Efficient Near-Optimal Dynamic Content Adaptation Applied to JPEG Slides Presentations in Mobile Web Conferencing

Habib Louafi, Stéphane Coulombe
Department of Software and IT Engineering
École de Technologie Supérieure, Université du Québec
1100 Notre-Dame Street West, Montreal,
Quebec, Canada H3C 1K3
Emails: habib.louafi.1@ens.etsmtl.ca, stephane.coulombe@etsmtl.ca

Umesh Chandra
Nokia Research Center
955 Page Mill Road, Palo Alto, CA 94304 USA.
Email: umesh.1.chandra@nokia.com

***Abstract*—In the context of mobile Web conferencing, slide documents are generally transcoded into JPEG format and wrapped into a Web page prior to delivery. Given the diversity of these devices and their networks, dynamically identifying the optimal transcoding parameters is very challenging, as the number of transcoding parameters combinations could be very high. Current solutions use the resolution of the target mobile device and a fixed quality factor as transcoding parameters. However, this technique allows no control over the resulting file size, which, if too large, might increase the delivery time and negatively affect users' experience. Another solution (content selection) which leads to better quality consists in creating several versions and, at delivery time, selecting the best one. However, such a solution is computationally expensive. In this paper, we propose a prediction-based framework which computes near-optimal transcoding parameters dynamically with far less computations. We propose five methods based on this framework. The first predicts near-optimal transcoding parameters, while the others improve their accuracy. From the set of documents tested, two of the proposed methods reach optimality 14% and 30% of the time, respectively. Moreover, the average deviation from optimality for the proposed methods varies from 6% to 3%, with a complexity varying from 1 to 5 transcoding operations.**

***Keywords*-mobile Web conferencing; dynamic content adaptation; transcoding; quality of experience; JPEG.**

I. Introduction

Professional documents, such as PowerPoint slides and Word documents are widely used and shared between peers in many Web conferencing and collaborative applications (e.g., GoogleDocs and Zoho). This has been largely facilitated by the use of the Web as a content delivery platform. Furthermore, the Web has allowed the holding of meeting-conferences with slide decks shared and presented synchronously to all participants, connected via their Web browsers. In this regard, many solutions have been proposed for PCs and laptops [1], [2]. However, when mobile phones are considered in such meeting contexts, the content (slides) must be adapted to meet the constraints (supported formats, maximum resolution and file size) and environment (data rate) of the target mobile phones [3]–[5]. As JPEG is widely used and supported by mobile devices, professional documents are generally transcoded into JPEG-based Web pages, in which each page (or slide) is converted into a JPEG image and wrapped into an XHTML skeleton page.

Professional document adaptation used in current products is not device-independent; that is, documents are optimized only for a few specific mobile devices [6], [7]. To reach a wide variety of mobile devices, the straightforward solution therefore consists in using the target mobile device's resolution in conjunction with a fixed quality factor (e.g., 80); which we will call FQF, for *fixed quality factor*. Although this technique provides good visual quality, it does not consider the resulting file size. The fact though is that the adapted content's file size affects its delivery time, and might thus increase the end-user's waiting time. This can be especially problematic when a high-resolution image is delivered over a low bitrate network. In a meeting context, the user may find himself waiting for slides while the presenter is talking about them, which creates a serious usability problem. JPEG parameters must therefore be optimized for each mobile user, using a certain quality of experience criterion considering visual quality and delivery time. A good discussion, supported by real examples, of the selection of the resolution and quality factor parameters of transcoded JPEG images leading to near-optimal visual quality under a file size constraint, can be found in [8].

An exhaustive adaptation solution, also called content selection, to this problem would consist in creating, often offline, several versions of the content using various combinations of quality factor and scaling parameters [3]–[5]. At delivery time, the best version providing the optimal user's experience, evaluated using a good quality of experience metric, is selected to be delivered. Though this solution provides the best experience, it also leads to high processing complexity in generating all the versions, which further, require a great deal of storage space. Its complexity and performance depend on the granularity in use, which would be 100 versions if 10 quality factor and 10 scaling parameter values were selected.

In this paper, we propose a prediction-based framework which has the advantages of the two previously presented approaches (exhaustive and FQF), but not their drawbacks. This framework, which is based on quality and file size predictors of JPEG images subject to changing their resolution and quality factor [8]–[10], enables us to identify near-optimal (compared to the exhaustive method) transcoding parameters at delivery time without performing exhaustive transcodings. We present several methods based on this framework. The first one, which is presented in [11], predicts near-optimal transcoding parameters, while the others, are variants of the first one, but with improved accuracy. Each method offers an interesting compromise between performance (how close it is to the optimal quality of experience) and complexity (number of transcoding operations required).

We show that the quality of the content adapted using this framework is near-optimal. From the set of documents tested, two of the proposed methods reach optimality 14% and 30% of the time, respectively. Further, the average deviation from optimality for the proposed methods varies from 6% to 3%, with a complexity varying from 1 to 5 transcoding operations, which makes them very appealing.

This paper is structured as follows. In section II, we present the problem statement. In section III, we propose metrics for evaluating the quality of experience. Section IV details the proposed prediction-based framework as well as various methods based on that framework. In section V, we present the experimental setup used to validate the proposed methods. The experimental results are presented and discussed in section VI. Finally, section VII concludes the paper.

II. Problem Statement

Let $\mathcal{C}$ be a professional document (e.g., PowerPoint or Word document), which is normally composed of a set of pages (or slides) c_k. Formally, $\mathcal{C}$ can be written as follows: $\mathcal{C} = \{c_k\}_{k=1}^{n}$, where n is the total number of pages that compose $\mathcal{C}$. For a given page c_k, referred to here as the original content, let $\mathcal{W}(c_k)$ and $\mathcal{H}(c_k)$ be its width and height, in pixels, respectively.

