Noname manuscript No.
(will be inserted by the editor)

A TOPSIS-based QoE Model for Adapted Content Selection of Slide Documents

Habib Louafi · Stéphane Coulombe · Mohamed Cheriet

Received: date / Accepted: date

Abstract In certain platforms, such as Google Docs, documents are adapted for specific mobile device types, and the installation of their applications is required. Although the documents can be accessed via Web browsers, their correctness is not guaranteed. In content selection, the document is adapted into various versions, from which the optimal one, based on a quality of experience (QoE) criterion, is delivered. Existing works evaluate the QoE of each content version using the user's preferences and context parameters, such as device resolution and network bitrate. They ask the user to weight the context parameters, and then combine them using the simple-additive-weighting (SAW) method. However, not all users are familiar with the context parameters, and cannot understand their relationship with the requested content. Besides, not all parameters are compensatory to be summed up. In this paper, we propose a TOPSIS-based QoE model to address the two aforementioned drawbacks. We use the context parameters to define high-level functions understandable by all users, and combine them using the TOPSIS method. Experimental results show the convenience of our QoE model and its reliability over the SAW-based method, as well as the weighting-product (WP) method, which is used as an alternative to the SAW one.

Keywords Mobile computing · QoE · QoS · user preferences · context awareness · content adaptation · SAW · WP · TOPSIS

H. Louafi
Synchromedia Laboratory for Multimedia Communication in Telepresence,
École de technologie supérieure, Université du Québec, Montreal, Canada
E-mail: habib.louafi.1@ens.etsmtl.ca

S. Coulombe
Department of Software and IT Engineering,
École de technologie supérieure, Université du Québec, Montreal, Canada
E-mail: stephane.coulombe@etsmtl.ca

M. Cheriet
Synchromedia Laboratory for Multimedia Communication in Telepresence,
École de technologie supérieure, Université du Québec, Montreal, Canada
E-mail: mohamed.cheriet@etsmtl.ca

Authors' manuscript accepted for publication in *Multimedia Tools and Applications*
https://doi.org/10.1007/s11042-018-5887-7

The final publication is available at link.springer.com

1 Introduction

In certain platforms, such as Google Docs [10], slide documents (e.g., PowerPoint) are shared between users that are often connected with different types of terminals (laptop, smart phone, tablet, etc.). Platforms such as Google Docs [10] and Zoho Show [46] convert uploaded documents into their proprietary formats and wrap them in Web pages that can be rendered by Web browsers [9]. For users connected via mobile devices, the content is however adapted only for specific mobile terminals, and the installation of appropriate applications is required [11,45]. Slides are originally created for standard desktop computers, featuring large screens and resolutions, and having sufficient memory to store the slides and powerful CPU capabilities to process and visualize them. To reach a wide variety of mobile devices, the slide documents must be adapted to mobile device constraints (low resolution, small screen size, low memory) to be accepted. One solution consists in adapting the content (e.g., slide) into several versions, and at delivery time, selecting the best one, according to a quality criterion. Since Web browsers are installed on most mobile devices, the slides are adapted into Web-based documents, to be able to reach a wide range of users and ensure synchronization between the presenter and mobile participants, particularly in conference meetings [2] and mobile learning situations [35]. Moreover, as mobile devices use different communication networks, having different characteristics (bitrate and latency), the delivery time for a slide may vary from one network to another. Usually, the network characteristics are used in selecting the optimal adapted slide from those accepted by the target mobile device. The user preferences are also considered in this process to ensure delivery of the adapted slide that matches the user's needs (e.g., fast delivery, better visual aspect). For instance, a mobile device with a higher screen resolution, and connected to a higher network bitrate, is capable of receiving a rich slide with higher resolution without affecting the end-user waiting time.

Fig. 1 shows a generic content selection architecture for mobile conference meetings. Before the meeting, the slides are uploaded, from which a set of adapted versions are created and stored on a server. During the meeting, when a slide is to be presented and shared between participants, an *optimal content selector* module evaluates the adapted versions and selects the optimal one. To that end, a QoE metric is needed to quantify the quality of each adapted content version.

Research involving content adaptation for mobile devices has led to the notion of *context* (memory, bitrate, screen resolution, etc.), and to the concept of context-aware quality of service, also called QoS awareness or quality of experience (QoE) [23,24, 41,43,42,44,14,27]. This concept has led to the development of metrics to evaluate the quality of the adapted content targeted by mobile devices. In general, different factors should be taken into account when designing a QoE metric; these include content (or service), the end-user (who pays for the service, who uses the service), the service provider, service cost, etc.

QoE models proposed in the literature deal directly with context parameters, such as network bitrate and latency. They model the behavior of the context parameters, weight them using values supplied by the end-user, and sum the weighted values to quantify the quality of adapted content or services.

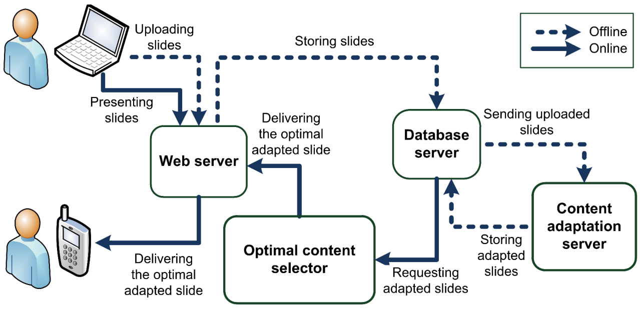


Fig. 1: Generic architecture of a content selection system for mobile conference meetings. The dashed lines represent the operations performed before the meeting, and the solid ones indicate those performed during the meeting.

However, these techniques are not always understandable by all users. Even those who are familiar with these terms, can still not see the relationship between the terms and the quality of the content they request. More importantly, they use the additive technique, which consists in summing the qualities associated with the context parameters, to evaluate the adapted content quality. This evaluation method is not really accurate, but is in fact misleading, as is detailed in section 3.

In this paper, we propose a user-friendly and reliable QoE model. Instead of directly using the context parameters, which we qualify as low-level parameters, the proposed QoE model hides them from the end-user and exposes high-level quality functions that are understandable by all users. Regarding the combination of these quality functions to compute the QoE of each content, we propose the use of well-known scoring method, called TOPSIS [40]. Experimental results show the applicability of the proposed QoE model as well as its effectiveness compared to those proposed in the literature. Our main contributions can be summarized as follows:

- Proposal of a user-friendly QoE model that hides the context parameters from the end-user and exposes high-level quality functions, namely, the visual quality and delivery time. The model is extensible, and other quality functions, such as the battery life, can be considered.
- Proposal of a reliable QoE model based on the TOPSIS method, which is more accurate and well informed than the SAW and WP, which combine the parameters arbitrarily.

The reminder of the paper is organized as follows: In section 2, a mathematical formulation of the problem of optimal adapted content selection is presented. Section 3 reviews previous work related to quality and user preference evaluation in the context of content adaptation. In section 4, we present our proposed QoE model. Sections 5 and 6 present the experimental setup and results. In section 7, we evaluate the complexity and limitations of the proposed QoE model. Lastly, section 8 concludes the paper.

2 Problem statement

Not all mobile devices come with the appropriate applications, such as MS-PowerPoint or OpenOffice-Impress, for reading and visualizing the slides. Even when these applications are available on mobile devices, they cannot ensure synchronization between the presenter and meeting participants. Therefore, it is preferable to convert slides into Web pages capable of being rendered by mobile Web browsers, which are already installed on mobile devices, and can ensure synchronization. In this section, we mathematically formulate the problem of selecting an optimal adapted content version from a set of previously created content.

Let c_k be a presentation slide and $c_k^{\boldsymbol{t}}$ its adapted Web page version created using a transcoding parameter combination vector $\boldsymbol{t}$. In this paper, we consider changing the resolution of the slide, its visual quality, and its format. As a target format, we consider two widely used formats [4]: JPEG, which is a raster format, and XHTML, which is text-based. The first one is very popular on the Internet and supported by nearly all mobile Web browsers, while the second one is the basic format that is understandable by all mobile Web browsers. With these two formats, it is possible to reach a wide variety of mobile devices. Thus, we have:

$$\boldsymbol{t} = (f, z, QF) \tag{1}$$

where $f \in \{\text{JPEG}, \text{XHTML}\}$, $z \in]0, 1]$, and $QF \in]0, 100]$ are the target format into which the slide is adapted, the scaling parameter used to change the resolution of the slide, and the quality factor that determines the visual quality of the adapted slide, respectively. When $f = \text{JPEG}$, each slide is converted into a JPEG image using z and QF and wrapped in a Web page skeleton. When $f = \text{XHTML}$, the components (image and text) of the slide are converted separately and wrapped in a Web page skeleton. In this case, text elements are down-sized in resolution using z, and images are adapted using z and QF. As a result, we have, for each content c_k, a set of adapted contents $\{c_k^{\boldsymbol{t}}\}$, from which we want to identify the optimal version (denoted $c_k^{\boldsymbol{t}^*}$) that can be accepted by the target mobile device D and provides the mobile user with the best quality possible. First, we evaluate the set of adapted contents that can be accepted by D, and then identify the optimal one.

Let $\mathcal{W}_k = \{c_k^{\boldsymbol{t}}\}$ be the set of all the adapted versions of c_k. From $\mathcal{W}_k$, the subset of adapted contents that can be accepted by D are those satisfying its constraints. It is given by:

$$\begin{aligned}\mathcal{W}_k^D = \{c_k^{\boldsymbol{t}} \mid S(c_k^{\boldsymbol{t}}) \leq S(D) \text{ and } W(c_k^{\boldsymbol{t}}) \leq W(D) \text{ and}\\ H(c_k^{\boldsymbol{t}}) \leq H(D) \text{ and } f \in F(D)\}\end{aligned} \tag{2}$$

where $S(c_k^{\boldsymbol{t}})$, $W(c_k^{\boldsymbol{t}})$ and $H(c_k^{\boldsymbol{t}})$ are the data size, width and height of the adapted content $c_k^{\boldsymbol{t}}$, respectively. Similarly, $S(D)$, $W(D)$, $H(D)$ and $F(D)$ represent the maximum accepted file size, maximum resolution width and height, and set of supported formats, respectively.

The optimal adapted content version is given by:

$$c_k^{\boldsymbol{t}^*} = \underset{c_k^{\boldsymbol{t}} \in \mathcal{W}_k^D}{\arg\max}\, \mathcal{Q}_E(c_k^{\boldsymbol{t}}, D) \tag{3}$$

where $\mathcal{Q}_E$ is a QoE measure that evaluates the quality of experience, which represents the core of our proposed QoE model presented in section 4. Before delving into the details of our proposed QoE model, we first review background concepts and discuss important related work.

