1 Review A

Section I. Overview
This paper is 60% identical (to the words) to a paper by the same authors "Measureing Service in Multi-class Networks" in InfoCom 2001. The previous paper is not referenced or mentioned.

The new material is for web server experiments, and some expansion on the previous explanations (I counted 14 pages new out of 33, being generous). The writing of the new material is of inferior quality. Whether this is sufficient novelty is a matter of editorial policy. There is substantial new material.

The reference to the authors' previous and largely identical paper, on which this is clearly based, needs to be included, with clear indications of where the novelty here lies.

Submitting the paper without this acknowledgement is an ethical infringement which I would say makes the paper borderline acceptable on ethical grounds.

Dense material, with gaps. I would like to insist on their summarizing the method with an algorithm.

The rsearch is ambitious and shows, in a limited application of its ideas, an interesting level of success. However the explanation is incomplete, the language is sometimes contorted or confusing, and the limitations are not discussed.

The contorted and confusing language tends to be in the new material (not in the Infocom version).

Section II. Summary and Recommendation

Section III. Detailed Comments

Major: clear reference to the previous version and what is new here.

———

(A.1) The first version of the paper is submitted to TPDS on 01/09/2002 and it did not reference our IEEE Infocom '01 conference publication "Measuring Service in Multi-Class Networks" by oversight. However, we have sent a corrected submission of the paper on 01/10/2002, which had a reference to the conference publication in the footnote of the front page. On 01/11/2002 we received a confirmation from TPDS

assistant that the revisited version of the paper is received and that it will be forwarded to the Editor-in-Chief. It is obvious that only the first version of the paper (submitted on 01/09/2002) has been forwarded to the reviewers.

All of the above is fixed in the latest version of the paper.

The work presented in the submitted paper is significantly advanced now compared to the conference publication. It now considers unmeasured cross-traffic (incorporated in the framework through the concept of variable available capacity) and application to quality of service web servers.

———

```
Major: The calculation of the likelihood ratio is sufficiently complex
that a reader would have great difficulty in implementing it. It
requires expression by an algorithm, that pulls the steps together in
sequence. For instance I cannot figure out whether gamma star is always
calculated and used in the scoring, or just sometimes. And the
computation of measures like R with an arrow over it (bottom of p 13)
need a more systematic explanation (what does "normalized" on the
third-last line of p 13 mean? ...I couldn't find the explanation).
```

———

(A.2) As requested by the reviewer, we have added a step-by-step joint measurement/inference algorithm that pulls the steps together in sequence. The algorithm is presented in the Figure 9.

As indicated in the step 4 of the Figure 9 and Equation (18), γ^* is *always* calculated and used in scoring.

Next, the word "normalized" on p. 13 (now p. 12) refers to the relation between values $\vec{R}_k^{i,S}$ and $S^i(I_k)$. $S^i(I_k)$ is a service envelope of class i in the interval of length I_k. Samples of this random variable form a set of measurements $\vec{R}_k^{i,S} * I_k$. When each of these samples is normalized (divided by the time interval I_k), we obtain the appropriate set of *rate* service measurements $\vec{R}_k^{i,S}$. We have modified the text to describe this issue more clearly.

———

```
Major: the method is approximate and the evaluation is limited to two
classes. These facts must limit the conclusions that can be drawn, but
the "conclusions" section does not go into any evaluation of
limitations. Surely the authors can do this. Can I go beyond two
classes? Does the paper support going beyond two classes? What
assumptions are most fragile?
```

———

(A.3) We address the above objections by making the following changes:

i) In the revisited submission, we generalize the presentation for $k > 2$ classes. In this context, we decouple inference methodologies for SP and WFQ schedulers and provide separate p-equations (Equations (10) and (11)) that are applicable for $k > 2$.

We added sub-section IV.B.1 (*SP Relative Priority Estimation*) to address the above issues.

ii) We added a sub-section IV.D (*The Algorithm Summary and Discussion*) where we discuss the computational complexity limitations.

iii) Finally, the above issues are included in the conclusions part.

———

```
The term "time scale" is important to the ideas, and is sometimes used
inappropriately. In the introduction of envelopes in sec 3.1, measures
are a function of time intervals. Only later, when a set of intervals of
each length is used, is the term suitable to describe the set.
```

———

(A.4) We have corrected the terminology in section 3.1 (now III.A) as pointed out by the reviewer.

