ELSEVIER

Computer Communications 24 (2001) 93–104

computer communications

www.elsevier.com/locate/comcom

Content-selection strategies for the periodic prefetching of WWW resources via satellite

M. Dikaiakos[a,*], A. Stassopoulou[b]

[a]Department of Computer Science, University of Cyprus, P.O. Box 20537, Nicosia, Cyprus
[b]Department of Computer Science, Intercollege, Nicosia, Cyprus

Received 4 September 2000; accepted 4 September 2000

Abstract

In this paper we study satellite-caching, that is, the employment of satellite multicasting for the dissemination of prefetched content to WWW caches. This approach is currently being deployed by major satellite operators and ISPs around the world. We introduce a theoretical framework to study satellite-caching and formalize the notions of Utility and Quality of Service. We explore two charging schemes, Usage- and Subscription-based pricing, and propose a framework for negotiating the provision of the satellite-caching service between a satellite operator and its potential clients. We use this negotiation framework to compare theoretically the two pricing schemes at hand. We apply our modeling to formulate the selection of Web-content for satellite-multicasting as a combinatorial optimization problem. We study the complexity of Web-content selection and prove it is NP-complete. Finally, we propose and implement an approximation algorithm for content selection, and conduct experiments to assess its efficiency, validity and applicability.

Keywords: Web caching; Satellite multicasting; Content selection; Quality of service; Pricing

1. Introduction

World Wide Web usage represents the largest single source of traffic on Internet and is expected to grow further with the rise of Internet usage [2,19] and the advent of new Web-based applications [14,27,32]. The increased WWW usage has resulted in heavy workloads on popular Web servers and local networks. Currently, these loads are difficult to meet; in the future, if Web use continues to grow as fast, systems and networks face the danger of being eventually overwhelmed.

As a way of coping with increased Web-loads, the Internet community has widely adopted and implemented Web caching. The fundamental idea is simple: whenever a user seeks a hyper-document on the Web, instead of automatically connecting to the Web server designated by the corresponding URL, his Web-client checks if the hyper-document is available at a "nearby" cache. In that case, the user receives a cached copy of the document, thus avoiding a slow connection to the originating server of the document. The potential gains from Web caching are obvious: caching popular documents on a local or wide-area network reduces the incoming traffic to this network and the load imposed on originating Web servers. Furthermore, users are expected to experience much shorter response times when receiving documents from nearby caches than from distant servers.

In the complex hierarchy of wide-area and local networks that connect Web-clients to information sources, there are several places where documents can be cached. In particular, caching can take place on proxy servers that reside at a local or regional network. Such servers are used extensively by network administrations, Internet Service Providers (ISPs) and corporations seeking to provide their user communities with improved WWW access. The use of proxy servers as Web caches raises numerous research issues related to proxy server performance, effective caching and efficient caching architectures (e.g. see Refs. [5,9,10,11,20,22,31,34,35]). The wide-scale deployment of hierarchical and cooperative Web caches opens new grounds for the development and implementation of cache-based techniques to sustain adequate levels of Web-performance, like prefetching [15] and content dissemination [6,16]. Our conjecture is that, besides the expected performance advantages, such techniques represent a promising field for the exploration of emerging schemes for pricing and charging Internet-content [12,18,23,24].

* Corresponding author.
E-mail addresses: mdd@ucy.ac.cy (M. Dikaiakos), athena@intercol.edu (A. Stassopoulou).

0140-3664/01/$ - see front matter
PII: S0140-3664(00)00292-9

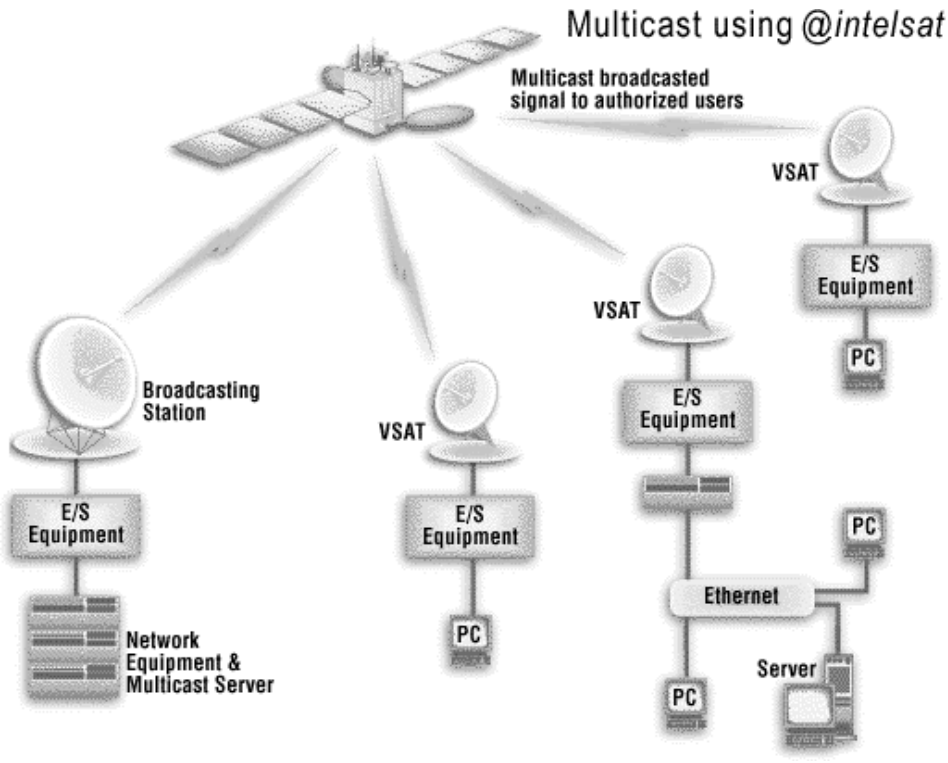

Fig. 1. Multicasting Web-content (Image courtesy INTELSAT).

In this paper, we address the employment of satellite multicasting to disseminate Web content to the caches of Internet-Service Providers, in a scheme called *satellite-caching*; this approach is currently deployed by major satellite operators and ISPs around the world (e.g. see Refs. [1,3,4,21]). We introduce a theoretical framework to study issues pertinent to this problem and formalize the notions of *Utility*and *Quality* of *Service*perceived by clients of satellite-multicasting services. We explore two charging schemes, *Usage*- and *Subscription*-based *pricing*, and propose a framework for negotiating the provision of the satellite-caching service between the satellite operator and its potential clients. We use this negotiation framework to compare theoretically the two pricing schemes at hand. We apply our modeling to formulate the selection of Web-content for satellite-multicasting as a combinatorial optimization problem. We study the complexity of *Web-content selection* and prove it is NP-complete. Finally, we propose and implement an approximation algorithm for content selection, and conduct experiments to assess its validity and applicability.

The rest of this paper is organized as follows. Section 2 presents the principles of satellite-caching and discusses related work. Our theoretical modeling is described in Section 3. Section 4 provides a definition of the Web-content-selection problem and studies its complexity. Section 5 introduces our approximation algorithm for Web-content selection and presents our experimental study. We conclude in Section 6, where we also discuss issues for future work.

2. Multicasting Web-content via satellite

Satellites are used extensively to multicast digital content, i.e. to dispatch information to specific groups of users within a satellite network. Information is multicast either for free or according to usage or subscription-based pricing. Recently, satellite networks have been adopted as an alternative choice to terrestrial networks for TCP/IP provision by ISPs that seek to establish access connectivity to global Internet, by backbone operators that wish to extend their terrestrial networks anywhere in the world (see Fig. 1) [1,3,4].

