Research Article

Designing Efficient Mobile Gateway for Impoverished Regions Based on Self-Learning Strategy

Yutong Zhou,¹ Wei Shi,¹ Fei Song,² and Ilsun You³

¹*Institute of Education and Economy Research, University of International Business and Economics, Beijing, China*

²*School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China*

³*Department of Information Security Engineering, Soonchunhyang University, Asan, Republic of Korea*

Correspondence should be addressed to Ilsun You; ilsunu@gmail.com

Received 5 August 2016; Revised 27 October 2016; Accepted 13 November 2016

Academic Editor: Dusit Niyato

Copyright © 2016 Yutong Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multiple wireless access technology had tremendously extended the coverage of Internet from developed cities to impoverished regions. As the significant network equipment, mobile gateway had been assigned more and more tasks. However, the efficiency issues are still challenging. Motivated by the latest progress in self-learning, we proposed a new scheme for mobile gateway design. Firstly, the network architecture and supported new features of mobile gateway are discussed to provide a macroscopic impression. Secondly, the application scenarios and functional modules are explored from network and user oriented plane's perspectives. The implementation of intraplane and interplane interactions is carefully illustrated. Thirdly, the transitions of multiple states are analyzed based on possible network context. Fourthly, three typical self-learning schemes are briefly examined and two representative algorithms are adopted. Necessary procedures for deployment are also demonstrated via pseudocodes. Then the interactive process in both network and user oriented cases is investigated. In order to validate the performance, a comprehensive topology with background traffic is established. Simulation results reveal that our hybrid scheme could overcome single parameter based scheme in most of the scenarios.

1. Introduction

Self-learning is an emerging technology being utilized in various contexts to maximize user experiences when limited facilities and workers are involved [1–3]. It is supposed to be a mechanism which could optimize available configurations and resources to achieve the best possible result customized to customers' need. With the latest progress in wireless technology [4–6], more choices and conveniences are brought to people, together with changes to economic development patterns. Based on the Solow Growth Model [7], in the long run, the growth of economy can be attributed to technological progress. However, even though the changes technology brings are tremendous, today there are still numbers of people who cannot enjoy its benefits equally. According to United Nations (UN) report, there are 836 million people living in extreme poverty [8]. In such circumstances, how to utilize technology to achieve a faster and better result for

impoverished people to catch up and progress together with developed regions is crucial in poverty alleviation.

Due to original habits and heritage, a large number of impoverished people live in relatively isolated rocky mountain areas, snow-covered areas, and minority groups gathered country borders with extreme geographic and natural conditions. Without sufficient connections with the outside world, these impoverished regions would experience little to none improvements. In order to change the contiguous poor regions fundamentally, it is essential to provide sufficient education and trainings for local people, along with opening a window for them. Nevertheless, as the living and traffic conditions in these places are difficult, sending poverty alleviation professionals, government, and teaching staffs to those places can be challenging. Staying in these areas to provide hands-on help and to visit again for follow-up guidance would also be hard. From this aspect, a system which can provide long-distance assistance, easy to learn

and to access, can provide smooth connections for real-time guidance which is in demand [9, 10].

The poverty regions in China are usually spread out especially in these aforementioned areas, where the mobile Internet infrastructures are very basic, with poor signal coverage and unstable connections. Local peoples' educational levels and Internet usage are also limited. Making them "online," getting to know the latest world, and being able to acquire the sufficient information via current reliable Internet facilities are important for their life quality improvement [11, 12].

Motivated by previous facts, we believe that a self-learning mechanism applied in wireless Internet would greatly benefit habitants living in contiguous impoverished regions. A mobile gateway with self-learning feature would tailor to users' needs accordingly. It would adapt to people's preferences (such as time, contents, and locations) to provide more stable connections, faster real-time videos [13], travel planning [14], energy management [15], social activities [16], and so forth.

The main contributions of this paper could be summarized in two aspects:

(1) A novel design approach of mobile gateway is proposed for impoverished regions based on self-learning strategy. The specific steps are explored and detailed.

(2) A simulation platform is carefully established to validate the performance of different self-learning schemes. Several advices for scheme selection are also provided.

The basic structure of this paper is as follows: In Section 2, the design details of mobile gateway are presented. The network model and considerations are given at the beginning. Both the application scenarios and the functional modules are demonstrated with internal interactions. The necessary states transitions during the self-learning process are illustrated under different network context. Then three typical self-learning categories are examined and two algorithms are selected. The interactive process from both network oriented and user oriented perspectives is provided. In Section 3, the performance of our new scheme is validated based on several comprehensive simulation scenarios. Section 5 concludes the entire paper and points out future work.

2. The Design Details of Mobile Gateway

In order to present design principles and specific processes comprehensively, five important parts are introduced. At the beginning, the network model and considerations are introduced to provide a macro impression of upgrade place in hierarchical layers. Then it is necessary to investigate application scenarios and functional modules inside mobile gateway. After that, the self-learning engine together with state transition procedures are proposed and analyzed. As the key part, the self-learning algorithm selection approaches are discussed in depth. Finally, implementation notices are reported based on the results of the self-learning strategy.

2.1. Network Model and Considerations. It is important and necessary to present the basic network model we used

FIGURE 1: Network model of mobile gateway.

during the mobile gateway designing process. The traditional network architecture with Transmission Control Protocol (TCP) and Internet Protocol (IP) is kept to guarantee the compatibility with other Internet facilities. Moreover, the application layer and data link layer are also untouched to make the best use of existing usages and to provide more spaces for new services and access modes.

Figure 1 illustrates the operation scope and corresponding considerations of the mobile gateway. In transport layer, TCP and User Datagram Protocol (UDP) had been adopted by most traffic and modified to many versions. Reliable transmissions services (such as file exchanging, web browsing) could be achieved by using congestion control, sequence control, retransmission of lost packets and other TCP functionalities. Unreliable transmission applications (such as audio chatting, video sharing) could be enabled if UDP is selected. As a well-known successor of TCP, the Stream Control Transmission Protocol (SCTP) maintains the previous congestion control and proposes new approaches to support multihoming, multistream, potential failures, and so forth. Similar to SCTP, the Datagram Congestion Control Protocol (DCCP) is designed to remedy the shortcomings of UDP. Multihoming is also provided and multiple congestion control schemes could be freely selected based on the negotiation between the sender and the receiver. In network layer, the mobility issues severely restrict the evolution of network. Due to the ambiguity of IP address in identity and location, the Quality of Service (QoS) of users is hard to guarantee during the mobile scenarios. To solve this problem, Mobile IPv4 (MIPv4), Mobile IPv6 (MIPv6), Proxy Mobile IPv6, and other related schemes are proposed.

Our target is to enhance the previous features inside mobile gateway by enabling multipath and supporting mobility. We emphasize that only the transport layer and the network layer are the main venues for modification. Although multiple SIM cards are utilized, they only provide underlying connections for upper layers.

2.2. Application Scenarios and Functional Modules. The internal structure of mobile gateway can be viewed as a combination of application scenarios and functional modules (shown in Figure 2). The two perspectives, that is, the network oriented plane and the user oriented plane, are included to illustrate the differences.

The network oriented plane in mobile gateway aims to build steady connections to current Internet and remote cloud platforms. The possible application scenarios can be

FIGURE 2: The applications and functions of mobile gateway.

divided into “multiple paths” (all the available paths will be utilized concurrently), “triple paths,” “dual path” (the fractional utilizations of multiple paths are further optimized based on the network states, service types, etc.), “primary + standby” (only one path is marked as primary on purpose for the transmission and other paths will be marked as standbys), and “single path” (only one path is utilized with no backups). To meet the above requirements, the necessary functional modules should include the following: (1) Path establishment: according to the number of SIM cards and Internet access interfaces, all possible paths will be established during the initiation phase. Such procedures are foundations and preconditions to support all kinds of application scenarios. In a concurrent utilization case, since the default setting is to set up only one path, the invitation of multiple paths might be ignored or refused by the corresponding end host. The proper solution should be made to avoid unexpected interruptions. (2) Packet scheduling: there are two types of packets inside mobile gateway, that is, normal packets and retransmission packets. Both of them have to be assigned to specific paths. Therefore, the path selection scheme is crucial for the transmission throughput. By taking advantage of the self-learning results, a suitable path for the next packet could be selected. Besides, the flow control, congestion control, sequence control, and so forth should be considered during the packet sending process as well. In an optimized transfer case, flexible and reliable distribution schemes could be applied, for instance, sending data packets through one path and receiving ACK from another path. (3) ACK processing: comparing with the data packet, the size of ACK is quite small. The number of ACKs could be further reduced when using one to confirm more data packets. By learning, for example, the content, arriving frequency, received interface of ACKs, the mobile gateway should be able to determine the current situations of different paths and corresponding ends. In the primary and standby case, the core elements of self-learning engine could help the sender to timely replace the unsuitable primary path from standby group. (4) States maintenance: the states here contain massive information (one path is usable or not, the value of available

bandwidth, the value of drop rate, current burst situation, etc.). Adding (or removing) the new paths (or the existing paths) also belongs to the scope of state maintenance. These state resources provide sufficient inputs for self-learning engine.

The user oriented plane in mobile gateway focuses on observing, analyzing, and responding massive local access. The potential application scenarios can be classified as “push-based” (broadcast via video and audio), “pull-based” (monitoring or supervision), and “interaction-based” (sensor network and personal usage). To achieve these effects, the necessary functional modules should include the following: (1) Data collection: all the users’ actions could be converted into various data. Both local and remote storage should be enabled to expand the capacity. The superior sample mechanism is required to reduce the volume. The task-driven gathering will be more efficient if the target could be obtained in advance. In sensor network and personal usage cases, data collection could be assigned to lower level gateways. (2) Task management: one huge task must be split into multiple subtasks to increase efficiency. Several key factors, such as task complexity, local resource capacity, and time consumption, should be fully considered. The aggregation of subtask will be executed as soon as all the results are available. In video and audio broadcast cases, the priority of these streams could be generated based on self-learning algorithms, and transmission quality will be ensured using such methods. (3) Algorithm selection: it is clear that the efficiency of the entire mobile gateway is based on performance of algorithms. Based on requirements of tasks, a set composed by different property algorithms should be built and updated continuously. In certain extreme situations, manual selection is also acceptable. In monitoring cases, both the tasks and the targets of supervision are changing at all time. Inflexible and unreliable elements in this set must be removed as soon as possible. (4) Result validation: when the results of self-learning are obtained, the validation must be carried out at once. If a “yes” is returned, the subsequent actions could be proceeded as usual. If a “no” is returned, the causes will be analyzed and reported to other functional modules. Such validation might be repeated several times to guarantee the correctness.

The interactions among different functional modules could be classified into the “intraplane” and “interplane” categories. All the necessary actions could be achieved on top of the specific bus of mobile gateway (a combined block with eight arrowheads is shown in the bottom part of Figure 2). Yellow and red lines are selected to indicate the interactions inside the plane and outside of the plane, respectively. We emphasize that all the modules in both planes could be flexibly connected with each other and complete cooperative work. The details about the process are as below.

Intraplane Interactions. For the network oriented part, after the initiation of relevant registers and state machines, the path should be established at the beginning. The number of paths being utilized during the transmission must be determined based on the capacity of end hosts. Then the packet forwarding and ACK processing should be started

immediately when the paths are ready (marked by yellow line in Figure 2). The retransmission path selection should be done by “packet scheduling” module. “ACK processing” will pay more attention to delayed acknowledgments, duplicated ACKs, missing ACKs, and so forth. Information exchanging between two modules must be frequent to ensure a maximized utilization of available bandwidth. All these previous information must be reported to the “states maintenance” module. The responses will be generated and sent back to the source accordingly. For the user oriented part, “data collection” module is not only the basis for other modules, but also the controllee. Before the collection is initiated, multiple parameters (such as data type, sample rate, labeling schemes, and storing space) must be determined. After each period, a digest should be sent to other modules to update the original information and trigger necessary adjustments of collection patterns. The “task management” and “algorithm selection” modules have to be operating together (marked by yellow line in Figure 2). When a specific task arrives, the parser of “task management” module will abstract the requirements and analyze what kind of algorithm is needed. Then the “algorithm selection” modules will evaluate the available algorithms. When a good match is found, the task will be distributed and executed. Otherwise, the compromised match will be used. The “result validation” will be triggered as soon as the tasks are completed. Both the function and efficiency could be examined. The results will be reported back to the relevant modules for further processes.

