A simulation-based algorithm for solving the resource-assignment problem in satellite telecommunication networks

Author: David Fernández Piñas, Master in Free and Open Source Software, UOC

Tutors: Sergio González Martín, Ángel Juan Pérez, IN3, UOC

Abstract

This paper proposes an heuristic for the scheduling of capacity requests and the periodic assignment of radio resources in geostationary (GEO) satellite networks with star topology, using the Demand Assigned Multiple Access (DAMA) protocol in the link layer, and Multi-Frequency Time Division Multiple Access (MF-TDMA) and Adaptive Coding and Modulation (ACM) in the physical layer. The objective is to design an algorithm that allows processing a given traffic profile with packet expiration time as delay constraints and a maximum packet loss rate, using the minimum possible spectrum bandwidth. When there is not any structure imposed to the MF-TDMA super-frame, the resource-assignment problem becomes a combinatorial problem which can be seen as a two-dimension (2D) oriented strip packing problem with additional constraints. The well-known Best Fit Decreasing Height (BFDH) heuristic for 2D packing is used as a basis for the proposed allocation algorithm, which should be able to obtain a set of candidate solutions in the order of a few hundredths of milliseconds. Later it is proposed to randomize and parallelize the heuristic in order to produce several candidate solutions, among which to select the optimum, which is the one that minimizes the overall bandwidth consumption.

Keywords: GEO satellite network, DAMA, MF-TDMA, ACM, Radio Resources Management, RRM, 2D oriented strip packing, network dimensioning, allocation, scheduling, QoS, randomization, parallelization.

1. Introduction

Resources management in communications networks deals with the sharing of transmission resources among the users of the network. A transmission resource provides an amount of communication capacity. From the user perspective is viewed as something needed in order to be able to send an amount of bits on a given time period with a minimum reliability. Depending on the medium access design, a transmission resource can be a chunk of spectrum, a time slot on a chunk of spectrum, a Code Division Multiple Access (CDMA) code, or a combination of these (Açar 2001).

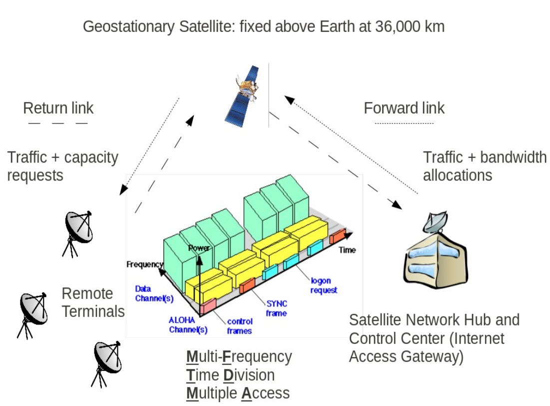

Figure 1: MF-TDMA GEO Satellite Communication System (adapted from Fairhurst 2001)

In case of having just one hub on a satellite network, as shown in Figure 1, the satellite return link, which goes from the terminals to the hub, is the only shared medium that requires a Medium Access Control (MAC) protocol.

As an example, the proposed MAC protocol by the Digital Video Broadcasting - Return Channel Satellite (DVB-RCS) standard (ETSI 2005) is the Demand Assigned Multiple Access (DAMA) protocol. With DAMA, a Random Access (RA) channel using a protocol such as Slotted Aloha (SA) (Roberts 1975) can be used as a signaling channel to convey small Capacity Requests (CR) towards the hub. On the hub, a resource allocation process periodically processes all the CR received from terminals in order to build a Burst Time Plan (BTP), which indicates the amount of transmission resources assigned to each terminal for its data traffic transmissions until the next BTP reception.

Resource allocation algorithms are not defined within the DVB-RCS standard (ETSI 2005) nor is any specific method suggested within the DVB-RCS guidelines (ETSI 2005). Each commercial DVB-RCS system uses its own confidential algorithms, which are a mean to differentiate a DVB-RCS system from the competitor's.

The structure of this paper is as follows: section 2 introduces the conceptual framework, which discusses the concepts involved in the problem being addressed. Section 3 performs a problem accurate description and offers a formal model description in terms of a minimization problem with certain assumptions and constraints. Section 4 performs a literature survey of related papers on the telecommunications field, 2D packing heuristics and meta-heuristics and also about the future possible application of randomization and parallelization to heuristics. Section 5 explains our approach for solving the problem stated in section 3. It is explained the heuristic general design and also how randomization and parallelization could be applied to the heuristic in the future. Section 6 shows the pseudo-code of the heuristic and points possibilities for implementation of a randomized version. Section 7 shows the computational experiments performed so far with the design heuristic and in section 8 the obtained results are discussed. Finally, section 9 details future work and section 10 summarizes the main contributions and findings of this paper.

2. Conceptual Framework

The Radio Resources Management (RRM) problem on a satellite network, introduced by the use of the DAMA protocol, is basically how to periodically distribute available bandwidth and time, which define a Super Frame (SF), for data traffic transmissions among a group of terminals requesting capacity. Depending on the assumptions and constraints considered, the resources usage optimization problem can be formulated in different ways. Hereafter are introduced and clarified some concepts involved with this topic, in order to describe the problem in detail later in section 3.

2.1 Traffic Profile, Quality of Service (QoS) and types of CR

The terminals have to process different types of traffic belonging to different applications. A broadband multimedia network used for Internet access can process a different traffic profile depending on whether the user is a consumer or a professional (prosumer). Moreover, the traffic profile can be different if the satellite network is used for an specific application such as remote monitoring, point-of-sales transactions or aeronautical communications (Air Traffic Management or ATM), instead of Internet and web access.

Voice traffic has been traditionally characterized on telephony networks using a Poisson process, with call durations following an exponential distribution, while data traffic has been characterized using a self-similar model using the Pareto distribution for both, packet sizes and packet inter-arrival times (Becchi 2008). The parameters for the models are usually derived from live traffic captures statistics of packet sizes and inter-arrival times.

Another approach is to use a live traffic packet capture and then create an potentially infinite traffic profile with the same statistical characteristics than the captured sample using the bootstrapping technique (e.g. Teknomo, 2006), which can be done by numbering the packets in the live traffic packet capture and then using a random uniform distribution for re-sampling the packet capture and indefinitely select a next packet that will be generated.

Each type of traffic in the profile can also have some QoS constraints associated. For example, voice must be served with a maximum delay and packet loss to be intelligible. There can be also requirements for the maximum voice call establishment time. While some data applications traffic, such as FTP, email or P2P can be served on a best-effort basis, i.e using remaining bandwidth after processing traffic with QoS constraints, interactive applications, such as web browsing, chat or remote consoles can have latency constraints (e.g. it can be considered that the great majority of users will give up if a web page takes more than 10 seconds to load, or even less), so the CR for some data traffic can have an associated expiration time, after which it is not useful to allocate the requested capacity, because the connectivity would be considered too bad for the given application and the user would have desisted or switched to an alternative mean to perform the required communication. Some data applications, e.g. network management, can consider also that the maximum expiration time can be achieved sometimes, while a threshold in this number of times is not exceeded.

The terminals of a satellite network do not request directly a portion of Super Frame Bandwidth (SF_Bw) and transmission time they need. Instead, they will periodically report its link condition in terms of signal power and the amount of capacity needed for each CoS. In the DVB-RCS standard there are two types of requests defined: volume-based, which correspond to VBDC (Volume Based Dynamic Capacity) CR, and rate-based, which correspond to both, RBDC (Rate Based Dynamic Capacity) and CRA (Constant Rate Assignments) CR. In the DVB-RCS guidelines (ETSI 2005), it is recommended to map rate-based requests to streams of voice and video, while volume-based requests to data traffic. RBDC granted capacity has an associated timeout relatively short, so the resources have to be re-requested periodically, while CRA is granted while the terminal is logged on into the network. Another approach (Mitchell 2004) is to use a high timeout value for rate requests, but to implement the possibility of sending a capacity release signaling message.

2.2 Spectrum bandwidth

The overall spectrum bandwidth of the satellite network is a fixed parameter. It is ideally determined during the design of the network, taking into account the foreseen amount of traffic to convey and the capacity provided by the link budget and the modulation and coding techniques used by the satellite modems. It can come imposed also by the available capacity of the satellite used or even by the budget available for spectrum by the satellite network operator. The bandwidth of the satellite network is usually leased to a satellite operator and payed on a monthly basis, used or not. The bandwidth available, jointly with the link budget, determine an upper theoretical limit to the maximum amount of traffic the satellite network will be able to forward, given by the capacity computed by the well known Shannon formula (Shannon 1948).

2.3 Adaptive Coding and Modulation

More and more, specially with the new satellite systems in Ka band, Adaptive Coding and Modulation (ACM) is used in order to keep a constant Bit-Error-Rate (BER) in spite of changing link budget conditions, by selecting a more robust MODCOD when Signal-to-Noise Ratio (SNR) is low.

The MODCOD that a terminal uses for transmission is determined by its particular link budget condition on a given time, which depends on the location of the terminal on the satellite coverage zone and also on climate conditions (rain attenuation). The link budget condition can be estimated by the SNR measured by the terminal on the forward link.

Each MODCOD that can be used for transmission provides a different capacity granularity, i.e. an amount of link layer bytes that can be sent on its physical layer burst format. The MODCOD symbol rate determines the amount of bandwidth its bursts use, while the coding and framing structure the link layer bytes that can be conveyed. Usually, the rates, and so the bandwidths, have values that are multiples of a minimum value in order to ease network synchronization in time, necessary for the TDMA. Moreover, usually it is assumed that in case a burst is not completely filled by user data, padding must be performed, although packing can be used also, with a associated packing timeout to configure.

2.4 Superframe structure

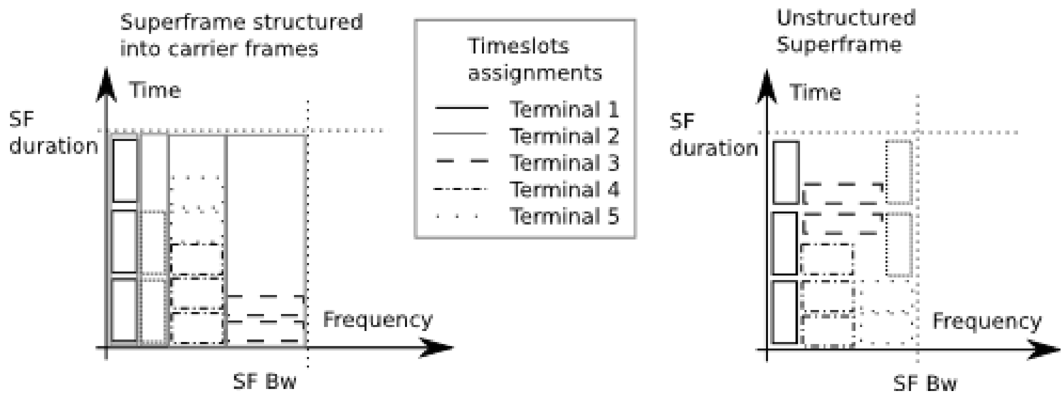

Figure 2: Structured and unstructured superframes timeslots assignments

As shown in Figure 2, the bandwidth (Bw) is usually subdivided in carriers, which can have the same or different bandwidth. Each carrier can contain slots of the same or different durations, due to the use of the same or different MODCOD (MODulation and CODing) schemes. The sequence of timeslots of a given carrier is called a frame. The group of frames of the different carriers forms the Super Frame (SF).

As shown also in Figure 2, it could be possible to define an unstructured SF, i.e. not divided into frames, where timeslots of different bandwidths and durations are packed. This scheme is more complex computationally. It can also lead to an increase in the signaling data needed to transmit the BTP, which is better to minimize to keep overall spectrum utilization low, considering both traffic and signalling in both, the return and the forward link. For the description of the BTP it is necessary to send: the frequency, bandwidth, starting time and duration of each burst allocated to each terminal. On the other hand the unstructred SF approach allows the maximum optimization regarding the overall bandwidth needed, as shown in Figure 2. Because of this an unstructured SF is considered in this study, but in order to minimize the signaling required, the allocation algorithm proposed gives preference to contiguous, in time, and at the same frequency and bandwidth, to the joint packing of transmissions from the same terminal, as explained later in section 5.

2.5 Superframe duration and assignments period

Although in order to reduce the amount of bandwidth for the RA signaling channel, different allocation periods can be considered according to the different traffic Classes of Service (CoS), with different delay requirements: voice, chat, email, bulk data transfers... usually, because of simplicity, the assignments period (TA) is equal to the SF period (SF_T) for all traffic.

There is a trade-off in the assignment of a period to DAMA capacity assignments. The larger the period, a more efficient assignment can be performed, using more sophisticated placing algorithms, and with less signaling messages (CR), but the responsiveness of the satellite communications system will be lower, which can make a bad user experience for interactive applications, such as database requests, web browsing, chat or interactive remote consoles.

