I.J. Computer Network and Information Security, **2011, 2, 1-11**
Published Online March 2011 in MECS (http://www.mecs-press.org/)

Transaction-based QoS management in a Hybrid Wireless Superstore Environment

Shankaraiah and Pallapa Venkataram
Protocol Engineering Technology Unit, Electrical Communication Engineering
Indian Institute of Science, Bangalore, India.
Email: {shankaraiah,pallapa}@ece.iisc.ernet.in

Abstract—Hybrid wireless networks are extensively used in the superstores, market places, malls, etc. and provide high QoS (Quality of Service) to the end-users has become a challenging task. In this paper, we propose a policy-based transaction-aware QoS management architecture in a hybrid wireless superstore environment. The proposed scheme operates at the transaction level, for the downlink QoS management. We derive a policy for the estimation of QoS parameters, like, delay, jitter, bandwidth, availability, packet loss for every transaction before scheduling on the downlink. We also propose a QoS monitor which monitors the specified QoS and automatically adjusts the QoS according to the requirement. The proposed scheme has been simulated in hybrid wireless superstore environment and tested for various superstore transactions. The results shows that the policy-based transaction QoS management is enhance the performance and utilize network resources efficiently at the peak time of the superstore business.

Index Terms—Hybrid Wireless Network, Superstore, Policy-based Management, Policy Engine, QoS

I. Introduction

Quality of Service (QoS) one of the major factors needs to be deployed for efficient resource usage in the wireless service provider networks. Various research groups are investigating the automation and the simplification of the management process using policies as a promising solution to address the needs of transaction-based QoS management. These policies are rules for how network resources are used and how users are serviced. These policies define a set of rules that guide the behavior of network components. Once defined by network administrators, these policies are translated into network-level or device-level policies and stored in a Policy Repository (PR).

Modern hybrid wireless superstore networks have greatly enhanced their capability and complexity. As a result, more and more bandwidth-consuming and time-critical transactions are deployed in the same network with superstore applications, which makes it difficult to manage computing and network resources to meet user transaction QoS requirements.

In a superstore wireless environment, an increasing number of users are likely to adopt mobile transactions. These transactions may have very distinct requirements and some of them may require a rich amount of network resources and/or bounded delays. The main characteristic of transactions involving a financial value is the criticality of its completion with deadline. The unique requirements of mobile transactions suggest the introduction of new metrics for QoS.

Many of the critical transactions, such as the one involving financial values are likely to be atomic and require low response times. Therefore, the QoS of superstore transactions should include the parameters of transaction completion probability and transaction response time. Both transaction completion probability and response time are functions of resources allocated at the transaction level.

The rest of the paper is organized as follows. Section II discusses the related work. Section III describes the Policy-based transaction aware QoS management architecture and its components. Section IV gives the analytical model of the proposed work. The section V gives simulation and performance evaluation. Finally, Section VI concludes this paper and also shows the future work to be considered.

II. Related works

Wireless network operators are more interested in the use of 802.11-based wireless technologies (802.11a/b/g/n) to provide access to these services at so-called hotspots (Superstore, company, hotel lobbies, cafes, etc.) [1]. This is because WiFi use license-free radio spectrum to provide low-cost, easily deployable, high-data-rate wireless services. In these hotspots, WiFi technologies provide nomadic high-speed wireless access to current wired Internet Protocol (IP)-based networks.

Wireless network operators [2], [3] can assist quality of service (QoS)-sensitive IP applications like voice over IP (VoIP) over the UMTS packet switched (PS) domain by using the Session Initiation Protocol (SIP)-based IP multimedia subsystem (IMS). Recently, the 3GPP is extending the policy based QoS control architecture [4], [5], [6] for UMTS IMS services to satisfy the end to-end QoS requirements of other application services in the UMTS PS domain.

The Internet Engineering Task Force (IETF) has described a policy framework [7] within which sets of policy rules defined in the form of policy models [8] are converted into network/device configurations in an administrative domain. The policy rules are stored in the policy repository from which the policy decision point (PDP) Or Policy Decision Function (PDF), retrieves the appropriate policy rules in response to policy events that are triggered by the transactions QoS services.

Policy Enforcement Point (PEP) is place on a server that enforces policies for admission control and policy decisions in response to a transaction request from a user wanting to access a resource on a network server. The PEP is a component of policy-based management. When a user tries to access a service on a server that uses policy-based access management, the PEP will describe the user's transaction attributes to other entities on the system.

The PEP will give the Policy Decision Point (PDP) the job of deciding whether or not to admit the transaction based on the description of the transactions attributes and available bandwidth. Transaction policies are stored on the server and are analyzed by the PDP. The PDP translates the acquired policy rules into a set of transaction QoS mechanism configuration actions based on the capabilities of the PEP and the current network conditions. The PEP then executes these PDP-supplied actions to handle the triggering policy events in accordance with the requested transaction QoS services.

End-to-end communications are likely to contain multiple administrative domains controlled by same network operators. To provide a consistent end-to-end service in a multi-operator multi-domain environment, the authors of [9] proposed a hybrid policy architecture for UMTS IMS, in which the hierarchical architecture is employed within a single operator's multi-domain network, and the peering architecture interconnects multiple operator's networks.

In the paper [10], a policy-based transaction-aware QoS management architecture for a hybrid wireless superstore Environment has been presented. The scheme operates at the transaction level for the downlink transaction QoS management. A policy for the estimation of QoS parameters, like, delay, jitter, bandwidth, availability, packet loss for every transaction before scheduling on the downlink transaction has been followed.

Different wireless environment may associate with different administrative domains and different degrees of internetworking [11], [12], [13], [14]. The security capabilities and policies may differ between public, superstore, corporate and residential networks. The policy-based transaction QoS management architecture extended from the policy control architecture presented in [15], [16]. A superstore [17], [18], [19], [20] owner may own a hotspot provided within the area of his/her superstore. A broadband service provider may coordinate multiple hotspot owners and resell the broadband services to customers.

The objective of this paper is to propose how policy-based transaction QoS can be realized over hybrid wireless superstore environment. A policy-based QoS architecture for the WiFi domain created within the policy framework defined by the IETF is proposed. We discuss how the policy-based QoS architectures of the GSM and WiFi domains can be integrated in the GSM-WiFi networking scenarios.

III. PROPOSED POLICY-BASED TRANSACTION AWARE QOS MANAGEMENT

Our QoS management protocol is fully centralized, and is based on the ideas of client-server interaction. Each mobile terminal is connected to a server via WiFi or GSM. The query based mechanism is invoked when an incoming request the server that received it.

The transaction supporting middleware, receive the transaction and classify the transaction into different categories in each wireless domain and the QoS requirements are divided into segments on individual domains.

The Policy Engine and QoS Monitor cooperate each other to meet the transaction QoS requirements of its domain. The policy engine consists of Policy Management Tool (PMT), PDP (policy decision point), PR(policy repository) and PEP(policy enforcement point). PMT offers a policy configuration interface to create/edit/modify policies. PR is a policy container. PDP is responsible to make decisions for policy requests launched by PEP, and PEP is the place where the policy is executed.

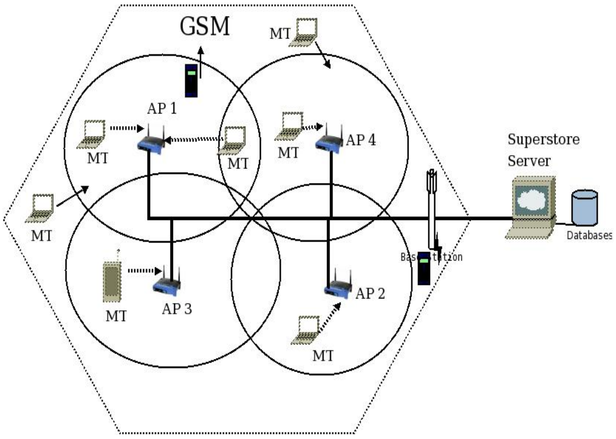

Fig 1: Hybrid Wireless Superstore network architecture

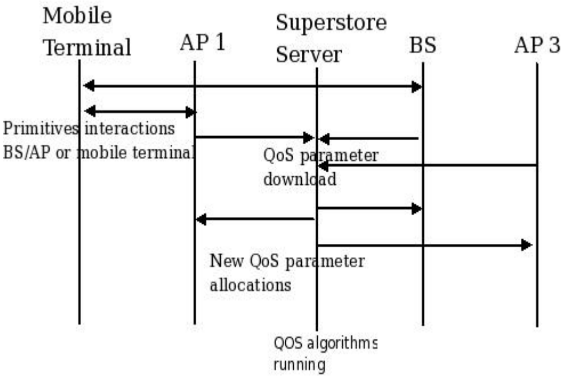

Fig 2: Protocol interaction in a Hybrid Wireless Superstore Network.

Figure 2 shows the idea we develop in this article. The mobile terminal reaching the WiFi must connect with access point. The connection procedure is always initiated by the mobile terminal and it can associate with any access point or BS which are present in that instant as shown in Fig 1. Let's suppose that AP 3 is chosen by new mobile terminal. The load distribution among the APs will be highly uneven, results in performance degradation perceived by the other mobile terminal connected with AP 3. The QoS contracts may be violated. The available resource of the hybrid wireless network depends on the number of active mobile terminals and their traffic.

To achieve balancing in terms of QoS offered to the mobile terminals among APs and BS. We have to compute a balancing algorithm each time a new arrival of new access points or BS or the mobility of existing mobile terminals. Our algorithm has to find the best APs or BS which offers the best QoS for user's transactions. Thus, we have to get information on connected AP, traffics and user's QoS required. This information has to be in data base. The superstore server periodically downloads a set of QoS parameters from each access points or BS. It runs the algorithms at server to find the best network which provide the requested QoS.

A. Policy for QoS management

Policies or policy rules are typically expressed as condition of action pairs, e.g. IF Transaction = financial AND NAME = Money transfer THEN bandwidth > 10Mbit/s, jitter=0.05ms, packet loss=5packets/s, etc

A simple example of expansion may look like this:
<?XMLversion = 2.1 encoding = UTF − 8? >
These information are sent to Transaction sensor to generate transaction QoS parameters, by mapping the Transaction Class(TC), QoS Class(QC), and approved User Class(UC) to certain Transaction QoS Metrics(TQM). The mapping can be presented by abstract expression:

$$(TC_i, QC_j, UC_k) \to TQM_n \qquad (1)$$

The transaction specific TQM may look like this:

```
\begin
<?XMLversion = ”2.1”encoding = UTF − 8 > xmlns :
xsd = “http”; //www.w3.org/2001/XMLSchema′ ′ >
< T ransactionInstancename = T ransactionname >
< T ransactionT ype > Servertype </TransactionType >
< userT ype > usergroup < /userT ype >< QoSSpe >
< Response > res < /Response >
< Bandwidth > bw < /Bandwidth >
< Availability > av < /Availability >< /QoSSpe >
< /T ransactionInstance >
< /xsd : schema >
end
```

In hybrid wireless superstore environment, delay can be measured in either one way or round trip delay. Delay can be estimated by knowing the starting time of the transaction sent and time at which response can be obtained. Let t_{start} be the starting time of the transaction sent and t_{end} is the time at which response is obtained. Then one way delay(D) can be estimated by

$$D = \frac{t_{end} - t_{start}}{2} \qquad (2)$$

Similarly the packet loss can be estimated by knowing sequence number of each packet sent and received the acknowledgment in each transaction. Let S_n be the number of packets sent and S_r be the number of packets received the acknowledgments. Then percentage of packet loss can be estimated in hybrid wireless network is

$$\%\ packet\_loss = \left(\frac{S_n - S_r}{S_n}\right)100 \qquad (3)$$

We can estimate the jitter as follows: the value of packet spacing at the receiver compared with packet spacing at the sender for a pair of packets. For example, if S_i is the sending time for packet i, and R_i is the receiving time for packet i, then for two packets i and j, inter-arrival jitter $D(i, j)$ may be expressed as:

$$D(i, j) = (R_j - S_j) - (R_i - S_i) \qquad (4)$$

The value of inter-arrival jitter can demonstrate the packet by-packet delay. If D is greater than zero, it means that the transmit time of packet j is longer than packet i. We can assume packet j is queued in a while. Depending on this phenomenon, jitter ratio is defined to approximate the ratio of queued packets. The jitter ratio is defined as the ratio of queued packet. When the increasing queue reaches the maximum limit of the buffer, the following arriving packets at the server will be dropped. Therefore, the ratio of packets dropped will be approximated as the ratio of queued packets. The parameter that we call Jitter Ratio can be defined as follows:

$$J_r = \frac{D}{R_i - R_{i-1}} \qquad (5)$$

Jitter ratio is an important to determine whether the packets are queued or not. We will apply jitter ratio estimation in QoS mechanism to enhance its performance in hybrid wireless superstore environment.

We have planned to keep packet loss, jitter and delay under consideration to maintain the QoS of each transaction. In order to maintain the QoS, we are allocating Bandwidth (BW) to each transaction.

Policy-based Transaction QoS Management (PTQM) is a management paradigm that uses policies for Transaction QoS management. Policies are rules that govern a transaction behavior, usually implemented in a form of **if(condition) then (action)** sentences.

The PTQM paradigm allows an abstraction of the superstore administers specific configuration details through use of a policy manager element. The automation character of the transaction management approach reduces the equipment management effort. This is of special importance in large superstore transaction management scenarios like a communication operator network. Moreover, during the configuration process, it avoids possible human error due to the repetitive actions performed by human operators. Furthermore, central implementation of the configuration actions eases implementation of conflict detection in the configuration elements as well as implementation of transaction support for the configuration actions.

B. Architecture of PTQM

Figure 3 depicts the architecture of PTQM. We assume simply that a superstore application deployed in hybrid wireless networks is made up by a set of transaction services. In this section, we will present the transaction supporting middleware, Policy Engine and QoS Monitor successively

C. Functioning of PTQM

The architecture of policy-based transaction-aware QoS Management is described by considering individual blocks.

Transaction supporting middleware

Transaction Supporting Middleware contains Service Register, Transaction sensor and Transaction QoS classification component. In order to present more precise transaction QoS requirements, Transaction application must extend their service template description document. We could extend the IETF service template to include such information as service type and transaction QoS level requirements.

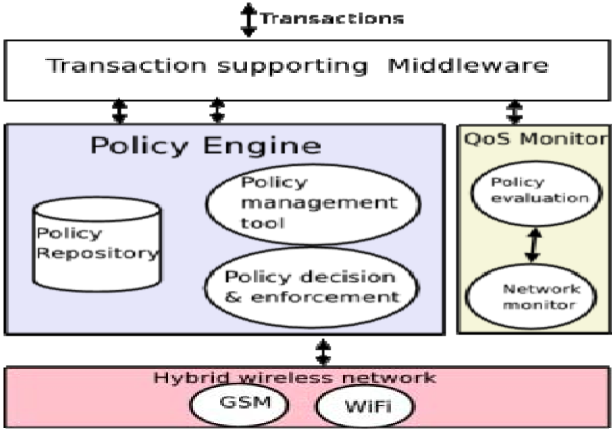

Fig . 3: Architecture of PTQM.

Transaction classification

The main objective of this module is to classify the given transactions into different groups. The grouping is done based on the parameters: type of data involved in each transaction, time of operation, type of device used, location of operation, type of transaction, mobility and type of traffic generated by each transaction. The transaction QoS classification component will partition the transaction QoS requirements into smaller categories on every domain that the transaction covers. The classification should also take into account the performance metrics of specific networks and must meet the transaction requirements as a whole.

Policy Engine

Transaction QoS requirements are divided into segments on individual domains, where Policy Engine and QoS Monitor cooperate to meet the transaction QoS requirements of its domain.

Policy is a definite business rule or goal to guide and determine present and future decisions, which can be understood and executed within a particular context[21] [22]. The Policy-Based Management(PBM) has four core functional components: Policy Management Tool (PMT), Policy Repository (PR), Policy Decision Point (PDP), and Policy Enforcement Point (PEP).

PMT offers a policy configuration interface to create/edit/change policies. PR is a policy container. PDP is responsible to make decisions for policy requests launched by PEP, and PEP is the place where the policy really executed, usually embedded in network elements. The policy Engine is based on PBM, and we make some necessary extensions to support transaction awareness. Before we elaborate on the framework of Policy Engine, we would like to present the Transaction Information Model first. According to the PBM framework, the network administrator uses PMT to edit superstore business policies. In the architecture, however, the PMT gets business policies from transaction supporting middleware as well, validates and translates them into concrete policies, then finally distributes them in the corresponding wireless domain.

The policy validation will be achieved in terms of syntax and semantics. Syntax validation is a kind of form validation, while semantics validation mainly care with policy conflicts detection and resolution. Methods can be various to detect conflicts, depending on what policy specification languages are in use. As in our prototype, the method of hyper-dimensional space is adopted [13].

A transaction QoS requirement is expected to be simple, independent of specific QoS mechanism. An example of this could be “financial transaction“between users from Subnet A and users from Subnet B that should have bandwidth larger than 10mbps and jitter less than 50ms. Such business policies are easy to understand and configure, but not easy for network/devices to read and execute. So policy translation is needed here to bridge the gap between human-friendly business policies and machine-friendly technique policies. As the core of PMT, the policy translation component plays a key role in improving transaction-awareness adaptability of networks.

The technique policies are then distributed and stored in the PR, where they could be retrieved or modified later. The PR can be implemented in the superstore Server.

The PDP and PEPs are the core of PTQM during runtime, which perform policy decision, evaluation and enforcement. To do this, it can work in the hybrid, to deal with different QoS mechanism of networks. As this part is not notably differentiated from PBM, we will not describe it in details.

QoS Monitor

The policy engine partly solves the problem of transaction QoS requirements capture, translation and policy execution. The QoS Monitor consists of the Network Monitor, Performance Database and Policy Evaluation. On one hand, the Network Monitor uses active or passive measure methods to collect network performance metrics (NPM), such as packet loss rate, delay, jitter throughput, etc.

On the other hand, Policy Evaluation compares the transaction QoS metrics (TQM) with objectives that has been prescribed as policy. As we know, not all the measured NPM parameters can be directly used in policy evaluation; therefore parameter translation is needed to reveal TQM. As a simple example, we would assume that TQM be expressed by as follows:

$$TQM = \{T_{avail}, T_{res}, T_{jitter}, T_{utility}, T_{bw}, T_{profit}\} \quad (6)$$

In this expression, T_{avail} means service availability, which rests on network connectivity and round trip delay, T_{res} means transaction response time, which rests on network delay, etc.

In the same way, NPM can be expressed as:

$$NPM = \{N_{traffic}, N_{delay}, N_{loss}\} \quad (7)$$

Therefore we can establish a certain mapping between elements in TQM and elements in power set of NPM as shown in Fig 4.

By this means, the TQM information could be measured and monitored, so that we can perform a policy evaluation. We can also store this transaction network information in performance database, from which we may conclude and draw out policy translation rule to fill the rule base. The table I describes the QoS parameters of GSM and WiFi networks. The function of PTQM module is given in the Algorithm 1.

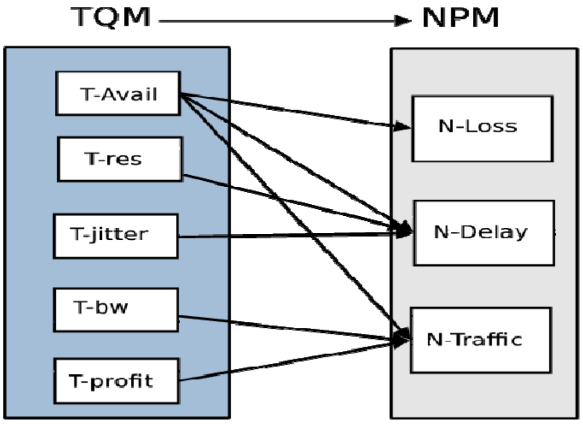

Fig 4: TQM and NPM mapping

TABLE I
QOS PARAMETERS OF GSM AND WIFI NETWORKS

Parameters	GSM	WiFi.
Latency	700ms-4s	20-25ms
Jitter	4s	8ms
Packet error rate	1-2%	1-10%
Maximum Available BW	12Kbps	11Mbps

IV. Analytical Model

We consider a hybrid wireless networks consisting of GSM and WiFi for superstore applications. In every X seconds (where X is a constant interval), we are collecting the number of transactions. These transactions include handoff (h) transactions, transactions waiting (w) in the queue and new transactions (n) arrived during an interval X. The transaction model, on the other hand, depends on the transaction traffic, which determines the resource available for transaction traffic in an interval. Transactions are exponentially distributed with average duration $\frac{1}{\mu}$.

Let $R = \{r_1, r_2, r_3, r_3, ...r_N\}$ be the maximum amount of resources available in the hybrid wireless network. The resources R should be shared between N customer shopping transactions. The handoff transaction has highest priority when compare to new transactions. Each transaction requires r_j amount of resources and have an exponentially distributed service time with mean $\frac{1}{\mu_j}$.

The number of transactions in the interval X is random variable Y. Let λ be the mean arrival rate of transaction in an interval X. Let λ_n and λ_h be the mean arrival rate of new and handoff transaction respectively. The transaction arrival rate in GSM or WiFi is given by $\lambda = \lambda_n + \lambda_h$. Transaction exponentially distributed with an average transaction interval time in GSM and WiFi are $\frac{1}{\mu_g}$ and $\frac{1}{\mu_w}$ respectively. The Poisson process that describes a collection of arrivals for which the inter arrival times are independent and exponentially distributed and probability that total number of new transactions arrived in an interval X is given by

$$P(Y_n = x_n) = \frac{(\lambda_n X)^{x_n} e^{-\lambda_n X}}{x_n!} \quad (8)$$

Algorithm 1 PTNQM algorithm

```
Begin
Collect all transaction in the interval X.
while Not end of customer transaction do
    Accept the transaction details (TDs), Send TDs ) $\Rightarrow$ TransactionClassifier(TC)(T).
    if Any request from TC(T) to find QoS subrequirements then
        Fetch them from customer.
    end if
    QoSsubrequirements $\Leftarrow$ TransationClassifier(T).
end while
Send TDs and QoS requirements $\Rightarrow$ Policy Engine.
if Any information require from TD then
    Fetch from TD.
end if
Send TDs and QoS requirements $\Rightarrow$ PMT and PDP, Get the policy $\Leftarrow$ PR
Pass the TDs $\Rightarrow$ PEP
if Transaction=Y then
    BW > 10Mbps, Jitter < 5ms,delay < 0.05sec and Loss < 5packets;
else
    if Transaction=Y1 then
        BW > 1Mbps, Jitter < 2ms,delay < 0.25sec and Loss < 2packets;
    else
        if Transaction=Y2 then
            BW > 2Mbps, Jitter < 1ms,delay < 0.2sec and Loss < 3 packets;
.....
```

```
end if
end if
end if
QoS monitor
while Not end of transaction do
Accept the NPM, Send NPM $\Rightarrow$ PR.
if Any request from NPM to find QoS performance metrics then
Fetch them from networks.
end if
PDP, PR $\Leftarrow$ NPM
Map the NPM with TQM and evaluate the policies
end while
if Customer transaction is end then
Disconnect the client session.
end if
End
```

The probability that total number of handoff transactions arrived in an interval X is given by

$$P(Y_h = x_h) = \frac{(\lambda_h X)^{x_h} e^{-\lambda_h X}}{x_h!} \qquad (9)$$

The probability for total number of downlink transactions arrived in an interval X is given by

$$P(Y = N) = \frac{(\lambda X)^{N} e^{-\lambda X}}{N!} \qquad (10)$$

The average number of transactions in the interval X is given by

$$E[Y] = \lambda_n * X + \lambda_h * X + q(Tr) \qquad (11)$$

Where q (Tr) is total number of transactions remains in the previous interval without getting enough resource and its value is zero initially.

A. Transaction categorization

The transaction categorization is done based on type and importance of that transaction in superstore business applications. Let there be's' transaction types, N downlink transactions may be distributed between 's' types. Let $S_r (1 \le r \le s)$ be the random variable which counts the number of transactions of type 'r' among the given N transactions. Let λ_r be the average rate of occurrence of each type of transaction and n_r represents the number of transaction of type 's'. The probability of number of downlink transactions in each type is a Poisson and given by

$$P(S_r = n_r) = \frac{(\lambda_r X)^{n_r} e^{-\lambda_r X}}{nr!} \qquad (12)$$

The expected value of each type of transaction is given by

$$E(S_r) = \sum_{r \in Type\_s} \lambda_r X \qquad (13)$$

We arrange the downlink transactions according to their type and importance in the superstore business as shown in the Table II. The 's' types of transaction and there are N transactions in the duration X. For simplicity, we are dividing the type of transactions into four levels and distributing all the transactions into four level queues based on their types. We are considered only four levels in order to reduce the complexity of the systems. Let m_0, m_1, m_2 and m_3 be the number of transactions of level 0, level 1, level 2 and level 3. Let M_l be the random variable which indicates number of transactions in each levels. The probability for number of transactions belong to level 'l' is

$$P(M_l = m_l) = \frac{(\sum_{r \in level\_l} \lambda_r X)^{m_l} e^{(\sum_{r \in level\_l} \lambda_r X)}}{m_l!} \qquad (14)$$

The total average transactions in each level is given by

$$E[M_l] = \sum_{r \in level\_l} \lambda_r X \qquad (15)$$

where 'l' is a index value and it can take 0, 1, 2 and 3.

$$N = m_0 + m_1 + m_2 + m_3; \qquad (16)$$

B. QoS Parameter Estimation

Wireless QoS estimation scheme is applicable in active mode of operation for an Access Points (AP)/Base Station (BS) to ensure the QoS properly. The QoS estimation is done only for new transactions but for the handoff transactions previous resource allocation is considered. This algorithm runs at superstore server to estimate the resource requirements of downlink transactions that has been received successfully with a small time interval. The transaction response time can be calculated by knowing the average time required to send one packet from server to destination device in a hybrid wireless network. The receiving time of the response message (rb_i) is used as the packet interval end point. Therefore, the duration of the packet interval time (PIT) is computed by

$$PIT_j = rb_j - st_j \qquad (17)$$

Where st_j is the packet interval start time $st_j = t_0$ for i = 0.

The transactions data size (tds) can be calculated by knowing the number packets in each transaction.

$$tds_j = m_j PL \qquad (18)$$

Where m_j is the number of packets in each transaction and PL is the packet length. The packet length may vary from 500 to 1500 bytes. Now bandwidth requirements of the j^{Th} transaction can be estimated by using the equation

$$b_j = \frac{tds_j}{(m_j * PIT_j + latency)} \qquad (19)$$

for j = 0, 1, 2 ....

Where 'latency' is the minimum delay of the corresponding network. The total bandwidth required by all the transactions in the interval X is given by

$$B_{total} = \sum_{j=1}^{N} b_j \le B \qquad (20)$$

The available bandwidth in the networks is given by

$$B_{Available} = B - \sum_{r=1}^{S} b_r \qquad (21)$$

Let S_n be the number of transaction packets sent and S_r be the number of packets received the acknowledgments. Then percentage of packet loss can be estimated in hybrid wireless network is

$$\%\ packet\ \_\ loss = \left(\frac{S_n - S_r}{S_n} \right) 100 \quad (22)$$

We can estimate the jitter as follows: the value of packet spacing at the receiver compared with packet spacing at the sender for a pair of packets. For example, if S_i is the sending time for packet i, and R_i is the receiving time for packet i, then for two packets i and j, inter-arrival jitter $D(i, j)$ may be expressed as:

$$D(i, j) = (R_j - S_j) - (R_i - S_i) \quad (23)$$

The value of inter-arrival jitter can demonstrate the packet by-packet delay.

TABLE II

TRANSACTIONS CATEGORIZATIONS

Transaction Types	*Number of transactions*
1	λ1.X
2	λ2.X
3	λ3.X
…	…
S	λs.X

C. Performance Parameters

Bandwidth Utilization(BU)

Let b_r be the bandwidth required for each transaction of type r {r=1,2,... s}. Then average bandwidth required for each transaction is given by

$$BU = \frac{\sum_{r=1}^{s} \lambda_r.X.b_r}{N} \qquad (24)$$

For Level 3.

$$BU_3 = \frac{\sum_{r=\frac{3s}{4}+1}^{s} \lambda_r.X.b_r + \lambda_h.X.b_r + q(Tr)b_r}{m_3} \qquad (25)$$

For Level 2.

$$BU_2 = \frac{\sum_{r=\frac{s}{2}+1}^{\frac{3s}{4}} \lambda_r.X.b_r}{m_2} \qquad (26)$$

For Level 1.

$$BU_1 = \frac{\sum_{r=\frac{s}{4}+1}^{\frac{s}{2}} \lambda_r.X.b_r}{m_1} \qquad (27)$$

For Level 0.

$$BU_0 = \frac{\sum_{r=1}^{\frac{s}{4}} \lambda_r.X.b_r}{m_0} \qquad (28)$$

Average transaction delay

The average transaction queuing delay is defined as the ratio of total service time from the first transaction to transaction m_j for priority queue 'l' to the total number of transaction within update interval X for priority queue 'l'.

Let T_{st} = Total service time from the first transaction to transaction' m_l ' for priority queue 'l'. and N= Total number of transaction within update interval X for priority queue 'l'. Then total average transaction queuing delay can be denoted as D and it is given by

$$D = \frac{T_{st}}{N} \qquad (29)$$

Let $\frac{1}{\mu_r}$ be the mean service time required for each transaction of type r {r=1,2,... s}. Then total average transaction delay required for each transaction is given

$$D_{total} = \frac{\sum_{r=1}^{S} \lambda_r.X.\frac{1}{\mu_r}}{N} \text{ seconds} \qquad (30)$$

Total average delay in level 3 is D3

$$D_3 = \frac{\sum_{r=\frac{3s}{4}+1}^{S} \lambda_r.X.\frac{1}{\mu_r} + \lambda_r.X.\frac{1}{\mu_r} + q(Tr)\frac{1}{\mu_r}}{m_3} \qquad (31)$$

Total average delay in level 2 is D2

$$D_2 = \frac{\sum_{r=\frac{s}{2}+1}^{\frac{3s}{4}} \lambda_r.X.\frac{1}{\mu_r} + \sum_{r=\frac{3s}{4}+1}^{S} \lambda_r.X.\frac{1}{\mu_r}}{m_2} \qquad (32)$$

Total average delay in level 1 is D1

$$D_1 = \frac{\sum_{r=\frac{s}{4}+1}^{s} \lambda_r.X.\frac{1}{\mu_r}}{m_1} \qquad (33)$$

Total average delay in level 0 is D0

$$D_0 = \frac{\sum_{r=1}^{s} \lambda_r.X.\frac{1}{\mu_r}}{m_0} \qquad (34)$$

Transaction blocking probability

Blocking probabilities for each priority level 'l'(i.e.'l'=0,1,2,3) are obtained recursively starting from the highest priority level 'l'. For each priority level we model the system as M/G/ n_r / n_r . Where n_r is a maximum number of transactions of type r, which can be established with a maximum amount of bandwidth each priority level can utilize BU_l and is equal to

$$n_r = \frac{BU_l}{b_r} \qquad (35)$$

The blocking probability is obtained from the Erlang-B formula

$$P_b = \frac{\dfrac{\rho_r^{n_r}}{n_{r!}}}{\sum_{r=1}^{n_r} \dfrac{\rho_r^{n_r}}{n_{r!}}} \qquad (36)$$

where $\rho_r = \dfrac{\lambda_r}{\mu_r}$.

The average number of transactions of priority level 'l' in the system is obtained from Littles formula

$$E[m_l] = \rho_r (1 - P_b) \qquad (37)$$

System utilization ρ_r for priority level 'l' is equal to $\rho_r = \dfrac{E[m_l]b_r}{BU_l}$. The average number of transactions of priority level 'l' in the system, which are blocked is obtained from Littles formula

$$E[blk_r] = \rho_r P_r$$

(38)

The number of transactions remain in the queue without getting service is $q(Tr) = E[blk_r]$

V. Simulation

We consider a superstore wireless simulation environment to test the proposed policy-based QoS management scheme. It consists of a WiFi network, which is used to consolidate all customer transactions details and transmit to the local server. The mobile customer can be connected to superstore via GSM network. In a superstore, the request transactions generally take few kilobytes, whereas the response transaction can be multiple megabytes. Hence it is important to consider the downlink transactions rather than up link transactions. For all customer launched transactions the server allocates resource based on their priority. The following assumptions are considered in simulation.

- We consider Hybrid Wireless Network(HWN) consisting of two resource access networks: GSM and WiFi(e.g., IEEE 802.11b).
- The arrival of new/handoff transaction in a cell forms a Poisson process.

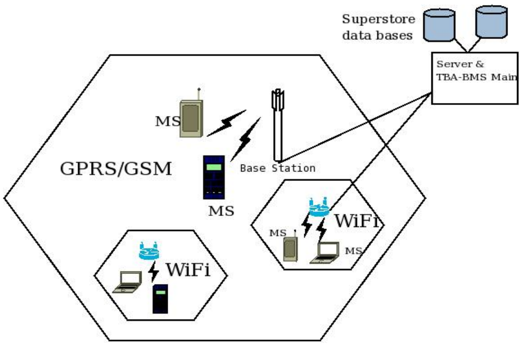

Fig 5: Simulation Environment

We consider different transactions, each of which have a different QoS requirements. In order to estimate the available QoS parameter, we consider various transactions, few transactions going from GSM to WiFi. Some transactions are initiated in GSM or WiFi networks themselves.

The superstore wireless network simulation topology is shown in Fig.5. We assume that the handover probability from WiFi to GSM and GSM to WiFi is 95%. We consider 30 different transactions, each of which have a different bandwidth requirements.

To illustrate, we consider the WiFi and GSM capacities are 11Mbps and 12Kbps respectively. The reservation signaling cost before the establishment of new/handoff connection is set to 0.01%of WiFi and session time is at 148.3s. In order to estimate the available bandwidth, we consider 400 transactions, few transactions going from GSM to WiFi. Some transactions are initiated in GSM or WiFi networks themselves.

To estimate the resource under superstore wireless networks, we consider that the TCP sinks are connected to the bottleneck router through wireless link of 11Mbps. Packet loss is assumed to be only due to errors in wireless links and packet error rate in the wireless link is assumed to be 10%. Each simulation experiment was run until its stable state.

To measure the system bandwidth, we first examine the normalized system utilization defined as the amount of data transmitted in unit time normalized with the system capacity. The simulation was carried out several times with a set of transactions.

A. *Performance evaluation*

Fig.6 shows results measured for transaction delay in seconds with policy-based QoS management and without policy-based QoS management incorporated.

To evaluate the scalability of the system and its ability to cope under different loads, the transaction delay tests were repeated over the WiFi and GSM access technology. We consider framework at a rate of 20, 40, 60, and 80 transactions request per minutes. From figure, we notice that delay produced by policy-based QoS management is less than without policy QoS management scheme

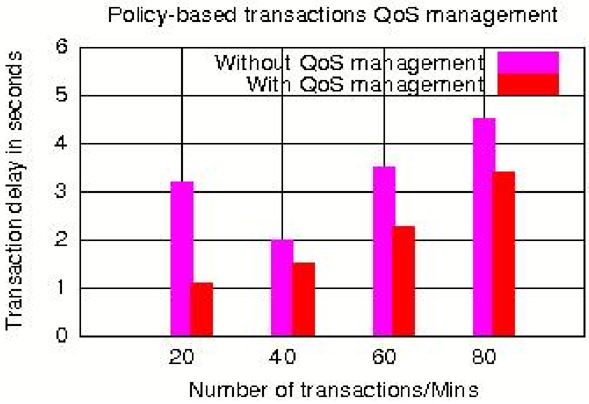

Figure 6: Transactions/minute VS Transaction delay

The explanation of the number of transaction versus resource utilization is shown in Fig.7. From the figure we notice that the policy-based QoS management scheme utilizes the available resources efficiently in the superstore at peak time of the business.

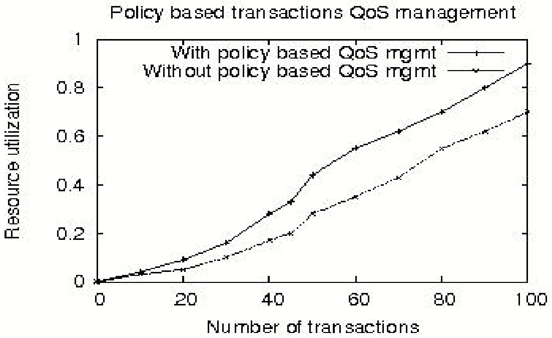

Figure 7: Number of transactions VS resource utilization

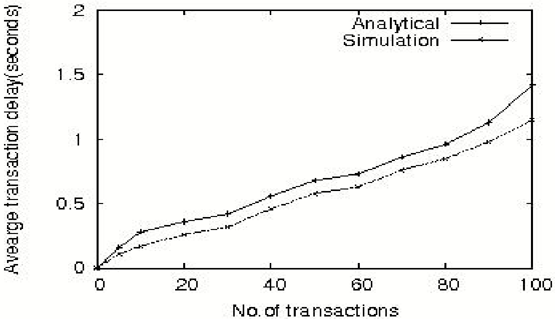

Figure 8: Number of transactions VS Transaction delay.

The average transaction delays for the analytical and simulation models are shown in Fig.8. The results for the analytical model described in this paper are very close to the simulation model for all transaction loads. In the simulation model, if a new transaction arrives within the update interval, the delay will be zero because there is at least one transaction which is terminated in the interval, making bandwidth available for the new arrived transaction.

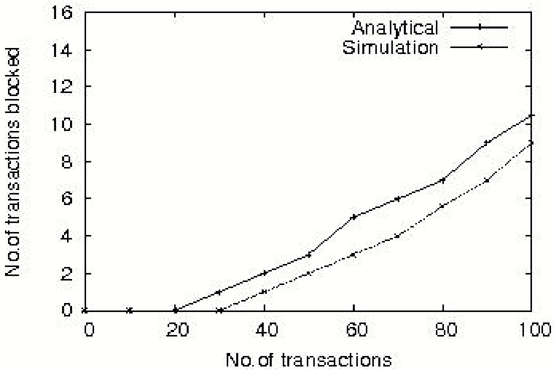

Figure 9: Number of transactions VS Transaction blocked without resource availability.

The number of transaction versus number of transaction blocking is shown in figure 9. We observe that there is no transaction blocking probability for low arrival rate. The transaction blocking probability begins to take non-zero values when the number of transactions reaches 40 under a analytical modeling and for number of transactions 50 for the the simulation. The figure also reveals that the analytical results are close to the simulation results.

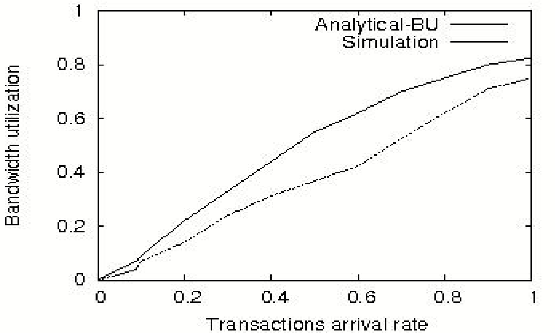

Figure 10: Transactions arrival rate VS bandwidth utilization

The number of transactions versus bandwidth utilization is shown in Fig. 10, we see that bandwidth utilization increases as transaction rate increases. The proposed scheme allows the network to intelligently allocate bandwidth of each admitted transaction by scheduling the transaction according to transaction priority and gives maximum bandwidth for high priority transaction. We also observe that at the low and moderate number of transactions per transaction sensitivity levels the bandwidth is used more efficiently.

Fig. 11 describes the response time of the transaction requests as a function of the total transaction arrival rate. From the Fig. 11, the response time of the high-priority requests with the QoS-aware load balancing policy is always less than that with the QoS-unaware load balancing policy.

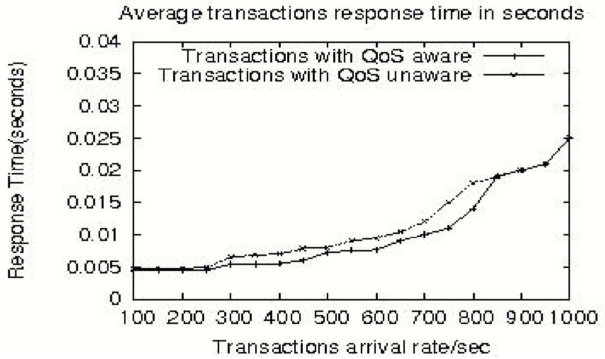

Figure 11: Response time in seconds VS Transaction arrival rate/seconds

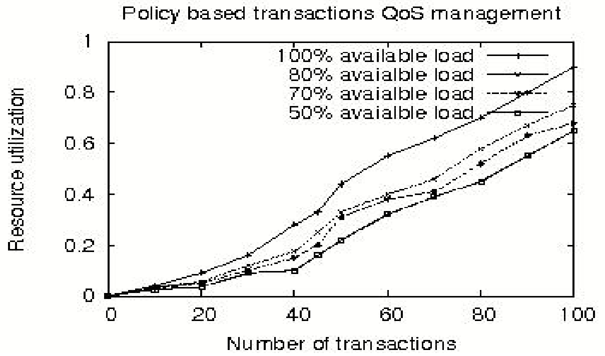

Figure 12: Number of transactions VS resource utilization for different background loads.

The number of transactions versus resource utilization for different available loads is shown in Fig. 12, we see that resource utilization increases as transaction rate increases. The proposed scheme allows the network to intelligently allocate resources for each admitted transaction by scheduling the transaction according to transaction priority and gives maximum bandwidth for high priority transaction.

VI. CONCLUSION

A policy-based QoS architecture in the hybrid wireless domain has been described. A simple policy-based architecture to control the transaction QoS mechanism in a hybrid wireless network has been proposed. These proposed QoS policy architectures can minimize setup delay and policy exchange load while maximizing network capacity for the hybrid wireless superstore networking scenario. Several issues and challenges, which are the foci of our current and future work, are foreseen in the proposed multidomain transaction QoS policy architecture. They are

- Policy negotiation in a hybrid wireless network architecture is a slow process. Proper network design must be done to reduce the delay.
- The security of the communications channel between the interconnected policy entities should be addressed in hybrid wireless superstore network .

REFERENCES

[1] J. Blau, Wi-Fi Hotspot Networks Sprout Like Mushrooms, IEEE Spectrum, pp. 1820, Sept. 2002.

[2] T. Bostrom, T. Goldbeck-Lowe, and R. Keller, Ericsson Mobile Operator WLAN Solution, Ericsson Rev., pp. 3643 , no. 1, 2002.

[3] T. Yamazaki: Adaptive QoS Management for Multimedia Applications in Heterogeneous Environments: A Case Study with Video QoS Mediation, IEICE Trans. Commun., November 1999.

[4] W. Zhuang et al., Policy based QoS Architecture in IP Multimedia Subsystem of UMTS, IEEE Network, vol. 17, pp. 5157, 2003.

[5] 3GPP TS 23.228 (v. 5.2.0), IP Multimedia Subsystem-Stage 2 (Rel. 5), Oct. 2001.

[6] 3GPP TR 23.917 (v. 0.4.0), Dynamic Policy Control Enhancements for End-to-end QoS (Rel. 6), Dec. 2002.

[7] R. Yavatkar, D. Pendarakis, and R. Guerin, A Framework for Policy- Based Admission Control, RFC 2753, Jan. 2000.

[8] B. Moore et al., Policy Core Information Model Version 1 Specifications, RFC 3060, Feb. 2001.

[9] W Zhuang et al., Multi-Domain Policy Architecture for IP Multimedia Subsystem in UMTS,Proc. IFIP/IEEE Net-Con 2002 Network Control and Eng. for QoS, Security and Mobility with Focus on Policy-Based Net., pp. 2738, Oct. 2002.

[10] Shankaraiah and P.Venkataram,"Policy-based QoS management in a hybrid wireless superstore environment,"The 3rd international Conference on Network security, Wireless Communication and Trusted computing(NSWCTC)-2011 Wuhan, China, April 23-24, 2011

[11] A. Westerinen, J. Schnizlein, Terminology for Policy-Based Management, IETF rfc 3198, Nov 2001.

[12] D. Durham et al., The COPS (Common Open Policy Service) Protocol, RFC 2748, Jan. 2000.

[13] 3GPP TR 22.934 (version 6.0.0), Feasibility Study on 3GPP System to Wireless Local Area Network (WLAN) Networking (Rel. 6), May 2002.

[14] 3GPP TS 23.234 (v. 1.0.0), WLAN Subsystem System Description (Rel. 6),Sept. 2002.

[15] T. Abdelzaher, N. Bhatti, Web server QoS management by adaptive content delivery, in: Proc. 7th Int. Workshop on QoS, May, 1999.

[16] Braden, R., Clark, D. & Shenker, D., Integrated Services in the Internet Architecture: an Overview, RFC 1633, June 1994.

[17] Internet of Things in 2020 a road map for the future, http://www.smartsystems-integration.org/public/internet-of-things/. [Accessed: July 12, 2009].

[18] Ee Lim Tan, Wen Ni Ng, Ranyuan Shao, Brandon D. Pereles, and Keat Ghee Ong, A wireless, passive sensor for quantifying packaged food quality, Sensors,Volume 7, Aug. 2007.

[19] Asthana, A., Cravatts, M., Krzyzanowski, P.: An indoor wireless system for personalized shopping assistance. In Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, California, IEEE Computer Society Press 69-74 (December 1994).

[20] Brewer, A., Sloan, N., Landers, T.: Intelligent Tracking in Manufacturing, Journal of Intelligent Manufacturing, 245-250,October 1999.

[21] Dinesh C. Verma.Simplifying Network Administration Using Policy- Based Management, IEEE Network Magazine, March 2002.

[22] T. M. T. Nguyen et al., COPS Usage for SLS Negotiation (COPS-SLS), work in progress, IETF, Feb. 2002.

Pallapa Venkataram received his Ph.D degree in Information Sciences from the University of Sheffield, U.K. in 1986. He is currently a Professor of Electrical Communication Engineering with Indian Institute of Science, Bangalore, India. Prof. Pallapa's research interest includes protocol engineering, wireless networks, network management, computational intelligence applications in communication, mobile computing security, and multimedia systems. He is a Fellow of IEE (England), Fellow of IETE (India), and a Senior member of IEEE Computer Society. Dr. Pallapa is the holder of a distinguished visitor diploma from the Orrego University, Trujillo, Peru. He has authored more than three books, and published over 250 papers in International/national Journals/conferences.

Shankaraiah received his B.E. degree in Electronics and Communication Engineering from Mysore University, Mysore, India, in 1994, M.Tech. Degree in Digital Electronics and Communication Systems from Mysore University in 1997. He has more than 10 years of teaching experience in engineering. From 2007 he is a research scholar in PET unit, department of Electrical Communication Engineering in Indian Institute of Science, Bangalore, India. His research interest includes bandwidth management, Quality of Service (QoS) management, topology management, and Energy management for hybrid wireless superstore environments. He is a student member of IEEE and life member of India Society for Technical Education (LMISTE).