

Schedulability analysis of flows scheduled with FIFO: Application to the Expedited Forwarding class

Steven Martin¹ and Pascale Minet²

¹LRI, Paris-Sud University
91405 Orsay, France
steven.martin@lri.fr

²INRIA Rocquencourt
78153 Le Chesnay, France
pascale.minet@inria.fr

Abstract

In this paper, we are interested in real-time flows requiring quantitative and deterministic QoS (Quality of Service) guarantees. We focus more particularly on two QoS parameters: the worst case end-to-end response time and jitter. We consider a FIFO (First In First Out) scheduling of flows. The FIFO scheduling is the simplest one to implement and very used. We first establish a bound on the worst case end-to-end response time of any flow in the network, using the trajectory approach. We present an example illustrating our results. Finally, we show how to apply these results to the EF (Expedited Forwarding) class in a DiffServ (Differentiated Services) architecture.

1. Context and motivations

In this paper, we are interested in real-time applications that require bounds on the worst case end-to-end response times and jitters to have a behavior compliant with their specifications (e.g. voice over IP and control-command applications). That is why we focus on deterministic guarantees of end-to-end response times and jitters in a packet network. We will show how to determine these times depending on the flow scheduling used in the network. In this paper, we show how to apply a worst case analysis to the distributed case, using the trajectory approach.

These results can be applied in a DiffServ (*Differentiated Services*) architecture to determine the worst case end-to-end response time granted to flows belonging to the EF (*Expedited Forwarding*) class, assuming that routes of EF flows remain fixed, once assigned.

2. Problematic

We investigate the problem of providing a deterministic guarantee (i.e. an upper bound) on the end-to-end response time to any flow in a network. As we make no particular assumption concerning the arrival times of packets in the network, the feasibility of a set of flows is equivalent to meet the requirement, whatever the arrival times of the packets in the network. We assume that time is discrete. Indeed, results obtained with a discrete scheduling are as general as those obtained with a continuous scheduling when all flow parameters are multiples of the node clock tick [1]. Moreover, we assume the following models.

2.1. Models

Scheduling model All nodes in the network schedule packets according to the FIFO algorithm.

Network model We consider a network where links interconnecting nodes are supposed to be FIFO and the network delay between two nodes has known lower and upper bounds: L_{min} and L_{max} . Moreover, we consider neither network failures nor packet losses.

Traffic model We consider a set $\{\tau_1, \dots, \tau_n\}$ of n sporadic flows. Each flow τ_i follows a fixed path¹ \mathcal{P}_i that is an ordered sequence of nodes whose first node is the ingress node of the flow. Moreover, a sporadic flow τ_i is defined by:

- T_i , the minimum interarrival time (abusively called period) between two successive packets of τ_i ;
- C_i^h , the maximum processing time on node h of a packet of τ_i . By convention, $C_i^h = 0$ if $h \notin \mathcal{P}_i$;

¹For that purpose, we can use source routing or MPLS.

- J_i , the maximum release jitter of packets of τ_i at its ingress node. A packet is subject to a release jitter if there exists a non-null delay between its generation time and the time where it is taken into account by the scheduler;
- D_i , the end-to-end deadline of τ_i , that is its maximum end-to-end response time acceptable. A packet of τ_i generated at time t must be delivered at $t + D_i$.

2.2. Notations

We consider any flow τ_i , $i \in [1, n]$, following a path \mathcal{P}_i . We focus on the packet m of τ_i generated at time t .

Definition 1 Let m be the packet of flow τ_i generated at time t . Let m' be the packet of flow τ_j generated at time t' . On any node $h \in \mathcal{P}_i \cap \mathcal{P}_j$, priority of packet m is higher than or equal to this of packet m' if and only if m arrives before m' on node h .

We also denote:

- τ_i , a sporadic flow of the set $\{\tau_1, \dots, \tau_n\}$;
- R_i , the worst case response time of flow τ_i ;
- $\tau(g)$, the index number of the flow which packet g belongs to;
- m , the packet of flow τ_i generated at time t ;
- $W_{i,t}^h$, the latest starting time of m on node h ;
- $\mathcal{P}_i = [first_i, \dots, last_i]$, the path followed by flow τ_i , with $first_i$ (resp. $last_i$) the first node (resp. the last node) visited by τ_i in the network;
- $|\mathcal{P}_i|$, the cardinal of path \mathcal{P}_i , that is the number of nodes visited by flow τ_i ;
- $pre_i(h)$, the node visited by τ_i just before node h ;
- $suc_i(h)$, the node visited by τ_i just after node h ;
- $first_{j,i}$, the first node visited by flow τ_j on path \mathcal{P}_i ;
- $last_{j,i}$, the last node visited by flow τ_j on path \mathcal{P}_i ;
- $slow_i$, the slowest node visited by τ_i on path \mathcal{P}_i :
 $\forall h \in \mathcal{P}_i, C_i^{slow_i} \geq C_i^h$;
- $slow_{j,i}$, the slowest node visited by τ_j on path \mathcal{P}_i :
 $\forall h \in \mathcal{P}_i \cap \mathcal{P}_j, C_j^{slow_{j,i}} \geq C_j^h$;
- $S_{min_i}^h$, the minimum time taken by a packet of flow τ_i to go from its source node to node h ;
- $S_{max_i}^h$, the maximum time taken by a packet of flow τ_i to go from its source node to node h ;
- $\forall a \in \mathbb{R}, (1 + \lfloor a \rfloor)^+$ stands for $\max(0; 1 + \lfloor a \rfloor)$;
- $M_i^h = \sum_{h'=first_i}^{pre_i(h)} (\min_{\substack{j \in [1, n] \\ first_{j,i}=first_{i,j}}} \{C_j^{h'}\} + L_{min})$.