On the other hand, in case that the SF_T is small compared to the duration of the frames to allocate, it may make no sense to allocate resources using a complex heuristic, and the allocation will be less efficient.

The possible options for the request-assignment cycle period are:

- No period, the assignments are done on-line and published as soon as CR are received. This provides the most responsive possible system but is the most inefficient option.

- Fixed period of tenths of ms, e.g. 26 ms or 32 ms, or hundredths of ms, e.g. 100 ms or 200 ms. This period values are typically used on DVB-RCS networks for Internet access, aiming to provide high responsiveness and a minimum efficiency on the assignment of resources.

The period of allocations TA (Time of Allocations) determines the operation period of the resource allocation process, meaning that it is the minimum period in which allocation of resources to the terminals can be modified (Indra, UPC GRCM 2010). In that respect, in case that the resource allocation must ensure specific delay requirements it is convenient that the allocations period is much below these requirements, so that the resource allocator will have enough flexibility to modify assignments in front of incoming traffic. This is the case considered in this paper, where the traffic profile considered has strict Quality of Service (QoS) requirements in terms of delay, like Air Traffic Management (ATM) applications (ESA 2011). Taking into account a delay requirement given by TD (Time Delay) and that the procedure of requesting resources and the subsequent transmission will involve a minimum propagation delay of 1.5 RTT (Round Trip Time), a first condition to be held is:

`TA  << TD - 1.5 RTT`

Considering an engineering criterion `TA  <= 0.1(TD - 1.5 RTT)`

If `TD = 4.7 s` and `RTT = 540 ms`, then `TA  <= 390 ms`.

2.6 Other restrictions and characteristics of the RRM problem

Another restriction that must be considered by the resource allocation algorithm is the fact that a terminal has just one transmitter, so that it can not send more than one burst simultaneously, i.e. simultaneous transmission on different frequencies is not possible.

The resource allocation process is periodically executed in two phases (Lee 2003) or three if ACM is considered (Lee 2004). First, the requested capacity is computed overall per terminal and CoS and prioritized by some criteria, e.g. giving priority to terminals with worst link conditions, keeping a proportional fairness (considering link conditions or not, i.e. ACM) and giving priority to some CoS (Vazquez 2005) or just by following an Earliest Deadline First (EDF) policy with received CR, in case they include expiration time information (Modiano 1997). In the case of using EDF it can be considered also the volume requested to avoid postponing long messages in favor of shortest ones (Indra, UPC GRCM 2010).

If ACM is used and the SF is structured, the second step is computing the number of carriers of each type needed. This computation can have a longer period than SF_T in case it is a slow fading that depends only on meteorological conditions (Aroumont 2008) and not on the terminals movement (fast fading).

On the last step, the BTP is built and resources are assigned to terminals and CoS in the process, following at least the mentioned constraints for time and frequency assignments: overall SF structure, bandwidth and duration, and not allowing simultaneous transmissions on different frequencies by one terminal.

CR that can not be allocated on a given period are left for the next, and eventually they will get more priority then. The case of interest, of course, is when the network is congested and there are always more CR to allocate than the capacity of the SF, but this situation can not be permanent, i.e. it must happen without entering the case of a constantly growing queue of CR, which would indicate that the network is overloaded.

In general, two questions must be addressed when designing a dynamic bandwidth allocation procedure (Morell 2008): first, how much structure is imposed to the multiple access scheme, then, within the given structure, how are resources optimally distributed. As it has been commented in section 2.4, a highly structured approach gives less degrees of freedom and simplifies the optimization. Giving no predefined structure to the MF-TDMA SF, which is the optimum in terms of minimum bandwidth utilization (see Figure 2), leads to an NP-hard combinatorial problem.

3. Problem accurate description

We consider a satellite network with one hub and several terminals, as shown in Figure 1, using an unstructured MF-TDMA SF, as shown in the right side of Figure 2. The terminals can use for its data and voice messages transmissions three types of bursts, with the same area, depending on its link conditions: bad, average and good; with bandwidths Bw, 2Bw and 3Bw; and durations 3T, 1.5T and T respectively, shown in Figure 3.

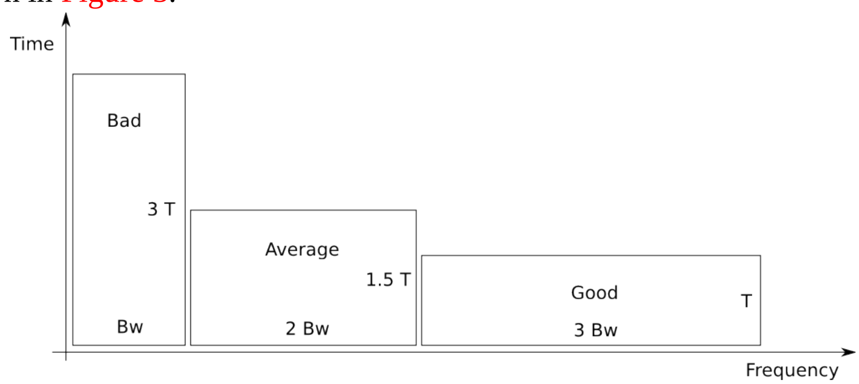

Figure 3: Three types of Radio Frequency (RF) bursts for three channel conditions

A synthetic traffic profile of packets to be processed by the satellite network, using a predefined number of RF bursts with a given timeout has been generated and is available at Fernández 2012. The bootstrapping technique (Teknomo 2006) can be used to generate a traffic profile as long as needed from these generated traces. Traffic profile QoS requirements in terms of latency are specified in terms of expiration times for a group of RF bursts corresponding to a message to be transmitted. It is assumed that all the bursts belonging to a message to transmit must be sent before the overall message expiration time.

The traffic profile determines the types and amount of CR the DAMA allocator will receive by the RA channel used for CR sending. The dimensioning of the RA channel for CR has been left out of scope of this study. It is assumed that the input traffic profile represents already received CR at the DAMA allocator with corresponding timeout values updated after its transmission by the CR RA channel, using more than one retransmission with a negligible probability.

Figure 4 shows the overall number of bursts of all three types to pack that appear each SF_T of 390 ms in the most loaded traffic profile considered (file #3).

Figure 5 shows that the CR scheduling problem can be illustrated like a kind of Tetris like game (Fahey, 2007). The traffic profile trace file time column indicates the time at which a CR for a message transmission need arrives at the allocator. It is located at a distance “timeout” of current time and is comprised of a set of bursts to transmit of a given bandwidth and duration, according to one of the three types shown in Figure 3. In the traffic profile it has been assumed that all bursts belonging to a single message are of the same type, i.e. that transmitter MODCOD finally used is known (obtained by a real traffic capture), so there is no need to simulate the ACM process and loop.

As they are received, the allocator stores CR in order of increasing timeout on a buffer per transmitter. When a new CR arrives it could have to modify the timeouts of already present CR so there is no time overlap of burst transmissions from a single terminal, due to the constraint of each terminal having just one transmitter. It could happen that as a result of the insertion of the just arrived CR some messages get a negative timeout, which means transmission capacity in the short time is overloaded. This can happen if the scheduling policy is too conservative and waits too much, and almost to the transmission timeout expiration, to perform the scheduling of the CR and its corresponding allocation.

A configurable limit, so called obliged transmission threshold can be configured in the proposed scheduling heuristic. The minimum value of the obliged transmission threshold is the SF period. Then, there is a trade-off between imposing a limit to the packet losses due to transmitter overload, by setting a high enough obliged transmission threshold, and the amount of bandwidh required to satisfy the QoS constraints in terms of delay and packet losses of a given traffic profile, although not always setting a lower obliged transmission threshold leads to less bandwidth needed for a given traffic profile, as results show.

Regarding packet losses due to transmitter overload condition, three overall limits will be considered to obtain results, selected by the value of the obliged transmission (Tx) threshold, shown in Table-1, assumed to be representative of ATM data applications QoS needs in terms of PLR (Packet Loss Rate).

CoS	Expected PLR due to overload	Obliged Tx Threshold
High	0.0 (no losses allowed)	High
Medium	Some losses	(High+SF_T)/2
Low	Potentially maximum losses	SF_T

Table-1 – CoS considered PLR

Figure 4: Number of new bursts to pack that appear each assignment period in file #3 (Fernández 2012)

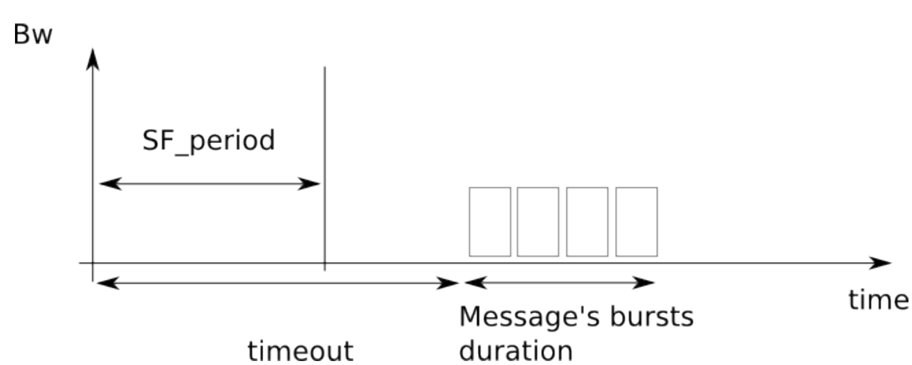

Figure 5: Traffic profile trace parameters illustrated

The goal of RRM is to optimize bandwidth utilization while satisfying the QoS of different messages (Xu 2011), with different delivery time restrictions T_out, that belong to several users. So the objective of the study is to find the minimum bandwidth (SF_Bw_min) needed to process a given traffic profile TPB, during T_end seconds, on the described type of satellite communications network. In order to find this minimum bandwidth an allocation algorithm, in this case a bursts packing algorithm (explained later in section 5), has been developed that solves the next minimization problem:

Minimize first SF_Bw(TPB, SF_T, lambda) = SF_Bw_min, and secondly BTP size subject to:

- Each SF_T seconds, a given SF j has to be filled with a set TPB_j of N bursts b with a given frequency f and time t dimensions from a possible set, see Figure 3, and a timeout T_out each one that is updated each SF_T.

  $TPB_j = \{b_i(f_i, t_i, T\_out_{(i,j)}), i=1..N\}$

  $(f_i, t_i) \in \{(Bw, 3T), (2Bw, 2T), (3Bw, T)\} \Rightarrow SF\_Bw\_min \bmod Bw = 0$

- T_end = m·SF_T, where m is the number of SF needed to process the traffic profile. T_end is determined by the traffic profile latency QoS requirements.
- SF_T computed from the most stringent data CoS expiration time (TD) and minimum RTT, which constraints the maximum execution time of the allocation algorithm (<< SF_T) and is a multiple of T (shortest timeslot duration), so instead of 390 ms, computed in section 2.5, 360 ms will be considered, assuming T=20 ms:

  $SF\_T \leq 0.1(TD - 1.5\ RTT)$

  $SF\_T \% T = 0$

- All bursts in the profile must be processed. Burst lost (lambda) allowed due to transmissions exceeding the expiration time due to overload, as shown in Table 1.
- A single terminal can not transmit in more than one frequency simultaneously, i.e. a terminal should not be assigned more capacity than it is able to use.
- In order to minimize BTP signaling size (amount of data required to notify the assignments to the terminals) transmitters frequency hopping must be minimized, i.e. it is preferred to send a single transmitter bursts of the same bandwidth in a time row whenever possible.
- Remaining capacity after allocations of bursts are performed is not assigned to any terminal.

In section 5, the approach proposed for solving this problem is explained, but before, the next section 4 shows a literature review of this topic.

4. Related work (literature survey)

4.1 Telecommunications literature on RRM

The earliest approaches found to the problem of the optimization resources usage on satellite networks using Bandwidth-on-Demand (BoD) do not consider ACM. They aim to optimize the resource sharing in case of congestion, i.e. more capacity is requested than resources available; as an effort to provide the maximum fairness among streams competing for the available resources. The optimization problem is modeled as an integer optimization problem that is derived from Game Theory by Açar (2001). Essentially, the solution aims to maximize the product of the amount of resources allocated to each stream without exceeding the size of the resource pool that is being shared.

In contrast, Priscoli (2004) proposes the decoupling of the congestion control and the BoD mechanism by using control theory concepts (Smith predictor) and modeling the system as a time-delay system. The terminals make CR in order to track a given maximum reference sizes of its transmission buffers and avoid over requesting. In case of congestion the max-min fairness criterion is applied, i.e. all users with unsatisfied CR are allocated the same amount of capacity, optionally applying some weights. With this model, the introduction of ACM does not change the algorithm. Bandwidth variations due to changing channel conditions are considered perturbations, like the ones due to another traffic on the network, to be compensated by the control-theoretic algorithm, which tries to keep a reference queue size (Pietrabissa 2008). This reference queue size determines the network utilization and is adapted in function of estimated network congestion, which is performed by monitoring the queue size itself.

