IEIE Transactions on Smart Processing and Computing

QoS-, Energy- and Cost-Efficient Resource Allocation for Cloud-Based Interactive TV Applications

Gosala Kulupana, Dumidu S. Talagala, Hemantha Kodikara Arachchi, and Anil Fernando

Centre for Vision, Speech and Signal Processing (CVSSP), Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, Surrey GU2 7XH, United Kingdom. Tel: +44 1483 686030, Fax: +44 1483 686031, {g.kulupana, d.talagala, h.kodikaraarachchi, w.fernando}@surrey.ac.uk

*****Corresponding Author: Gosala Kulupana, g.kulupana@surrey.ac.uk

**Regular Paper: None*

**Invited Paper: None*

**Review Paper: None*

**Extended from a Conference: Preliminary results of this paper were presented at the IEEE ICCE-Asia 2016. This present paper has been accepted by the editorial board through the regular reviewing process that confirms the original contribution.*

Abstract**:** Internet-based social and interactive video applications have become major constituents of the envisaged applications for next-generation multimedia networks. However, inherently dynamic network conditions, together with varying user expectations, pose many challenges for resource allocation mechanisms for such applications. Yet, in addition to addressing these challenges, service providers must also consider how to mitigate their operational costs (e.g., energy costs, equipment costs) while satisfying the end-user quality of service (QoS) expectations. This paper proposes a heuristic solution to the problem, where the energy incurred by the applications, and the monetary costs associated with the service infrastructure, are minimized while simultaneously maximizing the average end-user QoS. We evaluate the performance of the proposed solution in terms of serving probability, i.e., the likelihood of being able to allocate resources to groups of users, the computation time of the resource allocation process, and the adaptability and sensitivity to dynamic network conditions. The proposed method demonstrates improvements in serving probability of up to 27%, in comparison with greedy resource allocation schemes, and a several-orders-of-magnitude reduction in computation time, compared to the linear programming approach, which significantly reduces the service-interrupted user percentage when operating under variable network conditions.

Keywords**:** Social Multimedia Applications, Cloud Computing, Resource Optimization, Social Networking, QoS, Operational Cost

1. Introduction

The television (TV) is one of the most widely used multimedia consumer devices of all time. However, with the emergence of new communications and networking technologies, a paradigm shift is being experienced in traditional TV viewing. TV viewers who used to passively consume broadcast content can now interact with multimedia content and with like-minded viewers through the TV program itself. This concept, commonly known as interactive TV (ITV), has been rapidly evolving over the past decade, and has led to use cases where multiple personalized video multicasts are delivered to groups of users [1]. It is expected that the single-stream broadcasting concept will therefore be superseded by this style of personalized streaming, where groups of users (known as a social user group) can interact with each other through the ITV application, regardless of their geographic locations.

However, ITV users being geographically distributed leads to several challenges that must be overcome by a network's resource-allocation mechanisms when deploying these applications in practice. For example, the users' variable demand for resources and the strict transmission delay requirements being imposed (crucial to realizing true interactivity [2]) result in media processing (video compression, video merging, etc.) and distribution for such applications becoming quite cumbersome. Cloud computing when incorporated into the media processing chain as in [3]–[5], can address fluctuations in demand through the virtualization of resources (through improved scalability, reliability, and cost-effectiveness), but the increased physical separation between the service and the consumer introduces further challenges, such as latency [6], that must be overcome by the mechanisms that dynamically allocate the necessary network resources. In fact, the cloud concept itself is evolving toward inter-clouds, multi-clouds [7] and software-defined networking (SDN) concepts, which make geographically distributed cloud data centers viable candidates to underpin not only future ITV applications but many other distributed applications as well. Hence, the challenges and solutions for network resource allocation presented in this work for the ITV application scenario are both relevant and applicable to the broader field of network resource allocation in general.

To illustrate the concepts relevant to this particular resource allocation scheme, consider the following ITV application scenario, where it is essential that one maintains synchronization of the members of a group of users (i.e. the social user group) and guarantees an acceptable quality of service (QoS) for each individual. Here, media processing for a social user group is defined as occurring in a single processing cloud. Nevertheless, the ITV application provider must also consider how to maximize the revenue generated, through means such as minimizing the infrastructure energy utilization and equipment rental costs, and still provide good QoS to the end users. Those three facets are diverging requirements that must be balanced for optimal use of the available resources. To this end, this work proposes a heuristic resource allocation scheme to allocate computational and networking resources in a multi-group, distributed, interactive, dynamic multicast video transmission application scenario, such that not only is QoS maximized, but operational costs (energy consumption and monetary costs) are also minimized.

The remainder of this paper is organized as follows. First, a discussion of the state-of-the-art resource allocation schemes for cloud-based systems is presented in Section 2. This is followed by a formal definition of the problem and the optimization criteria in Section 3. The proposed methodology and the heuristic algorithmic solution are described in Section 4. The simulation environment is described in Section 5, the performance of the proposed methods is evaluated and compared with existing resource allocation methods in Section 6, followed by concluding remarks in Section 7.

2. Related Work

The optimal allocation of cloud computing resources to competing tasks has been extensively studied in the literature. For example, Yuan et al. [8] suggested that the routing protocols should reflect both server and network energy consumption while achieving energy efficiency in cloud-based multimedia services. To this end, Beloglazov and Buyya [9] proposed a three-step energy-saving initiative for virtualized cloud data centers in order to maintain QoS. In the first phase, virtual machines (VMs) that can be migrated are identified based on CPU usage thresholds. In the second and third phases, those VMs are filtered based on the network traffic load and the energy requirements of the associated servers. In contrast, a heuristic workload consolidation approach to energy minimization was presented [10] wherein the users' QoS was modeled in terms of the length of the task queue in the core-network switches. Similarly, Wang et al. modeled VM placement within the cloud data centers as an energy-minimization problem [11], which was solved using particle swarm optimization. The common theme in these techniques is that they only considered the server attributes, whereas the users' QoS and energy consumption in the access network devices (e.g., switches, routers, connecting links) were implicitly disregarded. Although access layer QoS has recently been modeled during the data center allocation process [12], [13], these methods neither considered the energy efficiency nor selected a suitable optimization scheme for energy management.

