

©2008 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Dynamic Buffer Tuning: An Ambience-Intelligent Way for Digital Ecosystem Success

Jackei H.K. Wong¹, Allan K.Y. Wong¹, Wilfred W.K. Lin¹, Tharam S. Dillon²

¹Department of Computing, Hong Kong Polytechnic University, Hong Kong SAR

²DEBII, Curtin University of Technology, Perth, Australia

cswklin@comp.polyu.edu.hk, jwong@purapharm.com, csalwong@comp.polyu.edu.hk and Tharam.Dillon@cbs.curtin.edu.au

Abstract

Ambient intelligence is an important element for the success of digital ecosystems which usually are made up of many collaborating distributed nodes. The operations of these nodes affect one another as chain reactions. When one node had failed, it could bring down the whole ecosystem. Dynamic buffer tuning is an ambience-intelligent mechanism because it has the ability to sense the ambient changes and then makes necessary proactive changes on the fly to avoid buffer overflow. As a result the end-to-end communication channel is more dependable, leading to shorter response time and happier clients. Therefore, dynamic buffer tuning should be generally beneficial to digital ecosystem system performance. In this paper we demonstrate this point by using the FLC (Fuzzy Logic Controller) dynamic buffer tuner to quicken the pervasive medical consultation response of the TCM (Traditional Chinese Medicine) Pervasive Digital HealthCare System as an example.

Index Terms: digital ecosystem, ambience intelligence, dynamic buffer tuning, fast system response, mobile Internet, pervasive medical consultation

1. Introduction

In this paper we propose to use the FLC (Fuzzy Logic Controller) dynamic buffer tuner to shorten the response or service roundtrip time (RTT) of the pervasive medical consultation system, TCM-PDHS. The PDHS (Pervasive Digital HealthCare System) in focus caters to Traditional Chinese Medicine (TCM), and it is called T-PDHS hereafter for brevity. T-PDHS is a distributed digital ecosystem on the mobile Internet. It consists of collaborating entities/species, which includes mobile devices (e.g. mobile phone and PDA), remote dispensaries, dedicated surrogates (specialized server entities with intelligence), and smart spaces that support different wireless technologies. The "FLC in T-PDHS" setup will be verified by simulations.

In pervasive consultation a physician can treat patients anywhere, anytime. What the physician needs is a mobile device to interact with the rest of the T-PDHS over a mobile Internet (or web-based) infrastructure. The mobile device hosts an intelligent logical agent, which can carry out different tasks including the following over the web: a) create an electronic medical record for the patient to be remotely stored, b) retrieve a patient's record for the current process of consultation and diagnosis, c) consult other remote expert physicians in order to make a more conclusive collaborated diagnosis, d) get computer-aided diagnostic analysis and supporting information (e.g. laboratory reports), and e) remote dispensing. The usability of a pervasive consultation system relies on various factors that include: a) correct information transfer over the web, and b) fast system response. The focus here is how the FLC ambience intelligence can help T-PDHS gain short pervasive medical consultation response to the benefit of patients. It serves as a demonstration example of how dynamic buffer tuning can enhance the quality of pervasive consultation.

Figure 1 summarizes the web-based pervasive T-PDHS infrastructure as follows:

- a) It is operating over the mobile Internet that supports both wireless and wireline communications in a seamless fashion.
- b) The end-to-end client/server interaction can be wireless and wireline based. The surrogate server dedicated to a smart space maintains its own request queue [2, 3].
- c) The T-PDHS has many surrogates which collaborate over the wireline high-speed network in the infrastructure. The surrogate of a smart space serves those physicians within. If a physician has moved from one smart space to another, the service would continue due to the location-aware capability.
- d) Physicians interact with their surrogate via their mobile SFF devices in a wireless manner. Once a SFF (e.g. PDA) is in a smart space, which is practically a communication cell (e.g. inside a train station), end-to-end

wireless client/server interaction can begin [3, 12, 1].

- e) If a surrogate cannot serve a request it seeks help from others, in the cyber foraging mode [3]. This mode of distributed collaboration, under Markovian conditions, can be depicted by the M/M/n model, where n is the number of collaborating nodes/surrogates (or information stations). It yields speedup defined by $S = \frac{(1 - \delta/n)}{(1 - \delta)}$ from the

distributed parallelism, where δ is the surrogate utilization [5].