Other authors, instead of maximizing the fairness between users, try to maximize the overall throughput of the return link. They model the problem as a linear integer programming problem and try to minimize the sum of the overall differences between the capacity requested and assigned subject to several constraints (Lee 2003). When introducing ACM, the SF is divided into at least two different types of carriers. The problem of maximizing the overall throughput is then modeled as a non-linear integer programming problem (Lee 2004). Then it is not possible anymore to use an algorithm to find the exact solution to the optimization problem in the available time, because it becomes NP-complete, so an heuristic is proposed based on the similarity of this problem to the Knapsack problem. A weight is assigned to each traffic and delay class of each user, which allows prioritizing traffic depending on applications characteristics.

Aroumont (2008) presents an algorithm, although not evaluated, that considers again the fairness among users as the optimization criteria, but considering also ACM. The proposed allocation algorithm is based on a water-filling approach. (Morell 2008) considers that the introduction of ACM leads to the need of a cross-layer approach. The resulting RRM problem is cast into a Network Utility Maximization (NUM) problem, which in turn maximizes fairness among users. In that paper it is considered that if no structure is imposed to the SF, the problem becomes NP-hard. Using results of the Game Theory, it is proposed that the utility function to maximize when sharing resources among different users is the product of assigned resources.

Note that in these mentioned works the capacity demand constraints are presented from a general perspective (fairness, throughput...), without relating them to specific requirements in terms of delay bounds (CR expiration times) and packet losses, as it is the case in this study and also in the ANTARES study (ESA 2011) for ATM communications.

In other approaches, the proposal is not to optimize the use of resources, but just implementing an algorithm, subject to some characteristics, policies and constraints, and then evaluating its performance, maybe benchmarking it with another algorithms. This is the case of (Mitchell 2004), where the resources are requested on a burst-by-burst basis and assigned following a round-robin policy. In another example (Vazquez 2005), the terminals with worst link conditions are given allocation priority.

In (Booton 2008) paper, the DAMA protocol is not performing allocations on an SF basis, as in the previously mentioned works. It is used to allocate or deallocate satellite bandwidth time slots on a more long-term basis. The allocations are done on-line, i.e. as CR arrive, not periodically. In that case, it is also not seen as feasible finding the exact solutions to the optimum allocation. The allocation problem is addressed as a packing problem and Best Fit heuristics are proposed for the problem solution and compared with the previously used First Fit heuristics.

Although not directly related to satellite networking, it is worth mentioning that the problem of allocating user connections to a set of base stations, using bin packing algorithm based heuristics, is also considered by other authors.

Xing (2005) uses the First Fit Decreasing heuristic with additional mechanisms in order to accommodate flows, requiring a certain bandwidth and delay QoS constraints, to five available access networks. The bandwidth of each access network is analogous to the capacity of a bin. The bandwidth requirement of a traffic flow is analogous to the size of an item to pack. The objective is reducing power consumption of user terminals while respecting the application preferences, a priori defined by users.

Mariz (2006) also draws a parallel with the bounded space variable-size on-line bin packing problem in order to deal with the problem of allocating user services onto a set of communication resources on cooperating access networks. Applications are the objects to be packed and access networks are the bins. There is a finite number of bins (bounded space) and each of them can have a different size (variable-size). Applications to pack are not known in advance (on-line problem) and applications' size also depends on the access network they will be finally assigned to. It evaluates three well-known bin packing algorithms: First Fit, Best Fit and Worst Fit. In addition, an heuristic called Less Voice is evaluated, which selects the access network where the cost of the application bandwidth relative to voice cost is minimum. The Less Voice heuristic is found to provide the best performance in terms of blocking probabilities and it imposes the least slow down for elastic sessions. This is because this heuristic takes into account resource consumption of each application class. A random heuristic is evaluated also as a worst case reference for the others.

4.2 2D oriented orthogonal strip packing algorithms

According to (Booton 2008), the BFDH would be the the best heuristic for 2D oriented orthogonal strip packing, at least better than the First Fit heuristic they previously were using.

The Bottom-Left (Chazelle 1983) is another heuristic typically considered for 2D packing problems due to its simplicity. (Lesh 2004) introduces an algorithm that makes a branch-and-bound exhaustive search based on the Bottom-Left heuristic for the 2D oriented orthogonal strip packing problem. Additionally, the algorithm is able to quickly determine if a given set of rectangles can be perfectly packed before running more expensive or less accurate algorithms. It is reported finding the optimum packing for 17 rectangles in less than a second, for 25 rectangles in 2 minutes and the larger problems with 30 rectangles in several hours, on a Linux machine with a 2 GHz processor running unoptimized Java code. Anyway, these times are orders of magnitude higher to the hundredths of milliseconds that we are pursuing (<360 ms).

It mentions also that the most natural permutation to choose for the Bottom-Left heuristic, and the one that works well also in practice, is to order the rectangles by decreasing height (bandwidth in our case), although it would be natural also to try sorting by decreasing width (duration in our case), area (the product of bandwidth and duration in our case), perimeter (in our case a sum of time and duration scalars) and then take the best of the four solutions. It is noticed that, when using the Bottom-Left heuristic, by ordering the set of bursts to allocate by decreasing width (duration in our case) it is guaranteed an allocation with a total height (bandwidth occupation) at most three time the optimum, but the heuristic is not competitive as an approximation when sorted by decreasing height (bandwidth in our case). It is also mentioned that it has been demonstrated that there are examples for which the Bottom-Left heuristic cannot produce the optimum packing under any ordering.

Then, it comments that other heuristics could be considered because of its potential for better solutions, but they would take substantially more time than the Bottom-Left heuristic, such as genetic algorithms or simulated annealing, which are meta-heuristics.

Finally, it explains also their use of the Smallest-Gap heuristic, based on trying to fill the smallest horizontal gap first, which is suspected that slightly outperforms the Bottom-Left heuristic. In parallel, a variation of the Bottom-Left, called Left-Bottom, is also tested, and a variation trying to fill the smallest vertical gap first, that are able to solve some cases where the Bottom-Left and the Smallest-Gap heuristics fail.

According to Ntene (2007), off-line 2D oriented orthogonal strip packing problems may be solved using exact algorithms, level heuristics or plane heuristics. In level heuristics the strip is partitioned into horizontal levels according to the tallest item packed on the level. In plane algorithms the strip is not partitioned and items may be packed anywhere in the strip.

In her thesis, 542 benchmark data sets among six categories (Vuuren 2006) are used to compare twenty six algorithms from existing literature: by the average packing height achieved, the frequency with which the smallest packing height is achieved and the execution time. There is a trade-off of execution time with the average packing height. A computerized system was developed using Visual Basic 6.0 that implements all mentioned algorithms in her thesis, that could recommend industry managers the most adequate algorithm for a given packing case. The final recommendation is that if the user is interested in obtaining results rapidly, then the algorithms in the Best Fit and First Fit classes are the best choice.

Imahori (2007) shows an efficient implementation of the Burke algorithm (Burke et al 2004) that requires linear space and O(n log n) time, where n is the number of items to pack. It calls it Best Fit, but it is an heuristic slightly different to the one reviewed by Ntene (2007) as BFDH. This other implementation of Best Fit dynamically selects the next rectangle to place according to available horizontal space and offers two alignment options (tallest or shortest), while in the original Best Fit implementation from 1990 (Coffman 1990), the rectangles are placed strictly according to its decreasing height order and aligned to the left. In general, the Best-Fit heuristic can place thousands of rectangles within 100 ms, but it cannot guarantee a constant approximation ratio in the worst case, as the Bottom-left or the Next Fit, although in practice it is shown that its solutions are just 10% over the optimal. It is worth mentioning that in our case, the rectangles to pack are grouped and prioritized according to its timeout, so the heuristic is completely determining the next item to pack. Moreover, items can not be rotated, because the time and frequency dimensions are not exchangeable and are determined by the channel conditions, so it is considered that the original Best Fit heuristic implementation (Coffman 1990) matches better our case than the Burke algorithm efficient implementation of Best Fit by Imahori (2007).

4.3 Meta-heuristics, randomization and parallelization

Hopper (2000) states that only a few researchers have experimented with meta-heuristics (such as genetic algorithms, naïve evolution or simulated annealing) for 2D packing problems. The majority of literature uses a meta-heuristic to determine the order in which the items are to be packed, combined with a placement heuristic such as the First Fit, the Best Fit or the Bottom Left. The items to be packed are ordered by decreasing height, then the meta-heuristic randomly swaps two randomly selected items a random number of times.

Even though the use of a meta-heuristic for items ordering increases the execution time, this hybrid approach usually works best than the placement heuristic with a non-increasing height ordered items, but it is strongly dependent on the nature of the placement routine and the problem size.

Regarding the clustering of rectangles, the findings of his survey indicate that even though initially genetic algorithms generated solutions improve rapidly, in the end, after some iterations, simulated annealing outperforms genetic algorithms and naïve evolution as meta-heuristics for ordering items to pack, generating denser layouts (Hopper 2001). Genetic algorithms performs similar to naïve evolution, so the crossover operator, which is the only difference between this two heuristics is not resulting in any difference. In terms of computing time, genetic algorithms and naïve evolution are more efficient than simulated annealing, which requires longer run times.

The quality of a packing is mainly determined in function of achieved height, but a weighted sum considering 30% for the packing density and 70% for the achieved height is used also (Hopper 2000), in order to consider also how tightly the items are packed in solution quality evaluation.

In genetic algorithms for packing problems, order-based chromosomes are used to represent packing sequences. The genetic cross-over operators must be designed to support the inheritance of important layout features.

Although, as mentioned, all follow a two-stage approach, three types of packing problems solution approaches involving genetic algorithms can be distinguished (Hopper 2001).

In the first type, the genetic algorithm is used only to determine the ordering of items to be packed by a placement routine. A second type tries to incorporate more layout information into the data structure of the genetic algorithm that determines the ordering of items to pack, in order to allow the inheritance of certain features that are determined later by the placement routine. First type order-based genetic algorithms achieve layouts of similar density as the second type, that include layout information into the genetic algorithm data structure. The third type applies the genetic algorithm into the layout itself, which is the approach followed also on the application of simulated annealing and tabu search meta-heuristics.

On a paper by Dagli and Poshyanonda (1997), reviewed in Hopper (2001), moreover to a genetic algorithm to generate the ordered list of items to pack following a Bottom-Left heuristic, an artificial neural network (ANN) is used to find the best match of the item to place on available empty areas.

Other algorithms reviewed there apply meta-heuristics manipulating the initial layout obtained by using an heuristic (Ratanapan and Dagli 1997). The efficacy of this method is not established since no comparisons are made with other approaches, but it can generate layouts of up to 97% packing density.

It is also reviewed there a paper from Kröger (1995) which states that genetic algorithms, improved with hill-climbing, achieve significantly better results than the original heuristic, random search and simulated annealing. The genetic algorithm improved with hill-climbing generates solutions closer to the best-known ones. Moreover, the concept of meta-rectangles is applied to simplify the problem, which are groups of adjacent densely packed rectangles combined into a single rectangle. This idea has been used in the design of the heuristic presented in this paper, which combines into a single burst, the transmissions from the same terminal in case they have the same bandwidth. The genetic algorithm cross-over operator ensures that meta-rectangles are transmitted to the offspring. This reduces the run times and improves the average best solutions found.

Although mainly dealing with genetic algorithms, his paper also reviews the application of other meta-heuristic such as simulated annealing, tabu search and neural networks to the 2D strip packing problem.

The first results of the application of simulated annealing to the 2D strip packing problem (Dowsland 1993) indicate that it is only capable of producing near optimal solutions, to be improved by other optimization techniques. Later results from Leung et al. (1999) indicate again that genetic algorithms outperform simulated annealing. Anyway, the efficiency of the solution search process depends on careful construction of the cooling schedule.

Han and Na (1996) use simulated annealing to improve an initial packing layout with reasonably good quality created by an ANN. The learning algorithm of the neural network is based on a Kohonen network. In order to achieve denser layouts, a force that drives items downwards and leftwards is introduced.

Tabu search and simulated annealing usually outperform genetic algorithms in 2D packing problems, but due to lack of benchmarking it is difficult to decide which of both meta-heuristic is better for packing problems (Hopper 2001).

Tabu search is a search technique that is guided by the use of adaptive or flexible memory structures. It is different to heuristic methods such as simulated annealing and genetic algorithms, because it contains some in-built memory mechanisms that prevent the search algorithm from returning to recently executed moves for a number of iterations. A tabu list is maintained containing all movements not allowed in the current iteration step. The search is guided by an objective function used to find the best next movement among the possible ones. Fewer solution approaches using tabu search have been proposed than with genetic algorithms and simulated annealing.

