Efficient Fault Tolerant Mobile IP in Wireless Networks
Using Load Balancing Approach

1Shiladity Bose and 2Chittaranjan Hota
1ADOBE Systems India Private Limited
Adobe Towers, I-1A, City Centre, Sector: 25A, NOIDA-201301, India
2Computer Science and Information Systems Group,
Birla Institute of Technology and Science, Pilani, Rajasthan, 333031, India

Abstract: This research starts by analyzing the pre-failure condition in a wireless network with Mobile IP
and proposes a novel approach using load balancing concept of foreign agents to tolerate their failures. It
involves distributing the traffic to the foreign agents in a way that each foreign agent has comparable
traffic. In the proposed approach, we don’t require any recovery procedure whenever a foreign agent
recovers from failure. A recovered foreign agent automatically gets its share of traffic as soon as it moves
back into the system. We analyze our approach through simulation. The results indicate that the approach
improves the pre-failure condition in a wireless network with Mobile IP and this improvement in turn results
in an efficient fault tolerant approach to tolerate failure of foreign agents.

Key words: Mobile IP, load balancing, foreign agent, fault tolerant, wireless network, radio access network

INTRODUCTION

The 3G networks (Kasera and Narang, 2004) support variety of services like voice, data and video by
introducing a new multiple access technique for the air
interface in the form of Widespread Code Division
Multiple Access. The task of integrating these wireless
networks into the Internet is handled by Mobile IP
(Perkins, 2002) which continuously supports the
network connections of the mobile users despite their
movement across wireless networks. Mobile IP
achieves this by introducing several mobility agents
(Foreign Agent and Home Agents) into the architecture
of a wireless network. Failure in such a mobility agent
can abruptly terminate the network connections of
several mobile users and thus the issue of fault
tolerance of Mobile IP in wireless networks becomes
utmost importance. This paper proposes a novel
technique i.e., load balancing to ensure fault tolerance
in Mobile IP in wireless networks. Our fault tolerant
algorithm transfers load from faulty foreign agent to
non-faulty foreign agents. When the faulty foreign
agent recovers, it automatically gets its share of load by
the load balancing algorithm. Here, the term load refers
to the number of mobile users the particular foreign
agent can handle. We have also proposed a hand off
algorithm that redirects the data requests of the failed
foreign agent to the new foreign agent that will support
these users in future. Our approach is efficient as it
uses load balancing mechanism to bring up automatically new foreign agents.

SYSTEM MODEL

The system model in Fig. 1 shows the architecture of a 3G wireless network (Sarikaya, 2000) with
Mobile IP. The architecture has major components as Mobile nodes, Radio Access Networks (RAN)
(De Vriendt et al., 2002), Core network and Interconnection network. Mobile nodes are the mobile
stations with mobile node capabilities as per (Perkins, 2002). Radio access network resides between the user
equipment and the core network. The mobile nodes pass their data requests to the RAN which delivers
these to the core network. The reply to these data requests is passed on by the core network to the RAN
which in turn passes these on to the mobile nodes. The
air interface of the RAN is used by the mobile nodes to
perform their wireless data sessions. Core network is an
IP based network which has the mobility agents
(Adelstein et al., 2005), which use other routers to pass
on the data requests of the mobile nodes to the

Corresponding Author: Shiladity Bose, ADOBE Systems India Private Limited, Adobe Towers, I-1A, City Centre, Sector:
25A, NOIDA-201301, India
Internet. Interconnection network connects the RANs to the core network. This network logically links all the RANs to the Foreign Agents (FAs). The interconnection network can be an IP, ATM or Frame relay network. The Operation and Maintenance center (OAM&M) (Mistry et al., 1997) detects faulty agent and runs our load balancer to automatically shift the load of the faulty component to the other active ones. The faults are assumed to occur in the FA only. The home agents are assumed to function without failure. The failure of a foreign agent can be detected when an agent advertisement is not received within a period of time. Once a FA fails it stops sending any agent advertisement and all the pending data requests with it are lost.

RELATED WORK

Lin and Arul (2003) proposed a fault tolerant approach in wireless network with Mobile IP. In their approach, before failure, a particular FA services the mobile nodes of a Radio Access Network (RAN). An interconnection network is introduced between the RANs and the core network which facilitates the interconnectivity of any FA to any RAN. Thus, there is logically a path from every FA to every RAN. Now, when a foreign agent fails, a fault tolerance procedure takes over and selects a backup set of failure free foreign agents which can take up the load of the failed foreign agent. A system initiated handoff takes place to move the failure affected Mobile Nodes (MNs) to the selected failure-free FAs. This handoff does not induce any location changes on the MNs. Our approach uses a load balancing algorithm to distribute uniformly the number of mobile user connections amongst the foreign agents. This approach reduces the need of the recovery procedure when a failed foreign agent restarts.

