A Unified Metric for Quality of Service Quantification

Pedro Alipio
Critical Software S.A.
Parque Industrial de Taveiro
3045-504 Coimbra, Portugal

Solange Rito Lima
University of Minho
Department of Informatics
4710-057 Braga, Portugal

Paulo Carvalho*
University of Minho
Department of Informatics
4710-057 Braga, Portugal

ABSTRACT

Internet service providers usually express the quality of network services through a set of values determined according to several network performance parameters periodically collected or measured. However, for common end-users, these values do not give an overall idea of the quality of the network services as they stand for different units and evaluate different perspectives of each service quality. In this context, this paper proposes the definition of a service-oriented unified metric which quantifies a global Quality of Service (QoS) indication by processing standard QoS parameters through a fuzzy controller. The proposed methodology, based on fuzzy logic and tested on Xfuzzy 3.0 platform, allows to close the gap between a high-level QoS perspective and the effective QoS measurements at lower protocolar levels. The definition of a single per-service QoS metric can be useful to simplify control tasks such as QoS routing, SLA negotiation and auditing.

Categories and Subject Descriptors

C.2 [**Computer-Communications Networks**]: Network Operations; I.2 [**Artificial Intelligence**]: Deduction and Theorem Proving

General Terms

Management, Measurement, Performance

Keywords

Quality of service (QoS), QoS metrics, Fuzzy sets, Fuzzy logic

1. INTRODUCTION

The management of today's multiservice networks strongly relies on the assessment and control of each service quality levels. Depending on each service characteristics, the quality of service (QoS) offered to user applications and services is evaluated through a set of specific metrics. The Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) and the IP Performance Metrics (IPPM) working group of the Internet Engineering Task Force (IETF) have been committed on defining concrete metrics for measuring the quality, performance, and reliability of Internet delivery services [3, 4, 8]. The defined set of QoS and performance parameters, although very useful from a traffic engineering point-of-view, are far from the common user perception and understanding, making difficult service negotiation and auditing. In fact, when establishing a Service Level Agreement (SLA) and corresponding Service Level Specifications (SLSs), QoS requirements are frequently expressed by less obvious parameters such as: (i) a delay expressed either as the worst case bound (e.g. delay is less then 100 ms) or as a quantile (e.g. delay is less then 20 ms for 98% of packets during 5 minutes); (ii) a delay variation (jitter) expressed either as the bound or as the quantile; and (iii) a packet loss ratio. SLSs may also include qualitative performance parameters instead or in addition to quantitative parameters. An example of a qualitative parameter is the delay expressed through the linguistic values low, medium or high. The semantics of these parameters and how they are mapped to specific values (or interval) is mainly derived from the QoS definition at the lower level, i.e. they may differ depending on the network infrastructure (e.g. Diffserv, MPLS or ATM).

In this scenario, the present study proposes a novel and simple strategy to derive high-level unified QoS metrics for Internet services resorting to fuzzy logic principles [9]. Attending to the specificity of the problem, which combines the difficulty of handling multiple low-level QoS parameters with the blur boundaries of user perceived QoS, the use of fuzzy logic to achieve a unique per service QoS metric brings a clear advantage and simplicity to the solution. Fuzzy logic is conceptually easy to understand, it is tolerant to imprecise data, it can model sets through non-linear functions of arbitrary complexity and the rules are written using natural language. Thus, it is in fact very suitable to solve these type of problems as it allows mapping measurements into fuzzy sets describing each of the parameter's values and it includes the proper inference mechanisms to reason over rules describing the service requirements resulting on a unified metric expressing the overall quality of each Internet service.

Although several works have resorted to fuzzy theory principles to model QoS control solutions [5, 2], this paper provides a new contribution in the field of QoS measurement and monitoring by proposing a fuzzy controller for mapping multi-metric QoS descriptions into a single value, providing a macro indicator of the service quality.

