfied better than those that receive capacity in the end and get the remaining slots. To eliminate this event, we propose a new algorithm which requires that, prior to slot assignment, the RCST data list would be sorted according to a value which shows how well each terminal requests were satisfied during the previous allocation. This value is called RSR and can be defined in various ways, depending on the class of traffic for which the algorithm is desired to perform fairer. In our simulations, the RSR is computed by Equations (8) and (9) for the request schemes RS1 and RS2, respectively.

$$\mathrm{RSR}_{\mathrm{RS1}} = \frac{\text{VBDC slots assigned}}{\text{Total slots requested}} \tag{8}$$

$$\mathrm{RSR}_{\mathrm{RS2}} = \frac{\text{DE slots assigned}}{\text{Total slots requested}} \tag{9}$$

The numerators in Equations (8) and (9) are the number of VBDC and DE slots assigned, whereas the denominators represent the sum of all slots requested by the terminal. VBDC and DE capacity was chosen to affect mostly the value of RSR, because it is the capacity of interest that receives the least priority. The RCST data list is sorted each time in ascending RSR order. In this way, the RCSTs with large RSR values in the previous slot allocation become the last ones to obtain capacity at the new allocation round. As a result, the occurrence of terminals, which constantly request large amounts of slots, being favoured is diminished. Thus, terminals are treated in a more balanced way. The RSR field of each RCST data list entry is updated after each capacity assignment so that the list would be properly sorted for the next allocation process. The proposed algorithm pseudocode for both the request schemes is as follows.

```
Sort RCST data list in ascending RSR order

// CRA and RBDC/AF capacity assignment
For each RCST data list entry
      Go to the carrier with the largest number of free slots
      Assign to the corresponding RCST the amount of CRA and RBDC/AF slots
      defined in this entry

// VBDC/DE capacity assignment
For each RCST data list entry
      If (the RCST corresponding to the current entry
      has already received slots)
            Go to the carrier containing these slots
      Else
            Go to the carrier with the largest number of free slots

      Assign to the corresponding RCST the amount of VBDC/DE slots defined
      in this entry up to the amount requested or up to the end of the carrier

For each RCST data list entry
      Update RSR
```

The same algorithm, yet without sorting the RCST data list, is used for the RR scheme. Henceforth, the proposed algorithm will be referred to as Fair Slot Allocation (FSA).

6. PERFORMANCE EVALUATION

6.1. Simulation setup

To evaluate the proposed allocation scheme, we implemented a simulator of the BSA network shown in Figure 1 in accordance with the DVB-RCS standard [2, 6]. We consider 16 RCSTs and one NCC situated on the ground. A capacity request needs 260 ms [11] to reach the NCC, whereas another 260 ms is needed by the TBTP message to travel back to the RCSTs, resulting in a 520-ms scheduling lag due to propagation delay. The scheduling lag also includes the safe frame period at the RCST, which is required to ensure that all the RCSTs have received the TBTP on time. According to [2], the safe frame period shall not exceed 90 ms. In this study, we consider a scheduling lag equal to 800 ms that includes 520 ms of round-trip propagation delay plus 280 ms of processing times and safe frame time. Capacity allocation is performed every 26.5 ms (frame duration), whereas capacity requests are transmitted every 800 ms. The capacity request period was chosen equal to the scheduling lag so that each request reserves capacity until the next request takes effect. RCST signalling information and capacity request messages are transmitted in the SYNC slots at the beginning of each frame. The duration of each simulation was set to 20 000 s; since during this period, it was observed that the system reaches a steady state combined with reasonable computing times.

Aggregate traffic in each queue is modelled by MMPP [12] and PMPP [13, 14] sources. MMPP sources generate short-range-dependent (SRD) traffic, while PMPP sources generate long-range-dependent (LRD) and self-similar traffic [15]. Aggregate traffic in both the cases is modelled as a switched Poisson process alternating between the two states with rate parameters λ_1 and λ_2. The sojourn times of these states are exponentially distributed in case of MMPP and Pareto distributed in case of PMPP. The packet arrival process during each state is Poisson with arrival rates λ_1 and λ_2, respectively. For the MMPP model, the state sojourn times are computed by an exponentially distributed random number generator with parameter $\lambda_1 = 1/T_i$, where T_i is the average duration of state *i*.

In case of PMPP sources, the Pareto distribution is used for the computation of the state intervals. The Pareto distribution is described by two parameters: Shape parameter *a*, which defines the shape of the distribution and scale parameter *b*, which affects the magnitude of the numbers generated. For LRD traffic generation, *a* should be bounded on (1,2) [13]. In this study, the shape parameter was set to 1.5, a typical value for rather bursty traffic, while the scale parameter is determined by the average state duration desired. This is achieved as follows: The

Pareto distribution has a mean of $b(a/a-1)$; hence, if the desired average sojourn time is T, the scale parameter is computed by the equation $b = T(a-1/a)$.

Table II shows the traffic Scenarios simulated along with the rate parameters λ_1 and λ_2 used to produce the corresponding load and AF/DE traffic mix. The traffic Scenarios were performed for slow switching (SS) traffic with average state sojourn times $T_1 = T_2 = 3.25\,\mathrm{s}$ and fast switching (FS) traffic with $T_1 = T_2 = 0.5\,\mathrm{s}$, using the same rate parameters in both the cases.

Table II. Simulation scenarios and state rates for slow and fast switching traffic.

Scenario	Load (%)	AF (%)	DE (%)	AF λ_1 (slots/s)	AF λ_2 (slots/s)	DE λ_1 (slots/s)	DE λ_2 (slots/s)
1	90	50	50	259.4	148.1	259.4	148.1
2	90	20	80	104.1	59.0	424.5	227.5
3	90	80	20	424.5	227.5	104.1	59.0
4	75	50	50	224.1	115.6	224.1	115.6
5	75	20	80	78.1	57.8	342.0	201.4
6	75	80	20	342.0	201.4	78.1	57.8

Table III. Fairness index for SS traffic.

					Fairness index			
					RS1		RS2	
Scenario	Model	Load (%)	AF (%)	DE (%)	FSA	RR	FSA	RR
1	MMPP	90	50	50	0.404574	0.349441	0.960757	0.960338
2	MMPP	90	20	80	0.836493	0.80037	0.957286	0.950236
3	MMPP	90	80	20	9.69×10^{-5}	9.76×10^{-5}	0.976448	0.971912
4	PMPP	90	50	50	0.37839	0.335836	0.961602	0.961104
5	PMPP	90	20	80	0.574728	0.683395	0.95679	0.949724
6	PMPP	90	80	20	10.08×10^{-5}	9.79×10^{-5}	0.978372	0.973748
7	MMPP	75	50	50	0.875553	0.829879	0.984749	0.981823
8	MMPP	75	20	80	0.911335	0.901719	0.985353	0.978233
9	MMPP	75	80	20	0.0496947	0.0482574	0.993457	0.989452
10	PMPP	75	50	50	0.870645	0.824181	0.984507	0.981644
11	PMPP	75	20	80	0.912277	0.903189	0.985771	0.978817
12	PMPP	75	80	20	0.0735685	0.0943659	0.993918	0.989968

Table IV. Fairness index for FS traffic.

					Fairness index			
					RS1		RS2	
Scenario	Model	Load (%)	AF (%)	DE (%)	FSA	RR	FSA	RR
1	MMPP	90	50	50	0.575177	0.505914	0.965627	0.964444
2	MMPP	90	20	80	0.83892	0.809427	0.961643	0.954434
3	MMPP	90	80	20	1.258×10^{-4}	1.197×10^{-4}	0.978853	0.974815
4	PMPP	90	50	50	0.632034	0.553329	0.965889	0.964538
5	PMPP	90	20	80	0.791252	0.781069	0.960129	0.952847
6	PMPP	90	80	20	0.004536	0.004415	0.980136	0.975994
7	MMPP	75	50	50	0.895138	0.86132	0.988397	0.98529
8	MMPP	75	20	80	0.911744	0.903368	0.987347	0.980687
9	MMPP	75	80	20	0.0999238	0.095506	0.99426	0.990756
10	PMPP	75	50	50	0.897051	0.864173	0.988135	0.984992
11	PMPP	75	20	80	0.912279	0.904182	0.987518	0.980913
12	PMPP	75	80	20	0.25615	0.247668	0.99436	0.990885

DOI: 10.1002/sat

Each Scenario was performed twice for each allocation algorithm (FSA and RR), once for RS1 and once for RS2. Simulation results are discussed in the following section.