———

```
I have difficulty with the (centrally important) p function in Eq 7. It
seems to apply only to two classes, while the paper claims to be about
any number of classes (the same applies to the following, unnumbered
function for EDF). If the "other" class can just as easily be an
aggregate, this needs to be clearly stated. Otherwise, either the claims
of the paper should be limited, or the more general expressions are
needed.
```

———

(A.5) The reviewer is right in pointing out that "n" meant to denote an aggregate of traffic classes. This is now clearly stated in the paper, as we introduce a new notation for X_k^i, Y_k^i and Z_k^i. Accordingly, we provide the more general expressions (Equations (10), (11) and (12)) for SP, WFQ and EDF, respectively) that are applicable to any number of classes. Note that Equation (7) from the first version of the paper has now become Equation (11), while the unnumbered equation for EDF has now become Equation (12). Finally, the corresponding p-equation for SP is now included in the text as Equation (10).

———

```
In the important Table 1, the notation C(t) appears to mean Ct... the
product, rather than a function... if so, it would be much easier to
understand if stated that way. The units of C should be defined.
```

———

(A.6) The reviewer correctly points out that $C(t)$ in Table 1 (now Equations (1), (2) and (3)) means the product Ct, rather than a function. We have fixed the above by making appropriate changes.

———

In Eq 7, isn't X sub k just a model for B sup n sub k? It would be a LOT easier to follow if the normal distribution (top of p 15) were described as a model for B.... in any case X should be indexed by n as well as by k, and the same for Y below it.

———

(A.7) Actually, in Equation (7) from the first version of the paper (now Equation (11)), X_k was a model for $S^i(I_k)$. In any case, we believe that the changes made in (A.5) resolve this issue also. More precisely, random variables X, Y and Z in Equations (10), (11) and (12) are now indexed with class' index i, while the relationship among indexes i and n are now clearly stated in definitions of X, Y and Z.

———

The weight phi is unnormalized in some discussion (see the middle of p 9, where the share is computed by normalizing it) and is assumed elsewhere to be normalized already (see p 18, phi sub 2 is 1 - phi sub 1). This should be clear and consistent.

———

(A.8) We have unified the discussion throughout the paper by assuming that weights ϕ_i are normalized.

———

The terminology about envelopes is sufficiently new to most readers, that it should be used with correct syntax. On p 11, the term "rate arrival envelope" surely should be "arrival rate envelope", and similar for service rate. Both orders are used elsewhere, continuing the confusion.

———

(A.9) We have made the terminology consistent by using "arrival rate envelope" and "service rate envelope" throughout the paper.

———

p 13 WHY is mean + 1.6 time deviation a service envelope? This seems to be completely arbitrary. What point is being served here?

———

(A.10) $Mean + 1.6\,deviation$ is used as *an example* of a general α-envelope ($mean + \alpha\,deviation$). Having in mind that most readers are not familiar with the concept of envelopes, our first goal was to depict an example envelope to help the reader's intuition. Second, we wanted to show an important difference between arrival envelope and its *normalized* representation, which is arrival *rate* envelope. Finally, the goal was to show the dependence of traffic variability on time scales, an important characteristic of the request workload used for obtaining accurate scheduler inferences.

———

p 12 line 4 The use of bits as a unit of throughput seems to be wrong, since they are hardly monitoring the arrival and departure of each bit. Define an better service rate unit, or relate the unit of monitoring (packets, requests) to the unit of throughput.

———

(A.11) The reason for using bits instead of the actual units of monitoring (packets, requests) is simply because the former may have different size. We have added an explanation to this effect in the paper.

———

p 21 in the definition of gamma, top of p 21, n needs to be explained all over again. Does it refer to an aggregate of a set of classes, or is the definition restricted to a two-class system, and n is the "other" class?

———

(A.12) In the first version of the paper, "n" is meant to denote an aggregate of a set of (all other) classes. In this context, we have changed the notation in the definition of γ.

———

p 23 line 2 what is the meaning of 65 - 68 sources, and 25 - 28 sources. Does it mean "from 65 to 68 sources?" Why a range, rather than just one number?

———

(A.13) Terms "$65-68$ sources" and "$25-28$ sources" mean "from 65 to 68 sources" and "from 25 to 28 sources". We have changed the text accordingly. This shows that the number of flows in the system was varied in the simulations. The goal was to simulate flow arrivals and departures, as in a real system. These flow-level arrival-departure effects are important since they can introduce longer time-scale fluctuations and non-statonarities that could eventually bias the estimation. However, the simulation showed that our scheme is robust even if the number of flows is varied within a range of 5 to 10% of the system load.

———

p 23, a confusion has crept in about the work "interval", which appears now to mean the measurement window T rather than the interval I sub k. This needs to be clarified in this paragraph.

———

(A.14) The word "interval" is used for I_k, and has been mistakenly used for the measurement window T. This is fixed now.

———

p 24 line 6 What other side? What sides are there here?