Lodi et al 1999 apply to the 2D bin packing problem the idea of generating an initial layout using a simple heuristic, which is then improved by tabu search using two possible movements. The first one attempts to remove an item from the worst bin, redistributing it among the other used bins. The second one tries to accommodate the item recombining the items of two other bins. The performance obtained by this method results better than bin packing heuristics and is comparable to a branch-and-bound (exact) algorithm.

Blazewicz et al (1993) use the tabu search to improve an initial packing layout of irregular items. An item is selected, then several new possible positions are tried and the best one is kept. Items that have changed their position during recent iterations are the members of the tabu list.

One of the main characteristics of meta-heuristic, as opposed to local search methods such as hill-climbing, is that they contain a means of escaping locally optimum solutions, by temporarily accepting solutions of lower quality. Healy and Moll (1996) introduce the possibility of performing downhill movements in local search methods such as hill-climbing. Pargas and Jain (1993) develop an stochastic optimization algorithm that combine hill-climbing and genetic algorithms characteristics. Unfortunately the methods where not compared to other meta-heuristic techniques.

Although applied to routing problems, Gonzalez-Juan (2012), introduces an example of how to outperform deterministic heuristics by transforming them into multi-start probabilistic algorithms. A random behavior can be introduced during the solution-construction process of 2D packing heuristics, instead of in the items ordering phase (Hopper 2000). By using biased probability distributions, such as the geometric one, the heuristics construction process is randomized without losing its inherent logics or 'common sense'. This approach is parameter free and can be massively parallelized.

When the constructive heuristic has an iterative stage, as it is the case for the BFDH heuristic for 2D packing, where previous levels are revisited to find the 'best next step movement', which is the one that leaves less free space on the level or the one that is able to pack the biggest burst in our proposed algorithm (see section 5); in order to randomize the construction it can be done the following: First, the feasible next steps are sorted following the heuristic criteria (in this case, by the remaining horizontal free space on the level, the less the better or by the biggest burst that can be packed in our proposal), with the best candidates being placed at the top of the list. Then, each candidate step is assigned some probability given by a biased distribution (e.g. a geometric or a decreasing triangular one). At the heuristic next step selection time, instead of choosing the absolute 'best candidate for the next movement', all the candidates are considered, with best steps- according to the heuristic criteria - having a greater probability of being selected. It is expected that by integrating this process in a multi-start schema, many different feasible solutions can be obtained in parallel, some of them possibly outperforming the one obtained with the corresponding deterministic heuristic. This is our proposed approach to randomization and parallelization of the heuristic presented in this paper, to be further studied and analyzed.

Work done by Juan et al (2010) suggests that a geometric distribution with any parameter randomly selected between 0.10 and 0.25 can be used. Each time the next heuristic selection step must be performed, a (quasi-) geometric distribution is randomly selected. This distribution is then used to assign (quasi-) exponentially diminishing probabilities to each eligible next step according to its position in the heuristic criteria sorted list of next steps.

Juan et al (2010) emphasize, citing other references, that moreover to solution quality and computing time, simplicity of implementation, flexibility and the lack of previous parameters fine-tuning or setup processes, which tend to be non-trivial and time-consuming, are also important qualities to look for in an algorithm for optimization.

5. Our approach

According to our problem description, illustrated in Figure 2, the bandwidth minimization problem described in section 3 can be modeled as an off-line 2D oriented orthogonal strip packing problem (Ntene 2007) with the mentioned additional restrictions, to solve each SF_T. The width of the strip is fixed (SF_T) and we want to minimize the height (SF_Bw), by packing at least all the bursts which timeout is less than the configured obliged transmission threshold, which is selected to have a maximum PLR due to transmitter overload events, which are detected by the expiration of transmission timeouts of packets pending transmission. The minimum bandwidth required is determined just by this obliged transmission bursts. Then, there is the packing of extra bursts that are not going to expire the next SF_T. The packing of this extra optional transmissions does not lead to the usage of more bandwidth than up to currently found SF_Bw_min value. They are added to increase the packing efficiency minimizing also the overall system latency offered to incoming traffic.

The bursts to transmit must be packed up orthogonally and without possibility of reorientation (no 90º rotation allowed), because time and frequency dimensions of bursts are not exchangeable. The packing is done up in the frequency dimension, without overlapping.

As packing problems are generally NP-hard, it means that it is unlikely that a time-efficient algorithm will be found which is capable to find the optimum solution. This observation directs towards solving the problem approximately using an heuristic, which is an strategy for solving optimization problems approximately by constructing good enough solutions at a reasonable computational cost.

In order to solve this minimization problem, the BFDH heuristic for 2D oriented strip packing (Coffman 1990) has been selected as a basis. Some changes have been introduced to cope with the additional restrictions explained in section 3 and also to simplify the implementation as much as possible. The proposed resulting heuristic can be executed in a few milliseconds (<<SF_T) in a common laptop or desktop computer. Moreover, it is an heuristic with a constructive phase criteria that could be later randomized and executed in parallel using Monte Carlo simulation, a technique that makes use of random numbers and statistical distributions to solve certain problems.

Figure 9 describes the approach using a flow chart, which is explained hereafter.

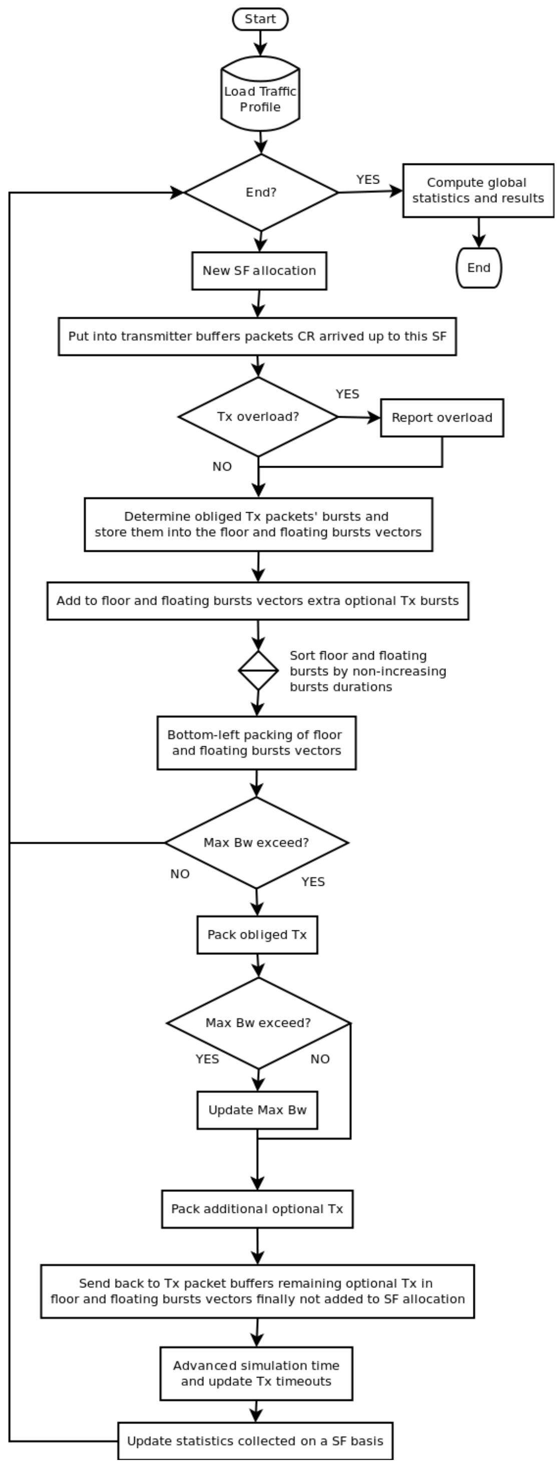

Figure 9: Flow chart describing the overall approach

Each SF_period there is first a stage of collection and aggregation of candidate transmissions from pending CR buffers, with the objective of later minimizing the bandwidth required, but also the signaling required for the transmission of the capacity allocations towards terminals. This first stage is divided in two: the collection of the obliged transmissions first, then the optional transmissions. The obliged transmission bursts are the ones between the current time and the obliged transmission threshold in the transmitters buffers, shown in solid color in Figure 6. If they are not allocated this SF, the PLR due to overload could be exceeded.

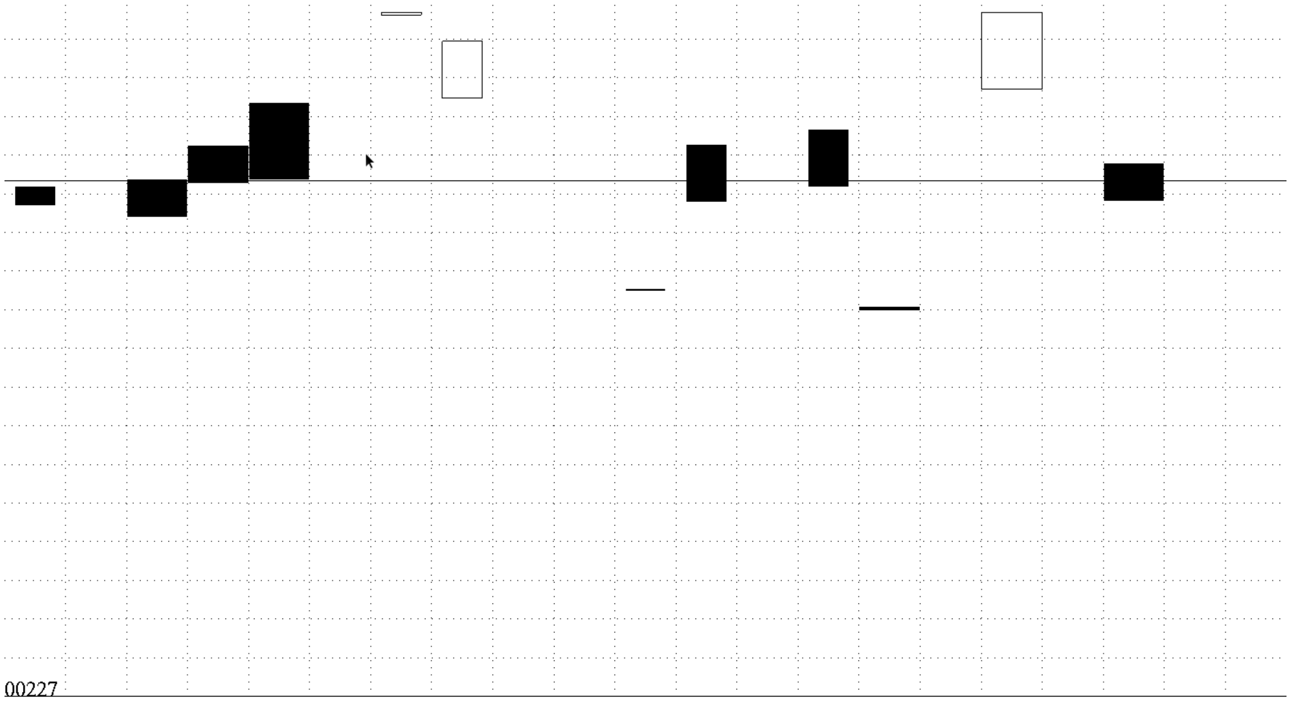

Figure 6: On a given SF period, CR exceeding the obliged Tx threshold are selected (solid color), while optional transmissions can be optionally added (outlined)

Then, the optional transmissions are selected following an EDF scheduling policy. The optional transmissions can be aggregated to the already existing obliged transmission bursts wherever possible, in order to minimize the signaling required for the allocations.

At the end of this first stage of collection and aggregation of candidate transmissions there are two vectors of floor and floating bursts, with both, obliged and non-obliged transmission bursts. The floor bursts vector contains the first transmissions of each terminal on current SF, while the floating bursts vector contains subsequent ones, but which bandwidth is different than previous transmissions. At the end of this first stage of candidate transmissions collection and aggregation, the floor and floating bursts vector are ordered by non-increasing duration of bursts and simply bottom-left packed, as shown in Figure 7.

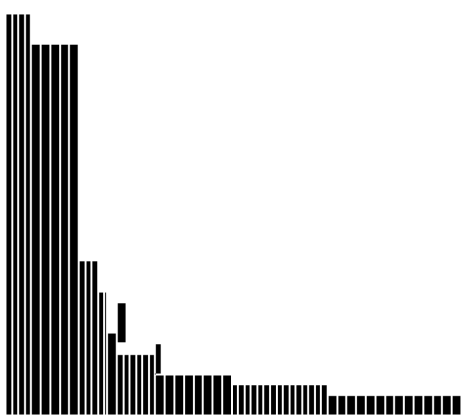

Figure 7: Floor and floating bursts ordered and preliminarily packed

In case this initial simple packing, shown e.g. in Figure 7, exceeds the maximum bandwidth used by all SF up to now or initially configured, the bursts in the floor and floating bursts vectors will be repacked more efficiently using our proposed heuristic, based on the Best Fit packing, which is explained hereafter, in order to obtain a minor value for bandwidth required in available time.