In the context of a distributed multimedia application, it is essential to simultaneously optimize both the application and the network layer resources. Goiri et al. formulated a resource optimization problem for delay-bound cost minimization [14] that achieves this objective. There, the authors leveraged a mixed integer linear programming (MILP) solver and a heuristic method to solve the energy minimization problem. It, however, considered allocation of a domain of users to a single processing node, disregarding the optimal network route selection, and is therefore fundamentally different from the current application scenario where multiple groups of users may be allocated to multiple processing nodes.

The Tabu search–based algorithm [15] describes data centers optimally allocating to processes by considering routing optimization between the backbone routers, access nodes, and processing nodes. In this approach, the tasks are assigned to processing nodes such that QoS is maximized while satisfying a set of processing constraints. Subsequently, a combined Tabu search–based heuristic and MILP solver was incorporated to solve the constrained optimization problem [15]. However, several crucial differences exist, in comparison with the proposed scenario. First, the optimization scheme of Larumbe and Sanso [15], as with most other methods in the literature, does not consider the multicasting nature of the interactive application scenario—a critical factor in the energy minimization problem [8]. Secondly, the formation of coherent user groups, which is an important aspect of future interactive applications, was not considered. Lastly, the Larumbe and Sanso decision time is too great to facilitate the interactive behavior of the application [15]. In order to overcome the shortcomings outlined above, a heuristics-based application and network layer resource allocation scheme is proposed in the following sections.

3. The Proposed Scheme

3.1 System Description

An example of the application scenario described in this work is illustrated in Fig. 1. Here, four users belonging to two user groups are connected to two Internet service providers (ISPs). The remaining network is comprised of four cloud data centers and three routing nodes. During the course of this study, we assumed that the following conditions are valid, in general, with respect to the ITV distribution system.

(A.1). All nodes in the network support multicasting.
(A.2). Users may join, withdraw, or migrate from a social user group, and may create new user groups.
(A.3). A single processing node serves each user group.
(A.4). Processing nodes may act as routing nodes and can participate in the media distribution process.

Let $G\ (V,\ E)$ be the ITV system, which consists of $V=\{S,A,R\}$ (the set of nodes), including $S=\{s_1,\ s_2,\dots,\ s_S\}$ (the set of processing nodes, i.e., clouds), $A=\{a_1,a_2,\dots,\ a_A\}$ (the set of access nodes, i.e., ISPs), and $R=\{r_1,\ r_2,\dots,\ r_R\}$ (the set of routing nodes in the network). Let E be the set of interconnections (i.e., edges) between all the nodes, and let $U=\{u_1,\ u_2,\dots,\ u_U\}$ be the set of users who are connected to the ITV system and who belong to the set of social user groups $N=\{n_1,\ n_2,\dots,\ n_N\}$. User u of social group $n \in N$ who is connected to access node $a \in A$ can be represented as,

$$u \mapsto u_n^a = \begin{cases} 1 & \text{if } u \text{ is a member of social group } n \text{ and} \\ & \text{is connected to access node } a, \\ 0 & \text{otherwise} \end{cases}$$

The main decision variables relevant to this resource allocation problem can be defined as follows:

$$x_{i,j}^{n,a} = \begin{cases} 1 & \text{if the edge from node } i \text{ to } j \text{ is used in the} \\ & \text{multicast tree for users in group } n \text{ who} \\ & \text{reside in the access node } a, \\ 0 & \text{otherwise} \end{cases}$$

$$y_{i,j}^{n} = \begin{cases} 1 & \text{if the edge from node } i \text{ to } j \text{ is used in the} \\ & \text{multicast tree for social user group } n, \\ 0 & \text{otherwise} \end{cases}$$

$$z^{n,s} = \begin{cases} 1 & \text{if processing node } s \text{ processes the media} \\ & \text{content of social user group } n, \\ 0 & \text{otherwise} \end{cases}$$

The decision variable $x_{i,j}^{n,a}$ corresponds to the use of logical link (i,j) with respect to access node a, whereas $y_{i,j}^{n}$ corresponds to the usage of physical link (i,j). The required and available resources are denoted in Table 1.

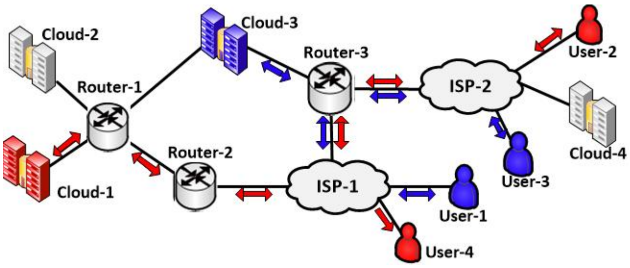


Figure 1. An example network architecture diagram of an ITV distribution system. Four users in two social user groups connect to two ISPs where Cloud-1 and Cloud-3 act as the media processing nodes.

Table 1. Parameters and Notations

Parameter	Notation
Transmission bandwidth of the interactive media multicast required by social user group n	B_n
Available bandwidth from node i to node	$B_{i,j}$
Available bandwidth of edge $e \in E$	b_e
Processing capability required by user group n	P_n
Available processing power at processing node s	p_s
Average link delay from node i to node j	$D_{i,j}$
Maximum delay tolerated by the ITV application	Δ
Maximum allowable delay from the processing node to the a^{th} access node for the n^{th} user group (i.e., $\Delta_{n,a} = \Delta$ - max $(D_{a,u})$, for all u in group n connected to a).	$\Delta_{n,a}$

The various constraints imposed by the ITV distribution system are summarized below. Thus, for social user group n, processing node s, routing node r, and access node a,

(C.1). $$\sum_{j \in E_s^O} x_{s,j}^{n,a} - \sum_{j \in E_s^I} x_{i,s}^{n,a} = z^{n,s}$$

$E_s^O \Rightarrow \{$ outgoing edge nodes from s $\}$and
$E_s^I \Rightarrow \{$ incoming edge nodes to s $\}$