Satellite networks are further employed to expand the performance gains achieved by Web-caching hierarchies deployed on Internet. In particular, satellite operators have started providing satellite-caching services, which consist of periodic multicasting of Web content to subscribed clients. Customers are typically Internet Service Provides that wish to "enrich" their cache hierarchies with Web content, without overloading their terrestrial links. To this end, satellite operators use their terrestrial connections to Internet backbones in order to "pull" a collection of WWW-objects into their multicast servers. A multicast server sends the collection of content to the satellite through an up-link channel; the collection is subsequently broadcast ("pushed") to authorized subscriber-organizations through the satellite's down-link channels.

A subscriber stores this content into an *institutional* Web-cache, which is typically the "parent" in a Web-caching

hierarchy established on top of proxy servers such as Squid [35]. This process is repeated periodically throughout a day, materializing a *periodic push* scheme for information dissemination (see the taxonomy in Ref. [7]).

In essence, the satellite operator (*content-distributor*) provides Web content to various client-organizations around the world (typically ISPs). On their behalf, organizations purchase this service as a means for "prefetching" Web content through existing, under-utilized satellite links. Subscribers redistribute the content to their user-base through established Web-caching schemes.

According to an alternative scheme, requests for documents that are not found in the caching hierarchy of a subscriber ISP, are obtained by the satellite-operator's terrestrial site and broadcast to all subscriber-ISP caches [29,30]. With this scheme, a cache ends up storing the documents requested by approximately all the clients connected to the satellite distribution [29,30].

The basic premise behind satellite-caching services is that the "prefetched" content covers adequately the interests of subscribers, improves the hit-ratio of installed Web-caches and, therefore, relieves overloaded terrestrial TCP/IP connections [30]. The soundness of this premise and the overall feasibility of the proposed approach depend on a number of open issues, such as:

- The design of content-selection algorithms — such algorithms should take into account client utility and distributor costs. Note that utility is a measure of the "pleasure" a client derives from the consumption of a particular service or good.
- The *profiles* of potential subscribers, which represent their information interests, the size of their customer-base, the level and cost of their terrestrial Internet-connectivity, etc.
- The scheduling of data broadcasts. The way clients perceive and formalize the utility they expect to receive with the adoption of satellite-caching.
- The charging schemes proposed by content-distributors and the negotiation framework that can be established between distributors and clients to reach flexible and mutually profitable pricing mechanisms.

2.1. Related work

The development of wireless and satellite networks, and the expanding availability of assymmetric high-bandwidth links have created a lot of interest on issues related to data broadcasting. A large number of projects have examined various aspects of information dissemination over broadcast channels. There are two basic approaches for data delivery through broadcasting: *pull-based*and *push-based[7]*. In the former, user requests are forwarded directly to a broadcasting server, which responds by broadcasting information over a satellite down-link channel to its clients. In the latter, users cannot inform directly the server about their requests. Therefore, the server relies on its knowledge of past user-access patterns to decide what information to broadcast.

A major issue in data broadcasting is the organization of data in an optimal broadcast schedule. This problem is addressed by Su et al. in Ref. [33]: the authors formulate the scheduling problem as a deterministic and as a stochastic Markov Decision Process for push-based and pull-based systems, respectively. In a similar context, Aksoy and Franklin study algorithms for scheduling the dissemination of data in an "aperiodic pull" scheme [7,8]. Under this scheme, client-requests that cannot be served locally are sent to a broadcasting server via terrestrial links. The server collects requested information and uses the proposed scheduling algorithm to decide the sequence of data-item broadcasts over the satellite.

The problem of determining information caching strategies for minimizing storage and network costs is examined in the context of personalized video-on-demand services by Papadimitriou et al. [25]. The authors define a formulation for modeling storage and network costs. This formulation is used to determine optimal video transmission and caching schedules according to individual preferences that determine the video requested and the expected viewing times.

A combination of Web caching with multicasting is examined by Rodriguez et al. in Ref. [28]. The authors model Internet as a multi-level hierarchy of WWW caches and introduce a formulation to analyze the combination of pull/push schemes with hierarchical Web caching. Furthermore, they propose a *hierarchical caching push* scheme, according to which clients subscribe to documents available in an origin server residing at the root of the hierarchy. Clients receive document updates or update notifications from the origin server through caches higher in the hierarchy. Emphasis is given on the employment of hierarchical Web caching as a minimal-modification alternative solution to Internet multicasting.

Finally, satellite distribution as a way for prefetching WWW resources is addressed by Rodriguez and Biersack in Refs. [29,30]. The authors examine a scheme according to which, documents requested for the first time by any subscriber, are broadcast to all subscribers' caches via the satellite. A theoretical formulation is introduced to analyze the performance of cache-satellite distribution. The application of this model gives interesting predictions on the hit-ratio and latency improvements achieved with, and the storage-capacity and satellite-bandwidth requirements raised by, the adoption of this scheme.

Most of the schemes described above deal with the tuning of information-dissemination systems to better serve immediate user-requests either through improved multicast-scheduling algorithms, "lighter" multicast architectures, or optimized caching schedules. In contrast, our approach looks into the case where the broadcast channel is used simultaneously with terrestrial links and in conjunction with Web-caching hierarchies deployed. Information