The remaining of this article has the following structure: first, a brief overview of fuzzy logic is provided in Section 2; the fuzzy controller for generating a unified QoS metric is specified in Sec-

*Corresponding Author (pmc(at)di.uminho.pt)

SIMUTools/QoSim 2009 2-6 March, Rome, Italy
Copyright 2009 ICST ISBN 978-963-9799-45-5.

tion 3, taking the telephony service as an example; and the conclusions are included in Section 4.

2. FUZZY LOGIC OVERVIEW

Fuzzy logic has two different meanings [7]. In a narrow sense, fuzzy logic is a logical system, which is an extension of multivalued logic. But in a wider sense, which is in predominant use today, fuzzy logic is almost synonymous with the theory of fuzzy sets, a theory which relates to classes of objects with blunt boundaries in which membership is a matter of degree. In this perspective, fuzzy logic in its narrow sense is a branch of its wider sense.

Fuzzy logic starts with the concept of a fuzzy set [9]. A fuzzy set is a set without a crisp, clearly defined boundary. It may contain elements with only a partial degree of membership. Degrees of membership are often confused with probabilities. However, they are conceptually distinct; fuzzy truth represents membership in vaguely defined sets, not likelihood of some event or condition. Fuzzy sets are based on vague definitions of sets, not randomness. Formally, a fuzzy set is defined by the following:

Definition A fuzzy set A on a universe U is characterized by a membership function $\mu(x)$ that takes the values in the interval $[0, 1]$. A fuzzy set A in U may be represented as a set of ordered pairs. Each pair consists of a generic element x and its grade of membership, i.e. $A = \{(x, \mu(x))|x \in U\}$ and $\mu(x) = \{\mu_x^1, \mu_x^2, ..., \mu_x^i\}$.

Although sets can overlap in boolean logic, the transition at the border of the set is instantaneous. At the border of the set, the element x is a member of the set or it is not, this is illustrated in Figure 1 (a). As x approaches this border, small changes in x can cause significantly different reactions in the system as x changes from set 1 to set 2. With fuzzy logic this transition at the borders of sets is gradual, allowing for partial membership in both sets, as illustrated in Figure 1 (b). Small changes in x cause a more gradual change in the system output.

2.1 Fuzzy Operators

The basic connective operations in classical set theory are those of intersection, union and complement. These operations on characteristic functions can be generalized to fuzzy sets in more than one way. However, one particular generalization, which results in operations that are usually referred as standard fuzzy set operations, has a special significance in fuzzy set theory. In the following, only the standard operations are introduced by the following formal definitions:

Definition The fuzzy intersection operator $\cap$ (fuzzy AND connective) applied to two fuzzy sets A and B with the membership functions $\mu_A(x)$ and $\mu_B(x)$ is given by

$$\mu_{A\cap B}(x) = min\{\mu_A(x), \mu_B(x)\}, x \in U.$$

Definition The fuzzy union operator $\cup$ (fuzzy OR connective) applied to two fuzzy sets A and B with the membership functions $\mu_A(x)$ and $\mu_B(x)$ is given by

$$\mu_{A\cup B}(x) = max\{\mu_A(x), \mu_B(x)\}, x \in U.$$

Definition The fuzzy complement operator (fuzzy NOT operation) applied to a fuzzy set A with the membership function $\mu_A(x)$ is given by

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x), x \in U.$$

2.2 Hedges

Another important feature of fuzzy systems is the ability to define "hedges," or modifier of fuzzy values. These operations are provided in an effort to maintain close ties to natural language, and to allow for the generation of fuzzy statements through mathematical calculations. As such, the initial definition of hedges and operations upon them will be quite a subjective process and may vary from one implementation to another. Nonetheless, the system ultimately derived operates with the same formality as classic logic. Frequently, the following hedges are defined:

Definition The fuzzy modifier $very$ (or $strongly$) applied to a fuzzy set A with the membership function $\mu_A(x)$ is given by

$$\mu_{veryA}(x) = \mu_A(x)^2, x \in U.$$

Definition The fuzzy modifier $moreorless$ applied to a fuzzy set A with the membership function $\mu_A(x)$ is given by

$$\mu_{moreorlessA}(x) = \mu_A(x)^{\frac{1}{2}}, x \in U.$$

Definition The fuzzy modifier $slightly$ applied to a fuzzy set A with the membership function $\mu_A(x)$ is given by

$$\mu_{slightlyA}(x) = 4 \times \mu_A(x) \times (1 - \mu_A(x)), x \in U.$$

2.3 Rules

Fuzzy sets and fuzzy operators are the subjects and verbs of fuzzy logic. These if-then rule statements are used to formulate the conditional statements that comprise fuzzy logic. A single fuzzy if-then rule assumes the form "if x is A then y is B" where A and B are linguistic values defined by fuzzy sets on the ranges (universes of discourse) X and Y, respectively. The if-part of the rule "x is A" is called the antecedent or premise, while the then-part of the rule "y is B" is called the consequent or conclusion. An example of such a rule might be:

If delay is low **then** service is good

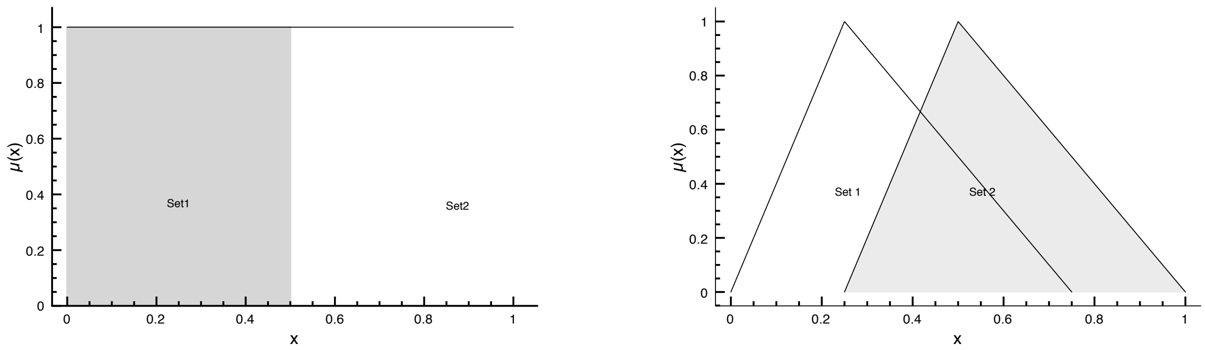
Note that low is represented as a number between 0 and 1, and so the antecedent is an interpretation that returns a single number between 0 and 1. On the other hand, good is represented as a fuzzy set, and so the consequent is an assignment that assigns the entire fuzzy set B to the output variable y. In the if-then rule, the word "is" gets used in two entirely different ways depending on whether it appears in the antecedent or the consequent.

Interpreting an if-then rule involves distinct parts: first evaluating the antecedent (which involves fuzzifying the input and applying any necessary fuzzy operators) and second applying that result to the consequent (known as implication). In the case of two-valued or binary logic, if-then rules do not present much difficulty. If the premise is true, then the conclusion is true. If we relax the restrictions of two-valued logic and let the antecedent be a fuzzy statement, which means that it is true to some degree of membership, then the consequent is also true to that same degree.

The consequent specifies a fuzzy set be assigned to the output. The implication function then modifies that fuzzy set to the degree specified by the antecedent. The most common ways to modify the output fuzzy set are truncation using the min function.

2.4 Defuzzification

These fuzzy outputs need to be converted into a scalar output quantity so that the nature of the action to be performed can be determined by the system. The process of converting the fuzzy output is called defuzzification. Before an output is defuzzified all the fuzzy outputs of the system are aggregated with an union operator by using the union operator. There are many defuzzification techniques. Some of them may only be applied to singleton outputs, others to continuos ones.



(a) Boolean sets

(b) Fuzzy sets

Figure 1: Sets example.