3 Background and related work

Mobile devices are very diversified, with many different features, such as resolution, memory and screen size. The communication networks (GPRS, LTE, etc.) are diversified as well, and have different bitrate and latency values. To deliver content, such as Web pages or enterprise documents, to these mobile devices, the content must be adapted to satisfy the target mobile device characteristics and network conditions, and provide the mobile user with the best adapted content possible. Some proposed solutions [27] consider only the communication network and ignore the visual aspect of the adapted content. With such solutions, there is no garantee that the delivered content will be appreciated by the end-user. Other proposed solutions [16,18,12], however, focus only on the appearance and rendering of the content on the mobile device, and ignore the communication network conditions. Similarly, such solutions do not guarantee the delivery of the content, and if the content is delivered, it may not be consumed by the target mobile device or may provide a poor end-user experience. Therefore, the mobile device characteristics, including the network conditions, should be used in adapting content. To that end, the device can first be identified from the header of the protocol used by the request (e.g., HTTP UA-header) [38], and then its capabilities are extracted from a local capability database. Alternatively, the standard User Agent Profile (UAProf) can be used [29]. The terminals UAProf description may provide URLs where the capabilities can be retrieved on the Web, or may explicitly provide them [7]. The information collected, which represents the context parameters, is used in computing the best (ideally optimal) adapted content version. The context parameters are first classified into two sets, as follows [41,43,42,44]:

- *Constraint parameters*: This set contains the parameters that should be respected by the adapted content, in order to ensure that it is accepted by the mobile device. For example, the file size of the adapted content is considered a constraint, and should not exceed the mobile device's memory size that is reserved for this purpose.
- *Quality parameters*: In this set, we find the parameters that affect the quality of the adapted content. For instance, the higher the color depth of the adapted content, the greater its quality, up to a point of saturation, after which no quality improvement can be perceived.

This classification is very interesting, however, it is not shared by all researchers, and some parameters are perceived differently in different research works, as we explain in the subsection 3.1. We show how the constraint and quality parameters can be used to evaluate the accepted adapted contents and their qualities.

3.1 Evaluation of the accepted adapted contents

Research [24,41] has shown that first-order logic inference can be used to identify the adapted content set supported by a mobile device. With this method, the characteristics of each adapted content are compared with the features of the target mobile device. For instance, at the very least, for adapted content to be accepted by a mobile device, its file size should not exceed the mobile device memory. For an adapted content version c_k and a target mobile device D, this constraint can be formulated as follows:

$$\text{if } \big(S(c_k) \leq S(D)\big) \text{ return } 1 \text{ else return } 0 \tag{4}$$

where $S(c_k)$ and $S(D)$ are the file size of c_k and memory size of D, respectively. Some researchers [41] consider the bandwidth and network latency as quality parameters (not constraints). To the authors of [24], however, they are constraints, and can be compared with a threshold representing the transmission time tolerated by the end-user.

3.2 Evaluation of the adapted content quality

As discussed above, the quality of the adapted content is affected by various quality parameters. Since the quality parameter values are not necessarily within the same range and do not use the same system of units, most research studies propose to normalize them before evaluation. The normalization process for each parameter is achieved by plotting a curve representing its behavior, with the values on the curve being confined between 0 and 1. The quality of the adapted content is often expressed as a sum of the normalized quality functions [24,41].

Richards et al. [31] have proposed to model the behavior of the quality parameters by means of a family of logarithmic curves. They believe that all the quality parameters can be modeled by logarithmic curves by changing their sensitivity. This belief is not shared within the whole scientific community and other researchers proposed other techniques and curves, such as using a second order curve [24]. In these curves, a minimum point (M) and an ideal point (I) are defined as boundaries. The values that are lower than M are the unsatisfactory ones, and those that are higher than I provide no improvement in user satisfaction. The values between M and I represent increased improvement in perceived satisfaction. After the behavior of each quality parameter is modeled, its value is mapped to its corresponding logarithmic curve, and a normalized value between 0 and 1 is generated. According to the authors of this method, this formula is not unique, and there is no theoretical or empirical support for the use of either this formula or parameter behavior modeling.

Lum et al. [24] propose to model the behavior of each quality parameter using linear or second-order curves. The current value of the quality parameter is mapped to its corresponding curve to produce a normalized value between 0 and 1. For instance, to normalize the color depth behavior, they propose to use the following second-order function:

$$qv = a \cdot qs^2 + b \cdot qs + c \tag{5}$$

where qs is the quality parameter and qv its normalized value.

The authors of this method believe that it is possible to model the majority of the quality parameters, if not all, using linear or second order curves. Once again, there is no theoretical or empirical work supporting their claim.

After the quality parameters are modeled, the normalized values are then weighted by values supplied by the end-user (user preferences). The sum of the weighted values represents the score of that adapted content. However, the idea of directly weighting the quality parameters and summing them to evaluate the adapted content quality is not really an accurate strategy, for at least two reasons:

1. First, not all end-users are able to know the context parameters, especially those of the network (e.g., bandwidth and network latency). Even if this information is known, users may not understand the relationship between these parameters and the quality of the content they requested (e.g., the impact of the bit rate on the battery life). An interesting approach would be to give the end-user a mechanism for expressing his preferences with respect to information that is meaningful to him. For example, the system should ask the user what kind of content he prefers: one providing good visual quality, that can be delivered rapidly, or content that does not rapidly drain the battery. We would describe such a mechanism as being at a high-level of abstraction, in that it deals with information that can be handled by the end-user.
2. Secondly, the scores are weighted and the weights are summed. Some parameters should not be combined in this way because they are not compensatory. For instance, if the bandwidth is close to zero, the adapted content should simply not be delivered. However, in this case, if the values of the other context parameters are high, that of the bandwidth will be compensated, and so the quality score will be high, which is misleading. In fact, this method, called SAW (*Simple Additive Weighting*), is one of a set of scoring methods used in the resolution of MADM (*Multiple Attribute Decision Making*) problems [15]. It can be used in a context where the parameters (attributes) satisfy the so-called compensatory property [20, 8], that is, the loss or gain in one attribute can compensate for a loss or gain in the others. However, this property is not always shared. For instance, it is not shared in terms of color depth and bandwidth, and so compensation is not possible in this case. For MADM problems, such attributes are said to be not comparable [15].

3.3 User preferences evaluation

User preferences are usually considered in the content adaptation process [13,24,41, 43]. They tell the system what the importance of each quality parameter is in the user's perception. They affect the adapted content quality by assigning more or less importance to the quality parameters, which affect the final selection of the optimal adapted content. Most existing research works weight the quality parameters, such as mobile terminal resolution and the network bandwidth and latency, directly. To that end, some researchers use ranking techniques [24], while others use linguistic terms [41], such as "unimportant", "important", and "very important".

However, as explained earlier, not all users understand the context parameters or are able to weight them directly. We qualify this kind of evaluation as a low level evaluation. It is more appropriate to allow the user to express his preferences by means of terms that have real meaning to him, such as: *best visual quality*, *fast content*, and *least energy consuming content*. A similar, but limited solution was proposed by Han et al. [13] to model the user preferences. This solution uses a slide bar that can be adjusted by the end-user to express his preferences regarding the download speed of an adapted image. The boundaries of this slide bar are: slower download (less distillation and higher quality) and fast download (more distillation and lower quality). However, although limited to the download speed, the solution could be generalized to express the QoE by also taking into account the visual aspect.

4 Proposed QoE model

Conversely to QoE models proposed in the literature, we propose a reliable and user-friendly QoE model. First, using the quality parameters, we define high-level quality functions that are understandable by all users. Then, we hide the quality parameters and expose these quality functions to the end-users and ask them to express their preferences regarding these functions. This makes our proposed QoE model user-friendly as the end-users do not need to deal with the quality parameters directly. Secondly, instead of using the SAW method, which is used in the literature to combine quality parameters, we propose to use the TOPSIS method to combine those high-level quality functions. This method renders the proposed QoE model more reliable as compared to those already proposed (see section 6). Although, the TOPSIS method was developed long time ago, to the best of our knowledge, this is the first paper pointing out the weaknesses of the SAW method in content adaptation, and proposing the adaptation and use of the TOPSIS one. More importantly, none of the published papers covering content adaptation has mentioned that the combination method used is a SAW method, and falls under the MADM scoring methods.

We will now define the quality functions. According to Kuipers et al. [19], in the adaptation of audio-visual content, three aspects should be considered, namely:

1. The quality of the content at the source, that is, the quality of the adapted content before delivery.
2. The quality of service (QoS), which is affected by the delivery of the adapted content over the network.
3. The human perception of the adapted content (audiovisual quality, usability, the time required to obtain it, etc.).

From these three aspects, it is clear that QoE is affected by the QoS (term we reserve for networking only), but is not limited by it. At a high-level of abstraction, the QoE of adapted content is affected by its audiovisual quality and the delivery time. The first expresses how the content is appreciated audio-visually, and the second expresses the impact of the total delivery time on the appreciation of the content. Note that this is not the only way of evaluating QoE, and, as explained in [19]; like the QoS evaluation, it constitutes a rich area for research. Based on this work, we consider two high-level functions, namely **visual quality** and **delivery time**, to evaluate the QoE of adapted contents. Their evaluation is presented in the following sub-sections.

4.1 Evaluation of the visual quality

In this paper, we are concentrating on the adaptation of slide documents into JPEG- and XHTML-based Web pages, and as a result, we propose to evaluate the visual quality as follows. Note that this evaluation was first introduced by Louafi et al. [21], and we therefore refer readers to this paper for more details.

$$\mathcal{Q}_V^{\mathrm{SSIM}}(c_k^{\boldsymbol{t}}, D) = \frac{\sum_{i=1}^{n(c_k^{\boldsymbol{t}})} \mathcal{A}(c_{k,i}^{\boldsymbol{t}}) \cdot \mathcal{Q}_V^{\mathrm{SSIM}}(c_{k,i}^{\boldsymbol{t}}, D)}{\sum_{i=1}^{n(c_k^{\boldsymbol{t}})} \mathcal{A}(c_{k,i}^{\boldsymbol{t}})} \tag{6}$$

where $c_{k,i}$ is an embedded component (text or image) of c_k and $c_{k,i}^{\boldsymbol{t}}$ represents its transcoded version using vector $\boldsymbol{t}$. $\mathcal{A}(c_{k,i}^{\boldsymbol{t}})$ is the area it occupies in the adapted slide, and $\mathcal{Q}_V^{\mathrm{SSIM}}(c_{k,i}^{\boldsymbol{t}})$ is its visual quality as measured by the SSIM image quality metric [39]. $n(c_k^{\boldsymbol{t}})$ is the number of embedded components of $c_k^{\boldsymbol{t}}$, which is given by:

$$n(c_k^{\boldsymbol{t}}) = \begin{cases} \text{Number of components within } c_k^{\boldsymbol{t}} & \text{if } f = \text{XHTML} \\ 1 & \text{if } f = \text{JPEG} \end{cases} \tag{7}$$

For convenience, we use the SSIM image quality metric [39] but other metrics could be selected as well. Assuming that the visual quality of text components will not be affected when adapted (scaled down), we evaluate $\mathcal{Q}_V^{\mathrm{SSIM}}(c_{k,i}^{\boldsymbol{t}})$ by:

$$\mathcal{Q}_V^{\mathrm{SSIM}}(c_{k,i}^{\boldsymbol{t}}) = \begin{cases} \mathrm{SSIM}(c_{k,i}^{\boldsymbol{t}}, c_{k,i}) & \text{if } c_{k,i}^{\boldsymbol{t}} \text{ is an image} \\ 1 & \text{if } c_{k,i}^{\boldsymbol{t}} \text{ is text} \end{cases} \tag{8}$$

where the SSIM is measured between the original component and its transcoded version, both scaled to a specific resolution (that of the target mobile terminal - called viewing conditions). A full discussion of the viewing conditions can be found in [5, 6].

