Cooperative Caching in Space Information Networks

Anders Fongen
Norwegian Defence University College (FHS)
Lillehammer, Norway
email: anders@fongen.no

Lars Landmark
Norwegian Defence Research Establishment (FFI)
Kjeller, Norway
email: Lars.Landmark@ffi.no

***Abstract*—Members of a Low Earth Orbit (LEO) satellite group need to coordinate their activities in order to improve the quality and timeliness of the services provided to terrestrial clients. As a report from a work in progress, we explore different coordination patterns on a distributed cache and study the hit rate, the training ratio and the space requirements. The query model employs a Scale Free Distribution as this is proven to more accurately model the request patterns to many Internet services.**

***Keywords*—LEO satellites; cooperative caching; space information networks; Scale free distribution**

I. INTRODUCTION

A Space Information Network (SIN) is an information system located in space [1]. The concept is an evolution of satellite networks as they are known from the 1960s to present day, where satellites transform from "radio mirrors" with plain wideband transponders, towards networks of interconnected satellites providing connectivity services based on store-and-forwarding of data packets. This evolution represents an *increasing system complexity* in the spacecrafts.

In order for a SIN to provide information and computational services in addition to connectivity services, the theory and methods of *distributed systems* become invaluable tools. The transition from connectivity services to computational services extends the state space of the service sessions, and the distribution and transfer of state, e.g., related to a handover operation, become important fields of study and interesting design problems [2]. Besides, the protection of the new service endpoints occurring in a SIN is essential and requires key and certificate management in the SIN structure [3].

A SIN offering storage services is likely to offer this service as a mutable secondary/slave storage replica, since a data backup needs to exist somewhere. Among many interesting research questions, the problems related to *distributed cache management* are the focus of this paper and will be analyzed and presented in detail.

The performance of *Discovery Services* is an important factors in the efficiency of an information system. These services offer the retrieval, caching and distribution of essential information like X.509 Public Key Certificates, Domain Name System (DNS) name/address pairs, Uniform Resource Locators (URLs), link topology information, etc. Optimal performance of discovery services requires a well balanced and tuned cooperative caching system in the SIN in order to support the relying information services efficiently. E.g., a slow DNS service will hamper the performance of an otherwise well tuned Web service.

Which advantages can be achieved through the deployment of a SIN? Two main characteristics of the services distinguish a SIN from ordinary Internet services:

1) Global coverage for mobile clients,
2) Very low latency.

The round-trip time through a satellite at 500 km altitude can be as low as 3.3 milliseconds. Low latency is also one key property of 5G, which will enable new time sensitive cooperative applications like remote surgery, autonomous vehicles, etc.

An important choice in our SIN studies is to include the earth's population density into the analysis and resource planning. In particular for lower altitudes, the satellites will spend large fractions of their time over inhabited areas, mixed with shorter intervals of extremely high density. It is likely that the rate of incoming requests will follow a similar pattern. An appealing idea is to allow idle satellites to offload busy ones, since neighbouring satellites in the network can communicate through high speed inter-satellite links (even optical links).

For the remainder of this paper, the organization is as follows: Section II will present related research on this topic; a discussion on the design of a satellite constellation will follow in Section III. The design of satellite clusters for task distribution will be presented in Section IV, followed by a detailed discussion of the Scale Free Distribution principles in Section V. The experiment series, first based on an isolated cache and next in a satellite constellation, are presented in Sections VI and VII. Finally, some conclusive remarks are given in Section VIII.

II. RELATED RESEARCH

The term *Space Information Network* has been used to describe networks of satellites and high altitude aircrafts (drones, balloons) with different service levels. Existing satellite networks like Iridium and Starlink [4] offer only communication services, the latter on a very large scale and with high bandwidth. A number of authors have proposed "Cloud Computing in Space" through the addition of larger satellites with sufficient energy and computing resources for taking on these tasks [5][6].

The results presented in this position paper will not deal with technical details in the communication technology, but rather view the SIN as a distributed system which borrows its analysis and solutions from the field of distributed computing. The authors are not aware of other efforts to investigate cooperative caching mechanisms specifically for a SIN.

