Computer Networks 77 (2015) 144–152

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

# A coalitional game approach for fractional cooperative caching in content-oriented networks

Phuong Luu Vo [a,b], Duc Ngoc Minh Dang [b], Sungwon Lee [b], Choong Seon Hong [b,*], Quan Le-Trung [c]

[a] Department of Computer Science and Engineering, International University – VNUHCM, Viet Nam
[b] Department of Computer Engineering, Kyung Hee University, Republic of Korea
[c] Faculty of Computer Network and Communication, University of Information Technology – VNUHCM, Viet Nam

ARTICLE INFO

*Article history:*
Received 27 August 2013
Received in revised form 17 November 2014
Accepted 12 December 2014
Available online 26 December 2014

*Keywords:*
Cooperative caching
Coalitional game
Linear programming

ABSTRACT

In content-oriented networks, popular contents are replicated at the intermediate nodes to enhance content delivery performance. Under cooperative caching, the caching nodes collaborate to leverage one another's cache capability and to reduce the amount of traffic transferring inside the network. This study considers the cooperation among service providers (SPs). The transferable-payoff coalitional game model is applied for analysis. We investigate the stability of the grand coalition and show that the dual-based cost allocation is in the core. A linear program (LP) minimizing the network bandwidth-expense is used for the characteristic function of the game model. However, solving the LP is a challenge because of a large amount of contents in the network. The Dantzig–Wolfe decomposition approach is further applied to decompose the large-scale problem into many subproblems, which can be solved in parallel. The analysis provides not only a deeper insight into the cooperative cache among SPs but also content placement and distribution strategies as a solution to the LP.



## 1. Introduction

In content-oriented networks, the contents are replicated at the caching nodes placed in the service provider's (SP) networks. The users obtain the contents at the intermediate nodes instead of downloading from the original sources, thus decreasing the time and cost of content delivery to the end users. Ideally, all contents should be cached in each storage node. However, a large amount of contents exists in the network whereas storage capacity is limited. Therefore, the nodes only cache the most popular contents. When a node receives a request for a particular content item, the content is directly sent back to the end user if the node has the content in its storage; otherwise, the content is fetched from the other nodes in the network or from the content repository. It has been known that the nodes should cooperate in caching to reduce the bandwidth cost of the traffic transferred within the network. Given the demands of the contents, the optimal content placement is proposed by solving a large-scale mixed integer linear program (LP) minimizing the total bandwidth cost of the network [1–4]. In case the integer constraints are relaxed, the problem models the fractional cooperative caching, which is considered in this work.

Cooperative caching at the SP level aims to mitigate the transit traffic cost among SPs [5–8]. The SPs in Ref. [5] cooperate in a selfish or altruistic manner to serve their subscribers. The algorithms of both cases are proven to converge to Nash equilibriums. The authors in Ref. [6]

* Corresponding author.
*E-mail addresses:* phuongvo@hcmiu.edu.vn (P.L. Vo), dnmduc@khu.ac.kr (D.N.M. Dang), drsungwon@khu.ac.kr (S. Lee), cshong@khu.ac.kr (C.S. Hong), quanlt@uit.edu.vn (Q. Le-Trung).

http://dx.doi.org/10.1016/j.comnet.2014.12.005
1389-1286/

use the Vickrey–Clarke–Groves auction game model to design a mechanism for the cache servers owned by selfish wireless SPs truthfully cooperating with each other. Also using cooperative caching, the authors in Ref. [7] propose an optimization model to minimize the inter-SP traffic while considering the inter-SP traffic pattern, cache server resource allocation, and SP peering agreements. The work in Ref. [8] addresses the question of whether content-level peering is stable and efficient without explicit coordination between the neighboring cache networks. The cache allocation of autonomous systems at content-level peering is stable regardless of whether the autonomous systems coordinate [8].

Meanwhile, the application of the coalitional game model in communication networks has recently received increasing attention [9–16]. Ref. [9] provides a concise introduction and survey on this kind of game. In a coalitional game, the players cooperate into groups called *coalitions* to reduce their operational cost. A coalition structure called *coalition partition* is stable if a cost allocation scheme ensures that no player/group of players would want to leave their current coalition; otherwise, their cost will increase. An optimal partition is the stable partition having a minimal aggregate cost. In general, a coalition formation algorithm is required to form a coalition partition. The problem of finding an optimal partition is a hard problem and the optimal algorithm has exponential complexity. Therefore, a sub-optimal algorithm converging to a Nash equilibrium point is usually proposed. The readers can refer to Refs. [10–12] for more details on this kind of game. In another kind of coalitional game, the player always benefits from cooperating [13–16]. The grand coalition is stable under some cost allocations.

This study considers the SP-level cooperation in caching in light of coalitional game. We explore the main questions of whether the cooperation always benefits and how the operational cost is reasonably shared among the SPs. The transferable-payoff coalitional game model is applied to the formulation. The SPs tend to form coalitions to reduce their operational cost. We show that the core of the game is non-empty. The dual-based cost allocation stabilizes the grand coalition. Therefore, the SPs always benefit when joining the grand coalition. Three cost allocation schemes, namely, Shapley, nucleolus, and dual-based, are also discussed in this study.

The characteristic function used in the coalitional game model is an LP problem minimizing the network bandwidth-expense. Solving the LP also yields the optimal content placement and distribution strategies. However, the cooperative caching LP is a large-scale optimization problem because of the large number of content items in the network. Utilizing the block-angular structure of the large-scale problem, we decompose it into many subproblems which can be solved in parallel. Each subproblem is associated with one content. To the best of our knowledge, this study is the first to consider collaborative caching among the SPs with the use of a coalitional game model.

The structure of this paper is as follows. Section 2 describes the game model, proves the non-emptiness of the core, and presents some typical cost sharing schemes, *i.e.*, dual-based, Shapley, and nucleolus. Section 3 discusses the decomposition approach that addresses the large-scale LP. Section 4 details the numerical results and Sections 5 concludes the work.

## 2. Bandwidth-expense coalitional game

### 2.1. Transferable-payoff coalitional game model

Let us consider a set of SPs $\mathcal{N}$ providing content delivery service to their users. The nodes in an SP network is assumed to cooperate in caching to serve the requests to the SP. When a node $i$ receives a request for content $m$, node $i$ continues to forward content $m$'s request to all the other nodes it cooperates if it has only $x_i^m$ fraction of content $m$ already in its storage. Consequently, node $i$ receives $y_{ij}^m$ fraction of content $m$ from node $j$ and $z_i^m$ fraction from content $m$'s repository (see Fig. 1). The content would usually be downloaded from the caching nodes rather than from the repository. The cost of transferring one unit of content $m$ from its repository to serve the request at node $i$ is higher than that from the other node $j$, *i.e.*, $c_i^m > c_{ij}$. We assume that the demands for the contents are constants in the considered time period (Table 1 shows the main notations used in this paper).

