CACHE RELOCATION FOR VOD IN WIRELESS NETWORK USING MOBILE AGENT TECHNOLOGY

Hasan Al-Sakran
King Saud Univ./ Riyadh, Saudi Arabia,
Yarmouk University /IT College, Irbed, Jordan.
Email: halsakran @ksu.edu.sa

ABSTRACT

Recent advances in mobile computing and wireless technologies resulted in the emergence of new services such as Video-on-Demand. However, high rates of disconnection, low bandwidth, and other problems still strongly affect the quality of such services. To address some of these challenges, we propose a new model for the caching of video content that takes advantage of the cooperation among distributed caching proxies containing hot VoD titles in wireless base and support station caches. Our model also considers the constant movement of mobile users. Mobile agents are used to transfer cached video content from one base station to another. As the user moves from place to place, a mobile agent can bring the user specific cache video content to his device's cache or to the base station under which the user is currently located.

Keywords: *Mobile agents, cache relocation, wireless network, VoD, VHO.*

1. INTRODUCTION

The significance of mobile computing in the computing industry is obvious. More and more smart mobile devices are supported and used all over the world. Mobile users require new services and enhanced capabilities for their mobile devices. One of these services is providing Video-on-Demand (VoD) in a timely fashion. Telecommunication companies project that Video-on-demand services will grow significantly over time, providing varied programs for e-learning, home entertainment, news, and other applications. Recent advances in wireless network technologies and mobile computing have made all of these applications possible. However the low bandwidth, high rate of disconnection, high error rate of wireless links create numerous challenges.

Considerable amounts of network bandwidth are required in order to deliver real-time VoD streaming to multiple subscribers due to the large volume of unicast traffic from Video Head Office (VHO) to users that such a network generates. Storing some of the most accessed video content in caches closer to subscribers will reduce traffic and therefore reduce network costs.

Caching is a technique for improving storage system. It is fundamental for both performance and functionality of networks and devices. Caching frequently accessed video content at locations close to a subscriber is an effective solution for improving the quality of wireless web applications. It has been a topic for numerous studies aimed at the improvement of the wireless network performance.

In mobile environments, caching can contribute to reductions in power consumption and network bandwidth, and allows disconnected operation. It is also very useful tool for reducing latency in accessing remote data. It is particularly effective for retrieval of referenced data that is unlikely to change frequently like VoD. Another advantage of caching video data locally to mobile nodes is the ability to retrieve data from a nearby node, rather than from a more distant support station.

There are major issues that should be considered in video content caching in wireless information systems:

1. Caches can use only a limited amount of memory space. If this space is full then some of the cached video content is replaced by a newly received item.
2. In order to reuse cached video, the system should be able to find it either at a proxy or at support stations caches.
3. Due to cache size limitation, the choice of cache replacement method to find a subset of items suitable for eviction from cache becomes essential. These methods can be: Least Recently Used, First In First Out, or Random.

Caching algorithms for mobile devices cache video content are based on temporality, locality and priority [1]. A temporality based caching algorithm caches information associated with a period of time based on the mobile client's

needs. Location based caching algorithms cache data related to a particular location. And priority based algorithms associate information with a priority value obtained from the mobile user.

The proposed architecture heavily relies on relocating caches of mobile devices, proxies maintained on base stations, and support stations. Moving selected video contents to a mobile device cache will reduce the efforts in mobile-based information retrieval. The cache management scheme involves relocation of all or a fraction of proxy's or support stations' caches to the most-likely-to-be visited areas. A movement prediction algorithm is used to determine the future location of the mobile device.

In this paper, we discuss how the mobile agent technology can be implemented to accelerate video browsing in the context of the proposed architecture. The rest of the paper is organized as follows. Section 2 surveys related work on caching schemes in wireless environments. After that we discuss system architecture, agent technology, types of agents used in the proposed architecture, and mobile caching operations and strategies in Section 3. In Section 4, we present the performance analysis. The conclusions are drawn and the directions of the future research are discussed in Section 5.