———

(A.15) The sentence beginning with "On the other side, we perform $\cdots$" is related to the sentence "We perform two types of experiments." in p 24 line 3 (from the first version of the paper). We have changed the beginning of the sentence in "In the second type of experiments, we perform $\cdots$".

———

```
p 24 The whole example in sec 5.1.2 is muddied by the fact that
(apparently) the WFQ hypothesis is incorrect for the simulation. Well,
it is a simulation, so what was the discipline? Why not put this fact up
front so the reader understands what the section is about (which is,
presumably, what happens when the scheduler doesn't fit any hypothesis
in the model)
```

———

(A.16) Section 5 (*Experimental Investigations*) is structured such that sub-section 5.1 (now V.A) shows results on parameter estimation, while sub-section 5.2 (now V.B) shows results on scheduler inference. Accordingly, sub-section 5.1.2 (now V.A.2) only shows the results of the relative class weights in a QoS web-server scenario, while the appropriate scheduler inference results are presented in subsection 5.2.2 (now V.B.2) ("The percentage of correct scheduler detection averaged over all 500 tests is 0.53. Namely, it is 1.0 for FCFS (all 250 tests for FCFS scheduler were correct), while it is 0.06 for WFQ (only 15 out of 250 decisions were correct). This is because $\cdots$").

———

```
p 26 the discussion below the figure seems to be plain wrong... the
correctness drops well below 0.94.
```

———

(A.17) Figure 16 (now Figure 12(b)) depicts the probability of correct decision versus time scales, while the *final* correctness probability is obtained by applying the *majority* rule over *all* time scales. In the particular example, the final (overall) correctness probability is 0.94. This means that in 47 out of 50 experiments the majority test correctly determined the scheduler, while it failed in 3. However, note that this does not prevent the *per time-scale* correctness probability to be less than 0.94.

———

```
p 27 the fair shares were chosen to be the same as the arrival ratio...
doesn't this limit the behaviour of the system and give a special case?
```

———

(A.18) This actually gives a special case, but only in the sense that the particular arrival ratio is within the *measurable region* for given fair shares. The measurable region (explained in detail in section V.C) is defined as the minimum number of each class' flows needed such that unknown parameters are estimated within 5% of their correct value. While it was not stated explicitly, these regions also limit the scheduler

inference correctness probability, as the probability sharply drops when the number of flows in the system drops below the measurable region minimum. We have added an explanation in the paper (in section V.C) to better bring out this point.

———

```
p 27 again... this isn't an experiment on variable system capacity
versus fixed... it is an experiment in which the workload is MORE
variable than before, or LESS variable. This kind of variability is
already present in the previous work. Variable capacity might refer to
changes in the number of servers during the measurement. Badly named.
```

———

(A.19) It is indeed true that the workload in the web-server scenario is much more variable than in the networking scenario. However, we intend the term "variable capacity" in the web-server scenario to express the variability of a web-server's service which is *independent* of the workload from other classes. For example, consider a single-class web server fed by the constant-rate stream of requests. Even under these idealistic circumstances, the service times for these requests will be non-constant (variable) due to effects such as different CPU service times, disk service times and variable file sizes.

This type of variability (not caused by the behavior of "other" classes) is *not* present in the previous work. It is now included in the system modeling part as explained in section II.C. Further, it is included in the algorithm through Equation (13). Finally, the impact of this important effect on the scheduler inference problem is experimentally evaluated in the section V.B.2. We revisited the text to clarify this point.

———

```
p 29 Table 2 is impossible to understand, and the discussion wanders.
For instance the statement is made that gamma star should not be too
big... but no evidence is presented. Present more discussion, and make
it more to the point.
```

———

(A.20) Due to space constraints, we were forced to exclude the Table 2 from the paper. However, we have modified the discussion to make this point more clear.

———

```
Spelling is moderately poor, e.g. "then" for "than" on p 29, 5 lines
from the bottom.
```

———

(A.21) We have corrected the typo pointed out by the reviewer.

———

2 Review B

Section I. Overview

Some missing references are papers that have looked in estimation of non-QoS network characteristics (such as link or path capacity or available bandwidth) using passive or active measurements. Some other references (e.g., related with specific QoS problems) can probably be skipped as only tangentially related.

———

(B.1) We agree that such references are relevant and added several references illustrative of the non-QoS field [15-18].

———

Even though the paper is well-written, there are several typos. Here is a partial list:
- explixitely (p.5)
- asses (p.7)
- performe (p.27)

———

(B.2) We have corrected the typos pointed out by the reviewer.

———

Section II. Summary and Recommendation

Section III. Detailed Comments

Some specific technical comments follow:

1. A major concern that I have is that the proposed techniques are probably not applicable in the general case of k>2 classes. This point is rather "hidden" in the paper at a footnote (page 5), where it is stated that the number of classes is k=2 "for simplicity". There would be several hard problems, however, if the techniques were applied in the case of k>2 classes.