The proposed packing algorithm, is a level based heuristic inspired on the Best Fit packing as explained by Ntene 2007, which pseudocode is shown in section 6, before our proposed algorithm pseudocode.

Our proposed packing algorithm, starts by packing first just the obliged transmissions contained in the floor and floating bursts vectors. After the candidate transmissions collection and aggregation phase, both, obliged and optional candidate transmission bursts are blended in the floor and floating bursts vectors ordered by non-decreasing duration. As the obliged transmissions will be packed first, in order to compute the minimum bandwidth required, it could be that the algorithm is executed faster if the floor and floating bursts vectors are split each one into two vectors, one with obliged transmissions and the other with optional transmissions, instead of doing the searches in the vectors with all bursts (obliged and optional) blended and skipping the optional transmissions in this first step. But this has not been tested and it has not been considered by now, because the algorithm executes at three times the real-time speed when compiled with optimizations (-O3), as a worst case, although it could be a future performance improvement to consider.

Similarly to the original Best Fit packing heuristic, levels are created where bursts are packed, but in our proposal, in order to minimize the signaling required, each level must pack bursts of the same bandwidth.

An slight variation, introduced to simplify the implementation, is also that instead of selecting for packing the item that leaves less residual empty space on any level, the biggest burst (in terms of area) that can be packed on an existing level is selected. If there is not found any candidate burst to be packed on an existing level, the next biggest area burst in the floor bursts vector is selected and packed on a new level. The process continues until all obliged transmissions in the floor and floating bursts vectors are packed into current SF.

As a result we get a minimum bandwidth occupation, which can be higher than the maximum used up to now or not. In case it is higher, the maximum value registered is updated to be reported later as the value to which the system bandwidth should be dimensioned to serve the given traffic profile with the required QoS. In case the maximum bandwidth obtained up to now is exceeded, repacking of bursts with different bandwidths could be considered, but this option has not been implemented due to the complexity it implies and the few expected gain it could provide, according to computational results obtained and shown in section 7.

After obtaining this minimum bandwidth required result, extra packing of bursts can be done in order to occupy the SF as much as possible and get a better system latency and resources utilization efficiency. The packing of additional optional bursts continues as far as there are candidates found that can be packed into existing levels, but, of course, no new levels can be created to pack optional transmissions.

Figure 8 shows an example of an allocation of obliged (filled) and optional (outlined) burst transmissions on a SF, and the remaining empty space.

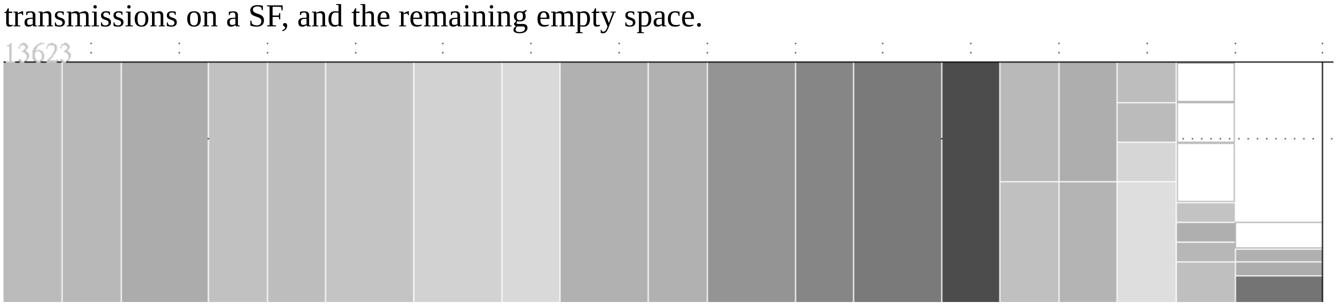

Figure 8: An example of an allocation minimizing both bandwidth required and signaling

Bursts selected as optional transmissions but finally not allocated are sent back to its corresponding packet transmitter queues as if they were never used for any purpose.

Then, the time advances another SF_period and the appeared message bursts transmission needs are positioned in its transmitter queues. The process explained is then repeated for the next to come SF.

The simulation ends when the traffic profile End Of File (EOF) is reached, but bootstrapping of traffic profile could be done in order to generate an arbitrarily long profile (e.g. as long as needed to obtain needed precision on measured statistical results).

Note that the algorithm has been implemented with the purpose of system dimensioning. It is implemented as a procedure to determine the minimum bandwidth required to process a given traffic profile using as much bandwidth as needed. In order to be used on a real system for resources assignment, it should be slightly modified, to perform the packing the same way, but discarding packets when bandwidth initially configured is not enough, instead of increasing it.

6. Pseudo-code

In this section the packing algorithm, presented in section 5 and outlined in Figure 9 steps in case maximum bandwidth is exceeded, is explained in more detail using pseudo-code notation.

As a reference, this is the original BFDH pseudo-code from (Ntene 2007) in which our algorithm is based:

```
Description: Packing a list of rectangles into a strip of fixed width and infinite
height. The list of rectangles is fully specified in advance, before packing
commences.
Input: The number of rectangles to be packed n, the dimensions of the rectangles
w(L i ), h(Li ) and the strip width W.
Output: The height of a packing obtained in the strip.
1: level ← 0; h(level) ← 0; i ← 1; LevelNum ← 1
2: Renumber the rectangles in non-increasing order by height such that h(L 1 ) ≥
h(L2 ) ≥ . . . ≥ h(Ln )
3: Pack rectangle Li left justified at the bottom of the strip; h(level + 1) ←
h(Li )
4: for i = 2, . . . , n do
5:    search all existing levels for the level with sufficient space and has
      minimum residual horizontal space
6:    if such a level exists then
7:          pack rectangle Li left justified
8:    else [there is insufficient space in all existing levels]
9:          create a new level above the top-most level and pack rectangle Li
10:         LevelNum ← LevelNum + 1; level ← LevelNum; h(level) ← h(level - 1) +
h(Li )
11:   end if
12: end for
13: print H = h(level)
```

It has been mentioned that our proposed packing algorithm needs to perform two steps, one to pack the obliged transmissions, which determine the minimum bandwidth required, and a second one to pack additional optional transmissions that can be advanced to maximize packing efficiency and minimize system latency.

Moreover, in order to minimize signaling and interferences and to ease network synchronization in a real system implementation, bursts are packed only in levels with the same bandwidth for already present bursts, otherwise a new level is created.

This is the pseudo-code of our proposed algorithm for allocation of bursts based on the Best Fit packing and satisfying mentioned placement constraints:

Description: This procedure is called when bandwidth used after initial simple ordering of floor and floating bursts and bottom-left packing exceeds maximum bandwidth used up to now. The purpose of this algorithm is to perform a more efficient packing in the available fixed time of SF period, trying to reduce bandwidth used to a value less or equal to the maximum bandwidth used up to now.

The list of bursts to pack has been selected in advance, before packing commences. All the obliged Tx bursts must be packed, then additional optional transmission bursts can be packed to fill the SF remaining space.

The bursts belonging to a single Tx id cannot be sent simultaneously.

Input: The bursts to be packed ordered by decreasing duration and grouped in two vectors:

floorBursts: start at t=0

floatingBursts: start at t>0 because there is a floorBurst that must be transmitted before.

Each burst allocation in the SF is described by its dimensions (bandwidth and duration) and start and end attributes. There is also a nextEnd attribute that indicates whether there is any subsequent burst transmission to pack from the same terminal this burst belongs to.

Other inputs are the SF duration and the current SF maximum bandwidth used, which could be increased just due to obliged transmissions, not to optional ones.

Output: The bandwidth used of a packing obtained in the superframe and the vector of levels and burst allocations performed on each level.

```
1. level ← 0;  LevelNum ← 1;
2. Pack first obliged Tx floor burst (longest duration) floorBurst(0) left justified at the bottom of the strip; h(level) ← h(floorBurst(0))
3. for each obliged Tx burst in the floor and floating bursts vector
4.    search all existing levels for the level with sufficient space to pack the biggest possible area obliged Tx burst in the floor and floating vectors on an existing level gap with the same bandwidth than the candidate burst.
5.    if such a level with such a gap and such a burst exist then
6.         pack found obliged Tx burst in level gap as much bottom-left justified as possible, considering the potential existence of previous and next transmissions from the same terminal.
7.    else [there is insufficient space in all existing levels or no bandwidth match]
8.        create a new level above the top-most level and pack first obliged Tx floor burst (longest duration) or floating burst if there are not more floor bursts.
9.        level ← LevelNum; LevelNum ← LevelNum + 1; h(level) ← h(level - 1) + h(SelectedBurst)
10.   end if
11. end for each
12. print Bandwidth_Used = h(level)
```

7. Computational experiments

The resource allocation process described before has been implemented as a C/C++ console application, GPLv3 licensed and available at SourceForge (Fernández 2012). The heuristic is deterministic and does not make use of any random number generator. The implementation of the packing heuristic pseudo-code shown in previous section has been inspired on Java code from the Burke Best Fit implementation of the Two- Dimensional Loading Capacitated Vehicle Routing Problem (2L-CVRP) problem by Juan 2012, and developed iteratively in a prototype application.

A laptop Toshiba Tecra M10 with an Intel® Core™ 2 Duo CPU at 2.4 GHz , 4 GiB of RAM and running GNU/Linux distribution Ubuntu 10.04, with an x86_64 Linux kernel 2.6.32-43-generic, was used to perform all tests.

In Table 2 are listed the results obtained for each file in the traffic profile if bandwidth is infinite, i.e. a message can be transmitted as soon as it arrives.

File #	Max. Bw Used	SF# Max. Bw	Max. Num. Active Tx
1	20	52356	10
2	32	3569	16
3	247	46014	97
4	114	10357	49
5	43	56976	19

Table 2 – Results for each traffic file considering infinite bandwidth available

The reported figure "Max num of active Tx" refers to the maximum number of terminals with queued packets, not to the maximum number of carriers on a SF. The "Max Bw used" figure units are not Hz, but Bw, the bandwidth of the narrowest burst (in terms of bandwidth) shown in Figure 3.

Table 2 allows comparing each file in terms of traffic volume contained. There are mainly two types of files: #1, #2 and #5, with a lower load, and #3 and #4 with a higher load.

In Table 3 are listed the bandwidth needed values obtained with our packing heuristic for capacity allocation when no packet losses are admitted, and the resulting obliged Tx thresholds considered, which seems that mainly depends on the traffic profile of voice calls present in the traffic file.

File #	Max. Bw Used	SF # Max. Bw	Max. Num. Active Tx	Obliged Tx Threshold (ms)
1	10	16609	34	1080
2	15	20283	65	1200
3	82	42469	1150	3500
4	45	13623	270	12740
5	21	227	39	5165

Table 3 – Results for each traffic file considering limited bandwidth available and PLR=0

It is worth mentioning that the obliged transmission thresholds specified are not the absolute minimum required for each file, but the first found by doing several iterations by manually executing the simulations with different candidate values.

Table 4 compares the resulting needed bandwidth in the case of limiting bandwidth used for PLR=0 to the case of having infinite bandwidth.