2. RELATED WORK

A number of studies have been aimed at improving performance of mobile networks. Most of the research was focused on reduction of the need to access the VHO since this operation is the most expensive in terms of time delays and resources. A number of solutions to this problem have been described in [2]-[16]. In [2]-[5], attempts have been made to resolve this problem by providing proxy caches between the mobile client and the VHO. Caching of frequently accessed data items on the client side at mobile devices and /or at locations close to the mobile clients is an effective solution for improving the performance of wireless video services applications in terms of improved accessibility and reduced access cost. None of these systems however addressed the user movement issues. As a result, whenever a user changes his point of attachment to a new support station in the mobile network he has to start from the beginning without taking advantages of previously gathered cached video content. Such systems require moving of cached video content from the present spot to another support station at a new location. Fleming discusses the design of a multithreaded proxy server that pre-fetches documents to mobile clients, reduces the resolution of large bitmaps, and communicates with clients using a new multiple hypertext stream protocol [6]. This method improves data availability when the wireless link is disconnected. The automatic reduction of resolution of large bitmaps improves performance over slow channels. A new caching technique in the wireless relay networks for video on demand service was proposed by Xie and Hua [7]. Luss in [8] proposes an algorithm to minimize the total of the cost of servers, cost of assigning program families to servers, and the cost of bandwidths used to broadcast videos. Chen and Chen [9], and Xiaoling *et al*. [10] proposed different video compression algorithms to improve transmission rate.

Haq and Matsumoto [11] and Hadjiefthymiades *et al*. [12] proposed the use of mobile agent technology to minimize the effect of problems encountered when using the Web over wireless networks and described possible solutions. However these systems employ a large number of agents for different activities, which is very likely to overload the network. These systems do not take advantage of the mobile devices' caches.

To improve performance, we use a path prediction algorithm to locate the next base station on the subscriber's path. Prediction is based on the user's movement history saved in the knowledge base and his current location. The use of a path prediction algorithm in a mobile network allows the efficient use of limited network resources such as disk capacity and bandwidth. A number of path prediction algorithms have been proposed in the wireless networking literature. Liu–Maguire in [13] proposed an algorithm that is based on a mobile motion prediction technique to estimate the future location of a roaming user according to his movement history. Liu et al. in [14] used pattern-matching techniques and extended self-learning Kalman filters to predict the future location of roaming user. More recent works in this area include the algorithms proposed by Akoush and Sameh in [15], and Burbey and Martin in [16]. The former deploys Bayesian learning for neural networks for predicting the location of a mobile user in wireless networks. Burbey and Martin implemented the prediction-by-partial-match based on a data compression algorithm as a predictor of future locations.

3. SYSTEM ARCHITECTURE

In this work we demonstrate an adaptive caching scheme in hierarchal or distributed mobile environment network as shown on Figure 1.

We assume a mobile network model which consists of mobile clients, base stations, and fixed or mobile hosts called support stations. Mobile clients access the mobile network through base stations, which are interconnected to form wireless LANs and are connected to the Internet through fixed or mobile support stations.

3.1. Overview of the main entities of the proposed mobile network model:

Mobile devices: These devices have their own cache memory space. When the mobile device is activated, it connects to the nearest available base station and initiates a request for video contents. Initially, the mobile device's and the base station's caches are empty. The caches are filling up as more and more requests from the mobile user are received and fulfilled. The best performance is provided if the requested video is present in the mobile device cache. Otherwise the mobile client has to send his request to a local base station.

Base stations (BS): Each station assigns cache memory space for each mobile user attached to it. All stations are capable of supporting the cache relocation mechanism for multiple mobile users.

Support stations (SS): Support stations maintain connections to multiple base stations. They are responsible for communicating with the server and retrieving the videos requested by the mobile user unless a recent copy of the video content exists in its own cache.

Each base or support station accommodates multiple agents.

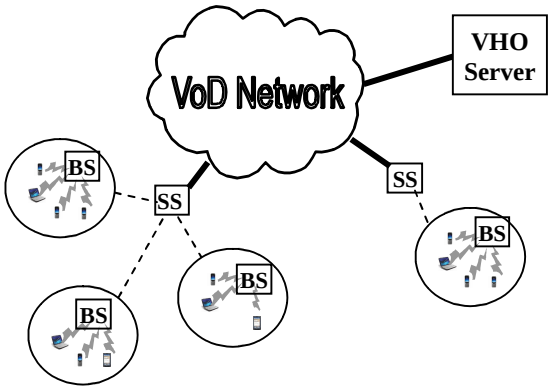

Figure 1. System Architecture.