———

(B.3) The two-class scenario is used for the sake of simplicity of presentation. However, in the revisited submission, we generalize the presentation for $k > 2$ classes as also discussed in Reviewer A comments. Detailed answers on Reviewer B's concerns are given below.

———

First, the authors mention (p.15) that the detection of the Strict Priorities (SP) scheduler is a special case of the detection of the WFQ scheduler,
for \phi_1 * \phi_2 =0. This is true for two classes, but NOT for k>2 classes. How can we detect an SP scheduler (that every router supports, as far as I know) for k>2 classes? Notice that the case of k>2 classes is important in practice, as most routers support at least 4 or 8 classes, and if an ISP goes into the trouble of providing QoS they would probably do so for more than just 2 classes.

———

(B.3.1) It is indeed true that SP scheduler is a special case of WFQ scheduler only for $k = 2$, and that this does not hold for $k > 2$. However, the algorithm is generalizable to inference problems for $k > 2$. In the revisited submission we provide separate p-equations for SP and WFQ (Equations (10) and (11)) that are applicable for $k > 2$. The key step of the methodology in the $k > 2$ scenario is to first estimate the most-likely class priorities under the SP scheduler hypothesis, and then to compare the probability of this event against the most likely WFQ scenario. We added sub-section IV.B.1 (*SP Relative Priority Estimation*) to address the above issues.

———

A second reason that the k>2 case is hard: most of the statistical inference techniques proposed in the paper are iterative, searching numerically a k-dimensional (or higher) space. They may be tractable for the case of k=2, but quickly become intractable with a few more classes.

———

(B.3.2) The reviewer correctly points out that the algorithm becomes computationally more intensive as the number of classes increases. We revisited the text to explicitly point out this issue and narrow the scope to a moderate number of classes. In any case, we address several other issues regarding this criticism below.

i) In our implementation of scalable admission control, (http://www.ece.rice.edu/networks/software) we showed that *multi-class* envelope computations are feasible in real-time. While we actually did not further implement our scheduler inference techniques, it should be expected that the above implementation in junction with a network processor can successfully handle computation burdens in real time for a moderate number of classes, e.g., four to eight, typically used in multi-class networks.

ii) Note that the scheduler inference algorithm can be implemented such that the envelope data is collected within measurement windows that last several seconds, followed by pauses of several tens of seconds or even minutes, which would provide enough time for a numerical search.

The above issues are now discussed in the paper in the section IV.D.

———

A third reason that the k>2 case is hard: as the number of WFQ classes

increases, the bandwidth sharing between classes becomes much less predictable, because the space of possible bandwidth allocations increases exponentially. Inferring that the scheduler is truly WFQ, or even harder, estimating the class weights, may be practically impossible with say 4 or 8 classes.

———

(B.3.3) The relative class weight estimation can be performed only over time intervals when *all* classes are backlogged since it is only during such intervals that all classes incur their lower bounds in service. Such intervals cause peaks at the lower clipping of the service rate distribution and also maximize the joint distribution of Equation (15). While the probability of appearance of such intervals actually decreases as the number of classes increases (for given arrival class distributions), note that the variance of arrival traffic plays a key role in revealing the scheduler type. As the variance of arrivals becomes larger, the probability of clipping lower service bound increases. This issue is included in the algorithm through the rate-variance condition of Equation (18). In other words, estimating WFQ weights is feasible in $k > 2$ scenario, as long as the variances of each class' arrivals are big enough to cause time intervals in which all classes are backlogged. As above, we have revisited the text regarding this issue.

———

2. Regarding Figure-1 (and the related text): to apply the proposed techniques, we would need a network monitor both before and after a network router. I mean that both the arrival times at the input interfaces and the departure times at the output interfaces are needed.

———

(B.4) The reviewer is right when stating that both arrival times at the input interface and the departure times at the output interface are needed. In the case of a network shown in Figure 1, we simply resort to the common assumption of a "single bottleneck", and thus assume that the queuing delay on a bottleneck queue dominates the entire path's queuing delay. Further, by using the minimum end-to-end delay as an estimate of a propagation delay, it is possible to *approximate* the actual arrival/departure times using only edge-based measurements. Further details on measurement methodology can be found in reference [22].

———

3. Something must be said about the many different variations of WFQ (for instance, the discussion of page 6 refers to GPS and not to the WFQ scheduler). Viewed as rate-latency servers, the various GPS approximations differ in their latency component. Can the proposed technique estimate the latency component? There must be a relation between the "estimation timescales" issue that the paper discusses and the latency provided by different WFQ variations.