Ghosh and Varghese (1998) described mechanisms to allow redundant home or foreign agents to enhance the reliability of Mobile IP in case of home or foreign agent failures. Redundant agents can take over from each other in case of failure and also distribute load amongst themselves. This changeover is transparent to mobile nodes and also adheres to Mobile IP specifications. They need also a synchronization protocol which is not required in our approach.

Cheng et al. (2005) proposed a fault tolerant multi agent based mobile IP schema which handles the failure of a mobile agent in a network. In their approach, the mobile agent keeps a mobile agent table and a binding update table to manage the binding. They have also proposed a handoff, take over and recovery mechanism for the mobile environment. In our approach, binding update mechanism does not require any tables and hence is smoother. Also, our handoff mechanism does not require any buffering mechanism hence reduce the resource requirements at an agent.

THE PROPOSED APPROACH

In the approach suggested in Lin and Arul (2003), the FA serving record of a RAN is set to the identifier of a FA resulting in a static binding of a RAN to a FA. Thus, a fixed FA will serve any mobile node in the service area of a RAN. In our approach, the agent advertisement of any Foreign Agent is broadcast to all the RANs over the interconnection network. Thus, any mobile node coming up in the service area of any RAN
can request registration with any foreign agent which results in a dynamic selection of Foreign Agent to service a mobile node rather than fixing a FA to service a RAN. Our approach uses load balancing between FAs. Let us assume that every Foreign Agent has an optimal load i.e., a load that can be handled by it without considerable blocking. Let us quantify this load, as the optimal load that can be handled by any FA is X. An FA having load more than X is an FA with high traffic and an FA having load less than X is said to have low traffic and an FA with load equal to X is said to have optimal traffic.

**PROPOSED ALGORITHMS**

**Load balancing algorithm:** The load balancing algorithm shown in Fig. 2 can be part of the OANetM Network. The algorithm runs periodically, this time period can be changed based on the network conditions. Based on the data collected at the OA and M, the algorithm determines the load at all the FAs and then checks to see whether any FA is at high traffic. If so it selects a FA with low traffic and moves some mobile nodes from the FA with high traffic to the selected FA with low traffic. The load balancing algorithm performs this distribution only if some low traffic FA has load less than optimal load. If all FAs are at load higher than optimal load then the system works like a common system without load balancing. We make the following assumptions:

- The load at a FA is directly proportional to the number of mobile nodes supported by that foreign agent.
- The service capacity of all the foreign agents is same, i.e., the number of data requests that a foreign agent can service per unit time is same for all foreign agents in the system.

The data structures used in the load balancing algorithm are described as below:

- `no_of_fa` = Total number of foreign agents in system.
- `optimal_number` = number_of_mobile_nodes supported corresponding to optimal load.
- `available_resources[j]` = total number of free resources available at foreign agent j.
- `optimal_resources` = max_number_of_resources at foreign agent - optimal_number
- `resources_donated[l][j]` = number of resources donated by foreign agent j to foreign agent l.

```c
/* initialize the resources_donated matrix to 0 */
for(l = 0; l < no_of_fa; l++)
  for(j = 0; j < no_of_fa; j++)
    (resources_donated[l][j] = 0 )

/* load balancing algorithm */
for(l = 0; l < no_of_fa; l++)
  for(j = 0; j < no_of_fa; j++)
    if(available_resources[l] > optimal_number)
      /* high traffic */
      diff = no. of mobile nodes to be moved from the FA l so that foreign agent l has
      optimal number of mobile nodes and hence is at optimal traffic */
      diff = no. of mobile nodes supported by foreign agent l - optimal_number
      for(j = 0; j < no_of_fa; j++)
        if(j != l and foreign agent j is failure free)
          /* if the number of free resources that foreign agent j has is more than the optimal_resources value then the foreign agent can donate some resources to the heavily loaded foreign agent l */
          if(available_resources[j] > optimal_resources)
            /* Low Traffic */
            (resources_donated[l][j] = diff)
            (available_resources[l] -= diff)
            (available_resources[j] += diff)
          else
            /* only donate no. of resources which are in excess to optimal_resources */
            available_resources[j] -= optimal_resources
            available_resources[l] -= optimal_resources
            diff = diff - resources_donated[l][j]
        /* if all diff. number of mobile nodes moved to other foreign agents */
        break
      if(diff != 0)
        break
    } }
}
```