To address the third requirement in designing a QoE metric [19], the obtained $\mathcal{Q}_V$ values were mapped to their corresponding MOS values (mean opinion score), which are more appropriate for representing the human perception. The mapping of SSIM to MOS (or DMOS) using logistic functions is well-known in the field of objective quality assessment [34]. Therefore, the obtained MOS values of each slide can be interpreted as follows: from 0 to 0.2 : Bad, from 0.2 to 0.4 : Poor, from 0.4 to 0.6 : Fair, from 0.6 to 0.8 : Good, and from 0.8 to 1.0 : Excellent.

Therefore, we have:

$$\mathcal{Q}_V(c_{k,i}^{\boldsymbol{t}}) = \mathrm{MOS}(\mathcal{Q}_V^{\mathrm{SSIM}}(c_{k,i}^{\boldsymbol{t}})) \tag{9}$$

4.2 Evaluation of the delivery time

Using the data size of the adapted content and the network characteristics, we evaluate the delivery time as follows:

$$\mathcal{T}_{D}(c_k^{\boldsymbol{t}}, D) = \frac{S(c_k^{\boldsymbol{t}})}{N_{B}(D)} + N_{L}(D) \tag{10}$$

where $S(c_k^{\boldsymbol{t}})$, $N_{B}(D)$, and $N_{L}(D)$ are the adapted content file size, the network bitrate and latency, respectively.

4.3 User preferences articulation

Rather than asking the end-user to weight the quality parameters directly, we propose to ask the user to weight or express his preferences regarding the proposed high-level quality functions ($\mathcal{Q}_{V}$ and $\mathcal{T}_{D}$). Technically, in this case, for both quality functions, the end-user can be asked to enter ranked values (e.g., visual quality: 80% and delivery time: 20%). Alternatively, he can be asked to select linguistic expressions (e.g., visual quality: *very sensitive* and delivery time: *less sensitive*). Note that when linguistic expressions are used, they are converted into numerical values, using fuzzy functions, for example, as proposed in [41].

For simplicity, in this paper, we use the weight ranking technique. To express the relative importance between the quality functions, we propose to define the weights as follows:

$$\begin{aligned} w_{V}(D) + w_{T}(D) = 1 \\ 0 \leq w_{V}(D), w_{T}(D) \leq 1 \end{aligned} \tag{11}$$

where $w_{V}(D)$ and $w_{T}(D)$ are the weight values expressing the user preferences regarding $\mathcal{Q}_{V}$ and $\mathcal{T}_{D}$, respectively.

4.4 QoE evaluation

The set of transcoding parameter combinations obtained by varying $\boldsymbol{t} = (f, z, QF)$ leads to an infinite solution space when considering fractional resolution scalings. Therefore, in practice, we use quantization to limit the solution space while spanning a wide variety of adapted contents. In this paper, we use the following quantized values of z and QF:

$$\widetilde{z} = \{0.1, 0.2, \ldots, 1\}, \;\; \widetilde{QF} = \{10, 20, \ldots, 100\} \tag{12}$$

Let $\widetilde{\boldsymbol{t}} = (f, \widetilde{z}, \widetilde{QF})$ be the quantized transcoding parameter combinations. Thus, we note by $\widetilde{\mathcal{W}_k}$ the quantized solution space, and by $\widetilde{\mathcal{W}_k^D}$ the space of the quantized solutions that can be accepted by D (see (2)).

To evaluate the QoE of each adapted content, we use the TOPSIS method, which consists in computing the solution that is the closest to an ideal point and the farthest from a negative ideal point. It includes six steps and herein the adaptation of these steps to the problem at hand.

Step 1: Normalization of the $\mathcal{Q}_V$ and $\mathcal{T}_D$ of each adapted document $c_k^{\tilde{\boldsymbol{t}}}$ from $\widetilde{\mathcal{W}}_k^D$ as follows:

$$\begin{aligned} \mathcal{Q}_V^n(c_k^{\tilde{\boldsymbol{t}}}, D) &= \mathcal{Q}_V(c_k^{\tilde{\boldsymbol{t}}}, D) \Bigg/ \sqrt{\sum_{c_k^{\tilde{\boldsymbol{t}}} \in \widetilde{\mathcal{W}}_k^D} (\mathcal{Q}_V(c_k^{\tilde{\boldsymbol{t}}}, D))^2} \\ \mathcal{T}_D^n(c_k^{\tilde{\boldsymbol{t}}}, D) &= \mathcal{T}_D(c_k^{\tilde{\boldsymbol{t}}}, D) \Bigg/ \sqrt{\sum_{c_k^{\tilde{\boldsymbol{t}}} \in \widetilde{\mathcal{W}}_k^D} (\mathcal{T}_D(c_k^{\tilde{\boldsymbol{t}}}, D))^2} \end{aligned} \tag{13}$$

where $\mathcal{Q}_V^n$ and $\mathcal{T}_D^n$ are the normalized values of $\mathcal{Q}_V$ and $\mathcal{T}_D$ respectively.

Step 2: Weighting the normalized $\mathcal{Q}_V$ and $\mathcal{T}_D$.

$$\begin{aligned} \mathcal{Q}_V^w(c_k^{\tilde{\boldsymbol{t}}}, D) &= w_V(D) \cdot \mathcal{Q}_V^n(c_k^{\tilde{\boldsymbol{t}}}, D) \\ \mathcal{T}_D^w(c_k^{\tilde{\boldsymbol{t}}}, D) &= w_T(D) \cdot \mathcal{T}_D^n(c_k^{\tilde{\boldsymbol{t}}}, D) \end{aligned} \tag{14}$$

where $\mathcal{Q}_V^w$ and $\mathcal{T}_D^w$ are the weighted normalized $\mathcal{Q}_V$ and $\mathcal{T}_D$ respectively.

Step 3: Determining the ideal and negative ideal solutions.

The ideal solution, $(\mathcal{Q}_V^o, \mathcal{T}_D^o)$, is evaluated by:

$$\begin{aligned} \mathcal{Q}_V^o(c_k^{\tilde{\boldsymbol{t}}}, D) &= \max_{c_k^{\tilde{\boldsymbol{t}}} \in \widetilde{\mathcal{W}}_k^D} \left(\mathcal{Q}_V^w(c_k^{\tilde{\boldsymbol{t}}}, D) \right) \\ \mathcal{T}_D^o(c_k^{\tilde{\boldsymbol{t}}}, D) &= \min_{c_k^{\tilde{\boldsymbol{t}}} \in \widetilde{\mathcal{W}}_k^D} \left(\mathcal{T}_D^w(c_k^{\tilde{\boldsymbol{t}}}, D) \right) \end{aligned} \tag{15}$$

The negative ideal solution, $(\mathcal{Q}_V^-, \mathcal{T}_D^-)$, is evaluated by:

$$\begin{aligned} \mathcal{Q}_V^-(c_k^{\tilde{\boldsymbol{t}}}, D) &= \min_{c_k^{\tilde{\boldsymbol{t}}} \in \widetilde{\mathcal{W}}_k^D} \left(\mathcal{Q}_V^w(c_k^{\tilde{\boldsymbol{t}}}, D) \right) \\ \mathcal{T}_D^-(c_k^{\tilde{\boldsymbol{t}}}, D) &= \max_{c_k^{\tilde{\boldsymbol{t}}} \in \widetilde{\mathcal{W}}_k^D} \left(\mathcal{T}_D^w(c_k^{\tilde{\boldsymbol{t}}}, D) \right) \end{aligned} \tag{16}$$

Step 4: Computing the Euclidean distance of each adapted content from the ideal solution and from the negative ideal solution. They are given by:

$$\begin{aligned} d^o(c_k^{\tilde{\boldsymbol{t}}}, D) &= \sqrt{\left(\mathcal{Q}_V^o(c_k^{\tilde{\boldsymbol{t}}}, D) - \mathcal{Q}_V^w(c_k^{\tilde{\boldsymbol{t}}}, D)\right)^2 + \left(\mathcal{T}_D^o(c_k^{\tilde{\boldsymbol{t}}}, D) - \mathcal{T}_D^w(c_k^{\tilde{\boldsymbol{t}}}, D)\right)^2} \\ d^-(c_k^{\tilde{\boldsymbol{t}}}, D) &= \sqrt{\left(\mathcal{Q}_V^-(c_k^{\tilde{\boldsymbol{t}}}, D) - \mathcal{Q}_V^w(c_k^{\tilde{\boldsymbol{t}}}, D)\right)^2 + \left(\mathcal{T}_D^-(c_k^{\tilde{\boldsymbol{t}}}, D) - \mathcal{T}_D^w(c_k^{\tilde{\boldsymbol{t}}}, D)\right)^2} \end{aligned} \tag{17}$$

where d^o and d^- are the Euclidean distances from the ideal solution and from the negative ideal solution, respectively.

Step 5: Computing the relative closeness (C^*) of each adapted content to the ideal solution:

$$C^*(c_k^{\tilde{\boldsymbol{t}}}, D) = \frac{d^-(c_k^{\tilde{\boldsymbol{t}}}, D)}{d^o(c_k^{\tilde{\boldsymbol{t}}}, D) + d^-(c_k^{\tilde{\boldsymbol{t}}}, D)} \tag{18}$$

Step 6: Considering the $\mathcal{Q}_E$ of each adapted content as its relative closeness to the ideal solution (C^*), we update (3) as follows.

$$c_k^{\widetilde{\boldsymbol{t}^*}} = \arg\max_{c_k^{\tilde{\boldsymbol{t}}} \in \widetilde{\mathcal{W}}_k^D} C^*(c_k^{\tilde{\boldsymbol{t}}}, D) \tag{19}$$

Then, instead of solving (3), we solve (19) on the accepted adapted content solution space $\widetilde{\mathcal{W}}_k^D$. That is, we evaluate the closeness, $C^*(c_k^{\tilde{\boldsymbol{t}}}, D)$, of all the accepted adapted content versions, which represents their respective $\mathcal{Q}_E$ values, and then select the adapted content version that yields the highest closeness value.

