# Adaptive Overlay Routing of long-lived flows in Ambient Networks

Zhaohong Lai, Alex Galis, Chris Todd

University College London, Dept. of Electronic & Electrical Engineering, Torrington Place, London WC1E 7JE

{z.lai, a.galis, c.todd }@ee.ucl.ac.uk

**Abstract:** One key design requirement for Ambient Networks is the provisioning of the service-aware transport overlays. In order to fulfil this requirement, the overlay routing needs to adapt to changes in service specific QoS or specific network conditions and context. In this paper, it discusses a novel solution that the overlay routing's stability and performance can be effectively improved by rerouting long-lived flows in the alternative (or new discovered) path while short-lived flows stay in the old path. In addition, this paper proposes an adaptive sensor design to detect the long-lived flow based on the feedback of the available bandwidth.

## 1 Introduction

The Ambient Network (AN) project is aimed to foster co-operation between the next generation, heterogeneous wireless networks, in order to gather resources within and across ANs to provide new services [3]. One innovative design in the Ambient Network is to provide the service-aware transport functionalities, which is implemented within a set of service-specific overlays, which is called as Service-aware Adaptive Transport Overlays (SATOs) [7]. As to fulfil this aim, the overlay routing needs to be performed as stable as possible to meet the service specific QoS as well as the consideration of adapting the changing overlay network conditions when a new network joins in the AN. To achieve this goal, a novel solution that reroutes long-lived flows in the alternative (or new discovered) paths when the overlay path conditions are degraded is proposed. To perform the overlay routing at the flow-level bring many advantages for the AN. For example, as stated in [16], the packet-level routing like multipath forwarding will cause the end-to-end performance degradation. One typical example is that packet-level forwarding in different paths will introduce the disordering effect at the receiver. This will lead to unnecessary DUP ACK responses that force TCP to retransmit packets. Other benefits for the overlay routing of long-lived flows are detailed in section 2.

This paper is structured as follows. The next section discusses the benefits for the overlay routing of long-lived flows in ANs. In addition, it also examines the stability features in the overlay routing of long-lived flows. In section 3, it analyses the drawbacks of the existing long-lived flow classifier designs. Section 4 includes a proposal for a long-lived flow sensor and the associated design for the overlay routing of long-lived flows. Finally, a short conclusion is presented in section 5.

## 2. Background

### 2.1 The benefits for the overlay routing of long-lived flows in ANs

One key feature defined within the AN project is the (de)composition concept of Ambient Networks [3]. For instance, a small AN such as a Personal AN can compose with the train station AN when arriving in the train station, forming then a larger AN. In case of (de)composition, all the entities involved in the ANs should also compose or interoperate and this composition process should be completely transparent for the services and the end-users. In figure 2.1, four sub ANs join in an AN which includes underlying and overlay networks. Due to the changes of composition, the overlay network conditions are subsequently affected like the overlay path 1 becomes congested. But, there is another path available via Onode2. To perform this adaptive overlay routing, a decision needs to make on what kind of flows should be switched to alternative path. Many studies show that about 10%-20% of the long-lived flow (LLF) flows account for over 80% of the total traffic [1][2]. For example, in [2], it shows that 20% of the flows have more than 10 packets but these flows carry 85% of the total traffic.

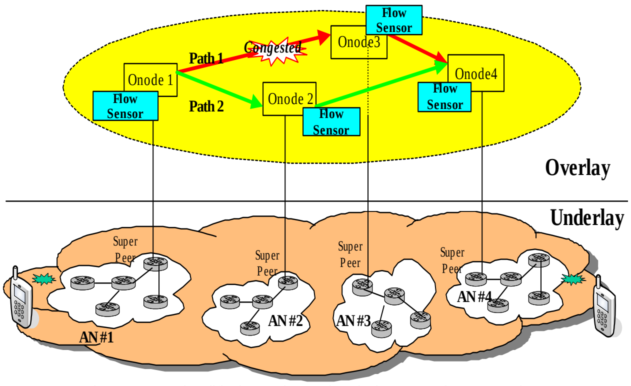

Figure 2.1: a simplified Overlay and underlay network strucutre in ANs

This is depicted below where the solid lines show the proportion of flows that have over 10 packets clearly drop dramatically, but its packet proportion (the dashed lines) account for the major part of the traffic[**ibid**].The overlay routing of long-lived flows at leave have three benefits to suit the AN dynamic feature. First, it improves the routing efficiency because the LLF rerouting cost is much less than short-lived flows (SLF). Second, it highly maintains the routing stability as only 10%- 20% (normally less) long-lived flows needs to be rerouting. This also is detailed in the next section. Third, it provides another way to be aware of service-specific QoS requirements when the network conditions are degraded as most services having specific QoS requirements produces the long-lived flows i.e. a video streaming application.