**Fig. 2:** The load balancing algorithm that distributes traffic amongst foreign agents.
The fault tolerance algorithm: Whenever an agent advertisement is not received by OARdM within a period of time it knows that the FA from which it did not receive the advertisement has failed. It then runs the fault tolerance procedure. Here, the no_of_affected_mobile_nodes is equal to the number of mobile nodes being serviced by the faulty FA. The fault tolerance algorithm initially calculates the number of free resources available with each of the failure free FAs. It also finds out the total resources available with all the failure free FAs.

\[
\text{total_free} = \text{sum of free resources available at all the failure free FAs.}
\]

Now the free fraction is calculated for each failure free FA, by the following ratio:

\[
\text{free_fraction}[i] = \frac{\text{free_resources}[i]}{\text{total_free}}
\]

The above ratio tells us what fraction of the total free resources does the FA have with it. Let the number of mobile nodes supported by the failed FA be \( Y_{\text{failed}} \). We evaluate the ratio:

\[
\text{no_of_resources} = \text{free_fraction[i]} \times Y_{\text{failed}}
\]

The above ratio finds out what is the corresponding number of mobile nodes with the failed FA for the fraction of free resources that a failure free FA has. Now if this number is greater than the free resources [I] then we donate all the resources available at I to share the mobile nodes of the failed FA. The number free_resources [I] becomes 0. If no_of_resources is less than free_resources [I] then we donate no_of_resources to support the failed FA.

We initialize, free_resources [q] = number of free resources with foreign agent q, total_free = 0, free_fraction [q] = fraction of total free resources with foreign agent q. Figure 3 describes our fault tolerance algorithm.

```
for each failure free foreign agent q
  { 
    free_resources[q] = max_number_of_resources_at_foreign_agent * no_of_mobile_nodes_supported_by_FA[q]
    total_free = total_free + free_resources[q]
  }
for each failure free foreign agent q
  { 
    free_fraction[q] = (free_resources[q] / total_free)
  }
for each failure foreign agent q
  { 
    no_of_resources_req = ceil(no_of_affected_mobile_nodes * free_fraction[q])
    if(no_of_resources_req > free_resources[q])
      { 
        donated[q] = free_resources[q]
        free_resources[q] = 0
        total_free = total_free - free_resources[q]
      }
    else
        { 
          donated[q] = no_of_resources_req
          free_resources[q] = free_resources[q] - no_of_resources_req
          total_free = total_free - no_of_resources_req
        }
  }
```

Fig. 3: Fault tolerance algorithm that decides the non-faulty FAs to share the load of faulty ones

Handoff in the proposed approach: Originally a handoff process switches the wireless connections of the mobile node maintaining its network connection when a mobile node moves from one radio coverage area to another. The handoff in the proposed approach does not introduce any location changes in the mobile nodes rather it only redirects the data requests of the mobile node (undergoing handoff) to a new foreign agent which is going to support its network connections in future. There can be two scenarios in the proposed approach in which we may require handoff. First, when the load balancing algorithm runs and decides to move mobile nodes from one foreign agent to another, second when a foreign agent has failed we have to move the mobile nodes of the failed foreign agent to failure free foreign agents. Figure 4 describes the handoff algorithm that can be a part of the OA and M network.

Failure recovery: In (Lin and Arul, 2003) whenever a FA recovers from failure, the failure recovery procedure brings the recovered FAs back into the system. In our approach, no separate failure recovery procedure is required. As and when the failed foreign agent recovers it comes into the system with zero load. So, when the load balancing algorithm runs it automatically moves mobile nodes from other heavily loaded foreign agents to the recovered FA.

EVALUATION AND SIMULATION RESULTS

We analyze our approach based on the worst-case scenario, where there is a constant arrival of heavy traffic at the FA rather than a Poisson input of traffic. In any network, the traffic is bound to be sometimes heavy and sometimes low and Poisson traffic is a good way of
Inputs:
Load Balancing Algorithm:
resources donated[i][j] = number of mobile nodes moving from foreign agent j to foreign agent i
Fault Tolerance Algorithm:
donated[i] = number of resources donated by foreign agent j to the failed foreign agent.