**Table 1**
Notations.

| | |
|---|---|
| *Parameters:* | |
| $\mathcal{M}$ | Set of contents |
| $\mathcal{S}$ | Coalition/subset of service providers |
| $\mathcal{N}$ | Grand coalition, set of all the service providers |
| $\boldsymbol{n}(\mathcal{S})$ | Set of all caching nodes in coalition $\mathcal{S}$ |
| $B_i$ | Storage capacity of caching node $i$ |
| $s^m$ | Size of content $m$ |
| $d_i^m$ | Demand for content $m$ at node $i$ in the considered time period |
| $c_{ij}$ | Cost for one unit of content from node $j$ to node $i$ in the considered time period |
| $c_i^m$ | Cost for one unit of content $m$ from its original source to node $i$ in the considered time period |
| *Variables:* | |
| $x_i^m$ | Fraction of content $m$ stored on node $i$ |
| $y_{ij}^m$ | Fraction of content $m$ from node $j$ serving the demand on node $i$ |
| $z_i^m$ | Fraction of content $m$ from original source serving the demand on node $i$ |

The coalitional game in characteristic form with transferable payoff is applied to model the cooperation among SPs [9,17]. Each SP is a *player* in the game. A *coalition* is a subset of SPs ($\mathcal{S} \subseteq \mathcal{N}$) collaborating in caching the contents, *i.e.*, one node can receive/fetch the content from/to the other nodes within the coalition to serve its requests. A coalition could be an individual SP or all SPs. In the second case, it is called *grand coalition*. The coalitional game is denoted by a pair $(\mathcal{N}, v)$, where $v$ is the characteristic function that maps any coalition $\mathcal{S} \subseteq \mathcal{N}$ to a real value quantifying the cost of $\mathcal{S}$, *i.e.*, $v(\mathcal{S})$.

Given coalition $\mathcal{S}$, the amount of cost that SP $k$ in $\mathcal{S}$ receives from the division of $v(\mathcal{S})$ constitutes SP $k$'s payoff. A *cost allocation* with respect to coalition $\mathcal{S}$ is the set of payoffs/shared costs of all the SPs in $\mathcal{S}$.

The characteristic function is described using a bandwidth-expense minimization problem as follows. In any coalition $\mathcal{S}$, the amount of bandwidth cost for serving the demand for $m$ at node $i$ is the total of amount of bandwidth cost to transfer the fractions of content $m$ from the other caching nodes and the repository, *i.e.*, $s^m d_i^m \left( \sum_{j \in \mathbf{n}(\mathcal{S}), j \neq i} c_{ij} y_{ij}^m + c_i^m z_i^m \right)$. Hence, the total amount of the bandwidth cost of coalition $\mathcal{S}$ to serve its demands is given by $\sum_{m \in \mathcal{M}} \sum_{i \in \mathbf{n}(\mathcal{S})} s^m d_i^m \left( \sum_{j \in \mathbf{n}(\mathcal{S}), j \neq i} c_{ij} y_{ij}^m + c_i^m z_i^m \right)$. We need to choose a content placement and fetching strategies minimizing the bandwidth expense of coalition $\mathcal{S}$:

$$\mathbf{Primal}(\mathcal{S}): \text{Min.} \sum_{m \in \mathcal{M}} \sum_{i \in \mathbf{n}(\mathcal{S})} s^m d_i^m \left( \sum_{\substack{j \in \mathbf{n}(\mathcal{S}) \\ j \neq i}} c_{ij} y_{ij}^m + c_i^m z_i^m \right)$$

$$\text{st.} \sum_{m \in \mathcal{M}} s^m x_i^m \leqslant B_i, \forall i \in \mathbf{n}(\mathcal{S}), \quad (1)$$

$$y_{ij}^m \leqslant x_j^m, \quad \forall i, j \in \mathbf{n}(\mathcal{S}), i \neq j, m \in \mathcal{M}, \quad (2)$$

$$x_i^m + \sum_{\substack{j \in \mathbf{n}(\mathcal{S}) \\ j \neq i}} y_{ij}^m + z_i^m \geqslant 1, \forall i \in \mathbf{n}(\mathcal{S}), m \in \mathcal{M}, \quad (3)$$

$$x_i^m \geqslant 0, y_{ij}^m \geqslant 0, z_i^m \geqslant 0, \quad \forall i, j \in \mathbf{n}(\mathcal{S}), i \neq j, m \in \mathcal{M}. \quad (4)$$

Constraint (1) is the storage capacity constraint of each node, *i.e.*, total storage of all contents/fraction of contents at node $i$ must be less than or equal to $i$'s storage capacity $B_i$. Constraint (2) implies that the fraction of content $m$ from node $j$ serving the requests to node $i$ must not exceed the amount of content $m$ stored at node $j$. Constraint (3) guarantees that the amount of content $m$ stored at node $i$ and the total amount of content $m$ received from the other nodes and the repository must be greater that or equal to the whole content to serve the requests for content $m$ at node $i$. The equality of constraint (3) actually holds at the optimal point. Hence, constraints $x_i^m \leqslant 1$, $y_{ij}^m \leqslant 1$, and $z_i^m \leqslant 1$ are implicitly implied.

The *characteristic function* with respect to coalition $\mathcal{S}$ is defined as the minimal bandwidth cost when the SPs in $\mathcal{S}$ cooperate. Therefore, $v(\mathcal{S})$ is the solution to **Primal**$(\mathcal{S})$ problem.

Notably, **Primal**$(\mathcal{S})$ can be utilized to formulate the cooperative caching among SPs in a larger scale. At a higher level, each SP includes several clusters, each of which can contain tens to thousands of nodes. Each node in the above formulation can be considered a cluster. Accordingly, $x_i^m$ is the fraction of content $m$ stored on cluster $i$, $y_{ij}^m$ is the fraction of content $m$ fetched to cluster $i$ from cluster $j$, and $z_i^m$ is the fraction of content $m$ fetched from the original source to cluster $i$.

### 2.2. On the core of the coalitional game

Let $\mathbf{p} = (p_k)_{k \in \mathcal{N}}$ be the *cost allocation* with respect to grand coalition $\mathcal{N}$. We have $\sum_{k \in \mathcal{N}} p_k = v(\mathcal{N})$, which is the solution to **Primal**$(\mathcal{N})$ problem. With cost allocation $\mathbf{p}$, the total cost of any subset $\mathcal{S}$ in the grand coalition is $p(\mathcal{S}) = \sum_{k \in \mathcal{S}} p_k$. This value is different from that of $v(\mathcal{S})$ yielded by coalition $\mathcal{S}$ (*i.e.*, only the SPs in $\mathcal{S}$ cooperate). The core of the game is defined as follows.

**Definition 1.** The *core* of the coalitional game is the set of cost allocations $\mathbf{p}$ such that $p(\mathcal{S}) \leqslant v(\mathcal{S})$, $\forall \mathcal{S} \subseteq \mathcal{N}$.

If the core is non-empty, then there exists a cost allocation that the shared cost of any subset $\mathcal{S}$ is less than the cost of coalition $\mathcal{S}$, *i.e.*, $v(\mathcal{S})$. No SP or group of SPs has the incentive to deviate from the grand coalition since its cost will increase if it leaves. The grand coalition is called *stable* under allocation $\mathbf{p}$.

Theoretically, a cost-based coalitional game is convex if the characteristic function is submodular, *i.e.*, $v(\mathcal{S}) + v(\mathcal{T}) \geqslant v(\mathcal{S} \cup \mathcal{T}) + v(\mathcal{S} \cap \mathcal{T})$ for any $\mathcal{S}, \mathcal{T} \subseteq \mathcal{N}$. It is known that the core of a convex game is non-empty. However, the bandwidth-expense coalitional game considered in this paper does not have convexity (see Example 1). Nevertheless, the non-emptiness of the core is always obtained according to Theorem 1.

