

QUALITY OF SERVICE FOR PERCEPTUAL CONSIDERATIONS: AN INTEGRATED PERSPECTIVE

G. Ghinea and G.D. Magoulas

Department of Information Systems and Computing
Brunel University
Uxbridge, Middlesex, UB8 3PH, UK

ABSTRACT

The vision of a new generation of network communication architectures, which deliver a *Quality of Service* based on intelligent decisions about the interactions that typically take place in a multimedia scenario, encourages researchers to look at novel ways of matching user-level requirements with parameters characterising underlying network performance. In this paper, we suggest an integrated architecture that makes use of the objective-technical information provided by the designer and the subjective-perceptual information supplied by the user for intelligent decision making in the construction of communication protocols. Thus, this approach, based on the *Analytic Hierarchy Process*, incorporates not only classical Quality of Service considerations, but, indeed, user preferences as well. Furthermore, in keeping with the task-dependent nature consistently identified in multimedia scenarios, the suggested communication protocols also take into account the type of multimedia application, which they are transporting. Lastly, our approach also opens the possibility for such protocols to dynamically adapt based on a changing operating environment.

1. INTRODUCTION

The existence of multiple, generally conflicting objectives and constraints and the considerable uncertainty about the consequences of various alternatives complicate many practical design decisions [1]. In the field of interactive distributed multimedia applications the design process takes place in an environment in which the goals, constraints and consequences of possible actions are not precisely known. The success of such applications will depend on their quality as perceived by the user. This in turn introduces additional problems in the design process, since the system has to also take into account user profiles, the type of applications at hand and their perceptual impact on users, as well as the *Quality of Service* (QoS) provided by the communications network.

Although the problem of multimedia application-level performance is closely linked to both the user perspective of the experience as well as to the service provided by the underlying network, it is rarely studied from an integrated viewpoint. The focus of our research has been the enhancement of the traditional, technical view of QoS with a user-level defined *Quality of Perception* (QoP).

This is a measure, which encompasses not only a user's satisfaction with multimedia clips, but also his/her ability to perceive, synthesise and analyse the informational content of such

presentations. As such, in previous work we have investigated the interaction between QoP and QoS and its implications from both a user perspective [2] as well as a networking angle [3].

Although the problem of multimedia application-level performance is closely linked to both the user perspective of the experience as well as to the service provided by the underlying network, it is rarely studied from an integrated viewpoint. Clearly, this is a very unsatisfactory state of affairs, and indeed the literature itself reveals relatively few instances of research being done in the area of bridging the application-network gap. Usually, three approaches are suggested in the literature.

The first approach tries to bridge the application-network gap *implicitly*. By this it is understood that there is no explicit mapping between application-level user requirements and the QoS provided by the network. What happens rather is that the user specifies, usually through a Graphical User Interface, his/her desired presentation quality [4]. This approach usually assumes a technically-aware user capable of specifying parameters such as desired playback frame rate, spatial resolution, or acceptable synchronisation delay between the audio and video streams.

In the second approach an *explicit* mapping linking application-level user requirements to network QoS is actually given. Such a mapping can either be defined on a per layer basis or directly between application and network-level parameters[5].

The last approach is, in essence, a more restrictive version of the first. What happens here is that the user is played short-duration *probes* of differing qualities of the multimedia material in question and (s)he then specifies which of the given sample qualities is acceptable [6]. Thus, any choice that the user might make in as far as desired multimedia quality goes is guaranteed to be delivered - at least in the initial stages - by the network.

We have addressed the problem of bridging the application-network gap from a multi-attribute decision making perspective. We have sought to use this approach to integrate results from our work on Quality of Perception with the more technical characterisation of Quality of Service, with an ultimate aim of providing an adaptable communications protocol geared towards human requirements in the delivery of distributed multimedia. Accordingly, the structure of this paper is as follows: Section 2 describes the framework that shall be used as a testbed for the new approach. While Section 3 details the elements of the Analytic Hierarchy Process (AHP) necessary for understanding our approach, Section 4 shows how these can be used in practice to obtain a QoP-oriented ordering of QoS parameters. Lastly, conclusions are drawn and possibilities for future work are identified in Section 5.

