International Journal of Scientific & Engineering Research, Volume 5, Issue 8,August-2014 627
ISSN 2229-5518

Improved Burst Transmisssion Scheduling for Heterogeneous Devices

Shobhit Shukla

Abstract— In mobile TV applications the encoded video streams of TV channels are transmitted to handheld devices in bursts. This is because the receivers can receive a burst of data and turn off their RF circuits, while they continue to decode the received stream. This maximizes energy saving in these low battery backup devices. Earlier scheduling algorithm has a drawback that it assumed all the receivers to belong to a single class i.e. same resolution. In reality wide range of receivers are available. So, here, another algorithm is proposed which divides each channel into multiple layers, collates those layers and transmits it on a single carrier wave. Thus, every receiver can extract exactly those layers which it is capable of decoding. This Improved Burst Scheduling Algorithm helps maximize energy saving without degrading the quality.

Index Terms— DVB-H, MBMS, NP-Complete, SVC, RF Circuit, Logical Layer, Envelope.

1 Introduction

MOBILE TV is one of the emerging technology which is expected to be very popular in the near future. The problem that lies in broadcasting TV channels to these small handheld devices is its low battery backup. The RF circuits need to be ON to receive data of the TV channels and therefore the power consumption is high. One of the solutions to this problem can be to keep the RF circuits OFF for most of the time. To achieve this channel data has to be sent in bursts.

The main motivation to send channel data in bursts is that, the decoders in mobile devices decode at a much slower rate compared to the bandwidth available in the broadcasting medium. So data is sent considering receiver buffer capacity to avoid overflow. Once this amount of data is received in burst, the devices can switch off RF circuits for sufficiently long time, during which decoding of received data takes place. This process saves energy significantly.

Now a technology is needed which helps us to realize this idea. Digital Video Broadcasting for Handheld (DVB-H) [3] devices is a broadcast system which is derived from the previously used technology DVB-T [7] for terrestrial television. DVB-H is designed especially for low battery backup devices and the protocol stack of DVB-H [5] is shown in Fig.1. Hence this technology is an implementation of the algorithm for power saving and the technique is named as Time Slicing and it is reproduced in Fig.2 for mobile TV networks. The present paper focuses on optimizing this technique.

- *7Shobhit Shukla is currently working as a faculty in Institute of Engineering & Technology (IET), UPTU, Lucknow, U.P., India, PH-+91-8743000507. E-mail: shobhitshukla89@gmail.com*

In earlier algorithm [1] for transmitting TV channels each channel belonged to different predefined classes. This differentiation between channels by dividing them into classes helped in efficiently utilizing channel bandwidth and enhancing viewer experience. Data for every channel was sent in the form of bursts. The bursts were scheduled in a manner such that no two bursts conflict in time, and no receiver buffer overflow/ underflow occur besides more constraints like equal frame length etc. However, in this algorithm the data of all channels has to be transmitted with same bit rate which leads to severe quality degradation in case of heterogeneous devices as the resolution can vary hugely from device to device which are receiving the data simultaneously.

Another paper [4] presents further improvement on this Burst Scheduling algorithm where every channel is divided into layers, each layer can be received separately by the receivers which start displaying the content as soon as the base layer is received. This layering concept helps in dividing the receivers into different classes. The class of devices which has low resolution and hence cannot process high bit rate data can receive lower-bit-rate layers only and continue decoding without any problems. But, this approach is detrimental to energy saving as on the receiver side RF circuit has to be switched ON/OFF frequently due to multiplicity of layers. Moreover, the transmission becomes cumbersome as each layer uses different carrier wave.