To be rendered by the target mobile device, the original content must be adapted. To achieve this, different transcoding parameters values can be used for each page.

In the following, we will consider the adaptation of slides decks into JPEG-based Web pages. The embedded JPEG image's quality and file size are affected mainly by two parameters: resolution and quality factor. Let $\mathcal{P} = \{(z, QF)\}$ be the set of transcoding parameters couples that can be used to adapt the original content, where $0 < z \leq 1$ and $0 < QF \leq 100$ are the scaling parameter (*zoom*) used to adjust the resolution of the embedded JPEG image and its target quality factor, respectively. In other words, the whole slide is adapted into a Web page that contains only one JPEG image; which is transcoded using the two parameters under consideration, z and QF.

We define $\mathcal{T}$ as the transcoding operation that adapts the original page c_k (slide) into a JPEG-based Web page using the transcoding parameters z and QF, as follows:

$$\mathcal{T} : \mathcal{C} \times \mathcal{P} \rightarrow \mathcal{C}^{z,QF}$$
$$c_k \times (z, QF) \mapsto c_k^{z,QF}$$

where $\mathcal{C}^{z,QF}$ is the set that contains all the possible adapted content versions that can be created by $\mathcal{T}$ from $\mathcal{C}$, using all parameters from $\mathcal{P}$. $c_k^{z,QF}$ represents the adapted content version of c_k created using z and QF.

Let D be the target mobile device and $\mathcal{W}(D)$, $\mathcal{H}(D)$ and $\mathcal{S}(D)$ be its maximum permissible image width, image height and file size (in bits), respectively.

From the set of adapted content versions that can be created from c_k using $\mathcal{T}$, only a subset can be rendered by D. Let $\mathcal{R}_{c_k}^{D}$ be the set of transcoding parameters couples that can be used to create these renderable versions:

$$\mathcal{R}_{c_k}^{D} = \Big\{(z, QF) \mid \mathcal{S}(c_k^{z,QF}) \leq \mathcal{S}(D) \text{ and } z\mathcal{W}(c_k) \leq \mathcal{W}(D) \text{ and } z\mathcal{H}(c_k) \leq \mathcal{H}(D)\Big\}$$

where $\mathcal{S}(c_k^{z,QF})$ is the file size of the adapted content $c_k^{z,QF}$. Since there could be multiple transcoding parameters' couples leading to versions renderable by D, the objective is to compute the ones that maximize the user's quality of experience, which we denote here by $\mathcal{Q}_E(c_k^{z,QF}, D)$, and which will be defined in the next section. The optimal parameter values are given by:

$$\big(z^*(c_k, D), QF^*(c_k, D)\big) = \underset{(z,QF) \in \mathcal{R}_{c_k}^{D}}{\arg\max}\ \mathcal{Q}_E(c_k^{z,QF}, D) \quad (1)$$

Note that there may be several solutions to (1). In this case, the parameters leading to the best visual quality are arbitrarily selected.

III. Quality of Experience Evaluation

The quality of the delivered adapted content, as experienced by the end-user, is determined by three elements [12]:

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 regarding the adapted content.

The first and third elements express how the content is appreciated visually (visual quality), while the second expresses the impact of the total delivery time on the appreciation of the content (transport quality). That is, the quality of experience of the adapted content ($\mathcal{Q}_E$) can be expressed by two factors: its visual quality ($\mathcal{Q}_V$) and transport quality ($\mathcal{Q}_T$). Based on these two factors, we propose an evaluation of the quality of experience $\mathcal{Q}_E$ as follows:

$$\mathcal{Q}_E(c_k^{z,QF}, D) = \mathcal{Q}_V(c_k^{z,QF})\mathcal{Q}_T(c_k^{z,QF}, D) \quad (2)$$

where $\mathcal{Q}_V(c_k^{z,QF})$ and $\mathcal{Q}_T(c_k^{z,QF}, D)$ are limited to the interval [0,1], and represent the visual quality and the transport quality, respectively. As discussed in [11], we propose the product of $\mathcal{Q}_V$ and $\mathcal{Q}_T$, rather than the sum, to prevent large disparities in $\mathcal{Q}_V$ and $\mathcal{Q}_T$ from being able to produce a high $\mathcal{Q}_E$. Indeed, the product is more suitable in this context than the sum, since $\mathcal{Q}_V$ and $\mathcal{Q}_T$ are not compensatory. For instance, sending a very aggressively compressed JPEG image will result in $\mathcal{Q}_T \approx 1$ (very lightweight image) and $\mathcal{Q}_V \approx 0$ (very distorted image). In this case, the sum will be close to 1 (which is misleading), while the product will be close to 0, which is more reasonable. In fact, before combining two or more attributes to get a single measure that reflects the nature of the problem in context, these attributes should first be classified into compensatory and non-compensatory attributes. This helps identify the attributes that can be summed (compensatory ones) from the others. This aspect is widely studied in the marketing and decision making fields [13], [14].

Although further research and validation are required to establish a metric that accurately matches the user's experience, the proposed metric is adopted here to illustrate the benefits of performing prediction-based dynamic content adaptation over existing methods. Similar benefits are expected with other metrics which consider a compromise between visual quality and delivery time.

A. Visual Quality Evaluation

Since the adapted content is a Web page that contains a single JPEG image, the adapted content's visual quality corresponds to the visual quality of that JPEG image. We have:

$$\mathcal{Q}_V(c_k^{z,QF}) = \mathcal{Q}_V(\mathcal{I}_k^{z,QF}) \tag{3}$$

where $\mathcal{I}_k^{z,QF}$ is the transcoded JPEG image that is embedded in the adapted content (Web page) $c_k^{z,QF}$. It is created from c_k using the transcoding parameters z and QF. The visual quality of the embedded image can be evaluated using a full-reference objective metric such as PSNR or SSIM [15].