**Example 1.** Consider three SPs $\{1, 2, 3\}$. Each SP has one caching node with 2 GB of the storage capacity. Assume that the network has two contents, and the demands for the contents follows Zipf–Mandelbrot distribution, which is usually used to model content popularities [20]. The demand for $m$-th popularity content at SP $i$ is given by

$$d_i^m = r_i \frac{(q+m)^{-a}}{\sum_{m \in \mathcal{M}} (q+m)^{-a}}, \quad (5)$$

where $q$ is the shift parameter, $a$ is the shape parameter of the distribution, and $r_i$ is the request rate to SP $i$. Assume that $q = 10$, $a = 1$, and $r_i = 1$ request per minute. The costs between the two SPs are $c_{12} = c_{13} = c_{23} = 2$. The costs from repository to nodes are $c_1 = c_2 = c_3 = 4$. Let $\mathcal{S} = \{1, 2\}$ and $\mathcal{T} = \{2, 3\}$. Subsequently, $v(\mathcal{S}) + v(\mathcal{T}) = 8$, whereas $v(\mathcal{S} \cup \mathcal{T}) + v(\mathcal{S} \cap \mathcal{T}) = 9.74 > v(\mathcal{S}) + v(\mathcal{T})$.

**Theorem 1.** *The core of the bandwidth-expense coalitional game* $(\mathcal{N}, v)$ *is non-empty.*

**Proof.** We prove Theorem 1 by showing that the dual-based cost allocation is in the core. The proof also includes the construction of the cost allocation for SPs with respect to the grand coalition. The dual problem of **Primal**$(\mathcal{S})$ is given by

$$\textbf{Dual}(\mathcal{S}) : \text{Max.} \sum_{i \in \boldsymbol{n}(\mathcal{S})} \left( \sum_{m \in \mathcal{M}} \gamma_i^m - B_i \alpha_i \right)$$

$$\text{st.} s^m \alpha_i - \sum_{j \in \boldsymbol{n}(\mathcal{S}), j \neq i} \beta_{ji}^m \geqslant \gamma_i^m, \forall i \in \boldsymbol{n}(\mathcal{S}), m \in \mathcal{M} \quad (6)$$

$$s^m d_i^m c_{ij} + \beta_{ij}^m \geqslant \gamma_i^m, \forall i, j \in \boldsymbol{n}(\mathcal{S}), i \neq j, m \in \mathcal{M} \quad (7)$$

$$s^m d_i^m c_i^m \geqslant \gamma_i^m, \forall i \in \boldsymbol{n}(\mathcal{S}), m \in \mathcal{M}, \quad (8)$$

$$\alpha_i \geqslant 0, \beta_{ij}^m \geqslant 0, \gamma_i^m \geqslant 0, \forall i, j \in \boldsymbol{n}(\mathcal{S}), i \neq j, m \in \mathcal{M}, \quad (9)$$

where $(\alpha_i)_{i \in \boldsymbol{n}(\mathcal{S})}, (\beta_{i,j}^m)_{i,j \in \boldsymbol{n}(\mathcal{S}),\ j \neq i,\ m \in \mathcal{M}}$, and $(\gamma_i^m)_{i \in \boldsymbol{n}(\mathcal{S}),\ m \in \mathcal{M}}$ are Lagrange multipliers associated with constraints (1)–(3), respectively. From the strong duality, the primal solution also coordinates with the dual solution. Let $(\alpha_i^*, \beta_{ij}^{m*}, \gamma_i^{m*})$ for all $i, j \in \boldsymbol{n}(\mathcal{N}),\ j \neq i,\ m \in \mathcal{M}$ be the dual optimal values with respect to the grand coalition, *i.e.*, $\mathcal{S}$ is replaced by $\mathcal{N}$ in **Primal**($\mathcal{S}$) and **Dual**($\mathcal{S}$) problems. The shared cost for SP $k$ with respect to the grand coalition is defined as follows:

$$p_k = \sum_{k \in n(\{k\})} \left( \sum_{m \in \mathcal{M}} \gamma_i^{m*} - B_i \alpha_i^* \right). \quad (10)$$

Subsequently, $\sum_{k \in \mathcal{N}} p_k = v(\mathcal{N})$.

We prove the above dual-based cost allocation scheme is in the core by showing $p(\mathcal{S}) \leqslant v(\mathcal{S})$ for any subset $\mathcal{S} \subseteq \mathcal{N}$. The following inequalities hold:

1. $s^m \alpha_i^* - \sum_{j \in \boldsymbol{n}(\mathcal{S}),\ j \neq i} \beta_{ji}^{m*} \geqslant s^m \alpha_i^* - \sum_{j \in \boldsymbol{n}(\mathcal{N}),\ j \neq i} \beta_{ji}^{m*} \geqslant \gamma_i^{m*}$ for all $i \in \boldsymbol{n}(\mathcal{S}), m \in \mathcal{M}$;
2. $s^m d_i^m c_{ij} + \beta_{ij}^{m*} \geqslant \gamma_i^{m*}$ for all $i, j \in \boldsymbol{n}(\mathcal{S}), i \neq j, m \in \mathcal{M}$;
3. $s^m d_i^m c_i^m \geqslant \gamma_i^{m*}$, for all $i \in \boldsymbol{n}(\mathcal{S}), m \in \mathcal{M}$; and
4. $\alpha_i^* \geqslant 0, \beta_{ij}^{m*} \geqslant 0, \gamma_i^{m*} \geqslant 0$, for all $i, j \in \boldsymbol{n}(\mathcal{S}), i \neq j, m \in \mathcal{M}$.

Thus, $(\alpha_i^*, \beta_{ij}^{m*},\ \gamma_i^{m*})$ for all $i, j \in \boldsymbol{n}(\mathcal{S}),\ j \neq i,\ m \in \mathcal{M}$ is a feasible point of **Dual**($\mathcal{S}$). Since $v(\mathcal{S})$ is the optimal value of **Primal**($\mathcal{S}$) as well as **Dual**($\mathcal{S}$) as the definition of characteristic function of the game,

$$p(\mathcal{S}) = \sum_{k \in \mathcal{S}} p_k = \sum_{k \in \mathcal{S}} \sum_{i \in n(\{k\})} \left( \sum_{m \in \mathcal{M}} \gamma_i^{m*} - B_i \alpha_i^* \right)$$
$$= \sum_{i \in \boldsymbol{n}(\mathcal{S})} \left( \sum_{m \in \mathcal{M}} \gamma_i^{m*} - B_i \alpha_i^* \right) \leqslant v(\mathcal{S}).$$

Therefore, the core of the bandwidth-expense coalitional game is non-empty. □

### 2.3. Shapley and nucleolus values

Aside from the dual-based cost allocation, two other well-known cost allocations for a transferable-payoff coalitional game, *i.e.*, Shapley and nucleolus values, are discussed in this section as follows.

(1) *Shapley*: Shapley values emphasize on the fairness of cost allocation. Denote the marginal contribution of SP $k$ to coalition $\mathcal{S}$ be the amount of decrease in cost when SP $k$ joins coalition $\mathcal{S}$, *i.e.*, $\Delta_k(\mathcal{S}) = v(\mathcal{S} \cup \{k\}) - v(\mathcal{S})$. The Shapley value of SP $k$ equals to

$$\phi_k = \sum_{\mathcal{S} \subseteq \mathcal{N} \setminus \{k\}} \frac{|\mathcal{S}|!(|\mathcal{N}| - |\mathcal{S}| - 1)!}{|\mathcal{N}|!} \Delta_k(\mathcal{S}). \quad (11)$$

The Shapley value is indeed the average of SP $k$'s marginal contributions over all possible orderings. The number of permutations of coalition $\mathcal{S}$ without SP $k$ is $\frac{|\mathcal{N}|!}{(|\mathcal{N}| - |\mathcal{S}| - 1)!|\mathcal{S}|!}$. Therefore, the average value over all coalitions is given by Eq. (11). The Shapley value is known as the unique cost allocation that satisfies symmetry, dummy, and additivity axioms [17]. However, it may not be in the core of the coalitional game (see Example 2).