(C.2). $$\sum_{j \in E_a^I} x_{i,a}^{n,a} = 1\ ,$$

$E_a^I \Rightarrow \{$incoming edge nodes to a $\}$

(C.3). $$\sum_{j \in E_r^O} x_{r,j}^{n,a} - \sum_{i \in E_r^I} x_{i,r}^{n,a} = 0$$

$E_r^O \Rightarrow \{$ outgoing edge nodes from $r\}$and
$E_r^I \Rightarrow \{$ incoming edge nodes to $r\}$

(C.4). $$\sum_{n \in N} y_{i,j}^{n} \times B_n \le B_{i,j}$$

(C.5). $$\sum_{n \in N} z^{n,s} \times P_n \le p_s$$

(C.6). $$\sum_{(i,j) \in E} x_{i,j}^{n,a} \times D_{i,j} \le \Delta_{n,a}$$

(C.7). $$\sum_{s \in S} z^{n,s} = 1$$

(C.8). $$\sum_{i \in E_j^I} y_{i,j}^{n} \le 1\ ,$$

$E_j^I \Rightarrow \{$ incoming edge nodes to $j\}$

(C.9). $$\forall a \in A\ \ y_{i,j}^{n} \ge x_{i,j}^{n,a}$$

The behavior of the processing clouds is described by (C.1). Here, if the n^{th} social user group's content is processed by server s ($z^{n,s}$=1), a single outgoing edge from server s is activated to carry the traffic for social user group n. If s functions as a relay node, it accepts the media from a single edge in the set of incoming edges (E_s^I) and will pass it to another edge in the set of outgoing edges (E_s^O). Similarly, in order to satisfy the multicasting requirements, (C.2) ensures that an access node receives media over a single incoming edge in the set of incoming edges E_a^I . The relaying function defined for the set of routing nodes r in (C.3) is similar to that in (C.1). Constraint (C.4) ensures that the physical bandwidth of each edge is sufficient to carry the media streams traversing that particular edge. Similarly, (C.5) specifies that processing node s has sufficient processing capacity to process all social user groups allocated to it. In order to realize true interactivity, the end-to-end link delay of each user should be kept below predefined threshold Δ. This is captured in (C.6). Constraint (C.7) ensures that single processing node s processes the n^{th} social user group, thereby eliminating any synchronization issues that may arise when multiple users are engaging with the same media content. Constraint (C.8) ensures that only one physical link exists between node pair (i,j). (C.9) describes how multiple logical links can map to a single physical link, through which the constraints defined for the physical quantities in (C.4) and (C.8) become effective.

3.2 Multi-Objective Cost Function

In this subsection, we explain the formulation of the multi-objective cost function, which consists of three components: group QoS cost (cumulative QoS costs of all user groups), energy cost (cumulative energy consumption of all user groups) and monetary cost (cumulative monetary costs of serving all user groups) in the ITV system. Subsections 3.2.1 to 3.2.3 describe the scenarios where the individual cost functions are minimized, whereas Subsection 3.2.4 illustrates the proposed multi-objective cost function formulation.

3.2.1 Group QoS Cost

Each user's QoS can be modeled as the sum of end-to-end link QoS parameters from the processing node to the user. Here, we adopt a similar approach to Kim and Choi who proposed a QoS cost metric for Internet protocol TV (IPTV) systems [16], and extend this to maintain an acceptable quality of experience (QoE) in the ITV system by imposition of a delay bound. The link QoS cost metric for the ITV application can therefore be modeled (assuming media are transmitted at an approximately fixed rate) as

$$Qc_{i,j}^{L} = \alpha_1 \times (L_{i,j}) + \alpha_2 \times (J_{i,j}) + \alpha_3 \times (D_{i,j}) \qquad (1)$$

where $Qc_{i,j}^{L}$ refers to the QoS cost of the link from node i to node j, $L_{i,j}$ refers to the average packet loss rate along the link from i to j, and $J_{i,j}$ refers to the jitter in the path; $\{\alpha_1, \alpha_2, \alpha_3\}$ are appropriately selected constants that adequately parameterize the QoS cost metric for an IPTV scenario [16].

The n^{th} group's QoS cost from processing node s to access node a, weighted by the number of users in the group, can now be expressed as

$$\phi_{1,s}^{n,a} = \left(\sum_{(i,j)\in E} Qc_{i,j}^{L} \times x_{i,j}^{n,a} \right) \times \sum_{u\in U} u_n^a \qquad (2)$$

while the QoS cost from the access nodes to the users can be expressed as

$$\phi_2^{n,a} = \left(\sum_{u\in U} Qc_u \times u_n^a \right) \qquad (3)$$

where Qc_u denotes the QoS cost from user $u \vdash u_n^a$ to his/her access node. We define the n^{th} group's QoS cost as the summation of (2) and (3), normalized by the number of users in the group, as

$$\phi_s^n = \frac{1}{\left(\sum_{a\in A} \sum_{u\in U} u_n^a\right)} \sum_{a\in A} \{\phi_{1,s}^{n,a} + \phi_2^{n,a}\} \qquad (4)$$

3.2.2 Energy Consumption Cost

The total energy consumption in the ITV system can be modeled as the sum of energy consumption in the processing clouds and the routing devices throughout the network. The energy dissipation in the routing nodes is modeled on earlier work [17], [18]. Hence, the n^{th} social group's energy cost can be expressed as

$$\psi_s^n = \left(\sum_{(i,j)\in E} Ec_{i,j}^n \times y_{i,j}^n \right) + \left(\sum_{u\in U} Ec_u \times u_n^a \right) + Ec_s^n \qquad (5)$$

where Ec_s^n is defined as the sum of the incremental energy consumption of node j and link (i,j) due to the transmission of packets of the n^{th} social user group. Ec_u denotes the energy cost from user $u \mapsto u_n^a$ to his access node, and Ec_s^n denotes the incremental energy cost due to the processing that takes place at processing node s.