B. Transport Quality Evaluation

The second factor that affects the user's quality of experience is the transport quality. The latter is affected by the total delivery time, which is comprised of the time required to perform the adaptation operation plus the time taken by the adapted content to reach the target mobile device. The total delivery time, T_d, can be defined as:

$$T_d\big(c_k^{z,QF}, D\big) = \frac{\mathcal{S}\big(c_k^{z,QF}\big)}{N_B(D)} + N_L(D) + S_L(D) + T_L(c_k^{z,QF}) \tag{4}$$

where:

- $\mathcal{S}\big(c_k^{z,QF}\big)$ is the file size in bits of $c_k^{z,QF}$.
- D is the target mobile device and $N_B(D)$ and $N_L(D)$ are the bit-rate and the latency of the network to which it is connected, respectively.
- $S_L(D)$ is the server latency. For a device D, it represents the time spent by the request in the server (e.g., in the queue) waiting to be processed. It is affected by the performance of the server and the number of users' requests waiting to be processed.
- $T_L(c_k^{z,QF})$ is the transcoding latency. It represents how long the adaptation operation takes to complete. It depends on the original content c_k and the transcoding parameters z and QF in use. It can be estimated based on past transcoding operations. On high-end computers, this value should be small.

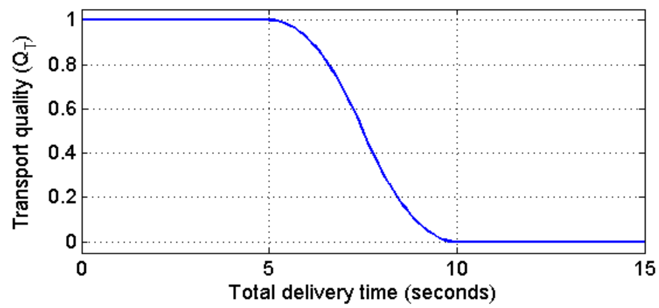

Figure 1. Transport quality behaviour of a given user ($\alpha = 5s$ and $\beta = 10s$)

The transport quality is inversely proportional to the total delivery time. We propose its evaluation using a *Z-shaped built-in membership function* (Zmf) [16]. This function expresses the behaviour of the end-user's appreciation of (or frustration with) the adapted content as a function of its waiting time. An example of such a behaviour is depicted in Fig. 1. In fact, the appreciation or frustration varies from one individual to another, which is why the values of α and β (see Fig. 1) are used. The value of α expresses the period of time during which the end-user is fully satisfied with the response time. The appreciation is reduced to 50% when the response time is at $(\alpha+\beta)/2$ and when it reaches the value of β, the appreciation falls to zero. Formally, we have:

$$\begin{aligned}\mathcal{Q}_T(c_k^{z,QF}, D) &= \mathrm{Zmf}(x, [\alpha, \beta]) \\ &= \begin{cases} 1, x \le \alpha \\ 1 - 2\big(\frac{x-\alpha}{\beta-\alpha}\big)^2, \alpha \le x \le \frac{\alpha+\beta}{2} \\ 2\big(\frac{x-\beta}{\beta-\alpha}\big)^2, \frac{\alpha+\beta}{2} \le x \le \beta \\ 0, x \ge \beta \end{cases}\end{aligned} \tag{5}$$

where $x = T_d\big(c_k^{z,QF}, D\big)$).

IV. Proposed Methods and Models

In this section, to solve (1), we present a prediction-based framework that computes near-optimal transcoding parameters compared to those that can be obtained by an exhaustive method. This framework can be used by various methods, with each having its specific performance and complexity. The first method ,which is detailed in [11], is the most basic and dynamically estimates near-optimal transcoding parameters using predictors of transcoded JPEG image quality and file size [8]–[10]. These estimated parameters present some imprecisions, but represent a good starting point on which the other proposed methods improve. These methods are in fact variants of the first one, that improve the accuracy but at the price of increased complexity.

A. Method 1 - Estimation

In this method [11], we estimate near-optimal transcoding parameters dynamically, without performing any transcoding operation in advance. Using the predicted quality of transcoded JPEG images subject to change of resolution and quality factor [8], we can estimate the visual quality of the adapted content's embedded image and consequently that of the adapted content (see (3)). Also, for a given mobile device D (when $N_B(D)$, $N_L(D)$ are known), we can use the predicted relative file size of transcoded JPEG images computed in [9], [10] to estimate the total delivery time (see (4)) and the transport quality (see (5)). Thus, we can estimate, for any set of parameters, the quality of experience of each adapted content $\mathcal{Q}_E(c_k^{z,QF}, D)$; so, by solving (1), we can determine the near-optimal transcoding parameters. Using the predictors tabulated in [8]–[10], we therefore can estimate the resulting $\mathcal{Q}_E(c_k^{z,QF}, D)$ associated with every set of transcoding parameters and select the best parameters, denoted $z_1^*(c_k, D)$ and $QF_1^*(c_k, D)$ (optimal parameters using method 1).

Note that in this paper as well as in [8]–[10], quantized values of z and QF were used instead of continuous ones in order to limit the parameters space. That is, using a granularity of $\Delta z = 0.1$ and $\Delta QF = 10$, the quantized values of z and QF used were $\widetilde{z} \in \{0.1, 0.2, 0.3, \ldots, 1\}$ and $\widetilde{QF} \in \{10, 20, 30, \ldots, 100\}$, respectively. The solution space thus consists of 100 distinct combinations of parameters. With such a solution space, an exhaustive method would perform 100 transcodings and select the best parameters $z^*(c_k, D)$ and $QF^*(c_k, D)$. With the proposed method 1, instead, we estimate the optimal solution and perform a single transcoding operation. Since the estimates are not always accurate, other methods which improve on this one can be used, and are presented in the following sub-sections.