3.2. Agent Technology

Agents are the autonomous software programs that reside within an open and unpredictable environment, sense and act upon it to achieve their goals. They use information, gathered by sensing environment or upon receiving messages from other agents or users, and an inference engine to decide on the actions to be performed. Some of the properties that distinguish agents from the regular programs are: autonomy, ability to learn, reactive or proactive decision making, continuity, mobility, etc.

Agents usually are small in size, and cannot work alone; it's not a complete application. They work in conjunction with an agent host and other agents. During their life span agents are required to interact with other agents and / or humans in order to sense environment, get necessary information, etc. A communication language or a proxy is used for such interactions between agents and other entities. Cooperation / collaboration with other agents might be essential in order to complete its task. Sometimes an agent may refuse to execute certain tasks due to an unacceptable high load on the network resources or simply because this would cause damage to other users.

Due to the popularity of the software agents a large number of agent frameworks appearing almost incessantly. Some of agent platforms are: Java Agent Development Environment (JADE), Java Agent Development Framework-Lightweight Extensible Agent Platform (JADE-LEAP), D'Agents, Mole, Hive, Voyager, Jini, Aglets, etc.

3.3. Agents Used in Proposed Architecture

Three types of software agent are employed in proposed architecture as shown on Figure 2. Two of them are static: the interface agent and the task agent. The third is mobile. Each agent is designed to represent specific functionalities.

The interface agent resides on a mobile device and serves as a communication point between a mobile user and the rest of the wireless network. It is created when a user initially connects to the VoD network. This agent's tasks are relatively simple. It interacts with a browser to intercept user's requests. After collecting a request the agent employs a mobile agent to relay it to the local base station and waits for new requests. As soon as the requested video content is delivered by mobile agents, the interface agent passes it back to the browser to be displayed to the mobile user.

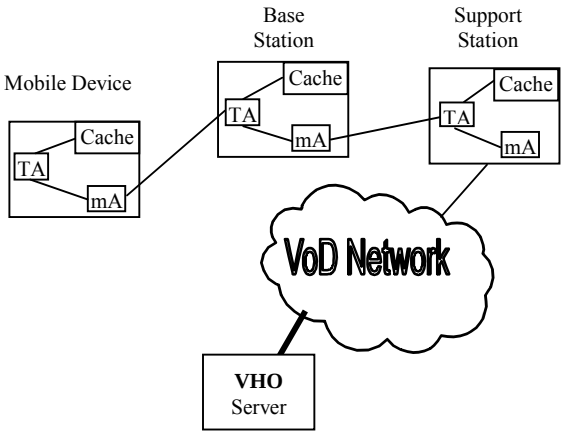

Figure2. Mobile Caching Operation.

The task agents are located on the base and the support stations. Both BSs and SSs are capable of generating multiple agents that move the content from memory allocated to the user from the station of origin to the new location based on prediction algorithm (PA). The task agent checks the cache of the corresponding node for the requested content. If the content is available, the agent will create a mobile agent that will transport the content to the requesting entity. Otherwise, a mobile agent will carry the request to the next entity, located higher in the network hierarchy.

Mobile agents travel either between a mobile device and a base station or between a base station and a support station. Their function is to move all or part of the cache allocated to a mobile device on a corresponding base station to the user, or from local support station to the base station.

3.4. Mobile Caching Operations and Strategies

In this paper, a cache model that takes advantage of the mobile device's cache and the cooperation among distributed caching proxies in wireless base and support stations caches is proposed (shown on Figure 1). This model also considers the constant movement of mobile users. A mobile agent is employed to fetch cache contents from one base station to another. As the user moves from place to place, a mobile agent can bring the user-specific cache to the base station under which the user is currently located.

When a moving mobile client requests a video and this video cannot be found in its local cache, a mobile agent is created at the client's device that carries this client's request to a certain proxy server at the base station, and some action will be taken based on the following situations:

Case 1: The requested video is stored in the proxy server cache:

The proxy server creates a mobile agent to carry the cached video content and sends it to the mobile client, so that the request for specific video can be fulfilled. The mobile device cache is updated with new content based on the received information.