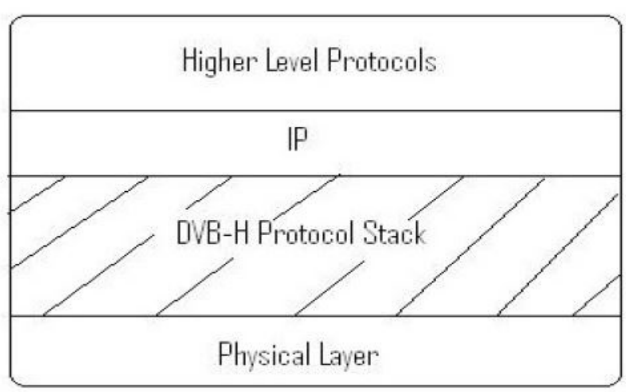

Fig. 1. Protocol Stack of DVB-H

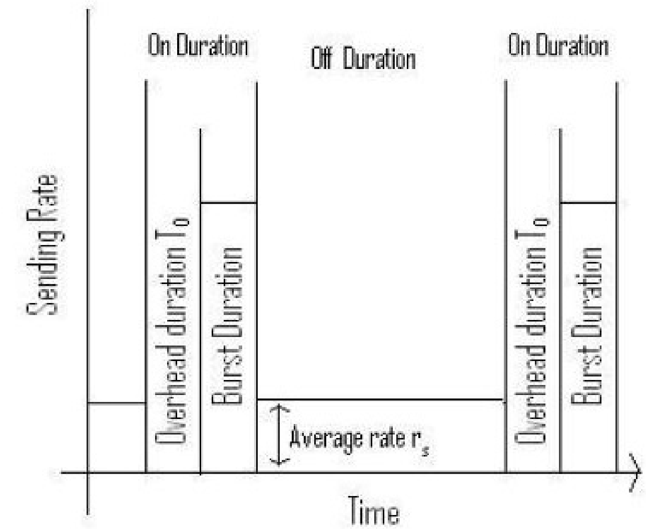

Fig. 2. Time Slicing in Mobile TV

As mentioned before, DVB-H broadcast system is shown in Fig.3. This is a dedicated broadcast network. Traditionally cellular networks were concentrating on unicast transmission. This was fine as long as number of users were limited in an area. But say in an urban area where many users concurrently view same TV content, broadcasting will save bandwidth for

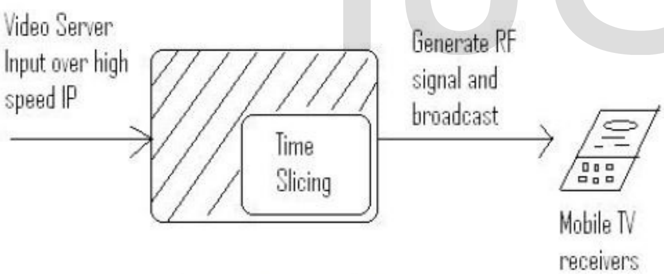

Fig. 3. Modules of Mobile TV broadcasting system

us. The 3G partnership project [2] which works on cellular networks hence proposed Multimedia Broadcast and Multicast Service (MBMS).

2 Related Work

2.1 Problem Definition

Consider S number of TV channels and a wireless medium with bit rate R kbps. Using this medium the base station broadcasts each TV channel s ($1 \leq s \leq S$) whose bit rate is r_s kbps in bursts. Now in order to save energy RF circuit has to be switched ON as minimum number of times as possible on the receiver side. Besides, transmission schedule should be such that maximum energy saving is achieved. So once a burst of data is received, the RF circuits are switched OFF. Also it is switched ON just a small overhead time T_0 before receiving next burst of data to achieve synchronization.

Let energy saved in transmitting the channel s be γ_s. Since there are S channels, we can say that the total average energy saved in broadcasting all S channels is,

$$\gamma = \left(\sum_{s=1}^{S} \gamma_s\right)/S$$

All these scheduling as well as transmission in bursts is done recursively till entire data is transmitted. This individual time window that is repeated is called a frame.

With the above definitions we state the Burst Scheduling [1] problem as, "To find an optimal transmission schedule for bursts of all S TV channels so as to maximize γ." The characteristics of transmission schedule are the following:

i. The number of bursts per channel per frame.
ii. The burst start time x_s , burst end time z_s and burst length y_s.
iii. Ensuring that no two bursts collide.
iv. Absence of buffer overflow or underflow.

2.2 Example

First thing to note when buffer is referred that it is the LINK LAYER [8] buffer. This is because in video streaming systems there are different buffers in each layer, say application layer buffer etc. But in Burst Scheduling Algorithm in reference [1], link layer buffer is considered as we are working on the time

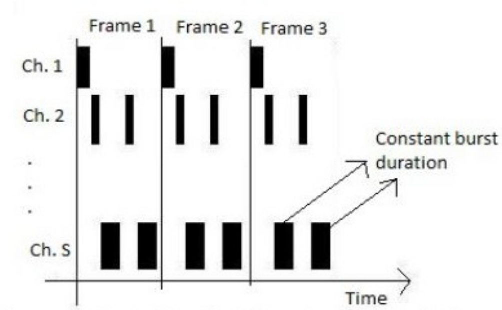

Fig. 4. Burst Scheduling Problem in Mobile TV networks

slicing component which is in this layer itself.

For example consider Fig. 4 in which every channel has bursts of varying sizes compared to others, and no collision of bursts in time occurs ensuring that the common wireless medium is effectively and reliably used. Sometimes, each channel can have more than one burst in each frame so as to avoid buffer underflow/overflow. At the same time, the algorithm is structured such that unnecessarily no extra bursts are scheduled for any channel as it reduces the energy saving metric γ.

ISSN 2229-5518

This algorithm also needs to maintain some constant value for receiver buffer at the beginning of each frame, failing which results in either overflow (if after every frame buffer retains some residue data) or underflow (if it decodes faster than the sending rate) as reproduced in Fig. 5. The rate at which receiver decodes the data is much slower, represented by the slope in Fig. 5 as $-r_s$, whereas the rate increases during a burst at the rate of $R-r_s$.