B. Method 2 - Estimation and Interpolation

In this method, we use the estimated optimal transcoding parameters $z_1^*(c_k, D)$ and $QF_1^*(c_k, D)$ and its estimated four nearest neighbors. We let the solution space be continuous, suppose that the optimal transcoding parameters are within the region covered by these five points, and model the quality of experience in this region by a bivariate quadratic function defined as:

$$f(x, y) = ax^2 + bx + cy^2 + dy + e \tag{6}$$

where x and y represent z and QF in a continuous space, respectively. The optimal point is where the gradient is null:

$$\frac{\partial f}{\partial x} = 2ax + b = 0, \ \frac{\partial f}{\partial y} = 2cy + d = 0 \tag{7}$$

Using the estimated optimal point and its four estimated nearest neighbors, we compute the coefficients a, b, c, d, and e. Then, using (7), we compute the estimated interpolated optimal transcoding parameters $z_I^*(c_k, D)$ and $QF_I^*(c_k, D)$. We expect the interpolated optimal point to be close to the actual optimal. Two adapted contents are created using the estimated and interpolated transcoding parameters, and the better one is selected as the near-optimal adapted content computed by this method. The optimal transcoding parameters obtained by this second method are thus:

$$\big(z_2^*(c_k, D), QF_2^*(c_k, D)\big) = \operatorname*{arg\,max}_{(z,QF)\in\mathcal{N}_e^2} \mathcal{Q}_E(c_k^{z,QF}, D) \tag{8}$$

$$\text{with } \mathcal{N}_e^2 = \Big\{\big(z_1^*(c_k, D), QF_1^*(c_k, D)\big), \big(z_I^*(c_k, D), QF_I^*(c_k, D)\big)\Big\} \tag{9}$$

In terms of complexity, this method requires 2 transcodings.

C. Method 3 - Estimation and One-Step Diamond Search

In this method, we use the estimated optimal adapted content (from method 1) and its four nearest neighbors. Contrary to method 2, these five points are transcoded versions rather than merely estimated, and the best among them is selected. The optimal transcoding parameters obtained by this third method are thus:

$$\big(z_3^*(c_k, D), QF_3^*(c_k, D)\big) = \operatorname*{arg\,max}_{(z,QF)\in\mathcal{N}_e^5} \mathcal{Q}_E(c_k^{z,QF}, D) \tag{10}$$

where $\mathcal{N}_e^5$ is a set containing 5 elements: the estimated optimal parameters and their four nearest neighbors. Thus:

$$\mathcal{N}_e^5 = \Big\{\big(z_1^*(c_k, D), QF_1^*(c_k, D)\big), \big(z_1^*(c_k, D) \pm \Delta z, QF_1^*(c_k, D)\big), \big(z_1^*(c_k, D), QF_1^*(c_k, D) \pm \Delta QF\big)\Big\} \tag{11}$$

Therefore, this method requires 5 transcoding operations.

D. Method 4 - Estimation and Two-Steps Diamond Search

In this method, we identify and create the estimated optimal point and its four nearest neighbors (as we did in method 3), and the optimal point is identified. If the estimated optimal point computed by method 1 is the same as the one computed by method 3, no further processing is performed and the estimated optimal parameters are $\big(z_3^*(c_k, D), QF_3^*(c_k, D)\big)$. Otherwise, $\big(z_3^*(c_k, D), QF_3^*(c_k, D)\big)$ is used as a starting point and we identify and create its four nearest neighbors, one of which was already created from method 3. The optimal one among them becomes the optimal point computed by this method. The optimal transcoding parameters obtained by this fourth method are thus:

$$\begin{aligned}\big(z_4^*(c_k, D),QF_4^*(c_k, D)\big) = \\ \underset{(z,QF)\in\mathcal{N}_e^{5,8}}{\arg\max}\ \mathcal{Q}_E(c_k^{z,QF}, D)\end{aligned} \tag{12}$$

where $\mathcal{N}_e^{5,8}$ is the set of transcoding parameters used in this method (their number is of 5 or 8). It is given by:

$$\begin{aligned}\mathcal{N}_e^{5,8} = \mathcal{N}_e^{5} \cup \Big\{ &\big(z_3^*(c_k, D) \pm \Delta z, QF_3^*(c_k, D)\big), \\ &\big(z_3^*(c_k, D), QF_3^*(c_k, D) \pm \Delta QF\big)\Big\}\end{aligned} \tag{13}$$

The complexity of this method is either 5 or 8 transcoding operations. We have 5 transcodings ($\mathcal{N}_e^{5} = \mathcal{N}_e^{5,8}$) when $\big(z_3^*(c_k, D), QF_3^*(c_k, D)\big)$=$\big(z_1^*(c_k, D), QF_1^*(c_k, D)\big)$, and 8 transcodings otherwise ($\mathcal{N}_e^{5} \subset \mathcal{N}_e^{5,8}$).