2. ADAPTATION MECHANISM

The *Dynamically Reconfigurable Stacks Project* (DRoPS) provides an infrastructure for the implementation and operation of multiple adaptable protocols [7]. The core architecture is embedded within the Linux operating system, is accessible through standard interfaces, such as *sockets* and the UNIX *ioctl* (I/O control) system calls, has direct access to network devices and benefits from a protected, multiprogramming environment. The architecture allows additional QoS maintenance techniques, such as flow shaping (to smooth out bursts in traffic), at the user or interface level, and transmission queue scheduling, at the device queue level.

DRoPS based communication systems are composed of fundamental mechanisms, called *microprotocols*, that perform arbitrary protocol processing operations. The complexity of processing performed by a microprotocol is not defined by DRoPS and may range from a simple protocol function, such as a checksum, to a complex layer of a protocol stack, such as TCP. In addition, the protocol mechanisms encapsulated within a microprotocol may be implemented in hardware or software. If appropriate hardware is available, the microprotocol may merely act as a wrapper, calling the relevant hardware function. Like many modular operating systems Linux supports loadable modules. These software objects encapsulate microprotocols, allowing code to be dynamically loaded into a running operating system and executed without the need to recompile a new kernel. Each such microprotocol can be implemented via a number of adaptable functions, summarised in Table 1.

Table 1 Adaptable functionality in DRoPS

Protocol mechanism	Implementations
Sequence control	none complete
flow control	none window based
Acknowledgement scheme	IRQ PM-ARQ
Checksums	none block checking full CRC

A *protocol* defines header formats, private data structures and an unordered set of microprotocols from which communication systems may be fabricated. Individual protocols are differentiated by these characteristics as well as by the semantics of the protocol. Whilst a protocol defines the structure and resources available for constructing a communication system, a *protocol stack* defines a unique instantiation assigned to a particular connection. In terms of microprotocols, a protocol stack is an ordered set drawn from some *parent* protocol and combined to form a functional communication system. Each connection is assigned a protocol stack for its sole use, the configuration of which may vary from other stacks derived from the same parent.

3. THE ANALYTIC HIERARCHY PROCESS

The human nature of the decision-making process makes its formulation very difficult. Note that in the design process, the subjective judgement of the end-user plays a large part in many of the design objectives and constraints, resulting in inherent imprecision. The end-user may not clearly specify a desired parameter value and prefers the use of linguistic phrases to describe the objectives. Some examples of such linguistic phrases are: "A is absolutely more important than B"; "A is equally

important as B"; or more complicated ones, such as "What is the worth of a specific microprotocol in terms of a user adaptivity criterion?". Although information about questions like the previous ones is vital in making correct design decisions, it is very difficult, if not impossible, to quantify them correctly. In other words, the main problem with these pairwise comparisons is how to quantify the linguistic choices selected by the decision-maker during the evaluation of the pairwise comparisons. All methods that use the pairwise comparisons approach eventually express the qualitative answers of a decision-maker into some numbers. Below we briefly describe a well-known approach, proposed by Saaty [1], called the Analytic Hierarchy Process, to deal with this problem. In this approach the decision maker (designer and user) has to express his/her opinion about the value of a single pairwise comparison at a time.

The first step in AHP is a *structuring of the hierarchy*, thus determining the relative criteria and alternatives. To this end, the first level of the hierarchy is used to denote the overall objectives of the decision problem. The second level is occupied by criteria for assessing the accomplishment of the objectives, while the third level contains available actions or alternatives.

The second step in AHP is a *pairing of comparisons* to yield preference weights priorities. The main task of this stage is to determine numerical measures to the relative importance of the criteria and to the relative performance of the alternatives on these criteria. It consists of two sub-procedures:

1. Determine the relative importance of the criteria
2. Determine the relative standing of each alternative with respect to each criterion.

Table 2. Relative importance between activities.