- f) The communication between a physician and the surrogate may have varied traffic patterns at different times. The communication streams from different physicians would merge at the SAP (service access point – Figure 1; the “+” symbol means merging) before requests enter into the surrogate’s queue. The pattern of the merged traffic can change suddenly and surge the surrogate’s request queue to overflow its buffer easily. As a result, there would be widespread retransmissions that produce long service RTT (i.e. slow response) and even occasional communication blackouts. In light of telemedicine [15] for which T-PDHS is an example, such phenomena are not acceptable at all. A logical solution to prevent such mishaps is to ensure that the buffer always covers queue under all traffic conditions. This is exactly the rationale behind the dynamic buffer tuning paradigm [6].

- g) All activities inside the channel for end-to-end communication are considered at the system level. Usually a request sent from a client would be routed through many routers, which have their own local reception queues, before reaching the destination. To prevent local routing congestion a router may throttle any sender that send too much in a short time by choke packets. This process is called active queue management (AQM), but this does not lower the chance of overflow due to merged traffic at the user level (i.e. surrogates and clients) [6]. In affect, AQM may even inadvertently aggravate retransmission because the message could be dropped at the user level due to buffer overflow only after it has passed through the channel and accrued a long delay. This is an intrinsic cost/effective problem in the AQM approach.

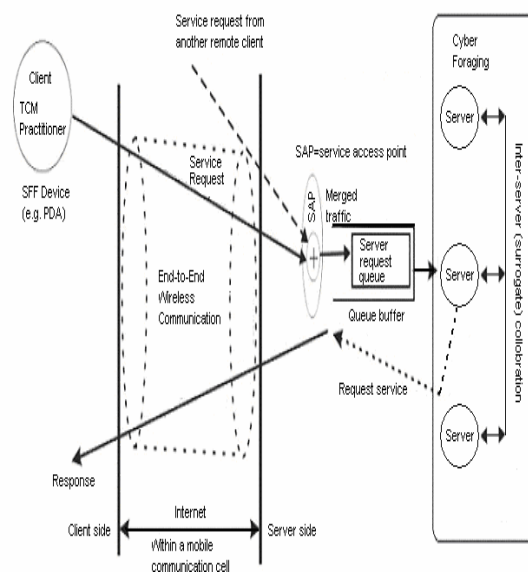


Figure 1. Pervasive T-PDHS infrastructure on the mobile Internet

Figures 2 and 3 are screen captures that show two wireless operations via the T-PDHS respectively: a) login request by a TCM physician to initiate pervasive medical consultation, and b) a request for the surrogate to retrieve the relevant patient record. The control bar shows some of the T-PDHS control icons, namely, Login, Patient Record, Prescription, and Dispensing. Each icon can be exploded for more detailed manipulations and operations.

In fact, field tests of the *basic* T-PDHS prototype with no dynamic tuning support indicated that its response time would vary over the 24 hours. Our analysis indicated that the response time variations were caused by the transient mass transit population through the smart space [7]. This finding concurs with those by others (e.g. [4, 8]). The mass transit can seriously increase the volume of communication between SFF mobile clients and the surrogate at peak hours. Although congestion may be alleviated by setting a maximum number of SFF-surrogate connections in a smart space, this solution cannot prevent surrogate buffer overflow due to traffic ill effects. The SFF-surrogate traffic intensity and pattern can change suddenly for the web follows the power laws [9]. It is especially difficult to predict meaningfully how/when traffic volume and pattern would change in a pervasive setup for wireless and wireline communications intertwine in a stochastic manner [10]. This paper is focused on how to apply the FLC to deal with the ill effects by traffic volume but not traffic

patterns. The negative impact by traffic patterns on system stability and response time will be investigated in the future work.



Figure 2. T-PDHS login interface



Figure 3. Patient record retrieval

2. Related Work

We propose to use the FLC (Fuzzy Logic Controller) to eliminate surrogate buffer overflows in the T-PDHS so that service roundtrip times for client/surrogate (server) interactions are shortened. FLC is the “PIDC + Fuzzy Logic + the $\{0, \Delta\}^2$ objective function” combination. The PID (i.e. P+I+D) controller or PIDC for brevity works with three main control elements: proportional (P) control in terms of the QOB (queue length over buffer length) ratio, integral (I) control in terms of the history of changes in the queue length, and derivative control (D) in terms of the rate of change in the queue length (i.e. dQ/dt).

By itself PIDC does not have a safety margin Δ and therefore overflows under serious perturbations [6]. The added fuzzy logic divides the PIDC control domain into a set of smaller fuzzy control regions, which are individually supported by a specific fuzzy rule or a “don’t care” state. The fuzzy rules together ensure that the buffer size always covers the queue length in an adaptive manner by the Δ safety margin. The inert “don’t care” states require no execution time and in this way they contribute to lower the overall FLC control cycle time. Figure 4 shows the algorithmic PIDC operation.