Case 2: The requested video is not cached at the proxy server:

The proxy server creates a mobile agent, sends it to its support station and waits for a response. If that support station has the matching cache content, it populates the received mobile agent with the requested video, which then is routed back to the requesting proxy server. After that, the search process is continued as in case 1.

Case 3: The requested video is not in the support station cache:

A mobile agent is created at the support station and sent through the network to the specific VHO that has the requested video content. The retrieved content is delivered to the support station, and its copy is saved in station's cache. Then the mobile agent delivers the requested video to the base station local to the mobile client, saving another copy of the video content in the base station's cache. At last the requested content is carried to the mobile user, which also saves a copy in its own cache.

Based on the path prediction algorithm, moving of the base station cache's video contents to the expected base station may be completed by a number of different strategies:

1. One or more mobile agents may move the whole base station's cached video content to the nearest base station.
2. Only some of the cached videos are moved to most expected base station by single mobile agent.
3. Multiple mobile agents may move the base station's cached video contents to the most-likely-to-be visited base stations in fractions, for example the nearest one will have complete content, the second nearest 60% and so on.

4. SYSTEM PERFORMANCE

In the proposed model a mobile client may launch an agent from his device into a wireless network. This agent visits the client's base station server, obtains the required video content, if available, and then returns back to the place of its origin. It means that the target data is moved with the agent thus increasing its size by the size of that video. If the requested video has not been found on the base station, then a new agent is created and sent to the appropriate support station. In the event that the requested content is found in the support station's cache, the agent returns to the requesting base station with the retrieved video content. If not, then a mobile agent is created and dispatched to the specific VHO to continue the search.

The two major parameters affecting mobile agent performance are the mobile agent size and the time that agent requires to migrate between servers. The larger the size of a mobile agent, the more time is required to move between servers.

An agent migration between any two nodes consists of the following steps:

1. Agent Serialization: Construction of a message representing agent's data and code;
2. Agent transfer: Sending the serialized agent (message) to the target server over TCP/IP or ATP. At the target server a thread is constructed and the message is delivered to this thread;
3. Agent de-serialization: Reconstruction of the agent's data in the original form.

The mobile client creates an agent A_c which contains the client request to be executed. This agent moves to a base station i where it obtains required video if available, then returns to the place of its origin. For case 1, the total time (T_{case1}) that an agent requires to migrate from the client to a base station and back is described by given below equations (1) - (4):

$$T_{case1} = t_{ci} + P_i * t_{ai} + t_{ic} \quad (1)$$

$$t_{ai} = t_{si} + t_{pi} + t_{di} \quad (2)$$

$$t_{ic} = t_{ci} + S_i/R_{ci} \quad (3)$$

Where t_{ci} *and* t_{ic} – time to travel from client to base station i and back correspondingly;

P_i – probability of the requested video being on i-th base station;

t_{ai} – total processing time of mobile agent at base station i;

t_{si} – time needed for agent serialization at server i;

t_{pi} – time to process agent at station i;

t_{di} – time needed for de-serialization at station i and restart;

S_i – size of data;

R_{ci} – transmission rate of a link between base station and client;

R_{ic} – transmission rate of a link between client and base station.

For simplicity we assume $R_{ci} = R_{ic}$. Therefore :

$$T_{case1} = 2\, t_{ci} + P_i * t_{ai} + S_i/R_{ic} \quad (4)$$

For cases 2 and 3 total migration time can be derived as follows (5, 6):

$$T_{case2} = T_{case1} + 2\, t_{BS} + P_S * t_{ai} + S_i/R_{BS} \quad (5)$$

$$T_{case3} = T_{case2} + 2\, t_{S\text{-}HVO} + P_{HVO} * t_{ai} + S_i/R_{S\text{-}HVO} \quad (6)$$

Where t_{BS} – time to travel from base station to corresponding support station;

$t_{S\text{-}HVO}$ - time to travel from support station to corresponding HVO;

P_S and P_{HVO} – probabilities of requested video being on support station and HVO correspondingly;

R_{BS} – transmission rate between base and support stations;

$R_{S\text{-}HVO}$ - transmission rate between support station and corresponding HVO.

Calculation of time delays experienced while moving cached video content to the new location (next expected base station) have to be done with the consideration of the path prediction algorithm used and thus of the implemented strategy of cache relocation. Delay for moving all cache video content to the nearest base station only is given in (7):

$$T_{strategy1} = S_{Bi} / RB_{i,j} + t_{aj} \quad (7)$$

Where S_{Bi} – size of the base station cache content that belong to the given client;

$RB_{i,j}$ – transmission rate between base stations i and j.