4.5 Content selection process

In this section, we describe the *optimal adapted content selector* module presented in Fig. 1, which uses the proposed QoE model to select the optimal adapted slide from the set of adapted slides stored in a database server. The different modules that comprise the content selector module are presented in Fig. 2. The user preferences, the quality, and constraint parameters are extracted from the user's request. On the other hand, for each slide, the set of its adapted versions are selected (one by one) from the database server, and their file sizes and resolutions are extracted and validated against the constraint parameters to select only those that can be accepted by the target mobile device. We use the file sizes and resolutions of the accepted adapted slides and the extracted quality parameters to evaluate their visual quality $\mathcal{Q}_V$ and delivery time $\mathcal{T}_D$. The latter, as well as the extracted user preferences, are used to evaluate the QoE ($\mathcal{Q}_E$) of each accepted adapted slide. Lastly, the adapted slide yielding the highest $\mathcal{Q}_E$ value is selected as the optimal one and sent to the Web server for delivery.

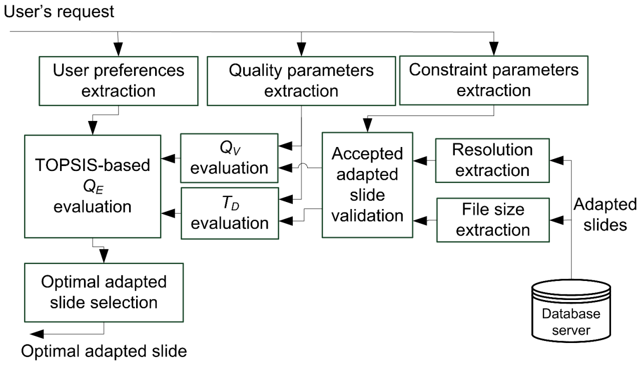

User's request
User preferences extraction
Quality parameters extraction
Constraint parameters extraction
TOPSIS-based Q_E evaluation
Q_V evaluation
T_D evaluation
Accepted adapted slide validation
Resolution extraction
File size extraction
Adapted slides
Database server
Optimal adapted slide selection
Optimal adapted slide

Fig. 2: The content optimizer.

5 Experimental setup

In this section, we show the applicability and the performance of the proposed QoE model in the adaptation of presentation slides. To that end, we developed a Java-based application that uses OpenOffice APIs to create programmatically (not manually) a set of OpenOffice Impress documents. Each of them was comprised of one slide, itself comprised of a text component and an image. The text and images were collected from the Internet, and their positions in the slide were set randomly. To cover a wide variety of slide characteristics, different values representing the percentage of the areas occupied by text (T) and image (I) components in the slide were used, as follows [21]:

$$\begin{aligned} \mathrm{T} &\in \{0\%, 10\%, 20\%, \ldots, 100\%\} \\ \mathrm{I} &\in \{0\%, 10\%, 20\%, \ldots, 100\%\} \end{aligned} \tag{20}$$

Using OpenOffice JPEG and XHTML filters, each document was adapted by varying the transcoding parameters $f \in \{\text{JPEG}, \text{XHTML}\}$, $\widetilde{z} \in \{0.1, 0.2, \ldots, 1\}$ and $\widetilde{QF} \in \{10, 20, \ldots, 100\}$. Thus, for each slide c_k, we created a set of adapted slides (Web pages), $\widetilde{\mathcal{W}}_k$. Thus, we have:

$$\widetilde{\mathcal{W}}_k = \{c_k^{\widetilde{\boldsymbol{t}}}\}_{\widetilde{\boldsymbol{t}}=(f,\widetilde{z},\widetilde{QF})} \tag{21}$$

Let D be a mobile device connected to an LTE communication network that has a bitrate of $N_B(D) = 6$ Mbps and a latency of $N_L(D) = 25$ ms [3].

We assume that all the adapted contents of $\widetilde{\mathcal{W}}_k$ satisfy the constraints of D expressed in (2). Thus, we have $\widetilde{\mathcal{W}}_k = \widetilde{\mathcal{W}}_k^D$.

For each slide c_k, we evaluate the visual quality and delivery time of its adapted versions $\{c_k^{\widetilde{\boldsymbol{t}}}\}$, using (9) and (10), respectively. Then, using (19), we evaluate their quality of experience and select the optimal one $c_k^{\widetilde{\boldsymbol{t}}^*}$.

It would be very interesting to compare our proposed QoE model with other TOPSIS-based methods. However, as we mentioned earlier, TOPSIS has never been applied to content adaptation. Thus, we validate the performance of the proposed QoE model with two less complex scoring methods that do not require extensive computations, namely SAW (simple additive weighting) and WP (weighting product) [40]. These two methods are widely used in various research domains [23–25, 41,32,22,33,17,30,1], and also in content adaptation [23–25,41]. The WP method has been used in [21] as an alternative to the SAW method, providing more accurate results. Therefore, as previously stated, for each adapted slide $c_k^{\boldsymbol{t}}$, we evaluate its visual quality, delivery time and quality of experience, using the a SAW- and WP-based methods.

5.1 SAW-based method

This method consists of two steps: normalization of $\mathcal{Q}_{_V}$ and $\mathcal{T}_{_D}$, then evaluation of $\mathcal{Q}_{_E}$. The normalization breaks down as follows:

$$\begin{aligned} \mathcal{Q}^n_{_V}(\widetilde{c^{\boldsymbol{t}}_k}, D) &= \mathcal{Q}_{_V}(\widetilde{c^{\boldsymbol{t}}_k}, D) \Big/ \max_{\widetilde{c^{\boldsymbol{t}}_k} \in \widetilde{\mathcal{W}}^D_k} (\mathcal{Q}_{_V}(\widetilde{c^{\boldsymbol{t}}_k}, D)) \\ \mathcal{T}^n_{_D}(\widetilde{c^{\boldsymbol{t}}_k}, D) &= \min_{\widetilde{c^{\boldsymbol{t}}_k} \in \widetilde{\mathcal{W}}^D_k} (\mathcal{T}_{_D}(\widetilde{c^{\boldsymbol{t}}_k}, D)) \Big/ \mathcal{T}_{_D}(\widetilde{c^{\boldsymbol{t}}_k}, D) \end{aligned} \tag{22}$$

where $\mathcal{Q}^n_{_V}$ and $\mathcal{T}^n_{_D}$ are the normalized values of $\mathcal{Q}_{_V}$ and $\mathcal{T}_{_D}$ respectively. Note that this is not the only way normalization can be done [26].

The $\mathcal{Q}_{_E}$ evaluation is given by:

$$\mathcal{Q}^{\text{SAW}}_{_E}(\widetilde{c^{\boldsymbol{t}}_k}, D) = w_{_V}(D) \cdot \mathcal{Q}^n_{_V}(\widetilde{c^{\boldsymbol{t}}_k}, D) + w_{_T}(D) \cdot \mathcal{T}^n_{_D}(\widetilde{c^{\boldsymbol{t}}_k}, D) \tag{23}$$

The solution yielding the highest $\mathcal{Q}_{_E}$ value is selected as the optimal one.

5.2 WP-based method

In the WP method the normalization is not required [26] and the $\mathcal{Q}_{_E}$ of each adapted slide $\widetilde{c^{\boldsymbol{t}}_k}$ is evaluated as follows:

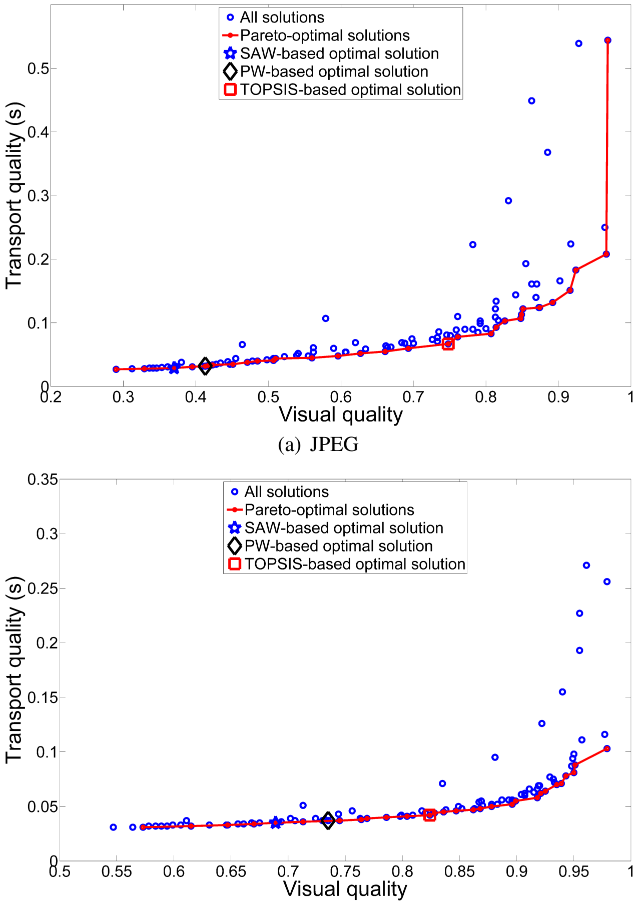
$$\mathcal{Q}^{\text{WP}}_{_E}(\widetilde{c^{\boldsymbol{t}}_k}, D) = \Big(\mathcal{Q}_{_V}(\widetilde{c^{\boldsymbol{t}}_k}, D)\Big)^{w_V(D)} \cdot \Big(\mathcal{T}_{_D}(\widetilde{c^{\boldsymbol{t}}_k}, D)\Big)^{-w_T(D)} \tag{24}$$

Similarly, the optimal solution is the one that possesses the highest $\mathcal{Q}_{_E}$ value.