The most used defuzzfiication strategy is the centroid method also known as Center of Area (COA) or Center of Gravity (COG) and it is calculated by the following equation:

$$COG(A) = \sum_{x_{min}}^{x_{max}} x \times A(x) / \sum_{x_{min}}^{x_{max}} A(x)$$

3. THE FUZZY CONTROLLER

In this section, we present the architecture of a unified QoS metric fuzzy controller and the process to obtain a final output consisting of a crisp value expressing the overall QoS for a particular service. The fuzzy controller was developed and tested using the free software platform Xfuzzy 3.0 [6]. As shown in Figure 2, the architecture includes the specification of fuzzy sets and membership functions, input and output variables and operators, rulesets, defuzzification method and the normalization process. At first, the controller reads values into the input crisp variables, then it fuzzifies those values and starts the inference engine. The inference process consists of applying rules and fuzzy operations (*and*, *or* and *implication*) resulting on a set over which a defuzzification method is applied. The result is a crisp value, which is then normalized into the $[0, 1]$ range. This specification is focused on an important network service - IP Telephony - due to its multiconstrained QoS nature. Note that, the present controller and underlying principles can be easily extended and applied to any type of service offered by an ISP just by creating new fuzzy sets and rulesets describing the service QoS specific requirements.

3.1 Variables and Fuzzy Sets

The unified QoS metric fuzzy controller takes throughput, delay, jitter and packet loss ratio as input variables. Throughput is measured in terms of the number of bits transmitted per second. Delay and jitter are measured by the maximum one-way IP packet delay and the maximum IP packet delay variation (IPDV) metrics over a time period, respectively. IPDV is calculated by the difference between the delay of two consecutive IP packets. Packet loss ratio is calculated by the ratio between the number of lost IP packets and the total number of transmitted IP packets over a time period.

The input variables express the network performance parameters and their universe of discourse. Except for delay and jitter, which are measured in seconds, the other input variables have different universes of discourse. As mentioned, throughput is measured in bits per second (bps) and packet loss ratio in percentage. For each of these variables, three fuzzy sets are specified: (i) low; (i) medium and (iii) high. These sets are specified by trapezoid and triangular shaped membership functions defined by the following:

Definition A fuzzy set μ given by a triangular function is defined as

$$\mu_{triangule}(x, a, b, c) = \begin{cases} 0 & , x \leq a \\ (x-a)/(b-a) & , a < x \leq b \\ (c-x)/(c-b) & , b < x \leq c \\ 0 & , x > c \end{cases}$$

Definition A fuzzy set μ given by a trapezoidal function is defined as

$$\mu_{trapezoid}(x, a, b, c, d) = \begin{cases} 0 & , x \leq a \\ (x-a)/(b-a) & , a < x \leq b \\ 1 & , b < x \leq c \\ (d-x)/(d-c) & , c < x \leq d \\ 0 & , x > d \end{cases}$$

These fuzzy sets, illustrated in Figures 3 and 4, define the linguistic values which input and output variables may take and their respective degree of membership. Input variables may be compared with the fuzzy sets low (L), medium (M) and high (H). The output variable (QoS) may be assigned, in the consequent part of the rules, with the linguistic values very low (VL), low (L), medium (M), high (H) or very high (VH) (see Figure 4) in order to obtain a smoother set of QoS output crisp values after defuzzification.

The ranges and transition values of the delay, jitter and packet loss ratio fuzzy sets presented on Figure 3 were defined taking in account the ITU-T recommendations for those QoS parameters. Although the throughput depends on the codification/decodification algorithm (codec), we assume a value of 64kbps as the minimum rate for voice transmission.