 ISBN: 978-1-61208-975-1

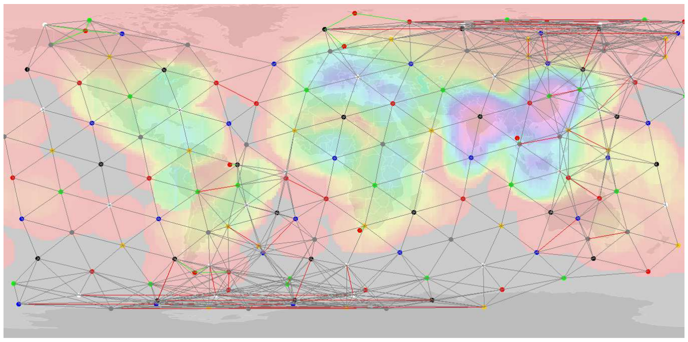

Figure 1. Screenshot from the satellite constellation model

Many retrieval and lookup operations are found to follow a so-called Scale Free Distribution (SFD). Queries for web pages, e-mail addresses, DNS names, etc., have been studied in this regard [7][8]. These results are important for our decision to apply SFD principles in our analyses and experimental design.

III. A CANDIDATE CONSTELLATION FOR STUDY

Satellite networks servicing civilian mobile clients using handheld equipment tend to operate in a LEO constellation. E.g., the orbit altitude of Iridium satellites is 781 km, which allows for lightweight ground terminals without the need for antenna deployment. The inclination of the orbit can be made so steep that the polar regions are fully covered, or given a lower angle to spend more time over the densely populated latitudes closer to the equator.

The choice of orbit altitude determines the diameter of the *footprint*, e.g., the circular region of the earth with potential connectivity, and the longest possible distance between the satellites which still allows for inter-satellite links and uninterrupted service for ground terminals. Simply stated, a lower orbit altitude reduces the design constraints on the ground terminals and provides higher communication capacity, but increases the cost due to the higher required number of spacecrafts.

For our SIN study, a software model has been made to study these trade-offs and to emulate the coordination activities between the satellites. The model also incorporates population density data which is readily available on the Internet [9].

Figure 1 shows a screenshot from the software model, containing 150 satellites with an orbit inclination of 75 degrees and an altitude of 500 km. The colors on the backdrop indicate the population density inside the footprint of a satellite in that position (contrary to the local density at that exact position). The colorization is considered to be a parameter for the estimation of the request rate received from ground surface clients. Other possible parameters, like local time and Internet penetration of the region, may be taken into regard at a later instance.

The satellites are given a color according to their *role*, which will be explained in Section IV. The lines between them indicate inter-satellite links and links to ground stations.

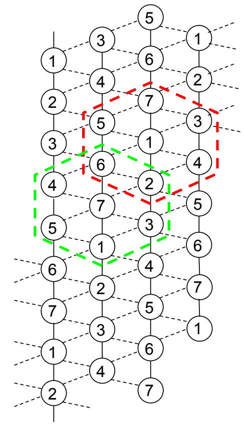

Figure 2. Patterns of relative positions given to satellite roles

Of special interest are links between satellites with the same color, because certain cache optimization techniques will be applied using those links.

In the Iridium system, the 66 satellites are divided into 6 orbits on different longitudes, but with the same inclination. The 6 orbits are separated with 30 degrees and consequently cover one hemisphere in the northbound direction, and the opposite hemisphere in southbound direction. The constellation avoids having the directions interleaved for reasons of handover time and Doppler shift.

Our candidate constellation chooses a similar arrangement, but rather puts the satellites into a spiral arrangement where trailing satellites are shifted eastward to compensate for the earth's rotation. This arrangement makes the row of oncoming satellites to follow the same track when observed from the earth's surface, which will preserve the connection quality across hand-overs.

IV. DISTRIBUTION OF WORKLOAD ACROSS SATELLITE CLUSTERS

Adjacent neighbors flying in the same directions are keeping company on a permanent basis and may form clusters for distributed processing of requests. The proposed constellation allows for fast and direct links to 6 neighbors (North, South, SE, SW, NE, NW) except for those on the "edge" orbit. Each satellite is given one of seven *roles* numbered 1-7 and they are given relative positions as shown in the pattern on Figure 2. For visualization purposes, the roles are represented using 7 different colors, as shown in Figure 1. The terms *roles* and *colors* are synonyms and will be used interchangeably.