6.2. Discussion of simulation results

The simulator implemented computes the queuing delay experienced by each packet in the terminal buffers. The fairness of the allocation process is measured using Jain's Fairness Index (FI) [16]:

$$f(x) = \frac{\left[\sum_i x_i\right]^2}{n \sum_i x_i^2} \tag{10}$$

where n is the number of samples checked for fairness.

In our simulations, after each capacity allocation i, the number of total slots assigned to a terminal j ($\mathrm{ASN}_{i,j}$) divided by the total requested number of slots ($\mathrm{REQ}_{i,j}$) is recorded. This value is called Request Satisfaction Index, $\mathrm{RSI}_{i,j}$ and is calculated by Equation (11). $\mathrm{ASN}_{i,j}$ does not comprise the FCA slots assigned to the terminal, thus $0 \leqslant \mathrm{RSI}_{i,j} \leqslant 1$.

$$\mathrm{RSI}_{i,j} = \begin{cases} \frac{\mathrm{ASN}_{i,j}}{\mathrm{REQ}_{i,j}} & \text{if } \mathrm{REQ}_{i,j} \neq 0 \\ 1 & \text{if } \mathrm{REQ}_{i,j} = 0 \end{cases} \tag{11}$$

By substituting x_i with $\mathrm{RSI}_{i,j}$, Equation (10) becomes:

$$f(\mathrm{RSI}) = \frac{\left[\sum_i \sum_j \mathrm{RSI}_{i,j}\right]^2}{N_{\mathrm{a}} N_{\mathrm{RCST}} \sum_i \sum_j \mathrm{RSI}_{i,j}^2} \tag{12}$$

where N_{a} is the number of capacity allocations performed during the simulation and N_{RCST} the number of active terminals.

We consider that capacity allocation is fair, if the amount of slots assigned per slots requested (RSI) for each RCST is close to one another resulting in a FI (Equation (12)) close to 1. Values of the index computed in the simulations are greater for the proposed FSA than the RR scheme in all except for three simulation Scenarios. More specifically, as shown in Table IV, the FSA yielded greater FI in all the simulations with smooth FS traffic for both request schemes. For the more demanding SS traffic and RS1 requests, the proposed allocation scheme performed better in terms of fairness in almost all the Scenarios apart from 3, 5 and 12 (Table III). As far as the capacity request calculation methods are concerned, the RS2 scheme generally yielded greater and more uniform FI values. The fairness gap between the two request schemes is more obvious for 80% AF–20% DE traffic mix. Under these circumstances, most of the capacity is allocated to the high-priority AF traffic, leaving a small amount of slots for DE traffic. In periods with light DE traffic, terminals are more or less assigned the requested VBDC/DE slots, resulting in reasonable values of RSI close to 1 (Equation (11)). However, in periods with heavy DE traffic, the cumulated VBDC/DE slots increase dramatically causing the RSI to plummet. This large fluctuation of the RSI results in very small FIs calculated in such Scenarios. On the contrary, the non-cumulative-rate-based requests of RS2 result in more uniform values of RSI, thus achieving a FI closer to unity. Consequently, in terms of FI, the capacity allocation algorithm becomes more effective using the RS2 scheme than using RS1. Moreover, considering the number of Scenarios in which FSA achieved greater FI, it can be concluded that FSA is more fair than RR.

Apart from the FI, we computed the queuing delay experienced by packets in the terminal buffers. Queuing delay is illustrated primarily in the form of tables containing the average AF and DE packet delay calculated in each Scenario. For each average packet delay given in Tables V–VIII, the equivalent 95% confidence interval is computed and shown in Figures 3–6. The confidence intervals are very small in most cases because of the large sample size used to estimate the average packet delays. In Figures 7–22, packet queuing delay is also presented in the form of survival function curves that depict the probability of queuing delay exceeding a certain amount ($R(d) = \Pr\{Delay > d\}$). Each figure illustrates AF or DE delay computed using the same request scheme, traffic burstiness (MMPP or PMPP), traffic state duration (FS or SS) and

DOI: 10.1002/sat

Table V. Average AF packet queuing delay in seconds for SS traffic.

					AF delay (s)			
					RS1		RS2	
Scenario	Model	Load (%)	AF (%)	DE (%)	FSA	RR	FSA	RR
1	MMPP	90	50	50	0.0621568	0.0636954	0.0698701	0.0700347
2	MMPP	90	20	80	0.0406758	0.0432653	0.0442215	0.0448145
3	MMPP	90	80	20	0.161652	0.165232	0.171073	0.171815
4	PMPP	90	50	50	0.0644102	0.0657732	0.0745263	0.0743345
5	PMPP	90	20	80	0.041006	0.0449141	0.0442934	0.0449088
6	PMPP	90	80	20	0.168798	0.171638	0.180985	0.180387
7	MMPP	75	50	50	0.0559638	0.0576826	0.061988	0.0613757
8	MMPP	75	20	80	0.0381604	0.0398369	0.0421233	0.0416173
9	MMPP	75	80	20	0.086424	0.0871381	0.102164	0.102109
10	PMPP	75	50	50	0.0583803	0.0601071	0.0651996	0.0642761
11	PMPP	75	20	80	0.0381416	0.0398188	0.0421452	0.0415941
12	PMPP	75	80	20	0.0889705	0.0902187	0.109244	0.108721

Table VI. Average AF packet queuing delay in seconds for FS traffic.

					AF delay (s)			
					RS1		RS2	
Scenario	Model	Load (%)	AF (%)	DE (%)	FSA	RR	FSA	RR
1	MMPP	90	50	50	0.0696254	0.0700454	0.0760077	0.075626
2	MMPP	90	20	80	0.0407015	0.0431827	0.0436575	0.0440788
3	MMPP	90	80	20	0.154563	0.155492	0.186044	0.184232
4	PMPP	90	50	50	0.0629674	0.0640221	0.0680238	0.068157
5	PMPP	90	20	80	0.041184	0.0433937	0.043531	0.0440734
6	PMPP	90	80	20	0.123863	0.125446	0.144229	0.143765
7	MMPP	75	50	50	0.0599448	0.0614694	0.0673545	0.0662492
8	MMPP	75	20	80	0.038154	0.0398114	0.0417437	0.0412449
9	MMPP	75	80	20	0.0925793	0.0936361	0.11473	0.114135
10	PMPP	75	50	50	0.055804	0.0574285	0.0619395	0.0612073
11	PMPP	75	20	80	0.0381127	0.0397573	0.04161	0.0411051
12	PMPP	75	80	20	0.0863011	0.0877967	0.096777	0.0966009

Table VII. Average DE packet queuing delay in seconds for SS traffic.

					DE delay (s)			
					RS1		RS2	
Scenario	Model	Load (%)	AF (%)	DE (%)	FSA	RR	FSA	RR
1	MMPP	90	50	50	1.29049	1.35589	0.87663	0.877931
2	MMPP	90	20	80	2.67716	2.29455	0.712725	0.903024
3	MMPP	90	80	20	1.66464	1.74403	1.14403	1.24204
4	PMPP	90	50	50	6.52442	7.30912	2.7639	2.60387
5	PMPP	90	20	80	13.8319	11.8643	1.97366	2.76786
6	PMPP	90	80	20	4.67244	4.98677	0.988249	1.09808
7	MMPP	75	50	50	0.325161	0.327867	0.294123	0.307097
8	MMPP	75	20	80	0.333029	0.3232	0.207659	0.239094
9	MMPP	75	80	20	0.36983	0.374738	0.394901	0.408362
10	PMPP	75	50	50	0.718662	0.603924	0.306406	0.318805
11	PMPP	75	20	80	0.477051	0.440568	0.218599	0.244344
12	PMPP	75	80	20	0.36058	0.369221	0.391445	0.402444

Table VIII. Average DE packet queuing delay in seconds for FS traffic.