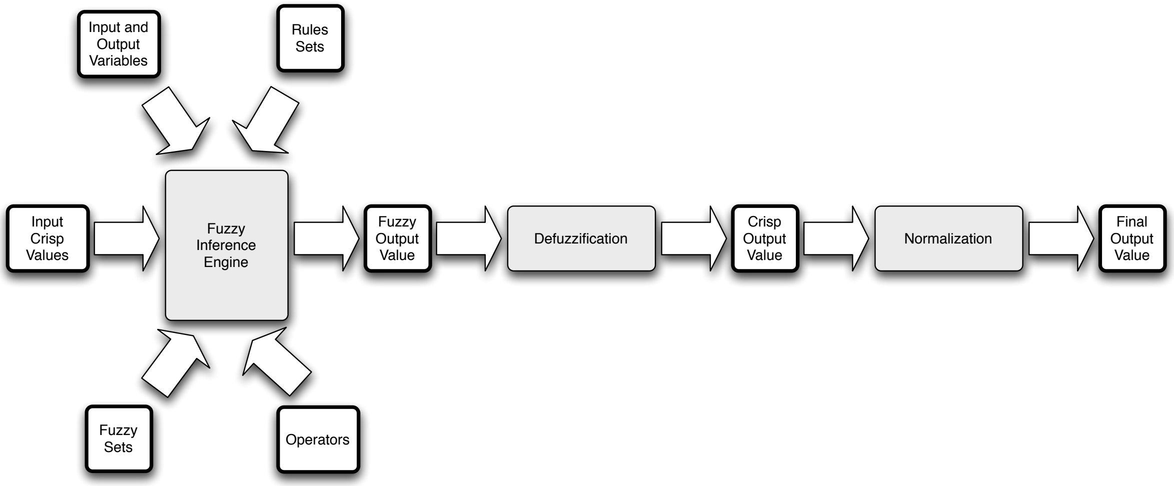


Figure 2: Architecture of the unified QoS metric fuzzy controller.

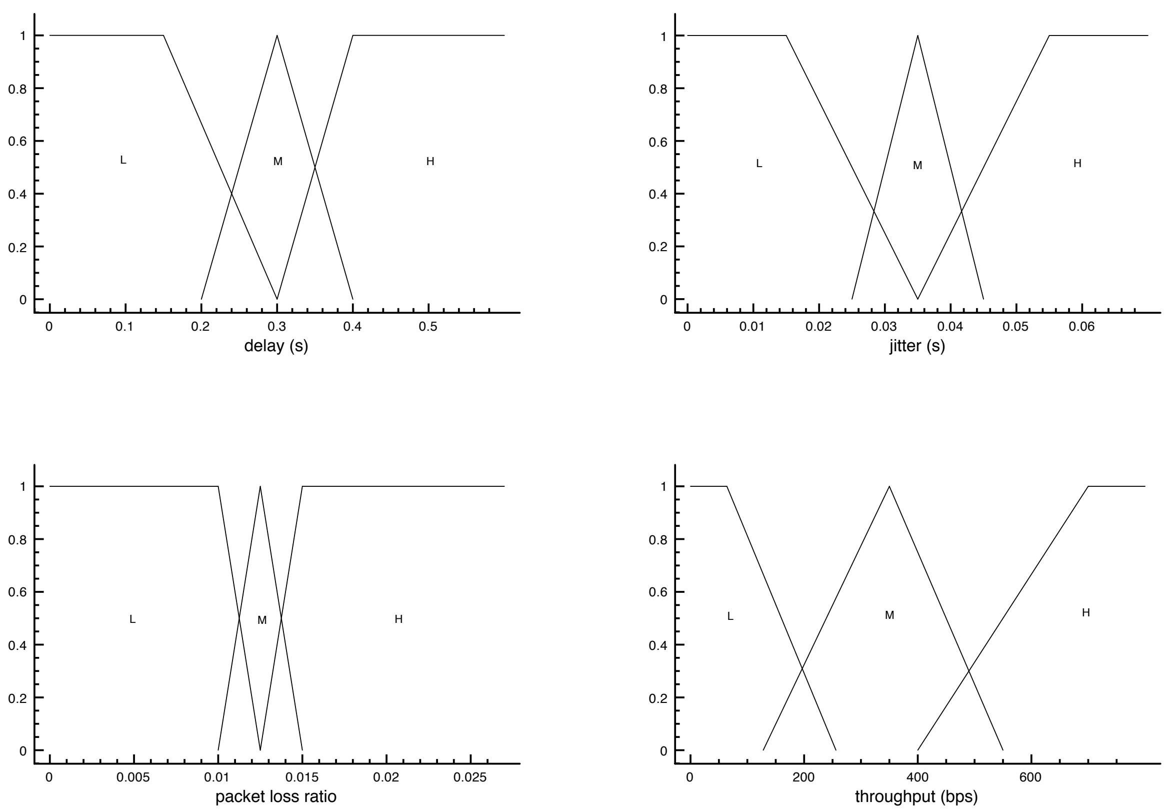


Figure 3: Fuzzy sets for delay, jitter, packet loss and throughput.