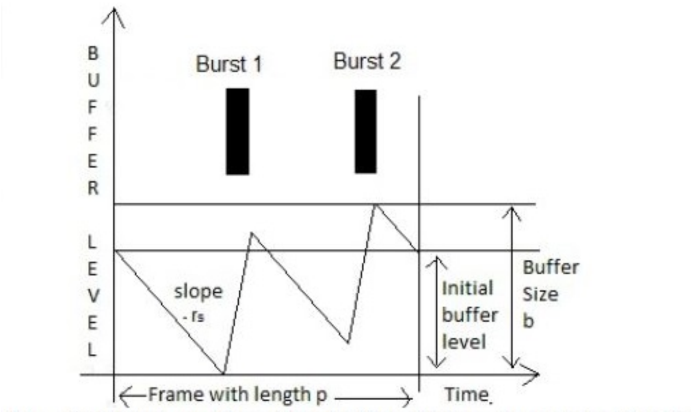

Fig. 5. Effect on Receiver buffer during successive bursts

2.3 Burst Scheduling is NP-complete

Deduction : To maximize energy saving, the number of bursts in each frame of length p have to be reduced. For this following parameters and assumptions are considered:

i. Maximized Energy Saving = Reduced RF circuit ON time.
ii. RF circuit ON time = burst time + overhead time T_0.
iii. A constant burst time is assumed as we consider steady schedules.

Therefore T_0 is the only parameter which can be reduced to minimize the RF circuit ON time which can be obtained by reducing number of bursts.

This problem i.e. "Minimizing bursts in frame of length p is NP-complete" is proved below.

It should be recalled that **The Task Sequencing Problem** which is known to be NP-complete [9] resembles the problem of minimizing bursts in a frame. The similarities in the two problems are summarized as follows.

Task sequencing problem has T tasks released at time x_t with length y_t and deadline z_t. Here a non-preemptive machine schedule has to be designed. Coming to our Minimizing bursts problem, number of channels can be mapped to number of tasks, i.e. S = T, $x_s=x_t$, $y_s=y_t$, $z_s=z_t$ where x is respective burst start time, y is burst length, and z is maximum time before which another burst of same channel has to be sent failing which results in buffer underflow. Using these x_s,y_s,z_s other parameters of our problem are defined like r_s is bit rate of TV channel s, u_s initial buffer level which has to remain constant at the beginning of each frame for a given Link Layer buffer b. A solution can be easily found using brute force approach in Polynomial time. But designing any better algorithm is currently considered as a near impossible task because it is equivalent to finding solution to non-preemptive task sequencing problem which is NP-complete. Hence we conclude that this problem is NP-hard.

Also this problem includes other constraints like collision free schedule and ensuring that no buffer violations occur, thereby proving that the Burst Scheduling problem is a NP-complete problem.

An important point to be made clear at this juncture is that though our problem looks like preemptive task scheduling problem[10], it is not so. The difference is that we are interested in reducing total number of bursts, whereas preemption simply increases the number of bursts.

2.4 Burst Scheduling is Algorithm

As of date finding a solution to NP-complete problem is not possible, so some assumptions about the bit rate of channels are made. This leads to the point where the total number of channels can be divided into certain standard classes, which are the practical round offs of the actual channel bit rates. This rounding off concept is makes the algorithm different from other existing algorithms which instead employ a procedure of generalizing all channels into one standard bit-rate. As explained earlier this results in very poor viewer satisfaction. With this assumption as base an overview of algorithm is presented and then analyzed with an example.

2.4 Overview of Algorithm

The channels are sorted in ascending order of bit rates. Then the channel s_1 will have the lowest bit rate r_1. All the other channels will have bit rate in multiples of this base bit rate. i.e. for any channel s, $r_s=2^i*r_1$, where i=0,1,2,.... An assumption about the wireless medium bit rate is made that $R=2^k*r_1$, for any positive integer k.

The optimal frame length can be written as $p^*=b/r_1$. Hence we divide each frame, into bursts of length p^*r_1, which guarantees no overflow of buffers. Therefore there are $(p^*R)/(p^*r_1)=R/r_1$ bursts in each frame. Every channel s is allocated r_s/r_1 bursts in each frame. Inter burst distance of $p^*/(r_s/r_1)$ seconds is maintained between bursts of same channel to avoid buffer underflow. With the schedule ready for different channels only consideration is to see to it that these scheduled bursts are conflict free.