					DE delay (s)			
					RS1		RS2	
Scenario	Model	Load (%)	AF (%)	DE (%)	FSA	RR	FSA	RR
1	MMPP	90	50	50	0.568602	0.572447	0.51622	0.538977
2	MMPP	90	20	80	0.607996	0.557506	0.417393	0.486499
3	MMPP	90	80	20	0.862561	0.88049	0.864173	0.908303
4	PMPP	90	50	50	1.8533	1.8292	0.857113	0.912992
5	PMPP	90	20	80	7.67664	5.10562	0.674323	1.04967
6	PMPP	90	80	20	4.21898	4.07357	0.771498	0.846953
7	MMPP	75	50	50	0.267032	0.270206	0.282332	0.288641
8	MMPP	75	20	80	0.196243	0.199611	0.211167	0.221328
9	MMPP	75	80	20	0.343635	0.349088	0.392324	0.402789
10	PMPP	75	50	50	0.278378	0.265823	0.254407	0.262206
11	PMPP	75	20	80	0.28122	0.265576	0.186012	0.201059
12	PMPP	75	80	20	0.333285	0.336318	0.350548	0.361265

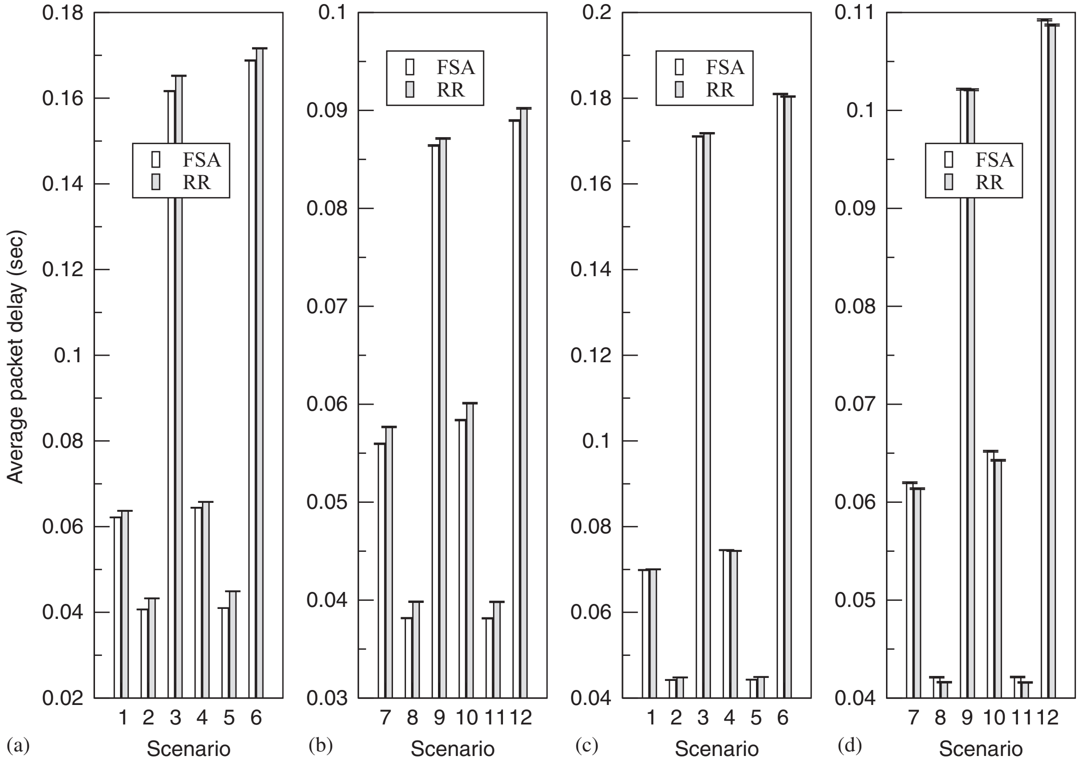

Figure 3. Estimated average AF packet queuing delays for SS traffic (Table V) along with their 95% confidence intervals: (a) RS1 90% load; (b) RS1 75% load; (c) RS2 90% load; and (d) RS2 75% load.

load. There are three pairs of curves in each figure, one for each AF-DE traffic mix examined (50–50, 20–80 or 80–20). In each pair, one curve corresponds to FSA and the other to RR.

First, we examine packet delay for the FSA and RR algorithms using the RS1 request scheme. Results in Tables V and VI show that the FSA algorithm achieves slightly smaller AF packet delays in all the Scenarios for both SS and FS traffic, which is also confirmed by Figures 7, 9, 11 and 13. On average, AF delays are improved by 3.29% for SS and 2.62% for FS traffic compared with RR. For the same request scheme, the average DE packet delays computed are enhanced with the FSA

DOI: 10.1002/sat

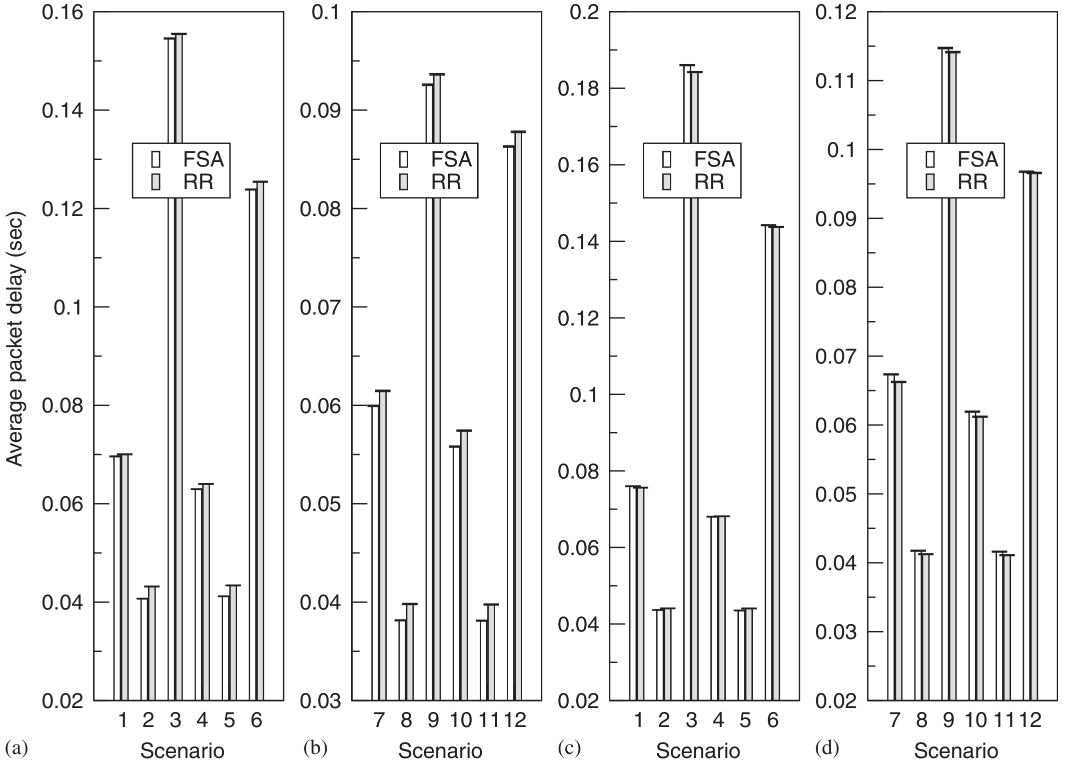

Figure 4. Estimated average AF packet queuing delays for FS traffic (Table VI) along with their 95% confidence intervals: (a) RS1 90% load; (b) RS1 75% load; (c) RS2 90% load; and (d) RS2 75% load.