6 Experimental results

6.1 Selected document example

To graphically illustrate the optimal solutions obtained by the three methods under consideration, we selected a document example. In Figs. 3(a) and 3(b), we show the optimal solutions obtained for JPEG and XHTML, respectively. In this example, it is clear that both SAW- and WP-based methods tend to select solutions with lower delivery times, which is good; however, this forces the visual quality to be very low. With respect to the ideal point of $(\mathcal{Q}_{_V} = 1, \mathcal{T}_{_D} = 0)$ (also called utopia in multi-objective optimization), the SAW- and WP-based methods provide optimal solutions that deviate significantly from the ideal point. On the other hand, as shown in these figures, the TOPSIS-based method provides optimal solutions that are closer to this ideal point, and balanced between $\mathcal{Q}_{_V}$ and $\mathcal{T}_{_D}$. This is quite reasonable, because in this scenario, we are using the same weight value for both quality functions $(w_{_V}(D) = w_{_T}(D) = 0.5)$. This behavior is expected, as the proposed TOPSIS-based method is designed to identify the solution closest to an ideal point $(\mathcal{Q}_{_V} = 1, \mathcal{T}_{_D} = 0)$ and the farthest from a negative ideal point $(\mathcal{Q}_{_V} = 0, \mathcal{T}_{_D} = \text{highest value})$, whereas with the SAW- and WP-based methods, the quality functions are combined arbitrarily.



(a) JPEG

(b) XHTML

Fig. 3: Results of the selected document obtained with $w_V(D) = 0.5$ and $w_T(D) = 0.5$.

6.2 Absolute deviation from utopia

To study the overall performance of the proposed QoE model, we compare the optimal solutions obtained with the three methods under consideration with the utopia point, which represents the ideal target. Therefore, we evaluate the weighted absolute deviation between the optimal solutions and utopia. Note that the absolute deviation is used only for illustration purposes, and other metrics, such as the Euclidean distance, can be used. As the TOPSIS-based method uses the Euclidean distance (see (17)), we opted for the weighted absolute deviation to avoid biasing the validation. It is evaluated as follows:

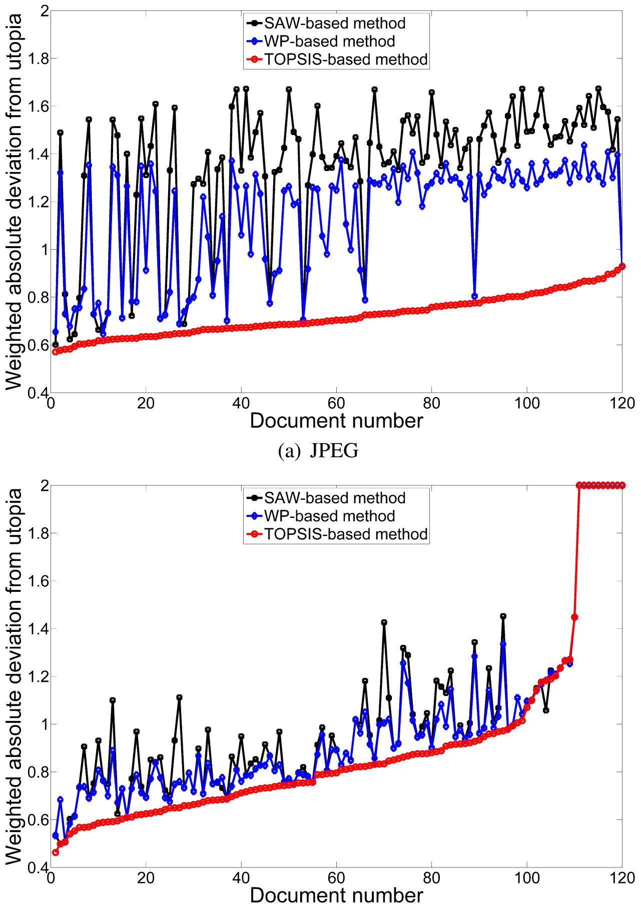
$$\mathrm{DEV}(\widetilde{c_k^{\boldsymbol{t}}}, y^o) = \left| \frac{\mathcal{Q}_{V}(\widetilde{c_k^{\boldsymbol{t}}}, D) - \mathcal{Q}_{V}(y^o)}{w_{V}(D)} \right| + \left| \frac{\mathcal{T}_{D}(\widetilde{c_k^{\boldsymbol{t}}}, D) - \mathcal{T}_{D}(y^o)}{w_{T}(D)} \right| \qquad (25)$$

where y^0 represents the utopia point with $\mathcal{Q}_{V}(y^o) = 1$ and $\mathcal{T}_{D}(y^o) = 0$.

The obtained results, computed for $w_{V}(D) = 0.5$ and $w_{T}(D) = 0.5$, are plotted in Figs. 4(a) and 4(b) for JPEG and XHTML, respectively. We observe that overall, our model presents optimal solutions closer to utopia than those obtained with the SAW- and WP-based methods. They are much closer in the case of XHTML. The SAW-based method, which is mostly used in the literature, behaves very poorly, whereas the WP-based one shows a variable behavior. Thus, overall, both SAW- and WP-based methods are not reliable, and should not be trusted and implemented in practical applications. On the other hand, the proposed TOPSIS-based model provides very reliable results, close to the ideal target (utopia).

6.3 Analysis of the obtained $\mathcal{Q}_{V}$ and $\mathcal{T}_{D}$

In this section, we analyze the optimal $\mathcal{Q}_{V}$ and $\mathcal{T}_{D}$ obtained by each method. We want to see if the obtained optimal solutions can improve the end user experience. To make this analysis graphically clear, we plot the visual quality and delivery time separately. Figs. 5(a) and 6(a) respectively show the optimal $\mathcal{Q}_{V}$ and $\mathcal{T}_{D}$ obtained with the SAW-, WP- and TOPSIS-based methods when the output format is JPEG. Those obtained for XHTML are presented in Figs. 5(b) and 6(b). We observe that the optimal $\mathcal{Q}_{V}$ obtained by the TOPSIS-based method are *Good*; between 0.65 and 0.92 for JPEG and between 0.68 and 1 for XHTML. On the other hand, those obtained by the SAW- and WP-based methods are much smaller overall, and highly variable. For JPEG, we note that many interesting points were reached by our TOPSIS-based method, and not by the SAW- and WP-based methods. For instance, in each group of ten documents, the first three or four obtained by the TOPSIS-based method present very low delivery times and higher visual quality at the same time (see Figs. 5(a) and 6(a)). Some of them present the same delivery time values for three methods, but much higher visual quality values for the TOPSIS-based method. For the other documents, it was not possible to reach a higher visual quality and lower delivery time simultaneously, compared to the other methods. The two other methods could not reach even those interesting points that we reached with our proposed method. Thought, in general, the delivery time obtained with the TOPSIS-based method is higher compared to that obtained with the SAW- and WP-based method, it is still very low and improves the QoE. This is based on studies performed in estimating the tolerable waiting time in Web browsing [28]. In [28], the authors found that up to 5 seconds of waiting time, the user is fully satisfied with the Web application, and between 5 and 10 seconds, the user satisfaction starts decreasing gradually. Beyond 10 seconds, the user is not satisfied with the application and starts doing something else, such as restarting or aborting the application. As shown in Figures 6(a) and 6(b), the delivery time we obtained is under 0.14 and 0.06 seconds for JPEG and XHTML, respectively, which is very appealing.

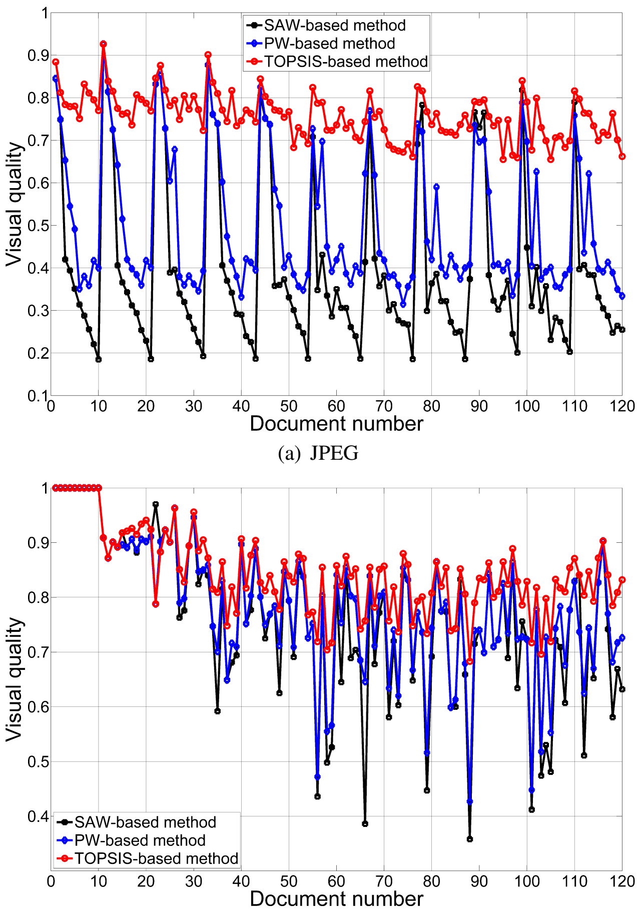


(a) JPEG

(b) XHTML

Fig. 4: Weighted absolute deviation from utopia, obtained with $w_V(D) = 0.5, w_T(D) = 0.5$. The documents are sorted according to the results of the TOPSIS-based method.

In the case of XHTML, we observe that the visual quality values obtained with the TOPSIS-based method are higher than those obtained for JPEG. Also, the delivery time values obtained with the TOPSIS-based method are lower for XHTML than those obtained for JPEG. This is due to the visual and delivery time evaluation (see (9) and (10)). The visual quality of XHTML format is thus higher because the visual quality of the adapted slide components is computed separately, and that of text components is always equal to 1. In the case of JPEG, however, the whole slide is converted into one JPEG image, and so its visual quality is computed as the image quality of that JPEG image. Regarding the delivery time, the XHTML-based adapted slide is lighter, as compared to the JPEG-based one, for the same reasons; i.e., for



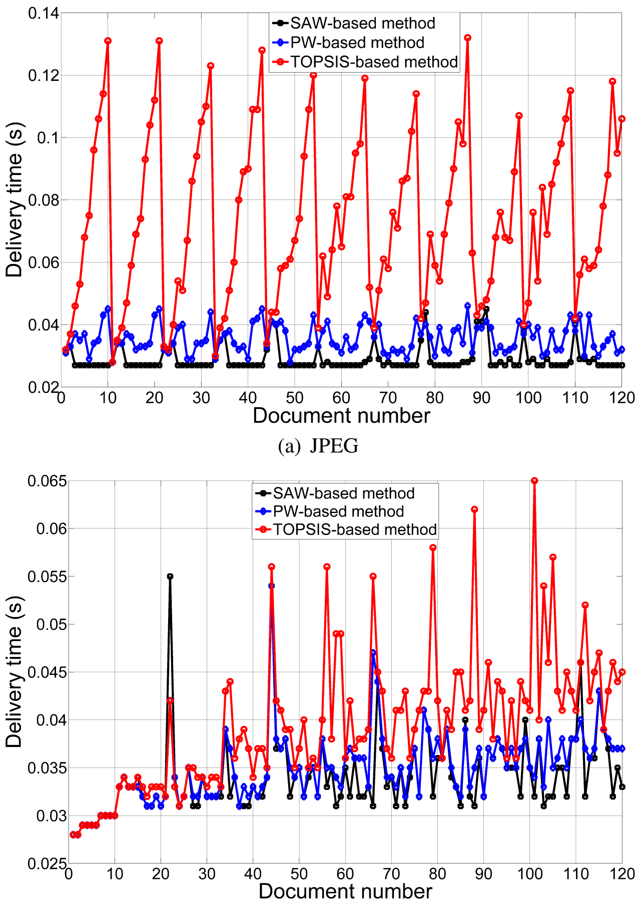
(a) JPEG

(b) XHTML

Fig. 5: Optimal $\mathcal{Q}_V$ obtained with $w_V(D) = 0.5, w_T(D) = 0.5$.

