Advance Handovers Arrangement and Channel Allocation in LEO Satellite Systems

Peng-Jun Wan Vincent Nguyen Hong Bai

Abstract— **Due to the high mobility nature of the Low Earth Orbit (LEO) satellite systems, satellite's coverage area changes with time causing extremely frequent handover rate. Therefore handover procedure has a significant impact on the success of LEO systems and handover failure rate is a major criterion in measuring system's performance. In this paper we propose a handover procedure called Advance Handovers Arrangement and Channel Allocation (AHACA) to maximally exploit the deterministic and predictable characteristics of LEO systems to reduce handover failures and optimize channel utilization. This method does not exclusively guard channels for handovers, instead it queues and reserves channels that are currently in use but will be released within expected queuing time.**

Keywords: Low earth orbit satellite, handover, channel allocation, call admission.

I. Introduction

Low Earth Orbit (LEO) systems such as the Iridium, Globalstar, and others have been proposed and implemented to complement and interface with existing Public Switch Telephone Network (PSTN) and Public Land Mobile Networks (PLMN) to provide a true global mobile communication. Orbiting at low altit' des ranging from 500 to 1500 km, these systems have allowed the size of the user's mobile terminal, transmission power, and latency delay in LEO network to be comparable to that of the PLMN. These advantages however come with a great challenge. LEO satellites must maintain a high-constant velocity resulting in a mobility of up to 7 km/sec scanning the earth surface. Due to this dynamic nature, calls and connections must be handed over from one satellite spot-beam's footprint (FP) to another within tens of seconds and from one satellite to another every few minutes. Because of such extremely frequent handover rate, the channel allocations and handover techniques are major factors determining the efficiency and performance of the system [3], [8].

Various handover schemes have been proposed in the literatures [1], [2], [4], [5], [9], [10]. The handover is in general initiated by the user terminals. Each active user terminal continuously monitors and averages relative signal strength and interference. It places a handover request when entering an overlap area and the average signal strength from the current base becomes weak and interference signal has reached predefined thresholds determined by the handover initiation algorithm [2], [9]. **Guard Channel** (GC) and **First Come First Serve-Queuing Handover** (FCFS-QH) are the two techniques widely used by the mobile satellites to handle the handover requests. *Guard channels* are a fixed or a dynamically adjustable common channel pool reserved exclusively for handovers. A handover request that arrives when a channel is not available will search for a guard channel. The handover call is then served using one of the guard channels if available, which will be replaced later whenever a channel becomes available. If guard channel is not implemented or not available, FCFS-QH technique can then be applied to the handover calls. The handover requests will wait in a queue for some channels or guard channels to be released. When a channel is released, the handover request which arrives the earliest gets served with this released channel. Handover request can be queued until Quality Of Signal (QOS) falls below an acceptable level or up to a maximum queuing time if a channel is not available.

Both techniques can reduce handover failure rate; however, they comes with the cost of higher blocking of new calls. Pre-defined number of guard channels can not accommodate significant change or increase of handover calls, and consequently in traffic conditions when number of handovers is minimal, unused guard channels are wasted because they can't be used for new calls. Dynamically adjustable guard channels can overcome this disadvantage to some extent; however, reserving an optimal number of guard channels requires a priori knowledge of traffic patterns and distribution. QH comes with a higher blocking rate since a new arrival won't get admitted or queued until all handover requests are served. The efficiency of QH, therefore, depends on the size of the overlap area. Wider overlap area allows longer maximum queuing time and thus reduce the handover failure probability, but it also increases blocking probability. This paper will provide yet another handover scheme called Advance Handovers Arrangement and Channel Allocations (AHACA). AHACA maximally exploits the deterministic mobility and predictability characteristics of LEO systems. It does not provide any guard channels. Instead, it queues and reserves channels that are currently in use but will be released within expected queuing time according to an optimal greedy algorithm. Thereby it reduces handover failure rate while still maximizing channel utilization.

The remaining of this paper is arranged as follows. In section II we briefly describe the predictable behavior

Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616. Email: wan@cs.iit.edu. This works is supported by ERIF 4-52534 funded by IIT.

Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616. Email: nguyvin@charlie.cns.iit.edu.

Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616. Email: baihong@charlie.cns.iit.edu.

0-7803-5796-5/99/$10.00 © 1999 IEEE

that will be later exploited by AHACA. In section III we present the AHACA strategy. In section IV we compare AHACA against the existing approaches. Finally Section V concludes this paper.

II. Predictable Behaviors of LEO Systems

The behavior of LEO satellite systems is quite deterministic and predictable, which is quite different from traditional mobile cellular networks. These unique characteristics of LEO system will be later exploited by our handover processing.