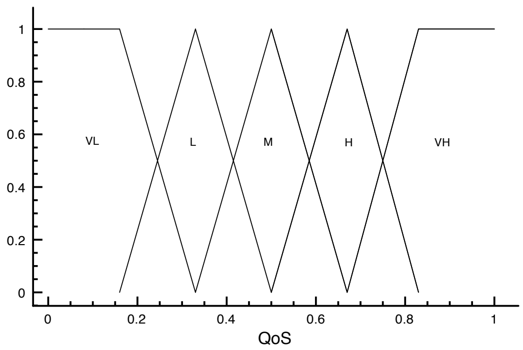


Figure 4: QoS fuzzy sets.

Table 1: Operators and hedges used by the controller.

Operator	Type	Function
and	operator	$min\{\mu_A(x), \mu_B(x)\}$
or	operator	$max\{\mu_A(x), \mu_B(x)\}$
implication, then	operator	$max\{\mu_A(x), \mu_B(x)\}$
>	operator	$x > max\{A(x)\}$
>=	operator	$x >= max\{A(x)\}$
<	operator	$x < min\{A(x)\}$
<=	operator	$x <= min\{A(x)\}$
not	operator	$1 - \mu(x)$
very, strongly	hedge	$\mu(x)^2$
moreorless	hedge	$\mu(x)^{\frac{1}{2}}$
slightly	hedge	$4\mu(x)(1 - \mu(x))$
COA	defuzzification	center of area

3.2 Hedges, Operators and Defuzzification

Table 1 presents the operators, hedges and the defuzzification method used by the controller. In addition to the operators referred in Section 2, ">",">=","<" and "<=", are defined here as "greater than", "greater or equal than", "less than" and "less or equal then", respectively.

3.3 Ruleset for the IP Telephony Service

This ruleset was built taking in account the configuration guidelines for implementing IP telephony on a network with differentiated services [1]. IP telephony is known as a time-sensitive and loss-sensitive service therefore tend to be severely affected by QoS degradation. Thus, the unified QoS metric fuzzy controller for IP telephony shall include the following rules:

```
Rule 1
if
   delay is very low and jitter is very low
   and loss is very low and throughput > low
then
   QoS is veryGood
Rule 2
if
   delay is low and jitter is low and loss is low
   and throughput > low
then
   QoS is good
Rule 3
if
   delay is slightly low and jitter is slightly low
   and loss is slightly low and throughput > low
then
   QoS is medium
Rule 4 if delay >= medium then QoS is bad
Rule 5 if jitter >= medium then QoS is bad
Rule 6 if loss >= medium then QoS is bad
Rule 7 if throughput is low then QoS is bad
Rule 8 if delay > medium then QoS is veryBad
Rule 9 if jitter > medium then QoS is veryBad
Rule 10 if loss > medium then QoS is veryBad
Rule 11 if throughput very low then QoS is veryBad
```

The first rule defines a very good QoS as very low delay, very low jitter, very low packet loss ratio and throughput above low. Throughput does not have to be high because it is possible to get excellent voice quality with a medium bandwidth (about 128kbps) and above. The hedge *very* is applied here to denote a subset of low with a very high membership. The second rule defines a good QoS as low delay, low jitter, low packet loss ratio and the throughput is considered to be as in the former rule. The third rules defines a medium QoS as slightly low delay, slightly low jitter, slightly low packet loss ratio and throughput as above low as the former rules. The hedge *slightly* is used here to express a subset of low with a very low membership. Rule 4, 5, 6 and 7, define bad QoS as a delay, jitter, or packet loss equal or above medium or a low throughput. Finally, a very bad QoS is defined by rules 8, 9, 10, and 11, as delay, jitter or packet loss ratio values grater the medium or very low throughput values. Note that any combination of parameters not included in this ruleset is deducted by the inference process as fuzzy systems may reason over incomplete information, e.g., the resulting QoS for low delay, very low jitter, low packet loss ratio and a medium throughput.