3.2.3 Monetary Cost

The monetary cost of the resources allocated to a user group can be expressed as

$$\xi_s^n = \left(\sum_{(i,j)\in E} Mc_{i,j}^n \times y_{i,j}^n \right) + \left(\sum_{u\in U} Mc_u \times u_n^a \right) + Mc_s^n \qquad (6)$$

where $Mc_{i,j}^n$ denotes the incremental rental cost for utilizing the path from i to j for user group n, Mc_u denotes the monetary cost from user $u \vdash u_n^a$ to her/his access node, and Mc_s^n is the incremental cost for renting processing capacity for user group n at processing node s.

3.2.4 Multi-Objective Cost Function

Subsections 3.2.1 to 3.2.3 above describe the optimization of a single cost function without evaluating the impact on the remaining costs. The inherent disadvantage of this approach is that optimization of a single cost function may give rise to under-optimized values for the remaining cost terms. Therefore, in this work, simultaneous optimization of multiple cost terms is proposed. The multi-objective link cost between node *i* and node *j* therefore becomes

$$Oc_{i,j}^{n} = \beta_1 \times Qc_{i,j}^{n} + \beta_2 \times Ec_{i,j}^{n} + \beta_3 \times Mc_{i,j}^{n} \quad (7)$$

where $\{\beta_1, \beta_2, \beta_3\}$ are appropriately selected constants that determine the priorities among the objectives. In this work, we selected an equal contribution from each objective towards the final cost function, and experimentally determined $\{\beta_1=0.36, \beta_2=0.15, \text{ and } \beta_3=0.60\}$. Similarly

$$Oc_{u} = \beta_1 \times Qc_{u} + \beta_2 \times Ec_{u} + \beta_3 \times Mc_{u} \quad (8)$$

$$Oc_{s}^{n} = \beta_2 \times Ec_{s}^{n} + \beta_3 \times Mc_{s}^{n} \quad (9)$$

Therefore, the overall cost function becomes

$$\tau_s^n = \left(\sum_{(i,j)\in E} Oc_{i,j}^{n} \times y_{i,j}^{n} \right) + \left(\sum_{u\in U} Oc_{u} \times u_{n}^{a} \right) + Oc_{s}^{n} \quad (10)$$

Finally, minimizing the multi-objective cost implies the following:

$$\text{minimize} \left(\sum_{n\in N} \tau_s^n \right)$$

4. Solution Methodologies

Three main approaches exist for solving the optimization problem described in the previous section: linear programming methods, greedy resource allocation methods, and heuristic methods.

4.1 Mixed Integer Linear Programming

The MILP approach can be devised to minimize equation (2), subject to constraints (C.1) to (C.9). Of several solvers that are capable of handling optimization problems with binary variables, we used the MOSEK [19] and YALMIP [20] toolboxes available in MATLAB. The obvious disadvantages associated with the MILP method are high memory utilization and significant execution times. We incorporated the MILP method to evaluate the potential to reach the optimal solution.

4.2 Greedy Resource Allocation

During greedy resource allocation, the best available resources are sequentially assigned to each social user group. Furthermore, once a user group is assigned with resources, those resources are immovable. As a result, unfair preference is given to initial users and user groups; thus, a resource scarcity occurs for later groups, and it ultimately becomes impossible to serve them by any means. This behavior is the main drawback associated with this technique. We select the greedy approach as a comparison method due to its being adopted by many existing techniques [10], [21], and for its inherent simplicity.

4.2 Proposed Heuristic Method

The proposed method is a three-step process aimed at overcoming the aforementioned undesirable characteristics of the MILP and the greedy resource allocation methods, as illustrated below.

Step 1: Determine the delay bound and minimal cost path from each pair of potential processing nodes and access nodes. The method proposed by Salama et al. [22] is applied, where the appropriate link cost $\{ Oc_{i,j}^{n} = \beta_1 \times Qc_{i,j}^{n} + \beta_2 \times Ec_{i,j}^{n} + \beta_3 \times Mc_{i,j}^{n} \}$ is considered during the tree construction process.

Step 2: Derive a set of multicast trees rooted at a particular processing node for each user group. Algorithm 1 is proposed to address this step.

Step 3: Optimally co-locate each user group's multicast tree, such that edge bandwidth and cloud processing limitations are satisfied. The algorithm presented by Kulupana et al. [23] is incorporated to address this step.

Algorithm 1 constructs the multicast tree in Step 2 for the multi-objective cost function defined in (7). For each user group *n*, multicast trees are constructed for every possible cloud *s* over a filtered network where edge bandwidths exceed the transmission requirements of the user group. Multicast trees in Algorithm 1 are constructed using the unicast trees in Step 1 and a combination of Dijkstra's algorithm and the Kompella algorithm [24].

In Algorithm 1, MULTICAST_REDUCTION_COST is called to ensure that intermediate nodes perform only multicasting. For a common intermediate node, *r*, the function evaluates the various unicast paths from *s* to *r*. If these are identical, they form part of the multicast tree from *s* to *r*. In the event that they are not identical, then the minimum delay path from *s* to *r* is selected. The cost is computed for multiple multicast trees (i.e., link cost only) before the addition of the processing node costs. The overall cost is then used to determine the minimum cost multicast tree and the processing node location. The time complexity of Algorithm 1 is in the order of $(|S||A|^3|V|^3)$.

Algorithm 1: Multicast tree generation for QoS, energy and monetary cost minimization.

procedure MULTICAST_TREE_ENERGY (G, $\mathcal{U}$, { $Oc^{n}_{i,j}$ },{ Oc^{n}_{s} }, n ,*initial_cloud, dynamic_flag*)

for $s \in S$ (where $p_s > P_n$)

if (dynamic_flag= true **and** s ≠ *initial_cloud*)

$\tau^{n}_{s} \leftarrow c_{\max}$ % Force current cloud to remain unchanged

end if

Initialize *mult_tree(n,s)* ⟵{s};

for *iteration* 1 to $|A|_0$

for $a \in \mathcal{A}$

if (a has users from social group n**;** $a \notin$ *mult_tree*)

Find the delay bound minimum cost path connecting node a to any node in the *mult_tree* [21]

min_cost$_a$ ⟵ Assign minimum cost.

if(*min_cost$_a$* < *min_cost*)

min_cost ⟵ *min_cost$_a$*

min_cost_node ⟵ *a*

uni_tree(n,s,a) ⟵ Store computed path.

end if

end if

end for

i_nodes ⟵ Find common intermediate nodes in *uni_tree* and *mult_tree*.

if (*i_nodes* do not exist)

mult_tree(n,s) ⟵ Assign *uni_tree* to multicast tree.

else

mult_tree(n,s) ⟵ MULTICAST_REDUCTION_COST

end if

$Oc^{n}_{s,a} \leftarrow$ Calculate cost of *mult_tree*.

end for

Calculate the overall cost, including processing cost.

