# Optimal Adaptation Decision-Taking for Terminal and Network Quality of Service

Debargha Mukherjee, *Member, IEEE*, Eric Delfosse, Jae-Gon Kim, *Member, IEEE*, and Yong Wang,

***Abstract*—In order to cater to the diversity of terminals and networks, efficient and flexible adaptation of multimedia content in the delivery path to end consumers is required. To this end, it is necessary to associate the content with metadata that provides the relationship between feasible adaptation choices and various media characteristics obtained as a function of these choices. Further, adaptation is driven by specification of terminal, network, user preference or rights based constraints on media characteristics that are to be satisfied by the adaptation process. Using the metadata and the constraint specification, an adaptation engine can take an appropriate decision for adaptation, efficiently and flexibly. MPEG-21 Part 7 entitled Digital Item Adaptation standardizes among other things the metadata and constraint specifications that act as interfaces to the decision-taking component of an adaptation engine. This paper presents the concepts behind these tools in the standard, shows universal methods based on pattern search to process the information in the tools to make decisions, and presents some adaptation use cases where these tools can be used.**

***Index Terms*—MPEG-21, Digital Item Adaptation, terminal and network constraints, decision-taking, adaptation, transcoding, requantization, rate shaping, scalable bit-streams.**

## I. Introduction

HETEROGENEOUS multimedia content delivery infrastructures and consumption devices present a huge obstacle in universal media access. Indeed, consumers use a growing variety of terminals to access multimedia content, over an equally diverse variety of networks with dynamically varying throughputs. To maximize consumer experience and ensure Quality of Service (QoS) commensurate with terminal and network capabilities and conditions, as well as user preferences, it is essential to adapt multimedia content in the delivery path to end consumers. Note here QoS is used loosely and does not correspond to network level guarantees.

Additionally, the set of rich media content and formats to be delivered is growing fast. This justifies a drive towards adaptation engines or modules thereof that use a universal processing model – which do not need frequent upgrades to support new formats and can even support proprietary ones.

Adaptation of various standardized formats has been extensively studied in recent years [1]-[10]. Invariably the focus of such work is computational efficiency of adaptation, since full decoding followed by re-encoding with parameters so that the terminal and network constraints are met, is often infeasible from complexity and delay considerations. This includes rate adaptation and resolution conversion for Discrete Cosine Transform (DCT) coded images [1][2], rate adaptation and spatio-temporal resolution conversion for pre-encoded MPEG-1/2/4 videos [3][4][5], object based transcoding [6], and rate-distortion optimized [7] as well as rate-distortion-complexity optimized [8] transcoding and streaming.

In many of these cases, there is a compute intensive decision-taking process involved for choosing the right set of parameters for adaptation that yields an adapted version of the content meeting terminal and network constraints. The computational efficiency of adaptation can be greatly enhanced if this process could be simplified, in particular by providing metadata that conveys pre-computed relationships between feasible adaptation parameters and media characteristics obtained by selecting them. This metadata is also the only means of providing information that cannot be directly obtained from a compressed bit-stream, such as distortion or fidelity measures compared to the original uncompressed data. The decision-taking process then just uses the information in the metadata along with terminal and network constraints to take decisions, without requiring any extraction of information through complex content manipulation. Further, a universal processing model for the decision-taking process in an adaptation engine can be derived, so that descriptions and engines created by different parties can interoperate.

Digital Item Adaptation (DIA) [11][12] is Part 7 of the interoperable Multimedia Framework currently being developed in the ISO/IEC MPEG standardization committee as MPEG-21 [13][14]. DIA aims to standardize various adaptation related metadata including those supporting decision-taking and the constraint specifications.

The rest of the paper is organized as follows. In Section II, the model for an adaptation engine is presented along with an introduction to various DIA components. Section III presents the decision-taking framework in detail. In Section IV, the

Manuscript received April 30, 2004, revised June 30, 2004. Part of this work was funded by the IST-OZONE project (IST-2000-30026).

Debargha Mukherjee is with Hewlett Packard Laboratories, Palo Alto, CA, USA (phone: 650-236-8058; fax: 650-857-2951; e-mail: debargha@ hpl.hp.com).

Eric Delfosse is with Interuniversity MicroElectronics Center (IMEC), Kapeldreef 75, B-3001 Leuven, Belgium (Phone: +32 16 28 10 68, Fax: +32 16 28 15 15, E-mail: delfosse@imec.be).

Jae-Gon Kim is with the Broadcasting Media Research Group, Electronics and Telecommunications Research Institute (ETRI), Korea (e-mail: jgkim@etri.re.kr).

Yong Wang is with the Electrical Engineering Department, University of Columbia University, NY, USA (e-mail: ywang@ee.columbia.edu).

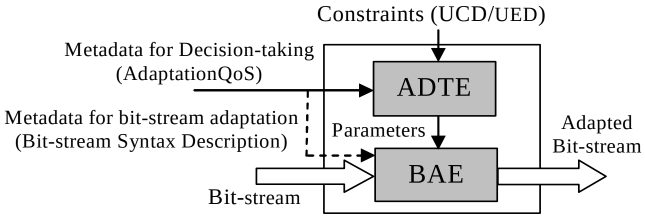


Fig. 1. Adaptation engine model.

optimization problem to be solved by a universal decision-taking process is described, along with some strategies for solving it. In Section V we show a variety of adaptation use cases involving various formats where the framework can be effectively employed to make adaptation decisions. Finally, conclusions and future directions are presented in Section VI.

## II. Adaptation Engine Model

The model of an adaptation engine envisaged in DIA is shown in Fig. 1. The engine absorbs various kinds of metadata to enable fast and efficient adaptation, for instance, based on terminal and network constraints. Functionally the engine is decoupled into two modules: an Adaptation Decision-Taking Engine (ADTE) that absorbs metadata that aids decision-taking as well as constraint specifications to make appropriate adaptation decisions; and a Bit-stream Adaptation Engine (BAE) that uses the decisions provided by the ADTE to perform the actual bit-stream adaptation.

An MPEG-21 DIA tool called AdaptationQoS represents the metadata supporting decision-taking in the figure. Another DIA tool called Universal Constraints Description (UCD) represents explicit constraint specifications. The adaptation constraints may also be specified implicitly by a variety of Usage Environment Descriptions (UED) providing network, terminal, user and natural environment characteristics and preferences that cover a major part of the standard. Examples of UED include display capabilities, audio/video capabilities of terminal, network characteristics, and so on. All of the above metadata is processed by the ADTE to take decisions in the form of a set of adaptation parameters used by the BAE.

The BAE in general can be specific for a given format. Its operation should be relatively simple given the decisions already made, so that the overall efficiency of adaptation is improved. For the special case of scalable bit-streams, such as JPEG2000 images [15] or fully scalable video proposed for standardization in MPEG-21 Part 13 [16][17][18], it has been shown [19][20] that by associating the content with additional metadata, both the ADTE and the BAE can have universal processing models. This leads to adaptation engines for scalable bit-streams that are fully format-independent. This additional metadata in DIA consists of Bit-stream Syntax Description (BSD) [21][22] and a transformation stylesheet for the BSD. In the current paper, we focus on metadata that enables decision-taking and constraint specifications.

Note that while the metadata and constraint specifications are normative in DIA, the implementation of the ADTE and the BAE using them is non-normative. Further, DIA does not restrict in any way the delivery architecture within which a compliant engine is used is practice. The model is applicable in a variety of scenarios, irrespective of whether adaptation is conducted in the server/transmitter or in a gateway, whether the ADTE and BAE operations are distributed or occur at the same node, etc. While these are interesting issues, such architectural options and their suitability for different delivery scenarios is beyond the scope of this paper. DIA also does not address network transport issues for either the metadata or the bit-stream, which is left entirely to a system implementation.

## III. Decision-Taking Framework

The purpose of the ADTE in an adaptation engine is to make an appropriate decision on how to adapt an input bit-stream, from among a set of available choices conveyed by the bit-stream description, based on specified input constraints. Furthermore, if the ADTE is to use a universal processing model it must not use any processing that is based on an understanding of the characteristics for specific media-types. This can only be accomplished if the decision-taking problem is expressed in the universal language of mathematics.

In MPEG-21 DIA, decision-taking is cast as a constrained optimization problem involving algebraic *variables* that represent adaptation parameters, media characteristics, usage environment inputs, or any combinations thereof. The solution, yielding the decision, can then be computed by a universal process independent of what the variables represent.

The framework provides for the decision-taking functionality to be differentiated with respect to sequential logical segments corresponding to partitionings such as Group of Pictures (GOP), Region of Interest (ROI), Tile, Frame etc., referred to as the *adaptation unit*. All variables are differentiated by adaptation unit. For streamed content, the adaptation unit in many cases is also a unit of transmission comprising a bit-stream segment and corresponding metadata for decision-taking.

For example, consider a fully scalable video bit-stream [17] which contains simultaneous temporal, spatial and SNR scalability. It may also have color scalability, but we do not consider this case here. The video bit-stream is organized into multiple sequentially transmitted groups of frames (GOF), each typically containing 16 or 32 frames. The GOF constitutes the adaptation unit abstraction. Every GOF is coded jointly into several temporal, spatial and SNR layers. During adaptation, the number of layers included for each GOF is represented by variables `NTEMP`, `NSPATIAL` and `NSNR`, which form a 3-dim logical hypercube structure for the decision space as shown in Fig. 2. Here adaptation removes the light blocks from the original. Note the exact algorithm used to compress the layers is immaterial, since the model will likely remain the same, irrespective of the algorithm. Other variables such as `BITRATE` can be defined as a function of `NTEMP`, `NSPATIAL` and `NSNR`, for each GOF.

Denote the set of variables for the *n*th adaptation unit as vector $\mathbf{I}[n] = \{i_0[n], i_1[n],..., i_{M-1}[n]\}$, $n = 0, 1, 2, \ldots$, where M

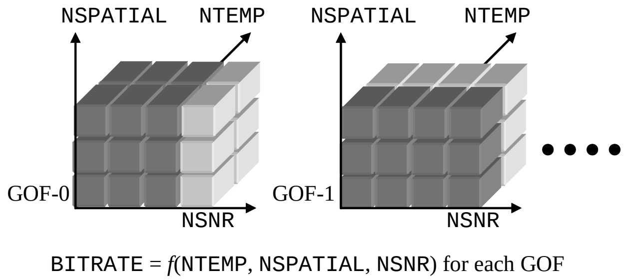


Fig. 2. Adaptation variables for fully scalable video.

is the number of variables. For each adaptation unit $n$, the optimization problem to be solved is given by:

$$\text{Maximize or Minimize } \{O_{n,j}(\mathbf{I}[n], \mathbf{H}[n])\},\ j=0,1,...,J_n-1$$
$$\text{subject to: } L_{n,k}(\mathbf{I}[n], \mathbf{H}[n]) = \texttt{true},\ \ k=0,1,\ldots,K_n-1$$

where $L_{n,k}(\mathbf{I}[n], \mathbf{H}[n])$, are Boolean expressions called limit constraints, and $O_{n,j}(\mathbf{I}[n], \mathbf{H}[n])$, are numeric expressions called optimization constraints. The vector $\mathbf{H}[n]$ in the expressions of $O_{n,j}$ and $L_{n,k}$ represents the history of all past decisions for adaptation units 0,1,…, $n$-1. In other words, if $\mathbf{I}^*[n]$ represents a solution to the problem for the $n^{th}$ adaptation unit, we can denote: $\mathbf{H}[n] = \{\mathbf{I}^*[0], \mathbf{I}^*[1], \ldots, \mathbf{I}^*[n-1]\}$. An ADTE makes decisions for the vectors $\mathbf{I}[n]$ sequentially for $n$ = 0,1,2,… The dependency on history of past decisions is needed in certain cases, as in Section V.D.

The number of optimization constraints ($J_n$) is arbitrary. If $J_n$=0, any solution in the *feasible* region – defined as the region of the solution space where the limit constraints are satisfied – is acceptable. The case $J_n$=1 defines a common single-criterion optimization problem with usually a unique solution. The case $J_n$>1 defines a multi-criteria problem [23][24], where any Pareto optimal solution in the feasible region is accpetable.

The DIA tools, AdaptationQoS and UCD used in combination, support the above decision-taking mechanism. Variables are termed IOPins and are defined in the AdaptationQoS description. In cases involving multiple adaptation units, there is one IOPin defined in AdaptationQoS that indexes successive adaptation units, while other IOPins are functions of this IOPin. The AdaptationQoS description also conveys the known interdependencies between IOPins using various data types called *modules* defined in the tool. These include look-up tables, numeric functions represented by an expression stack, or lists of values assumed for each adaptation choice termed utility functions. Note that the UCD or AdaptationQoS can still reference values from the UED, but the processing is driven by UCD or AdaptationQoS rules to ensure semantics-independent operation. Fig. 3 shows an example AdaptationQoS description with a variety of IOPins connected by module definitions as well as a UCD. This figure is further explained in Section IV.A. Specific examples of AdaptationQoS and UCD are provided in Section V.