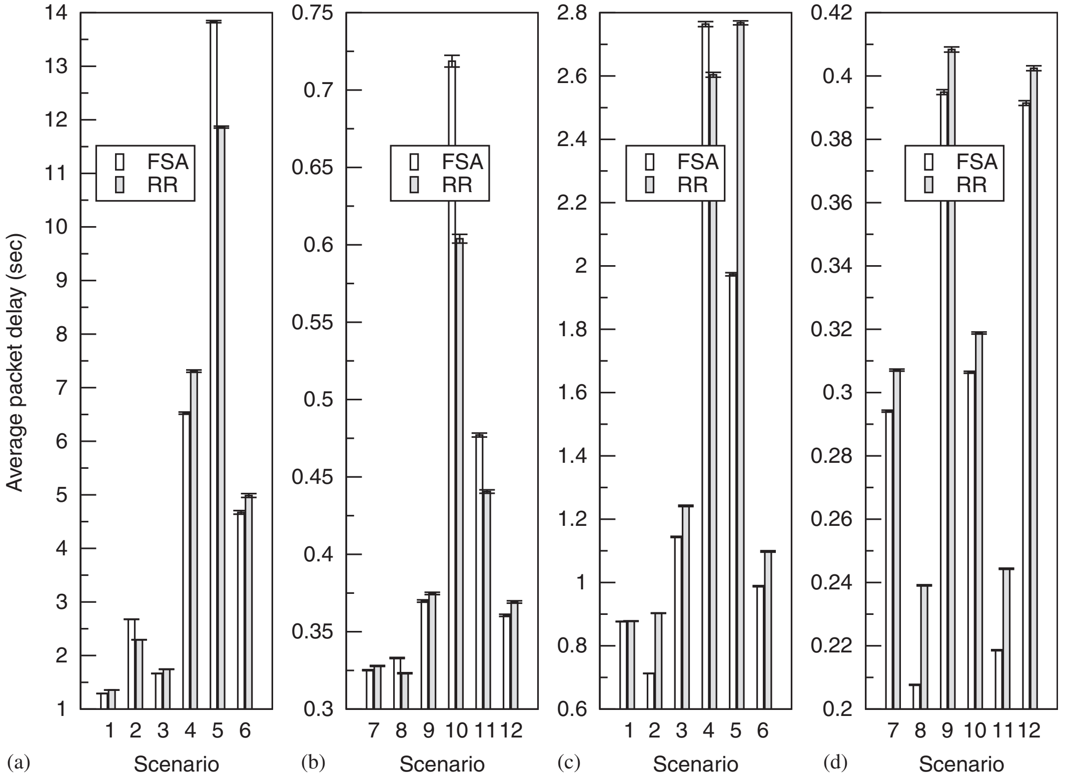

Figure 5. Estimated average DE packet queuing delays for SS traffic (Table VII) along with their 95% confidence intervals: (a) RS1 90% load; (b) RS1 75% load; (c) RS2 90% load; and (d) RS2 75% load.

DOI: 10.1002/sat

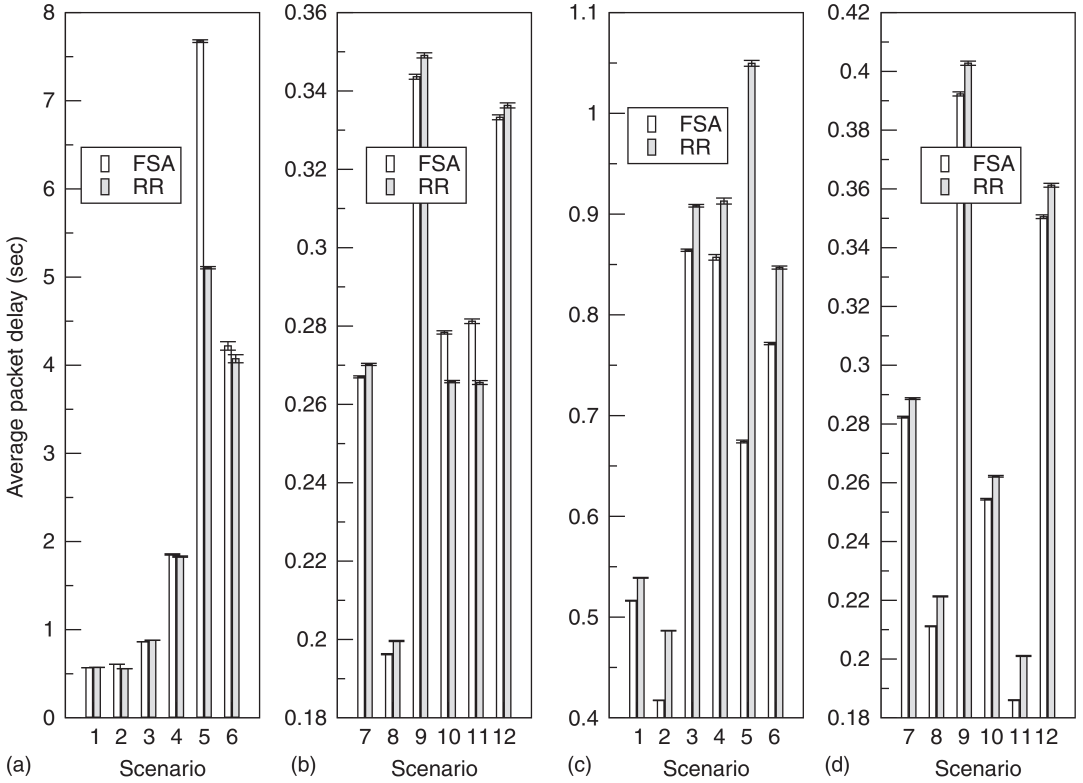

Figure 6. Estimated average DE packet queuing delays for FS traffic (Table VIII) along with their 95% confidence intervals: (a) RS1 90% load; (b) RS1 75% load; (c) RS2 90% load; and (d) RS2 75% load.

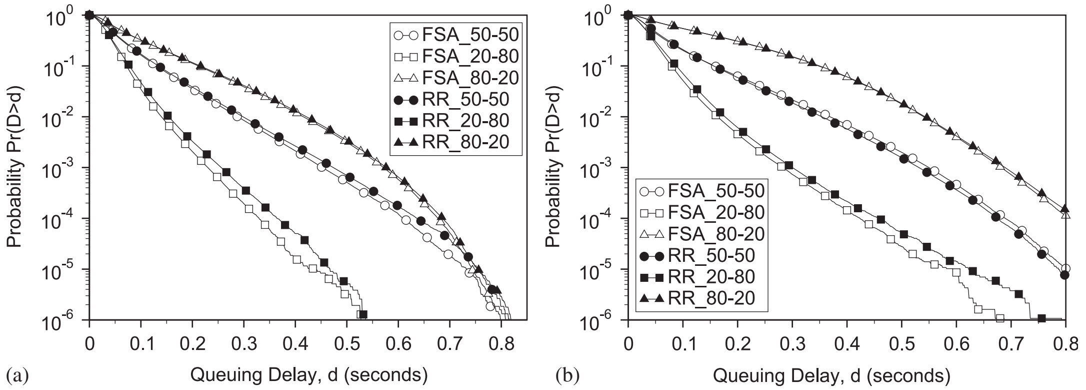

Figure 7. AF traffic, RS1 request scheme, MMPP FS sources: (a) 75% load and (b) 90% load.

scheme in almost half the Scenarios. Specifically, for SS traffic (Table VII, Figures 5(a, b)), the proposed scheme gave on average 4.41% better DE delay in 90% load Scenarios 1, 3, 4, 6 and 75% load Scenarios 7, 9, 12. For smoother FS traffic (Table VIII, Figures 6(a,b)), FSA performs better in 90% load Scenarios 1, 3 and 75% load Scenarios 7, 8, 9 and 12 by an average of 1.34%. A closer inspection of the FS traffic results shows that the combination of FSA and RS1 provides better DE performance in Scenarios with 80% AF traffic except for Scenario 6 (PMPP, FS, 90% load). In simulations with 50% AF–50% DE mix and FS traffic, FSA yielded smaller DE delays with MMPP sources (Scenarios 1, 7), whereas the performance deteriorated with the more demanding PMPP sources (Scenarios 4, 10). For the same traffic mix but with SS traffic, FSA performed better in 90% load Scenarios 1 and 4, while coming second in 75% load Scenarios 7 and 10. For 20%