JPEG, we have one JPEG image, and for XHTML, we have images and text, and the areas with no text and no images are not counted.

Statistically speaking, in Tables 1 and 2, we present the average optimal $\mathcal{Q}_V$ and $\mathcal{T}_D$ obtained by SAW-, WP- and TOPSIS-based methods for JPEG and XHTML, respectively, as well as their variances. The variance presented in these tables confirms the variability of $\mathcal{Q}_V$ for the SAW- and WP-based methods, compared to that of the TOPSIS-based one. It also confirms the variability of $\mathcal{T}_D$ for TOPSIS compared to SAW and WP.



(a) JPEG

(b) XHTML

Fig. 6: Optimal $\mathcal{T}_D$ obtained $w_V(D) = 0.5, w_T(D) = 0.5$.

Table 1: JPEG average values and their variances.

	Average		Variance	
Methods	$\mathcal{Q}_V$	$\mathcal{T}_D$ $(sec.)$	$\mathcal{Q}_V$	$\mathcal{T}_D\,(\times 10^{-3})$
SAW-based method	0.397	0.029	0.039	0.002
WP-based method	0.501	0.036	0.026	0.002
TOPSIS-based method	0.759	0.074	0.003	0.072

6.4 Human perception analysis

Studies performed to characterize the maximum tolerable waiting time in Web browsing suggest that 2 seconds is the limit [28]. This time can be increased to 38 seconds

Table 2: XHTML average values and their variances.

	Average		Variance	
Methods	$\mathcal{Q}_V$	$\mathcal{T}_D$ $(sec.)$ s	$\mathcal{Q}_V$	$\mathcal{T}_D\,(\times 10^{-3})$
SAW-based method	0.766	0.034	0.022	0.002
WP-based method	0.786	0.035	0.015	0.001
TOPSIS-based method	0.846	0.040	0.005	0.005

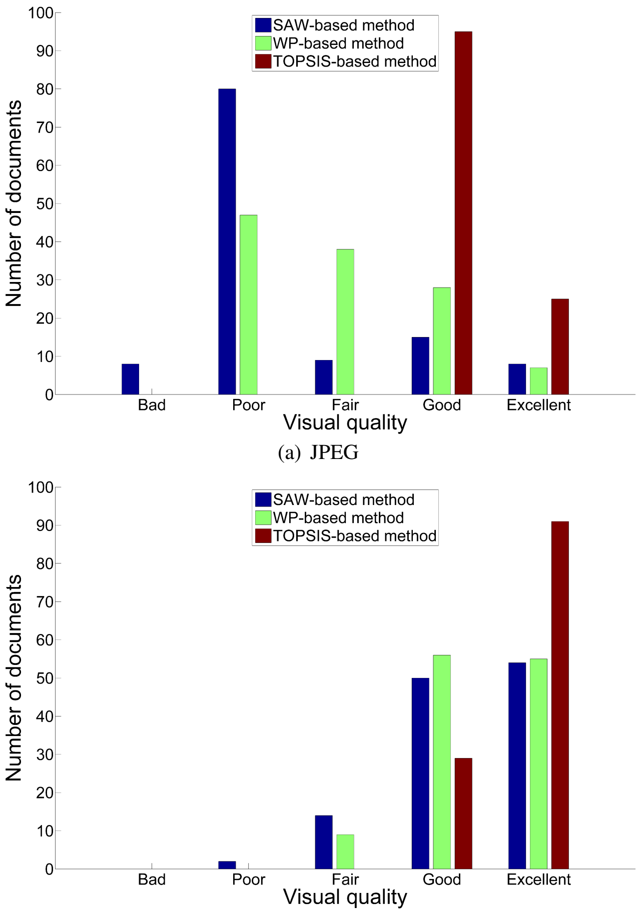
by adding a "status bar" to the Web application in order to keep the client interested. In our context, the maximum delivery time is less than 0.14 seconds (see Fig. 6(a)). Since the maximum delivery time we have is very small, it is important to analyze the visual quality of the adapted documents obtained with the three methods under consideration. To that end, we show the meaning of the obtained visual qualities using the mean opinion score, which reflects the human perception. The obtained visual qualities are grouped into five categories, as suggested by MOS, and presented in Figs 7(a) and 7(b) for JPEG and XHTML respectively. In this view, we observe that all the optimal documents obtained by the TOPSIS-based method have *Good* or *Excellent* visual quality (in the sense of mean opinion score). Those obtained by the SAW- and WP-based methods range mostly between *Bad* and *Fair* for JPEG, and between *Poor* and *Good* for XHTML. It seems that the SAW- and WP-based methods are constrained by a low $\mathcal{T}_D$ threshold, which pushes the $\mathcal{Q}_V$ lower. However, the results are obtained with neutral user preferences, $w_V(D) = 0.5, w_T(D) = 0.5$, and a fast communication network (LTE). Therefore, we expect to obtain higher $\mathcal{Q}_V$ values, as the network can tolerate delivering documents with higher data sizes (higher $\mathcal{T}_D$), and thus higher $\mathcal{Q}_V$. This shows clearly that the weights in SAW- and WP-based methods do not properly match the user's expectations since they are applied to different metrics, $\mathcal{Q}_V$ and $\mathcal{T}_D$, without much consideration regarding their respective scales.

6.5 Impact of user preferences

In this section, we show the impact of varying the user preferences on the behavior of the three methods under consideration. The objective is to see if the proposed TOPSIS-based QoE model is also reliable when different user preference combinations are used. To this end, we evaluate the optimal solutions of each method using various user preference combinations, as follows:

$$w_V(D) \in \{0.1, 0.2, \ldots, 0.9\}, \;\; w_T(D) = 1 - w_V(D) \tag{26}$$

Similarly, we evaluate the weighted absolute deviation of the obtained solutions from utopia. The averaged weighted absolute deviations from utopia are plotted in Figs. 8(a) and 8(b) for JPEG and XHTML, respectively. We observe that the distance to utopia obtained with our proposed TOPSIS-based method is smaller compared to that obtained with the SAW- and WP-based methods. We also note that the SAW- and WP-based methods mostly provide the same results.



(a) JPEG

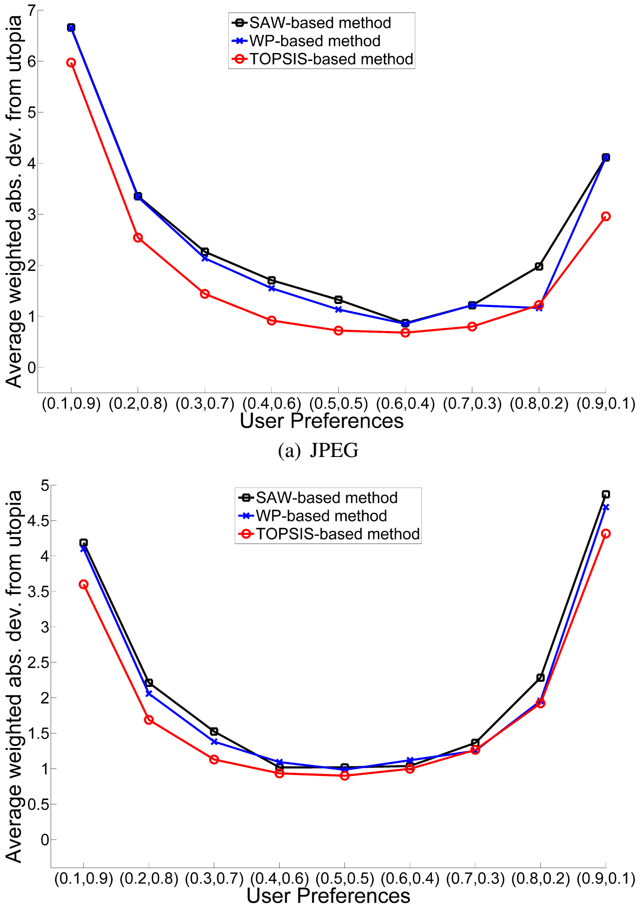
(b) XHTML

Fig. 7: Optimal $\mathcal{Q}_V$ converted into human perception based on MOS. Results obtained with $w_V(D) = 0.5, w_T(D) = 0.5$.

6.6 Impact of the communication network conditions

In previous subsections, we analyzed all the performance facets of the proposed QoE model with the conditions of an actual communication network, which is LTE. In this section, we show its behavior when the network conditions change. To this end, we run the simulations with two others mobile devices connected to the following networks [37,36]:

- *Scenario 1* (e.g., HSDPA network): bitrate of $N_B(D) = 1.8$ Mbps and a latency of $N_L(D) = 90$ ms.
- *Scenario 2* (e.g., WLAN network): bitrate of $N_B(D) = 22$ Mbps and a latency of $N_L(D) = 5$ ms.