The algorithm reproduced on the next page initially creates a node for each channel, and inserts it in a priority queue which is implemented as a binary heap. These are the leaf nodes of the binary tree about to be formed. To build a binary tree two nodes are taken recursively from the priority queue

ISSN 2229-5518

with lowest key values and an internal node is created. The key values are assigned to different nodes based on their bit rates. i.e. r_s/r_1. This merging of leaves into internal nodes continues till a tree of size $log(R/r_1)$ is obtained. Note that if $\sum_{s=1}^{S} r_s=R$, then the medium is fully utilized and hence the frame will be filled with computed bursts.

Each leaf of the tree obtained represents a key value, allocated to each channel. But at this point it is not known that which burst should take which slot in the frame. So one has to start from root and number the branches. The left branch is numbered 0 and right branch 1. From each leaf we trace back and note the value we get, which is the schedule for that particular channel node. Also if the channel is at a depth of k levels lesser than height of the tree then we prefix the schedule with k number of "x". Here x means either 0 or 1. i.e. say we have schedule for a channel as x10. It means that channel is at a depth of k=1 level lesser than the height of tree and it is allocated bursts 010 and 110.

Algorithm 1 Burst Scheduling

Compute optimal frame length $p^* = b/r_1$
Allocate a leaf node l for every channel s, let $l.ch = s$
Push all leaf nodes to a priority queue P with key r_s/r_1
while true **do**
  m1= pop_min(P), m2= pop_min(P);
  if m_2 is null or m1.key $<$ m2.key **then**
    if m_2 is not null **then**
      Push(P, m_2);
    end if
    Allocate a dummy node m_2, where m_2.key = m_1.key
  end if
  Create an internal node n with children n.left and n.right
  let n.left = m_1, n.right = m_2
  let n.key = m_1.key + m_2.key
  push(P,n)
  if n.key $\geq R/r\_1$ **then**
    break;
  end if
end while
if $|P| > 1$ **then**
  return ϕ
end if
let $T = \phi$
for all leaf node l **do**
  for $i = 0$ **to** l.key - 1 **do**
    start=offset + i x $(R/r_1)/l.key$; insertBurst(T, start x $p * r_1/R$,l.ch);
  end for
end for

2.5 Example for Burst Scheduling Algorithm

Let us take a wireless medium with bandwidth R=2048 kbps, and the buffer size b=1 Mb. Also let there be four TV channels categorized into 2 classes with channel s_1 representing base class r_1=256 kbps. Let channel s_2 and channel s_3 bit rate be r_2=r_3=512 kbps and s_4 has bit rate r_4=1024 kbps. We will now choose the optimal frame length, $p^*=b/r_1$, which is p^*=1024/256=4 seconds. Now this frame can accommodate R/r_1=8 bursts. Hence each channel 1 to 4 will have bit rates r_s/r_1 i.e. 1,1,1,4 respectively and one burst will be wasted. Hence in the 6th burst (110) no burst is transmitted but at the receivver end the RF ciruit switches ON which will lead to energy wastage.

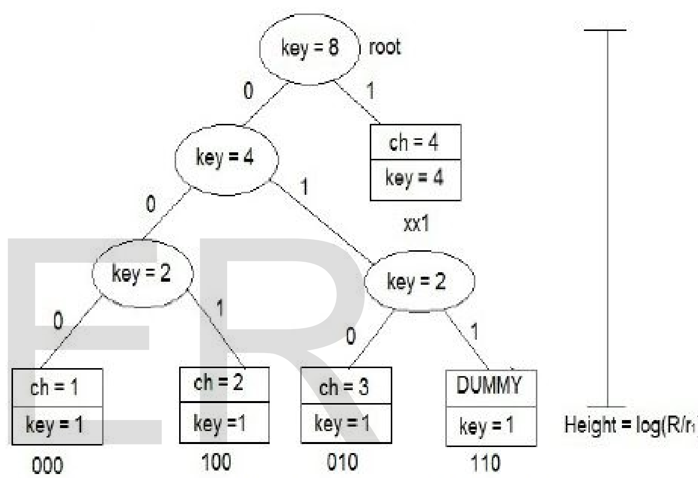

Fig. 6. Example of Burst Scheduling Algorithm

Look at Fig. 6 shown above, where there are 4 leaf nodes in the tree for this example. The schedules are calculated as explained before and written below each leaf node. The root node has key value R/r_1=8, which is the aggregation of total number of bursts. Using the slots computed for each channel the complete schedule is built as:

T = (0,1),(.5,4),(1,3),(1.5,4),(2,2),(2.5,4),(3,DUMMY),(3.5,4)

where in each set 1st element represents the start time for this particular burst and the 2nd element is the channel number. The current practice however is to use a uniform bit rate across all channels. We have discussed its disadvantages, but this algorithm can be extended to if required by taking each r_s=2^i*r_1 and i=0. Also note that as said before the burst where no channel burst is transmitted is represented using DUMMY node in the figure whose key value is equivalent to its left sibling.