(2) *Nucleolus*: The nucleolus cost allocation aims to minimize dissatisfaction of the SPs on their shared cost in $(\mathcal{N}, v)$ game. Let $\boldsymbol{p} = (p_k)_{k \in \mathcal{N}}$ be the cost allocation for the grand coalition. Denote $e(\boldsymbol{p}, \mathcal{S}) = p(\mathcal{S}) - v(\mathcal{S})$ be the excess of $\mathcal{S}$ with respect to $\boldsymbol{p}$. This value measures the "unhappiness degree" of coalition $\mathcal{S}$. If there exists a coalition with a positive excess, the cost allocation $\boldsymbol{p}$ cannot be in the core. Let $\boldsymbol{e}(\boldsymbol{p}) \in \mathbb{R}^{2^{|\mathcal{N}|}}$ be the excess vector with respect to allocation $\boldsymbol{p}$, *i.e.*, the list of excesses of all $2^{|\mathcal{N}|}$ coalitions. A vector $\boldsymbol{e}(p)$ is *lexicographically less than* a vector $\boldsymbol{e}(q)$ if there is an index $k$ that $\boldsymbol{e}_i(p) = \boldsymbol{e}_i(q)$ for all $i < k$ and $\boldsymbol{e}_k(p) < \boldsymbol{e}_k(q)$. The *nucleolus* of a coalitional game is a cost allocation that lexicographically minimizes the maximal excess. It is known that the nucleolus is unique and in the core of a coalitional game [17]. It satisfies symmetry and dummy axioms. Unlike the Shapley solution, the nucleolus value does not have a closed-form. Calculating the nucleolus requires solving a sequence of LPs [21].

The complexity of the calculation of the Shapley and nucleolus solutions exponentially grows in the number of SPs. Both require to solve many LPs: $2^{|\mathcal{N}|-1}$ LPs for the Shapley solution and $O(4^{|\mathcal{N}|})$ LPs for the nucleolus solution. Nevertheless, the dual-based cost allocation which stabilizes the grand coalition needs to solve only one LP, that is, **Primal**($\mathcal{N}$) or **Dual**($\mathcal{N}$).

**Example 2.** Consider a network with three SPs $\{1, 2, 3\}$. The numbers of nodes in the SPs are 1, 4, and 5 nodes, respectively. Each node has 2 GB of storage, and the number of contents is 5. The request rate to each SP equals to one request per minute and is uniformly distributed among the caching nodes within each SP. Table 2 presents the cost allocations according to three solution concepts: Shapley, nucleolus, and dual-based. The characteristic function values of the coalitions are given in Table 3.

**Table 2**
Cost allocations according to three solution concepts.

| | Dual-based | Nucleolus | Shapley |
|---|---|---|---|
| SP1 | 3.07 | 3.12 | 4.00 |
| SP2 | 1.93 | 1.98 | 1.67 |
| SP3 | 1.59 | 1.50 | 0.92 |

**Table 3**
The values of coalitions.

| | | | |
|---|---|---|---|
| $v(\{1\})$ | 6.13 | $v(\{1,2\})$ | 5.15 |
| $v(\{2\})$ | 2.61 | $v(\{1,3\})$ | 4.67 |
| $v(\{3\})$ | 1.6 | $v(\{2,3\})$ | 3.53 |

The nucleolus and dual-based cost allocations are in the core, whereas the Shapley's cost allocation is not, *i.e.*, $\phi_1 + \phi_2 = 5.67 > v(\{1,2\}) = 5.15$ (Tables 2 and 3). Hence, these two SPs have the incentive to leave the grand coalition and form a new coalition. The grand coalition is not stable with the Shapley cost allocation.

## 3. Solving the primal problem

We present the method of dealing with the large-scale **Primal**$(\mathcal{N})$ problem in this section. LP can be solved in a polynomial time. However, **Primal**$(\mathcal{N})$ is indeed a large-scale optimization problem because of a large number of content items in the network. Fortunately, **Primal**$(\mathcal{N})$ has a special structure: the block-angular form (Fig. 2), which can be decomposed into many subproblems using the Dantzig–Wolfe decomposition [18,19]. The non-separable constraints are $\sum_{m\in\mathcal{M}} s^m x_i^m \leqslant B_i$ for all $i \in \mathbf{n}(\mathcal{N})$. The constraints of the subproblem associated with content item $m$ are

$$y_{ij}^m \leqslant x_j^m, \quad \forall i,j \in \mathbf{n}(\mathcal{N}), i \neq j,$$
$$x_i^m + \sum_{\substack{j\in\mathbf{n}(\mathcal{N})\\ j\neq i}} y_{ij}^m + z_i^m \geqslant 1, \quad \forall i \in \mathbf{n}(\mathcal{N}),$$
$$x_i^m \geqslant 0, y_{ij}^m \geqslant 0, z_i^m \geqslant 0, \quad \forall i,j \in \mathbf{n}(\mathcal{N}), i \neq j.$$

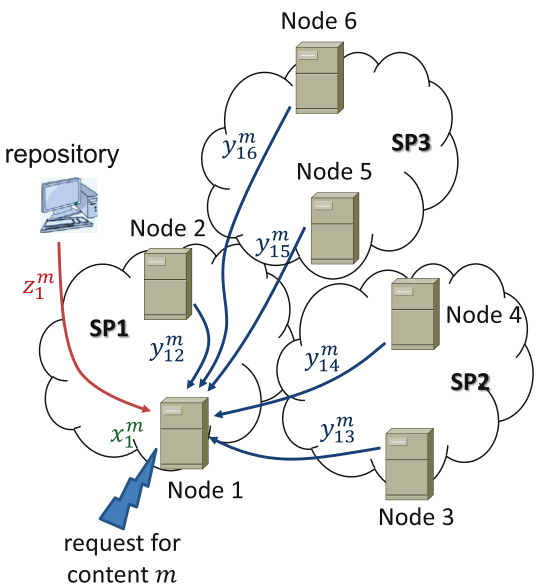


**Fig. 1.** Example of cooperative caching in a content-oriented network with three SPs. Node 1 receives $y_{1i}^m$ fraction of content from node $i$ and $z_1^m$ fraction of content from $m$'s repository to serve the request for content $m$ at node 1 containing $x_1^m$ fraction of content.