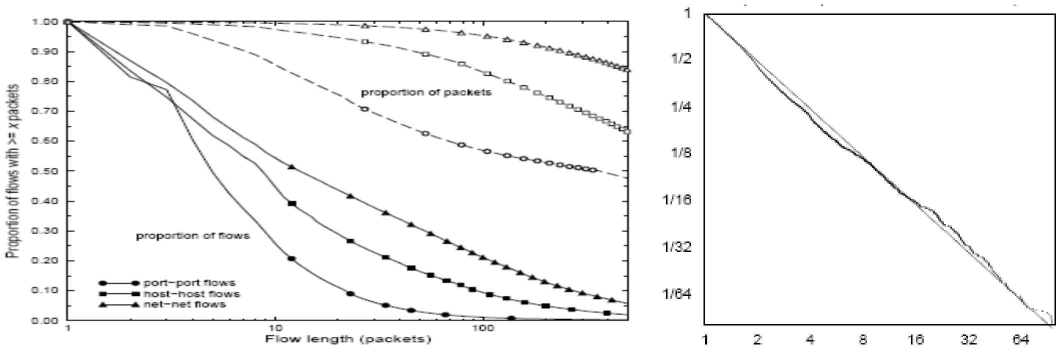

Figure 2.1: (a) the heavy-tailed distribution of IP traffic [2]; (b), the long-live flow distribution (logarithmic).

## 2.2 The manageable and stability characteristics of long-lived flows

The reason of LLFs that is more controllable and manageable than SLFs is because its distribution is heavy-tailed. In [12], it states that the upper 5% tail per FTPDATA burst fits well to a Pareto distribution with parameter $0.9 \le \alpha \le 1.4$. A Pareto distribution has the decreasing failure rate (DFR):

$$failure\_rate(t) = \frac{f(x)}{F(x)} = \frac{\alpha x^{-\alpha-1}}{x^{-\alpha}} = \frac{\alpha}{x} \qquad (2.3)$$

The equation shows that a Pareto alike IP long-lived flows will live longer as *x* increases, which indicates that it will use more network resource as shown in the log-log diagram in figure 2.1 (b). One of reasons causing the heavy-tailed feature of the long-lived flow is due to the effect of the TCP's inherent algorithm control. This is examined as follows. If T is denoted as one RTT length and $\mu$ stands for service rate of a TCP segment (a segment service time=$1/\mu$). The *cwnd* size is exponentially increased at the beginning according to TCP Reno's algorithm. Once it reaches the threshold value, the *cwnd* will be set to 1 and its threshold value will be halved. Let's assume the *cwnd* threshold size $W_t = 2^4$, then TCP will take $4 \times T$ time (from $t_1$ to $t_4$) to reach threshold. If there is no random loss happened during the step 1, TCP enters a periodic evolution with a repeated *cwnd* change comprising step 2 and step3. If TCP takes *n*-RTTs time to reach $W_t$ in the first Slow Start phase, it needs $T_\alpha$ to reach threshold during the second step: $T_\alpha = \log_2 2^{n-1} + 2^{n-1}$ (2.4)

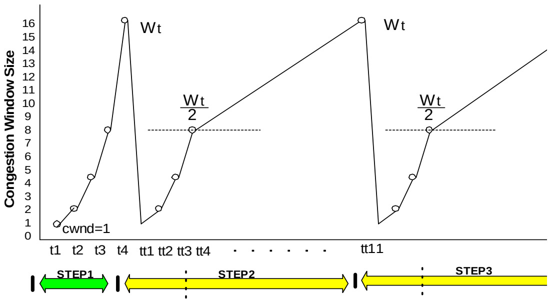

Fig.2.2 - TCP Flow cwnd Evolution (Threshold Size = 16)

Thus, in Fig 2.2, as *n*=4 in step 1, then the next expected time will be: $T_\alpha = 4 - 1 + 2^{4-1} = 3 + 2^3 = 11$.This shows that TCP spends about 73% (4/15) on the additive increase multiplicative decrease (AIMD) phase. As shown in table 1, the bigger *n* TCP spends on the slow start stage, the longer TCP will have to stay in AIMD. In practical, TCP *cwnd* evolution will be more complex than table 1 [10] and the complexity will finally lead to the heavy-tailed distribution of the LLF. The result from table 1 also tells the fact that long-lived flows are less burst and more stable than short-lived flows.