ISSN 2229-5518

3 Our Improvements

On proper examination of the above burst scheduling algorithm, though it appears to have solved the major problems of Mobile TV broadcasting, it has an inherent drawback. It assumes that the transmission is being done to the homogeneous devices. But the fact is that the system actually transmits channels to a wide variety of devices with different capabilities.

Another related paper [4] discusses about transmission to heterogeneous devices. When the H.264 video compression standard was extended by providing a new feature - Scalable Video Coding (SVC) [3], it had an impact on TV Broadcast Systems as well. The Scalable Video Coders can be used to divide the individual channels into multiple layers, based on types of devices we intend to support, and individual bursts are required for transmitting corresponding layer's data. So if there are m channels and n layers in each channel then *m*n* bursts are required in every frame for effective transmission as reproduced in Fig.7 [4].

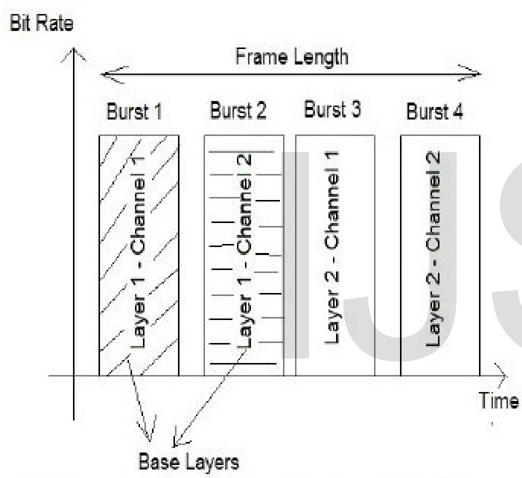

Fig. 7. Transmission Schedule for Multi Layer Streams

Here m=2 and n=2, we get 4 bursts (m*n) in one frame. Also note the shaded bursts, which are bursts carrying layer 1 of every channel. This is an important layer, also called as base layer. On receiving this layer the devices can start displaying the video content. There after any additional layers can be added based on the required quality enhancement and decoding capability of the device. Here the problem of transmitting to heterogeneous devices has been solved. So is everything working fine? The answer is still no. There is a flaw in the multi-layering approach too. Here one has to transmit TV channels belonging to different classes on a different carrier wave therby increasing the complexity of scheduling and the RF circuit also has to be kept ON most of the time to receive different layers, thus reducing energy saving. Therefore, our task now is to theorize an approach which efficiently takes care of both TV channel classes and support for heterogeneous devices.

We propose a schedule by modifying Burst Scheduling algorithm, where to support heterogeneous devices each channel is divided into layers and then collating these layers into a single envelope which will be transmitted on the network.

This improved burst scheduling algorithm is shown Algorithm 2.

Algorith

```
Divi
Defi
Defi
for :
  in
end
Senc
Con
Allo
Pusl
whil
```

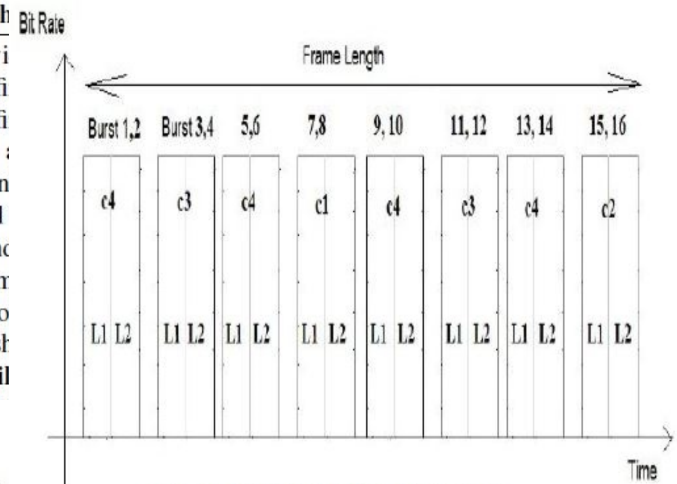

Fig. 9. Improved Burst Scheduling

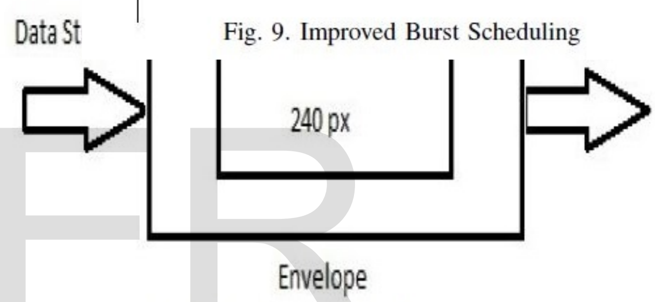

Fig. 8. Creating an Envelope

```
      break;
  end if
end while
if |P| > 1 then
  return φ
end if
let T = φ
for all leaf node l do
  for i = 0 to l.key - 1  do
    start=offset + i x (R/r1)/l.key ; insertBurst(T, start
    x p * r1/R,l.ch);
  end for
end for
RF scans header of envelope and accepts it else calls
error().
while true do
  Call Decode() for decoding the selected envelope.
  Match for the nearest resolution.
  Buffer and display.
  if error() or END() then
    break;
  end if
end while
```