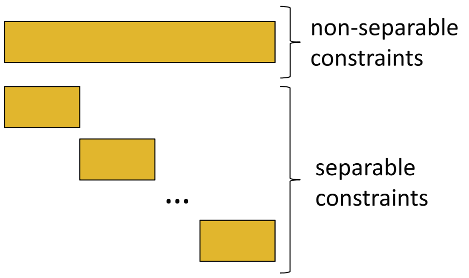


**Fig. 2.** The block-angular structure of the primal problem.

In each iteration, a new column is inserted into the master problem. At the $K$-th iteration, a new extreme point $(\boldsymbol{x}_K, \boldsymbol{y}_K)$, which is the solution to the subproblems of the previous iteration, is inserted into the basic. The cost usage with respect to $(\boldsymbol{x}_K, \boldsymbol{y}_K)$ is $b_K = \sum_{m\in\mathcal{M}} \sum_{i\in\mathbf{n}(\mathcal{N})} s^m d_i^m \left(\sum_{j\in\mathbf{n}(\mathcal{N}) j\neq i} c_{ij} y_{ij}^m + c_i^m z_i^m\right)$. The resource usage with respect to $(\boldsymbol{x}_K, \boldsymbol{y}_K)$ is $\boldsymbol{a}_K = [a_{1K} \ldots a_{|\mathbf{n}(\mathcal{N})|K}]^T$, where $a_{iK} = \sum_{m\in\mathcal{M}} s^m x_i^m, \forall i \in \mathbf{n}(\mathcal{N})$. The master problem has the following form:

$$\textbf{Master}: \quad \text{Min.} \quad \sum_{k=1}^{K} b_k \lambda_k$$
$$\text{st.} \quad \sum_{k=1}^{K} a_{ik}\lambda_k \leqslant B_i, \forall i \in \mathbf{n}(\mathcal{N}) \tag{12}$$
$$\sum_{k=1}^{K} \lambda_k = 1. \tag{13}$$

Let $\alpha_i$ be the optimal multiplier associated with storage constraint (12) with respect to node $i$ and $\sigma$ be the optimal multiplier associated with constraint (13). The subproblem associated with content $m$ with respect to the optimal multipliers is given by

$$\textbf{Sub}-\textbf{problem}^m : \text{Min.} \sum_{i\in\mathbf{n}(\mathcal{N})} s^m d_i^m \left( \sum_{\substack{j\in\mathbf{n}(\mathcal{N})\\ j\neq i}} c_{ij} y_{ij}^m + c_i^m z_i^m + \alpha_i \frac{x_i^m}{d_i^m} \right)$$
$$\text{st.} \quad y_{ij}^m \leqslant x_j^m, \forall i,j \in \mathbf{n}(\mathcal{N}), i \neq j,$$
$$x_i^m + \sum_{\substack{j\in\mathbf{n}(\mathcal{N})\\ j\neq i}} y_{ij}^m + z_i^m \geqslant 1, \forall i \in \mathbf{n}(\mathcal{N}),$$
$$x_i^m \geqslant 0, y_{ij}^m \geqslant 0, z_i^m \geqslant 0, \forall i,j \in \mathbf{n}(\mathcal{N}), i \neq j.$$

The solutions to the subproblems propose a new column inserted to the basic in the next iteration. The algorithm stops when the reduced cost $RC = v_K - \sigma$ is less than 0, where

$$v_K = \sum_{m\in\mathcal{M}} \sum_{i\in\mathbf{n}(\mathcal{N})} s^m d_i^m \left( \sum_{\substack{j\in\mathbf{n}(\mathcal{N})\\ j\neq i}} c_{ij} y_{ij}^m + c_i^m z_i^m + \alpha_i \frac{x_i^m}{d_i^m} \right).$$

**Algorithm 1.** Dantzig–Wolfe decomposition algorithm.

1: Initialize the basic $\boldsymbol{a} = \text{NULL}$, $\boldsymbol{b} = \text{NULL}$;
2: $K = 1$;
3: **while** $RC > 0$ **do**
4: Calculate amount of cost and storage with the new column, $\boldsymbol{a}_K,\ b_K$;
5: $\boldsymbol{a} = [\boldsymbol{a}\ \boldsymbol{a}_K]$ and $\boldsymbol{b} = [\boldsymbol{b}\ b_K]$;
6: Solve **Master** problem for the optimal multipliers $\alpha_i,\ \forall i \in \boldsymbol{n}(\mathcal{N})$ and $\sigma$;
7: Solve $|\mathcal{M}|$ subproblems parallelly corresponding to $\alpha_i,\ \forall i \in \boldsymbol{n}(\mathcal{N})$;
8: Calculate the reduced cost $RC$;
9: $K = K + 1$;
10: **end while**

Algorithm 1 describes the Dantzig–Wolfe decomposition algorithm solving **Primal**$(\mathcal{N})$ problem. For instance, the LP with 10,000 contents and 20 caching nodes in the grand coalition cannot be solved using a normal PC because of out of the memory. However, the problem is completely solved on the same PC through decomposition.

In case there is symmetry in **Primal**$(\mathcal{N})$, the size of the optimization problem can be reduced. Let $N = |\mathcal{N}|$ and $n = |\boldsymbol{n}(\mathcal{N})|$ be the number of SPs and the number of caching nodes, respectively, and $M = |\mathcal{M}|$ be the number of contents in the network. **Primal**$(\mathcal{N})$ problem has $Mn(n+1)$ variables and $Mn(2n+1)+n$ constraints. However, the reduced problem has only $MN(N+2)$ variables and $MN(2N+3)+N$ constraints if all the nodes in the SP have the same storage capacities and the costs for one unit of content $m$ transferred from the original source to every node in the SP are also the same.

To allow fractional caching, each content is stored in multiple chunks. Each chunk is identified by a chunk identification (ID). Provided that the fractions stored at the nodes of any content do not overlap, the optimal bandwidth expense can be achieved regardless of the order of the fractions stored at the caching nodes. To avoid the overlapping, the server that handles the master problem assigns the ranges of chunk IDs to the SPs, and each SP assigns the chunks to its nodes. A message passing mechanism can be used to inform between the server and the caching nodes. The nodes then store the fractions of the contents with the assigned chunked IDs.

## 4. Variations of the primal problem

We present two variations of **Primal**$(\mathcal{N})$ that the coalitional game framework in Section 2 can be applied analogously in this section. The first variation is the bandwidth-saving maximization problem addressed in Ref. [1]. In any coalition $\mathcal{S}$, all the contents must be downloaded from the repositories if there is not any replication in $\mathcal{S}$. Hence, the total traffic transferred within coalition $\mathcal{S}$ is $\sum_{m\in\mathcal{M}}\sum_{i\in\boldsymbol{n}(\mathcal{S})} s^m d_i^m c_i^m$. The total bandwidth saving is the subtraction of the total bandwidth cost with caching from the total bandwidth cost without caching, *i.e.*, $\sum_{m\in\mathcal{M}}\sum_{i\in\boldsymbol{n}(\mathcal{S})} s^m d_i^m c_i^m - \sum_{m\in\mathcal{M}}\sum_{i\in\boldsymbol{n}(\mathcal{S})} s^m d_i^m \left(\sum_{j\in\boldsymbol{n}(\mathcal{S}), j\neq i} c_{ij} y_{ij}^m + c_i^m z_i^m\right)$. With the use of the implicit constraint $x_i^m + \sum_{j\in\boldsymbol{n}(\mathcal{S}), j\neq i} y_{ij}^m + z_i^m = 1$ for all $i \in \boldsymbol{n}(\mathcal{S}), m \in \mathcal{M}$, the total bandwidth saving becomes $\sum_{m\in\mathcal{M}}\sum_{i\in\boldsymbol{n}(\mathcal{S})} s^m d_i^m \left(c_i^m x_i^m + \sum_{j\in\boldsymbol{n}(\mathcal{S}), j\neq i} (c_i^m - c_{ij}) y_{ij}^m\right)$. Therefore, the equivalent bandwidth-saving maximization problem is as follows

