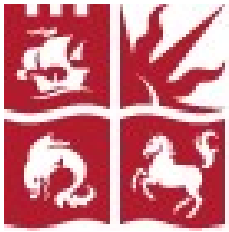
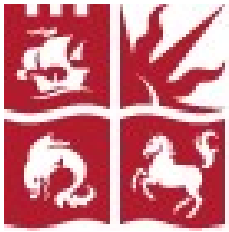


EENGM4221: Broadband Wireless Communications

Lecture 21: WiMax Service Classes and Scheduling

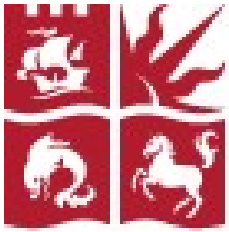
Dr Simon Armour

Service Classes (1)



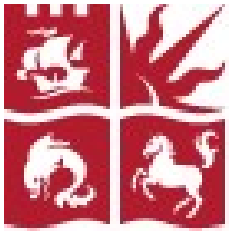
- 802.16 defined four different service classes to accommodate the different QoS needs of different services that the network may support. These are:
  - Unsolicited Grant Service (UGS)
  - Real Time Priority Service (RTPS)
  - Non-Real Time Priority Service (NRTPS)
  - Best Effort Service (BES)
- Subsequently, a fifth was added in 802.16e
  - Extended Real Time Polling Service (ERTPS)

Service Classes (2)



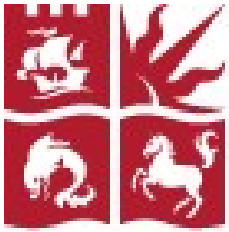
- Unsolicited Grant Service (UGS) is for Constant Bit Rate Services. The BS automatically schedules a fixed fraction of resource for this service without the service having to continually request it – it requests it once
  - This is very similar to Bluetooth's SCO connection
  - SSs may piggyback requests for extra resource onto UGS data packets and the BS may respond to these as it sees fit
  - This approach only really works for services generating fixed size packets on a regular basis
    - Under such conditions it is very efficient since no dedicated resource requests need to be exchanged between SS and BS once the service is set up

Service Classes (3)



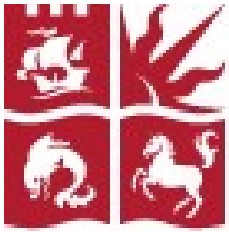
- Real Time Priority Service (RTPS) caters for non-CBR real time traffic
  - The BS does not automatically schedule resources for RTPS services
  - Instead, the BS schedules regular opportunities in the Uplink phase for SSs to request resources for their RTPS services
  - This creates more resource request overhead than UGS but is more adaptable to a service's needs

Service Classes (4)



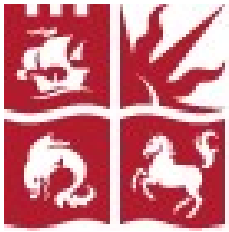
- Non-Real Time Priority Service (NRTPS) caters for non-CBR non-real time traffic
  - Works similarly to RTPS but also allows nodes to contend to use TDMA slots for resource requests
  - This enables the BS to schedule fewer resources for resource requests but makes the availability of resource to NRTPS services less reliable

Service Classes (5)



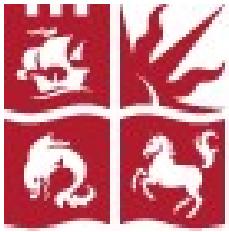
- Best Effort Services (BES)
  - No resource or resource request slots are scheduled by the BS
  - SSs may only obtain resource by contending to transmit resource request in spare UL Data frames
  - BES gets the worst deal

Service Classes (6)



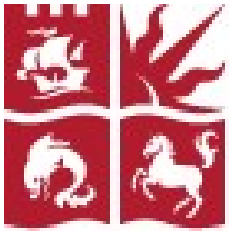
- Extended Real Time Polling Service (ERTPS)
  - Uses a similar approach to UGS of scheduling regular resource to a service but the service can use this resource either for data transmission or for resource requests
  - This accommodates time variant traffic requirements

802.16 Scheduling Principles - BS



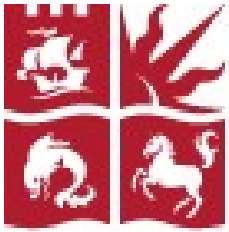
- The BS knows the needs of all downlink services and schedules appropriate resources for them into the current Downlink TDM phase if possible or the current Downlink TDMA phase if necessary
- The BS knows the basic needs of Uplink UGS services and schedules appropriate resources for them in the current Uplink TDMA phase
- The BS knows what resource requests it received in the previous frame and tries to allocate resources accordingly in the current frame in an attempt to provide best possible QoS
- The BS knows how many RTPS and NRTPS services it is supporting and schedules appropriate resources for the corresponding resource requests in the current Uplink TDMA phase

802.16 Scheduling Principles - SSs



- SSs receive the UL MAP and determine:
  - The resources allocated to them for data
  - The resources allocated to them for resource requests
  - The unallocated resources available which can be contended for
- SSs transmit data in the resources allocated to them for data and resource requests in the resources allocated to them for resource requests
- Any SSs with any remaining un-transmitted resource requests contend in free Uplink TDMA slots and try to transmit their requests
  - If they win, they know the BS is aware of their needs and will try to meet them in the next MAC frame
  - If they lose, they have to try again in the next MAC frame
  - Contention is based on a binary exponential backoff algorithm, similar to 802.11

GPC and GPSS SSs (1)

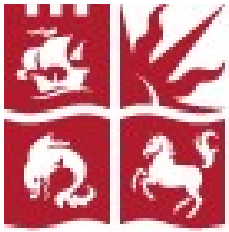


- Nodes in a network may be required to support multiple connections (services)
- The source node will have detailed knowledge of the current needs of the services it supports but the BS has the task of dividing resource between nodes
- For the downlink, things are simple. The BS knows the downlink needs of the services it supports and has complete control of the downlink schedule
- For the uplink, things are more difficult since the SSs know most about the uplink needs of the services they support but they do not have control of the schedule. Detailed communication of this need back to the BS adds overhead
- 802.16 allows two ‘classes’ of SS in terms of how the resource allocation between connections is achieved. GPC and GPSS.

Ref:

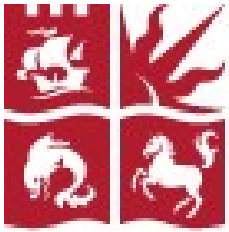
Warning: Acronym Overload!

GPC and GPSS SSs (2)



- Grant Per Connection SSs require the BS to tell them what resources are allocated to each connection they support
  - All scheduling is thus done in the BS; the SS can be fairly dumb in scheduling terms – it just does as its told
  - This generates some overhead since detailed scheduling information must be communicated from BS to SS
- GPSS SSs require only that the BS tell them their total resource allocation; they then divide this resource themselves between the connections they support
  - The resource allocation becomes two tier with an initial block allocation of resource made by the BS and a subsequent detailed allocation made by the GPSS SS
  - The SS must be more intelligent
  - Schedule information overhead is low
  - The SS has more flexibility. Combining this with more detailed, local knowledge about the needs of the services (connections) they support should enable a clever SS to achieve better uplink QoS provision
- GPSS is now the accepted norm in WiMax

Review of Lecture 21



- We reviewed the five traffic classes that 802.16 uses to facilitate good QoS provision
- We discussed the broad principles of scheduling facilitated by the standard; scheduling algorithms are vendor specific
- We discussed the option for more and less ‘intelligent’ SS in terms of the scheduling that they perform

Ref: