2012 International Conference on Information and Computer Networks (ICICN 2012)
IPCSIT vol. 27 (2012) © (2012) IACSIT Press, Singapore

Priority based Routing by Capacity Distribution

Subarno Banerjee [1], Supriya Roy [2] and P.K. Guha Thakurta [3]

[123] Department of Computer Science and Engineering
National Institute of Technology, Durgapur, India
[1]banerjee.subarno@gmail.com, [2]supriyaroy366@gmail.com, [3]parag.nitdgp@gmail.com

Abstract. In this paper, we propose a new Priority based Routing scheme to handle biased call request patterns efficiently. We introduce the idea of Capacity Distribution. Subsequently, a Routing Protocol has been proposed. The priority function is determined by two factors- Geographical context and Network usage patterns. The model has inherent Congestion control ability. The performance of the model is evaluated using extensive simulation techniques.

Keywords: Priority Routing, Capacity Distribution, Network Usage Pattern, Biased Call Pattern.

1. Introduction

In recent times, there has been an increasing motivation to switch from the wired networks to wireless networks due to its ease of access. With increasing number of mobile subscribers day by day, there is an ever increasing need of adapting the mobile communication systems. A major challenge for the network designers is to develop efficient call scheduling as well as routing techniques for the increasing traffic and reduce congestion at the same time to maintain the Quality of Service. The fundamental limitations of mobile communication are to be considered as bandwidth, power and channel capacity. Recent progress in wireless communication technology addresses a variety of challenges like efficient call routing, enhancing user experience with good scheduling algorithms, delay reduction, congestion control, energy efficiency, channel allocation, adapting to traffic and usage patterns, security and authentication issues.

The concept of Directed Routing was introduced in [1] wherein call requests are always forwarded to adjacent cells leading closer to the destination. This routing technique generates all possible paths within a range of path lengths and consequently selects the best path considering path length and congestion. The problems associated with previous models [3, 4] were minimized in this work. Propagation delay and distortion [6] were also minimized by forwarding calls to adjacent cells only. Also, the biased priority factors were amended. The performance of the model was consolidated in terms of average path lengths as well as with the variation of congestion probability.

Although the Routing Protocol in [1] was generic, the consideration of various priority factors was beyond the scope of the work. A random call request pattern is only an idealistic case. In reality, call request patterns are highly biased due to geographical context, user mobility patterns, network usage patterns, etc. Designing a routing scheme to efficiently handle all randomly possible requests is a difficult task. However, if the request pattern is known to have certain bias, this knowledge can be used to handle such call request patterns more efficiently and intelligently.

In this paper, we propose a Priority based Routing scheme based on studies of several biasing factors. The priority is indirectly enforced by the concept of Capacity Distribution. It is achieved by dynamic allocation of pair-wise channel capacity to control the behaviour of the routing scheme. It learns from the network usage statistics and utilizes this knowledge for efficient routing. Also, this technique has inherent Congestion control ability. The performance of the proposed scheme is evaluated by simulating the model.

2. Call Request Patterns

Call requests are never very random. They often follow trends and the patterns are influenced by several factors such as geographical context, user mobility patterns, network usage patterns, etc. The call request patterns can be characterized by studying network traffic traces. Large-scale trace-driven analysis has already been done by several researchers, as in [2]. We use their observations to characterize realistic call patterns. We have identified the following biasing factors that characterize call patterns:

2.1. Geographical Context

Call request patterns are highly biased due to geographical context of a locality. If a major city is located towards North with respect to an arbitrary station or cell, it is expected to receive a major share of its call requests to and from that direction. This is an intuitive factor and is not derived from trace analysis.

2.2. Temporal Bias

Call request patterns vary temporally. More call requests are expected to and from the industrial and commercial hubs during business hours; whereas more calls are requested towards residential suburbs during night hours. Evidently, call requests follow a well defined temporal pattern.

2.3. Mobility and Usage

Form the trace analysis in [2], we know that local users are more active than roaming users in usage. Also, about 66% of the users are stationary and the number of users decreases exponentially with their mobility range. 27% of the users are short-range roamers and only 6% are long-range roamers. The temporal behaviour of usage and mobility patterns has also been characterized in [2].

The above factors are used to target more realistic call request patterns and hence design the routing technique to handle them intelligently.

3. Capacity Distribution

The idea of Capacity Distribution is to enforce priority over routing by suitably allocating *Capacity*, the number of calls that can be forwarded simultaneously, between pairs of adjacent stations or cells. In the previous model [1], we had defined Capacity for a particular Base Station. We extend the model by defining Capacity between a pair of two adjacent stations. This will allow us to handle calls more intelligently and incorporate priority based routing.

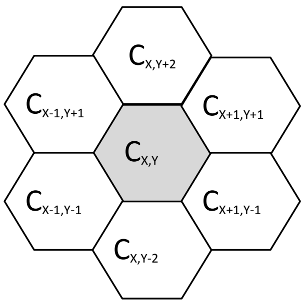


Figure 1: A Cellular Unit

Figure 1 shows the *Cellular Unit* for the cell $C_{X,Y}$ with its six adjacent cells. Each cell forms such a Cellular Unit. Each cell has six Capacity and Congestion values, one associated with each adjacent cell of its Cellular Unit.

Capacities-		Congestions-	
	$\varsigma(0,2)$		$\chi(0,2)$
	$\varsigma(1,1)$		$\chi(1,1)$
	$\varsigma(1,-1)$		$\chi(1,-1)$
	$\varsigma(0,-2)$		$\chi(0,-2)$
	$\varsigma(-1,-1)$		$\chi(-1,-1)$
	$\varsigma(-1,1)$		$\chi(-1,1)$

$\varsigma(a,b)$ is the capacity between $C_{x,y}$ and $C_{x+a,y+b}$;
$\chi(a,b)$ is the congestion between $C_{x,y}$ and $C_{x+a,y+b}$;
$a \in \{-1, 0, 1\}$; $b \in \{-2, -1, 1, 2\}$; $|a| + |b| = 2$

3.1. Determining Capacity Values

By suitably assigning the values for the six capacities ς, a Cellular Unit can be tuned to handle call requests by Priority based Routing. Some Priority based Routing models are discussed in [5].

3.1.1 Initialize by Geographical Context

Suppose, a large city is situated eastward from a cell. Obviously, more call requests will be received to and from the eastward direction. This Geographical Context factor gives us the initial capacity distribution. So, to incorporate this priority, $\varsigma(1,1)$ and $\varsigma(1,-1)$ will have larger values than the rest. This is the one time determination of the initial capacity distribution.

3.1.2 Periodic Update by Network Usage Statistics

Once, the model is in operation, the capacity distribution needs to be updated dynamically to adapt to the continuously changing traffic patterns. The new distribution is obtained by adding a weighted sum of the congestion values over a period of time to the existing capacity distribution. We propose the following simple update policies.

3.1.2.1 Periodic Proportional Sharing

The total Capacity C is shared proportionally to the congestion between cell pairs over a specific time interval τ. τ is called the Update Interval.

$$\varsigma(a,b)_{new} = C \times \frac{\chi(a,b)}{\sum \chi(a,b)}$$

3.1.2.2 Periodic Retentive Sharing

This is similar to the above update policy. In addition it retains a part of the initial capacity distribution.

$$\varsigma(a,b)_{new} = \rho \times \varsigma(a,b) + (1-\rho)\left[C \times \frac{\chi(a,b)}{\sum \chi(a,b)}\right]$$

Retentivity ρ is the fraction of initial capacity value that is retained and $0 < \rho < 1$.

3.2. Storing Capacity Values

Each cell has six Capacity and Congestion values for its Cellular Unit. However, these values overlap with corresponding values of adjacent Cellular Units.

$$\varsigma(a,b)_{X,Y} = \varsigma(-a,-b)_{X+a,Y+b}$$

So, they can be mutually shared for efficient storage and avoiding redundancy.

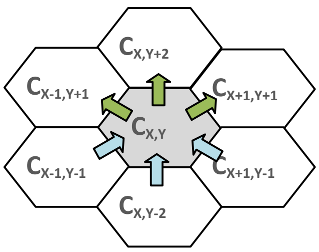


Figure 2: Storing Capacity Values

As shown in figure 2, we only store three values (shown by green arrows) per Cellular Unit. The other three (shown by blue arrows) are thereby stored at the adjacent Cellular Units.

3.3. Blocking and Borrowing

Capacity Distribution necessarily allots a Capacity value between a pair of adjacent cells. So, a call request is blocked whenever the Capacity between two adjacent participating cells is full, even if the total capacity of both are not fully used. This leads to underutilization of the channel capacity. We term this as *Blocking*.

Borrowing is one simple solution to prevent Blocking. Whenever a call request is blocked, one unit Capacity may be borrowed from the most underutilized cell pair to serve the request. However to maintain the actual capacity distribution, this borrowed capacity must be returned as soon as the call is ended. Also, unrestricted Borrowing might temporarily deviate the capacity distribution. So, there must be a limit on the extent of Borrowing. The better solution is therefore Restricted Borrowing which allows borrowing a limited part of one's Capacity in order to serve blocked call requests.

3.4. Congestion Control

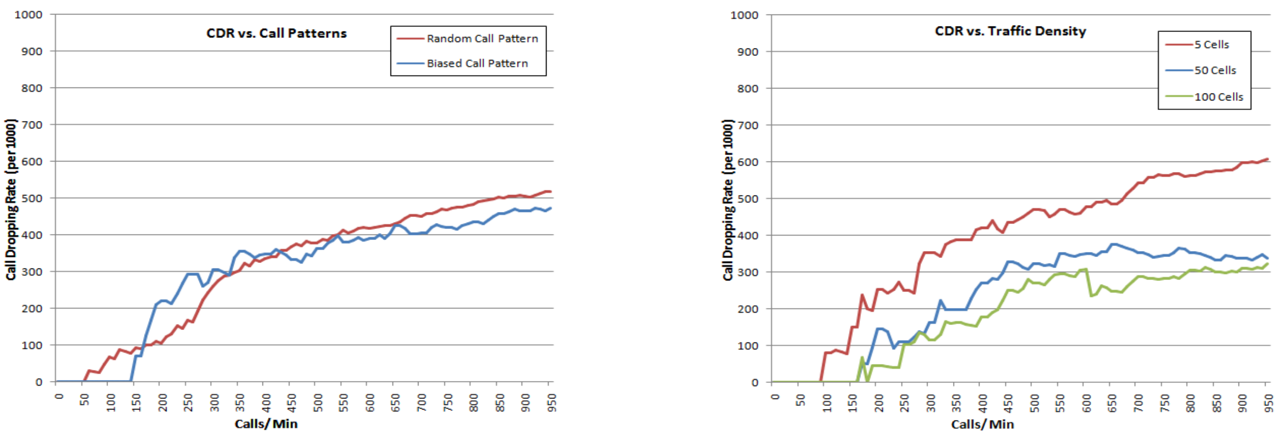
By Capacity Distribution, we dynamically allocate more capacity between those pairs of cells which are expected to serve more requests or are observed to be more active in call forwarding. Thereby, long duration calls between low priority cell pairs will block less number of frequent short duration calls between high priority cell pairs. This will reduce the *Call Dropping Rate*, the number of calls dropped per requests.

More adaptive the model is to the continuous changes in traffic and usage patterns, lesser is the number of call drops. Hence, this model has an inherent congestion control ability. Better congestion control can be achieved by properly setting the update interval τ, retentivity ρ or even by employing better heuristic update policies.

4. Simulation Results

We simulated the model by programming the behaviour functions of a small field of cells and then generating various call request patterns over it. We modelled a K×K cellular field and then routed call requests across it using the Coordinate based Directed Routing Protocol in [1]. The various parameters like field size, call request rates, call durations, geographical priority, capacity, update interval, retentivity, etc were configured suitably to simulate realistic conditions.