———

(B.5) The authors are aware of various GPS approximations, their rate-latency representations and the fact that different GPS approximations differ in their latency component. The reviewer correctly points out that there should be a relation between "estimation timescales" and the latency provided by WFQ variants. For example, one should expect that various GPS approximations have different Figure 12(b)-like characteristics. The lower the latency component is, the sooner the probability of correct decision should increase on shorter time scales. However, to systematically treat the problem, it is necessary to include the latency component in the service envelope model. In other words, it is necessary to first theoretically develop separate service envelopes for each of the GPS approximations, a generalization beyond the scope of this work.

———

4. I am not convinced with the authors' statement in page 7 that the proposed framework is extendible to any other scheduler. The statistical service envelope depends on both the scheduler and the class arrival empirical envelopes. I would expect that quite different schedulers can end up having roughly equivalent statistical service envelopes when they are given the "right" type of input traffic. An extreme example is if the arrival traffic is such that two classes are never backlogged at the same time: every scheduler will look the same with FCFS. This concern brings me to the next point.

———

(B.6) It is indeed true that quite different schedulers can end up having roughly equivalent statistical service envelopes given a certain type of input traffic. However, an important assumption in our algorithm is that the arrivals from different traffic classes are *statistically independent*. Under this assumption, it is expected that quite different schedulers should also have quite different statistical service envelopes. The reviewer's example of two classes not being backlogged at the same time is an extreme example of strong statistical dependence between arrivals of two classes, which violates the assumption.

The assumption on the independence of the arrivals is stated in reference [21]. We now explicitly add an explanation to this effect in the paper in section III.A.1

———

5. The passive measurements approach taken by the authors in this paper is, as previously mentioned, important and interesting. It may be hard to apply though (with accurate results) for multiple classes and arbitrary schedulers. It seems to me that the QoS classification and characterization problem that the paper focuses on can be more effectively solved with active measurement techniques in which the user sends special sequences of packets (belonging to all classes) and analyzes the delays that these packets experienced. In particular, these packet probes can have special patterns in order to force the scheduler to "reveal" itself, producing identifiable short-timescale delay signatures that are representative for each scheduler.

I recommend the authors to look into that direction. It should be noted here that active measurements do not need to be "intrusive" in long timescales. Getting an "impulse response" from a scheduler just needs a short-term special probe (similar to a delta function in system identification). The authors can look into the packet pair/train techniques used for capacity and available bandwidth estimation (even though those techniques assume FCFS servers, of course).

———

(B.7) We agree that active measurement techniques might also be applied to QoS classification problem. However, we consider the possible applications of such techniques to be beyond the scope of our work. Nevertheless, we point out several difficulties in applying such techniques to the QoS classification problem. First, note that unmeasured cross-traffic from different classes might be a major obstacle in obtaining accurate scheduler classifications. For example, it is well known that even in a single class scenario with FCFS server, cross traffic can cause large inaccuracies in the bottleneck bandwidth estimation when packet-pair like techniques are used. It should be expected that this problem becomes more pronounced in a multi-class system, where the cross-traffic from multiple classes might de-synchronize multi-class probes and blur the short-timescale delay signatures, especially for a larger number of classes and considering complex inter-class resource sharing rules. Second, note that the service times for the web-server requests can have large variance even in a single-class scenario, due to effects such as different CPU service times, disk service times and variable file sizes. Thus, there are significant challenges for active probing techniques from networking to be successfully applied to the web-server problem, especially in a multi-class scenario.

———

6. I have some concerns about the material related to EDF. First, I think that there is an important typo in the corresponding entry of Table 1 (instead of D_i it should be \delta_i, right?). Second, is it really important to study EDF? I am not aware of any routers that support it (even though it may be more popular in web servers). Third, I would like to see a statistical service distribution for EDF in Figure 16. Why do the authors only show such histograms for SP and WFQ? Fourth (and this is probably a major issue), the numerical estimation of the EDF delay bounds is not just a similar problem with the estimation of the WFQ weights. The WFQ weights are bounded in [0,1], and so we need to numerically search a bounded space. This is not the case (in theory at least) for EDF. The delay bounds are in [0, inf).

———

(B.8) First, the appearance of D_i in the Table 1 (now Equation (3)) is not a typo. While the point of most importance in EDF is the probability of violating the class delay bound δ_i, in [21] we derived an expression for the entire delay distribution of class i, for

all $D_i > 0$. The derivation of the expression and the proof can be found in the extended version of [21] that is downloadable from http://www.ece.rice.edu/networks/publications.html.

Second, EDF is a scheduler that provides important theoretical performance bounds used in admission control. It is known that EDF is optimal in a single link scenario in the sense that it provides the largest schedulable region among all non-preemptive policies. Regarding applicability in routers, note that there is work on feasible approximations of EDF, e.g., *Priority Queue Schedulers with Approximate Sorting in Output Buffered Switches, J. Liebeherr and D. E. Wrege, IEEE Journal on Selected Areas in Communications. Special Issue on Next Generation IP Switches and Routers, Vol. 17, No. 6, pp. 1127-1145, June 1999.*

Third, the reason for showing a histogram for WFQ was to emphasize the clipping effect that (we believe) can help the reader to better understand the material. Next, the reason for showing the histogram for SP was exactly the absence of this effect in the low priority class. However, since the histogram of EDF does not show any clipping effects either, we did not show it as this point has already been made with SP.

Fourth, the reviewer correctly points out that the space of delay bounds is unbounded in theory. However, in practice we can reasonably bound the delay bounds (typically of the order of several hundreds of milliseconds) and enable numerical search in a bounded space.

———

`7. Regarding the rate-limiter parameter estimation: a user would probably also be interested in measuring the depth of the token bucket. Is this feasible with the proposed techniques? To return to my previous point, I would expect that it is feasible with active measurements.`

———

(B.9) Yes, it is possible to estimate the depth of the token bucket. We present a simple method based on basic graphical representation of service envelopes. Consider a token bucket filter with bucket depth B^i and rate-limit parameter r^i in class i. According to Equation (16), the MLE estimates $\hat{r}_k^i$ of the rate-limit parameter r^i are obtained for each time scale I_k. Consider further a coordinate system where the x-axis denotes time-scales while the y-axis denotes class service in units of packets. If a straight line is fitted through the points $\hat{r}_k^i\, I_k$, then the point on y-axis that intersects with the fitted line gives an estimate of the bucket depth B^i in packets.

———

`8. In the experimental section: it is stated that T should not be too large because that may cause long-term fluctuations and non-stationarities that can bias the results. I feel that there is a major issue here that is not sufficiently investigated or at least discussed. Why can non-stationarities introduce bias in the estimated parameters?`

———

(B.10) The service envelopes are developed in [21] under the assumption that the arrival process is stationary. If the arrivals are non-stationary, then *no* time-invariant

distributions of both arrival and service envelopes can be derived. Next, regarding bias in the estimated parameters, note that the time-of-day-like non-stationarities in arrivals can drive the scheduler in and out of the measurable region. Any longer time interval outside the measurable region can bias the estimated parameters.

———

9. An important assumption in the paper is that the arrivals in a time interval I_k follow a Gaussian distribution. Even though I do not have a major problem with this assumption, I think that its use/necessity should somehow be justified a bit deeper. Would it be a valid assumption if the packet interarrivals (or on/off times in an on-off model) were distributed according to a heavy-tailed distribution?

———

(B.11) The motivation behind Gaussian traffic characterization is that it is very natural when a large number of sources are multiplexed (via the Central Limit Theorem). In fact, it has been shown in [26] that aggregation of even a fairly small number of traffic streams is usually sufficient for the Gaussian characterization of the input process. Gaussian processes are completely specified by their first two moments. This makes Gaussian traffic characterization ideal from a measurement point of view, since measuring statistics beyond the second moment is usually quite impractical. Regarding the flows with heavy-tail distributed inter-arrival times, it is well known that the superposition of these flows is well modeled with self-similar processes. Note that Gaussian processes can have arbitrary correlation structure and this includes LRD (when the covariance function is not summable) processes. Hence, Gaussian processes cover all second-order LRD or second-order self-similar processes which have been shown to be good models for characterizing actual traffic.

———

3 Review C

SECTION I. OVERVIEW
The work provides a set of schemes (estimations), which enable a service provider to determine the network's scheduling desciplines/parameters in use. While the mechanisms as such ar well known (MLE), their application into networking sounds reasonable. However, that level of generality as depicted in the abstract, can not be viewed in the text itself.

Title: Observation hasn't been used in the text, only measurement are utilized there. Unification would improve the context.

———

(C.1) In the title, we have used the word *observation* as a synonym for the word *measurement*.

———

Abstract: It suggests too much generality, which is not really existing. Though, the restriction to the three scheduling disciplines is mentioned.

Mix of examples and model in section 3 is obstructing. Models are models and examples should be separated. Section 5.2 is well done, even though lengthy.

The use of "new" terminology does not help the reader at all, what is a network's core QoS mechanism? What are server clients? Some clear terminology re-do is required.

———

(C.2) All of the above comments are addressed in the answers below.