Note that the semantics of the IOPins are immaterial within the ADTE because they are simply regarded as mathematical variables to solve in a generic optimization problem. However, they are very much important at the provider and receiver ends or other nodes from where the AdaptationQoS or UCD originates. That is because, the UCD creator in many cases would not be expected to know the identifier of the IOPin (variable) defined in the provider side AdaptationQoS description, corresponding to a given semantics. In order to enable linking of the UCD to the right IOPins in AdaptationQoS, DIA creates a number of dictionaries termed *classification schemes* to standardize terms having pre-defined semantics for representing media characteristics, usage environment characteristics, and segment decompositions. The AdaptationQoS associates the IOPins it defines with terms that are the closest in semantics, while the UCD creator uses the same terms to specify the problem, rather than use identifiers of the IOPins directly. The ADTE simply performs a textual match of the classification scheme terms used in AdaptationQoS and UCD to know how the constraints specified in UCD using semantics terms map to IOPins.

## IV. ADTE Optimization

Generally speaking, an ADTE can have several inputs to it, comprising an AdaptationQoS, and several UCDs or UEDs from various sources. Based on these inputs, the ADTE needs to make appropriate adaptation decisions, by solving one or more constrained optimization problems [25]. We first discuss the single UCD case, and then present options to cover multiple UCDs originating from different sources.

### A. *Optimization problem involving free variables*

The AdaptationQoS declares and defines several IOPins, some of which are independent, while others depend on other IOPins. Among the independent IOPins, some are assigned based on usage environment inputs either explicitly through the UCD or through data semantically referenced from a UED. Additionally, in cases involving multiple adaptation units there is one independent adaptation unit IOPin. The remaining independent IOPins, denoted $\mathbf{x}[n]$, comprise $N$ *free* variables that need to be optimized. This classification of IOPins is illustrated in Fig. 3. At the start of the optimization process, the ADTE performs simple analyses of the UCD and AdaptationQoS to determine the free IOPins. Then it performs

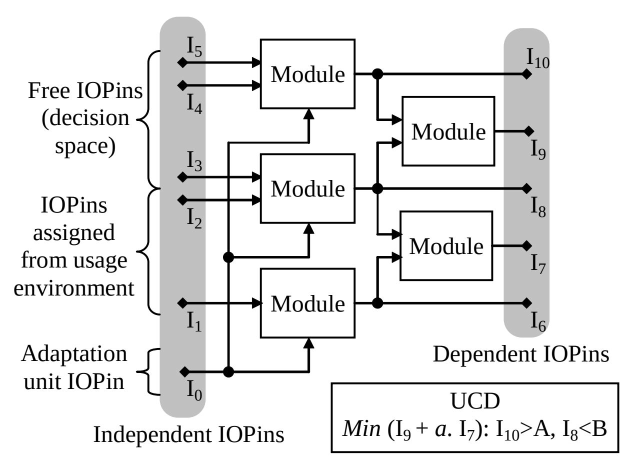


Fig. 3. Illustration of IOPins in AdaptationQoS.

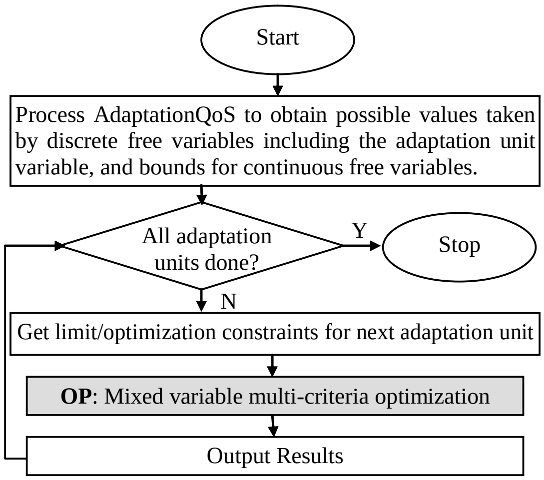


Fig. 4. Top-level ADTE operation.

the optimization for each adaptation unit sequentially.

A free IOPin can either be discrete, i.e. taking values from a finite discrete set, or continuous but bounded within two limits as provided in the IOPin definition in the AdaptationQoS. Denote $\mathbf{x}[n]=\{\mathbf{x}_d[n]' \ \mathbf{x}_c[n]'\}'$, where $\mathbf{x}_d[n]$ is the vector of $N_d$ discrete free variables, and $\mathbf{x}_c[n]$ is the vector of $N_c$ continuous free variables, and $N = N_c + N_d$. Also, note that the set of values for a discrete IOPin can either be unordered – corresponding to categorical variables, or ordered. Thus, the problem to be solved is a *mixed-variable multi-criteria optimization problem with general constraints* [23][25][31] defined over free IOPins for each adaptation unit. Fig. 4 shows the top level ADTE optimization flowchart.

For a universal ADTE, the generality of the optimization method used is of vital importance. Further, because the syntax of the UCD in MPEG-21 DIA allows representing arbitrarily non-linear functions for limit and optimization constraints, it is desirable to use methods that do not rely on derivative computations, but only black-box function computations as defined in the AdaptationQoS or the UCD.

### B. Search/Optimization Strategies

For the problem as defined above, we consider first the discrete-only case, then the continuous-only case, and then extend to tackling the general mixed continuous-discrete case.

*1) Discrete-only case:*

In this case, all the free IOPins are discrete variables: $\mathbf{x}[n]= \mathbf{x}_d[n]$, $N = N_d$, $N_c = 0$. In commonly occurring adaptation decision-taking scenarios, it is sufficient to search for the best parameters among a set of available discrete choices provided in the AdaptationQoS. Further, the number of free variables and the set of choices for each are usually small, so that even search by total enumeration is feasible. In this case, it is always possible to find a generic solution, irrespective of the nature of the functions and for any number of optimization constraints. Furthermore, the exhaustive search method suits well the nature of the UCD specification in DIA, where the limit constraints are represented as Boolean functions.

In multi-criteria optimization literature [23], a point $\mathbf{y}$ is said to *dominate* another point $\mathbf{z}$ with respect to a set of optimization metrics, if one of the metrics evaluated at $\mathbf{y}$ is better than that evaluated at $\mathbf{z}$, and no other metric evaluated at $\mathbf{y}$ is worse than that evaluated at $\mathbf{z}$. A point $\mathbf{y}$ is said to be Pareto optimal, if there is no other feasible point that dominates $\mathbf{y}$. The goal is to find the set of all Pareto optimal solutions for an arbitrary number of optimization constraints.

Given $N$ free variables for each adaptation unit, the ADTE starts with a null initialized list of solutions. Then it generates an $N$-tuple for each candidate solution, and evaluates the limit constraints to test feasibility. If feasible, the optimization metrics are evaluated and compared with the current running list of solutions to check for mutual domination. If the candidate is dominated by another existing solution, it is discarded. If not, the candidate is added to the list, but existing solutions that are dominated by the candidate if any, are discarded from the list. Once all candidates have been processed, the list yields the Pareto optimal solution set. Any particular solution from this set can be chosen as the final decision. A flowchart for the procedure is shown in Fig. 5 (a).