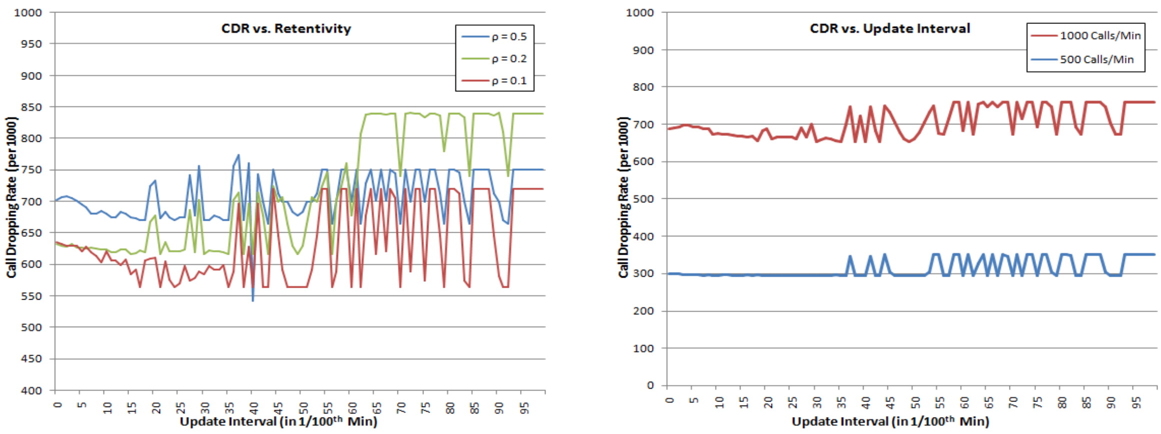
We evaluate performance in terms of Call Dropping Rate (CDR). In figure 3(a), the plots are actually aggregated values of several call request patterns. It is evident that biased call request patterns are handled more efficiently by the Capacity Distribution model. The CDR for biased call patterns rises steeply after the full capacity point (C=100) but tends to saturate faster than the random call patterns. Figure 3(b) shows the results of simulating the same traffic over various sizes of field. The CDR is higher in a field with denser traffic as expected.



(a) Call Patterns (b) Traffic Density

Figure 3: CDR vs. Traffic

Next, we explore the nature of adaptability of the model by varying the capacity updation with different update intervals (τ) and retentivities (ρ). The effect of changing these two parameters is observed to be dependent on the call patterns and call request rates as well. From figure 4(a), decreasing the update interval τ decreases the CDR in general. This means that the Capacity Distribution model responds quickly to the changing usage and traffic patterns. However, for higher call request rates, the best CDR is achieved at intermediate values of τ. So, a longer update interval is desirable for properly adapting to the frequent changes in a high call request rate. Also, the variation of CDR with retentivity ρ, as in figure 4 (b), is erratic in nature. A call request pattern phenomenally attains a low CDR at particular values of ρ. Hence, a suitable combination of both τ and ρ is required to adapt optimally to a call request pattern.



(a) Update Interval (b) Retentivity

Figure 4: CDR vs. Adaptability

5. Conclusions

Capacity Distribution is a new dynamic approach towards Priority based Routing. We identified some key biasing factors that characterize realistic call patterns and accordingly selected the priority factors. Since it responds to varying call request patterns, proper tuning and selection of update policy is required for optimal performance. The model achieves better performance with respect to CDR while maintaining simplicity and adaptability. Moreover, the model also has inherent congestion control ability.

We are currently extending the model for supporting temporal biases in call request patterns and other priority factors. Better adaptive update policies can also be designed for specific target call request patterns.

6. References

[1] P. K. Guha Thakurta, Supriya Roy, Subarno Banerjee, “Coordinate based Directed Routing Protocol”, International Conference on Network Communication and Computer, New Delhi, March 2011.

[2] Zhichao Zhu, Guohong Cao, Ram Keralapura and Antonio Nucci, “Characterizing Data Services in a 3G Network: Usage, Mobility and Access Issues”e, IEEE International Conference on Communications (ICC), July 2011.

[3] P.K.Guha Thakurta, Rajarshi Poddar and Subhansu Bandyopadhyay, “A New Approach on Co-ordinate based Routing Protocol for Mobile Networks”, IEEE Advanced Computing Conference, February 2010.

[4] P.K.Guha Thakurta and Subhansu Bandyopadhyay, “A New Dynamic Pricing Scheme with Priority based Tree Generation and Scheduling for Mobile Networks”, IEEE Advanced Computing Conference, March 2009.

[5] Babak Farzad, Neil Olver and Adrian Vetta, “A Priority-Based Model of Routing”, Chicago Journal of Theoretical Computer Science, 2008.

[6] Behrouz A. Forouzan, “Data Communications and Networking”, 4th edition, McGraw-Hill Forouzan Networking Series, 2007, ISBN: 0072967757.