Importance	Characterisation and Explanation
1	Equally important: Two activities contribute equally to the objective
3	Moderately important: Experience and judgement slightly favour one activity over another
5	Essential or strongly important: Experience and judgement strongly favour one activity over another
7	Demonstrated important: An activity is strongly favoured and its dominance has been demonstrate in practice
9	Absolutely important: The evidence favouring one activity over another is of highest possible order of affirmation
2,4,6,8	Intermediate values: When compromise is needed
Reciprocals of above, nonzero	If activity i has one of the above nonzero numbers assigned to it when compared with activity j , then j has the reciprocal value when compared with i

Table 2 presents the basic linguistic terms that can be assigned to a comparison between two activities/criteria. Intermediate terms can also be assigned when compromise is needed between two adjacent characterisations. These pairwise comparisons are quantified by using a scale. Psychological experiments have shown that individuals cannot simultaneously compare more than 7 objects (± 2) [8]. Thus, usually, a scale of nine grades, which describe the relative importance of the criteria, is suggested [1].

Let us define a_{ij} as the numerical judgement made by the designer/user in comparing criterion to criterion. Thus, a *pairwise comparison matrix* $A = [a_{ij}]$ can be constructed, where a_{ij} is an

element of the nine-point scale, i.e. $a_{ij} \in \{1,2,3,\dots,8,9\}$, and also $a_{ji}=1/a_{ij}$ holds [1], [9]. Now, let w_1, w_2, \dots, w_p be the weights for the p criteria, where w_i denotes the relative importance of the i th criterion. Note that, in order to obtain the preference weights of the criteria, w_i can be evaluated by the equation:

$$w_i = \frac{\left(\prod_{j=1}^p a_{ij} \right)^{1/p}}{\sum_{i=1}^p \left(\prod_{j=1}^p a_{ij} \right)^{1/p}} \quad i=1,2,\dots,p \quad (1)$$

and a higher priority setting corresponds to a greater importance.

Pairs among alternatives are compared with respect to the i th criterion and then a weight w_{ji} , which denotes how preferable is the alternative j with respect to the criterion i , is derived. As previously, there is a total of p pairwise comparisons in the matrix.

The final stage of the AHP is a *synthesis of preference weights* to yield composite priorities for alternatives. Here, the weighted product model [10] is used to compare alternative v with alternative u . This can be done by multiplying a number of ratios, one for each criterion. Each ratio is raised to a power, which is equivalent to the relative weight of the corresponding criterion, i.e.

$$P_{v/u} = \prod_{i=1}^p \left(\frac{w_{v,i}}{w_{u,i}} \right)^{w_i} \quad (2)$$

If the ratio $P_{v/u}$ is greater or equal to one then the conclusion is that the alternative v is more preferable than alternative u . In the maximisation case, the best alternative is the one that possesses the highest value among all others.

4. APPLICATION TO QoP MANAGEMENT

We have applied the formalism of the analytic hierarchy process, within the QoP framework, to obtain a method which, from combined user-, application- and network-level requirements, ultimately results in a protocol configuration specifically tailored for the respective requirements. Thus, within the QoP framework, each multimedia application can be characterised by the relative importance of the video (V), audio (A) and textual (T) components as conveyors of information, as well as the dynamism (D) of the presentation. This is in accordance with experimental QoP results which emphasise that multimedia QoP varies with:

- the number and type of media flows
- the type of application, and
- the relative importance of each medium in the context of the application.

On the other hand, there are five network level QoS parameters which we have considered in our model. These are: bit error (BER), segment loss (SL), segment order (SO), delay (DEL) and jitter (JIT). Together with the V, A, T and D parameters these will constitute, in the terminology developed by Saaty, the *criteria* on

the basis of which an appropriate tailored communication protocol is constructed. As has been highlighted before, in DRoPS, the functionality of this communication protocol is realised through a number of microprotocols. We have considered a number of 9 microprotocols, satisfying 4 broad functionality classes, to varying degrees, as given by Table 1.