E. Method 5 - Estimation and LRDU greedy search

In this method, we start from the estimated optimal point from method 1 and explore its neighborhood, following a given pattern, until convergence is reached (that is, the $\mathcal{Q}_E$ cannot be improved any further). We tested different patterns and found that for the problem at hand, following the LRDU (left-right-down-up) pattern improved the $\mathcal{Q}_E$ significantly with the least number of transcodings compared to other patterns. Before detailing the LRDU greedy search method, let us review what are the left, right, down, and up nearest neighbors of a given point. For instance, the nearest neighbors of the estimated point of method 1 are given by:

$$\begin{aligned}\text{Left neighbor:}&\big(z_1^*(c_k, D) - \Delta z, QF_1^*(c_k, D)\big)\\ \text{Right neighbor:}&\big(z_1^*(c_k, D) + \Delta z, QF_1^*(c_k, D)\big)\\ \text{Up neighbor:}&\big(z_1^*(c_k, D), QF_1^*(c_k, D) - \Delta QF\big)\\ \text{Down neighbor:}&\big(z_1^*(c_k, D), QF_1^*(c_k, D) + \Delta QF\big)\end{aligned}$$

The pseudo-code of the proposed LRDU greedy algorithm is presented in Algorithm 1. As before, we start from the estimated optimal point $\big(z_1^*(c_k, D), QF_1^*(c_k, D)\big)$. Then, we verify whether the left neighbor provides a better solution. If that is the case, we successively move to the left until there is no further improvement. Otherwise, we verify whether the right neighbor provides a better solution, and if so, we successively move to the right until there is no further improvement. The same process is then performed with the down and up directions. Each time a new point is evaluated, a new transcoding is performed. The optimal transcoding parameters obtained by this fifth method are thus:

$$\begin{aligned}\big(z_5^*(c_k, D),QF_5^*(c_k, D)\big) = \\ \underset{(z,QF)\in\mathcal{N}_e^{LRDU}}{\arg\max}\ \mathcal{Q}_E(c_k^{z,QF}, D)\end{aligned} \tag{14}$$

where $\mathcal{N}_e^{LRDU}$ is the set of points evaluated in this method, which can vary greatly, depending on c_k and D.

```
function LRDU_Search(c_k, D)
begin
    z ← z*_1(c_k, D), QF ← QF*_1(c_k, D)
    Q_E ← Q_E(c_k^{z,QF}, D))
    (z, QF, Q_E) ← LR_Search(c_k, D, z, QF, Q_E)
    (z, QF, Q_E) ← DU_Search(c_k, D, z, QF, Q_E)
    return (z, QF, Q_E)
end

function LR_Search(c_k, D, z, QF, Q_E)
begin
    if Q_E(c_k^{z−Δz,QF}, D)) > Q_E then
        (z, QF, Q_E) ← search(c_k, D, z, QF, Q_E, −1, 0)
    else
        if Q_E(c_k^{z+Δz,QF}, D)) > Q_E then
            (z, QF, Q_E) ←
            search(c_k, D, z, QF, Q_E, +1, 0)
        end
    end
    return (z, QF, Q_E)
end

function DU_Search(c_k, D, z, QF, Q_E)
begin
    if Q_E(c_k^{z,QF+ΔQF}, D)) > Q_E then
        (z, QF, Q_E) ← search(c_k, D, z, QF, Q_E, 0, +1)
    else
        if Q_E(c_k^{z,QF−ΔQF}, D)) > Q_E then
            (z, QF, Q_E) ←
            search(c_k, D, z, QF, Q_E, 0, −1)
        end
    end
    return (z, QF, Q_E)
end

function search(c_k, D, z_o, QF_o, Q_E, λ_z, λ_QF)
begin
    z ← z_o + λ_z Δz, QF ← QF_o + λ_QF ΔQF
    while Q_E(c_k^{z,QF}, D)) > Q_E do
        Q_E ← Q_E(c_k^{z,QF}, D)
        z ← z + λ_z Δz, QF ← QF + λ_QF ΔQF
    end
    return (z, QF, Q_E)
end
```

Algorithm 1: LRDU greedy search pseudo-code

Theoretically, the number of transcoding operations can be very high. However, in practice, experimental results (see section VI) showed that the number of transcoding operations was between 4 and 7 and, on average, was 5.2. This is quite reasonable, since we took advantage of the prediction of the estimated transcoding parameters, which were very reliable. Unlike an exhaustive search, we start from an estimated point that is relatively close to the optimal. Thus, we are still in the same range of transcoding operations as with previous methods.

V. Experimental Setup

To demonstrate the effectiveness of the proposed dynamic framework and methods, a set of 120 OpenOffice Impress presentations were created. To this end, we created a Java-based application that uses OpenOffice APIs (known under the name of UNO) to create these Impress documents [17]. To facilitate the validation, each presentation document was composed of one slide, with these slides themselves composed of text-boxes and images, and their positions set randomly. To span a wide variety of slide characteristics, quantized values, representing the percentage of areas occupied by images (I) and text-boxes (T), were used as follows:

$$I \in \{0\%, 10\%, 20\%, \ldots, 100\%\}$$
$$T \in \{0\%, 10\%, 20\%, \ldots, 100\%\}$$

Let V be this set of slide documents.

We wanted to create a set of optimally adapted contents, using an exhaustive method, to be used as references (ideal targets to attain). These optimally adapted contents would be used to evaluate the performance of various methods. To that end, using the OpenOffice HTML filter (the JPEG-based version), a set of Web pages were created by varying the transcoding parameters values $\widetilde{z} \in \{0.1, 0.2, 0.3, \ldots 1\}$ and $\widetilde{QF} \in \{10, 20, 30, \ldots 100\}$. Let D be a target mobile device with a resolution of 640×360, and connected to a communication network characterized by: $N_B(D) = 50$kbps and $N_L(D) = 3$ms. To compute the $\mathcal{Q}_E$ of these adapted contents, $\mathcal{Q}_V$ and $\mathcal{Q}_T$ were evaluated.