*2) Continuous-only case:*

In this case, all free IOPins are continuous variables: $\mathbf{x}[n] = \mathbf{x}_c[n]$, $N = N_c$, $N_d = 0$. We consider first the single-criterion problem, and then generalize to the multi-criteria case.

A family of unconstrained optimization methods called *direct search* [26] that rely only on function computations at given points, is well suited for black-box optimization problems such as the ADTE. A subclass of such methods termed *pattern search* was recently generalized under a theory of Generalized Pattern Search (GPS) [26][27][28] with strong convergence properties. GPS iteratively improves a solution by searching a set of trial steps around an incumbent solution using directions taken from an underlying lattice. If a step is found where the objective function improves, the incumbent moves to that point while the lattice scale remains the same or is increased. Else, the incumbent remains static, but the lattice scale is reduced. The simplest embodiment of GPS is *compass* or *co-ordinate search*, where trial steps are in the positive and negative directions of each variable for total $2N$ search steps.

Two ways to adopt GPS to handle constraints follow:

($i$) Convert a constrained problem into an unconstrained one by incorporating an *exterior penalty* term that penalizes the metric if limit constraints are violated based on the degree of violation. For a general constrained optimization problem:

$$Minimize\{ f(\mathbf{x}) : g_0(\mathbf{x}) \le 0, g_1(\mathbf{x}) \le 0, \ldots; h_0(\mathbf{x}) = 0, h_1(\mathbf{x}) = 0, \ldots\}$$

one form for the unconstrained function to minimize is:

$$\phi(\mathbf{x}, \mu) = f(\mathbf{x}) + \mu[\sum_i (Max[0, g_i(\mathbf{x})])^2 + \sum_i (h_i(\mathbf{x}))^2]$$

$\mu$ is a positive number called the *penalty factor* that is iteratively increased to a very large value, since the optimal solution is achieved only as $\mu \to \infty$. An advantage of exterior penalty methods is that an initial feasible point is not needed. The difficulty however is that from the unstructured Boolean UCD specification in DIA, algebraic forms of functions $g_i(.)$ and $h_i(.)$ as needed above, can only be obtained by a non-trivial UCD analysis and understanding process.

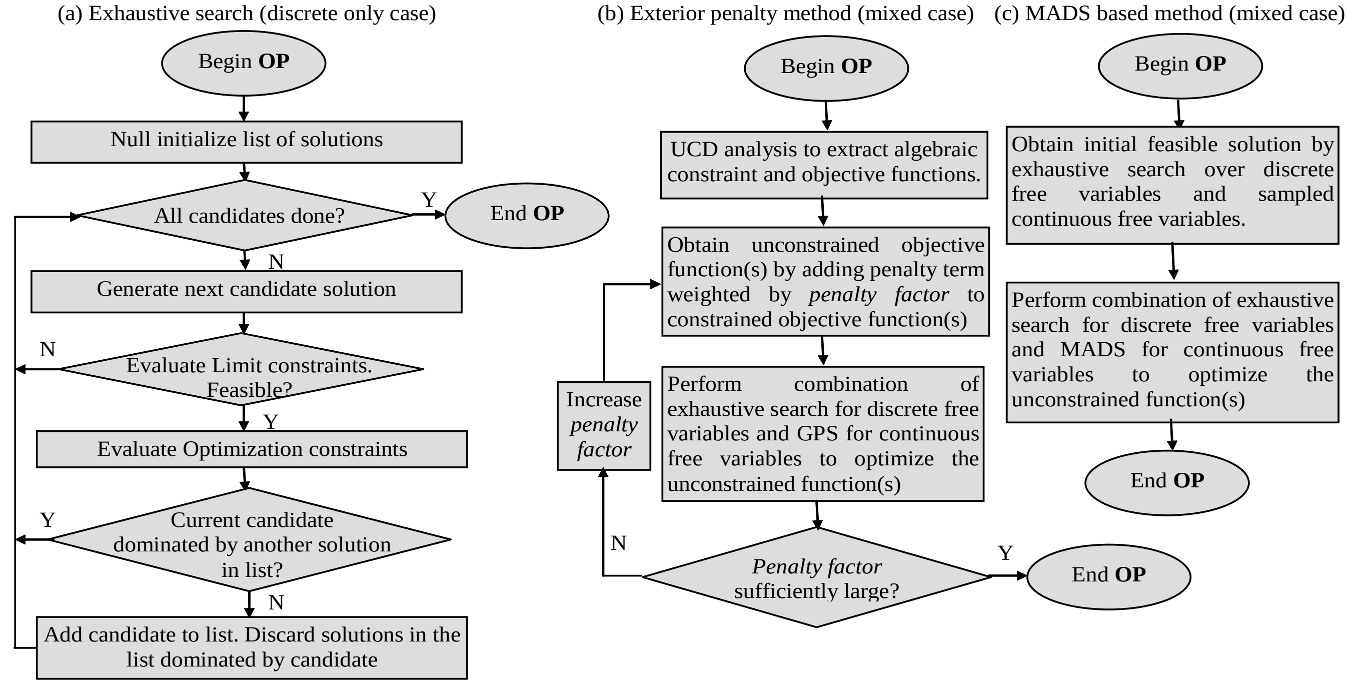


Fig. 5. Optimization strategies (a) Exhaustive search (discrete-only case); (b) Strategy based on exterior penalty method with GPS (mixed case) – requires UCD-analysis; (c) Strategy based on MADS (mixed case) – cannot handle equality constraints.

(*ii*) Use a variation of GPS that uses an *exact barrier* approach to simply disregard infeasible trial steps. Such a method works with Boolean black-box constraint functions returning a yes/no answer to indicate if a constraint is satisfied at a trial point or not, and requires no further UCD analysis. Recent work on Mesh Adaptive Direct Search (MADS) [29] specifically addresses this case and adds flexibility to GPS to search over a denser set of directions to ensure convergence. As in all barrier methods however, equality constraints cannot be handled. Nevertheless, for an arbitrary UCD this approach is promising. An initial feasible point is required in this approach. Since the AdaptationQoS provides the limits of the continuous variables, an exhaustive search in a coarsely sampled discretized space can be conducted to obtain it.

For multi-criteria problems, GPS methods are especially suitable, since only comparisons between points are needed. The criterion for comparing two points is simply changed to domination in the multi-criteria sense. Therefore, if any *one* Pareto optimal solution is desired, the methods described above apply to the multi-criteria case as well.