For each failure free foreign agent j
{
    let x - The number of mobile nodes to be moved to j given by resources donated[i][j] (if handoff due to load balancing) or donated[] (if handoff due to fault tolerance)

    if donated[j] or resources donated[i][j] != 0
    {
        1. move x nodes to FAj by making x visitor list entries in FAj.
        2. intimate the x mobile nodes of their new serving FAs.
        3. intimate the home agents of these x mobile nodes of the new serving FAs
           so that the HAs can service them.

        if handoff due to load balancing
        {
            intimate FAi (the agent from which mobile nodes are being taken off) to remove the entries corresponding to the x mobile nodes
            from its visitor list and free the care-of-addresses corresponding to the mobile nodes.
        }
    }
}

Fig. 4: The handoff algorithm

Simulating traffic in a real network. But there can be a really worse situation where there is a constant arrival of data requests at a FA. The blocking increases as the ratio λ_a/μ_a increases and any approach which works well for this worst-case situation is bound to perform well for a poisson traffic arrival. The symbols λ_a and μ_a are defined as below:

λ_a = constant arrival rate at FA (number of arrivals per unit time)
μ_a = average service rate at FA (The number of requests serviced per unit time is considered to be random as different data requests require different time to be serviced. So we take only the average service rate)

To simulate this constant arrival rate at the FA the following assumptions are made:

- Each mobile node generates traffic where the inter arrival time between two data requests made by the mobile node is distributed exponentially with mean 1/λ (λ = number of data requests per unit time produced by the mobile node).
- Any mobile node can simultaneously make more than one data request.
- If a RAN has X mobile nodes registered to it then the average incoming traffic per unit time from the mobile nodes to the RAN is X * λ (from assumption 1). We assume that the RAN forwards λ data requests of every mobile node per unit time to the foreign agents. Thus, the total number of data requests forwarded by a RAN per unit time to the foreign agents is X * λ (λ data requests of each mobile node).

Our simulations use a metric known as optimal load which is the amount of load a FA can handle without considerable blocking. Whenever we have optimal load condition our load balancing algorithm achieves the best distribution for the number of mobile nodes to be supported by a foreign agent which is given by,

X <= ceil [(X_1 + X_2 + X_3 + … + X_N) / N - 1] ….. (a)

From (a) and assumption 3 we have,

\[
\begin{align*}
\text{Number of arrivals per second at FA}_1 &= 1 - X * \lambda \\
\text{Number of arrivals at FA}_2 &= X * \lambda = \ldots \text{Number of arrivals per second at FA}_{N-1} &= X * \lambda \\
\end{align*}
\]

From (1) we can deduce that each FA has almost equal (if NX is not exactly divisible with N-1) number of mobile nodes. Each FA has to service X mobile nodes which is the optimal load thus the mobile nodes registered with all the FAs will get good service without considerable blocking hence the overall performance is improved. If the total load is greater than NX (optimal load) then the system works as a common system without load balancing. The proposed approach is applicable only till total load is less than NX.
We assumed below parameters for simulating the wireless network with Mobile IP that is based on the system model given in Fig. 1.

Exponential interarrival time from each mobile node to RAN with arrival rate \( \frac{1}{\text{sec}} \). Number of mobile nodes in the system = 12, Number of Foreign Agents = 3, Number of Home Agents = 1, Length of data queue at FA = 100, Time period for the load balancing algorithm to run = 5 sec. The maximum number of resources at foreign agents = 10 (10 care-of-addresses with a FA). The optimal load for a Foreign Agent was assumed to be 4 Mobile Nodes (< 50% of the maximum resources at a FA). Mobile node to foreign agent distribution for simulating the network without load balancing is FA 1: 7 Mobile Nodes, FA 2: 4 Mobile Nodes and FA 3: 1 Mobile Node. Figure 5 shows the results of our simulation run graphically and Table 1 shows with the values.

From the above results, we see that with load balancing approach when a foreign agent fails, the service rate is not reduced drastically. It remains almost same as to that before the failure. However, without load balancing, the service rate is reduced as the mobile node suddenly encounters more blocking.

CONCLUSION

This study presents an efficient approach to tolerate faults in Mobile IP in wireless networks. We have applied the load balancing technique to manage the load of all the non-faulty foreign agents with a uniform distribution. To transfer load from faulty foreign agents to non-faulty we have proposed a fault tolerant algorithm. Our approach does not require any recovery procedure. When a foreign agent restarts it automatically gets its share of traffic. We have shown simulation results that improves the pre-failure condition in a wireless network with Mobile IP and this improvement in turn results in an efficient fault tolerant approach to tolerate failure of foreign agents.

REFERENCES