If $\{(dQ/dt > \text{prescribed_positive_threshold}) \text{ OR } [(dQ/dt \text{ is_positive}) \text{ AND} (QOB_i > \text{prescribed_positive_threshold})]\}$
Then $L_{now} = L_{now} + ICM; L_{now} \geq L_{minimum}$
Else If $\{(dQ/dt < \text{prescribed_negative_threshold}) \text{ OR } [(dQ/dt \text{ is_negative}) \text{ AND} (QOB_i < \text{prescribed_negative_threshold})]\}$
Then
 $L_{now} = L_{now} - ICM; L_{now} \geq L_{minimum}$

Figure 4. The basic PID controller (PIDC) algorithm

The PIDC parameters are: ICM (*integral control mechanism*) for integral or I control, L_{now} for current buffer length, $L_{minimum}$ for the minimum buffer size estimated from the past performance, QOB_i as the “queue length (Q) over buffer length” ratio in the i^{th} PIDC control cycle for proportional control, and dQ/dt as the current rate of change in Q for derivative control. ICM (equation (A2) is defined in terms of the current

RIC value (defined in equation (A1) where l marks the current cycle l , $l \geq 1$). RIC uses the current mean queue length $Queue_{CA_estimated}$ predicted by the M^3RT (*micro mean message response time*) entity as shown in equation (A0) [13, 14], as well as the mathematical average of the queue length, $Queue_{Reference}$. Equation (A3) is the enhanced version for equation (A0). P_{CON} is the damping factor that smoothens the convergence towards the estimated mean M_i (now realized as $Queue_{CA_estimate}$) for the time window of interest. The width of the i^{th} window is defined by the total time required to collect the $(F-1)$ m_j^i number of samples, for $j = 1, 2, \dots, (F-1)$. F is the flush limit chosen for the M^3RT operation. M_0 is the first sample recorded after the M^3RT has started running, and M_{i-1} is the feedback of the last predicted result into the current prediction cycle. B_a in equation (A2) is a constant or “seed” chosen for the particular ICM implementation.

$$M_i = \frac{M_{i-1} + \sum_{j=1}^{j=F-1} m_j^i}{F} \dots\dots(A0), M_0 = m_{j=0}^{i=1}, i \geq 1$$

$$RIC_l = \frac{Queue_{CA_estimated} - Queue_{Reference}}{Queue_{Reference}} \dots\dots(A1)$$

$$ICM = B_a \sum_{l=1}^{\infty} RIC_l \dots\dots(A2)$$

$$Queue_{CA_estimate} = \frac{P_{CON} * Q_{CA_estimate} + \sum_{j=1}^{j=f} m_j^i}{P_{CON} + f} \dots\dots(A3)$$

The M^3RT algorithm is the *micro* implementation of the *Convergence Algorithm* (CA) [14], which is derived from the *Central Limit Theorem* and therefore predicts the mean of any waveform accurately. Being *micro* the tool runs as a logical object, which can be invoked for a prediction via message passing anytime and anywhere. In contrast, a *macro* tool must be installed at the nodes that represent the two ends of a logical channel before measurement can start. The M^2RT (*mean message response time*) IEPM (Internet End-to-End Performance Measurement)

tool is a *macro* example; the predecessor of M^3RT [16].

QOB Reference		dQ/dt					
		NL	NM	NS	PS	PM	PL
0.7 0.8 0.9	NL	-	-	-	-	-	-
	L	-	-	X	X	+	+
	G	-	-	X	X	+	+
	MG	+	+	+	+	+	+

Figure 5a The FLC[4x6] configuration/design example ([4x6] matrix of fuzzy control regions)

The fuzzy rules in the FLC maintain the given Δ safety margin about the reference point for the objective function $\{0, \Delta\}^2$, symbolically represented by “0”. For the FLC prototypes the reference point is instantiated as the chosen QOB (*queue length over buffer length*) ratio or QOB_R . For QOB_R equal to 0.8 (i.e. $\Delta = 0.2$) the FLC operates in the QOB range from 0.6 to 1. The extant FLC model maintains Δ by tuning only the ICM value on the fly, based on the QOB and dQ/dt parameters. For example, the experimental results presented in this paper were produced with $QOB_R = 0.8$. When the FLC control enters an inert “don’t care” state, it requires no action. The inertness of the “don’t care” states offsets the FLC computational complexity due the fuzzy logic presence. As a result the FLC execution time is comparable to its much simpler PIDC predecessor. As the example in this paper, Figure 5a is the matrix of fuzzy regions for the FLC[4x6] design. The “dot” defines the $QOB_R = 0.8$, and X marks a “don’t care” state.