*3) Mixed discrete-continuous case:*

In this case, some variables are continuous while the rest are discrete: $N_c \neq 0$, $N_d \neq 0$. This is the most general case for the ADTE problem. GPS/MADS based methods as described in the previous sub-section are equally applicable to these problems, with special handling for the discrete variables [30][31]. Trial steps in the mixed case should cover not only the usual pattern derived from a lattice over continuous variables, but also all possible *neighbors* for discrete variables. For general categorical discrete variables, all values are neighbors of each other, and hence all possibilities need to be searched. Each step of the MADS iteration thus causes a change in either the discrete or the continuous variables.

The discussion for the continuous and mixed variables cases in this sub-section is summarized by showing two viable strategies for handling the **OP** block in Fig. 4 for the mixed variable case – one based on exterior penalty functions shown in Fig. 5(b), and the other based on MADS shown in Fig. 5(c).

### *C. Handling multiple UCDs*

When an ADTE receives multiple UCDs from various sources, each defining an independent optimization problem, it is necessary to combine them in some way. In such cases, it is reasonable to require that all the limit constraints from all the UCDs be satisfied. Thus, the feasible region for the solution is the intersection of feasible regions for the individual UCDs.

For the optimization constraints however, there are several options as follows:

(*i*) Combine all the optimization constraints from all UCDs into an integrated multi-criteria problem, and then search for a Pareto optimal solution.

(*ii*) Prioritize UCDs based on their sources, and only search for a Pareto optimal solution for optimization constraints in one of the UCDs, ignoring those from the rest. For instance, the optimization constraints in the UCD sent from the provider side may override those sent from the consumer side.

(*iii*) Obtain Pareto optimal solution sets independently for optimization constraints in different UCDs, and then search for a solution in the intersection of the individual solution sets. If the intersection is null, a prioritization mechanism is used to pick a solution from the union of the individual solution sets.

While option (*iii*) is in many ways the best in terms of fulfilling the motivation of the individual UCDs, this is also the hardest to implement, because the ADTE should not only solve multiple problems, but also store all possible Pareto optimal solutions for each. In contrast, options (*i*) and (*ii*) require solving only a single problem, and do not even need storing all possible solutions. We advocate adopting either of these two approaches or a combination thereof.

## V. ADAPTATION USE CASES

In this section we describe a few use cases involving real bit-streams, specifying the decision-taking problem that aids adaptation, what metadata needs to be provided to enable decision-taking, some possible constraints that may be used to drive adaptation and how the decisions made are used for the actual bit-stream adaptation. The use cases A, B and D below use discrete-only optimization, while C uses mixed variable optimization in the decision space of free variables.

### A. JPEG image adaptation

A classical adaptation problem for JPEG images (or MPEG intraframes) is that of requantization. The goal is to adapt a compressed JPEG image to a rate lower than the original. A problem associated with any viable method with JPEG is that there is no guarantee of the adapted rate achieved. Consequently multiple passes are required to avoid both violating the rate constraint or over-adaptation resulting in high distortion. This problem can be readily handled however by providing pre-computed information in the AdaptationQoS that maps possible parameter values for a specific BAE scheme to the rate and distortion achieved for a given image. The methodology below applies not only to JPEG images but to other content types as well, and is presented as such.

The content provider makes available the AdaptationQoS that declares and defines the following IOPins:

- Free IOPins: `PARAM1`, `PARAM2`, …, denoting the set of parameters to be used with a specific adaptation scheme.
- Dependent IOPins: `CODESIZE` (rate) and/or `MSE` (distortion) obtained as a function of the chosen set of parameters (`PARAM1`, `PARAM2`, …)

A UCD may then request minimization of `MSE` subject to `CODESIZE` ≤ *average transmission rate supported by network* times *maximum tolerable delay*. Alternatively, it can request minimization of `CODESIZE` subject to `MSE` ≤ *maximum acceptable distortion value*. The ADTE readily finds the appropriate set of parameters in either case by searching in the space of free IOPins: `PARAM1` `PARAM2`, … etc. and passing it to the BAE that implements the scheme.

A common BAE for JPEG is one that uses a single parameter (`PARAM1`) representing a quality factor (for instance as suggested by the Independent JPEG Group) to use for generating an 8x8 quantization matrix to requantize DCT coefficients. An improvement yielding better quality was recently reported [2]. The ADTE returns the right quality factor to use for either UCD type, by searching the discrete space of available quality factors and the corresponding pre-computed rate and distortion achieved by choosing them.

Interestingly, not knowing the actual rate achieved for a given set of parameter(s) is a universal problem in a wide range of reported rate-distortion optimal adaptation methods for various content-types that depend on a Lagrangian parameter $\lambda$. The AdaptationQoS in these cases can provide the actual rate achieved as a function of $\lambda$ to enable deciding the right parameter $\lambda$ based on a given rate constraint. In this case $\lambda$ would be a continuous free variable.

### B. JPEG2000 image adaptation

A JPEG2000 [15] image contains simultaneous spatial, SNR and component (color) scalability. It may also have precinct-based scalability, but in this example we focus only on the first three. Such a bit-stream can be represented in a 3-dim logical hypercube structure with multiple layers along each of the three scalability dimensions. Useful adaptations of the bit-stream truncate the logical hypercube at the outer ends.

The task of the ADTE is essentially to decide on the number of spatial, SNR and component layers to include based on available constraints. A BAE would use the decisions to actually drop the layers that are not required, and also to conduct other minor update operations on the bit-stream to guarantee syntax conformance.

The content provider makes available the AdaptationQoS metadata that defines and declares the following IOPins:

- Free IOPins: `NSPATIAL`, `NSNR`, `NCOMP` – indicating number of spatial, SNR and component layers respectively.
- Dependent IOPins: `CODESIZE` (rate), `MSE` (computed by reconstruction at the highest resolution for both grayscale and color), `IMAGE_WIDTH`, `IMAGE_HEIGHT` and `ISCOLOR` (whether image is color or grayscale), each as a function of the free IOPins.

Based on this metadata, it is possible to flexibly derive adapted versions based on various considerations represented in the UCD. We show two possible UCDs below that entertain very different considerations for adaptation of an image.

The first UCD requests maximization of `IMAGE_WIDTH` (which also maximizes `IMAGE_HEIGHT`) subject to the following limit constraints: `IMAGE_WIDTH` ≤ *display width* provided; `IMAGE_HEIGHT` ≤ *display height* provided; `ISCOLOR` cannot be true unless the viewing *display is color capable*; `CODESIZE` ≤ *average transmission rate supported by network* times *maximum tolerable delay* provided. Note the preference here is for a large image, even if that means sacrificing color for a color display or sacrificing quality.

The second UCD requests minimization of `MSE` subject to the following limit constraints: `IMAGE_WIDTH` ≥ *minimum desired display width*; `IMAGE_HEIGHT` ≥ *minimum desired display height*; `ISCOLOR` matches the *color capability of the display* exactly; and `CODESIZE` ≤ *transmission rate supported by network* times *maximum tolerable delay* provided. Note that here the preference is for a better image quality, as long as the size is greater than a minimum desired.