First of all, accurate user position can be determined and available. Depending on the type of LEO systems, the user terminal can determine the user's position if given MSS's position, beam's size and its mapping coordinates specific to user's location on earth and by having user terminal to monitor and measure signal power from multiple satellite beams [6]. Another position determination method is to measure delays and Doppler shifts. This method had been done using two or more satellites such as in Global Positioning System (GPS) [7], [11], [12]. Since the probability of seeing only single MSS is high, it is highly desirable to determine user's position using single MSS. One possible modification is to determine position by measuring signal round-trip delays and Doppler shifts at both user terminal and MSS in two-way communication. The result is then analyzed with currently available data such as beam's size, previous position, and most recent and accurate signal measurement when two or more MSSs were available [7]. Both techniques are practical since user terminal constantly monitors signal power and reports to MSS for power control purposes. Current MSS's position, footprint's coverage and mapping of coordinates are provided by MSS since the information is maintained and updated for management and control purposes by the control center as MSS orbits the earth and thus is available to MSS and user terminal for reference. Position is established by each user terminal when call is admitted and updated during the call for routing and billing purposes.

Secondly, the direction of any handover is deterministic. Because of the high and deterministic mobility of LEO, user's mobility is negligible. Thus at any time when a call has to be handed over, the footprint which the call to be transited to is deterministic. In addition, because the user's position can be determined, the time by which the handover has to be completed is also predictable . Let R be the radius of the beam and v be the velocity of the satellite. Let (X, Y) be the coordinate system where $(0, 0)$ is the center of the footprint, and the X-axis is along the moving direction of the footprint. Consider a call by a user whose initial location is (x_u, y_u), which can be determined when the call is set up. The call can only travel a maximum distance equal to

$$x_u + \sqrt{R^2 - y_u^2}$$

within that footprint. Thus the maximum handover interval of the first handover is equal to

$$\frac{x_u + \sqrt{R^2 - y_u^2}}{v}.$$

If all footprints' sizes are the same then subsequent handover intervals are fixed and can be calculated as well. Thus the user terminal is capable of calculating and maintaining its handover sequence and intervals. However, if footprints' sizes are different, the satellite should inform user terminal the current footprint's coverage after each successful handover; user terminal then calculates and maintains the handover interval within that footprint. In either case, handover interval should be adjusted accordingly based on the point of handover in the overlap area to avoid running short or running over into the next interval. User terminal continues to passively monitor and calculate its position and informs gateway if the distance threshold is exceeded and when inter-satellite handover occurs to resolve any position error and ambiguity.

III. AHACA Strategy

In AHACA, when a call is connected, MSS should inform the user terminal the current footprint's coverage and its mapping coordinates. The user terminal then determine its handover sequence and calculate handover intervals using forth described method. As the average signal measurement and handover interval have reached predefined thresholds when a user terminal approaches an overlap area, the user terminal sends a handover request along with synch bits, its handover deadline, and handover candidate based on the handover sequence to MSS. Average signal strength monitored by the user terminal and thresholds must be used appropriately to ensure the accuracy of handover interval. To minimize the number unnecessary admit-then-handover sequence that increases handover and failure rate, we propose that any new call places inside the overlap area is handled by the appropriate trailing footprint for admission. In this case, it is determined by gateway when call is setup.

Each MSS must maintain three queues for each spotbeam footprint:

- *handover request queue* (**HO_Q**), is used for unserved handover requests received from the three preceding footprints. Each request is associated with its deadline by which it has to be served. These handover requests in the queue are sorted in the increasing order of their deadlines.
- *new call request queue* (**NC_Q**), is used for unserved new call requests from its own footprint. Each unserved new call request is also associated with a deadline by which it has to be served. These unserved new call requests are also sorted in the increasing order of their deadlines.
- *channel queue* (**CH_Q**), is used to maintain all channels provided by this footprint. Each channel is associated with an available time defined as follows: the available time of an idle channel's available time is zero, and the available time of a channel that is currently in use

is set to be the handover deadline of the user terminal that uses this channel. All channels are also sorted in the increasing order of their available times.

The channel reservation is the key part of the handover. A channels is eligible to a request if and only if its available time is less then the deadline of the request. The handover requests should have higher priority over new calls and as many handover requests as possible should be served to minimize the call dropping probability. There may exist many different reservations which all serve the maximal number of handover requests. Among them, the one which leaves most room, in terms of channel availability, to the new calls also allows the serving of the maximal number of new calls. In addition, in order to the increase the channel utilization, the queuing time of each new call request should be as short as possible. Such utilization improvement can bring additional revenue to the service providers. With these objectives in mind, the three queues are maintained and updated as follows.