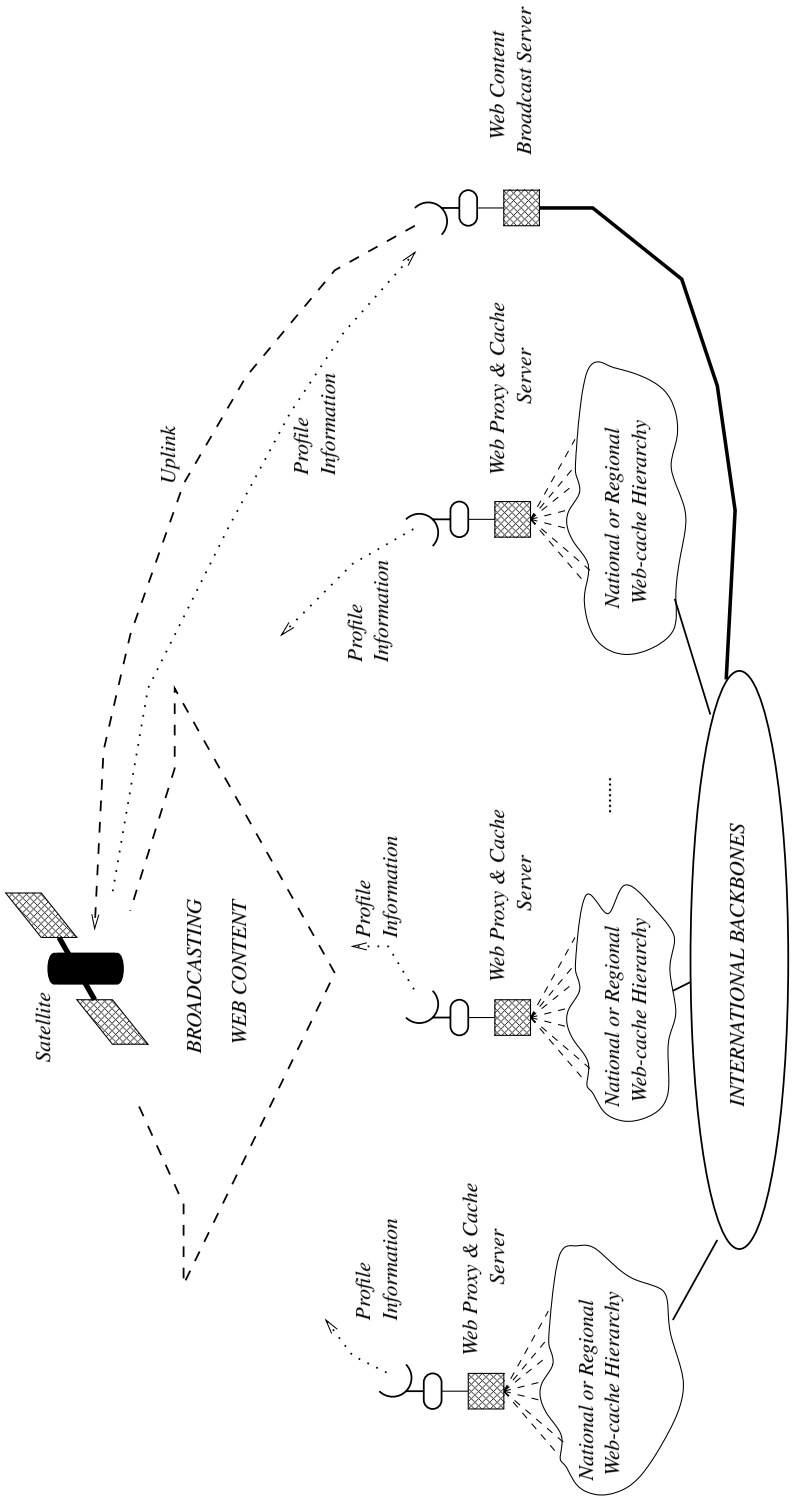

Fig. 2. Multicasting content to Web-caches.

multicasts over the satellite are "pushed" on to Web caches in an effort to prefetch data without overloading terrestrial networks. This is also the focus of the work presented in Refs. [29,30].

Our approach differs from this, however, in a number of ways: Firstly, we focus on schemes for periodic rather than continuous prefetching. Secondly, we study the application of satellite-caching for prefetching Web-content to groups of users that belong to different backgrounds. So, instead of considering a single user population, we consider multiple ones with possibly different characteristics (language, culture, size) and requirements. In that context, we introduce a novel formulation of the particular problem of content-selection, taking into account the utility and quality of the satellite-caching service.

3. A theoretical formulation of Web-content selection

3.1. Basic assumptions

In this paper we address the problem of content-selection by multicast operators. We assume that a content-distributor multicasts content according to a simple, periodic schedule. On every multicast, all client-organizations receive identical information. These assumptions correspond to the actual configuration of emerging satellite services that multicast WWW data on an international scale [21].

For the content-distributor to choose Web content appropriately, we assume further that it collects *profile information* from each client regularly; a profile represents the most recent information-needs of a client. Based on client-profiles, the distributor can select the content to be pulled from the Web and stored on the broadcasting server for the subsequent transmission (see Fig. 2).

For a theoretical formulation of the content-selection problem, we assume that the multicast operator has $\mathscr{M}$ clients. We represent the *URL-profile* of each client-organization i with a set A_i:

$$A_i = \{a_{i,j} | j = 1, \ldots, n_i\}, \qquad i = 1, \ldots, \mathscr{M} \tag{1}$$

where $a_{i,j}, j = 1, \ldots, n_i$ correspond to "popular" URLs in the user community of organization i. Notably, different clients may have profiles differing as widely as the interests of an ISP clientele in Cyprus and a regional network user-base in India; that is, they may differ both in terms of their size (n_i) and content ($a_{i,j}$'s). In practice, the URLs of an A_i-set can be extracted from the URL-traffic captured by the institutional cache of i.

Objective. Based on the contents of the A_i's, we want to compute a set $\mathscr{A}$ of URL addresses that the multicast operator will disseminate to its subscribers. $\mathscr{A}$ is called the *multicast profile*and is defined as follows:

$$\mathscr{A} = \{\alpha_k | k = 1, \ldots, N\} \tag{2}$$

where α_k's are the URLs disseminated to the satellite-caching subscribers.

The multicast profile should comply with two fundamental conditions. First, the elements of $\mathscr{A}$ should be chosen amongst the elements of the A_i's, so that:

$$\mathscr{A} \subseteq \bigcup_{i=1}^{\mathscr{M}} A_i. \tag{3}$$

Secondly, $\mathscr{A}$ should provide some "cover" to all A_i's, so that:

$$A_i \cap \mathscr{A} \neq \varnothing, \qquad \forall i \in \{1, \ldots, \mathscr{M}\}. \tag{4}$$

Note that the union of all A_i's would be an obvious choice for $\mathscr{A}$, as it satisfies conditions (3) and (4). Nevertheless, due to cost considerations we assume that the cardinality of the multicast profile should be much smaller than the cardinality of the union of A_i's, i.e.:

$$\|\mathscr{A}\| \ll \left\| \bigcup_{i=1}^{\mathscr{M}} A_i \right\|. \tag{5}$$

The required multicast profile should possess a certain level of "similarity" between the multicast profile and all client profiles (A_i's). To gauge this similarity, we define two metrics that can be used to assess the relevance between two URL-profiles.

Definition 1 (Resemblance). Let A and B be two URL-profiles with: $A = \{a_i | i = 1, \ldots, n_A\}$, and $B = \{b_i | i = 1, \ldots, n_B\}$, where the a_i's and b_i's correspond to URL addresses. Then, the resemblance between A and B is defined as follows:

$$\mathrm{res}(A, B) = \frac{\|A \cap B\|}{\|A \cup B\|}.$$

In practice, the resemblance of two profiles A and B represents the portion of the overall pool of elements of A and B belonging to both A and B. We can easily prove the following properties for resemblance: $0 \leq \mathrm{res}(A, B) \leq 1$, $\mathrm{res}(A, A) = 1$, $\mathrm{res}(A, B) = \mathrm{res}(B, A)$, and if $A \cap B = \varnothing$ then $\mathrm{res}(A, B) = 0$.

Definition 2 (Coverage). Let A and B be two URL-profiles with: $A = \{a_i | i = 1, \ldots, n_A\}$, and $B = \{b_i | i = 1, \ldots, n_B\}$, where the a_i's and b_i's correspond to URL addresses. Then, the coverage of set A by set B is defined as follows:

$$\mathrm{cov}(A, B) = \frac{\|A \cap B\|}{\|A\|}.$$

In practice, the coverage of profile A by a set B represents the percentage of A's elements that belong to B. We can easily prove a number of basic properties for coverage: $0 \leq$

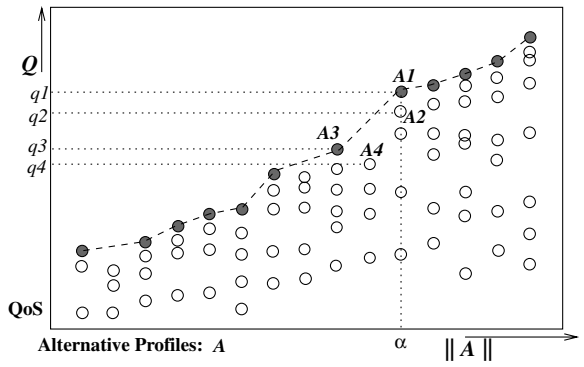

Fig. 3. A negotiation framework for satellite-caching.

$\mathrm{cov}(A,B) \leq 1$, $\mathrm{cov}(A,A) = 1$, $\mathrm{cov}(A,B) \neq \mathrm{cov}(B,A)$, and, if $A \cap B = \varnothing$, then $\mathrm{cov}(A,B) = 0$.

3.2. Pricing and quality-of-service models for Web multicasting

To select the URLs of the multicast profile $\mathscr{A}$, the multicast operator should aim at satisfying the utility requirements of all clients that adopt the satellite-based Web-content-dissemination service. The formalization of utility, however, depends, among other things, upon the pricing model agreed between the multicast operator and its clients. Here, we suggest two simple pricing models and explore how their adoption affects client utility and the calculation of $\mathscr{A}$ from A_i's.

Subscription-based pricing. To receive the Web multicasting service, clients of the content-distributor pay a fixed, monthly subscription fee covering leased satellite equipment and the periodic data feed.

Usage-based pricing. To receive the Web multicasting service, clients of the content-distributor pay a standard fee, covering leased satellite equipment, and a monthly fee proportional to the amount of bytes they receive from the satellite.

In both models, it is assumed that each client-organization i has adequate storage capacity for storing the broadcast content in its institutional cache. Furthermore, the institutional cache of i can discard content not deemed of interest to its user-base, i.e. not belonging to A_i.

Under subscription-based pricing, each client achieves optimal utility when receiving a selection of URLs that provide a maximal coverage of its profile (A_i); in other words, the client seeks the maximization of $\mathrm{cov}(A_i,\mathscr{A})$. Under Usage-based pricing, each client seeks to minimize the amount of useless information received and charged, i.e. $\mathscr{A} - A_i$, in addition to maximizing the coverage of its profile. This is equivalent to maximizing $\mathrm{res}(A_i,\mathscr{A})$.

These considerations dictate the client's perception about the quality of the proposed service. Therefore, we model the Quality-of-Service (QoS) offered by the multicast operator as follows.

Quality-of-Service. The QoS offered by a multicast operator to its client i is represented as a function $Q(\tau,A_i,\mathscr{A})$, where A_i is the URL profile of i, $\mathscr{A}$ is the multicast profile disseminated by the operator, and τ is the pricing model agreed between the operator and its clients. For Usage and Subscription-based pricing, Q is defined as follows:

$$Q(\tau,A_i,\mathscr{A}) = \begin{cases} \mathrm{res}(A_i,\mathscr{A}), & \text{where } \tau = \text{Usage-based pricing} \\ \mathrm{cov}(A_i,\mathscr{A}), & \text{where } \tau = \text{Subscription-based pricing} \end{cases} \tag{6}$$

From Eq. (6), we can easily see that $0 \leq Q(\tau,A_i,\mathscr{A}) \leq 1$.

3.2.1. Negotiating Web-multicasting services

The models presented in the previous section enable the satellite-operator to establish a framework of negotiation with its clients about the provision of the satellite-caching service. This framework entails three dimensions:

- the definition of the service provided, which we model by the multicast profile $\mathscr{A}$;
- the QoS, which we model according to definition (6);
- the price tag paid by a particular client for a given service and service-quality.

In an ideal situation, the operator and each client "negotiate" in order to reach a service agreement: following the collection of URL-profiles A_i from the clients, the satellite-operator calculates a number of alternative service-provisions in terms of alternative multicast-profiles $\mathscr{A}$. Each alternative multicast profile corresponds to a different QoS and is offered at a cost determined according to the pre-agreed pricing scheme.

Fig. 3 represents the space of alternative multicast profiles proposed by the satellite-operator to some client i. Proposed profiles are represented as circular points in a two-dimensional space: the horizontal dimension corresponds to the cardinality of multicast profiles whereas the vertical dimension corresponds to their respective QoS values. Notably, the operator could propose to its client a number of different multicast profiles with identical profile size but with different QoS values. For instance, in Fig. 3, profiles $\mathscr{A}_1$ and $\mathscr{A}_2$ have the same cardinality α; $\mathscr{A}_1$, however, offers an improved QoS over $\mathscr{A}_2$ since $q_1 > q_2$.

Each multicast profile is offered by the satellite-operator at a particular price. We assume that, under the pricing schemes introduced earlier, multicast profiles of the same cardinality $\|\mathscr{A}\|$ have the same cost; furthermore, that the more URLs are broadcast via the satellite, the higher the cost of the satellite service is. In other words, we make the following conjecture.

Conjecture 1. *For any two multicast profiles $\mathscr{A}$ and $\mathscr{B}$*

proposed by the satellite-operator to its clients, if $\|\mathscr{A}\| \leq \|\mathscr{B}\|$ *then* $price(\mathscr{A}) \leq price(\mathscr{B})$.

In summary, a client can choose among a set of triplets that define the satellite-caching service in terms of a proposed multicast profile, its quality, and its price. It is up to the client to agree upon the particular service deemed satisfactory. Taking into account the remarks above, it is not difficult to see that from the range of proposed multicast profiles of Fig. 3, a client is expected to negotiate for the "purchase" of only a small subset of profiles that we call *candidate profiles* and are marked as dark circles. The client has no reason to consider other profiles: for instance, profile $\mathscr{A}_2$ would be rejected since $\mathscr{A}_1$ offers a better QoS ($q_1 > q_2$) at the same price. Moreover, $\mathscr{A}_4$ would be rejected because profile $\mathscr{A}_3$ offers a better QoS ($q_3 > q_4$) at a price that is no worse than $\mathscr{A}_4$'s (since $\|\mathscr{A}_3\| < \|\mathscr{A}_4\|$). Candidate profiles are defined formally as follows.

Definition 3 (Candidate Profile). A multicast profile $\mathscr{A}$, proposed by a satellite operator to some client i, is called *candidate profile* if and only if, for any other proposed profile $\mathscr{B}$ such that $\|\mathscr{B}\| < \|\mathscr{A}\|$, it is: $Q(\tau, A_i, \mathscr{B}) < Q(\tau, A_i, \mathscr{A})$.

With these remarks in mind, it is not difficult to establish the following conjecture and prove Lemma 1.

Conjecture 2. *Among the range of multicast profiles that are proposed by a satellite operator to some client, the client will be willing to consider for purchase only candidate profiles.*

Lemma 1. *For a client i, the QoS of candidate profiles is monotonically increasing with respect to the candidate-profiles' cardinality. In other words, for any two candidate profiles* $\mathscr{A}$ *and* $\mathscr{B}$ *such that* $\|\mathscr{A}\| < \|\mathscr{B}\|$, *it is:* $Q(\tau, A_i, \mathscr{A}) < Q(\tau, A_i, \mathscr{B})$.

Proof. By contradiction, directly from Definition 3 and Conjecture 2.

3.2.2. Service configuration through QoS-guarantees

It is impractical to run separate, automated negotiations between the operator and its clients, each time the operator has to construct a multicast profile. Such an approach would require significant computation and communication resources and might not result to a single multicast profile satisfying all clients. Therefore, to make things simpler, the multicast operator can incorporate client considerations in a *service contract* proposed to potential clients. According to this contract, the satellite operator undertakes the responsibility of continuously broadcasting a *candidate multicast profile* that provides all clients with a minimum, guaranteed QoS level. This *QoS-guarantee* is offered to each client *i*through a *quality factor q*, which is accepted by both sides in the service contract. The quality factor defines the minimum guaranteed QoS level offered by the operator to all clients, through the following inequality:

$$Q(\tau, A_i, \mathscr{A}) \geq q \tag{7}$$

The utility requirements of the clients are accommodated in this contract through the quality factor q. Under such a scheme we can prove the following theorem.