Figure 1 illustrates the notations of $first_{i,j}$, $first_{j,i}$, $last_{i,j}$ and $last_{j,i}$ when (1) flows τ_i and τ_j are in the same direction and (2) flows τ_i and τ_j are in reverse directions.

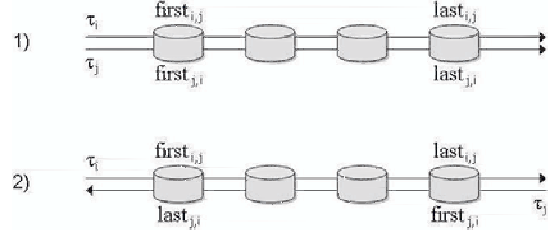


Figure 1. $first_{i,j}$, $first_{j,i}$, $last_{i,j}$ and $last_{j,i}$

Moreover, we assume, with regard to flow τ_i following path \mathcal{P}_i , that any flow τ_j following path \mathcal{P}_j , with $\mathcal{P}_j \neq \mathcal{P}_i$ and $\mathcal{P}_j \cap \mathcal{P}_i \neq \emptyset$, never visits a node of path \mathcal{P}_i after having left this path.

Assumption 1 \forall flow τ_i following path \mathcal{P}_i , \forall flow τ_j following path \mathcal{P}_j such that $\mathcal{P}_j \cap \mathcal{P}_i \neq \emptyset$, we have either $[first_{j,i}, last_{j,i}] \subseteq \mathcal{P}_i$ or $[last_{j,i}, first_{j,i}] \subseteq \mathcal{P}_i$.

To achieve that, the idea is to consider a flow crossing path \mathcal{P}_i after it left \mathcal{P}_i as a new flow. We proceed by iteration until meeting Assumption 1.

Definition 2 The end-to-end jitter of any flow τ_i , $i \in [1, n]$, is the difference between the maximum and minimum end-to-end response times of τ_i packets, that is equal to: $R_i - (\sum_{h \in \mathcal{P}_i} C_i^h + (|\mathcal{P}_i| - 1) \cdot L_{min})$.

3. Related work

Deterministic and quantitative guarantees can be provided by at least three approaches, that compute the worst case end-to-end response time of any flow:

- **The holistic approach** [2, 3]. This approach, the first introduced in the literature, considers the worst case scenario on each node visited by a flow, taking into account the maximum possible jitter introduced by the previous visited nodes. The minimum and maximum response times on a node h induce a maximum jitter on the next visited node $h + 1$ that leads to a worst case response time and then a maximum jitter on the following node and so on. This approach can be pessimistic as it considers worst case scenarios on every node possibly leading to impossible scenarios. Indeed, a worst case scenario for a flow τ_i on a node h does not generally result in a worst case scenario for τ_i on any node visited after h .

- **The network calculus approach** [4]. Network Calculus is a powerful tool recently developed to solve flow problems encountered in networking. Indeed, considering a network element characterized by a service curve and all the arrival curves of flows visiting this element, it is possible to compute the maximum delay of any flow, the maximum size of the waiting queue and the departure curves of flows. Results of such analysis are deterministic, provided that the arrival and service curves are deterministic. As bounds are generally used instead of the exact knowledge of the arrival and service curves, this approach can lead to an overestimation of the bounds on the end-to-end response times.

- **The trajectory approach.** This approach considers the worst case scenario that can happen to a message along its trajectory: the sequence of nodes visited. This approach is described in the next section.

4. Worst case analysis:

The trajectory approach

Unlike the holistic approach, the trajectory approach [5] is based on the analysis of the worst case scenario experienced by a packet on its trajectory and not on any node visited. Then, only possible scenarios are examined. For instance, the fluid model (see [6] for GPS) is relevant to the trajectory approach. More precisely, we consider any flow τ_i , $i \in [1, n]$, following a path \mathcal{P}_i consisting of q nodes numbered from 1 to q . We focus on the packet m of τ_i generated at time t .

With the FIFO scheduling, the processing of a packet can no longer be delayed after it has started. That is why we compute the latest starting time of m on its last node visited. To achieve that, we adopt the trajectory approach. This approach consists in moving backwards through the sequence of nodes m visits, each time identifying preceding packets and busy periods that ultimately affect the delay of m .

4.1. Latest starting time computation

To compute the latest starting time of packet m , we proceed as follows. We first determine the busy period² of level corresponding to the priority of m in which m is processed on node q . Let bp^q this busy period. We define $f(q)$ as the first packet processed

²A busy period of level \mathcal{L} is defined by an interval $[t, t']$ such that t and t' are both idle times of level \mathcal{L} and there is no idle time of level \mathcal{L} in (t, t') . An idle time t of level \mathcal{L} is a time such that all packets with a priority greater than or equal to \mathcal{L} generated before t have been processed at time t .

in bp^q with a priority higher than or equal to this of packet m . As flows do not necessarily follow the same path in the network considered, it is possible that packet $f(q)$ does not come from node $q - 1$. We then define $p(q - 1)$ as the first packet processed between $f(q)$ and m such that $p(q - 1)$ comes from node $q - 1$. Packet $p(q - 1)$ has been processed on node $q - 1$ in a busy period of level corresponding to the priority of $p(q - 1)$. Let bp^{q-1} this busy period. We then define $f(q - 1)$ as the first packet processed in bp^{q-1} with a priority higher than or equal to this of $p(q - 1)$. And so on until the busy period, on node 1, of level corresponding to the priority of packet $p(1)$ in which the packet $f(1)$ is processed (see Figure 2).

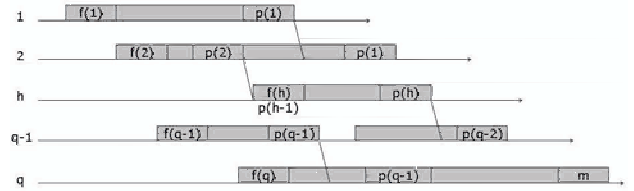


Figure 2. Response time of packet m

For the sake of simplicity, on a node h , we number consecutively the packets processed after $f(h)$ and before $p(h)$ (with $p(q) = m$). Hence, on node h , we denote $m' - 1$ (resp. $m' + 1$) the packet preceding (resp. succeeding to) m' . Moreover, we denote $a_{m'}^h$ the arrival time of packet m' on node h and consider the arrival time of packet $f(1)$ in node 1 as the time origin ($a_{f(1)}^1 = 0$).

By adding parts of the busy periods considered, we can express the latest starting time of packet m in node q , that is: the processing time on node 1 of packets $f(1)$ to $p(1)$ + L_{max} + the processing time on node 2 of packets $f(2)$ to $p(2)$ + L_{max} - $(a_{p(1)}^2 - a_{f(2)}^2)$ + ... + the processing time on node q of packets $f(q)$ to $(m - 1)$ - $(a_{p(q-1)}^q - a_{f(q)}^q)$.

We can notice that on any node $h \in \mathcal{P}_i$, if there exists no flow τ_j such that $h = first_{j,i}$, then $p(h - 1) = f(h)$ and so $a_{p(h-1)}^h - a_{f(h)}^h = 0$. In other words, if $p(h - 1) \neq f(h)$, there exists a flow τ_j such that $h = first_{j,i}$. In such a case, by definition of $p(h)$, all the packets in $[f(h), p(h - 1))$ cross path \mathcal{P}_i for the first time at node h . We can then act on their arrival times. Postponing the arrivals of these packets in the busy period where $p(h - 1)$ is processed would increase the departure time of m from node q . Hence, in the worst case, $p(h) = f(h + 1)$ on any node $h \in \mathcal{P}_i$. Thus, we get:

$$W_i^q(t) = \sum_{h=1}^q \left(\sum_{g=f(h)}^{f(h+1)} C_{\tau(g)}^h \right) - C_i^q + (q-1) \cdot L_{max}.$$

We now evaluate the maximum delay incurred by packet m due to packets with a priority higher than or equal to this of m . This delay is equal to: $X_{i,t} = \sum_{h=1}^q (\sum_{g=f(h)}^{f(h+1)} C_{\tau(g)}^h) - C_i^q$. By definition, for any node $h \in [1, q]$, $f(h+1)$ is the first packet with a priority higher than or equal to this of m , processed in bp^{h+1} and coming from node h . Moreover, $f(h+1)$ is the last packet considered in bp^h . Let us show that in this sum, if we count packets processed in bp^h and bp^{h+1} , only $f(h+1)$ is counted twice.

Lemma 1 *For any flow τ_j , if there exists a node $h \in \mathcal{P}_i$ with a packet $m' \in (f(h), f(h+1))$, then for any node $h' \in \mathcal{P}_i - \{h\}$, $m' \notin (f(h'), f(h'+1))$.*

Proof: We consider that flows τ_i and τ_j are in the same direction. Indeed, when flows are in reverse directions, the proof is obvious.

We first show that if $\exists h \in \mathcal{P}_i$ such that any packet m' of flow τ_j is processed in $(f(h), f(h+1))$ on node h , then there is no node $h' \in \mathcal{P}_i$ with h' visited after h by τ_j such that $m' \in (f(h'), f(h'+1))$. Indeed, by definition, m' arrives before $f(h+1)$ on node h . As packets are scheduled FIFO, m' leaves node h before $f(h+1)$. Moreover, links being FIFO, m' arrives on node $h+1$ before $f(h+1)$. Then, m' starts its transmission before $f(h+1)$ on node $h+1$. As on this node, the busy period considered starts with $f(h+1)$, the processing of m' is completed at the latest at the arrival of $f(h+1)$. Hence $m' \notin (f(h+1), f(h+2))$. In a similar way, we show that packet $m' \notin (f(h'), f(h'+1))$, for any $h' \in \mathcal{P}_i$, visited after h by τ_j .

We then show that packet $m' \notin (f(h'), f(h'+1))$, for any $h' \in \mathcal{P}_i$, visited before h by τ_j . We proceed by contradiction. If such a node $h' \in \mathcal{P}_i$, visited before h by τ_j would exist, packet m' would not belong to $(f(h), f(h+1))$, by the first part of this proof. Hence a contradiction. Hence the lemma. \blacksquare

We now distinguish the nodes visited before $slow_i$, the node $slow_i$ itself and the nodes visited after $slow_i$. By definition, for any node $h \in [1, slow_i)$, $f(h+1)$ is the first packet with a priority higher than or equal to this of m , processed in bp^{h+1} and coming from node h . Moreover, $f(h+1)$ is the last packet considered in bp^h . Hence, if we count packets processed in bp^h and bp^{h+1} , only $f(h+1)$ is counted twice. In the same way, for any node $h \in (slow_i, q]$, $f(h)$ is the first packet with a priority higher than or equal to this of m , processed in bp^h and coming from node $h-1$. Moreover, $f(h)$ is the last packet considered in bp^{h-1} . Thus, $f(h)$ is the only packet counted twice when counting packets processed in bp^{h-1} and bp^h . Hence, $X_{i,t}$ is equal to:

$$\underbrace{\sum_{h=1}^{slow_i-1} \left(\sum_{g=f(h)}^{f(h+1)-1} C_{\tau(g)}^h + C_{\tau(f(h+1))}^h \right)}_{\text{nodes visited before } slow_i} + \underbrace{\sum_{g=f(slow_i)}^{f(slow_i+1)} C_{\tau(g)}^{slow_i}}_{\text{node } slow_i} + \underbrace{\sum_{h=slow_i+1}^q \left(\sum_{g=f(h)+1}^{f(h+1)} C_{\tau(g)}^h + C_{\tau(f(h))}^h \right)}_{\text{nodes visited after } slow_i} - C_i^q.$$

Moreover, for any node $h \in [1, q]$, for any packet g visiting h , the processing time of g on node h is less than or equal to $C_{\tau(g)}^{slow_{\tau(g),i}}$. Then, as packets are numbered consecutively from $f(1)$ to $f(q+1) = m$, we get inequation (1). In addition, by considering that on any node $h \in [1, slow_i)$ (resp. $h \in (slow_i, q]$), the processing time of $f(h+1)$ (resp. $f(h)$) on node h is less than or equal to $\max_{j \in [1, n]} \{C_j^h\}$ and in the worst case, $f(h+1)$ is a packet coming from node h , we get inequation (2).

$$\sum_{h=1}^{slow_i-1} \left(\sum_{g=f(h)}^{f(h+1)-1} C_{\tau(g)}^h \right) + \sum_{g=f(slow_i)}^{f(slow_i+1)} C_{\tau(g)}^{slow_i} + \sum_{h=slow_i+1}^q \left(\sum_{g=f(h)+1}^{f(h+1)} C_{\tau(g)}^h \right) \leq \sum_{g=f(1)}^m C_{\tau(g)}^{slow_i} \quad (1)$$

$$\sum_{h=1}^{slow_i-1} C_{\tau(f(h+1))}^h + \sum_{h=slow_i+1}^q C_{\tau(f(h))}^h \leq \sum_{h=1}^q \max_{\substack{j \in [1, n] \\ first_{j,i} = first_{i,j}}} \{C_j^h\} \quad (2)$$

By (1) and (2), we get:

$$X_{i,t} \leq \sum_{g=f(1)}^m C_{\tau(g)}^{slow_{\tau(g),i}} - C_i^q + \sum_{h=1}^q \max_{\substack{j \in [1, n] \\ h \neq slow_i \\ first_{j,i} = first_{i,j}}} \{C_j^h\}.$$

The term $X_{i,t}$ is maximized when the workload generated by such flows is maximum. Then, we get Lemma 2.

Lemma 2 *Let m be the packet of flow τ_i generated at time t . When flows are scheduled FIFO, the maximum delay incurred by m due to packets having a priority higher than or equal to this of m is bounded by:*

$$\sum_{\substack{j \in [1, n] \\ j \neq i}} \left(1 + \left\lfloor \frac{t + A_{i,j}}{T_j} \right\rfloor \right)^+ \cdot C_j^{slow_{j,i}} + \left(1 + \left\lfloor \frac{t + J_i}{T_i} \right\rfloor \right) \cdot C_i^{slow_i} + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq slow_i}} \max_{\substack{j \in [1, n] \\ first_{j,i} = first_{i,j}}} \{C_j^h\} - C_i^{last_i}, \quad \text{with:} \\ A_{i,j} = S_{max_i}^{first_{j,i}} - S_{min_j}^{first_{j,i}} - M_i^{first_{i,j}} + S_{max_j}^{first_{i,j}} + J_j.$$

Proof: Packets of flow τ_j , $j \neq i$, with $\mathcal{P}_j \cap \mathcal{P}_i \neq \emptyset$, can delay m if they are generated at the earliest at time $a_{f(\text{first}_{i,j})}^{\text{first}_{i,j}} - S_{\max_j}^{\text{first}_{i,j}} - J_j$ and at the latest at time $a_m^{\text{first}_{j,i}} - S_{\min_j}^{\text{first}_{j,i}}$. Moreover, packets of flow τ_i can delay m if they are generated at the earliest at time $-J_i$ and at the latest at time t . As the maximum workload generated by any flow τ_j in the interval $[a, b]$ on node h is equal to $(1 + \lfloor (b-a)/T_j \rfloor) \cdot C_j^h$, $a_m^{\text{first}_{j,i}} \leq t + S_{\max_i}^{\text{first}_{j,i}}$ and $a_{f(\text{first}_{i,j})}^{\text{first}_{i,j}} \geq M_i^{\text{first}_{i,j}}$, we get the lemma. \blacksquare

We can now express the latest starting time of packet m on its last visited node.

Property 1 *Let m be the packet of flow τ_i generated at time t . When flows are scheduled FIFO, the latest starting time of packet m on its last node visited, denoted $W_{i,t}^{\text{last}_i}$, is bounded by:*

$$\sum_{\substack{j \in [1,n] \\ j \neq i}} \left(1 + \left\lfloor \frac{t+A_{i,j}}{T_j} \right\rfloor\right)^+ \cdot C_j^{\text{slow}_{j,i}} + \left(1 + \left\lfloor \frac{t+J_i}{T_i} \right\rfloor\right) \cdot C_i^{\text{slow}_i} \\ + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq \text{slow}_i}} \max_{\substack{j \in [1,n] \\ \text{first}_{j,i} = \text{first}_{i,j}}} \{C_j^h\} - C_i^{\text{last}_i} + (|\mathcal{P}_i| - 1) \cdot L_{\max},$$

with:

$$A_{i,j} = S_{\max_i}^{\text{first}_{j,i}} - S_{\min_j}^{\text{first}_{j,i}} - M_i^{\text{first}_{i,j}} + S_{\max_j}^{\text{first}_{i,j}} + J_j.$$

Proof: By Lemma 2. \blacksquare

4.2. Worst case end-to-end response time

The worst case end-to-end response time of the packet of flow τ_i generated at time t is equal to: $W_{i,t}^{\text{last}_i} + C_i^{\text{last}_i} - t$. The worst case end-to-end response time of flow τ_i is then equal to:

$$R_i = \max_{t \geq -J_i} \left\{ W_{i,t}^{\text{last}_i} + C_i^{\text{last}_i} - t \right\}.$$

In order not to test all times $t \geq -J_i$, we establish Lemma 3.

Lemma 3 *Let us consider a flow τ_i following a path \mathcal{P}_i . When flows are scheduled FIFO, we have for any time $t \geq -J_i$:*

$$W_{i,t+\mathcal{B}_i^{\text{slow}}}^{\text{last}_i} \leq W_{i,t}^{\text{last}_i} + \mathcal{B}_i^{\text{slow}},$$

$$\text{with: } \mathcal{B}_i^{\text{slow}} = \sum_{j \in [1,n]} \lceil \mathcal{B}_i^{\text{slow}} / T_j \rceil \cdot C_j^{\text{slow}_{j,i}}.$$

Proof: As $\forall (a, b) \in \mathbb{R}^+$, $\lfloor a+b \rfloor \leq \lfloor a \rfloor + \lfloor b \rfloor$, we have $W_{i,t+\mathcal{B}_i^{\text{slow}}}^{\text{last}_i}$ bounded by:

$$\sum_{\substack{j \in [1,n] \\ j \neq i}} \left(1 + \left\lfloor \frac{t+\mathcal{B}_i^{\text{slow}}+A_{i,j}}{T_j} \right\rfloor\right)^+ \cdot C_j^{\text{slow}_{j,i}} \\ + \left(1 + \left\lfloor \frac{t+\mathcal{B}_i^{\text{slow}}+J_i}{T_i} \right\rfloor\right) \cdot C_i^{\text{slow}_i} \\ + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq \text{slow}_i}} \max_{\substack{j \in [1,n] \\ \text{first}_{j,i} = \text{first}_{i,j}}} \{C_j^h\} - C_i^{\text{last}_i} + (|\mathcal{P}_i| - 1) \cdot L_{\max} \\ \leq \sum_{\substack{j \in [1,n] \\ j \neq i}} \left(1 + \left\lfloor \frac{t+A_{i,j}}{T_j} \right\rfloor\right)^+ \cdot C_j^{\text{slow}_{j,i}} \\ + \left(1 + \left\lfloor \frac{t+J_i}{T_i} \right\rfloor\right) \cdot C_i^{\text{slow}_i} + \sum_{j \in [1,n]} \left\lceil \frac{\mathcal{B}_i^{\text{slow}}}{T_j} \right\rceil \cdot C_j^{\text{slow}_{j,i}} \\ + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq \text{slow}_i}} \max_{\substack{j \in [1,n] \\ \text{first}_{j,i} = \text{first}_{i,j}}} \{C_j^h\} - C_i^{\text{last}_i} + (|\mathcal{P}_i| - 1) \cdot L_{\max} \\ \leq W_{i,t}^{\text{last}_i} + \mathcal{B}_i^{\text{slow}}. \quad \blacksquare$$

From the worst case analysis given in Section 4.1 and the previous lemma, we get the following property.

Property 2 *When flows are scheduled FIFO, the worst case end-to-end response time of any flow τ_i is bounded by:*

$$R_i = \max_{-J_i \leq t < -J_i + \mathcal{B}_i^{\text{slow}}} \{W_{i,t}^{\text{last}_i} + C_i^{\text{last}_i} - t\}, \quad \text{with:} \\ W_{i,t}^{\text{last}_i} = \sum_{\substack{j \in [1,n] \\ j \neq i}} \left(1 + \left\lfloor \frac{t+A_{i,j}}{T_j} \right\rfloor\right)^+ \cdot C_j^{\text{slow}_{j,i}} + \left(1 + \left\lfloor \frac{t+J_i}{T_i} \right\rfloor\right) \cdot C_i^{\text{slow}_i} \\ + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq \text{slow}_i}} \max_{\substack{j \in [1,n] \\ \text{first}_{j,i} = \text{first}_{i,j}}} \{C_j^h\} - C_i^{\text{last}_i} + (|\mathcal{P}_i| - 1) \cdot L_{\max} ; \\ A_{i,j} = S_{\max_i}^{\text{first}_{j,i}} - S_{\min_j}^{\text{first}_{j,i}} - M_i^{\text{first}_{i,j}} + S_{\max_j}^{\text{first}_{i,j}} + J_j$$

$$\text{and } \mathcal{B}_i^{\text{slow}} = \sum_{j \in [1,n]} \left\lceil \frac{\mathcal{B}_i^{\text{slow}}}{T_j} \right\rceil \cdot C_j^{\text{slow}_{j,i}}.$$

Proof: By Property 1 and Lemma 3. \blacksquare

5. Example

In this section, we give an example of bounds on the end-to-end response times of sporadic flows, when these flows are scheduled according to FIFO. We consider that the network meets: $L_{\max} = L_{\min} = 1$. We focus on the flow set $\{\tau_1, \tau_2, \tau_3, \tau_4, \tau_5\}$. All these flows have a period equal to 36 and enter the network without jitter. The maximum processing time of any packet of flow τ_i on node $h \in \mathcal{P}_i$ is assumed to be equal to 4. Table 1 gives the end-to-end deadline of each flow.

Table 1. End-to-end deadlines

	τ_1	τ_2	τ_3	τ_4	τ_5
D_i	40	45	55	55	50

The path taken by each flow is defined as follows:

- $\mathcal{P}_1 = \{1, 3, 4, 5\}$
- $\mathcal{P}_2 = \{9, 10, 7, 6\}$
- $\mathcal{P}_3 = \{2, 3, 4, 7, 10, 11\}$
- $\mathcal{P}_4 = \{2, 3, 4, 7, 10, 11\}$
- $\mathcal{P}_5 = \{2, 3, 4, 7, 8\}$.

Table 2 presents the worst case end-to-end response time of any flow τ_i , $i \in [1, 5]$, applying Property 2. This table also presents the worst case end-to-end response times obtained with the holistic approach. The benefit provided by the trajectory approach is very important, as we get an improvement higher than 25%. Consequently, all flows meet their end-to-end deadline with the trajectory approach, none with the classical one.

Table 2. End-to-end response times

	τ_1	τ_2	τ_3	τ_4	τ_5
trajectory approach	31	43	53	53	44
holistic approach	43	63	73	73	56

6. Application to the EF class

In this section, we show how to apply the results given in Section 4 to the EF class in a DiffServ model.

6.1. DiffServ architecture

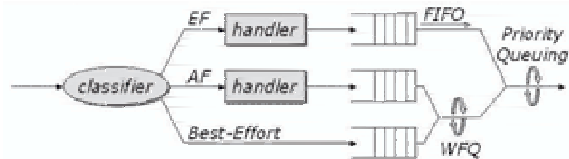
In the DiffServ architecture [7], traffic is distributed over a small number of classes. Packets carry the code of their class. This code is then used in each DiffServ-compliant router to select predefined packet handling functions (in terms of queuing, scheduling and buffer acceptance), called *Per-Hop Behavior* (PHB). Nodes at the boundary of the network (ingress and egress routers), perform complex treatments (packet classification and traffic conditioning) whereas nodes in the core network (core routers), forward packets according to their class code.

Several per-hop behaviors have been defined:

- the *Best-Effort Forwarding* PHB is the default one.
- the *Assured Forwarding* (AF) PHB group [8]. Four classes, providing more or less resources in terms of bandwidth and buffers, are defined in the AF service. Each class manages three different drop priorities representing the relative importance of a packet in the class.

- the *Expedited Forwarding* (EF) PHB [9]. Traffic belonging to the EF service is delivered with very low latency and drop probability, up to a negotiated rate. This service can be used for instance by IP telephony.

We consider that a DiffServ-compliant router implements Best-Effort, AF and EF classes. When a packet enters the node scheduler, it is scheduled with the other packets of its class waiting for processing. As illustrated by Figure 3, the EF class is scheduled with a *Fixed Priority* (FP) queuing scheme with regard to the other classes.

**Figure 3. DiffServ-compliant router**

Thus, the EF class is served as long as it is not empty. Packets in the Best-Effort and AF classes are served according to *Weighted Fair Queuing* (WFQ) [6]. In this way, EF traffic will obtain low delay thanks to FP scheduler and AF traffic will receive a higher bandwidth fraction than Best-Effort thanks to WFQ. Notice that resources provisionned for the EF class that are not used are available for the other classes.

6.2. EF class

Within the EF class, flows are scheduled FIFO. Moreover, the assumption generally admitted in a network is that packet scheduling is non-preemptive. Then, the scheduler of the node considered waits for the completion of the current packet transmission (if any) before selecting the next packet.

The definition of the EF PHB as given in [9] can be used to predict qualitative end-to-end delay guarantees. Beyond this definition, end-to-end guarantees are crucial for delay and jitter sensitive applications. Those QoS guarantees must be provided without requiring per flow processing in the core routers. Otherwise the solution would not be scalable. [10] shows that the worst case delay jitter for the EF traffic can be large in case of large networks.

The use of the FIFO scheduling algorithm for priority traffic in a network based on traffic aggregation (e.g. all flows in the EF class share a single FIFO queue) has been discussed in [11]. Nevertheless, the found delay bound is valid only for reasonably small EF traffic utilization.

In [12], a hybrid admission control scheme has been proposed, based on traffic shaping at border routers, to provide QoS guarantees in a DiffServ environment. The decision of the admission control is based on measurements realized to estimate resource allocation, leading to a higher utilization efficiency. As we are interested in deterministic guarantees, the admission control takes into account the worst case response times and jitters.

We see in the following subsection how to provide deterministic and quantitative guarantees on the end-to-end response times of flows in the EF class.

6.3. Application of our results

We can apply our results by considering the non-preemptive effect due to packets not belonging to the EF class.

The following lemma gives the maximum delay due to the non-preemption incurred by any packet of the EF class. For this, we denote:

- $j \in EF$ if flow τ_j belongs to the EF class;
- δ_i the maximum delay due to the non-preemption incurred by any packet of flow τ_i , $i \in EF$, on its path \mathcal{P}_i .

Lemma 4 *Let τ_i , $i \in EF$, be a flow following path $\mathcal{P}_i = [first_i, \dots, last_i]$. When flows are scheduled FIFO within the EF class, the maximum delay incurred by the packet of flow τ_i due to the non-preemption is bounded by:*

$$\begin{aligned} & \left(\max_{\substack{j \notin EF \\ first_{j,i}=first_i}} \{C_j^{first_i}\} - 1 \right)^+ \\ & + \sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \left(\max_{\substack{j \notin EF \\ first_{j,i}=h}} \{C_j^h\} - 1; \max_{\substack{j \notin EF \\ h \in (first_{j,i}, last_{j,i}] \\ first_{j,i} \neq first_{i,j}}} \{C_j^h\} - 1; \right. \\ & \left. 1_\alpha \cdot \left(\max_{\substack{j \notin EF \\ h \in (first_{j,i}, last_{j,i}] \\ first_{j,i}=first_{i,j}}} \{C_j^h\} - C_i^{pre_i(h)} + Lmax - Lmin \right) \right)^+, \end{aligned}$$

where $\max_{j \notin EF} \{C_j^h\} = 0$ and $1_\alpha = 0$ if there exists no flow not belonging to the EF class. Otherwise, $1_\alpha = 1$.

Proof: By induction on the number of nodes visited. On the first node visited, Lemma 4 is true. Assuming that Lemma 4 is true at rank h . We prove it at rank $h + 1$. Let us consider packet m of flow τ_i , $i \in EF$, generated at time t . Due to the non-preemption, on any node $h \in (first_i, last_i]$, a packet m' belonging to a flow τ_j , $j \notin EF$, can delay the execution of m if m arrives on node h while m' is

being processed. Then, we have to distinguish three cases:

- Node h is the first node of \mathcal{P}_i visited by flow τ_j ($first_{j,i} = h$). Hence, the maximum delay incurred by m due to flow τ_j meets: $C_j^h - 1$;
- Node h is not the first node of \mathcal{P}_i visited by flow τ_j ($h \in (first_{j,i}, last_{j,i}]$) and $first_{j,i} \neq first_{i,j}$. Hence, the maximum delay incurred by m due to flow τ_j meets: $C_j^h - 1$;
- Node h is not the first node of \mathcal{P}_i visited by flow τ_j ($h \in (first_{j,i}, last_{j,i}]$) and $first_{j,i} = first_{i,j}$. Packet m' leaves node $pre_i(h)$ at the latest at time $W_{i,t}^{pre_i(h)}$. Then, m' ends its processing on node h at the latest at time $W_{i,t}^{pre_i(h)} + Lmax + C_j^h$. As packet m arrives on node h at the earliest at time $W_{i,t}^{pre_i(h)} + C_i^{pre_i(h)} + Lmin$, the maximum delay incurred by m directly due to flow τ_j meets: $\max\left(0; C_j^h - C_i^{pre_i(h)} + Lmax - Lmin\right)$.