File #	Max. Bw Used	Max. Bw Used	Reduction %	Average SF	Max. SF

	(PLR=0, finite Bw)	(infinite Bw)		Resources Utilization %	Resource Utilization %
1	10	20	50	9.05	100
2	15	32	46.88	16.13	100
3	82	247	33.2	47.29	99.59
4	45	114	39.47	38.32	96.3
5	21	43	48.84	16.97	96.83

Table 4 – Bandwidth reduction obtained by our algorithm with respect to simple network overprovisioning (infinite bandwidth available)

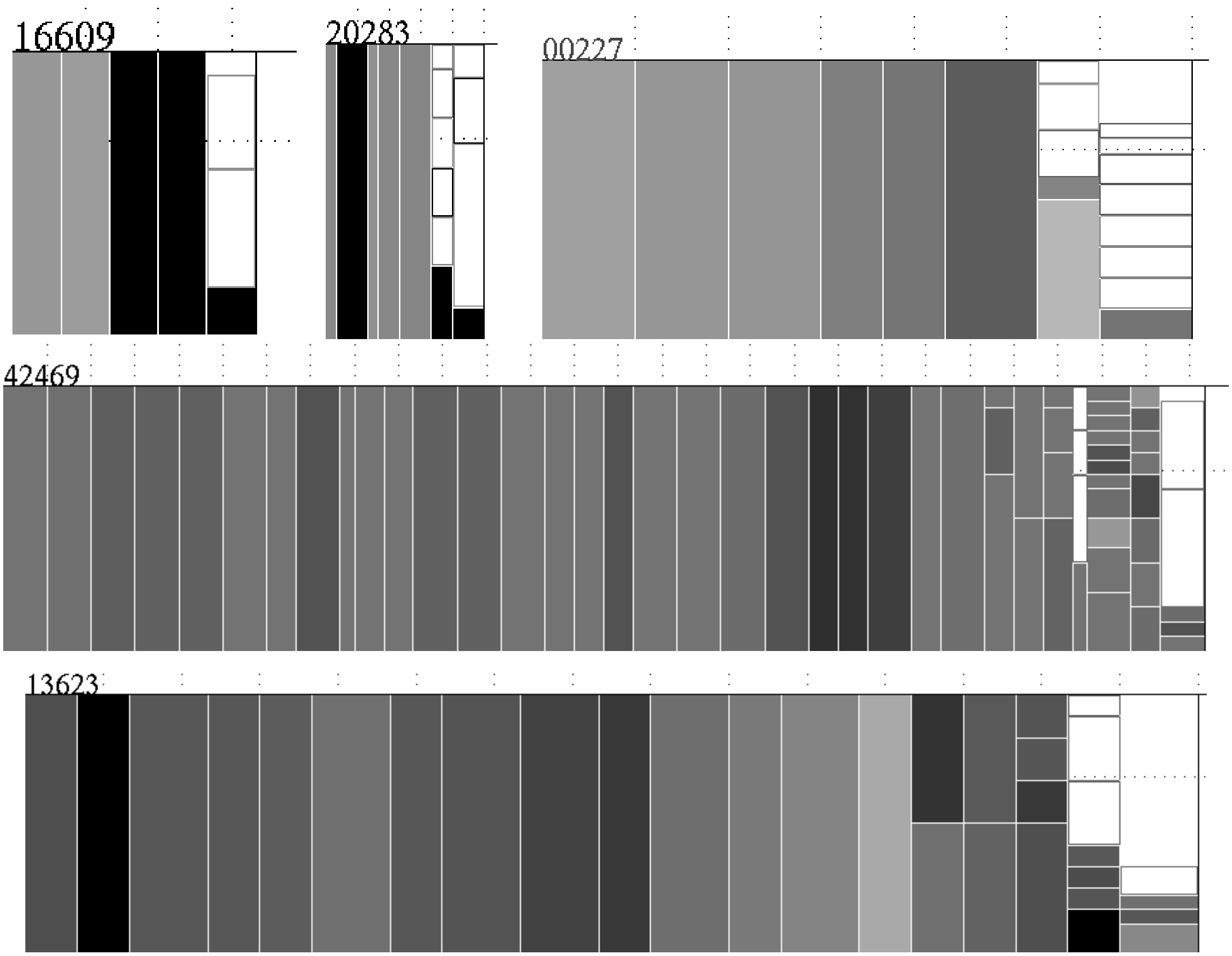

Figure 10: Allocations performed in the SF of maximum bandwidth usage for PLR=0 for files 1, 2, 5, 3 and 4

The console application generates some text files that can be used by an accompanying OpenGL based visualization program in order to plot the packing performed each SF period, as shown in Figure 10, which shows the allocations performed in the SF where the maximum bandwidth needed was reached for each input file, as listed above.

8. Discussion of results

In order to discuss the efficiency of our solution, the packing efficiency, measured as area used vs. area available has been obtained on a SF_T basis for each file and PLR case considered. In the measured efficiency values shown in Figure 11, Figure 12, Figure 13, Figure 14 and Figure 15 hereafter, the minimum resulting bandwidth needed was fixed since the beginning of the software execution.

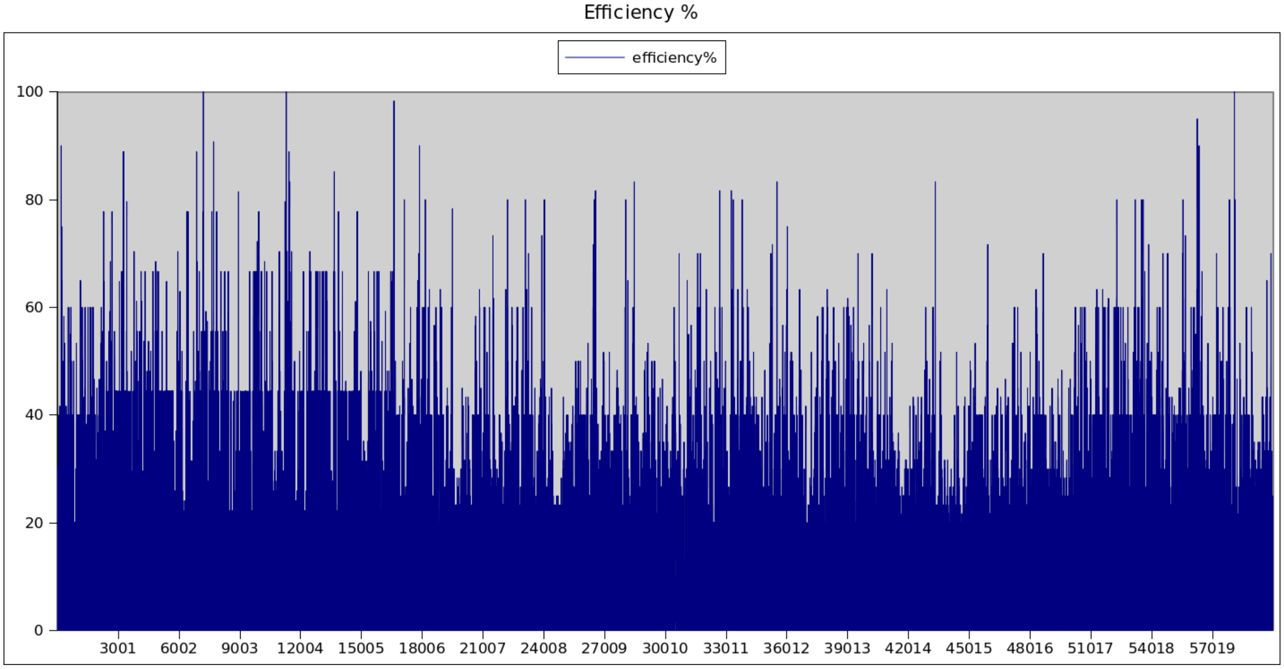

Figure 11: Packing efficiencies measured each SF for file #1 setting SF Bw = 10 since the beginning (target PLR = 0)

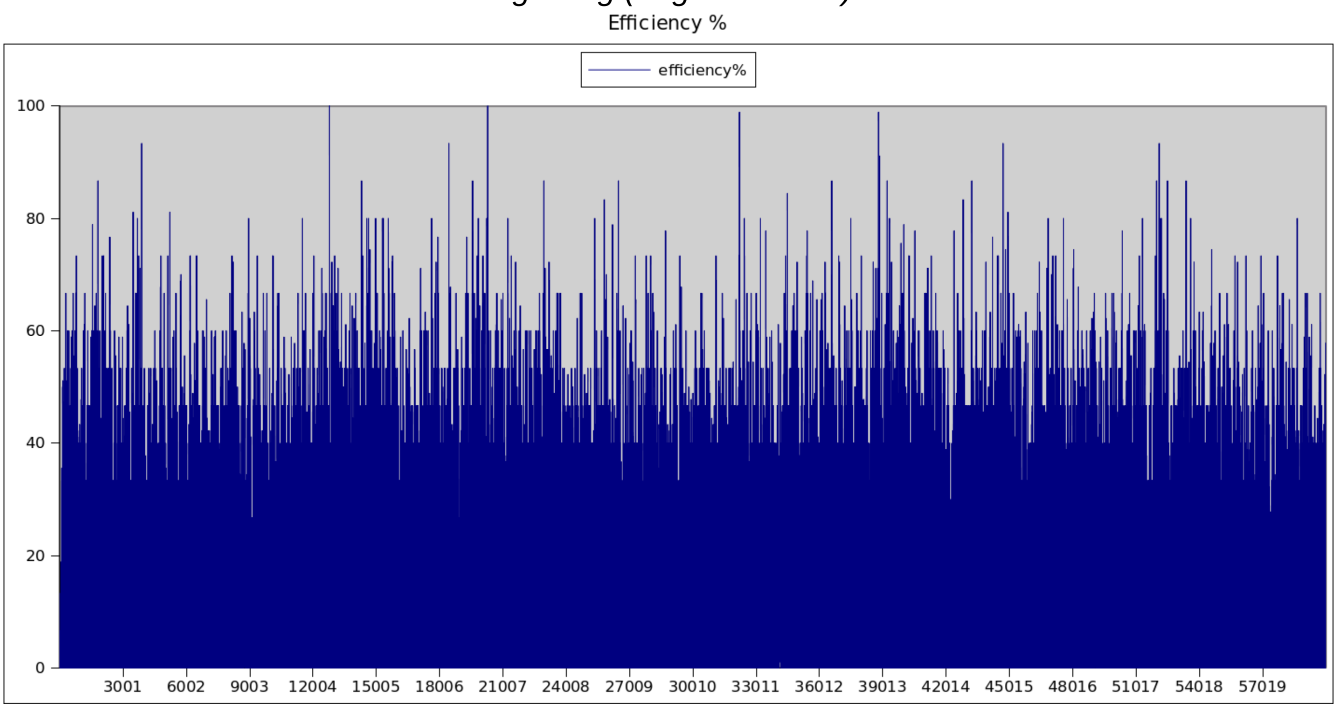

Figure 12: Packing efficiencies measured each SF for file #2 setting SF Bw = 15 since the beginning (target PLR = 0)

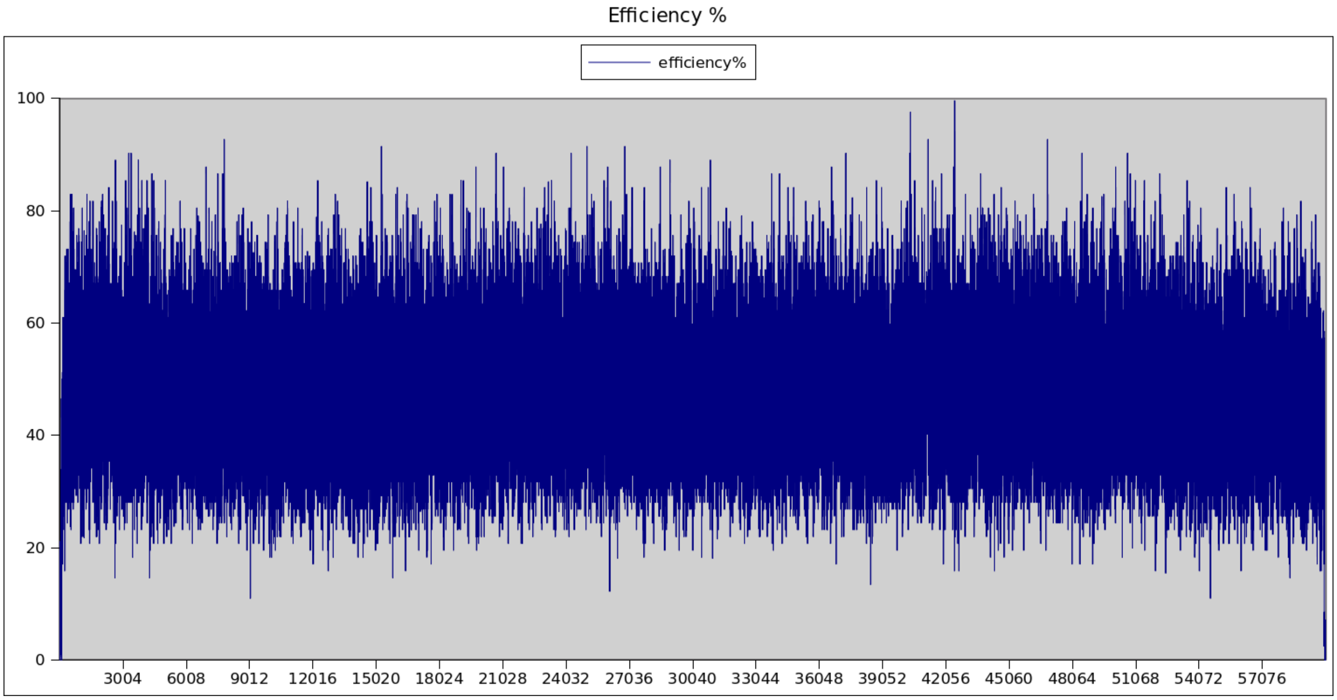

Figure 13: Packing efficiencies measured each SF for file #3 setting SF Bw = 82 since the beginning (target PLR = 0)