As shown in Algorithm 2, the data stream is first divided into logical layers based on layer resolution. These layers are then collected as input set known as envelope which contains the contiguous packets for each layer. This envelope is then sent on the network and the Burst Scheduling Algorithm 1 is applied to these envelopes. After the envelopes are scheduled and sent to the receivers then the RF circuit on the receiver end will scan the header of these envelopes and will get switched OFF. If the header matches with the desired channel then the device accepts the envelopes else it calls the error function which will display the error to the user. After accepting the envelope the receiver decodes the envelope and selects the layer with the nearest resolution match and displays it.

Fig. 8 shows the process of envelope creation. Here we consider two layers with resolutions of 240 and 360 pixels respectively. The data stream is divided into these logical layer and these layers are collected in an envelope as contiguous packets. This envelope is then sent to the network for scheduling and transmitting.

The schedule generated by our improved approach as shown in Fig. 9, consists of collating bursts of different layers belonging to same channel and transmitting them. This approach provides us with two benefits. Firstly, it allows the receiver to switch ON/OFF its RF circuit for a minimum number of times to receive the bursts of the respective channel and thus resulting in energy saving at receiver-end. Secondly, it also allows the sender to send the different layers of the same channel on a single carrier wave instead of different carrier waves for different layers therby resulting in tremendous ease of scheduling.

4 RESULTS

In this section we will list out the results of the various experiments done on our improved burst scheduling algorithm and compare with the results of the same experiments on the existing burst scheduling algorithm[6].

1) Energy Saving experiment 1

This experiment assumes the bandwidth of medium to be 2 MHz and bit rates of 4 channels to be transmitted are 250,250,500 and 1000kbps respectively and buffer size increases from 500 to 3000kb in steps of 500kb. The result of this experiment comparing the BSA [1] and improved BSA algorithms is shown in Fig. 10. As seen from fig. 10, the energy saving is much more using improved BSA. For smaller buffer size i.e. 500kb the energy saving obtained using improved BSA is 85% whereas for BSA it is only 70%. For larger buffer sizes i.e. above 500 to 3000kb the energy saving obtained using improved BSA rises to 92% whereas for BSA the rise is up to 84% only. Thus, for this experiment, the improved BSA facilitates additional energy saving of 9.5%.