Equation (8) describes the delay when only some of the cached videos are moved to the most expected base station:

$$T_{strategy2} = a * S_{Bi} / RB_{i,j} + t_{aj} \quad (8)$$

Where a – the fraction of the cache to be moved.

Time required for multiple agents to move cached video content from present base station to the most-likely-to-be-visited base stations in fractions is (9):

$$T_{strategy3} = max\{(a_1 * S_{Bi} / RB_{i,1} + t_{a1}), \ldots ,$$

$$(a_j * S_{Bi} / RB_{i,j} + t_{aj}), \ldots, (a_n * S_{Bi} / RB_{i,n} + t_{an})\} \quad (9)$$

To quantitatively evaluate the performance of wireless mobile device cache cooperation in structured mobile networks, we carried out a simulation. J2SDK was use to implement the simulator. Performance is evaluated in terms of cache hit ratio.We assume a predetermined number of base stations which are fixed and distributed randomly in the area. Each base station (BS) has a position (x_b, y_b), and a range, where it can accept the mobile signal. The range size is taken as a random number between 20 and 50 meters. Base stations can support a limited number of mobile devices. We assume 15 clients for the normal base station. We also assume a fixed number of support stations. Each can support an unchanging number of base stations. Its position is (x_s, y_s) and a maximum number of base stations it can support taken statistically as 7 for the normal support station. A mobile agent will be used for the transfer of video content among caches.

A number of mobile devices (users) is distributed randomly in the area. Each mobile device has a starting position (x_m, y_m) and a range where the mobile device wireless signal can be transmitted, provided there are base stations in the area.

The user is an object whose movement can be estimated based on the path prediction algorithm, so it has some movement scenario. This movement scenario is used to estimate the device's new location. It is assumed that mobile user stops for a small period of time, so its position (x_m, y_m) can be calculated, and stopping periods are taken into account.

The mobile tries to locate the nearest available BS. If the search is successful, then the mobile device is assigned a slot of the base station's cache, and it starts sending requests. A mobile client periodically checks it location to determine if it moved out of the base station's range. Using path prediction algorithm and measured coordinates the client's movement scenario is determined. The new base station is located and the content of the current base station cache is moved to the new station even before handover occurs. When the handover occurs the mobile client's previous cache slot can be reassigned a new client.

The simulator works as follows:

1. Build a number of fixed support stations with the previously specified attributes: (xs, ys), radius, base stations available.
2. Build a number of fixed base stations with the previously specified attributes: (xb, yb), radius, mobile clients available.
3. Build a number of mobile devices and randomly assign positions to them.
4. Create the moving scenario for the device based on the speed, the stop frequency and the stop duration of the mobile client.
5. Assign the mobile device to the nearest available base station, subject to it being assigned a number of clients that is less than or equal the maximum number it can support.
6. Move the mobile devices based on the movement scenario generated for them. At specified time intervals check the distance between the mobile device and the base station it is connected to. If it is greater than some threshold, look for a new base station and move the cache contents there.

When the handover occurs, the mobile device will be connected to a new station (according to the prediction algorithm) where its cached data items are transferred by an agent.

At the initial state the caches are empty. After the initialization steps (1-6), the mobile user searches for video in its own cache. If the video is found, 1 is added to the access counter; if not, a request for web video is sent to the base station and new search is conducted. If video is in the base station's cache the access counter is incremented by 1, video is delivered to the originator of the request and added to its cache. If not, another request is sent, this time at the corresponding base station. The same procedure is repeated again. In the case when video is not in the support station's cache, the simulator generates a random number representing the IP address of the web site where the requested video is located (VHO) and the request is delivered to this IP address. The video is retrieved and delivered to the SS where it stored in cache, then to the BS, and so on.

The system is tested in the stable state after the caches are partially filled with IP's.