| | | |
|---|---|---|
| n=4 | $T_\alpha$=3+$2^3$=11 | AIMD_ratio = 0.7272 |
| n=5 | $T_\alpha$=4+$2^4$=20 | AIMD_ratio = 0.80 |
| n=6 | $T_\alpha$=5+$2^5$=37 | AIMD_ratio=0.865 |
| n=7 | $T_\alpha$=6+$2^6$=70 | AIMD_ratio=0.9142 |

Table 1 - AIMD Statistical Ratio Value

## 3. The review on the long-lived flow classification

A long-lived flow is defined as a flow has long duration, high-bit rate and large flow size [16]. In [13][5], the application-based flow classification has been proposed. It classifies *host+port* flows into long-lived and short-lived flows according to their applications. But, as a great variation of flow duration exists for an application like *ftp, http*, its results are poor [4]. In [17], the solution of detecting a LLF is developed by evaluating the burstness ($\beta$) of TCP packets. If $\beta$>4 belongs to LLF and otherwise SLF. This mechanism is according to the phenomenon that a LLF's TCP *cwnd* generally is larger than that of the SLF [8][11]. This is because a LLF spends most of time in *Congestion Avoidance* phase but SLF staying in *Slow Start* phase. However, the scalability is a big issue for this mechanism as a *cwnd* size is difficult to be known at the layer 3. In [13] the packet-counter *X/T* based algorithm has been presented. *X* refers to the packet arrival number of a flow while *T* means timeout value. When a flow's *X* exceeds a given threshold within *T*, this flow is categorised into the LLF.

## 4. 4. The overlay routing of long-lived flows

### 4.1 The adaptive long-lived flow sensor design

Since *X* and *T* parameters are assigned statically [13], Hao et al argue that its result will not be accurate as the *X* value should be decreased when a network connection is degraded [4]. Thus, Wang et al state that the difficulty to classify the long-lived flows is how to find an adaptive value for *X* [15]. He also points out that the difficulty is that how to adjust the *X* value according to the network load. One key metric determining the network load condition is the available bandwidth (ABW) value. Thus, if the network available bandwidth is known, *X* can be adaptively adjusted according to the feedback of ABW. According to the BW equation of the packet-pair estimation mechanism, the BW is linear with packet size [14]:

$$bandwidth = \frac{packet\_size}{delay_{bottlenect}} \tag{4.1}$$

Therefore, by contrast, if we know the ABW, we can set the *X* value accordingly. Let *k be* the linear factor and *C* be the *X* threshold value, *k* and its responding *X* value can be computed as:

$$X = C \times k \text{ subject to } k = \frac{ABW}{Physical\_BW} \tag{4.2}$$

where the static physical bandwidth refers to the network path capacity, which can be measured using packet-pair mechanism like ***Nettimer*** [6]. *C* is the threshold of the packet counter value. The threshold value should be measured under the network condition when the network ABW is close to the network capacity. *Pathload* can be used to measured ABW values [9].

To apply this equation, another threshold value needs to be introduced. This threshold value is to determine when *k* should be used. This is because when the ABE value is sufficient, the packet counters *X* will not be affected. For example, a video streaming application will be performed well (normally) when ABW is over 10Mbps. Thus, the *X* value for this video flow will not be affected by the ABW once it is over 10 Mbps. The threshold value can be manually assigned. In principle, this threshold value needs to relate the congestion parameters. For instance, it should be equal to ABW when the network starts to be congested.

```
_long_lived_flow_sensor_algorithm:
{
   /* Initialisation */
    C_threshold = CONSTANT; // manually set
   physical_bw = getNetworkCapacity("pathrate");
   available_bw=testABW("pathload");
   threshold_k=fixed_X_BW_value;
   for (id in flows[id] )
            {
               if( ABW > threshold_k )
                        X[id]=fixed_X;
               else if {
                        k=abw/physical_bw;
                        X[id]=using_adaptive_rate(k);
            /* to call the adaptive rate function*/
                        }
            }
            return X[id];
}
```

Fig.4.1The algorithm of the long-lived flow sensor

### 4.2 The architecture for the overlay routing of long-lived flow

The architecture mainly involves four components as depicted in figure 4.2. Once the traffic coming in, it will be classified into if it is a LLF or not by the LLF sensor. The detected results will be stored in the flow register. The overlay BW sensor provides the available BW information for the calculation of the LLF sensor. It also decides when the long-lived flow information should be sent to the overlay routing agent like the conditions that the ABW is less than threshold value. The flow register stores the detected long-lived flow record. It also manages the information which flows have been rerouted. It will assign each flow a routing flag value ranging from

000~111. A timer for each flow also is set up. The timer has two purposes. One is if a LLF has not received any new coming packets when the timer expires, this LLF flow will be cleaned from the database. Another is that, together with the routing flag, it is to avoid the flapping issue between overlay paths. For example, if the flow's routing flag is just increased one, it is forbidden to be rerouted within the given timer value.