The adapted content visual quality, $\mathcal{Q}_V$, was computed using SSIM [15]. To evaluate the quality of a transcoded image, SSIM requires both the original image and its transcoded version. In our case, we had a slide and not an image. Therefore, for each slide, we created a JPEG image using $z = 1$ and $QF = 80$ to be used as original images. Moreover, to use SSIM, we needed to provide a resolution at which the two images (original and transcoded) should be scaled for comparison. Since the slides are to be rendered on D and their default resolution, as rendered on PC by OpenOffice, was 1058×794, the resolution to be used by SSIM can be determined, following the methodology of [8], by: $\min\left(\frac{640}{1058}, \frac{360}{794}\right) \approx 45\%$. This suggests a comparison of images at a resolution of 40% of the original image resolution ($\widetilde{z} = 0.4$). Therefore, (3) becomes:

$$\mathcal{Q}_V(c_k^{z,QF}) = SSIM_{0.4}(\mathcal{I}_k^{1,80}, \mathcal{I}_k^{z,QF}) \qquad (15)$$

Note that, the SSIM index exhibits a highly non-linear relationship with the MOS (Mean Opinion Score), which represents a true measure of the human perception of image quality [18]. Therefore, to address the third element regarding the $\mathcal{Q}_E$ design [12] (see section III), the computed SSIM values were not used directly, but rather, were converted into their corresponding continuous MOS values.

Now, we turn to the evaluation of the $\mathcal{Q}_T$ and $\mathcal{Q}_E$ of these adapted contents. Let us suppose that the appreciation of the end-user's of this mobile phone regarding the waiting time is determined by $\alpha = 5s$ and $\beta = 10s$ (see Fig. 1) [19], [20]. Since the file size prediction error in [9] can reach 15% and $\mathcal{Q}_E$ is highly sensitive to the delivery time, we are increasing the predicted file size by 15% to ensure that the transcoded file size will not lead to drastically lower $\mathcal{Q}_T$ than predicted (we sacrifice the quality slightly to ensure a good $\mathcal{Q}_T$ as it is much more sensitive to the file size). To facilitate the validation, the values of $S_L(D)$ and $T_L(c_k^{z,QF})$ (see (4)) were set to zero. Thus, using these values and the target mobile device's characteristics ($N_B(D)$ and $N_L(D)$), the $\mathcal{Q}_T$ and $\mathcal{Q}_E$ of each adapted content, was computed, using all possible parameters $\widetilde{z}$ and $\widetilde{QF}$, and the optimal was determined by solving (1). That is, for each original content c_k (slide), its optimal adapted version, using the exhaustive method, was computed and stored in vector $W^*_{E,k}$ defined, as follows:

$$W^*_{E,k} = \left[c_k, z^*(c_k, D), QF^*(c_k, D), \mathcal{Q}_E(c_k^{z^*(c_k,D),QF^*(c_k,D)}, D)\right]$$

Besides, to show the behavior of the FQF (fixed quality factor) method, we computed its transcoding parameters using a fixed quality factor of 80 and a scaling parameter of 0.4, which corresponds to the resolution of D. Again, for each original content c_k, we used these parameters to create its corresponding adapted content, computed its $\mathcal{Q}_E$ and stored all the information in a vector as follows:

$$W^*_{FQF,k} = \left[c_k, 0.4, 80, \mathcal{Q}_E(c_k^{0.4,80}, D)\right]$$

VI. Experimental Results and Discussion

Using the proposed framework, for each method and for each original content c_k of the set V, the near-optimal transcoding parameters were computed and used to create their corresponding adapted content versions. For each of these adapted contents, the $\mathcal{Q}_E$ was computed. The data obtained by each method was stored in its corresponding vectors. For method i and content c_k, we have:

$$W^*_{i,k} = \left[c_k, z^*_{i,k}, QF^*_{i,k}, \mathcal{Q}_E(c_k^{z^*_{i,k},QF^*_{i,k}}, D)\right]$$

where $z^*_{i,k} = z^*_i(c_k, D)$ and $QF^*_{i,k} = QF^*_i(c_k, D)$.

For each slide c_k, the near-optimal $\mathcal{Q}_E$ obtained by each method as well as those computed by the exhaustive (from $W^*_{E,k}$) and FQF (from $W^*_{FQF,k}$) methods were plotted. To make this visible, all the $\mathcal{Q}_E$ obtained were sorted according to those of the exhaustive method, as shown in Fig. 2. On the whole, the proposed methods have a $\mathcal{Q}_E$ close to that of the exhaustive method. However, the $\mathcal{Q}_E$ obtained by the FQF method is very variable, and this is highly visible for lower values of optimal $\mathcal{Q}_E$. FQF is especially problematic for large documents and low network bitrates.

Now, to show the improvements brought by methods 2 to 5 over method 1, the relative gains in $\mathcal{Q}_E$ were computed. The gain for content c_k and method i is computed as follows:

$$\frac{\mathcal{Q}_E(c_k^{z^*_i(c_k,D),QF^*_i(c_k,D)}, D) - \mathcal{Q}_E(c_k^{z^*_1(c_k,D),QF^*_1(c_k,D)}, D)}{\mathcal{Q}_E(c_k^{z^*_1(c_k,D),QF^*_1(c_k,D)}, D)} \times 100\%$$

The computed $\mathcal{Q}_E$ relative gains were plotted as scattered points, as depicted in Fig. 3. The sub-figures 3(a), 3(b), 3(c)

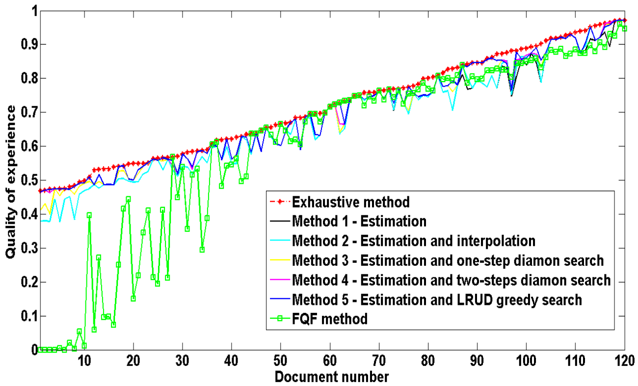

Figure 2. $\mathcal{Q}_E$ obtained by each method vs. that obtained by the exhaustive and FQF methods. The slides are sorted according to the exhaustive $\mathcal{Q}_E$