DOI: 10.1002/sat

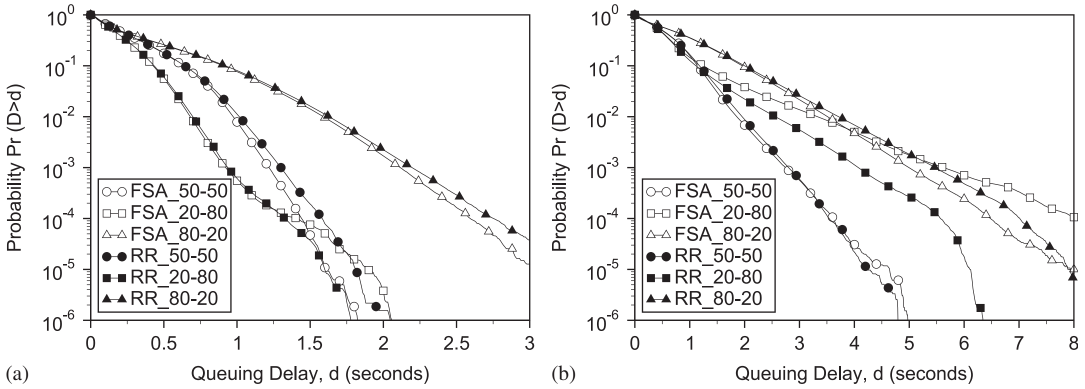

Figure 8. DE traffic, RS1 request scheme, MMPP FS sources: (a) 75% load and (b) 90% load.

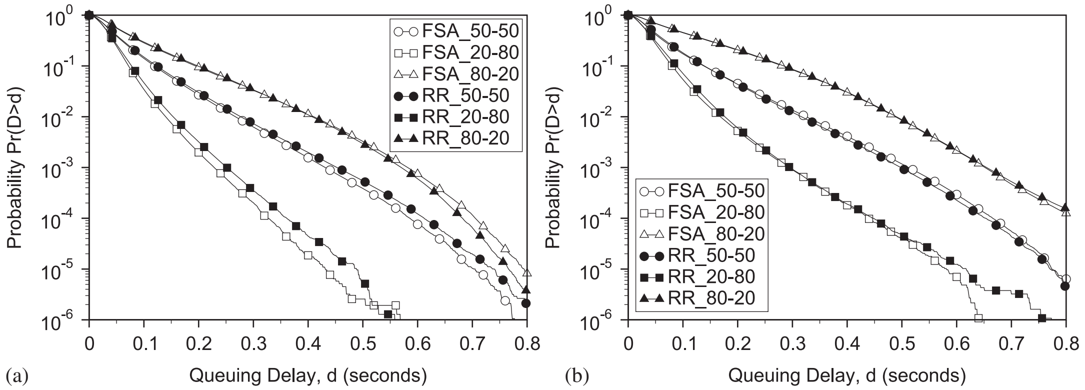

Figure 9. AF traffic, RS1 request scheme, PMPP FS sources: (a) 75% load and (b) 90% load.

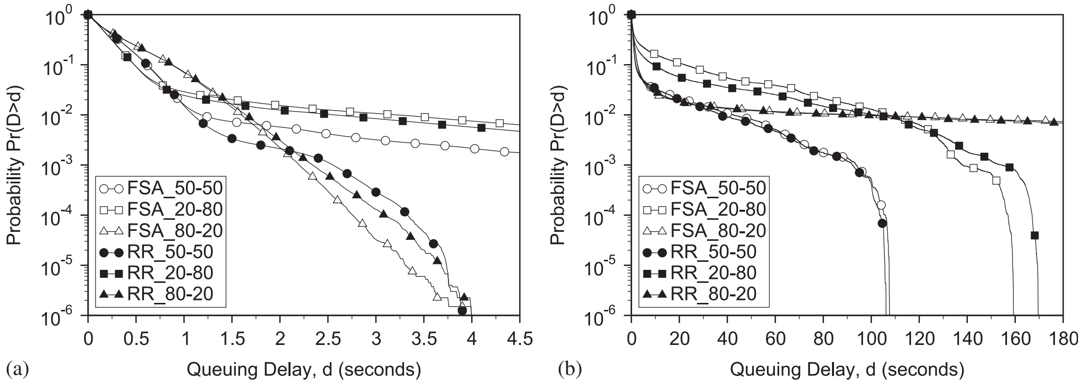

Figure 10. DE traffic, RS1 request scheme, PMPP FS sources: (a) 75% load and (b) 90% load.

AF–80% DE mix, FSA was inferior in all the SS Scenarios (2, 5, 8 and 11) and all but one (6, MMPP, 75% load) FS Scenarios. The same remarks can be carried out examining Figures 12 and 14 for SS traffic and Figures 8 and 10 for FS traffic. In general, using the RS1 capacity requests, FSA provides better AF performance than RR, whereas in terms of DE traffic, its performance worsens as DE traffic gradually increases.

Simulations of the two allocation schemes with RS2 capacity requests show that, contrary to RS1, FSA provides better DE performance while being slightly inferior in terms of AF delay. The

DOI: 10.1002/sat

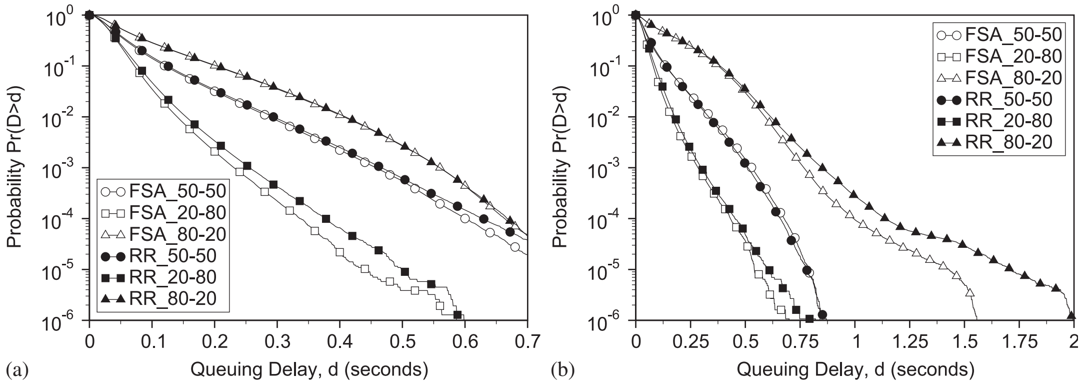

Figure 11. AF traffic, RS1 request scheme, MMPP SS sources: (a) 75% load and (b) 90% load.

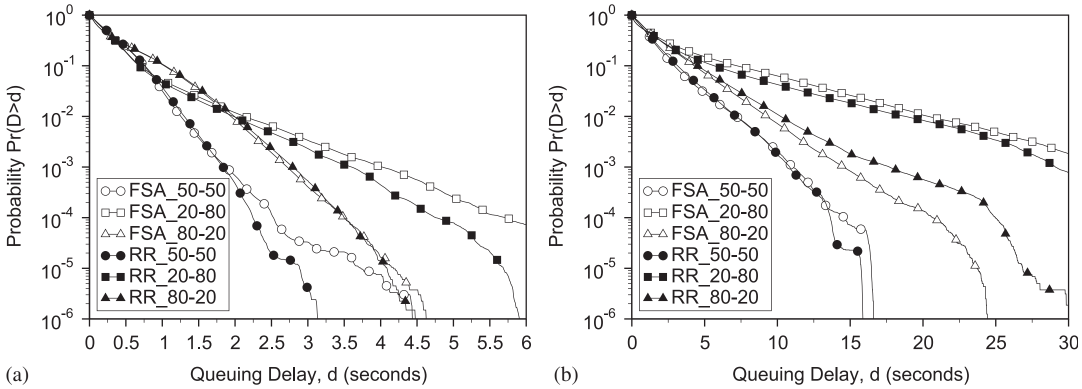

Figure 12. DE traffic, RS1 request scheme, MMPP SS sources: (a) 75% load and (b) 90% load.