Theorem 1. *Let* $\mathscr{A}_s$ *and* $\mathscr{A}_u$ *be two candidate multicast profiles of minimum cardinality that provide all clients with the QoS guarantee q under Subscription and Usage-based pricing, respectively. Then:* $\|\mathscr{A}_s\| \leq \|\mathscr{A}_u\|$.

Proof (by contradiction). We assume that:

$$\|\mathscr{A}_s\| > \|\mathscr{A}_u\| \tag{8}$$

given that $\mathscr{A}_s$ is a *minimum-cardinality* candidate profile under Subscription-based pricing, for any other candidate profile $\mathscr{B}$ with cardinality less than $\|\mathscr{A}_s\|$, there would be at least one client for which the QoS provided by $\mathscr{B}$ would be less than q, under Subscription-based pricing. This remark holds for $\mathscr{A}_u$ as well, according to our assumption (8). Therefore:

$$\exists j : \mathrm{cov}(A_j, \mathscr{A}_u) < q \tag{9}$$

From the definition of $\mathscr{A}_u$ and Eq. (6), however, it is:

$$\forall i, \mathrm{res}(A_i, \mathscr{A}_u) \geq q \tag{10}$$

Furthermore, from Definitions 1 and 2 of *Resemblance and Coverage*, we can easily see that:

$$\forall i, \mathrm{cov}(A_i, \mathscr{A}_u) \geq \mathrm{res}(A_i, \mathscr{A}_u) \tag{11}$$

Hence:

$$(10), (11) \Rightarrow \forall i, \mathrm{cov}(A_i, \mathscr{A}_u) \geq q,$$

which is a direct contradiction to inequality (9). Consequently, assumption (8) is wrong and therefore we conclude that $\|\mathscr{A}_s\| \leq \|\mathscr{A}_u\|$. □

What this theorem shows, in combination with Conjecture 1, is that if the satellite-operator and its clients accept the negotiation scheme presented earlier, a given level of the QoS-guarantee can be established under Subscription-based pricing at a *price at least as low* as under Usage-based pricing.

Besides the satisfaction of client-utility, however, the multicast operator is expected to pursue the maximization of profit it receives from the deployment of the Web-multicasting service. Under Subscription-based pricing, the operator's "income" is constant for a given number of client organizations. Therefore, we can assume that the operator seeks to minimize its collection and distribution costs in its selection of multicast content, while at the same time maintaining the QoS-guarantee agreed with its customers. We model the operator's costs with $\gamma \times \|\mathscr{A}\|$, a value

Table 1
Definitions and assumptions

Modeling basic elements of the Web-content-multicast service		
Service description	$\mathscr{A}$	
Operator cost-model	$\gamma \times \|\mathscr{A}\|,\ \|\mathscr{A}\| \ll \left\|\bigcup_{i=1}^{\mathscr{M}} A_i\right\|$	
Service requirements	$A_i,\ i = 1, \ldots, \mathscr{M}$	
Pricing models	*Subscription-based*	*Usage-based*
QoS model	$\mathrm{cov}(A_i, \mathscr{A})$	$\mathrm{res}(A_i, \mathscr{A})$
QoS-gurantee	$q \le \mathrm{cov}(A_i, \mathscr{A}),\ \forall i$	$q \le \mathrm{res}(A_i, \mathscr{A}),\ \forall i$

proportional to the total number of Web-objects disseminated, i.e. to the cardinality of $\mathscr{A}$. It should be noted that modeling distributor's costs proportionately to $\|\mathscr{A}\|$ is only an approximation as this does not take into account the *byte size* of objects.

The operator's income and benefits are proportional to $\|\mathscr{A}\|$, under Usage-based pricing. Consequently, we assume that the multicast operator seeks to send more content when selecting its multicast profile $\mathscr{A}$, that is to increase $\|\mathscr{A}\|$. Nevertheless, $\|\mathscr{A}\|$ cannot be increased up to $\left\|\bigcup_{i=1}^{\mathscr{M}} A_i\right\|$; in most cases, such an increase could violate the QoS-guarantee described by definition (6) and inequality (7), and/or exhaust storage and networking resources of the operator.

It should be noted that the examination of the Web-multicasting-service profitability for a varied number of clients is beyond the scope of this paper. Instead, we are interested in establishing the constraints placed upon the selection of $\mathscr{A}$ for a given number of clients ($\mathscr{M}$) and for the proposed formulations of client utility and operator profitability.

4. The complexity of Web-content selection

The simultaneous provision of the QoS-guarantee to all clients of the multicast operator influences the selection of the multicast profile according to the following constraint:

$$\min_{i=1,\ldots,\mathscr{M}} \{Q(\tau, A_i, \mathscr{A})\} \ge q. \tag{12}$$

Table 1 summarizes the definitions and basic assumptions we introduced to formalize the content-selection problem for Web multicasting. Here we focus on the solution of this problem under Subscription-based pricing, which is the scheme of choice in emerging WWW-multicasting services [21]. We propose the following formalization of the Web-content selection problem under Subscription-based pricing (alternatively, Web-content selection can be easily defined as an Integer Linear Programming problem [13]).

Web-content selection for Satellite Multicasting. For a multicast operator with $\mathscr{M}$ clients, find a multicast profile $\mathscr{A}$ with minimum cardinality $\|\mathscr{A}\|$ such that:

$$\mathscr{A} \subseteq \bigcup_{i=1}^{\mathscr{M}} A_i,$$

and

$$\mathrm{cov}(A_i, \mathscr{A}) \ge q, \qquad \forall i \in \{1, \ldots, \mathscr{M}\},\ 0 < q \le 1 \tag{13}$$

where A_i, $i = 1, \ldots, \mathscr{M}$ are the URL profiles representing the Web content requirements of the operator's clients, and q is a quality factor representing the QoS-guarantee agreed between these clients and the multicast operator.

According to the definition above, it is clear that Web-content selection is a *combinatorial optimization problem* [26]. Looking at the *recognition version* of this problem, it is not difficult to show that Web-content selection for Satellite Multicasting is NP-complete.

Theorem 2. *Given a set of URL profiles A_i, $i = 1, \ldots, \mathscr{M}$, a real number q, $0 < q \le 1$, and a positive integer δ[1], the problem of finding a set $\mathscr{A}$ such that:*

$$\mathscr{A} \subseteq \bigcup_{i=1}^{\mathscr{M}} A_i,$$

$$\frac{\|A_i \cap \mathscr{A}\|}{\|A_i\|} \ge q, \qquad \forall i \in \{1, \ldots, \mathscr{M}\}, \tag{14}$$

and

$$\|\mathscr{A}\| \le \delta,$$

is NP-complete.

Proof (by restriction). Web-content selection belongs obviously to NP. To prove that it is NP-complete, it suffices to show that it contains a known NP-complete problem as a special case. It is straightforward to do so with *Hitting Set* [17]. Given a collection C of subsets of a set S, and a positive integer K, Hitting Set asks if there exists a subset S' of S with $\|S\| \le K$, such that S' contains at least one element from each subset in C, i.e. $\|S \cap S'\| \ge 1$.

Let us consider instances of Web-content selection with q such that:

$$\max_{i=1,\ldots,\mathscr{M}} \frac{1}{\|A_i\|} \le q \Rightarrow q\|A_i\| \ge 1, \quad \forall i.$$

For this range of q's we can easily see that the satisfaction of constraint (14) is equivalent to $\|A_i \cap \mathscr{A}\| \ge 1$. Therefore, this restricted version of Web-content Selection is equivalent to the Hitting Set problem, with

$$S = \bigcup_{i=1,\ldots,\mathscr{M}} A_i,$$

$$C = \{A_i, i = 1, \ldots, \mathscr{M}\}, \text{ and } K = \delta.$$

The solution of Web-content Selection provides a set $\mathscr{A}$ such that $\|\mathscr{A}\| \le \delta$ and $\|A_i \cap \mathscr{A}\| \ge 1$. Therefore, $\mathscr{A}$ contains at least one common element from each subset in C, which means that it is the solution of the corresponding Hitting Set problem. □

Table 2
An approximation algorithm for Web-content selection

```
𝒜 = ∪_i A_i
keep = ∅
while (𝒜 ≠ keep) {
(1) select element a ∈ 𝒜, which maximizes ∑_i cov(A_i, 𝒜 − {a})
   if cov(A_i, 𝒜 − {a}) ≥ q, ∀i
      remove element a from 𝒜
   else
      keep = keep ∪ {a}
}
```

5. An approximation algorithm for content selection

We develop an approximation algorithm to find the required multicast profile $\mathscr{A}$. The algorithm implements the following requirement: find the set $\mathscr{A}$ with minimum cardinality, such that, if any of its elements is removed, the coverage for at least one of the sets A_i will fall below the preset quality bound q. A description of the algorithm follows.

We start with a profile $\mathscr{A}$ containing all the elements from every set A_i, i.e. $\mathscr{A} = \cup_i A_i$. We remove temporarily the first element, a, from set $\mathscr{A}$ and compute the new coverages $\text{cov}(A_i,\mathscr{A})$. Summing them we obtain the cumulative coverage obtained after removal of element a. Then, we place element a back into set $\mathscr{A}$ and repeat the process, removing in turn each element in set $\mathscr{A}$, and computing the total coverage resulted by each element's removal.

We choose for deletion the element that, if removed, will produce the maximum cumulative coverage — since the element that maximizes the total coverage, minimizes the loss by its removal.

The above process describes one iteration of the algorithm. The entire process is repeated until the profile set $\mathscr{A}$ has the minimum number of elements. This set is minimal in the sense that, by removing any one of its remaining elements, the coverage of at least one of the sets A_i will fall below the acceptable quality bound.

In order to control the number of iterations of the process, we introduce a set called *keep* where we add elements whose removal from $\mathscr{A}$ would cause the coverage of at least one of the sets A_i to fall below the quality bound. The condition that controls the iterations of the algorithm is to repeat while there are still elements in $\mathscr{A}$ that are *not* in *keep*. In other words, we continue the iterations while there are still elements in $\mathscr{A}$ that we could dispose without sacrificing the quality guaranteed for each set A_i. When we are left with the two sets, $\mathscr{A}$ and *keep*, containing the same elements, we can no longer remove any more elements and the process stops. The resulting set $\mathscr{A}$ is minimal and satisfies $\text{cov}(A_i, \mathscr{A}) \geq q$, $\forall i$. The algorithm is given in Table 2.

5.1. Analysis of the approximation algorithm

As it can be seen from the outline of the algorithm in Table 2, some set operations occur very frequently. These are: deletion, insertion and membership of an element in a set. Considering also the size of these sets, it was considered necessary to chose a set implementation by which the above set operations could be completed in constant time. The bit-vector (boolean array) implementation was therefore chosen, by which the ith bit is true (i.e. 1), if i is an element of the set.

The worst-case running time of the algorithm is basically the running time of the while loop. Statement (1), inside the loop, hides a nested for-loop which is analyzed as follows. The outer for-loop (not shown explicitly) iterates over all elements in $\mathscr{A}$ (the union) and the inner loop iterates over all sets A_i. Assuming we have at most N elements in $\mathscr{A}$ and at most M sets, then the worst case complexity of line (1) is $O(MN)$.

Line (1) is inside a while loop. It is easy to see that this loop iterates at most N times, i.e. as many times as the elements in $\mathscr{A}$. To see this consider the two extreme cases:

1. We need to keep the entire union. A situation which could arise if the quality is set to 1, i.e. none of the elements in $\mathscr{A}$ can be removed without violating the quality constraints. Then the while loop iterates until the set *keep*(initially empty) becomes equal to $\mathscr{A}$, i.e. after N iterations (as many as the elements in $\mathscr{A}$).
2. We need to remove all the elements from $\mathscr{A}$. A situation which could arise when quality is 0, i.e. none of the elements in $\mathscr{A}$ is necessary to satisfy the quality constraints. In such a case, every time round the while loop, we will remove one element until $\mathscr{A}$ becomes equal to set *keep*, i.e. the empty set. This happens after N iterations.

Since the while loop, iterates N times at worst, and line (1), which is the most complex inside the loop is $O(MN)$, then the entire algorithm is $O(MN^2)$.

5.2. Experimental study

To assess our Web-content selection algorithm we implemented it and ran numerous experiments. As we did not have access to the logs of established satellite-caching services, we used sets comprised of discrete values (integer numbers) produced by uniform and gaussian random number generators. We assume that each distinct random number corresponds to a different URL-address. Had we have access to satellite-caching logs, it would not have been difficult to map different URLs to integer numbers and conduct similar experiments. The experimental results reported in this section are representative of the

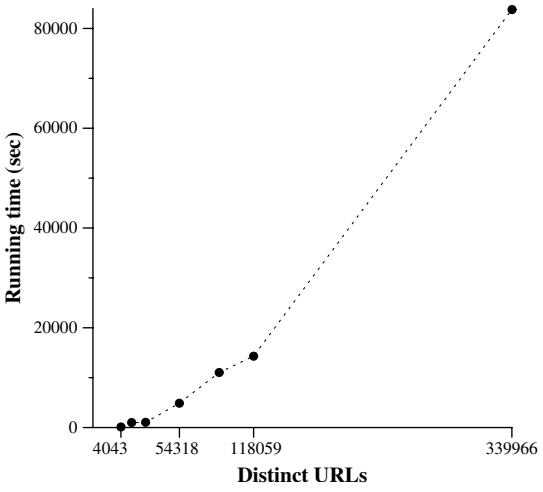

Fig. 4. CPU time vs. initial profile size.

suite of experimental data that we gathered. Through our experimental study we are seeking to:

1. Produce a rough estimate for the running time of our approximation algorithm on a variety of input sizes. This estimate can guide the selection of processing power necessary to establish satellite-schemes and may probe further work on more efficient heuristics and/or the employment of parallelization techniques.
2. Explore the effects that the choice of the quality factor has upon service characteristics, which determine the QoS delivered and the operator's cost.

Measurements from our experiments are shown in Figs. 4–6.