In both cases, the ADTE obtains appropriate solutions by searching the discrete space of free IOPins. But the AdaptationQoS description is agnostic of the considerations used to drive the adaptations. Fig. 6 shows an original image


(a) (b) (c)

Fig. 6. JPEG2000 image adaptation. (a) Original image, size 512x515. (b) Adapted with first UCD – max resolution 300x300, max codesize 6000 bytes, display grayscale. (c) Adapted with second UCD – min resolution 100x100, max codesize 6000 bytes, display color.

and two adapted versions generated by the above two UCDs with the same codesize constraint. The first adapted version in Fig. 6(b) is grayscale and is at half the resolution of the original. The second adapted version in Fig. 6(c) is color and has better quality, but is quarter the original resolution.

### C. Motion compensated predictive video adaptation

Here we discuss the rate adaptation of motion compensated predictive coded video like MPEG-X and H.26X. For such non-scalable bit-streams, a variety of rate-adaptation options exist with varying complexities, such as dynamic rate shaping [3], requantization [4], and object-based transcoding [6].

In [9] a utility-based rate adaptation framework was proposed, which was implemented by conducting frame dropping (FD) and coefficient dropping (CD) for adaptation of MPEG-4 video. FD adapts the source stream by skipping frames, while CD operates by truncating some high frequency DCT coefficients. The combination of FD and CD is attractive due to its simple implementation and flexible tradeoff between spatial and temporal qualities.

In order to enable prompt decision-taking subject to constraints, rate-distortion (R-D) information is collected per group of pictures (GOP) by computing sampled FD-CD operations, and transmitted as AdaptationQoS description to the ADTE. The ADTE uses this to obtain the optimal FD-CD decisions per GOP based on constraints, while the BAE uses the decisions to reshape the bit-stream by conducting FD-CD operations. User preferences can also be used in decision-taking to yield a valuable extension to traditional R-D optimization.

The content provider makes available the AdaptationQoS description that defines and declares the following IOPins:

- Adaptation unit IOPin: GOP.
- Free IOPins: `NPFRAME`, `NBFRAME`, `RCOEFF` – indicating respectively the number of P-frames to be dropped in a GOP, the number of B-frames to be dropped in a sub-GOP, and the sampled ratio of bit reduction by CD ∈ {0.0,0.1, …}, per adaptation unit which could be one or more GOPs.
- Dependent IOPins: `BANDWIDTH` (available bandwidth), `UTILIY` (quality of adapted video). In this example peak signal-to-noise ratio (PSNR) is adopted for Utility. Other quality measurements like mean opinion scale (`MOS`) may also be used.

The following are two typical kinds of optimizations in the FD-CD adaptation, which may be guided by the above AdaptationQoS metadata. First, for each adaptation unit, find optimal operation of FD-CD that maximizes `UTILITY` (`PSNR`), subject to `BANDWIDTH`≤*available bandwidth* (resource-constrained utility maximization). Second, for each adaptation unit, find optimal operation of FD-CD that minimizes `BANDWIDTH` (bit-rate), subject to `UTILITY` (`PSNR`) ≥ *minimum acceptable quality set by user* (e.g., 32 dB). Fig. 7 conceptually illustrates the first scenario. Given a target bit rate (780 *kbps* here), the available adaptation operations $a_i$ = (*FD*, *CD*) are calculated by ADTE and the optimal one that maximizes the selected utility is chosen and fed to BAE. Please note CD is a continuously adjustable parameter. Though only sampled data is provided, the exact CD ratio is obtained with adequate precision by mixed variable optimization, based on interpolated R-D data.

Adaptation of 1.5Mbps MPEG-4 video (CIF format with 30fps, GOP size=15, and sub-GOP size=3) down to about 200kbps under dynamic bandwidth constraints was demonstrated in [9]. There is little computational overhead for adaptation decision-taking. Real time FD-CD adaptation in BAE was shown to be feasible on moderate PC systems [10]. Even if the R-D information is unavailable (e.g., for live videos), a content-based utility function prediction approach [10] can be conducted so that the real time decision-taking can still be guaranteed.

### D. Fully scalable video adaptation

Fully scalable video bit-streams have already been introduced in Section III (see Fig. 2). The ADTE in this case, takes decisions on the number of temporal, spatial and SNR layers to include for each successive GOF (adaptation unit), based on current network and terminal constraints. For streaming sessions, the ADTE should be designed to take decisions for successive GOFs (adaptation units) synchronously with the transmission schedule, in order to accommodate dynamically varying network and usage conditions. Thus, the UCDs and UEDs that actually provide

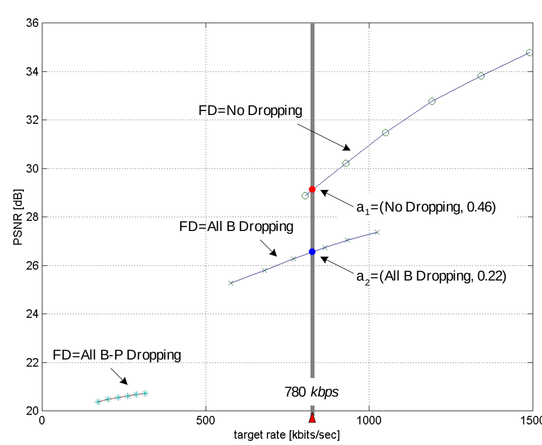


Fig. 7. The usage of FD-CD in video adaptation.

the constraints may change dynamically during a streaming session, causing the ADTE decisions for the currently processed and transmitted GOFs to also change accordingly.

The content provider makes available the AdaptationQoS description that defines and declares the following IOPins:

- Adaptation unit IOPin: `GOF`.
- Free IOPins: `NTEMP`, `NSPATIAL`, `NSNR` – indicating number of temporal, spatial, and SNR layers respectively.
- Dependent IOPins: `FRAMERATE` (temporal resolution), `BITRATE` (rate), `PQUAL` (an ad hoc perceptual GOF quality measure combining frame SNR computed at the highest resolution with framerate), `FRAMEWIDTH`, `FRAMEHEIGHT` – each as a function of free IOPins per GOF.

The following example UCD requests an adaptation. For the first GOF (adaptation unit), the UCD requests maximization of `PQUAL`, subject to: `FRAMEWIDTH` ≤ *display width* provided; `FRAMEHEIGHT` ≤ *display height* provided; FRAMERATE ≥ *a minimum desired value*; and `BITRATE` ≤ *average transmission rate supported by network*. For all subsequent GOFs, the `FRAMEWIDTH` and `FRAMEHEIGHT` limit constraints are replaced by one that requires them to remain the same as the previous adaptation unit. The ADTE obtains appropriate solutions for each adaptation unit sequentially by searching the discrete space of free IOPins. In line with provided constraints, it chooses the spatial resolution only for the first GOF, and maintains it the same for all subsequent GOFs. However, the temporal and SNR layers chosen keep changing depending on the video characteristics as provided in the AdaptationQoS and the current network conditions.