$\tau^{n}_{s} \leftarrow \tau^{n}_{s} + Oc^{n}_{s,a} + Oc^{n}_{s}$

end for

$\upsilon_n \leftarrow \min(\tau^{n}_{s})$

mult_tree ⟵ *mult_tree*(*n*,*s$_0$*); *s*≡*s$_0$* corresponds to the minimum cost processing node.

return *mult_tree*, υ_n

end procedure

procedure MULTICAST_REDUCTION_COST(*uni_tree*, *i_nodes*, *mult_tree*, *min_cost_node*, *n*)

for $r \in i\_nodes$

if (the s to r path in *uni_tree*(*n,s,min_cost_node*) and *mult_tree* are common)

mult_tree(*n,s*) ⟵ Assign as multicast path from *s* to *r*.

else

path_delays ⟵ Calculate delay along path from *s* to *r* in *uni_tree*(*n,s,min_cost_node*) and *mult_tree.*

if (*path_delay*(*mult_tree*) < *path_delay*(*uni_tree*))

No change in multicast path from *s* to *r*.

mult_tree ⟵ Add path from *r* to *a* from *uni_tree*.

else

mult_tree ⟵ Assign *uni_tree* path from *s* to *a*.

end if

end if

end for

return *mult_tree*

end procedure

5. Simulation Environment

The performance of the proposed heuristic resource allocation technique was evaluated in MATLAB using 200 Monte Carlo simulations of different network conditions. The simulations were carried out on a 32-core Dell PowerEdge R710 server (eight Intel Xeon Quad-Core E5520 2.2 GHz processors) with 144 GB memory. The resource requirements are prescribed by the interactive, personalized video distribution application described in Sec. I. For simplicity, we assume high-definition H.264 transmissions to each user group with a bandwidth of 8 Mbps [16], where the personalization (inclusion of interactive elements) of the stream requires, on average, 50,000 MIPS. In order to maintain an acceptable QoE, the maximum allowable interaction delay is restricted to 100 ms [2], and $\Delta_{n,a}$ is derived accordingly. The network is assumed to be made up of 10 ISPs (access nodes), 10 cloud computing resources (processing nodes), and 10 routing nodes (backbone routers). In order to evaluate the effectiveness of the proposed algorithms, the interconnections, load, and cost of these resources are varied as follows.

The number of interconnections between nodes of the network was determined at random; however, the available bandwidth, link latency, jitter, and packet loss were restricted to within 20 Mbps < $B_{i,j}$ < 60 Mbps, 20 ms < $D_{i,j}$ < 60 ms, 5 ms < $J_{i,j}$ < 60 ms, and 0.01% < $L_{i,j}$ < 0.1%, respectively [16]. The link latency between each user and the ISP (access node) is a random variable in the interval (10 ms, 20 ms). The networking cost for 8 Mbps transmission was determined by a random variable between \$100 and \$200 [25], while the incremental processing cost of creating the personalized content for a user group was a random value between \$300 and \$600 (mainly due to the content dependency of the processing operation and the differences in rental costs for cloud-based resources). The incremental networking energy usage by the routing nodes was assumed to be a function of incremental bandwidth [17]. This is denoted by a random variable between 1 W and 20 W. Energy consumption by the cloud computing resources is a function of the number of servers utilized for the processing task and the processing capability of each server. This was assumed to be a random quantity in the 5 kW to 10 kW interval, and was derived from work by Buyya et al. [26] for an application requiring 50,000 MIPS.

6. Performance Evaluation

The performance of the proposed heuristic algorithm is discussed here. Table 2 summarizes the minimum achievable cost for the multi-objective cost function, and the execution time for four optimization techniques: the MILP method, the proposed heuristic method, and two greedy resource allocation schemes. In the "greedy-networking" approach, greedy multicast trees are first created for each user group. However, during this tree construction phase, the unicast trees created for initial users within a group are not altered, even though a later user's delay requirements may be violated (and is therefore greedy). In such instances, later users need to be served through alternate routes. After constructing multicast trees for each user group, they are co-located sequentially, based on the available networking and processing resources. In the greedy-processing approach, the minimum-cost processing cloud is selected for each social user group, irrespective of the users' QoS requirements, and is followed by greedy multicast tree construction.

Table 2. Comparison of Average Cost and Average Execution Time for Different Social User Groups

	User Groups	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Average Cost	MILP	0.63	1.26	1.78	2.49	3.39	3.70	4.14	4.76	5.13	-	-	-	-	-	-
	Proposed	0.66	1.30	1.90	2.63	3.58	3.88	4.35	4.88	5.32	6.49	7.03	7.63	8.26	8.88	9.30
	Greedy-N	0.66	1.30	1.90	2.66	3.60	3.92	4.42	4.97	5.37	6.62	7.21	7.81	8.50	9.09	9.60
	Greedy-P [10,21]	0.66	1.30	1.91	2.72	3.60	3.92	inf	inf	inf	inf	inf	inf	inf	inf	inf
Average Time (s)	MILP	5	36	81	1277	4485	4738	7183	21324	46026	-	-	-	-	-	-
	Proposed	3	6	10	16	25	31	43	53	60	82	99	112	129	137	164
	Greedy-N	3	6	9	12	15	17	21	24	26	29	33	35	38	42	44
	Greedy-P [10,21]	0.4	0.7	1	1	2	2	inf	inf	inf	inf	inf	inf	inf	inf	inf

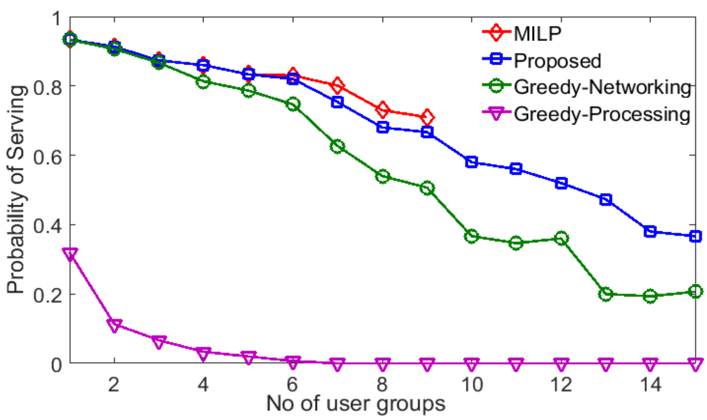


Figure 2. Serving probability of all social user groups in the ITV distribution system.