A. Handling of Handover Requests

Upon receiving a handover request, the MSS then inserts the request into the handover request queue at proper position. Then a channel reservation is performed. An optimal reservation which can matches the maximal number of handover requests to the channels can be formulated as a maximum bipartite-matching as follows. Suppose there are M handover requests

$$R = \{r_1, r_2, \cdots, r_M\}$$

and their deadlines are $d_1 \leq d_2 \leq \cdots \leq d_M$ respectively. There are N channels

$$C = \{c_1, c_2, \cdots, c_N\}$$

and their available times are $a_1 \leq a_2 \leq \cdots \leq a_N$ respectively. Then we can construct a bipartite graph $G = (R, C, E)$ in which there is an edge between a request $r_m \in R$ and a channel $c_n \in C$ in E if and only if $d_m < a_n$. It's obvious that an optimal channel reservation corresponds to a maximum matching in G and vice versa. Many polynomial-time algorithms are available to find a maximum matching in a bipartite graph. However, simpler greedy algorithm can be designed to take advantage of the special properties of the instances. Such algorithm is favorable as it requires less computational power and the output can be generated much faster, which are very essential to the MSS. Table I lists such greedy algorithm. For any request under concern, it finds a eligible channel, if there is any, whose available time is the closest to its deadline. This criteria intends to leave those channels which have earlier available time for new call requests, without sacrificing any handover requests. Thus this is the desired win-win situation. In the next, we prove the optimality of this simple greedy algorithm.

Theorem 1: The algorithm **Handover_Reservation** always supports the maximal number of handover requests.

Algorithm: Handover_Reservation
Input:
  deadlines: $d_1 \leq d_2 \leq \cdots \leq d_M$
  available times: $a_1 \leq a_2 \leq \cdots \leq a_N$
Output: *a maximum reservation schedule;*
begin
  $C = \{c_1, c_2, \cdots, c_N\}$
  For $m = 1$ to M do
    Let $n = \arg\max\{k : c_k \in C, a_k < d_m\}$;
    If $n \neq \emptyset$ then
      assign request r_m with channel c_n;
      $C = C - c_n$;
      If $C == \emptyset$ then stop;
end

TABLE I
GREEDY ALGORITHM TO FIND A MAXIMUM RESERVATION SCHEDULE FOR HANDOVER REQUESTS.

Proof: Suppose that the output of **Handover_Reservation** is

$$\Gamma = \{(r_{m_k}, c_{n_k}) : 1 \leq k \leq K\}$$

where $1 \leq m_1 < \cdots < m_K \leq M$. We first claim that Γ must be a subset of some optimal matching. Assume to the contrary. For any $1 \leq i \leq K$, let

$$\Gamma_i = \{(r_{m_k}, c_{n_k}) : 1 \leq k \leq i\}.$$

Let i be the *largest* integer between 1 and $K-1$ such that Γ_i is contained in some optimal matching, say *Opt*, but Γ_{i+1} is not contained in any optimal matching. Clearly in *Opt* either the request $r_{m_{i+1}}$ is matched a channel other than $c_{n_{i+1}}$ or the channel $c_{n_{i+1}}$ is matched with some request other than $r_{m_{i+1}}$. Therefore there are three following possible cases.

Case 1: In *Opt* the request $r_{m_{i+1}}$ is matched with a channel $c_j \neq c_{n_{i+1}}$ but the channel $c_{n_{i+1}}$ is unmatched. In this case, the matching

$$Opt - (r_{m_{i+1}}, c_j) + (r_{m_{i+1}}, c_{n_{i+1}})$$

is an optimal matching but it contains Γ_{i+1}, which contradicts to the selection of i.

Case 2: In *Opt* the request $r_{m_{i+1}}$ is unmatched but the channel $c_{n_{i+1}}$ is matched with a request $r_j \neq r_{m_{i+1}}$. In this case, the matching

$$Opt - (r_j, c_{n_{i+1}}) + (r_{m_{i+1}}, c_{n_{i+1}})$$

is an optimal matching but it contains Γ_{i+1}, which also contradicts to the selection of i.