We demonstrate the adaptation performance based on this AdaptationQoS and UCD on 288 frames of the CIF *Foreman* sequence, compressed using the MC-EZBC [18] inter-frame scalable video codec. The bit-stream consists of 18 16-frame GOFs, each with 5 temporal, 6 spatial and 5 SNR layers. The parameters in the UCD generate a QCIF resolution adapted video for the first GOF, which is maintained for all subsequent GOFs. We consider two cases: first where the average available transmission rate for the network is 700 Kb/s for the duration of the transmission, and the second where the constraints are dynamically updated every one-third of the video so that for the same average rate of 700 Kb/s, the available rates for each individual one-third are 700 Kb/s, 350 Kb/s and 1050 Kb/s respectively. Table 1 presents for both cases the actual bandwidth transmitted along with the number of temporal, spatial and SNR layers transmitted for each GOF. Note that the actual transmission rate is not always very close to the available, since the example deals only with 5 SNR layers and uses a completely format independent BAE based on the BSD framework in MPEG-21 to drop only whole layers, without any bit-stream parsing. Adaptation of 288 frames on a moderate PC in both cases took less than 0.5 s, which is much less than that required for 30 frames/s transmission and playback.

## VI. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper we have presented the concepts behind the decision-taking framework supported in MPEG-21 (Part 7) DIA to enable Terminal and Network QoS. Casting the adaptation decision-taking problem as a generic constrained optimization problem involving adaptation variables enables creation of universal decision-taking engines that take decisions based on terminal, network and preference constraints, irrespective of their semantics and content type.

Decision-taking by search over discrete variables covers the vast majority of practical adaptation scenarios existing today – some of which are presented in the paper. Viable strategies when one or more variables are continuous are also presented. We hope the initial directions presented here would lead to more thorough future work and exploration of scenarios where continuous variable adaptation decisions would be relevant.

Table 1. Dynamic adaptation to match available bandwidth. T/S/Q stands for Temporal/Spatial/SNR(Quality) layers preserved by adaptation. All BWs are in Kb/s.

| GOF | Constant BW | | | Dynamic BW | | |
|---|---|---|---|---|---|---|
| | Av. BW | Actual BW | T/S/Q Layers | Av. BW | Actual BW | T/S/Q Layers |
| 0 | 700 | 600 | 5/5/2 | 700 | 600 | 5/5/2 |
| 1 | 700 | 671 | 4/5/3 | 700 | 671 | 4/5/3 |
| 2 | 700 | 536 | 5/5/2 | 700 | 536 | 5/5/2 |
| 3 | 700 | 544 | 5/5/2 | 700 | 544 | 5/5/2 |
| 4 | 700 | 542 | 5/5/2 | 700 | 542 | 5/5/2 |
| 5 | 700 | 670 | 5/5/2 | 700 | 670 | 5/5/2 |
| 6 | 700 | 657 | 5/5/3 | 350 | 332 | 4/5/2 |
| 7 | 700 | 679 | 3/5/4 | 350 | 273 | 5/5/1 |
| 8 | 700 | 633 | 5/5/2 | 350 | 321 | 4/5/1 |
| 9 | 700 | 669 | 5/5/2 | 350 | 317 | 4/5/1 |
| 10 | 700 | 579 | 5/5/2 | 350 | 290 | 4/5/1 |
| 11 | 700 | 651 | 5/5/2 | 350 | 308 | 4/5/1 |
| 12 | 700 | 687 | 4/5/3 | 1050 | 889 | 5/5/3 |
| 13 | 700 | 521 | 5/5/2 | 1050 | 870 | 5/5/3 |
| 14 | 700 | 607 | 4/5/3 | 1050 | 978 | 4/5/4 |
| 15 | 700 | 665 | 5/5/3 | 1050 | 876 | 4/5/4 |
| 16 | 700 | 587 | 5/5/3 | 1050 | 827 | 4/5/4 |
| 17 | 700 | 553 | 5/5/3 | 1050 | 1010 | 4/5/4 |

## REFERENCES


[1] O. Werner, “Requantization for transcoding of MPEG-2 frames,” *IEEE Trans. Image Processing*, vol. 8, no. 2, pp. 179-91, Feb 1999.

[2] H. H. Bauschke, C. H. Hamilton, M. S. Macklem, J. S. McMichael, and N. R. Swart, “Recompression of JPEG images by requantization”, *IEEE Trans. Image Proc.*, vol. 12, no. 7, pp. 843-49, July 2003.

[3] A. Eleftheriadis and D. Anastassiou, "Constrained and general dynamic rate shaping of compressed digital video", *Proc. 2nd IEEE Int. Conf. Image Proc.*, Washington, DC, vol. III, pp. 396-399, October 1995.

[4] P. Assuncao and M. Ghanbari, “Post-processing of MPEG2 coded video for transmission at lower bit rates,” in *Proc. IEEE Int. Conf. On Acoustic, Speech, and Signal Proc.’96*, Atlanta, GA, vol. 4, pp. 1998-2001, May 1996.

[5] P. Yin, A. Vetro, B. Liu, H. Sun, “Drift compensation for reduced spatial resolution transcoding,” *IEEE Trans. Circ. and Systems for Video Tech.*, vol. 12, no. 11, pp. 1009-20, Nov 2002.

[6] A. Vetro, H. Sun, and Y, Wang, “Object-based transcoding for adaptive video content delivery,” *IEEE Trans. Circ. and Systems for Video Tech.*, vol. 11, pp. 387-401, Mar. 2001.