———

SECTION II. SUMMARY AND RECOMMENDATION

It may be done by minor revision for the paper's core (Section 4 and 5.2), however, the Section 2 and 3 are lengthy and should be focussed more clearly.

SECTION III. DETAILED COMMENTS

Abstract:
- Where are differentiated services available in the network? No public ones, only extremely dedicated ones or research networks.

———

(C.3) While it is true that differentiated service classes are not widely implemented in the network core, it is our experience that they are implemented at the network

edge. For example, one local ISP Phonoscope provides strict priority (SP) service within ingress-egress dedicated pipes. This is achieved with packet marking (tagging) at the edge and with SP scheduling in the ISP's core.

———

```
- What are "server clients", even though "network clients" seem to be
clear.
```

———

(C.4) The reviewer correctly points out that the phrase "server clients" sounds confusing. Our goal was to denote web-server clients. We have fixed the above by making appropriate changes to the paper.

———

```
- What is a system's core QoS functionality?
```

———

(C.5) The phrase "system's core QoS functionality" denotes the service differentiation functionality applied to traffic classes, i.e., the service discipline such as SP, WFQ or EDF. We have modified the text to describe this issue more clearly.

———

```
Targeted Systems
- 2.2.1: How can you quantify a service across administrative boundaries
without any standardized approach?
```

———

(C.6) The reviewer refers to the sentence "$\cdots$ network clients can use the framework for *quantifying* their service when only relative performance guarantees are provided or when end-to-end service is provided through more than one ISP." The point we wanted to make is that even if there are *no* strict end-to-end service guarantees, one can actually measure and estimate (quantify) the most likely service (e.g., bandwidth) obtained for different classes in a certain time-of-day slot. We have modified the sentence to describe this issue more clearly.

———

```
- 2.2 Keeping k to tow is fine, but the what about the complexity wrt
later on following analysis, if k=100?
```

———

(C.7) The reviewer correctly points out that the complexity with respect to the number of classes has not been addressed. This issue is now discussed in the section IV.D and as discussed in Reviewer A and B comments (see A.3 and B.3.1 to B.3.3).

———

```
- Why is the Network Calculus not being mentioned at all? It provides
means for determing QoS as well, independent of the client or server point
of view on a network.
```

———

(C.8) Network Calculus applies to deterministic service only and hence does not address or describe effects of one class on another. Indeed, deterministic service precludes inter-class resource sharing. Given inherent variability in web service and random effects of cross traffic, we required a statistical framework.

———

```
- Your reated work is sort of short in the sense, that only some is
referenced, and all measurement technology are missing.
```

———

(C.9) See the answer B.1.

———

```
- 2.3: The models leaves it open, if the network of figure 1 is within a
single administrative domain, or it is not. In either case the e2e SLA
issues need a clear definition. E2e SLA for a deicated path may exist, but
are not at all practical, since this approach wouldn't be scalable with
many input and output ports.
```

———

(C.10) The network of Figure 1 is within a single administrative domain. While the reviewer is correct when stating that an end-to-end SLA might have serious scalable limitations with many input and output ports, this is true only if per ingress-egress scheduling is performed at core routers. However, network providers usually overprovision the core, while service differentiation (or simply rate-limiting) is performed only at the edge.

———

```
Envelopes: This section covers too many references to [16] and may mislead to
the fact that a recapitulation of such work has been performed too extensively.
```

———

(C.11) We have reduced the number of references to [16] (now [21]).

———

```
- It remains unclear why figure 6 is being displayed here. This is not
conceptual work it is a clear example of a dedicated case.
```

———

(C.12) See the discussion in A.10.

———

```
Inference: Sec 4.1 argues "for simplicty  we consider a Gaussian
distribution".
But this sounds like an extreme theoretical case, since traffic - neither
aggregated traffic - will not be of such kind in real system. Justify and
explain your asumptions.
```

———

(C.13) See comment B.11.

———

```
- What do figures 8 and 9 tell us here? Examples are fine, but what is the
conclusion here?
```

———

(C.14) The reason for showing a histogram for WFQ was to emphasize the clipping effect that (we believe) can help the reader to better understand the material. Next, the reason for showing the histogram for SP was exactly the absence of this effect in the low priority class.

———

```
Experiments: Simulations certainly show the effctiveness, but is your approacg
generalizable by proof to other schedulers and traffic? You noted at the
least the former at an earlier stage. But is this a proof?
Certainly not. Therefore, your simulations show a dedicated effectiveness
for the cases considered only.
```

———

(C.15) In the paper, we use a statistical estimation/detection methodology called the Generalized Likelihood Ratio Test (GLRT). However, to apply this methodology, it is necessary to a prior know the expected distribution of the measured random variable. In our particular scenario, it is necessary to know the expected distribution of a service envelope, given the arrivals from other classes. Thus, our statement that the methodology is generalizable to any other scheduler means that it is *possible to apply* the methodology to other schedulers if theoretical service envelopes are derived. We do not claim that the inference results obtained for particular schedulers (SP, WFQ and EDF) are generalizable to other schedulers. In contrast, one should expect that some schedulers are *statistically* closer to each other than others.