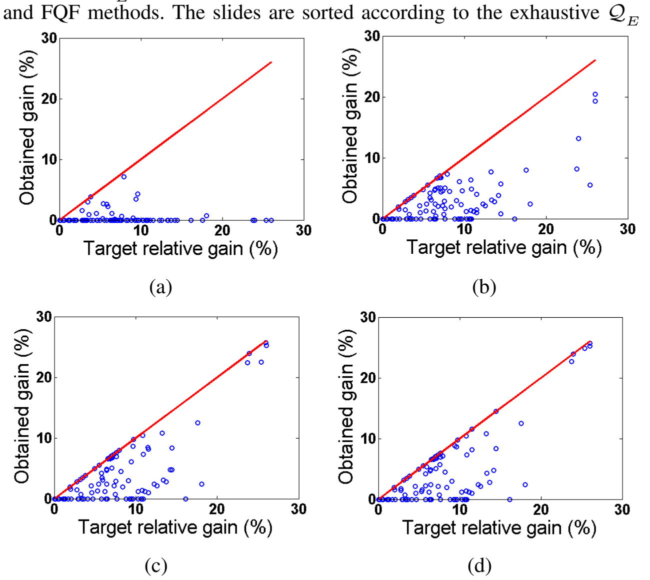

Figure 3. Obtained $\mathcal{Q}_E$ relative gains for methods 2, 3, 4, and 5 with respect to method 1. (a) Method 2 - Estimation and interpolation, (b) Method 3 - Estimation and one-step diamond search, (c) Method 4 - Estimation and two-steps diamond search, (d) Method 5 - Estimation and LRDU greedy search

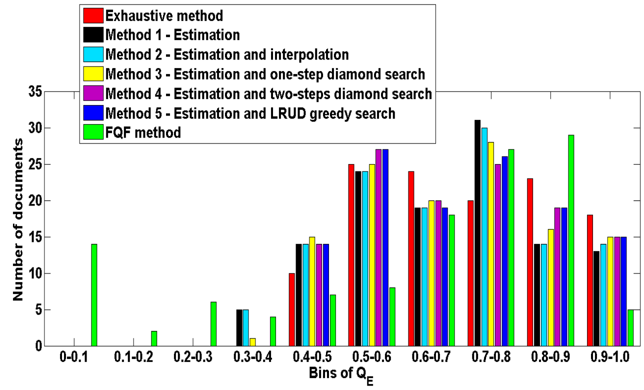

Figure 4. Performance of each method by $\mathcal{Q}_E$ slices of 10%

and 3(d) show the Q_E relative gain obtained by methods 2, 3, 4 and 5, respectively. The diagonal line represents the target relative gains, which were computed from the computed optimal adapted contents ($W^*_{E,k}$). The scattered points represent the different slides, and their positions indicate the obtained relative gain vs. the target relative gain.

Another view that shows the number of documents whose computed $\mathcal{Q}_E$ was improved by the proposed methods is depicted in Fig. 4. To show this aspect graphically, the $\mathcal{Q}_E$ range has been split into 10 bins ([0,0.1],]0.1,02],...,]0.9,1]), and the documents that are in the same $\mathcal{Q}_E$ bin were counted and their numbers plotted as a histogram. The figure shows that, using the FQF method, almost 25% of the documents are present in the first three bins (poor image quality bins), while these bins are empty for the other methods (ours in addition to the exhaustive one). For the FQF method, unlike the other methods, there is very few documents in the last bin (best image quality bin). This is also visible in Fig. 2 for very low or very high $\mathcal{Q}_E$. Besides, this figure shows a comparison between the proposed methods in terms of accuracy compared to the exhaustive method. Note that, except for method 4, the higher the complexity of the method in use the higher the accuracy (the number of documents in each bin gets closer to that of the exhaustive method).

In Fig. 5, the percentage of average $\mathcal{Q}_E$ obtained by each method, compared to that obtained by the exhaustive method, versus the average complexity of these methods, are plotted. On average, the $\mathcal{Q}_E$ obtained by method 1 (prediction only) is close to that obtained by the exhaustive method (94%) and is even closer when the other methods are used (from 94% to 97%). Method 5 reached 97%, which is 3% far from optimality with a complexity of close to 5 operations.

Though the average improvement in $\mathcal{Q}_E$ obtained by methods 2 to 5 is relatively small, this figure hides the fact that the obtained $\mathcal{Q}_E$ follows that needed to reach the optimal $\mathcal{Q}_E$. This is shown by Fig. 3, in which we see that when the target's relative gain is bigger, the improvement obtained is also bigger; conversely, the average relative gain is small because for some slides, the target gain is also small as well. It is also justified by the fact that the $\mathcal{Q}_E$ obtained by method 1 is around 94%, which is already good. Fig. 3 is very interesting as it shows that, using these methods, we can reach the optimal $\mathcal{Q}_E$ for a large number of slides, especially using method 5 (the scattered points that are on the diagonal line). Statistically speaking, from the set of documents V, 14% and 29% of them reached optimality when methods 1 and 5 were used, respectively. Also, the average deviation from optimality for the proposed methods ranged from 6% to 3%, with a complexity varying from 1 to 5 transcoding operations.