[7] J. Chakareski, J. Apostolopoulos, B. Girod, "Low complexity rate-distortion optimized video streaming," *Proc. IEEE Int. Conf. Image Proc.*, Singapore, October 2004.
[8] Mihaela van der Schaar, Y. Andreopoulos, "Rate-Distortion-Complexity modeling for network and receiver aware adaptation," *IEEE Trans. Multimedia*, *Special Issue on MPEG-21*, in preparation.
[9] J.-G. Kim, Y. Wang, and S.-F. Chang, "Content-adaptive utility-based video adaptation," in *IEEE Int. Conf. on Multimedia and Expo*, Baltimore, Maryland, July 2003.
[10] Y. Wang, J.-G. Kim, and S.-F. Chang, "Content-based utility function prediction for real-time MPEG-4 video transcoding," in *Proc. IEEE Int. Conf. Image Proc.*, Barcelona, Spain, Sep. 2003.
[11] "ISO/IEC 21000-7 FDIS Part 7: Digital Item Adaptation," *ISO/IEC JTC 1/SC 29/WG 11/N6168*, Dec 2003.
[12] A. Vetro, C. Timmerer "Digital Item Adaptation: Overview of standardization and research activities", *IEEE Trans. Multimedia*, *Special Issue on MPEG-21*, this issue.
[13] "Information Technology – Multimedia Framework (MPEG-21) – Part 1: Vision, Technologies and Strategy," ISO/IEC 21000-1," *2002*.
[14] J. Bormans, J. Gelissen, A. Perkis, *MPEG-21: The 21st century multimedia framework*, *Signal Processing Magazine*, IEEE, Volume 20, Issue 2, pp. 53 – 62, March 2003.
[15] D.S. Taubman and M.W. Marcellin, "JPEG2000: Image Compression Fundamentals, Standards and Practice," *Kluwer Acad. Pubs, 2002*.
[16] *Special issue on Subband/Wavelet Interframe Video Coding, Signal Processing: Image Communication*, vol. 19, no. 7, Aug. 2004.
[17] J. –R. Ohm, M. van der Schaar, J. W. Woods, "Interframe wavelet coding – motion picture representation for universal scalability," *Signal Processing: Image Communication*, *Special Issue on Digital Cinema*, Sept. 2004.
[18] S. –T. Hsiang, J. W. Woods, "Embedded video coding using invertible motion compensated 3-D subband/wavelet filter bank," Signal Processing: Image Communication, vol. 16, pp. 705-24, May 2001.
[19] D. Mukherjee and A. Said, "Structured Scalable Meta-formats (SSM) for Digital Item Adaptation," *Proc. SPIE, Internet Imaging IV*, Santa Clara, California, vol. 5018, pp. 148-67, Jan 2003.
[20] D. Mukherjee, Peisong Chen, Shih-ta Hsiang, John Woods and A. Said, "Fully scalable video transmission using the SSM adaptation framework," *Proc. SPIE*, *Visual Communications and Image Processing*, Lugano, Switzerland, vol. 5150, July 2003.
[21] G. Panis, A. Hutter, J. Heuer, H. Hellwagner, H. Kosch, C. Timmerer, S. Devillers, M. Amielh, "Bitstream syntax description: A tool for multimedia resource adaptation within MPEG-21," *Signal Processing: Image Communication, Special Issue on Multimedia Adaptation*, vol. 18, pp. 721-747, Sep 2003.
[22] S. Devillers, C. Timmerer, J. Heuer, H. Hellwagner, "Bitstream Syntax Description based adaptation in streaming and constrained environments," *IEEE Trans. Multimedia*, *Special Issue on MPEG-21*, this issue.
[23] Ralph E. Steuer, *Multiple criteria optimization: theory, computation and application*, Krieger Publishing Company, 1996.
[24] S. Lerouge, P. Lambert, R. Van de Walle, "Multi-criteria Optimization for Scalable Bitstreams," *Visual Content Processing and Representation*, *8th International Workshop VLBV 2003*, *Lecture Notes in Computer Science*, vol. 2849, Sept. 2003.
[25] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, 2003.
[26] T. G. Kolda, R. M. Lewis, V. Torczon, "Optimization by direct search: New perspectives on some classical and modern methods," *SIAM Review*, vol. 45, pp. 385-482, Aug 2003.
[27] V. Torczon, "On the convergence of pattern search algorithms," *SIAM J. Optimization*, vol. 7, no. 1, pp. 1-25, Feb 1997.
[28] C. Audet, J. E. Dennis, "Analysis of generalized pattern searches," *SIAM J. Optimization*, vol. 13, no. 3, pp. 889-903, Feb 2003.
[29] C. Audet, J. E. Dennis, "Mesh adaptive direct search algorithms for constrained optimization," *Technical Report TR 04-02*, Department of Computational and Applied Mathematics, Rice University, Houston, TX 2004 – submitted to *SIAM J. Optimization*.
[30] C. Audet, J. E. Dennis, "Pattern search algorithms for mixed variable programming," *SIAM J. Optimization*, vol. 11, no. 3, pp. 573-94, Nov 2000.
[31] M. A. Abramson, "Pattern search algorithms for mixed variable general constrained optimization problems," *Ph.D. Thesis*, Dept. of Comput. and Applied Mathematics, Rice University, Houston, TX, 2002.

**Debargha Mukherjee** (M'99) was born in Calcutta, India. He received the B. Tech. Degree in E&ECE from the Indian Institute of Technology, Kharagpur, in 1993; and the M.S. and PhD. degrees in ECE from the University of California, Santa Barbara, in 1995 and 1999 respectively. He is currently a Senior Member of Research Staff, in the Imaging Technology Department of HP Laboratories, Palo Alto, California.

In 1998, he received the IEEE Student Paper Award at the IEEE International Conference on Image Processing, Chicago, Illinois. His research interests include multimedia compression and networking, signal processing, and information theory.

**Eric Delfosse** received the M.S. degree of Civil Electrotechnical Engineer in Telecommunications from the Free University of Brussels (VUB), Belgium, in 1999. From 1999 till 2001, he has been a Researcher at the VUB's Telecommunication Research Group. In 2001, he joined the Multimedia Research group of IMEC research center, Leuven, Belgium. His current research interests include wavelet-based image coding algorithms, 3D Graphics and Quality of Service.

Since December 2001, he is an active contributor to the ISO/IEC MPEG standardization committee, focusing on MPEG-4 Animated Framework eXtension (ISO/IEC 14496-16) and MPEG-21 Digital Item Adaptation (ISO/IEC 21000-7).

**Jae-Gon Kim** received the B.S. degree in electronics engineering from Kyungpook National University, Korea, in 1990, the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Korea, in 1992. Since 1992, he has been a Senior Member of Research Staff, in the Broadcasting Media Research Group of Electronics and Telecommunications Research Institute (ETRI), Korea. He is currently the Team Leader of the Broadcasting Content Research Team. From 2001 to 2002, he was a staff associate at Columbia University, New York. His research interests include multimedia content analysis, representation, compression and networking.

**Yong Wang** received his B.S. and M.S. degrees from Tsinghua University, Beijing, China in 1999 and 2001 respectively, both in electrical engineering. He is currently a Ph.D. candidate supervised by Professor Shih-Fu Chang in the Department of Electrical Engineering at Columbia University, New York, NY.

From 2001, he has been as a research assistant in the digital video and multimedia (DVMM) group at Columbia University. During the summer of 2003, he was an intern working at HP labs, Palo Alto, CA. From 1999 to 2001, he was a visiting student at Microsoft Research Asia, Beijing, China. His research interests include video coding, adaptation, communication and content-based analysis.