Case 3: In *Opt* the request $r_{m_{i+1}}$ is matched with a channel $c_j \neq c_{n_{i+1}}$ and the channel $c_{n_{i+1}}$ is matched with a request $r_\ell \neq r_{m_{i+1}}$. In this case,

$$j \neq n_1, \cdots, n_i, n_{i+1}$$

and

$$\ell \neq m_1, \cdots, m_i, m_{i+1}.$$

In addition, $j < n_{i+1}$ for otherwise **Handover_Reservation** would not match the request $r_{m_{i+1}}$ with the channel $c_{n_{i+1}}$. Furthermore, $\ell > m_{i+1}$ for otherwise **Handover_Reservation** would match the request r_ℓ with some channel before considering the request $r_{m_{i+1}}$. So we have

$$d_\ell \geq d_{m_{i+1}} > a_{n_{i+1}} \geq a_j.$$

Thus the matching

$$\begin{aligned} &Opt - (r_{m_{i+1}}, c_j) - (r_\ell, c_{n_{i+1}}) \\ &+ (r_{m_{i+1}}, c_{n_{i+1}}) + (r_\ell, c_j) \end{aligned}$$

is an optimal matching but it contains Γ_{i+1}, which also contradicts to the selection of i.

Thus in either case, we have contradiction. This implies that there must exist an optimal matching, say *Opt*, that contains Γ as a subset. We now prove that $Opt = \Gamma$ once again by contradiction. Assume that there exists a pair

$$(r_\ell, c_j) \in Opt - \Gamma.$$

Then $\ell > m_K$ for otherwise **Handover_Reservation** would match the request r_ℓ with some channel before considering the request r_{m_K}. Note that **Handover_Reservation** stops when either all channels have been assigned or no remaining channel can match with any unmatched requests. But $\ell > m_K$ implies neither of the conditions is true after **Handover_Reservation** matches the request r_{m_K} with the channel c_{n_K}, which leads to a contradiction. Thus $Opt = \Gamma$. ∎

B. Handling of New Call Requests

Upon receiving a new call request, it is handled according to the procedure listed in Table II. If an idle channel is used, it will serve this new call request, and its position in the channel queue is updated. Otherwise this new call request is placed at the new call request queue.

```
Procedure: New_Call_Handling
Input: a new call request r;
begin
  if there is an idle channel c
    serve call r with channel c;
    calculate the available time of channel c;
    update the position of channel c in CH_Q;
  else
    insert call r into NC_Q;
end
```

TABLE II
PROCEDURE TO HANDLE A NEW CALL.

C. Handling of Channels Released from Handovers

When a channel is released from a handover, it is handled according to the procedure listed in Table III. If it is the reserved channel by some handover request, then this request is removed from the queue and is handed over to this channel, whose position in the channel queue is updated accordingly. If this channel is not reserved by any handover request, then all handover requests must all have their own reserved channels from the property of our handover reservation algorithm. So this channel can be used to serve new calls. If there is any unserved new channel, the new call request at the top of the new call request queue, which has the earliest deadline, is removed from the queue and is served by this channel, whose position is also updated accordingly. If there is no unserved new call, then this call is put on the top of the channel queue as an idle channel.

```
Procedure: Released_Channel_Handling1
Input: a channel c released from a handover;
begin
  if it is reserved by some handover request r
    remove r from HO_Q;
    handover r to channel c;
    calculate the available time of channel c;
    update the position of c in CH_Q;
  else
    if NC_Q is non-empty
      remove the top request r from NC_Q;
      serve call r with channel c;
      calculate the available time of channel c;
      update the position of c in CH_Q;
    else
      set the available time of c to be 0;
      place c on the top of CH_Q;
end
```

TABLE III
PROCEDURE TO HANDLE A CHANNEL RELEASED FROM A HANDOVER.

D. Handling of Channels Released from Call Terminations

When a channel is released from the termination of a call, it is handled according to the procedure listed in Table IV. If there is some handover request which has no reserved channel yet, then the handover request at the top of the handover request queue, which has the earliest deadline, is removed from the queue and is handed over to this channel. The position of this channel in the channel queue is then updated accordingly. After that a channel reservation is performed among the remaining handover requests in the queue. This reservation allows one new handover request to be assigned one reserved channel. If all handover requests have their own reserved channels, this channel is then used to serve new channels if there is any as in the case that a channel is released from a handover.

Procedure: Released_Channel_Handling2
Input: a channel c released from a call termination;
begin
  if *there is any unmatched handover request*
    remove the top request r from **HO_Q**;
    handover r to channel c;
    calculate the available time of channel c;
    update the position of c in **CH_Q**;
    update the reservation;
  else
    if NC_Q is non-empty
      remove the top request r from **NC_Q**;
      serve call r with channel c;
      calculate the available time of channel c;
      update the position of c in **CH_Q**;
    else
      set the available time of c to be 0;
      place c on the top of **CH_Q**;
end

TABLE IV
PROCEDURE TO HANDLE A CHANNEL RELEASED FROM THE TERMINATION OF A CALL.