By applying the AHP we obtain a total of 10 matrices. Nine of these matrices give the relative importance of the various microprotocols (*alternatives*, in Saaty's vocabulary) with respect to the criteria identified in our model, while the last of these matrices details pairwise comparisons between the criteria themselves. An example of one of the former type of matrices, of the different alternatives with respect to one of the criteria (bit error rate in this case) is:

$$ALTwrBER = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1/3 & 1/9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1/3 & 1/9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1/3 & 1/9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1/3 & 1/9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1/3 & 1/9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1/3 & 1/9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1/3 & 1/9 \\ 3 & 3 & 3 & 3 & 3 & 3 & 3 & 1 & 1/5 \\ 9 & 9 & 9 & 9 & 9 & 9 & 9 & 5 & 1 \end{bmatrix} \quad (3)$$

Here the considered microprotocols are, in order {no sequence control, strong sequence control, no flow control, window-based flow control, IRQ, PM-ARQ, no checksum algorithm, block checking, full Cyclic Redundancy Check}. As far as bit error rate is concerned, the only microprotocols that have an impact upon it are the checksum algorithms. The strongest of these algorithms, the full CRC, has the highest weighting in comparison with all the others, while a relatively weak block checking algorithm is considered to be moderately more important (weighting $a_{sj} = 3$) than all microprotocols from the other functionality classes. For a relatively static sequence (such as a news clip), the matrix of each criterion with respect to all the other criteria used in our model is given by:

$$CRwrCR = \begin{bmatrix} 1 & 1/2 & 1 & 1/4 & 1 & 2 & 4 & 4 & 3 \\ 2 & 1 & 1 & 1/4 & 1/3 & 5 & 4 & 5 & 4 \\ 1 & 1 & 1 & 1/3 & 1/2 & 4 & 6 & 4 & 4 \\ 4 & 4 & 3 & 1 & 5 & 6 & 7 & 6 & 5 \\ 1 & 3 & 2 & 1/5 & 1 & 4 & 6 & 6 & 6 \\ 1/2 & 1/5 & 1/4 & 1/6 & 1/4 & 1 & 1/2 & 3 & 2 \\ 1/4 & 1/4 & 1/6 & 1/7 & 1/6 & 2 & 1 & 3 & 2 \\ 1/4 & 1/5 & 1/4 & 1/6 & 1/6 & 1/3 & 1/3 & 1 & 1 \\ 1/3 & 1/4 & 1/4 & 1/5 & 1/6 & 1/2 & 1/2 & 1 & 1 \end{bmatrix} \quad (4)$$

Here, the respective criteria are, in order, $BER, SO, SL, DEL, JIT, V, A, T$ and D .

While all the previous nine matrices considered have a constant form, this last matrix is the only one whose values may fluctuate as a result of changes in the operating environment, as well as a consequence of changes in user preferences and perceptions. An average user, though, would have difficulty in a priori judgment of

varying technical parameters such as delay, jitter, error and loss rates on highly subjective attributes such as perception, understanding and satisfaction. Whilst this is true for QoS attributes at the level of the transport service, users are better able to quantify their requirements in terms of more abstract characteristics like the prioritisation of core multimedia components such as video, audio and text. Thus, the matrix given by (4) reflects this situation and could conceptually be split-up into 4 sub-matrices:

- A 5×5 matrix giving the relative importance of the *BER*, *SO*, *SL*, *DEL* and *JIT* criteria with respect to one another. This matrix will change dynamically during the course of the transmission of a multimedia clip. For example, as a result perhaps of a delay-intolerant audio application being subjected to a period of high network delays, the sub-matrix can reflect this situation by changing the weights to reflect a more radical bias in favour of the delay component.
- A 4×4 matrix, located in the bottom right of the matrix given by (4). Here user input can reflect personal choices of the relative importance of the video, audio and textual components in the context of the application, as well as a relative characterisation of the dynamism of the multimedia clip. These values can also change depending on the particular scene being visualised. Whilst, these values can be changed dynamically, a priori values in this case could reflect the result of consultations with users [2].
- A 5×4 matrix and a 4×5 (one of which is the transpose of the other) which reflect designer choices of the relative importance of the five QoS parameters considered on *V*, *A*, *T* and *D*. In our particular case, the elements of these matrices reflect the results of our previous work on QoP [2].