Interplane Interactions. In the above description, we ignored several information exchanged between network oriented and user oriented plane intentionally. However, these actions are quite essential for better supporting application scenarios. We use two cases to demonstrate it. The first one is a data-driven establishment for multiple paths. Instead of manual setting, the demand of collected data could be learned to decide the path usage patterns. It could be sent to the “states maintenance” module and then waiting for the unified scheduling (marked by red line in Figure 2). When the demand of collected data had been changed, the corresponding modification of path usage should be adjusted as well. It will definitely affect the process actions of “packet scheduling” and “ACK processing” modules. The second one is path-driven application supporting (not marked in Figure 2). Comparing with simplex network capacity, the user expectation varies and is constantly changing. Due to the priority of different traffic and limited bandwidth, it is common to see some services have to be denied. However, some light weight applications could be enabled to leverage the insufficient resource of paths. For instance, a video broadcast had been denied. If there are still tiny spaces in multiple paths, the “packet scheduling” module could directly invoke the “task management” module and enable a sensor network application. The maximum utilization might be achieved by using such a scheme.

2.3. State Machine Transition. As the significant elements of self-learning engine, the transition process of state machine should be highlighted. In order to better support the above

FIGURE 3: State machine transition inside mobile gateway.

functional modules, five states are included inside the mobile gateway.

The initiative state should be “State 0 (S0)” or “idle state” (illustrated in Figure 3). It will be automatically entered as soon as the system is switched on. An endless loop will be triggered simultaneously. The adopted internal and external interruptions must be monitored to handle the changes in different circumstances. The state transition might happen when an emergency task appears. “State 1 (S1)” is the basic level learning process based on local resources. Only the simple function could be achieved in this state, which is suitable for light weight applications. Part of the resource on board will be enabled. The state will be changed back to S0 if all operations have been finished. To avoid redundancy, we will not repeat similar return processes in the following. “State 2 (S2)” is the deep level local learning process. It may be triggered from S0 directly or via S1. The unoccupied capacity of hardware and software will be invoked to execute the task. It should be assigned the highest priority. If more than one task had been assigned, a specific queue should be established to schedule the sequence. The multithreading can be utilized if subtasks exist. “State 3 (S3)” is the synergetic level learning process, which can be achieved on top of multiple mobile gateways. If the task is too huge to be handled inside one device, the core elements of self-learning engine could seek help from adjacent neighbors. Several exact communication and negotiation steps are required. Some workloads still need to be done locally. Splitting the task reasonably is a crucial mission. Since the interactions with other mobile gateways must be finished in time, some nonurgent tasks could be suspended to save more resources to increase the speed of processing. “State 4 (S4)” is the advanced level learning process at the remote side. If all previous cases could not solve the problem, the task will be reported to the cloud platform. Similar to the previous case, these subtasks need to be locally achieved. Due to the latency brought by distance, the timeliness might be reduced.

A complicated series which had gone through all states is also provided in Figure 3. Such a process, which is marked

with red arrows and number (1) to (5), may bring longer latency. In some particular cases, the state crossing (transmission between S1 and S3, S2 and S4) is also acceptable, which is not demonstrated in Figure 3.

It is significant to consider the network context during the state machine transition, especially if S3 and S4 had been deeply involved. In S3's case, it mainly depends on local mobile gateways. Although the network resources are relatively sufficient, massive and useless data packet may also affect the task distribution and convergence. The advantages of short Round Trip Time (RTT) should be fully utilized to achieve the results effectively in S3. In S4's case, the heavy congestion is a challenge during the remote execution. Sometimes, the strong capacity of cloud platform could not be employed due to the bottleneck of network. The negative influence brought by large RTT must be avoided.

In fluctuated network contexts, several approaches could be adopted to guarantee the reliable operation of the state machine. For instance, by assigning part of available bandwidth for state signaling transmission, the performance of self-learning engine could be ensured. When there is no state which needs to be maintained or switched, the volume of reserved bandwidth should be released to other applications. If the practical situation of other traffic shrinks, the volume of reserved bandwidth could be increased to further enhance information exchange.

2.4. Self-Learning Algorithm Selections. After establishing the policy of state machine transition, the algorithm selection of self-learning should be investigated. The eligible candidates could be roughly separated into three categories. Supervised learning: there should be one or more specific models. The machine will attempt to learn based on the guidance. Labeled or regular data is highly recommended. Unsupervised learning: in some situations, it is hard to find a suitable model to provide pertinent advices for the machine. The patterns and rules can only be discovered based on classification, regression, and other actions without supervision. Reinforcement learning: to better handle new issues during learning process, the modification of basic model is very useful. The continuous feedback procedures are compulsory. In the following, we will present the way to apply these learning schemes into mobile gateway design.

For the network oriented plane, we select Perceptron as a representative scheme in supervised learning. Other unsupervised-based and reinforcement-based schemes are also deployable. In order to adapt to multipath scenarios, the original function should be modified to extend binary results. The new expression is shown in

$$y = f\left(\sum_{i=1}^d w_i x_i - b\right), \quad (1)$$

where d is the dimensionality, x is the d dimensional input vector, and w is the weight vector. Therefore,

$$\begin{aligned} w &= (w_1, w_2, \dots, w_d), \\ x &= (x_1, x_2, \dots, x_d). \end{aligned} \quad (2)$$

```

(1) determine the relevant parameters and dimension of
    vector  $x$ ;
(2) set the initial values for vector  $w$ ;
(3) set the principle of activation function;
(4) while the Perceptron is enabled
(5)   for path number  $r$  from 1 to  $m$  step 1
(6)     calculate the value of  $y_r$  based on formula (3);
(7)     if  $y_{\max} < y_r$  then
(8)       set  $y_{\max} = y_r$ , set  $p_{\text{num}} = r$ ;
(9)     end if
(10)  end for
(11)  select path  $p_{\text{num}}$  for data transmission;
(12)  for path number  $r$  from 1 to  $m$  step 1
(13)    if  $\Delta w$  have to be considered then
(14)      update  $w$  of path  $r$  based on formula (4);
(15)    end if
(16)  end for
(17) end while

```

ALGORITHM 1: The pseudocode of Perceptron utilization.

$f(\cdot)$ is the activation function. If we treat $-b$ as the weight value when i is equal to 0, formula (1) can be rewritten with matrix form

$$\begin{aligned} y &= f\left(\sum_{i=1}^d w_i x_i - b\right) = f\left(\sum_{i=1}^d w_i x_i + (-b) \cdot 1\right) \\ &= f\left(\sum_{i=1}^d w_i x_i + w_0 x_0\right) = f(w^T x). \end{aligned} \quad (3)$$

In order to follow the latest fluctuation of network parameters, the weight value should be adjusted based on the errors; that is,

$$w_{t+1} = w_t + \Delta w, \quad (4)$$

where w_t is the result after t times adjustments. It is also a vector. Δw can be further represented by the learning rate, expected output, and calculated output. The threshold value should be carefully set up to correctly select the best path during the transmission based on current network situation. The execution procedures of algorithm are given in Algorithm 1.

For the user oriented plane, we present K -means as a classical scheme in unsupervised learning. Again, it is also practicable to choose schemes from the other two categories. Since the sorted data can be matched with suitable multipath strategy directly, we first calculate the path selection possibility k . Assuming m is the available path number, then k should be the combination of application scenarios in Figure 2. Therefore,

$$k = \sum_{r=1}^m C_m^r = \sum_{r=1}^m \frac{m!}{r! (m-r)!}, \quad (5)$$

where r is the variable to represent the possible path number utilized by the users. According to basic definitions, assume

```

(1)  determine the relevant parameters and dimension
      of vector  $x$ ;
(2)  initiating the collection for data streams;
(3)  if all available paths could be used then
(4)    set  $k$  based on formula (5);
(5)  else set  $k$  = limited path selections possibility;
(6)  end if
(7)  set  $\mu_i$  by random vectors of group  $i$ ;
(8)  while the formula (7) is not found
(9)    for data stream number  $n \in c$ 
(10)   for group number  $j$  from 1 to  $k$  step 1
(11)    calculate the value of  $e(x, \mu_i)$ ;
(12)    if  $e_{\min} > e(x, \mu_i)$  then
(13)      set  $e_{\min} = e(x, \mu_i)$ , set  $g_{\text{temp}} = j$ ;
(14)    end if
(15)  end for
(16)  include this data stream into group  $g_{\text{temp}}$ ;
(17) end for
(18) calculate the  $\mu_i$  based on the latest values;
(19) end while

```

ALGORITHM 2: The pseudocode of K -means utilization.

we have n data streams and each of them has the same requirement. The property of each data stream is indicated by the real vector x . The dimension of x could be determined by the “data collection” module. The ultimate target is to partition these data streams into k groups ($k \leq n$). Let c_j ($j = 1, 2, \dots, k$) be the sets of data streams that belong to group j . It will be helpful to find the value μ_j ($j = 1, 2, \dots, k$) for each group to minimize the distance from the data streams to the group, which can be expressed by

$$\arg \min_c \sum_{j=1}^k \sum_{x \in c_j} e(x, \mu_j), \quad (6)$$

where c is the set of c_j and $e(x, \mu_j)$ is the square of the Euclidean distance between x and μ_j . Therefore, expression (6) can be rewritten as

$$\arg \min_c \sum_{j=1}^k \sum_{x \in c_j} \|x - \mu_j\|^2. \quad (7)$$

Although the problem seems a bit complicated, we may still use some heuristic approaches during the calculations. The entire procedures are detailed in Algorithm 2.

2.5. Interactive Process. Due to different targets and multiple approaches, we decide to present the interactive process of mobile gateway from two independent perspectives.

For the network oriented case, after the conventional handshakes, the agent in transport layer will establish a connection between the mobile gateway and corresponding ends (such as proxies, base stations, and satellites). When multiple access cards are equipped in a narrow space, the mutual interference should be prevented. Then the network parameter surveillance and estimation should be initiated as

FIGURE 4: The network topology for scheme simulation.

soon as the device is switched on. It is reasonable to include bandwidth, RTT, loss rate, and jitter into the basic set. Both the historical record and future trends could be analyzed to provide fine-grained information for following processes. In most conditions, since the remote ends are traditional equipment, they may not be able to offer new features to support the self-learning abilities. The functional modules in mobile gateways have to complete relevant tasks alone. During the data transmissions, the normal packets could be scheduled to one or more paths and the key packets should be forwarded on the path which is selected based on self-learning results.

For the user oriented case, firstly, multiple types of access need to be included. When the WiFi signal of mobile gateway is available, the users should connect to it via physical layer and Media Access Control (MAC) layer. Other types of wireless access could also be added by increasing extra interfaces. The registration procedures must be executed before the data exchange. Secondly, the self-learning functionality for pattern analysis will be activated as soon as the necessary data are successfully collected. The fluctuations of habits require more attentions. Some of the results should be exposed to the users as the reminder or suggestions. It is suggested that self-learning calculations should not affect normal network services. Thirdly, the priority of different users and multiple services should be provided based on the established rules or self-learning results. Both the end hosts and service providers could send the requests to modify the priority level. Fourthly, the connections among multiple local mobile gateways should be set up. It is beneficial not only for the end host (more options to access the Internet), but also for the mobile gateways (more choices to improve the efficiency).

3. Performance Simulation

The platform is the latest version of Network Simulator 2 [17]. We successfully implement our self-learning scheme into transport layer modules via C++. A more realistic multipath topology (shown in Figure 4) is established based on the traditional dumbbell shape. As the representative protocol, SCTP is adopted and the concurrent multipath transfer feature is enabled. The mobile gateway and the remote server (i.e., the sender and receiver) are located on the left and right

sides. Since the results of reversed directions could give us similar conclusions, we will not illustrate them here. Four intermediate routers compose two available paths. In order to study the influence triggered by hybrid background traffic, multiple generators and terminators are set up. Each node at the top area has three optional applications: FTP (over TCP) and CBR (over UDP). The bandwidths between generators (terminators) and intermediate routers are set to 100 Mb. And the latencies of these links vary from 1 ms to 50 ms. In order to highlight the bottleneck links (shown with lightning symbol in Figure 4), we configure the bandwidth and latency of other links to 500 Mb and 1 ms, respectively. The receive buffer at receiver side is set to infinity to reduce the effect of blocking issues. The drop rate routers are utilized and we release the limitation of queue length. Our target is to test the performance, in terms of throughput, when the bandwidth, RTT, and drop rate of bottleneck links are adjusted.

For measuring the throughput, the simulation time is set to 200 seconds and each case is repeated for 20 times with different seeds. The initial times of all flows are selected between 0 s to 10 s randomly. In all graphs, the average value and 90% confidence intervals (cutting 5% data from both top and bottom) are calculated for illustration. The values of bandwidth, RTT, and loss rate in two bottleneck links are, respectively, adjusted in different scenarios to emulate the network variations.

According to previous principles, we had implemented four schemes: self-learning with bandwidth (SLS-BW), with loss rate (SLS-LR), with RTT (SLS-RTT), and with hybrid considerations of these three parameters (SLS-ALL). Two separated scenarios are set up.

Scenario One: Fix Bandwidth and Change RTT and Loss Rate. Firstly, the value of bandwidth is set to 160 Mb and the value of loss rate is changing from 2% to 6% with a fixed step 1%.

If the value of RTT is set to 20 ms, the four curves are demonstrated in Figure 5(a). With the increasing of loss rate, the total throughput is getting down as well. The average values of four cases (from top to down) are 24136, 23892, 23271, and 22529, respectively. The advantage of SLS-ALL is quite clear in all cases. When the loss rate is small, that is, 2% and 3%, the gaps between SLS-ALL and SLS-RTT are not big (approximately equal to 1%). However, such a gap will be 2.3% when the loss rate is 6%. The curve of SLS-LR is ranking the third place. Since it mainly considers the effects triggered by packet drop, a path with less loss rate will be preferred. The largest gap between the maximum and the minimum is 788 when loss rate is 6%. The performance of SLS-BW is not as good as previous three schemes. The reason might be that the untouched bandwidth could not motivate the capacity of SLS-BW.

If the value of RTT is set to 140 ms, the four curves are illustrated in Figure 5(b). The basic tendency of them is still the same. Due to the longer latency in bottleneck, the overall throughput slightly decreases. We can clearly see that the first two values for SLS-ALL and SLS-RTT are quite close. The confidence intervals of the first two values are also similar. With the increasing of loss rate, the insufficient SLS-RTT is gradually exposed since it only calculates one network

parameter. The largest gap of average value is 1.6% when loss rate is 5%. The decreasing of SLS-LR is almost linear since it is more sensitive to the packet drop. The situation of SLS-BW has not been changed in this case. There is a biggest maximum and minimum gap, when the loss rate is 4%. It indicates that the adaptive ability of this scheme could be improved.

Secondly, the value of bandwidth is set to 160 Mb as well and the value of RTT is changing from 40 ms to 120 ms with step 20.

If the value of loss rate is set to 1%, the four curves are demonstrated in Figure 5(c). It could be seen that the confidence intervals for all cases are quite small, which means the algorithm could provide the relatively stable outputs. Another observation is that the speed of decreasing is gradually reduced with the RTT increasing. The gaps among four curves in different RTTs are also very close to each other, which means the variation of latency is a stable parameter for our learning schemes.

If the value of loss rate is set to 7%, the four curves are illustrated in Figure 5(d). Due to a higher loss rate, more packets will be dropped by the relevant links. Such a situation reduces the throughput for all cases. Comparing to Figure 5(c), the confidence intervals are larger. When more packets are retransmitted by the mobile gateways, it will increase the complexity of learning process and reduce the accuracy of estimation. The performance of SLS-LR is lower than SLS-RTT, which means the latency is still the dominant factor here.

Scenario Two: Fix RTT and Change Bandwidth and Loss Rate. Firstly, the value of RTT is set to 160 ms and the value of loss rate is changing from 2% to 6% with a fixed step 1%.

If the value of the bandwidth is set to 20 Mb, the four curves are demonstrated in Figure 6(a). Since the available capacity of network had been reduced, the throughput will be declined as well. The maximum value 12111 appears in SLS-ALL when the loss rate is 2%. Although the SLS-RTT is close as always (some results are higher than SLS-ALL), the average value is still ranking the second place. With the increasing of loss rate, such a gap is getting bigger, that is, 1.8%, 1.9%, and 2.2% for the last three cases. The performance of SLS-LR is still better than SLS-BW in such a bandwidth-constrained situation. The fluctuation situation, in terms of confidence intervals, is also similar to these two curves. The biggest average value gap 287 appears when the loss rate is 5%.

If the value of the bandwidth is set to 140 Mb, the four curves are illustrated in Figure 6(b). The overall tendency of different cases is similar to Figure 6(a). The maximum value 12208 appears in SLS-ALL when the loss rate is set to 2%. Although the bandwidth had been amplified to 7 times, the throughput has not increased dramatically. The decline of SLS-RTT is almost linear, which accelerates the deviation with SLS-ALL. The largest gap between them is 348 when the loss rate is 6%. Although the trends of SLS-RT and SLS-BW are approaching SLS-RTT, the ranking has still not been changed.

Secondly, the value of RTT is set to 160 ms as well and the value of bandwidth is changing from 40 Mb to 120 Mb with step 20.

FIGURE 5: Scenario one: fix bandwidth and change loss rate and RTT.

If the value of loss rate is set to 1%, the four curves are demonstrated in Figure 6(c). We can find that the throughput of all schemes is maintaining a small fluctuation. Due to the Y axes setting, the confidence intervals are not as big as they look like. With the increasing of bandwidth, some values of SLS-ALL and SLS-RTT are mixed together. The average values of SLS-RTT in different cases are basically unchanged. The curve of SLS-LR is still above that of SLS-BW in most cases.

If the value of loss rate is set to 7%, the four curves are illustrated in Figure 6(d). Larger packet drop rate triggers lower throughput for all cases. The ranking of all curves stay the same as before. The gaps between SLS-ALL and SLS-RTT

are quite close. It should be noticed that all the test results of SLS-RT are beyond that of SLS-BW.

4. Related Work

There are two main research directions which are highly related to our work, that is, self-learning strategy and multipath transmission. After a wide range of reviewing, some selective papers are carefully investigated as below.

4.1. Self-Learning Strategy Related. For the network oriented case, Eom et al. [18] proposed a new scheduler MALMOS

FIGURE 6: Scenario two: fix RTT and change loss rate and bandwidth.

for dynamically mobile offloading. Different from other application-dependent and static schemes, MALMOS contains novel training schemes to support flexible policy and to improve accuracy. The validation results illustrate that MALMOS has 10.9%~40.5% higher accuracy than other two policies. After analyzing the advantages of new technologies and insufficiency of autoscaling policies, Tang et al. [19] introduced SRSA to improve the performance of autoscaling. Two representative scenarios are selected to show the service volatility. The experiments illustrate that this work is better than threshold based policy and voting policy. Using self-learning in Internet traffic analysis is a popular tendency. By pointing out the inconvenience of complex adjustment and

detailed information, Apiletti et al. [20] designed SeLINA to obtain the useful knowledge from network flows with self-tuning property. Both the supervised and unsupervised cases are considered. A real system is established to evaluate the performance by analyzing the traffic from a large ISP. In order to solve the accuracy issue in traffic identification, Grimaudo et al. [21] present a new scheme named SeLeCT to recognize and classify the streams based on unsupervised algorithms and adaptive learning method. Multiple datasets are gathered from real traffic. The paper identifies that the accuracy of SeLeCT could reach 98% in specific conditions. A more comprehensive version of this work can be found in [22]. In order to solve the challenging issues of processing

big data in real time, Di Mauro and Di Sarno [23] illustrated a structure based on Apache Storm and Yahoo SAMOA. The distributed algorithm is applied to achieve stream data. The evaluation results identified the effectiveness of such structures. After introducing four types of machine learning techniques, Jaiswal and Lokhande [24] demonstrated a classification scheme for network traffic. Both the new captured dataset and standard dataset are adopted for investigation purpose. Six algorithms are implemented and compared. The results illustrate that the accuracy could reach 99% when Tree based schemes are utilized.

For the user oriented case, as a common and convenient application layer option, Hyper Text Transfer Protocol (HTTP) had attracted great attention and had been frequently examined in self-learning area. Claeys et al. [25] followed this trend and focused on the HTTP Adaptive Streaming (HAS) which had been widely used in video streaming for many prominent Internet companies. The original schemes are more suitable for the specific network settings and are not able to handle large ranges of dynamic environment. Therefore, the authors present an approach for HAS client design with self-learning ability. Evaluation shows some advantages of new HAS client. According to the weakness of rate adaptation in Dynamic Adaptive Streaming over HTTP (DASH), Chien et al. [26] proposed Machine Learning-Based Adaptive Streaming over HTTP (MLASH), a novel approach to work together with existing adaptation algorithm and enhance the overall accuracy. The simulation validated that the performance of MLASH can outperform the other traditional schemes in video quality. Sensor networks have also provided broad space for self-learning application, which is also very beneficial for impoverished regions. Marinca and Minet [27] investigated the relationship between dynamic actions of sensor nodes and the quality of communication links. Then a new prediction algorithm was proposed to assist the link selection during the data forwarding. The concept of forecaster is also detailed to estimate the link quality based on experts' suggestions. Motivated by the energy saving of sensor network, Chakraborty and Khan [28] built a new statistical model to solve the power efficiency issue in the routing process. Several statistical parameters and threshold values are included to add more features to the protocol. Uncommon patterns will be reported to the base station automatically. The evaluation illustrates the capacity of protocol, in terms of energy efficiency. As we mentioned above, femtocell equipment is crucial for deployment in impoverished regions and self-learning could extend its capability, such as spectrum usage and power management. Alnwaimi et al. [29] focused on disturbance issues and presented scheme for spectrum distribution based on Reinforcement Learning (RL). Two different learning approaches, that is, Bush Mosteller (BM) RL and the Roth-Erev RL, are adopted and modified. Simulation shows that the convergence ability of modified BM is better than the original scheme. The computational burden is also analyzed and discussed. Wang et al. [30] presented a new power allocation scheme for saving the energy of femtocell network. The important actions of main components are appointed and modeled. Self-learning algorithm is utilized in the scheme and two scenarios, that

is, continuous and discrete, are analyzed. The simulation is performed to evaluate the usability and efficiency of proposed scheme.

4.2. Multipath Transmission Related. Using multiple paths for packet transmission still has many issues. Halepoto et al. [31] were concerned about the influence given by shared buffer during the multipath transfer. The authors utilized the buffer separation scheme to guarantee that path with longer (shorter) latency will be assigned less (more) space in the buffer. The evaluation shows that the performance of the new scheme is as good as (better than) the original CMT-SCTP when two paths (four paths) are involved. In order to investigate the effects of using erasure coding in transport layer over multiple paths, Sarwar et al. [32] integrate the erasure code into congestion control mechanisms. A delay-based retransmission parameter is also added at the sender side. Simulation results illustrate that goodput can be enhanced by 10% to 80% in different cases. The performance of video transmission is also provided to validate the advantages. Due to the rapidly increasing signaling traffic in mobile networks, some researchers believe that multipath transfer is a key method to reduce the latency. Grinnemo et al. [33] compare the CMT-SCTP and original SCTP signaling. According to the evaluation results, the answer is "yes" when the bandwidths and RTTs for multiple paths are similar. And larger buffers at the send and receive sides will be helpful. With the question that whether or not multipath transport is suitable for delayed sensitive traffic, Yedugundla et al. [34] validated the transmission performance of video, online gaming, and web browsing services on top of 3G and WLAN coexisting scenarios when CMT-SCTP and MPTCP are utilized. Simulation, emulation, and practical experiment results illustrate that symmetric path is crucial for latency reduction. Since the congestion situation is quite common in current Internet [35], Xu et al. [36] investigated various congestion control schemes in multipath environment and compare them in different scenarios. It is also a good pointer to the latest progress in multipath protocol design. To reduce the negative effects triggered by disordered packets, Li et al. [37] proposed a bandwidth and RTT estimation approach based on extended vector Kalman filter. A path selection policy is also presented based on that. Two scenarios, that is, bandwidth sensitive and latency sensitive, are established in simulation environment. Results illustrate that the disordering is relieved and the performance is enhanced. To extend the usage scope of MPSCTP, Shailendra et al. [38] proposed a new approach for relieving the restriction of "no acknowledgment loss." Compared with congestion control window based scheme, the new approach had shown better performance. An implementation in Linux kernel is also presented. Motivated by Joint Resource Management (JRM), Sumesh et al. [39] presented a modified mobile CMT (m2CMT) scheme to further reduce the influence triggered by receive buffer blocking. Simulation shows that m2CMT could outperform mCMT and original CMT in given scenarios. Also aiming to solve the blocking issues

at the receiver side, Wallace et al. [40] introduce an on-demand scheduler (ODS) for CMT to allow the sender to wait for suitable chances to assign packet on a specific path. A novel congestion window update scheme is also proposed. Comparing with traditional scheduling method, simulation result illustrates that the throughput of ODS can be enhanced.

5. Conclusions

The coverage of mobile Internet in impoverished regions is urgently needed and has significant meanings. As the key element, the impact of mobile gateway had attracted extensive attentions. After analyzing the primary problems and possible solutions from different points of view, we proposed a design approach for mobile gateway based on self-learning strategy. It considered the characteristics of individual paths and overall performance of connections. During the investigation process, the network architecture and target features are firstly presented. Secondly the application scenarios together with functional modules inside mobile gateway are discussed. Thirdly, the core element of self-learning engine is demonstrated based on five different states. The influences of network context are also analyzed. Fourthly, three typical self-learning schemes are examined and two algorithms are introduced. The interactive process is provided from both the network and user oriented perspectives. The advanced characteristics of the new scheme are investigated based on simulation platform. Performance evaluations demonstrate that the SLS-ALL scheme outperforms several original schemes in different scenarios.

For the future work, it would be intriguing to further study the capacity of our design scheme in practical environments. The corresponding adjustments of implementations should be made based on measured results. Potential usages in more impoverished regions can be carefully extended as well.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was supported by the Soonchunhyang University Research Fund and Project of State Grid Corporation of China under Grant no. SGRIXTJSFW [2016] 377.

References

- [1] I. Bilogrevic, K. Huguenin, B. Agir, M. Jadliwala, M. Gazaki, and J.-P. Hubaux, "A machine-learning based approach to privacy-aware information-sharing in mobile social networks," *Pervasive and Mobile Computing*, vol. 25, pp. 125–142, 2016.
- [2] S. Aram, I. Khosa, and E. Pasero, "Conserving energy through neural prediction of sensed data," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 6, no. 1, pp. 74–97, 2015.
- [3] N. Morozs, T. Clarke, and D. Grace, "Heuristically accelerated reinforcement learning for dynamic secondary spectrum sharing," *IEEE Access*, vol. 3, pp. 2771–2783, 2015.
- [4] K. Guan, Z. D. Zhong, J. I. Alonso, and C. Briso-Rodríguez, "Measurement of distributed antenna systems at 2.4 GHz in a realistic subway tunnel environment," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 2, pp. 834–837, 2012.
- [5] C. Xu, S. Jia, L. Zhong, H. Zhang, and G.-M. Muntean, "Ant-inspired mini-community-based solution for video-on-demand services in wireless mobile networks," *IEEE Transactions on Broadcasting*, vol. 60, no. 2, pp. 322–335, 2014.
- [6] K. Guan, G. Li, T. Kuerner et al., "On millimeter wave and THz mobile radio channel for smart rail mobility," *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2016.
- [7] L. Guerrini, "The Solow-Swan model with a bounded population growth rate," *Journal of Mathematical Economics*, vol. 42, no. 1, pp. 14–21, 2006.
- [8] The United Nations, <http://www.un.org/>.
- [9] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, 2016.
- [10] F. Song, D. Huang, H. Zhou, H. Zhang, and I. You, "An optimization-based scheme for efficient virtual machine placement," *International Journal of Parallel Programming*, vol. 42, no. 5, pp. 853–872, 2014.
- [11] R. Soua and P. Minet, "Multichannel assignment protocols in wireless sensor networks: a comprehensive survey," *Pervasive and Mobile Computing*, vol. 16, pp. 2–21, 2015.
- [12] Z. Yi and T. Dohi, "Toward Highly Dependable Power-Aware Mobile Ad Hoc Network–Survivability Evaluation Framework," *IEEE Access*, vol. 3, pp. 2665–2676, 2015.
- [13] C. Xu, S. Jia, M. Wang, L. Zhong, H. Zhang, and G.-M. Muntean, "Performance-aware mobile community-based VoD streaming over vehicular Ad Hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 3, pp. 1201–1217, 2015.
- [14] F. Song, R. Li, and H. Zhou, "Feasibility and issues for establishing network-based carpooling scheme," *Pervasive and Mobile Computing*, vol. 24, pp. 4–15, 2015.
- [15] P. Li, R. Dargaville, F. Liu, J. Xia, and Y.-D. Song, "Data-based statistical property analyzing and storage sizing for hybrid renewable energy systems," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 11, pp. 6996–7008, 2015.
- [16] C. Xu, S. Jia, L. Zhong, and G.-M. Muntean, "Socially aware mobile peer-to-peer communications for community multimedia streaming services," *IEEE Communications Magazine*, vol. 53, no. 10, pp. 150–156, 2015.
- [17] "The Network Simulator NS-2," <http://www.isi.edu/nsnam/ns/>.
- [18] H. Eom, R. Figueiredo, H. Cai, Y. Zhang, and G. Huang, "MAL-MOS: machine learning-based mobile offloading scheduler with online training," in *Proceedings of the 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, (MobileCloud '15)*, pp. 51–60, San Francisco, Calif, USA, April 2015.
- [19] P. Tang, F. Li, W. Zhou, W. Hu, and L. Yang, "Efficient auto-scaling approach in the telco cloud using self-learning algorithm," in *Proceedings of the Global Communications Conference (GLOBECOM), 2015 IEEE*, pp. 1–6, San Diego, CA, USA, December 2015.
- [20] D. Apiletti, E. Baralis, T. Cerquitelli et al., "SeLINA: a self-learning insightful network analyzer," *IEEE Transactions on*