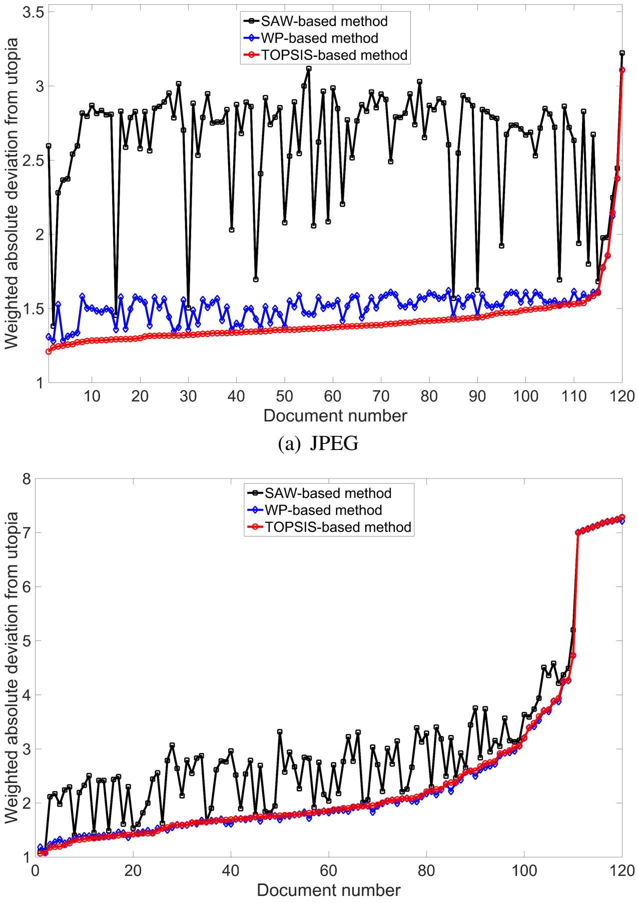
(a) JPEG

(b) XHTML

Fig. 8: Average weighted absolute deviation from utopia, obtained with various user preference combinations $\big(w_{_V}(D), w_{_T}(D)\big)$.

Note that, these values may change drastically in the same communication network. Figures 9 and 10 show the weighted absolute deviation from utopia for scenarios 1 and 2, respectively. We observe that the TOPSIS-based method performs better than the SAW- and WP-based methods. We observe also that the WP- and TOPSIS-based methods performs almost similarly in the first scenario when XHTML is used. This is however not a rule, but rather an exception.

On a final note, we can conclude that the WP-based method performs better than the SAW-based method, which is widely used in content adaptation, and the TOPSIS-based method is far the best among the three tested methods. In the next section, we analyse its complexity and limitations.



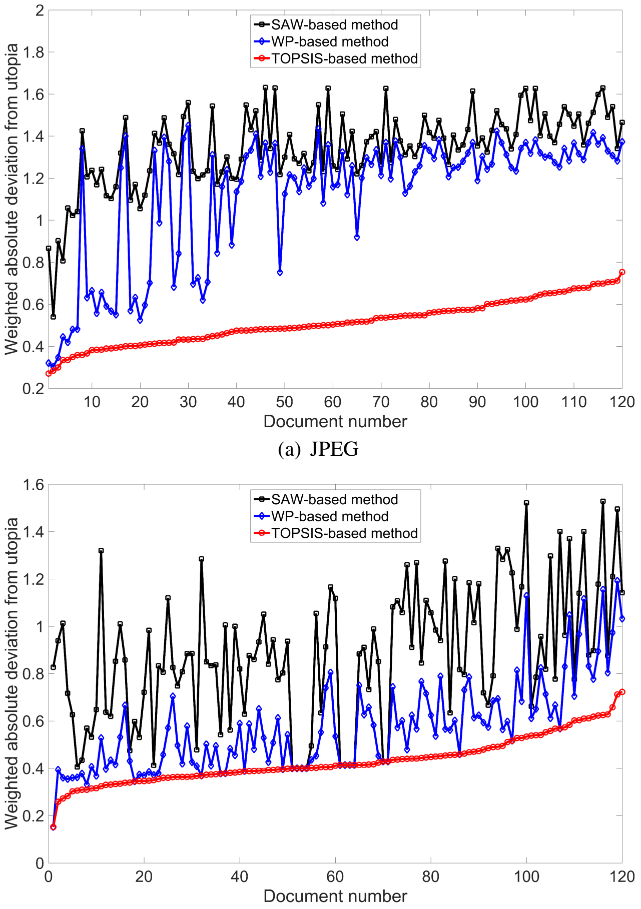
(a) JPEG

(b) XHTML

Fig. 9: Weighted absolute deviation from utopia, obtained with $w_V(D) = 0.5, w_T(D) = 0.5$, and $N_B(D) = 1.8$ Mbps, $N_L(D) = 90$ ms. The documents are sorted according to the results of the TOPSIS-based method.

7 Complexity and Limitations

In this section, we first study the asymptotic complexity of the proposed QoE model, which is based on the TOPSIS scoring method, then analyse its limitations. We show how complex this method is, compared to that used in the literature (SAW), but also to the WP one. Let the problem at hand be of n objectives or attributes (quality functions in our case), and m alternatives (number of adapted versions of a given content). The number of operations required by each method to evaluate the optimal solution are summarized in Table 3. At the asymptotic level, all these methods are of the same complexity $\mathcal{O}(n \times m)$, that is, on high-end computing servers, our proposed QoE model can be used without experiencing any processing latency. Therefore, the



(a) JPEG

(b) XHTML

Fig. 10: Weighted absolute deviation from utopia, obtained with $w_V(D) = 0.5, w_T(D) = 0.5$, and $N_B(D) = 22$ Mbps, $N_L(D) = 5$ ms. The documents are sorted according to the results of the TOPSIS-based method.

proposed model provides reliable results, compared to existing solutions, without adding significant processing overhead.

Regarding the limitation of the proposed QoE model, ideally, the optimal solution should lie on the Pareto-optimal front, which contains all the best trade-offs between the objective functions, $\mathcal{Q}_V$ and $\mathcal{T}_D$ in our case. All the other solutions are not suitable, as they are dominated (in the sense of multi-objective optimization) by at least one solution from the Pareto-optimal front. Unfortunately, the TOPSIS-based method does not guarantee a solution from the Pareto-front [26]. Indeed, even in the problem at hand, we identified an example in which the optimal solution obtained does not lie on the Pareto-optimal front, as shown in Fig. 11. Although, in this example, the

Table 3: Complexity (number of operations for n objectives and m alternatives)

Methods	Operation type	Number of operations	Asymptotic complexity
SAW-based	Min/max of alternatives	$n \times m$	$4(n \times m) + m$ $\simeq \mathcal{O}(n \times m)$
	Normalization	$n \times m$	
	Weighting alternatives	$n \times m$	
	Summing the attributes	$n \times m$	
	Evaluation of the optimal	m	
WP-based	Weighting alternatives	$n \times m$	$2(n \times m) + m$ $\simeq \mathcal{O}(n \times m)$
	Product of attributes	$n \times m$	
	Evaluation of the optimal	m	
TOPSIS-based	Normalization	$n \times m$	$6(n \times m) + 3m$ $\simeq \mathcal{O}(n \times m)$
	Weighting alternatives	$n \times m$	
	Positive and negative ideals	$n \times m$	
	Evaluation of d^* and d^-	$n \times m$	
	Evaluation of C^*	$2m$	
	Evaluation of the optimal	m	

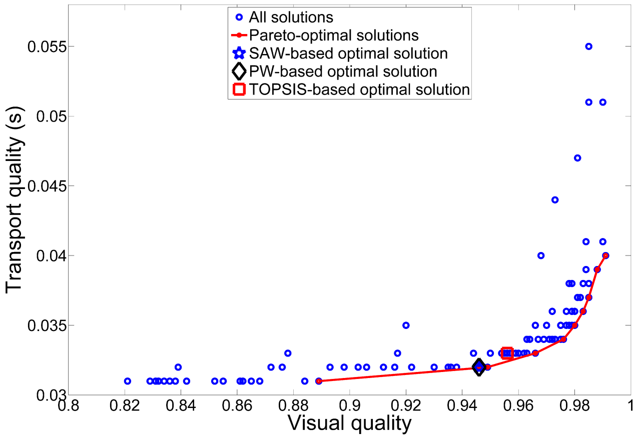


Fig. 11: Optimal solutions obtained with the SAW-, WP- and TOPSIS-based methods of a given slide. The TOPSIS optimal solution is not on the Pareto-optimal front.

TOPSIS optimal solution is better than those of the SAW- and WP-based methods, it does not belong to the Pareto-optimal front. In this case, it is clear that there is a least one solution that is superior to it. An adaptation of the TOPSIS method could resolve this issue, thereby ensuring optimal solutions from the Pareto-optimal front.

8 Conclusion

When mobile devices are involved in conference meetings, shared documents must be adapted to be accepted by the target mobile device and improve the end-user experience. The adaptation process uses the target mobile device's features, network conditions, and the end-user's preferences to evaluate the QoE of each adapted content prior to delivery.

Existing research on QoE evaluation of adapted content weights the context parameters, such as network bitrate, directly. However, not all users are able to understand these parameters, nor their impact on the requested adapted content quality. This renders the proposed QoE models user-unfriendly. Besides, the context parameters are summed up to quantify the quality of each adapted content. This quantifying method is not really accurate as not all the context parameters can compensate for each other.

In this paper, we propose a user-friendly and reliable QoE model that makes all the context parameters transparent to the end-user and exposes high-level quality functions understandable by all users. Furthermore, it uses the TOPSIS scoring method to combine these functions in order to quantify the quality of each adapted content, and thereafter select the optimal one. Experimental results show that the proposed model is convenient, as the end-user does not need to worry about the context parameters, and instead, manipulates quality functions that have real meaning to him. Regarding the combination of the quality functions, the results show that the proposed TOPSIS-based QoE model is very reliable, and provides optimal solutions closer to an ideal solution (utopia) than those presented in the literature. The QoE model we propose can be extended, in the sense that new quality functions, such as battery life, can be included. As the TOPSIS method does not guarantee optimal solutions from the Pareto-optimal front, it is worth investigating, in a future work, how this method can be improved to provide non-dominated optimal solutions.

References

1. Radhwan Mohamed Abdullah and Zuriati Ahmad Zukarnain. Enhanced handover decision algorithm in heterogeneous wireless network. *Sensors*, 17(1626), 2017.
2. Adobe Systems Incorporated. Adobe Connect 8: Web Conferencing for the Enterprise. http://www.adobe.com/products/adobeconnect.html, 1015. Accessed on 17 December 2017.
3. Association of professional wireless production technology. Realistic LTE Performance - From Peak Rate to Subscriber Experience. http://www.apwpt.org/downloads/realistic_lte_experience_wp_motorola_aug2009.pdf, 2009. Accessed on 17 december 2017.
4. Alexandro Bordignon, Fernando Varella, Julio Toss, Maria Lucia Kroeff Barbosa, and Valter Roesler. Mechanisms for interoperable content production among Web, Digital TV and Mobiles. *Informática Na Educação: teoria & prática*, 12(1):110–118, 2009.
5. S. Coulombe and S. Pigeon. Quality-aware Selection of Quality Factor and Scaling Parameters in JPEG Image Transcoding. In *2009 IEEE Symposium on Computational Intelligence for Multimedia Signal and Vision Processing*, pages 68–74, Mar 2009.
6. S. Coulombe and S. Pigeon. Low-complexity Transcoding of JPEG Images With Near-optimal Quality Using a Predictive Quality Factor and Scaling Parameters. *IEEE Transactions on Image Processing*, 19(3):712–721, Mar 2010.
7. Stephane Coulombe and Guido Grassel. Multimedia adaptation for the multimedia messaging service. *IEEE Communications Magazine*, 42(7):120–126, July 2004.
8. A. Dieckmann, K. Dippold, and H. Dietrich. Compensatory versus Noncompensatory Models for Predicting Consumer Preferences. *Judgment and Decision Making*, 4(3):200–213, 2009.
9. Google Inc. Convert a synced file to Google Docs format. https://www.diigo.com/list/angieema/google-drive, 2015. Accessed on 17 December 2017.
10. Google Inc. GoogleDocs. http://www.google.com/google-d-s/documents/, 2015. Accessed on 17 December 2017.