$$
\begin{aligned}
\text{Max.} & \sum_{m\in\mathcal{M}}\sum_{i\in\boldsymbol{n}(\mathcal{S})} s^m d_i^m \left(c_i^m x_i^m + \sum_{\substack{j\in\boldsymbol{n}(\mathcal{S})\\ j\neq i}} (c_i^m - c_{ij}) y_{ij}^m\right) \\
\text{st.} & \sum_{m\in\mathcal{M}} s^m x_i^m \leqslant B_i, \forall i \in \boldsymbol{n}(\mathcal{S}), \\
& y_{ij}^m \leqslant x_j^m, \quad \forall i,j \in \boldsymbol{n}(\mathcal{S}), i \neq j, m \in \mathcal{M}, \\
& x_i^m + \sum_{\substack{j\in\boldsymbol{n}(\mathcal{S})\\ j\neq i}} y_{ij}^m + z_i^m \leqslant 1, \forall i,j \in \boldsymbol{n}(\mathcal{S}), i \neq j, m \in \mathcal{M}, \\
& x_i^m \geqslant 0, y_{ij}^m \geqslant 0, z_i^m \geqslant 0, \quad \forall i \in \boldsymbol{n}(\mathcal{S}), m \in \mathcal{M},
\end{aligned}
$$

where the constraint $x_i^m + \sum_{j\in\boldsymbol{n}(\mathcal{S}), j\neq i} y_{ij}^m + z_i^m = 1$ is relaxed to $x_i^m + \sum_{j\in\boldsymbol{n}(\mathcal{S}), j\neq i} y_{ij}^m + z_i^m \leqslant 1$ in the maximization problem.

In the second variation, the constraint on the link bandwidth can be added to the primal problem. Similar to the problem considered in Ref. [2], let $r^m$ be the bit-rate of video $m$ (in Mbps) and $f_i^m(t)$ be the number of requests for video $m$ at node $i$ in time-slot $t$. Note that if the demand $d_i^m$ is the number of requests of content $m$ at node $i$ in a time period $T$, *e.g.*, minutes, hours, days, weeks, etc., then $d_i^m = \sum_{t=1}^{T} f_i^m(t)$. Denote $\mathcal{L}$ be the set of links connecting the nodes, $L(j,i)$ be the path from node $j$ to node $i$, and $L(o,i)$ be the path from repository to node $i$. Let $W_l, l \in \mathcal{L}$ be the link capacities (in Mbps). The capacity constraint is that the traffic on every link must not exceed the link capacity, *i.e.*, $\sum_{m\in\mathcal{M}}\sum_{i\in\boldsymbol{n}(\mathcal{S})} r^m f_i^m(t) \left(\sum_{j\in\boldsymbol{n}(\mathcal{S}),\ j\neq i,\ l\in L(j,i)} y_{ij}^m + z^m_{i\in\boldsymbol{n}(\mathcal{S}),\ l\in L(o,i)}\right) \leqslant W_l$ for all $t \in T$ and $l \in \mathcal{L}$. With this new constraint added to **Primal**$(\mathcal{S})$ problem, the analysis framework presented in Section 2 can be analogously applied. The dual problem of the new problem is shown in appendix.

## 5. Numerical results

In the experiments, the demands of the contents are assumed to follow Zipf–Mandelbrot distribution given by Eq. (5) with $q = 10$ and $a = 1$. The request rate to each SP is 10 requests per minute and evenly distributed among the caching nodes within the SP. The network is assumed to have 10,000 contents and the network topology is described in Fig. 3. When two SPs cooperate, the traffic between them is transferred via their gateways. The cost of the path between any two nodes is the number of hops. Accordingly, $c_{ij} = 2$ for all $i,j$ in the same SP and $c_{ij} = 3$ if $i$ and $j$ are in different SPs. Suppose that the cost to transfer one unit of any content $m \in \mathcal{M}$ from the original source to node $i$ for all $i \in \boldsymbol{n}(\mathcal{N})$ is $c_i^m = 10$. All the contents have the equal sizes, *i.e.*, $s = 2$ GB. The storage capacity of each node is assumed 500 GB unless specified otherwise.

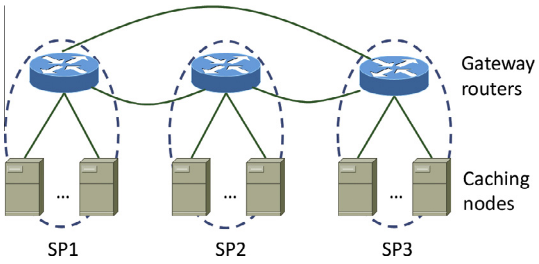


**Fig. 3.** Network topology in the simulations.

### 5.1. Effectiveness of the cooperation

In this experiment, we monitor the payoff of each SP when the number of SPs joining the coalition increase gradually. Each SP is assumed to have ten caching nodes. The Shapley and nucleolus solutions in this case are also the same as the dual-based solution. Fig. 4 shows that the dual-based cost per SP gradually decreases when the number of SPs joining into the coalition increases. Without cooperation, the shared cost for each SP is 64.53 GB per minute. The collaboration of two SPs causes the cost per SP to decrease to 54.21 GB per minute, which is 84% of the cost of non-cooperation. Collaborating 3 SPs results in the cost per SP dropping to 47.52 GB per minute, which is 73.6% of the cost of non-cooperation (Fig. 4). However, the cost does not linearly decrease as the number of SPs in the coalition increases. As shown in Fig. 4, the decrease in the shared cost of each SP in a larger coalition is not as steep as that in a smaller coalition. For example, when the coalition size is eight SPs, if a new SP joins the coalition, its cost decreases to about 50% compared to the non-cooperative cost. Nevertheless, each prior SP reduces its cost by only 1.89%. Intuitively, if the coalition already contains most of the items, then adding more SPs into the coalition does not bring considerable benefit to the current SPs in the coalition.

We now monitor the effect of shape parameter $a$ in the popularity distribution (5) to the costs of SPs. The shared cost of each SP in the coalition decreases as the number of SPs in the coalition increases, which agrees with the previous result (see Fig. 5). Moreover, given the size of the coalition, the effectiveness of the cooperative caching improves as the popularity distribution is steeper (*i.e.*, a smaller $a$) and vice versa.