The total time to receive a requested video (TT) can be expressed in the following way:

$$TT = Miss\ Rate_{MD} * Miss\ Penalty_{MD} \quad (10)$$

$$Miss\ Penalty_{MD} = T_{MC\ to\ BS} + Miss\ Rate_{BS} * Miss\ Penalty_{BS} \quad (11)$$

$$Miss\ Penalty_{BS} = T_{BS\ to\ SS} + Miss\ Rate_{SS} * Miss\ Penalty_{SS} \quad (12)$$

$$Miss\ Penalty_{SS} = T_{SS\ to\ HVO} \quad (13)$$

Where: $T_{MC\ to\ BS}$: time to travel from mobile client to base station;

$T_{BS\ to\ SS}$: time to travel from base station to support station;

$T_{SS\ to\ HVO}$: time to travel from support station to Video Head Office;

Figures 3-5 illustrate the effect of the cache sizes of mobile device and the corresponding caches in the base station and support station. As can be seen, the cache miss rate of the mobile device is 100% which is obviously due to the small cache size. As the cache size increases, the miss rate lessens. Similar dependency also holds for corresponding base and support stations.

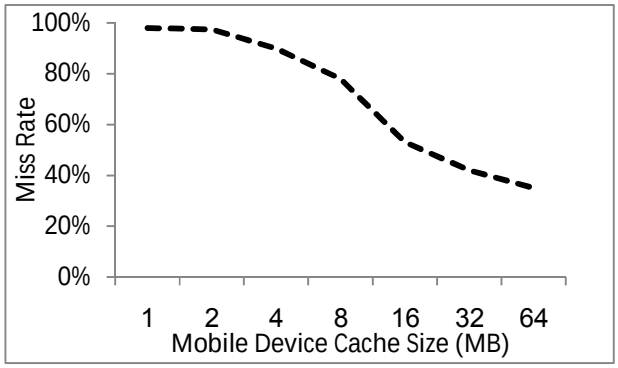

Figure 3. Cache miss ratio vs mobile device cache size

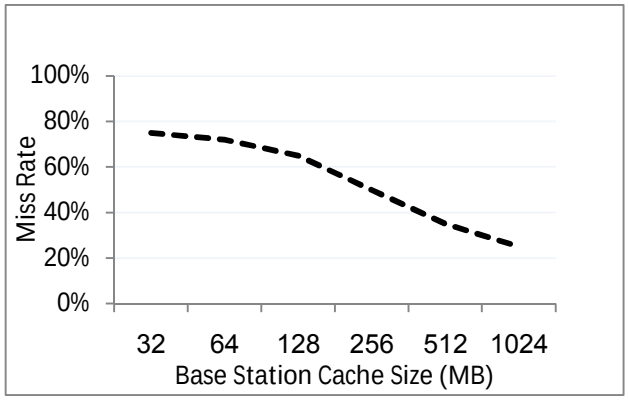

Figure 4. Cache miss ratio vs base station cache size

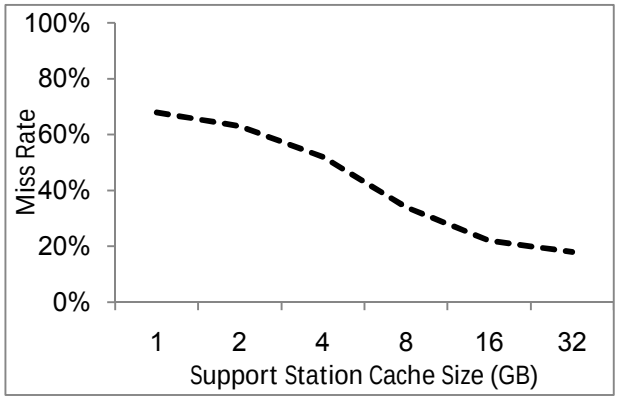

Figure 5. Cache miss ratio vs support station cache size

Figure 6 shows miss rate dependency on the size of the video for mobile device, base and support stations.

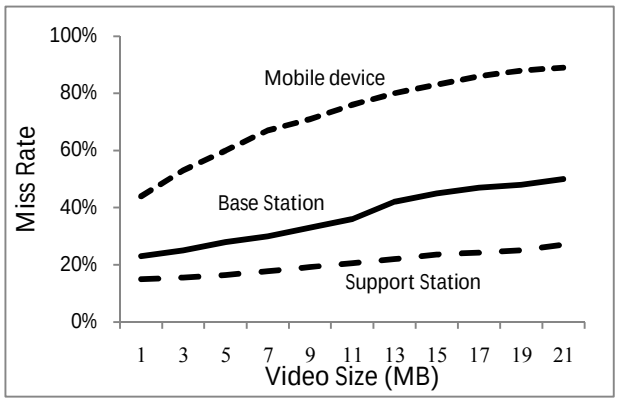

Figure 6. Cache miss ratio vs video size for mobile device, base and support stations.