- Network and Service Management*, vol. 13, no. 3, pp. 696–710, 2016.
- [21] L. Grimaudo, M. Mellia, E. Baralis, and R. Keralapura, “Self-learning classifier for Internet traffic,” in *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 3381–3386, Turin, Italy, April 2013.
 - [22] L. Grimaudo, M. Mellia, E. Baralis, and R. Keralapura, “SeLeCT: self-learning classifier for internet traffic,” *IEEE Transactions on Network and Service Management*, vol. 11, no. 2, pp. 144–157, 2014.
 - [23] M. Di Mauro and C. Di Sarno, “A framework for Internet data real-time processing: a machine-learning approach,” in *Proceedings of the 48th Annual IEEE International Carnahan Conference on Security Technology (ICCST '14)*, pp. 1–6, IEEE, Rome, Italy, October 2014.
 - [24] R. C. Jaiswal and S. D. Lokhande, “Machine learning based internet traffic recognition with statistical approach,” in *Proceedings of the 10th Annual Conference of the IEEE India Council (INDICON '13)*, Mumbai, India, December 2013.
 - [25] M. Claeys, S. Latre, J. Famaey, and F. De Turck, “Design and evaluation of a self-learning HTTP adaptive video streaming client,” *IEEE Communications Letters*, vol. 18, no. 4, pp. 716–719, 2014.
 - [26] Y.-L. Chien, K. C.-J. Lin, and M.-S. Chen, “Machine learning based rate adaptation with elastic feature selection for HTTP-based streaming,” in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '15)*, pp. 1–6, IEEE, Turin, Italy, July 2015.
 - [27] D. Marinca and P. Minet, “On-line learning and prediction of link quality in wireless sensor networks,” in *Proceedings of the IEEE Global Communications Conference (GLOBECOM '14)*, pp. 1245–1251, IEEE, Austin, Tex, USA, December 2014.
 - [28] S. Chakraborty and A. K. Khan, “A noble approach for self learning and cluster based routing protocol with power efficiency in WSN,” in *Proceedings of the 3rd International Conference on Communications and Signal Processing (ICCSP '14)*, pp. 773–777, IEEE, Melmaruvathur, India, April 2014.
 - [29] G. Alnwaimi, T. Zahir, S. Vahid, and K. Moessner, “Machine learning based knowledge acquisition on spectrum usage for LTE femtocells,” in *Proceedings of the IEEE 78th Vehicular Technology Conference (VTC '13)*, pp. 1–6, IEEE, Las Vegas, Nev, USA, September 2013.
 - [30] W. Wang, A. Kwasinski, and Z. Han, “Power allocation with stackelberg game in femtocell networks: a self-learning approach,” in *Proceedings of the 11th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON '14)*, pp. 354–362, IEEE, Singapore, July 2014.
 - [31] I. A. Halepoto, F. C. M. Lau, and Z. Niu, “Concurrent multipath transfer under delay-based dissimilarity using SCTP,” in *Proceedings of the 2nd International Conference on Computing Technology and Information Management (ICCTIM '15)*, pp. 180–185, Johor, Malaysia, April 2015.
 - [32] G. Sarwar, P.-U. Tournoux, R. Boreli, and E. Lochin, “ECMT-SCTP: improving performance of multipath SCTP with erasure coding over lossy links,” in *Proceedings of the 38th Annual IEEE Conference on Local Computer Networks (LCN '13)*, pp. 476–483, IEEE, Sydney, Australia, October 2013.
 - [33] K.-J. Grinnemo, A. Brunstrom, and J. Cheng, “Using concurrent multipath transfer to improve the sctp startup behavior for PSTN signaling traffic,” in *Proceedings of the 28th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA '14)*, pp. 772–778, IEEE, Victoria, Canada, May 2014.
 - [34] K. Yedugundla, S. Ferlin, T. Dreibholz et al., “Is multi-path transport suitable for latency sensitive traffic?” *Computer Networks*, vol. 105, pp. 1–21, 2016.
 - [35] F. Song, Y. Zhang, Z. An, H. Zhou, and I. You, “The correlation study for parameters in four tuples,” *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 19, no. 1-2, pp. 38–49, 2015.
 - [36] C. Xu, J. Zhao, and G.-M. Muntean, “Congestion control design for multipath transport protocols: a survey,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2948–2969, 2016.
 - [37] W. Li, W. Wang, X. Jing, and W. Liu, “Optimizing CMT performance by joint predictions of bandwidth and RTT,” *The Journal of China Universities of Posts and Telecommunications*, vol. 22, no. 4, pp. 81–91, 2015.
 - [38] S. Shailendra, R. Bhattacharjee, and S. K. Bose, “A multipath variant of SCTP with optimized flow division extension,” *Computer Communications*, vol. 67, pp. 56–65, 2015.
 - [39] E. P. Sumesh, K. R. Kashwan, and S. Karthik, “International conference on communication technology and system design 2011the modified mobile concurrent multipath transfer for joint resource management,” *Procedia Engineering*, vol. 30, pp. 963–969, 2012.
 - [40] T. D. Wallace, K. A. Meerja, and A. Shami, “On-demand scheduling for concurrent multipath transfer using the stream control transmission protocol,” *Journal of Network and Computer Applications*, vol. 47, pp. 11–22, 2015.