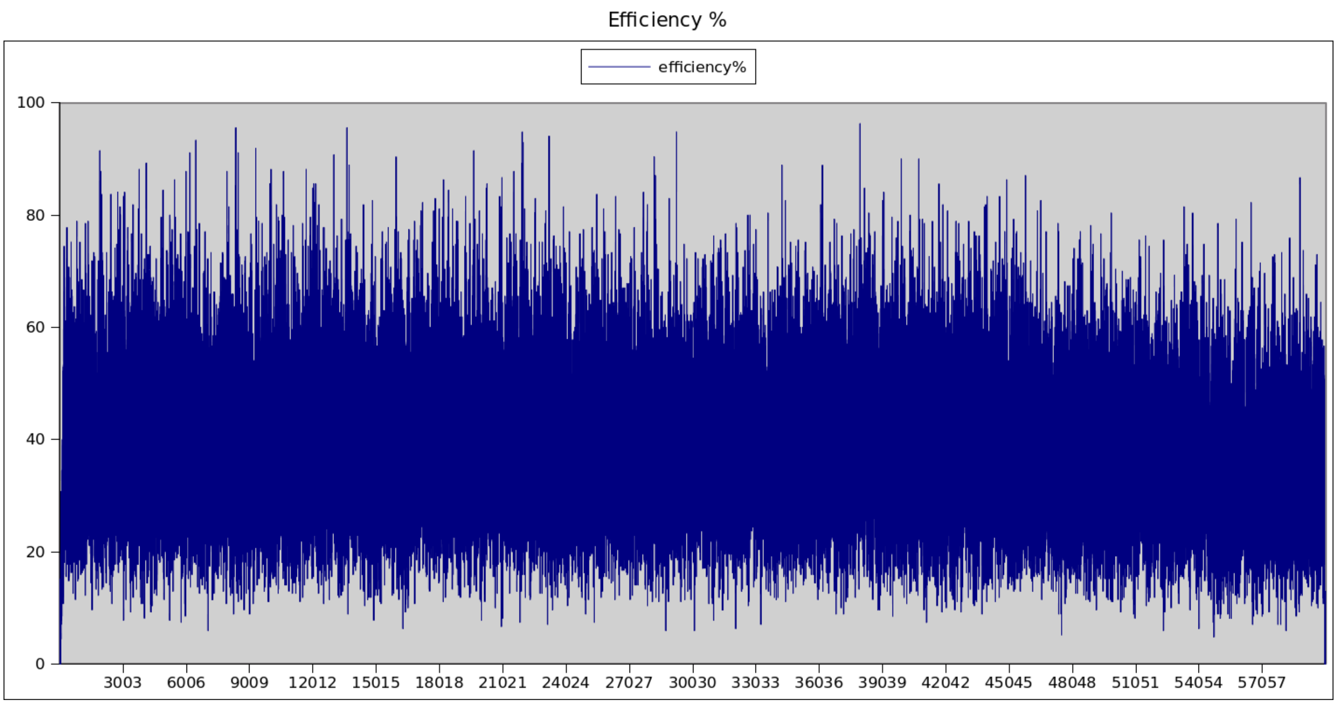

Figure 14: Packing efficiencies measured each SF for file #4 setting SF Bw = 45 since the beginning (target PLR = 0)

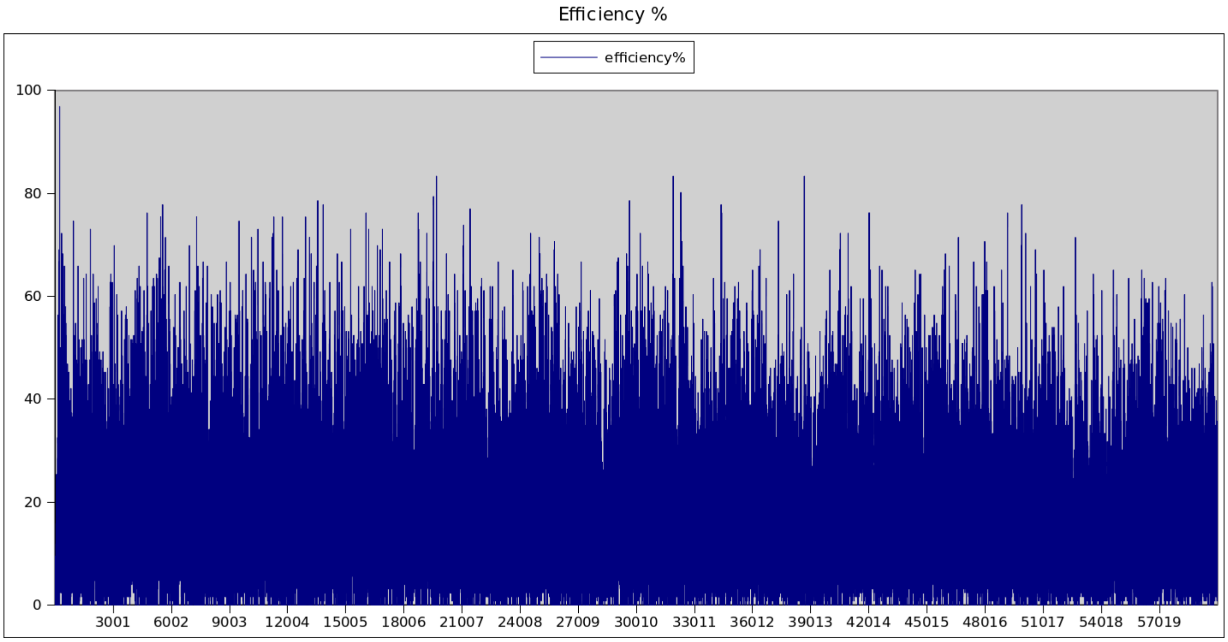

Figure 15: Packing efficiencies measured each SF for file #5 setting SF Bw = 21 since the beginning (target PLR = 0)

As a general conclusion, it is remarkable the fact that > 95% packing efficiency is reached some times in all files. The packing efficiencies and the maximum required bandwidth depend a lot on the considered traffic profile, which is quite bursty, so the average-case efficiency measured is relatively low, but it needs to be this way in order to satisfy the applications required QoS at a few critical moments. The more traffic a file contains the greatest the efficiency is due to the greatest possibility to exploit an statistical multiplexing of traffic.

Table 5 shows a rough estimation of the amount of signaling needed in the forward direction to convey allocations to terminals can be obtained by filtering, e.g. using the serial editor sed (`sed '1~2d' display_file.txt > assignments_file.txt`), the trace files describing the allocations performed each SF period, used by the OpenGL based viewer that has been used to generate the plots of packings each SF period, e.g. Figure 9. The average rate is obtained just dividing the amount of signalling by the six hours that take the traffic traces. The minimum peak rate, can be obtained by diving the longest line size in characters (bytes) - obtained e.g. with `wc -L < filename.txt` - describing the assignments in the files, by the SF_T. Another option is to dimension the signalling channel capacity for the worst case, i.e. a SF filled with the smallest bursts and all from different terminals, to be notified in just an SF_T.

File #	SF Bw at t0	Amount of Signaling	Minimum Peak Signalling Capacity (bits/s)
1	10	2.1 MB (2200448 bytes)	6089
2	15	4.0 MB (4192010 bytes)	10156
3	82	53.2 MB (55786541 bytes)	35045
4	45	28.6 MB (29992332 bytes)	21534
5	21	6.4 MB (6754431 bytes)	9712

Table 5 - Amount of signaling and minimum signalling channel capacity needed in the forward direction to convey allocations to terminals

Moreover to the higher computational time, one of the main cited drawbacks of following an unstructured approach for SF resources assignment is the high amount of signaling needed in order to broadcast the assignments to the terminals, because the division of the frame in carriers can completely change on an SF_T basis. But it must also be considered that the satellite links are quite asymmetric, where the capacity in the forward direction (from ground networks to satellite terminals) is usually much higher than in the return link direction. Whether or not the amount of signaling required by the proposed heuristic is acceptable or excessive will depend on the characteristics of the considered communication network, but it is an important point to take into account in the design trade-offs of the satellite network return and forward links.

Hereafter are compared the results obtained for each file when the obliged transmissions threshold is set to the other values considered in Table 1, which are the lowest possible value, i.e. SF_T and to a value between the minimum and the previously reported values for PLR=0 (see last column of Table 3).

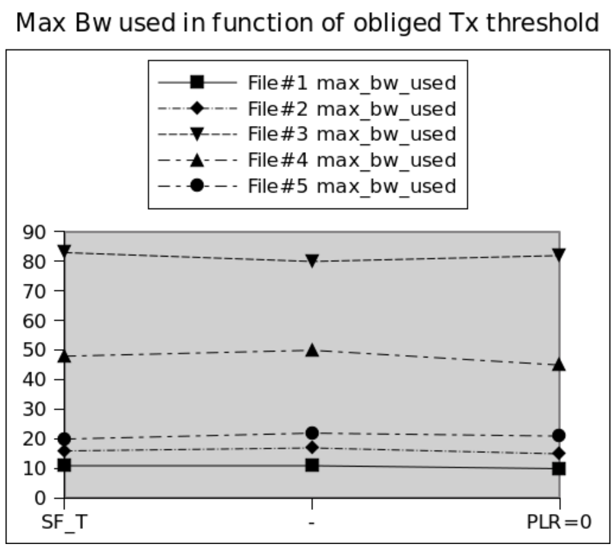

Figure 16: Maximum bandwidth used in function of obliged Tx threshold

Looking at Figure 16, there does not seems to be a dependency between the resulting maximum bandwidth used and the value set for the obliged transmissions threshold. The same can be said of efficiency, according to Figure 17.

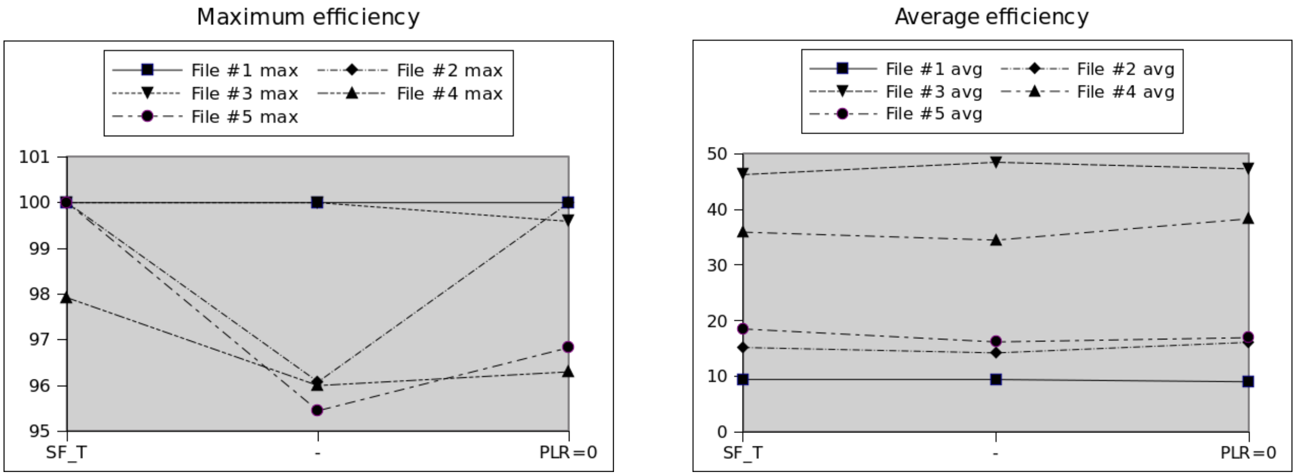

Figure 17: Maximum and average resources utilization efficiency in function of obliged Tx threshold

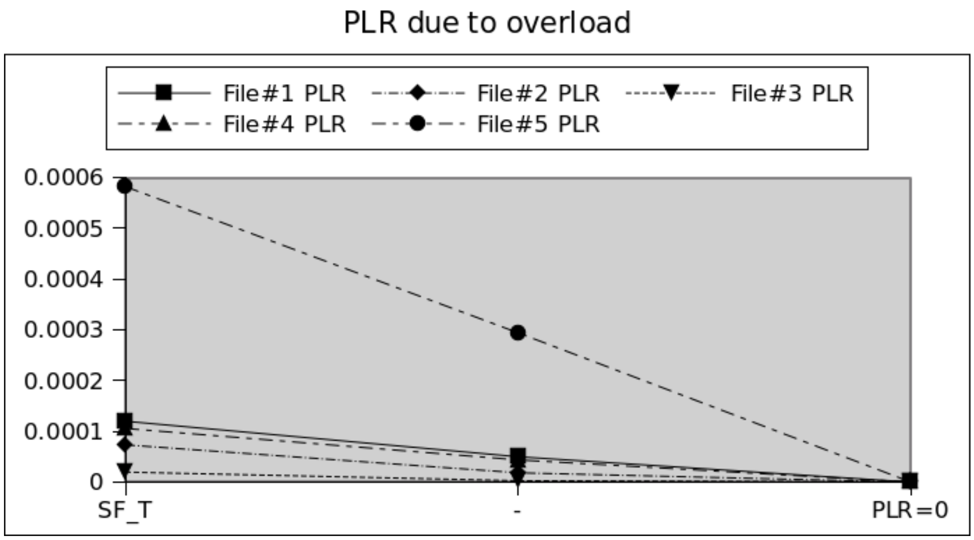

Figure 18: PLR due to transmitter overload in function of obliged Tx threshold

The PLR due to transmitter overload always decreases as the obliged transmission threshold increases, as shown in Figure 18. It has been found, by tracing the software, that the transmitter overloads are due to the expiration of streams of packets that must be transmitted continuously, without any gap time in the transmission, and which arrive at a rate exceeding the maximum transmission rate, e.g. arrival of 60 ms duration bursts each 20 ms.