In Fig. 4, we plot the CPU time versus the number of distinct URLs. The number of distinct URLs represents the size of the initial profile, i.e $\|\bigcup_i A_i\|$. The CPU times reported were taken from experiments where we computed the multicast profiles for five subscribers (i.e. $\mathscr{M} = 5$) and

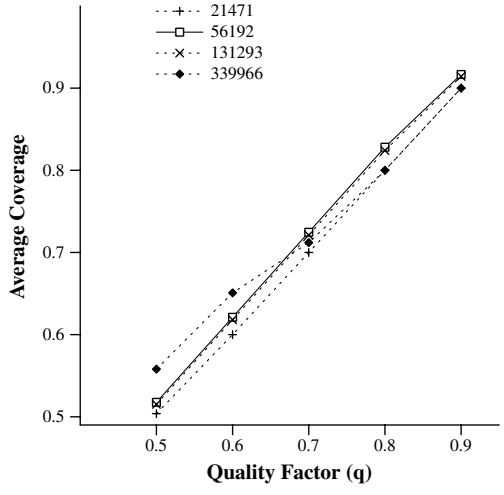

Fig. 5. Average coverage vs. quality factor.

for a quality factor of 0.75. Time measurements were taken on a Sun A26 Enterprize 250 Server, with an UltraSparc-II processor with 400 MHz clock and 512 MB of main memory. The characteristics of our input sets and the respective measurements of CPU times are summarized in Table 3.

In Fig. 5 we plot the relationship between the quality factor q and *average coverage*. The average coverage gives an estimate of the coverage of client profiles by the resulting multicast profile $\mathscr{A}$, and is defined as follows:

$$\text{average coverage} = \frac{\sum_{i=1}^{\mathscr{M}} \text{cov}(A_i, A)}{\mathscr{M}} \tag{15}$$

Fig. 5 shows four different graphs corresponding to four experiments with different client profiles (A_i) and initial-profile sizes ($\|\bigcup_i A_i\|$). As we can see from these graphs, average coverage increases linearly with q. This is expected, since q is a lower bound on the coverage. It should be noted that diagrams of this kind could be used by satellite-operators for exploring alternative service schemes that offer different QoS-guarantees to different customers or groups of customers.

In Fig. 6 we are associating the quality factor with the compression ratio, again for the four different input cases. The compression ratio is defined as the ratio of the size of the initial profile over the resulting multicast profile size:

$$\text{compression ratio} = \frac{\|\bigcup_i A_i\|}{\|\mathscr{A}\|} \tag{16}$$

Compression expresses the savings that the multicast operator achieves if it adopts the multicast profile computed by our algorithm instead of the union of all client profiles. All four graphs in the left diagram of Fig. 6 show an inverse relationship between compression and quality. This is due to the fact that an increased quality factor will result to an increase of the multicast profile size $\mathscr{A}$, since more elements from the initial profile, $\bigcup_i A_i$, will now have to be kept to preserve the increased quality. Therefore, an increase in the denominator of Eq. (16) will cause a decrease in the compression ratio. As the quality increases towards 1, all four graphs tend to the same compression ratio, 1. This is true for all four different initial profile sizes shown.

The right diagram of Fig. 6 displays the compression ratio versus the inverse quality factor ($1/q$). From this plot, we can see that $1/q$ is a *lower bound*of the compression ratio. Therefore, the quality factor accepted by the satellite operator and its clients, provides the operator with an *estimate* of its worst-case service costs, for the particular set of client profiles. Furthermore, this diagram provides evidence for the "quality" of our approximation algorithm since, for all of our experiments, this algorithm chooses multicast profiles $\mathscr{A}$ such that: $\|\mathscr{A}\| < q\|\bigcup_i A_i\|$.

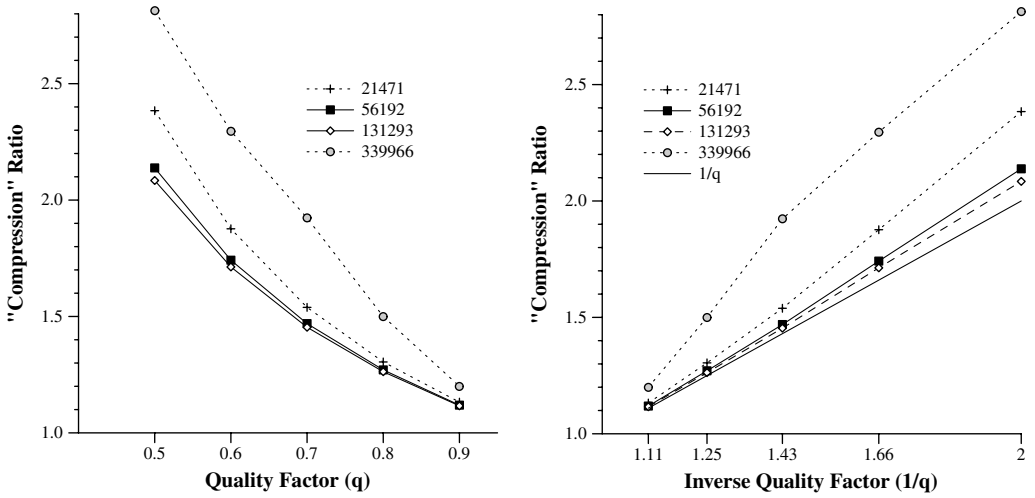

Fig. 6. Compression vs. quality.

6. Conclusions

In this paper we studied the problem of Web-content selection in the context of a periodic "push" scheme that uses satellite links to disseminate information to WWW-caches worldwide. This information dissemination takes place under a new service offered by satellite-network operators to subscriber ISPs around the world. Satellite-based dissemination is combined with hierarchical caching schemes deployed by the ISPs, providing prefetched Web-content to WWW-caching hierarchies of the ISPs. The main idea is to prefetch popular resources via satellite in order to avoid the overloading of already congested terrestrial Internet links.

To address the problem of Web-content selection, we have achieved the following:

- We introduced a novel theoretical framework that can be used to guide the selection of content for dissemination. This framework defines the notions of client Utility and QoS in the context of the satellite-caching service. Furthermore, it provides the satellite operator and its potential clients with a basis for negotiating the pricing of satellite-caching services.
- Based on our modeling, we proved that the multicast operator can guarantee, under Subscription-based pricing, a Qos at *least as good* as under Usage-based pricing, at the same or lower cost. This conclusion provides a basis for preferring the Subscription-based pricing scheme for satellite-caching services established upon the negotiation framework introduced here.
- Focusing on Subscription-based pricing (which is currently employed by satellite operators), we showed that Web-content selection can be formulated as a combinatorial optimization problem. Studying its complexity showed that it belongs to the class of NP-Complete problems.
- Next, we proposed an approximation algorithm to resolve Web-content selection in polynomial time $O(\mathcal{M}N^2)$, where $\mathcal{M}$ is the number of subscribers to the satellite-caching service and N represents the total number of distinct URLs requested by all subscribers.
- Finally, we implemented the algorithm and ran several tests on synthetic data, to gauge its validity and performance.

Table 3
An approximation algorithm for Web-content selection

Distinct URLs	CPU Time (in s)
4043	127
13 368	987.7
25 352	1054.61
54 318	4887.96
88 503	11029.57
11 8059	14312.38
33 9966	83789.91

Our experiments provide insights into a number of issues: First, the validity of the quality factor q, as a means for defining the satellite-caching service, is corroborated by the observation that q, not only does offer the QoS-guarantee to subscribers, but also provides the operator with a good estimate regarding the upper bound of its cost. Second, our implementation can sustain service scenarios that involve several multicasts per day. If, however, the total number of requested *distinct*URLs is very large (over half a million, approximately), running time becomes quite large for sustaining frequent multicasts per day, given that there is available bandwidth for such multicasts. This problem can be tackled either by the use of more computing resources, the adoption of parallelization, the development of linear-time complexity heuristics, or the adoption of algorithms that continuously adjust the multicast profile to ever-changing client profiles. Last, but not least, our experiments show that it makes sense to explore the merits of service schemes which are established upon different quality factors

for different subscribers and groups of subscribers. Such an exploration should also take into account models of client-profile resemblance and their effect on content selection.