2) Energy Saving experiment 2

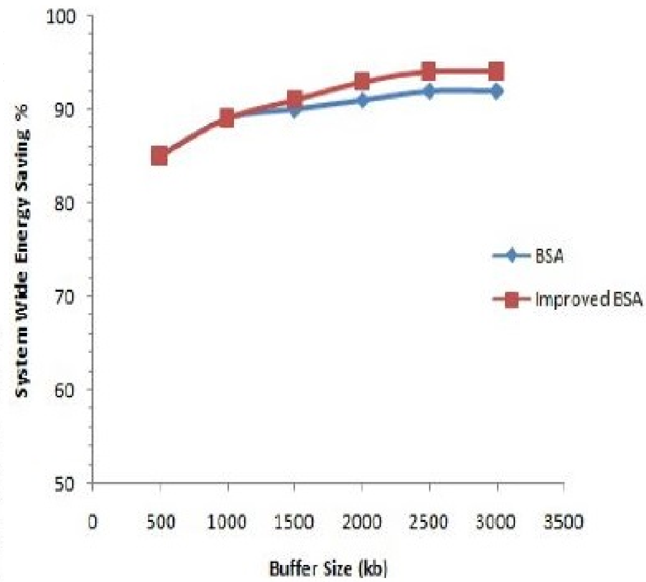

Fig. 12. Energy Saving Experiment 3

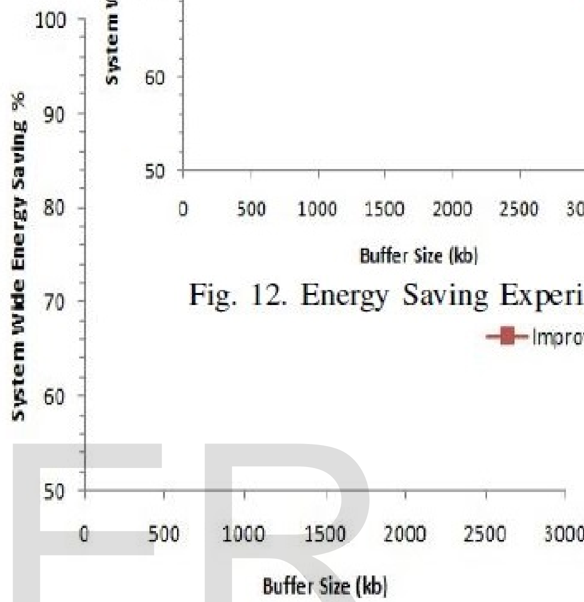

Fig. 11. Energy Saving Experiment 2

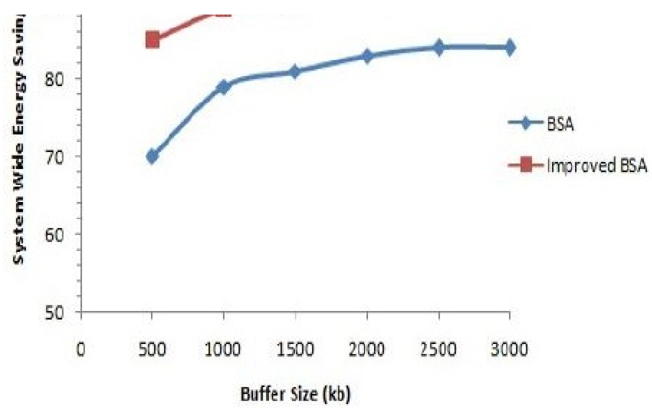

Fig. 10. Energy Saving Experiment 1

BSA. For smaller buffer size i.e. 500kb the energy saving obtained using improved BSA is 85% whereas for BSA it is only 77%. For larger buffer sizes i.e. above 500 to 3000kb the energy saving obtained using improved BSA rises to 92% whereas for BSA the rise is up to 88% only. Thus, for this experiment, the improved BSA facilitates additional energy saving of 4.5%.

ISSN 2229-5518

3) Energy Saving experiment 3

This experiment assumes the bandwidth of medium to be 2 MHz and bit rates of 4 channels to be transmitted are 250,250,250 and 250kbps respectively and buffer size increases from 500 to 3000kb in steps of 500kb. The result of this experiment comparing the BSA [1] and improved BSA algorithms is shown in fig. 12. As seen from Fig. 12, the energy saving is much more using improved BSA. For smaller buffer size i.e. 500kb the energy saving obtained using improved BSA and BSA are same i.e. 85%. For larger buffer sizes i.e. above 500 to 3000kb the energy saving obtained using improved BSA rises to 94% whereas for BSA the rise is up to 92% only. Thus, for this experiment, the improved BSA facilitates additional energy saving of 2.2%.

From the above experiments, it is observed that the energy saving using improved BSA is higher when the transmitted channels have widely different bit rates. Thus, we have been successful in obtaining additional energy saving while transmitting channels of different bit rates using a single carrier wave.

5 Program Screenshots

In this section we will list out the screenshots of the improved Burst Scheduling Algorithm which we implemented.

i. Input Variables: The input variables which include the medium bandwidth, buffer size, the number of TV channels to transmit, the number of devices to support and the individual bit rates of all channels are given in the Fig. 13.

ii. New Parameters: The input variables which include the medium bandwidth, buffer size, the number of tv channels to transmit, the number of devices to support and the individual bit rates of all channels are recomputed for scheduling purpose and shown in the fig. 14.

Fig. 14. Recomputed Input Variables

iii. Generated Schedule: The schedule generated by the improved Burst Scheduling Algorithm is shown in the Fig. 15.