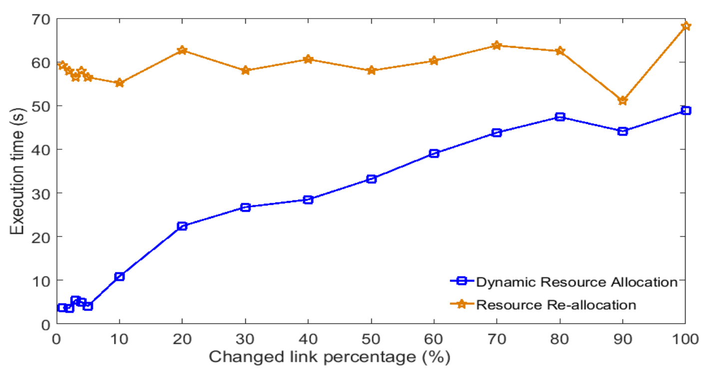
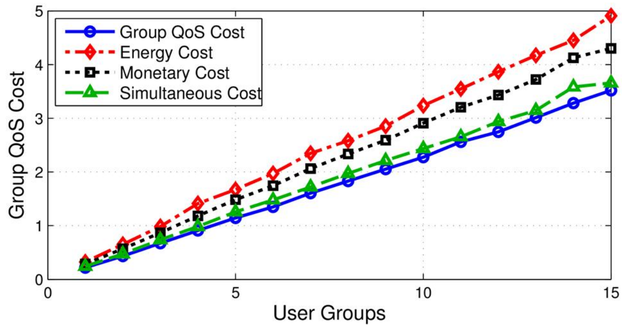


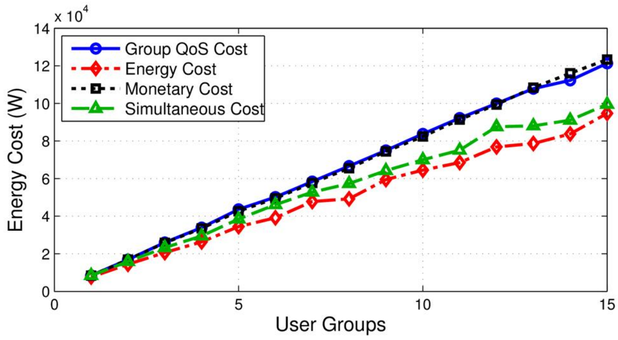
Figure 3. Execution time of the proposed resource allocation scheme in a dynamic networking scenario for a fixed user configuration for eight user groups.

Table 3. Cost and Unaffected User Percentage Comparison for Dynamic and Complete Re-allocation of Resources.

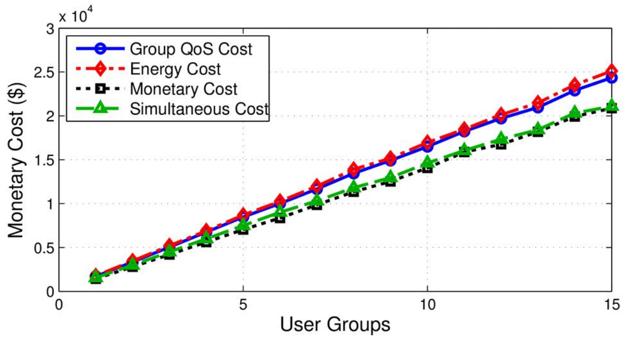
Changed link percentage (%)		1	2	3	4	5	10	20	30	40	50	60	70	80	90	100
Average cost	Dynamic allocation	6.20	6.32	6.23	6.21	6.32	6.11	5.90	5.93	5.85	5.77	5.69	5.69	5.53	5.67	5.58
	Full reallocation	5.20	5.21	5.18	5.18	5.21	5.19	5.07	5.23	5.09	5.17	5.09	5.14	5.05	5.14	5.05
Unaffected users (%)	Dynamic allocation	94.3	94.8	93.4	94.7	94.0	89.6	85.0	81.0	78.1	77.3	71.9	70.7	68.0	68.0	68.5
	Full reallocation	92.7	84.5	78.1	79.7	78.1	64.6	53.9	47.6	33.9	32.3	35.1	33.3	33.0	26.7	29.6



(a) Group QoS cost



(b) Energy cost



(c) Monetary cost

Figure 4. (a) Group QoS cost, (b) energy cost, and (c) monetary cost obtained from the proposed (heuristic) method. Four cost functions are considered for the scenarios corresponding to group QoS cost minimization, energy cost minimization, monetary cost minimization and simultaneous QoS, energy, and monetary cost minimization.

As illustrated in Table 2, the MILP approach is limited to the first nine user groups, due to the increasing memory and execution time required for larger numbers of user groups. In the meantime, the greedy-processing approach has not been able to find a feasible solution after the sixth user group. The MILP method, through its near exhaustive search, finds the least-cost solution. However, the proposed method outperformed the greedy techniques and only exhibits a marginal performance loss, compared to the MILP method. As further illustrated in Table 2, the execution times of the MILP approach are several orders of magnitude greater than the proposed method, making it undesirable for interactive applications. The execution time of the greedy-processing approach is much lower than the remaining methods since, unlike other methods, only one processing cloud is evaluated during the cost minimization used in this approach.