Below we provide an example that illustrates the applicability of the concepts presented above in a multimedia scenario. Assume that a currently executing BER-sensitive application suddenly experiences deterioration in BER. Then this parameter becomes absolutely important ($a_{ij}=9$) with respect to the others. This value is dynamically updated in the top-left 5×5 matrix. The result of the AHP then shows that the microprotocol that should preferentially be managed under the circumstances is the full CRC - the strongest such microprotocol available to correct the quality loss in BER. This is in contrast to the scenario applicable just before the sudden surge in BER, which gave preference to the strong sequence control microprotocol as a result of applying Relations (1)-(2) on the matrix shown in Relation (4).

5. SUMMARY

We have presented a method of obtaining, in the context of a multimedia application, a priority order of low-level QoS parameters, which would ensure that user level QoP is maintained at an optimum level. In our case, such QoP management would be ensured through dynamic protocol adaptation, as a result of changes in user preferences or the operating environment. Our approach has distinct advantages over previous work. Firstly, since it is already based on the results of extensive QoP tests [2], users themselves do not necessarily need to specify preferences in the interface of an application incorporating the mapping, thereby resulting in a more streamlined viewing process. It thus permits

the maintenance of high-level QoP (without user interference) as low-level QoS varies. Secondly, since the framework includes results linking a human's ability to perceive, analyse and synthesise informational results, it is a more comprehensive approach than preceding attempts. The viability of implementing this approach in practice and using it in a QoP-oriented management scheme for QoS parameters will be further investigated within the DRoPS framework.

6. REFERENCES

- [1] T. Saaty, *The Analytic Hierarchy Process* (New York: McGraw-Hill International, 1980).
- [2] G. Ghinea and J.P. Thomas, QoS Impact on User Perception and Understanding of Multimedia Video Clips, *Proceedings of ACM Multimedia '98*, Bristol, United Kingdom, 1998.
- [3] G. Ghinea, J.P. Thomas and R.S. Fish, Quality of Perception to Quality of Service Mapping Using a Dynamically Reconfigurable Communication System, *Proceedings of IEEE Globecom '99*, Rio de Janeiro, Brazil, 1999.
- [4] D. Reininger, D. Raychaudhuri and M. Ott, A Dynamic Quality of Service Framework for Video in Broadband Networks, *IEEE Network*, November/December 1998, 22 – 34.
- [5] A. Seneviratne, M. Fry, V. Withanan, V. Sapapramdu, A. Richards, and E. Horlait, Quality of Service Management for Distributed Multimedia Applications, *IEEE Communication Society, Proceedings of the Phoenix Conference on Computer and Communications*, Phoenix, Arizona, 1994.
- [6] K. Nahrstedt, K., A. Hossain and S. Kang, A Probe-based Algorithm for QoS Specification and Adaptation, *Proceedings of the 4th International Workshop on QoS (IWQoS)*, Paris, France, 1996.
- [7] R.S. Fish and R.J. Loader, A kernel based adaptable protocol architecture, *Proceedings of the 24th Euromicro Conference*, Västerås, Sweden, 1998.
- [8] G.A. Miller, The magical number seven plus or minus two: some limits on our capacity for processing information, *Psychological Review*, 63, 1956, 81-97.
- [9] E. Triantaphyllou E. and S. Mann, Using the analytic hierarchy process for decision making in engineering applications: some challenges, *International Journal of Industrial Engineering: Applications and Practice*, 2(1), 1995, 35-44.
- [10] S.J. Chen and C.L. Hwang, Fuzzy multiple decision making, *Lecture Notes in Economics and Mathematical Sciences*, 375 (New York: Springer, 1992) Allen J., Hunnicutt S., and Klatt D. *From text to speech: the MITalk system*. MIT Press, Cambridge, MA, 1987.