Observe that every satellite is surrounded by the other 6 roles, and that one satellite also serves its role in 6 surrounding clusters. Clusters are not disjoint and every satellite forms the center position of a cluster. Also observe that for satellites at the edge of the constellation, their "missing" neighbor at one side can be found two hops away to the opposite side, through the NE or SW neighbor.

Given this pattern, tasks may be divided into 7 different sub-tasks, and any satellite in the constellation is in a center position of a cluster which is able to execute it. This satellite may further invoke the resources of its 6 neighbors for the purpose of the task. In the case of a distributed cache, the cache entries may be evenly distributed among these 7 satellites and queries may be delegated to the cache instance which is a candidate for that query value. A hash function modulus 7 computed over the cache entry key is used for this purpose in this experiment.

The motivation for this distributed approach to a caching service is that the total storage capacity increases 7 times, to the cost of a link hop for 6 out of 7 replicas. The performance improvement gained from this design will be evaluated in Sections VI and VII.

V. A SCALE-FREE DISTRIBUTION OF CACHE REQUESTS

For the evaluation of cache efficiency, a vocabulary of 20,000 words was built and its terms were selected as search terms according to a *Scale Free Distribution* (SFD), also known by the name Zipf's law [10]. If the search term is not found in the cache, it is added to the cache from an authorized source and the operation is counted as a cache miss (cm), otherwise a cache hit (ch). The performance of the cache is represented by the cache hit fraction of the total number of operations $(ch/(ch + cm))$.

The cache is initially empty, and will generate only cache misses from the beginning. As the cache content is built, its performance gradually improves. When a cache miss results in addition of a term to a full cache, the *Least Recently Used* (LRU) term is removed from the cache to make room for the new term.

The SFD predicts that the observed frequency of a query term is inversely proportional to its *rank* r. The relative frequency f of the term t with rank r is expressed as

$$f(t_r) = \frac{a}{r} \tag{1}$$

where the value of a is determined so that

$$\sum_{r=1}^{v} \frac{a}{r} = 1 \tag{2}$$

and v indicates the size of the vocabulary in use.

The rationale for preferring SFD over a Uniform Distribution (UD) is that the SFD has been found to provide a good model of different communication and distribution patterns: E-mail addresses, flight structure between airports, road traffic patterns, sexually transmitted diseases, etc. [11]. The suggestion that a small number of websites have a large portion of the traffic sounds reasonable, and the lookup operations in a DNS cache are shown to be a candidate for an SFD based model [8].

With a vocabulary of 20,000 entries, an SFD will cause one of the 105 highest ranked values to be selected 50% of the time, since the sum of their relative frequencies is 0.5. This indicates that a cache efficiency of 0.5 (meaning 50% hit rate) is theoretically obtainable with only 105 entries in the cache. On the other hand, maximum hit rate with the 1000 highest ranked terms in the cache is only 0.71. It is therefore expected that from an initially empty cache, the efficiency will rise much faster during the "training phase" than what will be observed using uniform query distribution.

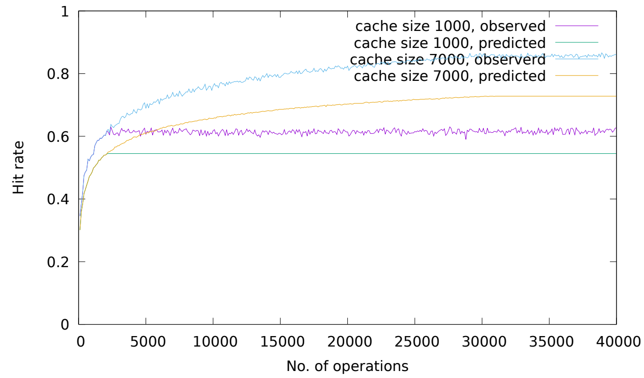

Figure 3. Experimental evaluation of local cache performance

For a cache miss to happen with search term t at a query operation when m number of entries are present in the cache, all these entries must contain terms different from t. Since t can take any value within the vocabulary, the following sum approximates the cache miss probability $p_{sfd}(cm)$

$$p_{sfd}(cm) = \sum_{r=1}^{v} (\frac{a}{r}(1 - \frac{a}{r})^m) \tag{3}$$

The prediction of cache miss probability with a uniform distribution of search terms can be similarly expressed as a unordered selection operation:

$$p_{ud}(cm) = \frac{\binom{v}{m}}{\binom{v-1}{m}} = \frac{v - m}{v} \tag{4}$$

For further examination, we will use the cache hit probability $p(ch)$

$$p(ch) = 1 - p(cm) \tag{5}$$

VI. ISOLATED CACHING EXPERIMENTS

Before looking at the cache performance inside a satellite constellation, the caches are studied in an isolated experimental environment. They expose the predicted behavior, in that they increase their efficiency during "training period" while the cache is filled up, after which only marginal improvements are observed. In Figure 3, the cache hit rate for 1000 and 7000 maximum number of entries is shown with the number of lookup operations along the x axis. The theoretical hit rates are also shown on the figure. They are somewhat lower than the observed numbers. The discrepancy is due to the ratio between vocabulary size and cache size, which Equation 3 assumes to be infinite.

Experimental efforts were made to optimize the performance of the cache through structural modifications, among which modification to the LRU-based replacement algorithm. If replacements increase the *average rank* of the cache entries, a higher hit rate was expected. It was indeed verified that a

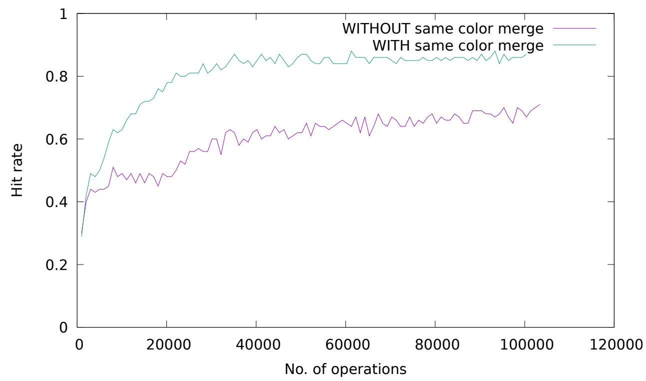

Figure 4. Cache performance inside the satellite constellation

cache with the 105 highest ranked terms obtains a 50% hit rate, but improving the average rank of the 1000 entries in a full cache did not yield significant results over the LRU-based algorithm. The highest possible efficiency of a 1000 element cache is 0.71, as that is the accumulated relative frequency of the 1000 highest ranked terms in the SFD distribution with 20,000 terms. The highest observed hit rate on a 1000 element cache under normal operation using LRU-based replacement algorithm was 0.63; for a 7000 element cache it was 0.86.

This first experiment shows results which are consistent with the theoretical calculations, due to the static data structures in use. In the next section, the more realistic environment of a SIN will be emulated for similar performance evaluations.

VII. Cache performance in a satellite constellation

The numbers observed in the experiment presented in Figure 3 serve as a baseline for a distributed and satellite based caching experiment, which will be described in this section.

The environment for a distributed cache residing in orbiting satellites is quite different from the experimental conditions present in the isolated experiment described in Section VI:

- The client does not interrogate the same cluster instance over time, but repeatedly uses different instances "trained" by different clients elsewhere on the planet. The query distribution will change over time and with the geographical region below.
- The cache is distributed over 7 satellites, each of which serves as a member of 7 different cache clusters and is jointly trained by each of them.
- The entire satellite fleet is supposed to hold fully trained caches first after a considerable number of query operations.

The prediction of cache performance under these conditions is likely a non-malleable problem, so a simulation result will be presented here. Seven ground terminals from different parts of the planet surface were configured to send term queries to their nearest satellite, which will contribute to the training of the satellite's cache and its immediate 6 neighbors. The total average hit rate was calculated and reported as a function of the total number of queries. Since we now evaluate distributed caches with 7 instances of 1000 entries each, we use the 7000 entries result from the isolated evaluation in Figure 3 as our baseline. The difference in performance will be the effect of the dynamic topology of the satellite infrastructure. The results from this series of experiments are presented as the purple line on Figure 4.

A. Same color merge

In a third series of experiments, an additional mechanism was added as satellites with the same color/role in the distribution pattern were allowed to merge the content of their caches during periods when they were able to connect. These periods occur when the satellites move across the polar regions, as well as when they meet in opposite directions at the east/west edges of the constellation.