Figure 2 illustrates the serving probability (i.e., the ability to find a feasible solution) of each method. The serving probability decreases as the number of social user groups increases due to the competition for resources among the user groups. Yet, it can be seen that the proposed method significantly outperforms the greedy methods, resulting in up to a 27% performance gain, while demonstrating similar overall cost-efficiency performance with respect to the MILP method. The impact of neglecting the network parameters during the processing cloud selection can also be clearly identified from the large performance gap between the greedy-processing method and the remaining methods.

The ability of the proposed method to adapt to varying network conditions is demonstrated in Figure 3. Both illustrated methods utilize the proposed multicasting approach. However, during dynamic allocation, a higher preference is given to the initial processing cloud. Thereby, alteration of the processing cloud of an already active social user group is made less likely. As illustrated in Table 3, the effectiveness of dynamic allocation is evaluated by allowing network parameters to change with certain probabilities. The resource allocation through dynamic allocation resulted in fewer service interruptions (a user is assumed to be affected when his/her serving cloud changes), albeit at a marginal increase in the cost. Furthermore, as depicted in Table 3, full resource re-allocation consumes a substantial amount of computing time, compared to dynamic allocation, and further justifies the selection of a dynamic resource–allocation mechanism.

The performance gain with the proposed method over the state-of-the-art methods was discussed in the previous paragraphs. The effectiveness of the proposed multi-objective cost optimization is described in this paragraph with the aid of the individual cost optimization results. As expected, Figs. 4(a)-(c) show that the minimum of either type of cost is achieved via individual minimization of the respective cost functions. Therefore, energy and monetary cost minimization, for example, would result in higher group QoS costs, in comparison with the group QoS cost–minimization scenario seen in Fig. 4(a). The simultaneous optimization of the composite cost function, however, is expected to reveal a lower group QoS cost than the energy- and monetary-cost–minimization scenarios, and a greater group QoS cost than the group QoS-cost–minimization scenario, as illustrated. A similar pattern is expected and demonstrated in the energy cost and monetary cost results illustrated in Figs. 4(b) and (c). An interesting observation revealed in Figs. 4(a)-(c) is the relatively smaller differences exhibited between the simultaneous composite cost–function minimization scenario and the actual minimum that can be achieved using the appropriate "proposed (heuristic)" approach. Although intuition suggests that this may be caused by an unequal weighting of the cost functions, the lack of results being significantly skewed (toward either of the individual cost function–minimization results) in any of the three subfigures discounts this possibility, and leads us to conclude that the simultaneous composite cost–function minimization method is well suited to joint minimization of all three types of costs. Furthermore, the results imply that a marginal reduction of a particular cost requires progressively greater sacrifices in the other two types of costs, and suggests that the simultaneous minimization of the composite cost function may be more beneficial from an application perspective.

7. Conclusion

In this paper, we propose an efficient scheme to allocate computational and network resources in a next-generation, distributed, interactive, multicast video transmission application. First, the assumptions and constraints applicable to this application are described, and the three costs of relevance to the system, namely group QoS cost, energy cost, and monetary cost, are defined as a single multi-objective cost function. Thereafter, a heuristic solution to this resource allocation problem is proposed (in terms of this cost function), which consists of three steps: end-to-end delay bound least-cost unicast tree generation, cost-minimized multicast tree generation, and dynamic multiple multicast tree co-location. Multiple Monte Carlo trials of different network and user configurations were simulated to evaluate the proposed method's performance, and were compared with results obtained from the optimal mixed integer linear programming and the state-of-the-art greedy resource allocation approaches. The simulation results suggest that the proposed method can achieve comparable performance to the MILP approach, with a several-orders-of-magnitude reduction in the computational time required. In addition, improvement in the ability to find a feasible resource allocation configuration of up to 27% is observed with respect to the greedy approaches. Next, the robustness of the proposed method under dynamic network conditions is illustrated. Finally, the possibility of achieving multi-objective cost optimization comparable to the individual cost minimization scenarios is demonstrated.

Acknowledgement

This work was supported by the EU FP7 project ACTION-TV (http://www.action-tv.net), funded under the European Commission's 7th Framework Program (Grant Number: 611761).

References

[1] "User Interaction Aware Content Generation and Distribution For Next Generation Social Television." [Online]. Available: web (CrossRef Link).

[2] "Response Times: The 3 Important Limits." [Online]. Available: web (CrossRef Link).

[3] SeungGwan Lee, et al., "Personalized DTV program recommendation system under a cloud computing environment," *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 1034–1042, May 2010. Article (CrossRef Link)

[4] G. M. Calixto, A. C. B. Angeluci, L. C. P. Costa, R. de Deus Lopes, and M. K. Zuffo, "Cloud computing applied to the development of global hybrid services and applications for interactive TV," *in Proc. of IEEE Int. Symp. Consumer Electronics*, pp. 283–284, June 2013. Article (CrossRef Link)

[5] M. Doke, H. Kaneko, N. Hamaguchi, and S. Inoue, "Engaging Viewers Through the Connected Studio: Virtual Participation in TV Programs," *IEEE Consum. Electron. Mag.*, vol. 1, no. 4, pp. 30–39, Oct. 2012. Article (CrossRef Link)

[6] T. Hoßfeld, R. Schatz, M. Varela, and C. Timmerer, "Challenges of QoE management for cloud applications," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 28–36, April 2012. Article (CrossRef Link)

[7] K. Hwang, G. Fox, and J. J. Dongarra, "Inter-cloud Resource Management," in *Distributed and Cloud Computing*, San Francisco: Morgan Kauffmann Publishers, p. 246, Dec. 2011.