Lastly, note that the scenario presented in this paper is rather conservative. From Fig. 2, we can see that the delivery time is problematic for only about 33% of the slides, where we see the FQF method performing very poorly (obviously the transport is problematic since FQF always yields good visual quality by setting $QF = 80$). If

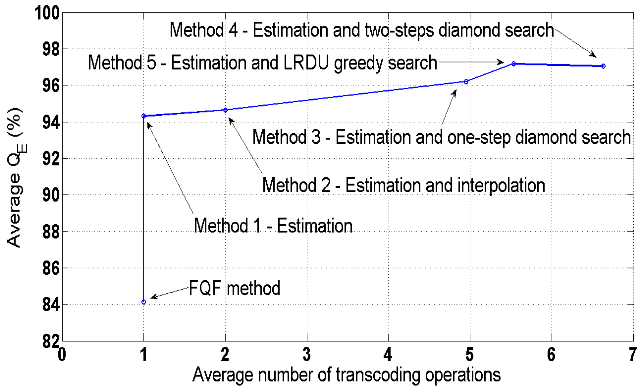

Figure 5. Obtained average $\mathcal{Q}_E$ vs. average complexity

we used a scenario with a lower bit rate, the number of problematic slides would increase for FQF and could easily reach 100% with a low enough bit rate. This would make the FQF method totally unusable. Of course, the opposite is also true, if the bit rate is high enough, the method FQF would provide excellent quality of experience. An advantage of the proposed methods is that they perform as well as possible under any circumstances. This is important as the bit rate can vary significantly during a Web conference session.

VII. Conclusion

Dynamically identifying the optimal transcoding parameters in the context of slides documents adaptation is not a straightforward task as the number of parameters combinations could be very high. To tackle this problem, in this paper, we proposed a dynamic framework composed of five methods to estimate near-optimal transcoding parameters. Each of the methods has its specific performance and complexity. The methods were tested on the adaptation of OpenOffice impress slides into JPEG-based Web pages.
The first method was based on the prediction of the quality and file size of JPEG images subject to change of their resolution and quality factor. It represents a good starting point, but exhibits some imprecisions, which were improved by the other four methods, which involve different levels of accuracy and complexity (number of transcoding operations). For some instances, the optimal transcoding parameters were reached and for others, the accuracy was improved significantly. Future work needs be conducted to show the applicability of these methods in other use cases, such as in the adaptation of slides into XHTML-based Web pages, which are composed of text and images.

References

[1] PresenterNet, "Interactive Web Conferencing." [Online]. Available: http://www.presenternet.com/

[2] Adobe Systems Incorporated, "Adobe Connect 8: Web Conferencing for the Enterprise." [Online]. Available: http://www.adobe.com/products/adobeconnect.html

[3] R. Mohan and J. Smith, "Adapting Multimedia Internet Content for Universal Access," *IEEE Transactions on Multimedia*, vol. 1, no. 1, pp. 104–114, Mar 1999.

[4] W. Lum and F. Lau, "User-Centric Content Negotiation for Effective Adaptation Service in Mobile Computing," *IEEE Transactions on Software Engineering*, vol. 29, no. 12, pp. 1100–1111, Dec 2003.

[5] R. Jan, C. Lin, and M. Chern, "An Optimization Model for Web Content Adaptation," *Computer Networks*, vol. 50, no. 7, pp. 953–965, 2006.

[6] Google, "GoogleDocs Mobile." [Online]. Available: http://www.google.ca/mobile/docs/index.html

[7] Zoho, "Zoho Mobile." [Online]. Available: https://show.zoho.com/login.do

[8] 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*. IEEE, Mar 2009, pp. 68–74.

[9] S. Pigeon and S. Coulombe, "Computationally Efficient Algorithms for Predicting the File Size of JPEG Images Subject to Changes of Quality Factor and Scaling," in *2008 24th Biennial Symposium on Communications*. IEEE, Jun 2008, pp. 378–382.

[10] S. Coulombe and S. Pigeon, "Low-complexity Transcoding of JPEG Images With Near-optimal Quality Using a Predictive Quality Factor and Scaling Parameters." *Image Processing, IEEE Transactions on*, vol. 19, no. 3, pp. 712–21, Mar 2010.

[11] H. Louafi, S. Coulombe, and U. Chandra, "Quality Prediction-Based Dynamic Content Adaptation Framework Applied to Collaborative Mobile Presentations," *IEEE Transactions on Mobile Computing*, vol. 99, 2012.

[12] F. Kuipers, R. Kooij, D. De Vleeschauwer, and K. Brunnström, "Techniques for Measuring Quality of Experience," in *Wired/Wireless Internet Communications*, ser. Lecture Notes in Computer Science. Springer-Verlag Berlin/Heidelberg, 2010, vol. 6074, pp. 216–227.

[13] L. Lee and R. Anderson, "A Comparison of Compensatory and Non-Compensatory Decision Making Strategies in IT Project Portfolio Management," 2009. [Online]. Available: http://aisel.aisnet.org/irwitpm2009/9

[14] A. Dieckmann, K. Dippold, and H. Dietrich, "Compensatory versus Noncompensatory Models for Predicting Consumer Preferences," *Judgment and Decision Making*, vol. 4, no. 3, pp. 200 – 213, 2009.

[15] 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*, vol. 13, no. 4, pp. 600–612, 2004.

[16] The MathWorks, "Z-shaped built-in membership function." [Online]. Available: http://www.mathworks.com/help/toolbox/fuzzy/zmf.html

[17] OpenOffice.org, "The OpenOffice.org API Project." [Online]. Available: http://api.openoffice.org

[18] H. Sheikh, M. Sabir, and A. Bovik, "A Statistical Evaluation of Recent Full Reference Image Quality Assessment Algorithms," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, Nov 2006.

[19] G. Ryan and M. Valverde, "Waiting in line for Online Services: A Qualitative Study of The User's Perspective," *Information Systems Journal*, vol. 16, no. 2, pp. 181–211, Apr 2006.

[20] F. Nah, "A study on Tolerable Waiting Time: How Long Are Web Users Willing to Wait?" *Behaviour & Information Technology*, vol. 23, no. 3, pp. 153–163, Jan 2004.