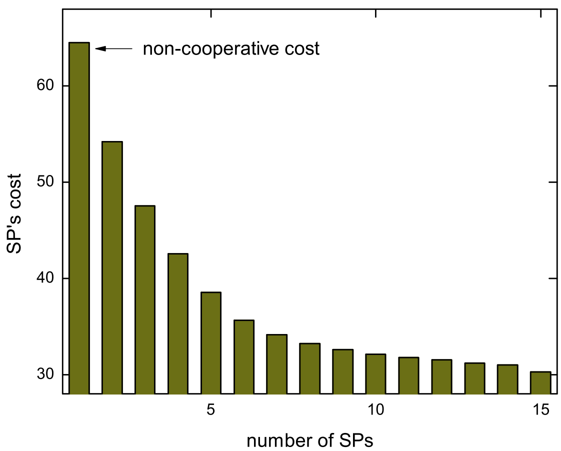


**Fig. 4.** Cost of SP as a function of the number of SPs in a coalition ($a = 1$).

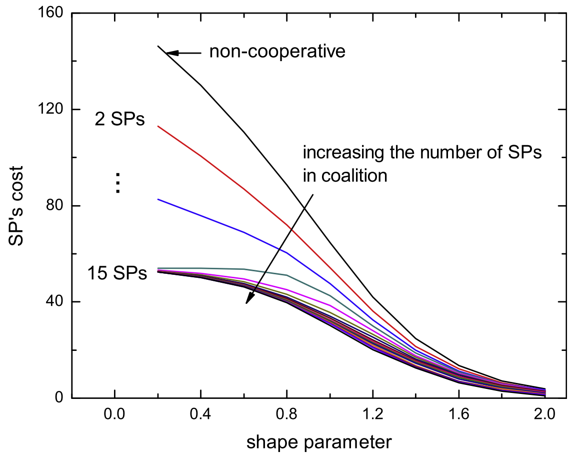


**Fig. 5.** Cost of SP as a function of shape parameter $a$ in the popularity distribution (5).

### 5.2. Shared cost of SP in coalition

In this experiment, the shared cost of each SP is examined when varying the SP's assets in the contribution to the coalition. The assets of SP are usually the number of nodes in the network and the storage capacities of the nodes. Logically, the cost of an SP with a higher contribution should be lower than that of the other SPs. We consider two SPs. All parameters of the first SP are fixed whereas those of the second SP are varied. The request rates to the two SPs are assumed to be equal.

To calculate the nucleolus values in case of two SPs, the excesses of coalitions $\{1\}$ and $\{2\}$ are $p_1 - v(\{1\})$ and $p_2 - v(\{2\})$, respectively. These values are negative. Meanwhile, the total excesses of these two coalitions are $p_1 + p_2 - v(\{1\}) - v(\{2\}) = v(\{1,2\}) - v(\{1\}) - v(\{2\})$, which are constant numbers. Hence, the minimum of the maximum rule leads to two equal excesses. Therefore, the nucleolus cost allocations of SP1 and SP2 are $\frac{v(\{1,2\})+v(\{1\})-v(\{2\})}{2}$ and $\frac{v(\{1,2\})+v(\{2\})-v(\{1\})}{2}$, respectively. These values are also equal to the Shapley values.

The cost per node of both SPs decreases when the number of storage nodes in SP1 is fixed to ten nodes and the number of nodes in SP2 is gradually increased (see Fig. 6a). However, the cost of SP2 decreases faster than that of SP1. The dual-based, Shapley, and nucleolus costs of SP2 are higher than those of SP1 when the number of nodes in SP2 is less than that in SP1 and vice versa. Therefore, the SP with more caching nodes in the coalition pays the less cost.

Similarly, Fig. 6b shows that when the storage capacity of SP1 is fixed and that of SP2 is increased, the shared cost of SP2 is lower than that of SP1 when the storage capacity of the nodes in SP2 is less than that in SP1, and vice versa. In three plots of Fig. 6, the non-cooperative caching always yields a higher cost than the dual-based, Shapley, or nucleolus costs.

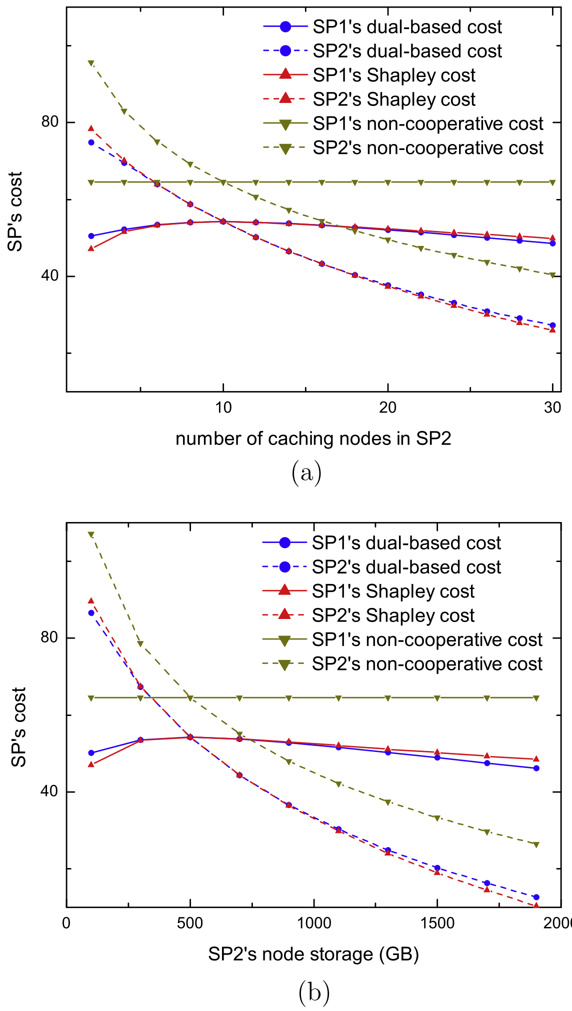


**Fig. 6.** Comparison of the cost of two SPs when the parameters of SP1 are fixed (*i.e.*, the number of nodes is 10, the storage capacity per node is 500 GB) and those of SP2 are varied. (a) Varying the number of the caching nodes of SP2 and (b) varying the storage of the caching nodes of SP2.

## 6. Conclusions

This study considers the cooperation among SPs in caching the contents using the coalitional game approach. The core of the bandwidth-expense coalitional game is non-empty. The dual-based cost allocation stabilizes the grand coalition. Other solution concepts are also introduced, *i.e.*, Shapley and nucleolus solutions. By utilizing the block-angular structure of the primal problem, the Dantzig-Wolfe decomposition approach is applied to solve the large-scale LP. The main problem is decomposed into many subproblems, which can be solved in parallel.

## Acknowledgments

This research was partially funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2014 and the ICT R&D program of MSIP/IITP, Republic of Korea (2014-044-011-003, Open control based on distributed mobile core network).

## Appendix A

Let $\delta_l(t)$ be the Lagrange multiplier associated with the link bandwidth constraint. The dual problem of the problem with the added link bandwidth constraints is given by

$$\text{Max.} \sum_{i\in\mathbf{n}(\mathcal{S})}\left(\sum_{m\in\mathcal{M}}\gamma_i^m - B_i\alpha_i\right) - \sum_{t\in T}\sum_{l\in\mathcal{L}}W_l\delta_l(t)$$

$$\text{st.} s^m\alpha_i - \sum_{j\in\mathbf{n}(\mathcal{S}),j\neq i}\beta_{ji}^m \geqslant \gamma_i^m, \forall i\in\mathbf{n}(\mathcal{S}), m\in\mathcal{M} \quad (14)$$

$$s^m d_i^m c_{ij} + \beta_{ij}^m + \sum_{t\in T}\sum_{l\in L(j,i)} r^m\delta_l(t)f_i^m(t) \geqslant \gamma_i^m, \quad \forall i,j\in\mathbf{n}(\mathcal{S}), i\neq j, m\in\mathcal{M} \quad (15)$$

$$s^m d_i^m c_i^m + \sum_{t\in T}\sum_{l\in L(o,i)} r^m\delta_l(t)f_i^m(t) \geqslant \gamma_i^m, \forall i\in\mathbf{n}(\mathcal{S}), m\in\mathcal{M}, \quad (16)$$

$$\alpha_i \geqslant 0, \beta_{ij}^m \geqslant 0, \gamma_i^m \geqslant 0, \delta_l(t) \geqslant 0, \quad \forall i,j\in\mathbf{n}(\mathcal{S}), i\neq j, m\in\mathcal{M}, l\in\mathcal{L}, t\in T. \quad (17)$$