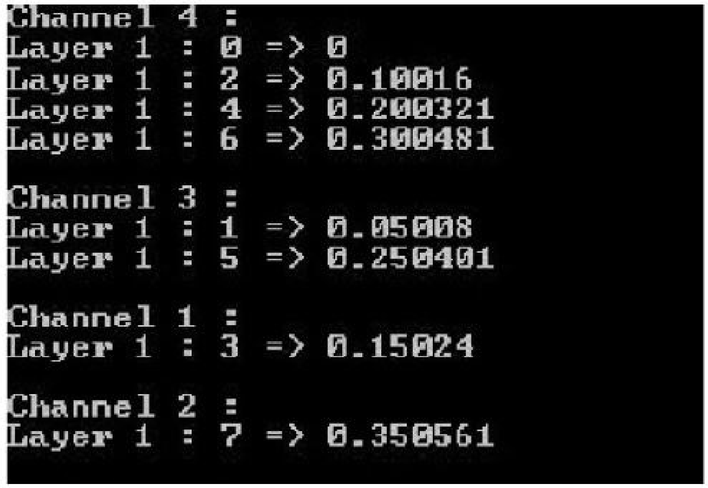

Fig. 15. Schedule generated by improved BSA

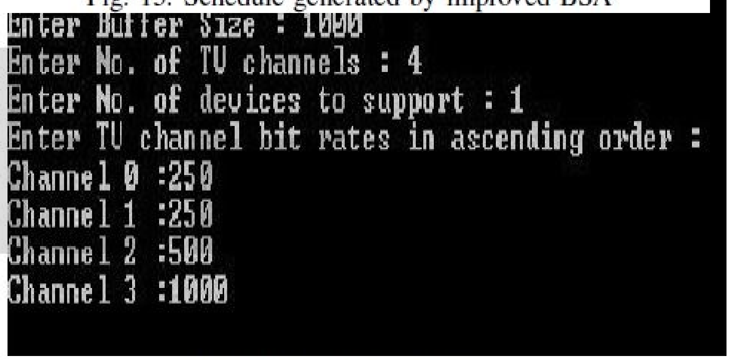

Fig. 13. Input Variables

iv. Energy Saving Achieved: The energy saving for a single device for each channel is shown in the fig. 16.

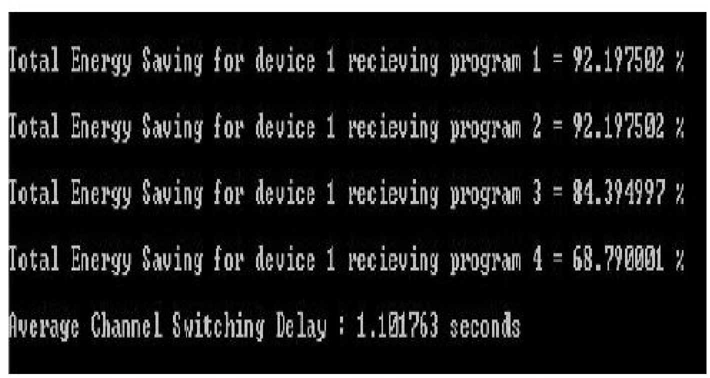

Fig. 16. Energy Saving by improved BSA

6 Conclusion

The Improved Burst Scheduling algorithm presented in

ISSN 2229-5518

this paper takes care of practical scenarios like heterogeneous devices with variable bit rate TV channels as well as maximizing energy saving on the receiver-end. As the bit rates of the transmitted channels vary widely the energy saving obtained using Improved BSA increases considerably as compared to BSA existing in literature. As the scheduling problem is NP-complete, the bit rate of channels is divided into specific classes. This can be used to provide service to users satisfying their requirements at reasonable cost along with better viewing experience. The multi-layered BSA described in reference 4 uses distinct carrier wave for each layer whereas the present Improved BSA uses only a single carrier wave to transmit all the layers thereby leading to tremendous ease of scheduling besides maximizing the energy saving. This could be obtained as the improved BSA permits the RF circuit to be switched OFF most of the time while receive different layers.

REFERENCES

[1] Mohammed Hafeeda, Member IEEE, "On Burst Transmission Scheduling in Mobile TV broadcast networks" IEEE transactions on Networks, April 2010.

[2] Frank Hartung, Uwe Horn "Delivery of broadcast services in 3G networks" IEEE transactions on broadcasting, vol 53, no.1, march 2007.

[3] "Digital Video Broadcasting - Handheld (DVB-H) home page," 2008, http://www.dvb-h.org/.

[4] " Multi Layer Video Broadcasting with Low Channel Switching Delays ", Cheng-Hsin Hsu and Mohammed Hafeeda.

[5] G. May, â€œThe IP Datacast system - overview and mobility aspects, in Proc. of IEEE International Symposium on Consumer Electronics(ISCE04), Reading, UK, September 2004, pp. 509514.

[6] Testbed and experiments for mobile TV (DVB-H) networks In MM ’08: Proceeding of the 16th ACM international conference on Multimedia (2008), pp. 995-996, doi:10.1145/1459359.1459546 by Mohamed Hefeeda, Cheng H. Hsu, Yi Liu.

[7] Digital Video Broadcasting (DVB); transmission system for handheld terminals (DVB-H). European Telecommunications Standards Institute (ETSI) Standard EN 302 304 Ver. 1.1.1, November 2004.7

[8] M. Kornfeld, Optimizing the DVB-H time interleaving scheme on the link layer for high quality mobile broadcasting reception, in Proc. of IEEE International Symposium on Consumer Electronics (ISCE07), Dallas, TX, June 2007, pp. 16.

[9] M. Garey and D. Johnson, Computers and Intractability: A Guide to The Theory of NP-completeness. W. H. Freeman, 1979.

[10] U. Schwiegeishohn, â€œPreemptive weighted completion time scheduling of parallel jobs, in Proc. of European Symposium on Algorithms(ESA96), Barcelona, Spain, September 1996, pp. 3951.