[8] H. Yuan, C. Kuo, and I. Ahmad, "Energy efficiency in data centers and cloud-based multimedia services: An overview and future directions," *in Proc. of Int. Conf. on Green Computing*, pp. 375–382, Aug. 2010. Article (CrossRef Link)

[9] A. Beloglazov and R. Buyya, "Energy efficient resource management in virtualized cloud data centers," *in Proc. of IEEE/ACM Int. Conf. on CCGrid*, pp. 826–831, May 2010. Article (CrossRef Link)

[10] D. Kliazovich, P. Bouvry, and S. Khan, "DENS: data center energy-efficient network-aware scheduling," *Cluster Comput.*, vol. 16, no. 1, pp. 65–75, Sept. 2011. Article (CrossRef Link)

[11] S. Wang, Z. Liu, and Z. Zheng, "Particle Swarm Optimization for Energy-Aware Virtual Machine Placement Optimization in Virtualized Data Centers," *in Proc. of Int. Conf. on Parallel and Distributed systems*, pp. 102–109, Dec. 2013. Article (CrossRef Link)

[12] G. Kulupana, D. S. Talagala, H. K. Arachchi, and A. Fernando, "Optimized resource distribution for interactive TV applications," *in Proc. of IEEE Int. Conf. on Consumer Electronics*, pp. 70–71, Jan. 2015. Article (CrossRef Link)

[13] G. Kulupana, D. S. Talagala, H. Kodikara Arachchi, and A. Fernando, "Resource allocation for cloud-based social TV applications using Particle Swarm Optimization," *in Proc. of IEEE Int. Conf. on Communications*, pp. 1226-1231, June 2015. Article (CrossRef Link)

[14] Í. Goiri, et al., "Intelligent placement of datacenters for internet services," *in Proc. Int. Conf. on Distributed Computing Syst.*, pp. 131–142, June 2011. Article (CrossRef Link)

[15] F. Larumbe and B. Sanso, "A Tabu Search Algorithm for the Location of Data Centers and Software Components in Green Cloud Computing Networks," *IEEE Trans. Cloud Comput.*, vol. 1, no. 1, pp. 22–35, Nov. 2013. Article (CrossRef Link)

[16] H. Kim and S. Choi, "A study on a QoS/QoE correlation model for QoE evaluation on IPTV service," *in Int. Conf. on Advanced Commun. Technology*, pp. 1377–1382, Feb. 2010. Article (CrossRef Link)

[17] "Measurement Based Network Element Power Modeling." [Online]. Available: web (CrossRef Link).

[18] G. Kulupana, D. S. Talagala, H. K. Arachchi, and A. Fernando, "Energy efficient resource allocation for cloud-based interactive TV applications," *in Proc. of IEEE Int. Conf. on Consumer Electronics-Asia*, Oct. 2016. Article (CrossRef Link)

[19] "The MOSEK optimization toolbox for MATLAB." [Online]. Available: web (CrossRef Link).

[20] J. Lofberg, "YALMIP : a toolbox for modeling and optimization in MATLAB," *in IEEE Int. Conf. on Robotics and Automation*, pp. 284–289, Sept. 2004. Article (CrossRef Link)

[21] D. Kusic, J. Kephart, and J. Hanson, "Power and performance management of virtualized computing environments via lookahead control," *Cluster Comput.*, vol. 12, no. 1, pp. 1–15, Oct. 2008. Article (CrossRef Link)

[22] H. F. Salama, D. S. Reeves, and Y. Viniotis, "A distributed algorithm for delay-constrained unicast routing," *in IEEE INFOCOM*, pp. 84–91, April 1997. Article (CrossRef Link)

[23] G. Kulupana, D. S. Talagala, H. K. Arachchi, and A. Fernando, "Optimized resource distribution for interactive TV applications," *IEEE Trans. Consum. Electron.*, vol. 61, no. 3, pp. 344–352, Aug. 2015. Article (CrossRef Link)

[24] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, “Multicast routing for multimedia communication,” *IEEE/ACM Trans. Netw.*, vol. 1, no. 3, pp. 286–292, 1993. Article (CrossRef Link)

[25] “Cloud vs. WAN Costs: A Breakdown.” [Online]. Available: web (CrossRef Link)

[26] R. Buyya, A. Beloglazov, and J. Abawajy, “Energy-efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges,” *arXiv Prepr. arXiv1006.0308*, July 2010. Article (CrossRef Link)

Gosala Kulupana received his BSc (Eng) with honors from University of Moratuwa, Sri Lanka, in 2011. From 2011 to 2014, he was an Engineer at Mobitel (pvt) Ltd., Sri Lanka. He is currently pursuing his PhD in the Centre for Vision, Speech and Signal Processing at the University of Surrey, United Kingdom. His research interests are in the areas of resource optimization, LTE, 5G, and error-resilient video coding.

Dumidu S. Talagala received a BSc Eng (Hons) in Electronic and Telecommunication Engineering from the University of Moratuwa, Sri Lanka, in 2007. From 2007 to 2009, he was an Engineer at Dialog Axiata PLC, Sri Lanka. He completed his PhD within the Applied Signal Processing Group, College of Engineering and Computer Science, at the Australian National University, Canberra, in 2013. He is currently a research fellow in the Centre for Vision, Speech and Signal Processing at the University of Surrey, United Kingdom. His research interests are in the areas of sound source localization, spatial sound field reproduction, and convex optimization techniques.

Hemantha Kodikara Arachchi received his BSc (Eng) degree with honors and his M.Phil. in Electronic and Telecommunication Engineering from University of Moratuwa, Sri Lanka, in 1997 and 2000, respectively, and obtained a PhD in Telecommunications from the Asian Institute of Technology (AIT) in 2004. At present, he is a Senior Research Fellow at the CVSSP Multimedia Communications Research Group at the University of Surrey, UK. His research interests are in video coding, video communication, QoE, and context-aware content adaptation. He has published over 70 peer-reviewed journal and conference papers.

Anil Fernando received a BSc (Eng) (First Class) in Electronic and Telecommunications Engineering from the University of Moratuwa, Sri Lanka, in 1995, and an MEng (Distinction) in Telecommunications from the Asian Institute of Technology (AIT), Bangkok, Thailand, in 1997. He completed his PhD in the Department of Electrical and Electronic Engineering, University of Bristol, UK, in February 2001. Currently, he is a reader in signal processing at the University of Surrey, UK. Prior to that, he was a senior lecturer at Brunel University, UK, and an assistant professor at AIT. His current research interests include cloud communications, quality of experience, intelligent video encoding for wireless systems, and video communications in LTE. He has published more than 290 papers internationally in these areas. He is a senior member of IEEE and a fellow of the HEA, UK.