Regarding the arrival traffic, note that we only state that the methodology is generalizable (can be applied) to a non-Gaussian arrival-traffic scenario (even though Gaussian traffic characterization is ideal from the measurement point of view, as pointed out in C.13). However, note that other important assumptions on the arrival traffic, such as stationarity and statistical independence among different classes still hold.

We have added an explanation in the paper to better bring out these points.

———

```
- Sec 5.2 is very persuading!
- Sec 5.3: Measurable regions, do they correspond to service curves?
The use of the "region" is not clear. If there is no traffic, there is no
chance to estimate anything, this by definition clear. Observing the
"configuration" of schedulers in such cases looks like "spying" and may not
be suitable.
```

———

(C.16) No, measurable regions do not correspond to service curves. We have modified the text to describe the measurable regions more clearly. While the reviewer correctly points out that there is no chance to estimate anything if there is no traffic, our goal was to investigate the utilization conditions when one can successfully use passive monitoring instead of active probing to do the inference. The experiment showed that our methodology can be applied under fairly moderate network utilization levels.

———

```
Conclusions: Line 1: "Networks and servers are " now the client is gone?
Clearly, re-work your terminology.
- "System's core QoS mechanisms" come up again, but haven't been defined
since their introduction in the abstract.
```

———

(C.17) We have fixed the above as pointed out by the reviewer.

———

```
- You showed a dedicated case in those sections above, but promised a much
more generic approach. May be you limit your working in the abstract/intro
to examples based on generic terms?
- The conclusions are simply missing, this section is a simple summary
only, even a weak one, since results are summarized in-advance in the
introduction section already.
- Why did you write section 6 at all?
```

———

(C.18) We have revised the conclusions.

———

```
Special issues:
- Abstract, line 5: ".. to assess a (?) systems's  mechanism*s* " ?
- Abstract's abbeviations SP etc. will be clarified too late (page 6,
top).
- Sec 3.1 line 1, " we review a (?) general traffic and service
characterization*s* "?
- Intro paragraphs of section 3 and 4 talk about part A, B, and C each
they are never labelled as such - however, numbers would do better.
- Page 23 was not printable as a PDF (due to errors in the figure).
```

———

(C.19) We have corrected the typos pointed out by the reviewer.

———

4 Review D

```
The paper only addresses the simplified cases, e.g. a
single router and no combination of three disciplines (i.e.
WFQ, SP and EDF). The reviewer wonders if they can
generalize to cases in which the three disciplines are
combined, and the network has multiple nodes.
```

———

(D.1) Reviewer correctly points out that the paper does not explicitly address cases of multiple routers with heterogeneous components (e.g., one EDF router and the next WFQ). There are two cases:

i) If an analytical characterization of the concatenated components is possible, i.e., if we can derive a statistical service envelope, then our approach is extendible to multiple nodes. For example, we have derived a multi-node envelope for CJVC, core stateless -jitter virtual clock, in ToN 2002.

ii) If the multiple elements are simply intractable, we cannot make a rigorous statement. Here, we can merely resort to the common assumption of a 'single bottleneck'. Yet, we do note that a network operator has little incentive to concatenate nodes for which the end-to-end behavior is unknown (such as the EDF/WFQ example is not solved) when well understood alternatives are available.

———

```
The authors should provide some details on how to use
their inference approach in practice. For example, does the
approach need to catch all the packets, or just sample some
ones (in this case, how to do sampling?)?
```

———

(D.2) The approach requires to catch all the packets in flight. However, note that in our implementation of scalable admission control, (http://www.ece.rice.edu/networks/software), we showed that *multi-class* envelope measurement and computations are feasible in a real time. The detailed measurement methodology regarding the network scenario is presented in reference [22], while a detailed description of the measurement architecture in the web-server scenario is presented in [13]. Due to space limitations, we have pointed out an interested reader to the above references for detailed information.

———

```
The inherent limitations of the proposed approaches haven't
been fully discussed. For example, what are the limitations of the Gaussian
approach?
```

———

(D.3) See the discussion in B.11.

———

The discussion of the related work is inadequate.

(D.4) See the answer B.1.

Minor errors: In page 8 and 14, "Part A", "Part B" and "Part C" etc should be updated to the correct section numbers.

(D.5) We have fixed the above as pointed out by the reviewer.