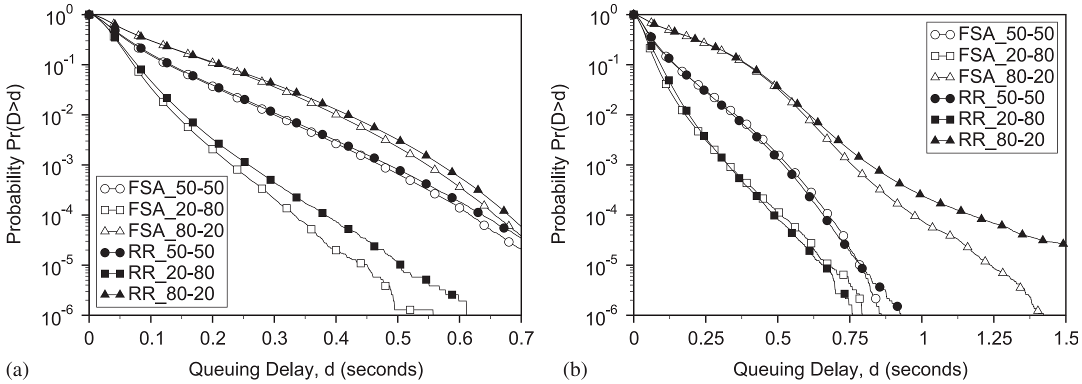

Figure 13. AF traffic, RS1 request scheme, PMPP SS sources: (a) 75% load and (b) 90% load.

respective simulation results concerning DE traffic are provided in Tables VII and VIII as well as in Figures 16, 18, 20 and 22. Packet delays computed for the FSA scheme indicate an average of 9.6% decrease in DE delay for SS and 8.07% for FS traffic with respect to RR except for one case. This case is Scenario 4 with 90% load, 50% AF–50% DE mix and SS traffic generated by PMPP sources (Table VII, Figure 22(b)). In terms of AF delay, the proposed allocation scheme performed slightly better in 90% load SS and SRD traffic Scenarios 1, 2, 3 and LRD traffic Scenario 5 (Table V, Figures 19(b) and 21(b)). In the remaining Scenarios, FSA AF performance was inferior

DOI: 10.1002/sat

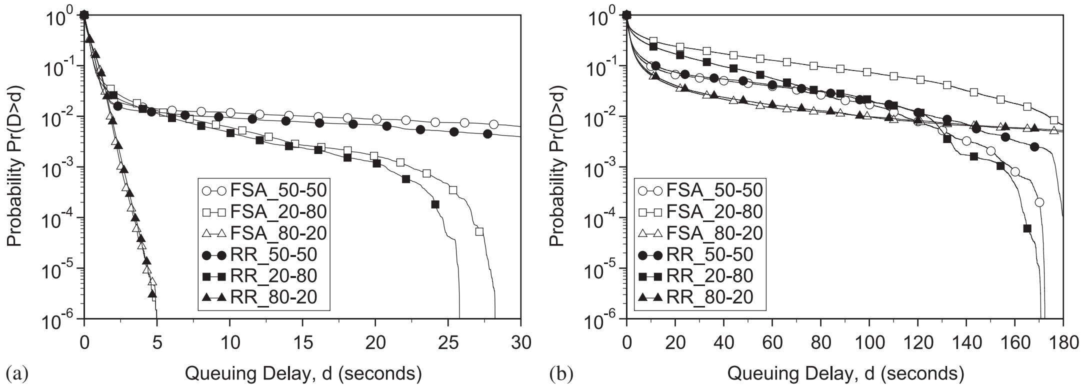

Figure 14. DE traffic, RS1 request scheme, PMPP SS sources: (a) 75% load and (b) 90% load.

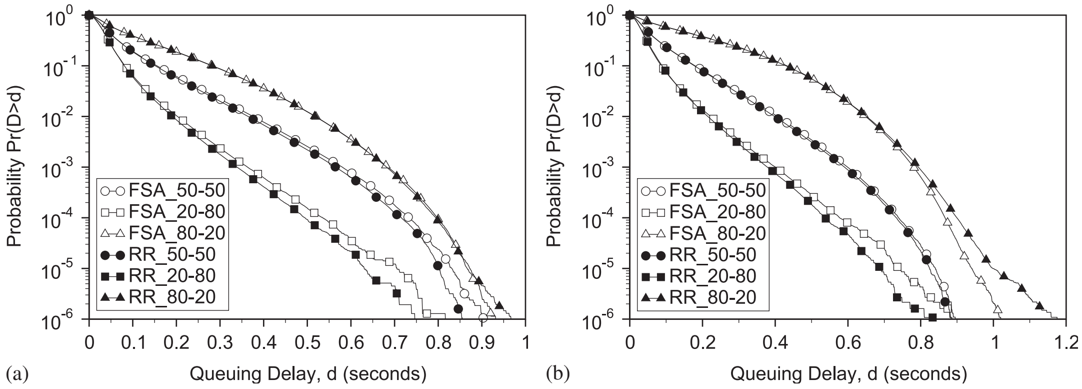

Figure 15. AF traffic, RS2 request scheme, MMPP FS sources: (a) 75% load and (b) 90% load.

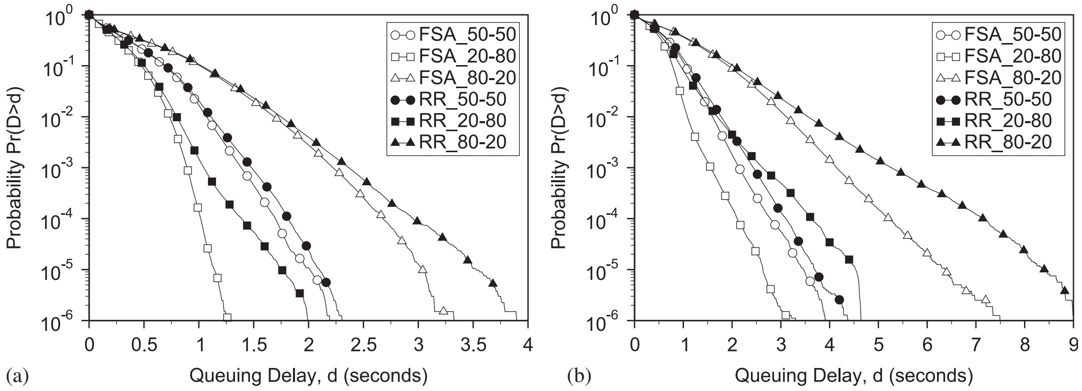

Figure 16. DE traffic, RS2 request scheme, MMPP FS sources: (a) 75% load and (b) 90% load.

by an average of 0.76%. For smoother FS traffic (Table VI), improvement was once more achieved in high load Scenarios 2, 4, 5, while in the rest, AF packet delays were on average 0.87% greater than those of RR. Although in some Scenarios, FSA provided better AF performance than RR and vice versa, the difference in AF delay between the two schemes is millisecond order with a maximum value of 1.812 ms in FS traffic Scenario 3 (Table VI, Figure 15(b)). This fact shows that AF performance is almost the same for both the schemes. Hence, FSA improvement in performance for RS2 requests concerns DE traffic with a minor negative impact on AF delay.

DOI: 10.1002/sat

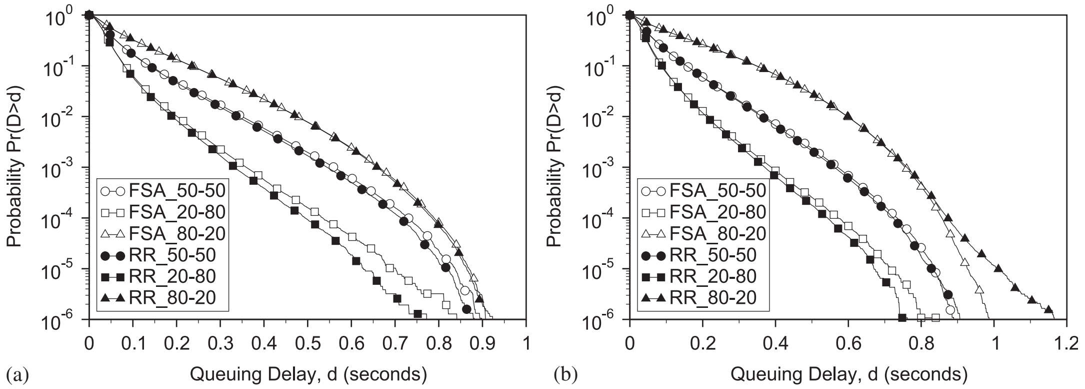

Figure 17. AF traffic, RS2 request scheme, PMPP FS sources: (a) 75% load and (b) 90% load.