Moreover, $C_j^h \leq \max_{j \notin EF} \{C_j^h\}$. Hence the lemma. ■

Then, we get the following property on the worst case end-to-end response time of any flow belonging to the EF class.

Property 3 *When flows are scheduled FIFO within the EF class, the worst case end-to-end response time of any flow τ_i , $i \in EF$, is bounded by:*

$$\begin{aligned} R_i &= \max_{-J_i \leq t < -J_i + B_i^{slow}} \{W_{i,t}^{last_i} + C_i^{last_i} - t\}, \text{ with:} \\ W_{i,t}^{last_i} &= \sum_{\substack{j \in [1,n] \\ j \neq i}} \left(1 + \left\lfloor \frac{t + A_{i,j}}{T_j} \right\rfloor\right)^+ \cdot C_j^{slow_{j,i}} + \left(1 + \left\lfloor \frac{t + J_i}{T_i} \right\rfloor\right) \cdot C_i^{slow_i} \\ &+ \sum_{\substack{h \in \mathcal{P}_i \\ h \neq slow_i}} \max_{\substack{j \in [1,n] \\ first_{j,i}=first_{i,j}}} \{C_j^h\} - C_i^{last_i} + (|\mathcal{P}_i| - 1) \cdot Lmax + \delta_i; \\ A_{i,j} &= S_{max_i}^{first_{j,i}} - S_{min_j}^{first_{j,i}} - M_i^{first_{i,j}} + S_{max_j}^{first_{i,j}} + J_j; \\ B_i^{slow} &= \sum_{j \in [1,n]} \left\lceil \frac{B_j^{slow}}{T_j} \right\rceil \cdot C_j^{slow_{j,i}} \text{ and} \\ \delta_i &= \left(\max_{\substack{j \notin EF \\ first_{j,i}=first_i}} \{C_j^{first_i}\} - 1 \right)^+ \\ &+ \sum_{\substack{h \in \mathcal{P}_i \\ h \neq first_i}} \left(\max_{\substack{j \notin EF \\ first_{j,i}=h}} \{C_j^h\} - 1; \max_{\substack{j \notin EF \\ h \in (first_{j,i}, last_{j,i}] \\ first_{j,i} \neq first_{i,j}}} \{C_j^h\} - 1; \right. \\ & \left. 1_\alpha \cdot \left(\max_{\substack{j \notin EF \\ h \in (first_{j,i}, last_{j,i}] \\ first_{j,i}=first_{i,j}}} \{C_j^h\} - C_i^{pre_i(h)} + Lmax - Lmin \right) \right)^+. \end{aligned}$$

Proof: By Property 2 and Lemma 4. ■

7 Conclusion

In this paper, we have shown how to compute the worst case response times of flows scheduled with FIFO, using the trajectory approach. This approach considers only realistic scenarios, unlike the holistic approach that leads to pessimistic results. An example has been given to illustrate the computation.

An interesting application of these results lies in the deterministic quantitative guarantee provided to the Expedited Forwarding (EF) class in a DiffServ architecture, assuming that the EF class is scheduled with the highest priority with regard to the other classes and flows within the EF class are scheduled FIFO.

References

- [1] S. Baruah, R. Howell, L. Rosier, *Algorithms and complexity concerning the preemptive scheduling of periodic real-time tasks on one processor*, Real-Time Systems, 2, p 301-324, 1990.
- [2] K. Tindell, J. Clark, *Holistic schedulability analysis for distributed hard real-time systems*, Microprocessors and Microprogramming, Euromicro Jal, Vol. 40, 1994.
- [3] M. Spuri, *Holistic analysis for deadline scheduled real-time distributed systems*, INRIA Research Report No 2873, April 1996.
- [4] J. Y. le Boudec, P. Thiran, *Network calculus: A theory of deterministic queuing systems for the Internet*, LNCS 2050, Springer-Verlag, September 2003.
- [5] J. Le Boudec, P. Thiran, *Network calculus: A theory of deterministic queuing systems for the Internet*, Springer Verlag, LNCS 2050, September 2003.
- [6] A. Parekh, R. Gallager, *A generalized processor sharing approach to flow control in integrated services networks*, IEEE ACM Transactions on Networking, 1994.
- [7] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, *An architecture for Differentiated Services*, RFC 2475, December 1998.
- [8] J. Heinanen, F. Baker, W. Weiss, J. Wroclawski, *Assured Forwarding PHB group*, RFC 2597, 1999.
- [9] V. Jacobson, K. Nichols, K. Poduri, *An Expedited Forwarding PHB*, RFC 2598, June 1999.
- [10] J. Bennett, K. Benson, A. Charny, W. Courtney, J. Le Boudec, *Delay jitter bounds and packet scale rate guarantee for Expedited Forwarding*, INFOCOM'2001, Anchorage, USA, April 2001.
- [11] A. Charny, J. Le Boudec, *Delay bounds in a network with aggregate scheduling*, QoFIS, Berlin, Germany, October 2000.
- [12] M. Gerla, C. Casetti, S. Lee, G. Reali, *Resource allocation and admission control styles on QoS DiffServ networks*, QoS-IP 2001, Rome, Italy, 2001.