Acknowledgements

The authors wish to thank Dr C. Makris of the Cyprus Telecommunications Authority, and Dr G. Paliouras of the National Center for Scientific Research "Democritos,"of Greece, for their helpful comments on earlier versions of this document.

References

[1] Broadcast Satellite Services. http://www.isp-sat.com.
[2] GVU's WWW User Surveys. http://www.gvu.gatech.edu/gvu/user_-surveys/.
[3] INTELSAT. http://www.intelsat.com/products/internet/atint.htm.
[4] Skycache. http://www.skycache.com.
[5] M. Abrams, C. Stanbridge, G. Abdulla, S. Williams, E. Fox, Caching proxies: limitations and potentials, in: Fourth International World Wide Web Conference, 1995. http://www.w3.org./Conferences/WWW4/.
[6] D. Aksoy, M. Altinel, R. Bose, U. Cetintemel, M.J. Franklin, J. Wang, S.B. Zdonik, A framework for scalable dissemination-based systems, in: Proceedings of the 1997 ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages & Applications, OOPSLA'97, 1997, pp. 94–105, ACM.
[7] D. Aksoy, M. Altinel, R. Bose, U. Cetintemel, M.J. Franklin, J. Wang, S.B. Zdonik, Research in data broadcast and dissemination, in: Proceedings of the First International Conference on Advanced Multimedia Content Processing, AMCP'98, Lecture Notes in Computer Science, Springer, 1999, pp. 194–207.
[8] D. Aksoy, M. Franklin, Scheduling for large-scale on-demand data broadcasting, in: Proceedings of the 1998 IEEE Infocom Conference, IEEE, 1998.
[9] M.F. Arlitt, C.L. Williamson, Web server workload characterization: the search for invariants, in: Proceedings of the Sigmetrics Conference on Measurement and Modeling of Computer Systems, ACM, 1996, pp. 126–137.
[10] A. Bestavros, R. Carter, M. Crovella, C. Cunha, A. Heddaya, S. Mirdal, Application level document caching in the Internet, in: Proceeding of IEEE SDNE' 96: The Second International Workshop on Services in Distributed and Networked Environments (IEEE, 1995). http://www.cs.bu.edu/(best/res/papers/sdne95.ps.
[11] P. Cao, S. Irani, Cost-Aware WWW proxy caching algorithms, in: Proceedings of the USENIX Symposium on Internet Technology and Systems, 1997, pp. 193–206.
[12] C. Courcoubetis, V.A. Siris, Managing and pricing service level agreements for differentiated services, in: Proceedings of the Seventh IEEE/IFIP International Workshop on Quality of Service, IWQoS'99, 1999.
[13] M.D. Dikaiakos, Utility and Quality-of-Service Models for Periodic Prefetching of WWW Resources. Technical Report TR-99-7b, Department of Computer Science, University of Cyprus, January 2000.
[14] M.D. Dikaiakos, D. Gunopoulos, FIGI: the architecture of an Internet-based financial information gathering infrastructure, in: Proceedings of the International Workshop on Advanced Issues of E-Commerce and Web-based Information Systems, IEEE-Computer Society, 1999, pp. 91–94.
[15] L. Fan, P. Cao, W. Lin, Q. Jacobson, Web prefetching between low-bandwidth clients and proxies: potential and performance, in: Proceedings of the Sigmetrics Conference on Measurement and Modeling of Computer Systems, 1999, pp. 178–187.
[16] M.J. Franklin, S.B. Zdonik, Data in your face: push technology in perspective, in: Proceedings ACM SIGMOD International Conference on Management of Data, Seattle, Washington, DC, USA, ACM Press, June 1998, pp. 183–194.
[17] M. Garey, D. Johnson,, Computers and Intractability, Freeman, New York, 1979.
[18] A. Gupta, D. Stahl, A. Whinston, The economics of network management, Communications of the ACM 42 (9) (1999) 57–63.
[19] D.L. Hoffman, W.D. Kalsbeek, T.P. Novak, Internet and Web use in the US, Communications of the ACM 39 (12) (1996) 36–46.
[20] B. Liu, G. Abdulla, T. Johnson, E. Fox, Web response time and proxy caching, in: WebNet'98, 1998, AACE.
[21] C. Makris, Personal Communication. Cyprus Telecommunications Authority, 1999.
[22] C. Maltzahn, K.J. Richardson, Performance issues of enterprise level Web proxies, in: Proceedings of the Sigmetrics Conference on Measurement and Modeling of Computer Systems, ACM, 1997, pp. 13–23.
[23] A. Odlyzko, The economics of the Internet: utility, utilization, pricing, and Quality of Service. Technical report, AT&T Labs-Research, 1998.
[24] A. Odlyzko, Paris metro pricing for the Internet, in: Proceedings of the ACM Conference on Electronic Commerce, EC-99, 1999.
[25] C.H. Papadimitriou, S. Ramanathan, P.V. Rangan, Multimedia information caching for personalized video-on-demand, Computer Communications 18 (3) (1995) 204–216.
[26] C.H. Papadimitriou, K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, Prentice-Hall, Englewood Cliffs, NJ, 1982.
[27] A. Pitsillides, G. Samaras, M.D. Dikaiakos, E. Christodoulou, DITIS: collaborative virtual medical team for the home healthcare of cancer patients, in: Proceedings of the Conference on the Information Society and Telematics Applications, European Commission, 1999.
[28] P. Rodriguez, E.W. Biersack, K.W. Ross, Automated Delivery of Web Documents through a Caching Infrastructure. Submitted for publication. http://www.eurecom.fr/(ross/MMNetLab.htm, 1999.
[29] P. Rodriguez, E.W. Biersack, Bringing the Web to the network edge: large caches and satellite distribution, in: Proceedings of WOSBIS'98, ACM/IEEE MobiCom Workshop on Satellite-based Information Services, 1998.
[30] P. Rodriguez, E.W. Biersack, Prefetching Web documents into large caches using a satellite distribution. ACM Special Issue of the Journal on Special Topics in Mobile Networking and Applications (MONET), in press, 2000. http://www.eurecom.fr/(rodrigue/.
[31] A. Rousskov, V. Soloviev, On performance of caching proxies, in: Proceedings of the Sigmetrics Conference on Measurement and Modeling of Computer Systems, ACM, 1998, pp. 272–273.
[32] G. Samaras, M.D. Dikaiakos, C. Spyrou, A. Liverdos, Mobile agent platforms for Web-databases: a qualitative and quantitative assessment, in: Proceedings of the Joint Symposium ASA/MA'99. First International Symposium on Agent Systems and Applications, ASA'99, Third International Symposium on Mobile Agents, MA'99, IEEE-Computer Society, 1999, pp. 50–64.
[33] C.J. Su, L. Tassiulas, V.J. Tsotras, Broadcast scheduling for information distribution. ACM/Baltzer, Journal of Wireless Networks Technology and Systems 5 (2) (1999) 137–147.
[34] R. Tewari, M. Dahlin, H.M. Vin, J. Kay, Design Considerations for Distributed Caching on the Internet. Technical Report UTCS TR98-04, Department of Computer Sciences, University of Texas at Austin, 1998.
[35] D. Wessels, K. Claffy, Evolution of the NLANR Cache Hierarchy: Global Configuration Challenges. Technical report, NLANR, 1996. http://www.nlanr.net/Papers/Cache96/.