9. Future Work

As explained in Juan et al (2010), several instances of the proposed packing heuristic, based on the BFDH heuristic could be launched in parallel using different alpha values uniformly distributed in the range (0.10, 0.025). Each heuristic, during the solution-construction process, at the time a new burst and packing level must be selected from the horizontal space remaining sorted list of levels, would apply the following probability distribution for the random variable X = “level k-th is selected at the current step”, where k=1,2,...,s, with s being the current size of the list:

$P(X = k) = alpha \cdot (1 - alpha)^{k-1} + epsilon$, for any k = 1,2,...,s

where

epsilon = 1 - sum(alpha · (1 – alpha)$^{k-1}$, k=1..s)

is a term that assigns a positive probability to every possible step, because of this it is a quasi-geometric distribution.

Notice that if the size s of levels list is large enough, the term epsilon is close to zero, therefore the alpha parameter can be interpreted as the probability of selecting the step with the best match to the BFDH heuristic matching criteria. Instances of the heuristic executed with a low alpha value (e.g. 0.05) consider a large number of levels from the list when selecting the next step, while instances running with a relatively high alpha-value (e.g. 0.35) use a more reduced list of potentially eligible levels.

The randomized versions, each one with its corresponding alpha parameter configured, would select the next step by generating a random number and would get the position of the next step to be selected using a geometric distribution.

The solution using the deterministic version of the heuristic would be generated also in parallel with the randomized versions. It would be used as an upper bound limit of what is considered a good solution in terms of bandwidth utilization.

As described before, our approach could make use of parallel execution of the heuristic to generate a set of random feasible solutions. Each solution can be computed in a few milliseconds using a PC.

10. Conclusions

It has been shown the design and performance of a new heuristic to solve the resource-assignment problem on telecommunications networks for time critical communications, assuming an unstructured approach to the problem. It is based on the BFDH heuristic for 2D packing problems.

This is a novel and innovative approach to the RRM in telecommunication networks in two dimensions. First, because unstructured approaches are frequently discarded in telecommunications literature due to computational cost, in favor of more structured approaches, easier to implement and characterize. But this paper shows that the unstructured approach to the problem is feasible for considered representative traffic profiles, without the need of extraordinary computing resources. On the other hand, it is well-known that innovations usually happen by merging two or more separate topics. This paper algorithm is quite innovative in its approach of solving a telecommunications networks problem from an operations research problem perspective, adapting an existing heuristic to the field of strip packing problems (BFDH) to the telecommunications networks resources management, by establishing an analogy between orders of strips of materials and the need to transmit RF bursts packed on a frame of time and spectrum.

10. References

Açar, G. (2001). “End-to-end resource management in geostationary satellite networks”. Ph.D. dissertation, Dept. Elect. Electron. Eng., Imperial Coll., London, U.K.

Fairhurst, G. (April 2001). “Broadband Satellite Overview”. http://www.abdn.ac.uk/erg/research/future-net/digital-video/bband-sat.html. Retrieved on March 2012.

ETSI (09/2005) EN 301 790 v1.4.1. “Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems”.

ETSI (03/2005) TR 101 790 v1.2.1 “Digital Video Broadcasting (DVB); Interaction channel for Satellite Distribution Systems; Guidelines for the use of EN 301 790 ”.

Roberts, L. G., Lawrence G. (April 1975). "ALOHA Packet System With and Without Slots and Capture". Computer Communications Review 5 (2): 28–42

Shannon, Claude E. (July/October 1948). “A Mathematical Theory of Communication”. Bell System

Technical Journal 27 (3): 379-423.

Indra, UPC GRCM. (10/2010). “ANTARES Communication Standard Design Justification. Annex XX: RRM analyses (return link)”, section 4.3.4, Design considerations with respect to the superframe period. issue 2. Ref.: ANTAR-B1-CP-TNO-2003-IE.

ESA (2011) “ANTARES Project”. http://telecom.esa.int/telecom/www/object/index.cfm?fobjectid=30228. Retrieved on March 2012.

Priscoli, F., Pietrabissa, A. (2004). ”Design of a Bandwidth-on-Demand (BoD) protocol for satellite networks modelled as time-delay systems”. Automatica 40, pp. 729-741.

Pietrabissa, A., Baudoin, C. (2008). “Radio Resource Management for Next Generation DVB-RCS Systems”. IP Networking over Next-Generation Satellite Systems, 2008, Springer, pp. 91 - 101

Lee, K-D., Chang, K-N. (2003). “Throughput-Maximizing Timeslot Scheduling for Interactive Satellite Multiclass Services”. IEEE Communications Letters, Vol. 7, No.6, June, 2003, pp. 263-265

Lee, K-D., Chang, K-N. (2004). “A Real-Time Algorithm for Timeslot Assignment in Multirate Return Channels of Interactive Satellite Multimedia Networks”. IEEE Journal on Selected Areas in Communications, Vol. 22, No.3, April, 2004, pp. 518-528

Mitchell, P. D., Grace, D., Tozer, T. C. (2004). “Burst Targeted Demand Assignment Multiple-Access for Broadband Internet Service Delivery Over Geostationary Satellite”. IEEE Journal on Selected Areas in Communications, Vol. 22, No.3, April, 2004, pp. 546-558

Vázquez-Castro, M.A., Ruggiano M., Ronga,L.S. and Werner, M. (2005). “Uplink Capacity Limits for DVB-RCS Systems with Dynamic Framing and Adaptive Coding”. Proc. 23rd In. Com. Sat. Sys. Conf., Rome, Sep 2005.

Modiano, E. (1997). “Scheduling packet transmissions in a multi-hop packet switched network based on message length”. Proceedings, Sixth International Conference on Computer Communications and Networks, Las Vegas, NV , USA.

Becchi, M. (2008). “From Poisson Processes to Self-Similarity: a Survey of Network Traffic Models”. http://www.cs.wustl.edu/~Jain/cse567-06/ftp/traffic_models1/index.html. Retrieved on April 2012.

Aroumont, A., Radzik, J., Bousquet, M., Castanet, L. (2008). “DVB-RCS Return link Radio Resource Management for Broadband Satellite Systems using Fade Mitigation Techniques at Ka band”. IEEE International Workshop on Satellite and Space Communications (IWSSC) 2008

Morell, A., Seco-Granados, G., Vázquez Castro, M.A. (2008). “Cross-Layer Design of Dynamic Bandwidth Allocation in DVB-RCS”. IEEE Systems Journal, Vol. 2, No.1, March, 2008, pp. 62-73

Booton, R. (2008). “Improving Satellite Bandwidth Utilization by Applying Combinatorial Optimization to the Integrated Waveform (DAMA UHF Satcom)”. IEEE 2008.

Xing, B., Venkatasubramanian, N. (2005). “Multi-constraint dynamic access selection in always best connected networks”. MOBIQUITOUS'05: Proceedings of the The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services. Washington, DC, USA: IEEE Computer Society, 2005, pp. 56-64.

Mariz, D., Cananea, I., Sadok, D., Fodor, G. (2006). “Simulative analysis of access selection algorithms for multi-access networks”. WOWMOM'06: Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks. Washington, DC, USA: IEEE Computer Society, 2006, pp. 219-227.

Chazelle, B. (1983). “The Bottom-Left Bin-Packing Heuristic: An Efficient Implementation”. IEEE Transactions on Computers, Vol. C-32, No. 8, August 1983.

Lesh, N., Marks, J., McMahon, A., Mitzenmacher, M., (2004). “Exhaustive approaches to 2D rectangular perfect packings”. Information Processing Letters, Volume 90, Issue 1, 15 April 2004, Pages 7-14.

Ntene, N. (2007). “An Algorithmic Approach to the 2D Oriented Strip Packing Problem”. Ph. D. dissertation, University of Stellenbosch, South Africa .

Vuuren (2006). 2D Strip Packing Problem Benchmark Data Sets. http://dip.sun.ac.za/~vuuren/repositories/levelpaper/spp%5B1%5D.htm. Retrieved on March 2012.

Imahori, S., Mutsunori, Y., (2007). “The Best-fit Heuristic for the Rectangular Strip Packing Problem: An Efficient Implementation”. Mathematical Engineering Technical Reports, September 2007. Dept. of Mathematical Informatics. Graduate School of Information Science and Technology. University of Tokyo. http://www.keisu.t.u-tokyo.ac.jp/research/techrep/data/2007/METR07-53.pdf Retrieved on May 2012.

Burke EK, Kendall G. & Whitwell G., (2004). “A New Placement Heuristic for the Orthogonal Stock-Cutting Problem”. Operations Research, 52(4), pp. 655–671.

Coffman Jr. EG & Shor PW. (1990). “Average-case Analysis of Cutting and Packing in Two Dimensions”. European Journal of Operational Research, 44, pp. 134–144.

Hopper, E., Turton, B.C.H., (2000). “An Empirical Investigation of Meta-heuristic and Heuristic Algorithms for a 2D Packing Problem”. European Journal of Operational Research 128/1, 34-57.

Hooper, E., Turton, B.C.H., (2001). “A Review of the Application of Meta-heuristic Algorithms to 2D Strip Packing Problems”. Artificial Intelligence Review, 16, 257-300.

Dagli, C. H. & Poshyanonda, P. (1997). “New approaches to nesting rectangular patterns”. Journal of Intelligent Manufacturing 8: 177–190.

Ratanapan, K. & Dagli, C. H. (1997). “An object-based evolutionary algorithm for solving rectangular piece nesting problems”. In Proceedings of the IEEE Conference on Evolutionary Computation, ICEC, 989–994. IEEE, Piscataway, NJ, USA.

Kröger, B. (1995). “Guillotineable bin-packing: A genetic approach”. European Journal of Operational Research 84: 645–661.

Dowsland, K.A. (1993). “Some experiments with simulated annealing techniques for packing problems”. European Journal of Operational Research 68: 389–399.

Leung, T. W., Yung, C. H. & Chan, C. K. (1999). “Applications of genetic algorithm and simulated annealing to the 2-dimensional non-guillotine cutting stock problem”. IFORS ’99. Beijing, China.

Han, G. C. & Na, S. J. (1996). “Two-stage approach for nesting in two-dimensional cutting problems using neural network and simulated annealing”. Proceedings of the Institute of Mechanical Engineers, Part B, Journal of Engineering Manufacture 210(B6): 509–519.

Lodi, A., Martello, S. & Vigo, D. (1999). “Approximation algorithms for the oriented two- dimensional bin packing problem”. European Journal of Operational Research 112: 158– 166.

Blazewicz, J., Hawryluk, P. & Walkowiak, R. (1993). “Using a tabu search approach for solving the two-dimensional irregular cutting problem”. Annals of Operations Research 41: 313– 327.

Healy, P. & Moll, R. (1996). “A local optimisation-based solution to the rectangle layout problem”. Journal of the Operational Research Society 47: 523–537

Pargas, R. P. & Jain, R. (1993). “A parallel stochastic optimisation algorithm for solving 2D bin packing problems”. In Proceedings of the 9th Conference on Artificial Intelligence for Applications, 18–25.

González, S., Juan A. A., Riera, D., Castellà, Q., Muñoz, R., Pérez, A., (2012). “Development and Assessment of the SHARP and RandSHARP algorithms for the Arc Routing Problem”. AI Communications, Volume 25, Number 2/2012

Juan A. A., Faulin J., Ruiz R., Barrios B., Caballè S., (2010). “The SR-GCWS hybrid algorithm for solving the capacitated vehicle routing problem". Applied Soft Computing, vol. 10, no. 1, pp. 215 – 224

Fahey, C. (2007). Tetris. http://www.colinfahey.com/tetris/tetris.html Retrieved on June 2012.

Xu, K., Pillai, P., Hu, Y.F. and Ali, M. (2011) "Interoperability among heterogeneous networks for Future Aeronautical Communication", Future Aeronautical Communications, INTECH Publishing.

Fernández, D. (2012). Traffic Profile Processing. http://sourceforge.net/projects/trafficprofilep/ Retrieved on July 2012.

Teknomo, K. (2006). Bootstrap Sampling Tutorial. http://people.revoledu.com/kardi/tutorial/Bootstrap/ Retrieved on August 2012.

Juan A. A., Domínguez, O., Faulin, J., Agustin, A. (2012). "Multi-Start Algorithm for Solving the Two-Dimensional Vehicle Routing Problem with Items Rotation ", International Transactions in Operational Research.