The rulesets for other service classes than IP telephony, i.e. for services not requiring a so tight QoS control, tend to be smaller leading to simpler fuzzy controllers.

3.4 QoS Crisp Values and Normalization

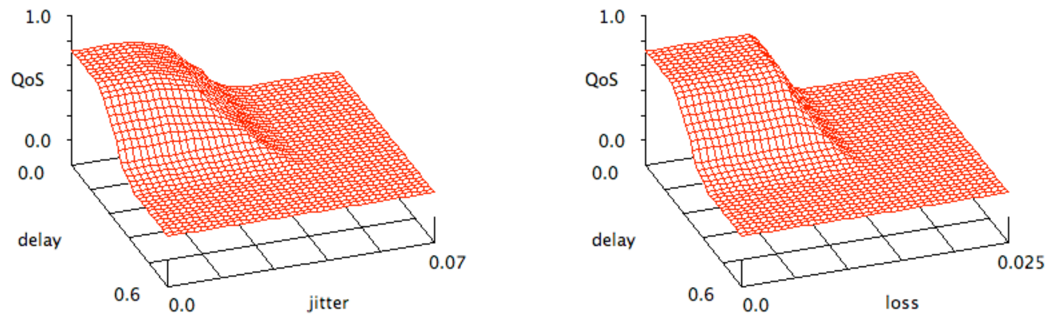
Figure 5 presents 3D surface charts which illustrate the relation between the unified QoS metric and the QoS performance parameters obtained by the COA defuzzification method. As these charts only have three axis, QoS parameters are compared on a pair basis. The parameters which are not represented in each chart take a reference value in order to produce the highest possible QoS, e.g, in Figure 5 (a), loss was considered null and throughput 1Mbps. The surface in each chart represents the universe of values which the QoS unified metric may take.

The normalization process consists of adjusting the resulting universe of values into the interval $[0, 1]$ to be interpreted as an overall percentage of QoS for a particular service. The min-max normalization method is used and it is defined by the following:

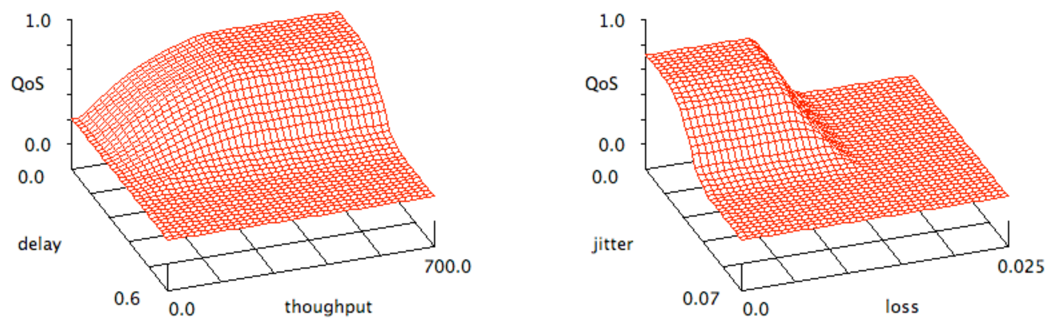
Definition The unified QoS metric normalization is given by the min-max normalization: $v'(i) = (v(i) - Min)/(Max - Min) \times (new\_Max - new\_Min) + new\_Min$ where, $v'(i)$ is the normalized value, $v(i)$ is the defuzzified crisp value, $[Min, Max]$ is the initial range and $[new\_Min, new\_Max]$ is the new range. Thus, as it is intended to obtain values within a interval of $[0, 1]$, replacing new_Min by 0 and new_Max by 1, it results on the following formula: $v'(i) = (v(i) - Min)/(Max - Min)$.