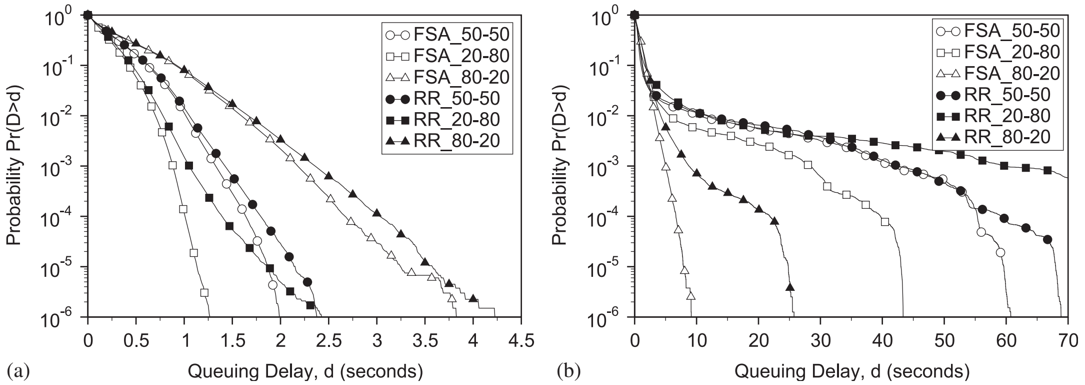

Figure 18. DE traffic, RS2 request scheme, PMPP FS sources: (a) 75% load and (b) 90% load.

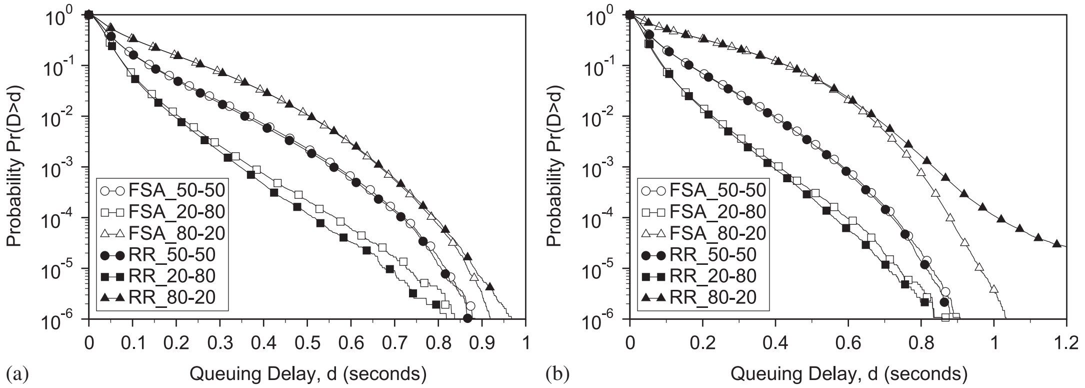

Figure 19. AF traffic, RS2 request scheme, MMPP SS sources: (a) 75% load and (b) 90% load.

Comparing the two request schemes in terms of packet delay, it is obvious that in RS1, a slight improvement in AF delay comes at the cost of poor DE performance especially in Scenarios 4, 5, 6 (Figure 5(a)) with SS and bursty LRD traffic. In these Scenarios, the extended DE delays are caused by the inability of RS1 to represent effectively the capacity needed each time by the RCSTs. In this way, low priority DE traffic is suspended, while the performance of higher priority AF traffic is not increased dramatically. On the contrary, RS2 requests proved to be more accurate and predictive, as DE performance is drastically improved in the same heavy traffic

DOI: 10.1002/sat

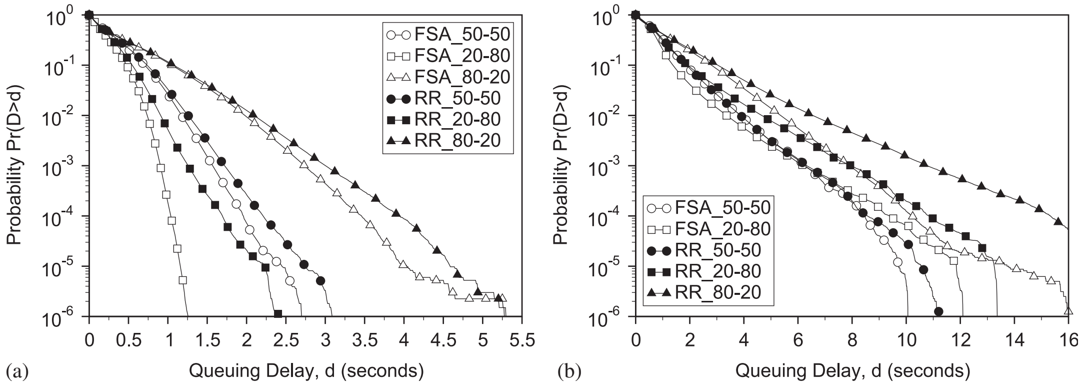

Figure 20. DE traffic, RS2 request scheme, MMPP SS sources: (a) 75% load and (b) 90% load.

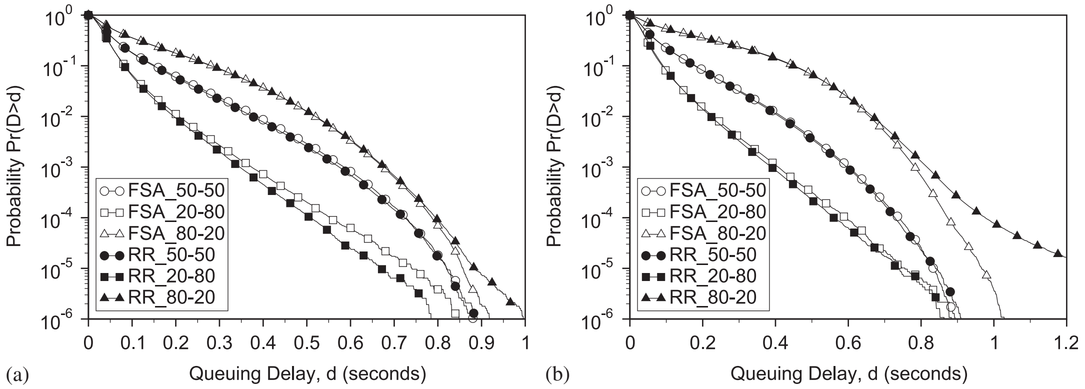

Figure 21. AF traffic, RS2 request scheme, PMPP SS sources: (a) 75% load and (b) 90% load.

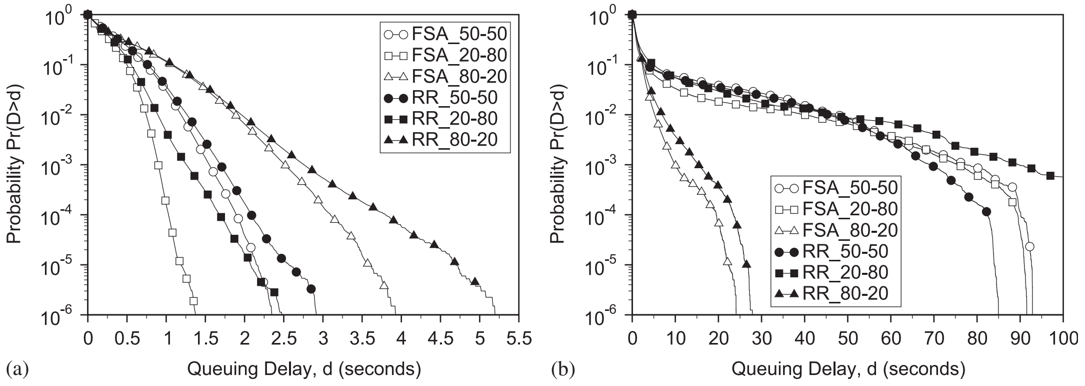

Figure 22. DE traffic, RS2 request scheme, PMPP SS sources: (a) 75% load and (b) 90% load.