To minimize the size of the data moved within the mobile agent, the agent compresses data obtained from the station, and returns to the mobile user device or station with the compressed cached data. On the mobile client or station site, the agent decompresses the cached data.

5. CONCLUSION AND FUTURE WORK

In this work we propose a caching model in which we cache the video content requested by the user in the mobile device's cache itself, and create a cooperative system between the mobile device, the proxy and supporting base station caches through the use of mobile agent technology.

The proposed system allocates cache video contents on intermediate sites (proxy and support stations) to reduce the latency of retrieval in a mobile network; it supports hierarchal cache updating to achieve cache consistency and thereby improve performance.

The next steps of our work will be designing an effective cache framework in mobile peer-to-peer networks that can handle heterogeneous mobile devices.

6. REFERENCES

[1] Mahgoub, I., and Ilyas, M. Mobile Computing Handbook, CRC Press, Dec 2004.

[2] Hadjiefthymiades, S., and Merakos, L. "Using Proxy Cache Relocation to Accelerate Web Browsing in Wireless/Mobile Communications", ACM, 2005.

[3] Wang, J., Du, Z., and Srimani, P. K. "Network Cache Model for Wireless Proxy Caching", IEEE proceedings for (MASCOTS'05), 2005.

[4] The MOWGLI Project: http://www.tml.hut.fi/Studies/Tik-300/Wireless/mowgli 1.html, 2003.

[5] Kumar, A., Misra, M., and Sarje,A. K. "A Weighted Cache Replacement Policy for Location Dependent Data" in Mobile Proceedings of the 2007 ACM symposium on Applied computing, March 11–15, 2007, Seoul, Korea.

[6] Fleming, T. B., Midkiff, S. F., and Davis, IV, N. J. "Improving the Performance of the World Wide Web over Wireless Networks", 1997.

[7] Fei Xie, Hua, K.A., "A caching-based video-on-demand service in wireless relay networks", IEEE Conferences: International Conference on Wireless Communications and Signal Processing, 2009.

[8] Luss, H.," Optimal Content Distribution in Video-on-Demand Tree Networks", IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 2010.

[9] Chen, Y.W., Chen, S.S., "A Broadcasting Scheme with Low Buffer Requirement for Users with Limited Bandwidth", Second International Conference on Computer Engineering and Applications (ICCEA), Indonesia, 2010.

[10] Xiaoling Qiu, Haiping Liu, Ghosal D., Mukherjee B., Benk, J., Wei Li, Bajaj R.., "Adaptive video compression rate optimization in wireless access networks", IEEE 34th Conference on Local Computer Networks, Zurich, 2009.

[11] Haq, M.A., and Matsumoto, M. "MAMI: Mobile Agent Based System for Mobile Internet", The 6th International Conference on Advanced Communication Technology, Volume 2, Issue, 2004 Page(s): 567 – 572.

[12] Hadjiefthymiades, S., et at. "Supporting the WWW in Wireless Communications through Mobile Agents", Mobile Networks and Applications, Vol. 7, 2002, pp.305-313, Kluwer Academic Publishers.

[13] Liu, G.Y., and Maguire, Jr., G.Q. "Efficient Mobility Management for Wireless Data Services," Proceedings of IEEE VTC'95, Chicago, IL 1995.

[14] Liu, T., et al. "Mobility Modeling, Location Tracking, and Trajectory Prediction in Wireless ATM Networks", IEEE Journal on Selected Areas in Communications, 16(6), August, 1998.

[15] Akoush, S., and Sameh, A. "Mobile User Movement Prediction Using Bayesian Learning for Neural Networks", Proceedings of the 2007 International Conference on Wireless Communications and Mobile Computing, ACM, 2007.

[16] Burbey, I, and Martin, T. "Predicting Future Locations Using Prediction-by-Partial-Match", MELT '08: Proceedings of the first ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments, ACM, Sept. 2008.