## References

[1] S. Borst, V. Gupta, A. Walid, Distributed caching algorithms for content distribution networks, in: Proceedings IEEE INFOCOM 2010, March 2010, pp. 1–9.
[2] D. Applegate, A. Archer, V. Gopalakrishnan, S. Lee, K.K. Ramakrishnan, Optimal content placement for a large-scale vod system, in: Proceedings of the 6th International COnference, ser. Co-NEXT '10, ACM, New York, NY, USA, 2010, pp. 4:1–4:12.
[3] H. Xie, G. Shi, P. Wang, Tecc: towards collaborative in-network caching guided by traffic engineering, in: Proceedings IEEE INFOCOM 2012, March 2012, pp. 2546–2550.
[4] J. Dai, Z. Hu, B. Li, J. Liu, B. Li, Collaborative hierarchical caching with dynamic request routing for massive content distribution, in: Proceedings IEEE INFOCOM 2012, March 2012, pp. 2444–2452.
[5] G. Dan, Cache-to-cache: could SPs cooperate to decrease peer-to-peer content distribution costs?, IEEE Trans Parallel Distr. Syst. 22 (9) (2011) 1469–1482.
[6] J. Dai, F. Liu, B. Li, B. Li, J. Liu, Collaborative caching in wireless video streaming through resource auctions, IEEE J. Sel. Area. Commun. 30 (2) (2012) 458–466.
[7] J. Dai, B. Li, F. Liu, B. Li, H. Jin, On the efficiency of collaborative caching in SP-aware p2p networks, in: Proceedings IEEE INFOCOM 2011, April 2011, pp. 1224–1232.
[8] V. Pacifici, G. Dan, Content-peering dynamics of autonomous caches in a content-centric network, in: Proceedings IEEE INFOCOM 2013, April 2013, pp. 1079–1087.
[9] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, T. Basar, Coalitional game theory for communication networks, IEEE Signal Proc. Mag. 26 (5) (2009) 77–97.
[10] W. Saad, Z. Han, T. Basar, A. Hjorungnes, J.B. Song, Hedonic coalition formation games for secondary base station cooperation in cognitive radio networks, in: IEEE Wireless Communications and Networking Conference (WCNC) 2010, April 2010, pp. 1–6.
[11] W. Saad, Z. Han, A. Hjorungnes, D. Niyato, E. Hossain, Coalition formation games for distributed cooperation among roadside units in vehicular networks, IEEE J. Sel. Area. Commun. 29 (1) (2011) 48–60.
[12] Z. Han, H. Poor, Coalition games with cooperative transmission: a cure for the curse of boundary nodes in selfish packet-forwarding wireless networks, in: 5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops, 2007 (WiOpt 2007), April 2007, pp. 1–8.
[13] E. Markakis, A. Saberi, On the core of the multicommodity flow game, Decis. Support Syst. 39 (1) (2005) 3–10. The Fourth ACM Conference on Electronic Commerce.

[14] C. Singh, E. Altman, The wireless multicast coalition game and the non-cooperative association problem, in: Proceedings IEEE INFOCOM 2011, April 2011, pp. 2705–2713.
[15] C. Singh, S. Sarkar, A. Aram, Provider-customer coalitional games, IEEE/ACM Trans. Network. 19 (5) (2011) 1528–1542.
[16] C. Singh, S. Sarkar, A. Aram, A. Kumar, Cooperative profit sharing in coalition-based resource allocation in wireless networks, IEEE/ACM Trans. Network. 20 (1) (2012) 69–83.
[17] M.J. Osborne, A. Rubinstein, A Course in Game Theory, vol. 1(0262650401), 1994.
[18] V. Chvatal, Linear Programming, W.H. Freeman and Company, 1983.
[19] S. Bradley, A. Hax, T. Magnanti, Applied Mathematical Programming, Addison-Wesley, 1977.
[20] L. Breslau, P. Cao, L. Fan, G. Phillips, S. Shenker, Web caching and zipf-like distributions: evidence and implications, in: Proceedings IEEE INFOCOM '99, vol. 1, March 1999, pp. 126–134.
[21] M. Maschler, B. Peleg, L.S. Shapley, Geometric properties of the kernel, nucleolus, and related solution concepts, Math. Oper. Res. 4 (1979) 303–338.

**Phuong Luu Vo** received her B.Eng and M.Eng degrees in Electrical-Electronics Engineering from Ho Chi Minh City University of Technology, Viet Nam in 1998, 2002, respectively and the Ph.D. degree at Kyung Hee University, Korea in 2014. She is a lecturer of School of Computer Science and Engineering at International University – Viet Nam National University Ho Chi Minh City. Her research interest is applying optimization theory, control theory, and game theory to allocate the resources in communication networks.

**Duc Ngoc Minh Dang** received his B.Eng and M.Eng degrees in telecommunications engineering from Ho Chi Minh City University of Technology, Viet Nam, in 2005 and 2007, respectively and the Ph.D. degree in Computer Engineering from Kyung Hee University, Korea, in 2014. From 2005 to 2008, he was a Senior Telecom Engineer with TMA Solutions, Viet Nam. Since 2008 to 2011, he joined Ton Duc Thang University, Viet Nam, where he worked as the Head of Electronics and Telecommunications. He is currently with Ton Duc Thang University. His research interests include the MAC protocols in wireless ad hoc networks and vehicular ad hoc networks.

**Sungwon Lee** received the Ph.D. degree from Kyung Hee University, Korea. He is a professor of the Computer Engineering Departments at Kyung Hee University, Korea. He was a senior engineer of Telecommunications and Networks Division at Samsung Electronics Inc. from 1999 to 2008. He is a editor of the Journal of Korean Institute of Information Scientists and Engineers: Computing Practices and Letters. His research interests include mobile broadband networks and communication protocols. He is a member of IEEE, KIISE, KICS and KIPS.

**Choong S. Hong** received his B.S. and M.S. degrees in Electronic Engineering from Kyung Hee University, Seoul, Korea, in 1983, 1985, respectively. In 1988 he joined KT, where he worked on Broadband Networks as a member of the technical staff. From September 1993, he joined Keio University, Japan. He received the Ph.D. degree at Keio University in March 1997. He had worked for the Telecommunications Network Lab. KT as a senior member of technical staff and as a director of the networking research team until August 1999. Since September 1999, he has worked as a professor of the Department of Computer Engineering, Kyung Hee University. He has served as a Program Committee Member and an Organizing Committee Member for International conferences such as NOMS, IM, APNOMS, E2EMON, CCNC, ADSN, ICPP, DIM, WISA, BcN, TINA, SAINT and ICOIN. His research interests include Ad hoc Networks, Power Line Communications, Network Management and Future Internet. He is a member of IEEE, ACM, IEICE. IPSJ, KIISE, KICS and KIPS.

**Quan Le-Trung** has got his Doctor of Technical Science (Dr.techn.) degree at Dept. of Telecooperation, Johannes Kepler University-Linz, Austria in 2007, then he has spent 9 months postdoc there, and another 24 months postdoc at Networks & Distributed Systems group, Dept. of Informatics, University of Oslo, Norway. He also got his Master of Engineering (M.Eng.) at Telecom program, Asian Institute of Technology, in Thailand in 2002, and the Bachelor of Engineering (B.Eng.) in Dept. of CSE, HCM CIty University of Technology, Viet Nam in 1997. His research interests span on a variety of areas in networking, and focuses now on the Wireless Embedded Internet, Cyber Physical Systems, Internet of Things.