Scenarios 4, 5, 6 (Figure 5(c)), despite the slight increase in AF delay. This fact also shows that in high network load, minor changes in AF performance affect positively DE traffic to a large extent.

The performance of the FSA algorithm, as shown by the results discussed earlier, is influenced greatly by the capacity request policy. This happens because the RCST data list is sorted prior to each allocation according to the calculated RSR ((Equations (8) and (9)) for each terminal. However, the RSR is affected by both the number of slots assigned and the number of slots requested. Therefore, the RSR is influenced by the allocation and request schemes used.

DOI: 10.1002/sat

As a result, it is the combination of the selected allocation and request schemes that determine the network performance.

7. CONCLUSIONS AND FUTURE WORK

In this article, we introduced an algorithm that regulates capacity allocation in the return channel of a DVB-RCS compliant satellite network. The proposed allocation scheme adjusts each time the order by which the terminals receive capacity according to a fairness criterion introduced. In this way, we managed to improve AF and DE traffic performance in the majority of the Scenarios simulated. We have also investigated the performance of two capacity request calculation schemes. The first one (RS1) yielded better results for AF traffic, whereas the second (RS2) provided better DE performance. Ultimately, simulation results showed that the proposed allocation scheme can improve both AF and DE traffic performance in many cases compared with RR allocation. However, a compromise between AF and DE traffic performance is needed, as the results show that they cannot be improved simultaneously. In our case, the combination of the proposed FSA algorithm with RS2 request mechanism [10] seems more capable of providing QoS along with high link utilization.

Possible issues of future study in this area could be the enhancement of the allocation scheme with CRA capacity assignment mechanisms and more complex fairness criteria. To be more precise, the link quality could also be considered an additional factor for giving priority in capacity assignment to the terminals. Capacity request calculation can also be improved with traffic prediction techniques to counteract the problems caused by the scheduling lag.

ACKNOWLEDGEMENTS

This study was performed within the framework of the SatNEx project.

REFERENCES

1. Liang X, Ong FLC, Pillai P, Chan PML, Mancuso V, Koltsidas G, Pavlidou FN, Caviglione L, Ferro E, Gotta A, *et al*. Fusion of digital television, broadband Internet and mobile communications-part ii: future service Scenarios. *International Journal of Satellite Communications and Networking* 2007; **25**(4):409–440. DOI: http://dx.doi.org/10.1002/sat.880.
2. ETSI EN 301 790 V151 (2009-01). Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems. *Technical Report*, ETSI, January 2009.
3. ITU-T G1010. End user multimedia QoS categories. *Technical Report*, International Telecommunication Union, November 2001.
4. ETSI EN 302 307 V112. Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications. *Technical Report*, ETSI, June 2006.
5. Iuoras N, Tho Le-Ngoc MA, Elshabrawy T. An IP-based satellite communications system architecture for interactive multimedia services. *International Journal of Satellite Communications and Networking* 2003; **21**(4–5):401–426. DOI:10.1002/sat.758.
6. ETSI TR 101 790 V131 (2006-09). Digital Video Broadcasting (DVB); Interaction channel for Satellite Distribution Systems; Guidelines for the use of EN 301 790. *Technical Report*, ETSI, September 2006.
7. Blake S, Black D, Carlson M, Davies E, Wang Z, Weiss W. An Architecture for Differentiated Service. RFC 2475 (Informational) Dec 1998. Available from: http://www.ietf.org/rfc/rfc2475.txt (updated by RFC 3260).
8. Grossman D, Heinanen J. Multiprotocol Encapsulation over ATM Adaptation Layer 5. RFC 2684 (Proposed Standard) September 1999. Available from: http://www.ietf.org/rfc/rfc2684.txt.
9. Gotta A, Potortì F, Secchi R. Simulating dynamic bandwidth allocation on satellite links. *WNS2 '06*: *Proceeding from the 2006 Workshop on ns-2*: *The IP Network Simulator*, ACM: New York, NY, USA, 2006; 8. DOI: http://doi.acm.org/10.1145/1190455.1190462.
10. Iuoras N, Le-Ngoc T. Dynamic capacity allocation for quality-of-service support in IP-based satellite networks. *IEEE Wireless Communications* 2005; **12**(5):14–20. DOI: 10.1109/MWC.2005.1522099.
11. ITU-T G114. One way transmission time. *Technical Report*, International Telecommunication Union, May 2003.
12. Fischer W, Meier-Hellstern K. The Markov-modulated Poisson process (MMPP) cookbook. *Performance Evaluation* 1993; **18**(2):149–171. DOI: http://dx.doi.org/10.1016/0166-5316(93)90035-S.
13. Le-Ngoc T, Subramanian SN. A Pareto-modulated Poisson process (PMPP) model for long-range dependent traffic. *Computer Communications* 2000; **23**(2):123–132. DOI: 10.1016/S0140-3664(99)00166-8.
14. Shah-Heydari S, Le-Ngoc T. Parameter estimation of switched Poisson process models for short- and long-range dependent traffic. 2003; **2**:1035–1038. DOI: 10.1109/CCECE.2003.1226072.

15. Liu SG, Wang PJ, Qu LJ. Modeling and simulation of self-similar data traffic. *Proceedings of the International Conference on Machine Learning and Cybernetics* (*ICMLC*), vol. 7, Guangzhou, China, 2005; 3921–3925. DOI: 10.1109/ICMLC.2005.1527623.

16. Jain R, Chiu D, Hawe W. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. *DEC Research Report TR-301*, Digital Equipment Corporation, Maynard, MA, U.S.A., September 1984.

AUTHORS' BIOGRAPHIES

Panagiotis Chatziparaskevas received his Diploma in Electrical and Computer Engineering from Aristotle University of Thessaloniki, Greece, in 2008. He has attended one-semester courses in the Department of Industrial Informatics and Instrumentation in the Polytechnic School of Grenoble, France, as an exchange student. His research interests are in resource allocation techniques, medium access control (MAC) protocols and in wireless terrestrial, satellite and vehicular networks. He has been involved in national and European research projects.

Georgios Koltsidas received his PhD degree and Diploma in Electrical and Computer Engineering from the Aristotle University of Thessaloniki, Greece, in 2009 and 2003, respectively. His major research interests are in routing protocols for mobile ad hoc and sensor networks. He is also interested in resource management techniques in wireless terrestrial and satellite networks. He has been engaged in lectures and laboratory courses and he has been involved in many national and European research projects.

Fotini-Niovi Pavlidou (S86, M87, SM00) received the PhD degree and the Diploma in Electrical Engineering from the Aristotle University of Thessaloniki (AUTh), Greece, in 1988 and 1979, respectively. She is with the Department of Electrical and Computer Engineering at AUTh engaged in teaching in the areas of mobile communications and telecommunications networks. Her research interests are in the field of mobile and personal communications, satellite communications, multiple access systems, routing and traffic flow in networks and QoS studies for multimedia applications over the Internet. She is participating in many national and international projects (Tempus, COST, Telematics, IST), and she has been chairing the European COST262 Action on Spread Spectrum Systems and Techniques for Wired and Wireless Communications. She has served as member of the TPC in many IEEE/IEE conferences and she has organized/chaired some conferences like the 'IST Mobile Summit2002', the 6th 'International Symposium on Power Lines Communications-ISPLC2002', the 'International Conference on Communications-ICT1998', etc. She is a permanent reviewer for many IEEE/IEE journals. She has published more than 100 papers in refereed journals and conferences. She has served as guest-editor for special issues in many journals. She is a senior member of IEEE, currently chairing the joint IEEE VT&AES Chapter in Greece.

DOI: 10.1002/sat