E. Handling of Expired Requests

If there is an unserved handover request whose deadline has expired, then it must be on the top of the handover request queue. This request is then removed from the handover request queue and the call is dropped.

Similarly, if there is an unserved new call request whose deadline has expired, then it must be on the top of the new call request queue. This request is then removed from the new call request queue and the call is dropped.

IV. Performance Comparisons

In this section, we compare the performances of AHACA and FCFS-QH. First of all, in FCFS-QH, a new call is admitted only when handover request queue is empty and a channel is available. In AHACA, the new calls can still get served immediately even if the handover request queue is not empty as long as there is an idle channel and all handover requests each have their own reserved channels. If there is no idle channel, the new call is put in a new queue and still could potentially get served by some channel released before its expiration. Thus AHACA potentially reduces the blocking rate of new calls.

Secondly, due to the varying sizes of the overlapping areas at different geographical locations, the deadlines of the handover requests might not follow their arrival order. In this case, the FCFS-QH might serve less handover requests than AHACA would do. This can be illustrated by the following example. Four handover requests whose handover deadlines are $2, 4, 3$ and 7 respectively. Four channels will be released at $1, 2, 3$ and 6 respectively. Then in FCFS-QH the third handover request will be dropped. But in AHACA all requests will be served. Thus AHACA potentially reduces the dropping rate of handover calls.

Finally, AHACA avoids the dilemma faced by Guide Channels by maximally exploiting the predictable behavior of LEO systems. It supports the maximal number of handover requesting while at the same time supporting the maximal number of new calls and maintaining the highest channel utilization.

V. Conclusion

Many MSS have been proposed and implemented to provide voice, data, multimedia, and HDTV services. Due to high mobility, limited resources, handoff has become a major factor in determining the efficiency and success of the system. Several handover strategies have been analyzed and proposed in different studies. The proposed AHACA scheme clearly shows an improvement over the widely used QH and GC in overall channel utilization and throughput. It is an optimal approach to maximize channel utilization with respect to both handover and new call success rate.

References

[1] E. Del Re, R. Fantacci, G. Giambene, Efficient Dynamic Channel Allocation Techniques With Handover Queuing For Mobile Satellite Networks, IEEE JSAC V 13, No 2, Feb 1995, pp. 397-405

[2] E. Del Re, R. Fantacci, G. Giambene, Call Blocking Performance for Dynamic Channel Allocation Technique in Future Mobile Satellite Systems, IEEE Proc. Com., V 143, No 5, Oct 1996, pp. 289-296

[3] B. Gavish , Low Earth Orbit Satellite Based Communication Systems - Research Opportunities, European Journal of Operational Research 99, 1997, pp. 166-179

[4] K. Kwok, P. Liu, K. Li, Channel Borrowing Techniques with Reservation Pool for LEO satellite Land Mobile Communication Systems, IEEE Fifth international conference on universal personal communication, 1995, pp. 129-132

[5] J. Lee, Y. Mun, B. Kim, Channel Allocation and Handover Schemes for Personal Communication Systems, IEEE Com., May 1996, pp. 943-947.

[6] N. Levanon, Quick position determination using 1 or 2 LEO satellites, IEEE Transactions on Aerospace and Electronic Systems, 1998, Volume: 34 3 , Page(s): 736 -754.

[7] K. Molnar, H. Koorapaty, Position location using multiple satellite beams, Vehicular Technology Conference, 1998. VTC.98. 48th IEEE Volume: 3, Page(s): 1945 -1949 Vol.3

[8] G. Pollini, Trends in Handover Design, IEEE Communications Mag., March, 1996, pp. 82-90

[9] G. Ruiz, T.L. Doumi, J. G. Gardiner, Handover Strategies in an Integrated Satellite/Terrestrial Mobile Radio System, IEE Telecommunication, March, 1995, No. 404, pp. 174-178

[10] V. Santos, M. Silva, J. Dinis, Performance Evaluation of Channel Assignment Strategies and Handover Policies for Satellite Mobile Networks, IEEE Forth international conference on universal personal communication, 1995, pp. 86-90

[11] P.V. Sudaroli, S. Kannan, N. Nagarajan, User Position Estimation for Search and Rescue Using Satellite in LEO constellation, IEE Proceedings - Radar, Sonar Navig., Vol 144, No. 1, Feb., 1997.

[12] W. Zhao, Tafazolli, B.G. Evans, Positioning Assisted Intersegment Handover for Intergrated GSM-Satellite Mobile System, Satellite Systems for Mobile communications and Navigation, IEE, May 1996.