11. Google Inc. Minimum requirements for Google Docs on a mobile browser. https://docs.google.com/document/d/1pi18-IBREwmBGcDBoNZKHd9uqyPX2NRGflYFTugQOws/edit, 2015. Accessed on 17 December 2017.
12. Kento Goto, Ryosuke Koshijima, and Motomichi Toyama. Generating Desktop and Mobile Web Pages from a Single SuperSQL Query. In *Proceedings of the 19th International Database Engineering & Applications Symposium*, IDEAS '15, pages 222–223, NY, USA, 2014. ACM.
13. R. Han, P. Bhagwat, R. LaMaire, T. Mummert, V. Perret, and J. Rubas. Dynamic Adaptation in an Image Transcoding Proxy for Mobile Web Browsing. *IEEE Personal Communications*, 5(6):8–17, 1998.
14. Jung-Lee Hsiao, Hao-Ping Hung, and Ming-Syan Chen. Versatile Transcoding Proxy for Internet Content Adaptation. *IEEE Transactions on Multimedia*, 10(4):646–658, June 2008.
15. C. L. Hwang and K. Yoon. *Multiple attribute decision making : Methods and applications*. Springer-Verlag, 1981.
16. He Jiang, Gao Tong, Hao Wei, Yen I-Ling, and Farokh Bastani. A Flexible Content Adaptation System Using a Rule-Based Approach. *IEEE Transactions on Knowledge and Data Engineering*, 19(1):127–140, Jan 2007.
17. Ignacy Kaliszewski and Dimitry Podkopaev. Simple Additive Weighting A Metamodel for Multiple Criteria Decision Analysis Methods. *Expert Systems with Applications*, 54:155 – 161, 2016.
18. Aaron Koehl and Haining Wang. m.Site: Efficient Content Adaptation for Mobile Devices. In *Middleware 2012*, volume 7662 of *Lecture Notes in Computer Science*, pages 41–60. Springer Berlin Heidelberg, 2012.
19. F. Kuipers, R. Kooij, D. De Vleeschauwer, and K. Brunnström. Techniques for Measuring Quality of Experience. In *Wired/Wireless Internet Communications*, volume 6074 of *Lecture Notes in Computer Science*, pages 216–227. Springer-Verlag Berlin/Heidelberg, 2010.
20. L. Lee and R. Anderson. A Comparison of Compensatory and Non-Compensatory Decision Making Strategies in IT Project Portfolio Management. http://aisel.aisnet.org/irwitpm2009/9, 2009. Accessed on 17 decemberr 2017.
21. H. Louafi, S. Coulombe, and U. Chandra. Quality Prediction-Based Dynamic Content Adaptation Framework Applied to Collaborative Mobile Presentations. *IEEE Transactions on Mobile Computing*, 12(10):2024–2036, Oct 2013.
22. Habib Louafi, Stéphane Coulombe, and Umesh Chandra. Robust QoE-aware prediction-based dynamic content adaptation framework applied to slides documents in mobile Web conferencing. *Springer Multimedia Tools and Applications*, 74(18):7883–7920, Sep 2015.
23. W. Y. Lum and F.C.M. Lau. On Balancing Between Transcoding Overhead and Spatial Consumption in Content Adaptation. *Proceedings of the 8th annual international conference on Mobile computing and networking MobiCom 02*, page 239, 2002.
24. W. Y. Lum and F.C.M. Lau. User-Centric Content Negotiation for Effective Adaptation Service in Mobile Computing. *IEEE Transactions on Software Engineering*, 29(12):1100–1111, Dec 2003.
25. W. Y. Lum and F.C.M. Lau. User-Centric Adaptation of Structured Web Documents for Small Devices. In *19th Int. Conf. on Advanced Information Networking and Applications (AINA'05)*, volume 1, pages 507–512. IEEE, 2005.
26. R.T. Marler and J.S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6):369–395, 2004.
27. Karan Mitra, Arkady Zaslavsky, and Christer Ahlund. Context-Aware QoE Modelling, Measurement, and Prediction in Mobile Computing Systems. *IEEE Transactions on Mobile Computing*, 14(5):920–936, May 2015.
28. F. F. Nah. A study on Tolerable Waiting Time: How Long Are Web Users Willing to Wait? *Behaviour & Information Technology*, 23(3):153–163, Jan 2004.
29. Open Mobile Alliance. User Agent Profile - Approved Version 2.0. http://www.w3.org/Protocols/rfc2616/rfc2616.html, 2006. Accessed on 17 December 2017.
30. Kosmas Pipyros, Christos Thraskias, Lilian Mitrou, Dimitris Gritzalis, and Theodoros Apostolopoulos. A new strategy for improving cyber-attacks evaluation in the context of tallinn manual. *Elsevier Computers & Security*, 2017.
31. A. Richards, M. Antoniades, V. Witana, and G. Rogers. Mapping User Level QoS from a Single Parameter. In *Proceedings of the International Conference on Multimedia Networks and Services*, Nov 1998.
32. K. Savitha and C. Chandrasekar. Vertical handover decision schemes using saw and wpm for network selection in heterogeneous wireless networks. *Global Journal of Computer Science and Technology*, 11(9):19–24, May 2011.

33. Mohamed Abdelkrim Senouci, Said Hoceini, and Abdehamid Mellouk. Utility function-based TOPSIS for network interface selection in Heterogeneous Wireless Networks. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2016.
34. H.R. Sheikh, Z. Wang, L. Cormack, and A.C. Bovik. LIVE Image Quality Assessment Database Release 2. http://live.ece.utexas.edu/research/quality/subjective.htm, 2006. Accessed on 17 December 2017.
35. Jun-Ming Su, Shian-Shyong Tseng, Huan-Yu Lin, and Chun-Han Chen. A personalized learning content adaptation mechanism to meet diverse user needs in mobile learning environments. *User Modeling and User-Adapted Interaction*, 21(1-2):5–49, 2011.
36. Kaixin Sui, Mengyu Zhou, Dapeng Liu, Minghua Ma, Dan Pei, Youjian Zhao, Zimu Li, and Thomas Moscibroda. Characterizing and Improving Wifi Latency in Large-scale Operational Networks. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services. ACM*, 2016.
37. P. Svoboda, F. Ricciato, W. Keim, and M. Rupp. Measured WEB Performance in GPRS, EDGE, UMTS and HSDPA with and without Caching. In *IEEE Int. Symp. on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–6, Jun 2007.
38. W3C. Hypertext Transfer Protocol. http://www.w3.org/Protocols/rfc2616/rfc2616.html, 1999. Accessed on 17 December 2017.
39. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
40. K.Paul Yoon and Ching-Lai Hwang. *Multiple Attribute Decision Making: An Introduction*. SAGE Publications, 1995.
41. Y. Zhang, S. Zhang, and S. Han. A New Methodology of QoS Evaluation and Service Selection for Ubiquitous Computing. In *Wireless Algorithms, Systems, and Applications*, volume 4138 of *LNCS*, pages 69–80. Springer Berlin / Heidelberg, 2006.
42. Y. Zhang, S. Zhang, and S. Han. Context-Aware Service Selection Engine for Ubiquitous Computing Application. In *6th World Congress on Intelligent Control and Automation*, pages 4269–4273. IEEE, 2006.
43. Y. Zhang, S. Zhang, and S. Han. Context-Based Qos Model and its Application in Ubiquitous Computing. In *International Conference on Machine Learning and Cybernetics*, pages 1517–1521. IEEE, Aug 2006.
44. Y. Zhang, S. Zhang, and H. Tong. Adaptive Service Delivery for Mobile Users in Ubiquitous Computing Environments. In Jianhua Ma, Hai Jin, Laurence Yang, and Jeffrey Tsai, editors, *Ubiquitous Intelligence and Computing*, volume 4159 of *Lecture Notes in Computer Science*, pages 209–218. Springer Berlin / Heidelberg, 2006.
45. Zoho. Zoho Mobile. http://www.zoho.com/mobile/, 2010. Accessed on 17 december 2017.
46. Zoho Corp. Zoho Show. https://www.zoho.com/docs/show.html, 2015. Accessed on 17 december 2017.

Habib Louafi holds an Engineering degree in Computer Science from the University of Oran (Algeria), an M.Sc. from the Université du Québec à Montréal (UQÀM), and a Ph.D. from École de technologie supérieure (ÉTS is a constituent of the Université du Québec network). He is currently working as a postdoctoral fellow at Synchromedia Laboratory for multimedia communication in telepresence (ÉTS). His fields of interest include mobile computing, cloud and mobile cloud computing, distributed systems, context-aware systems, QoE, content adaptation, collaborative mobile Web applications, and information security.

Stéphane Coulombe received a B.Eng. in Electrical Engineering from École Polytechnique de Montréal, Canada, in 1991, and a Ph.D. in Telecommunications (image processing) from INRS-Telecommunications, Montreal, in 1996. He is currently a Professor in the Software and IT Engineering Department at École de technologie supérieure (ÉTS is a constituent of the Université du Québec network). From 1997 to 1999, he was with Nortel Wireless Network Group in Montreal, and from 1999 to 2004, he was with the Nokia Research Center, Dallas, TX, as a Senior Engineer and as a Program Manager in the Audiovisual Systems Laboratory. He joined ÉTS in 2004, where he currently carries out research and development on video processing and systems, media adaptation, and transcoding. Since 2009, he has held the Vantrix Industrial Research Chair in Video Optimization.

Mohamed Cheriet is a full professor in the Automation Engineering Department at École de technologie supérieure, a constituent of the Université du Québec network. His expertise includes document image analysis, optical character recognition, mathematical models for image processing, pattern classification models, and learning algorithms, as well as perception in computer vision. Cheriet received a PhD in Computer Science from the University of Pierre et Marie Curie. He co-founded the Laboratory for Imagery, Vision, and Artificial Intelligence (LIVIA) and founded and directs the Synchromedia Consortium (Multimedia Communication in Telepresence) at the Université du Québec.