The process of merging caches during an encounter of two same-color satellites will add all entries from one cache into the other, unless it is already there. Since no ranking information is preserved in the cache, they are read from the start, where the higher ranked (and frequently used) entries are to be found. All copied entries will be equally "recently used" since the "recency" of an entry is represented by its position in the linked structure, not by metadata.

The result of this caching technique in the satellite constellation model is shown as the green line in Figure 4. For the 100,000 number of operations shown, the same-color merging process appears to give a significant advantage. Longer experiment runs show that the gap between the two lines narrows down to an insignificant difference after 700,000 operations. Consequently, the merge process merely speeds up the training phase, rather than creating a permanent improvement. The highest hit rate observed is 0.87, which is considered equal to the isolated evaluation with its 0.86 value for best hit rate.

B. Persistent performance improvement

In a more realistic experiment, the ground stations would choose query terms from different vocabularies, reflecting the diversity in language and culture between the regions of the earth. The vocabulary of query terms from the same region will also change over time, reflecting the changing interests of the population. For a distributed cache to maintain its performance in the presence of constantly changing query term vocabularies, *training speed* becomes an important factor. For this reason, the same-color merge process presented in Section VII-A is more than a ephemeral advantage in the initial phase of operation, but a property expected to yield higher cache performance during the entire operation.

VIII. Conclusion

Despite some simplifying assumption that every client picks query terms from the same distribution and vocabulary, these results show that a cache distributed across a high number of orbiting satellites can achieve the same result as a single instance of the same size. The resulting hit rate is excellent and offers a great reduction in network traffic related to lookup operations. It also shows the successful application of the

LRU-replacement algorithm on scale free distributed term collections. The findings related to cache training speed from Section VII-A are also a welcome contribution to a persistent improvement in the cache performance.

This particular experiment was designed with DNS services in mind, but will equally well predict the results for other SFD collections like e-mail addresses, web pages, X.509 Public Key certificates, collection of shared documents, dictionaries and thesauri, etc.

Future research activities on the SIN model will include the studies of state transitions, handover models, fault tolerance in the presence of failed satellites, routing methods, etc. One important idea in these activities is to consider the population density distribution of the planet in order to even out the workload of the satellites: Computing and communication tasks should be assigned to satellites with less population within its footprint.

REFERENCES

[1] L. Bai, T. de Cola, Q. Yu, and W. Zhang, “Space information networks,” *IEEE Wireless Communications*, vol. 26, no. 2, pp. 8–9, 2019.

[2] A. Fongen, “Application services in space information networks,” in *CYBER 2021*. Barcelona, Spain: IARIA, Oct 2021, pp. 113–117.

[3] ——, “Trust management in space information networks,” in *SECURWARE 2021*. Athens, Greece: IARIA, Nov 2021, pp. 14–18.

[4] “Starlink web site,” https://www.starlink.com/, [Online; accessed 04-Apr-2022].

[5] S. Briatore, N. Garzaniti, and A. Golkar, “Towards the internet for space: Bringing cloud computing to space systems,” in *36th International Communications Satellite Systems Conference (ICSSC 2018)*, 2018, pp. 1–5.

[6] S. Cao *et al.*, “Space-based cloud-fog computing architecture and its applications,” in *2019 IEEE World Congress on Services (SERVICES)*, vol. 2642-939X, 2019, pp. 166–171.

[7] D. N. Serpanos and W. H. Wolf, “Caching Web objects using Zipf’s law,” in *Multimedia Storage and Archiving Systems III*, C.-C. J. Kuo, S.-F. Chang, and S. Panchanathan, Eds., vol. 3527, International Society for Optics and Photonics. SPIE, 1998, pp. 320 – 326.

[8] J. Jung, E. Sit, H. Balakrishnan, and R. Morris, “DNS Performance and the Effectiveness of Caching,” in *1st ACM SIGCOMM Internet Measurement Workshop*, San Francisco, CA, November 2001, pp. 153 – 167.

[9] “Gridded population of the world v.4.11,” [Online; accessed 04-Apr-2022]. [Online]. Available: https://sedac.ciesin.columbia.edu/data/collection/gpw-v4/sets/browse

[10] G. K. Zipf, *Human behavior and the principle of least effort*. Cambridge, (Mass.): Addison-Wesley, 1949.

[11] A.-L. Barabasi, *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life*. Plume Books, April 2003.

 ISBN: 978-1-61208-975-1