4. CONCLUSION

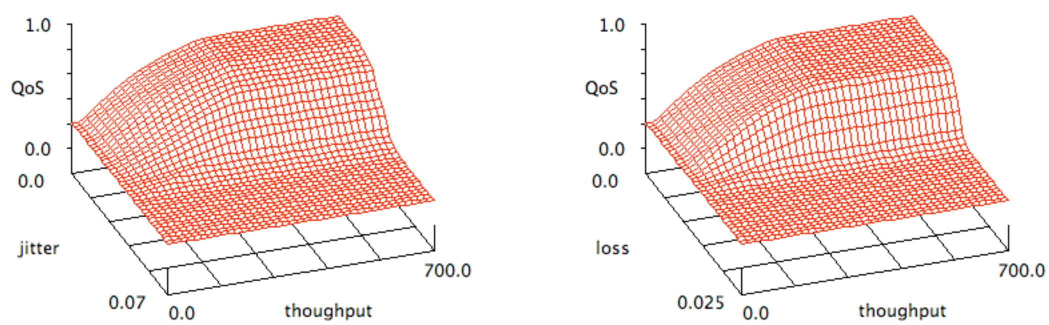
This paper has proposed the definition of a single high-level QoS metric as a way to quantify each service's quality within a multiservice network environment. This metric is obtained by processing standard QoS parameters through a fuzzy controller. The proposed



(a) Delay and jitter; (b) Delay and loss



(c) Delay and throughput; (d) Jitter and loss



(e) Jitter and throughput ; (f) Loss and throughput

Figure 5: Unified QoS metric crisp values.

methodology, based on fuzzy logic, here applied to the IP telephony service, can be easily extended to other network services.

From a practical perspective, the proposed solution may resort to QoS monitoring feedback of each offered service in order to infer about the corresponding unified QoS metric. We believe that this unified metric can be an useful indicator of the overall service quality, as a complement of common QoS metrics. In fact, it may assist QoS management tasks such as QoS routing, service negotiation and auditing, while giving to the user a more straightforward indication of the delivered service quality.

Further work is required to assess the effectiveness of the proposed controller and metric in experimental and real environments, considering a wider range of services and network activity.

5. REFERENCES

[1] BABIARZ, J., CHAN, K., AND BAKER, F. Configuration Guidelines for Diffserv Service Classes. RFC 4594, Aug. 2006. Informational.

[2] DIAO, Y., HELLERSTEIN, J. L., AND PAREKH, S. Optimizing quality of service using fuzzy control. In *DSOM '02: Proceedings of the 13th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management* (London, UK, 2002), Springer-Verlag, pp. 42–53.

[3] GLOSSBRENNER, K. Internet Protocol Data Communication Service - IP Packet Transfer and Availability Performance Parameters. ITU-T Recommendation I.380, 1999.

[4] IPPM-WG, I. IP Performance Measurements Working Group. http:/www.ietf.org/html.charters/ippm-charter.html.

[5] LI, B., AND NAHRSTEDT, K. A control-based middleware framework for quality-of-service adaptations. *Selected Areas in Communications, IEEE Journal on 17*, 9 (Sep 1999), 1632–1650.

[6] LÓPEZ, D., JIMÉNEZ, C., BATURONE, I., BARRIGA, A., AND SÁNCHEZ-SOLANO, S. Xfuzzy: A design environment for fuzzy systems. IMSE-CNM, Spain.

[7] NGUYEN, H., AND WALKER, E. *First course in fuzzy logic*. Boca Raton: Chapman and Hall/CRC Press, 1999.

[8] PAXSON, V., ALMES, G., MAHDAVI, J., AND MATHIS, M. Framework for IP Performance Metrics. RFC 2330 (Informational), May 1998.

[9] ZADEH, L. Fuzzy sets. *Information Control 8* (1965), 338–353.