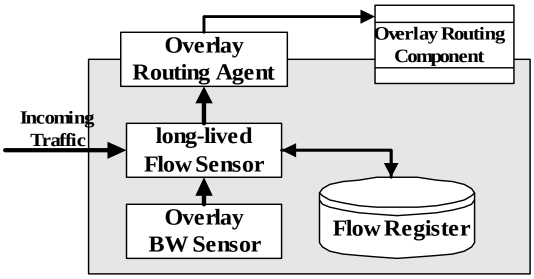

Fig.4.2 The architecture for the long-lived flow routing

## 5. Conclusions.

This paper analyses the efficiency and stability for the overlay routing of long-lived flows in Ambient Networks. The proposed new algorithm of the LLF sensor enables higher accuracy than the existing solutions as it is able to adapt to network load changes. The design of the overlay routing of LLFs also has been described.

## 6. Acknowledgments

This paper describes work partially undertaken in the context of the Ambient Networks (Phase 2) - Information Society Technologies project, which is partially funded by the Commission of the European Union.

## 7. References

[1] A. B. Downey. Evidence for long-tailed distributions in the internet. In Proceedings of ACM SIGCOMM Internet Measurment Workshop, 2001.

[2] A. Shaikh, J. Rexford, and K. G. Shin. Load-sensitive routing of long-lived IP flows. In Proceedings of ACM SIGCOMM'99, Cambridge, Massachusetts, USA, 1999.

[3] EU-IST project 507134, Ambient Networks, http://www.ambient-networks.org

[4] Hao Che, San qi Li, and Arthur Lin. Adaptive resource management for flow-based IP/ ATM hybrid switching systems. IEEE/ACM Transactions on Networking, 6(5):544--557, October 1998.

[5] K. Nagami, Y. Katsube, Y. Shobatake, A. Mogi, S. Matsuzawa, T. Jinmei, and H. Esaki, "Flow attribute notification protocol (FANP) specification," draft-rfced-info-nagami-00.txt, Feb.1997.

[6] K. Lai and M. Baker. "Nettimer: A tool for measuring bottleneck link bandwidth". Proc. of the USENIX Symp. On Internet Technologies and Systems, March 2001

[7] L. Cheng, K. Jean, R. Ocampo, A. Galis, "Service-aware Overlay Adaptation in Ambient Networks", International Multi-Conference on Computing in the Global Information Technology (ICCGI) 2006.

[8] I. Matta and L. Guo, "Differentiated Predictive Fair Service for TCP Flows," In Proceedings of ICNP'2000.

[9] M. Jain, C. Dovrolis, "Pathload: A Measurement Tool for End-to-end Available Bandwidth", In Proceedings of the 3rd Passive and Active Measurements Workshop, March 2002, Fort Collins CO.

[10] Morris, R.; "TCP behavior with many flows", Network Protocols, 1997. Proceedings., 1997 International Conference on 28-31 Oct. 1997 Page(s):205 - 211 Digital Object Identifier 10.1109/ICNP.1997.643715

[11] N. Cardwell, S. Savage, T. Anderson, "Modeling TCP Latency," IEEE/INFOCOM"2000, Tel-Aviv, Israel, March 2000.

[12] V. Paxson, "Empirically-Derived Analytic Models of Wide-Area TCP Connections," IEEE/ACM Transactions on Networking, 2(4), pp. 316-336, August, 1994.

[13] P. Newman, Tom Lyon and G. Minshall, "Flow Labelled IP: Connectionless ATM Under IP", in Proceedings of INFOCOM'96, SanFrancisco, April 1996.

[14] Robert L. Carter and Mark E. Crovella. Dynamic Server Selection using Bandwidth Probing in Wide-Area Networks. Technical Report BU-CS-96-007, Boston University, 1996.

[15] Weihua Wang; Chien-Chung Shen: An adaptive flow classification scheme for data-driven label switching networks, Communications, 2001. ICC 2001. IEEE International Conference on, Volume 8, 11-1, June 2001 Page(s):2613 - 2619, vol.8.

[16] Youngseok Lee and Yanghee Choi, "An Adaptive Flow-Level Load Control Scheme for Multipath Forwarding", Lecture Notes In Computer Science; Vol. 2093, Proceedings of the First International Conference on Networking-Part 1, Pages: 771 - 779, 2001,ISBN:3-540-42302-8.

[17] Yilmaz, S.; Matta, I., On class-based isolation of UDP, short-lived and long-lived TCP flows,Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2001. Proceedings. Ninth International Symposium on 15-18 Aug. 2001 Page(s): 415 – 422.