For the FLC[4x6] dynamic buffer tuner the linguistic variables are defined as follows:

- Current QOB ratio (or QOB_i)*: ML for *Much Less* than QOB_R , L for *Less* than QOB_R , G for *Greater* than QOB_R , and MG for *Much Greater* than QOB_R .
- Current dQ/dt* : NL for *Negative and Larger* than the threshold, NM for *Negative but Medium* to the threshold, NS for *Negative and Smaller* than the threshold, PS for *Positive and Smaller* than the threshold, PM for *Positive and Medium* to the threshold, and PL for *Positive and Larger* than the threshold.

The FLC control decision in the i^{th} control cycle depends on the current QOB_i and dQ/dt . It

may be *Addition (buffer elongation)* or “+”, *Subtraction (buffer shrinkage)* or “-” or *don’t care* “X”. Different fuzzy rules can be formulated for different FLC designs, for example:

Rule 1: If (QOB_i is L) AND (dQ/dt is NM) Then Action is “-”(Subtraction) AND $L_{new} = L_{old} - ICM$

Rule 2: If (QOB_i is L) AND (dQ/dt is NS) Then Action is “X”(Don’t care) AND $L_{new} = L_{old}$

Rule 3: If (QOB_i is L) AND (dQ/dt is PS) Then Action is “X”(Don’t care) AND $L_{new} = L_{old}$

Rule 4: If (QOB_i is L) AND (dQ/dt is PM) Then Action is “+”(Addition) AND $L_{new} = L_{old} + ICM$

L_{new} represents the new buffer length adjusted from the old value L_{old} by the adjustment size ICM computed by FLC on the fly. Figure 5b and 5c are membership functions of the two control parameters for FLC[4x6] tuner:

QOB and $\frac{dQ}{dt}$. The $\frac{dQ}{dt}$ membership function is in gradient, and the y-axis of Figure 6b is the degree of membership measurement. The x-axis is the gradient difference between two successive $\frac{dQ}{dt}$ measurements. For this design the values from a to f are: $a=0.003$, $b=0.002$, $c=0.001$, $d=0.001$, $e=0.002$ and $f=0.003$. Figure 5c shows the QOB membership function for the same design, and the x-axis is the QOB ratio that changes in a dynamic manner. The values for the p , q , r , s , t , u are respectively: 0.65, 0.7, 0.75, 0.85, 0.9 and 0.95. The current $\frac{dQ}{dt}$ and QOB values decides which fuzzy region that the FLC should operate. For example, if the degree of the $\frac{dQ}{dt}$ membership function is between b and c (i.e. $y = 0.5$) and that for QOB is between q and r , four fuzzy regions are candidates: [SL,NM], [SL,NS], [L,NM] and [L,NS]. With respect to Figure 6a, the possible operations are: -, -, and X, and the majority rule adopted by the FLC mechanism in this case selects the minus (-) operation. In fact, different selection rules can be formulated for specific aims. It is totally up to the FLC user.

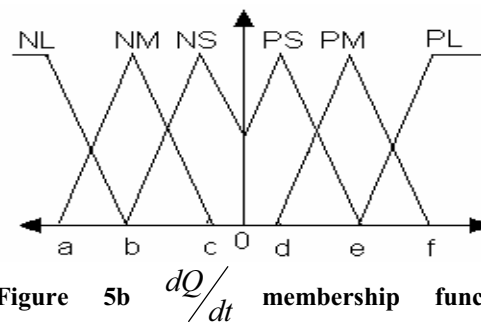


Figure 5b $\frac{dQ}{dt}$ membership function

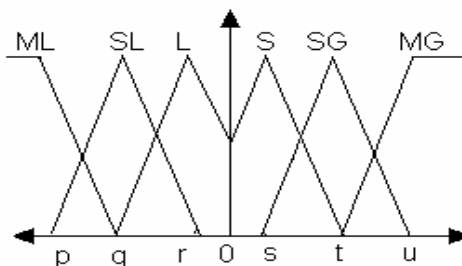


Figure 5c Membership function for QOB

3. FLC to Shorten T-PDHS Service RTT

The service RTT in the pervasive medical consultation process depends on the *average number of trials* (ANT) to get a successful transmission in an end-to-end physician/surrogate interaction. ANT depends on ρ , which is the channel error probability to cause retransmissions. The impact of ρ on ANT can be visualized from

the equation, $ANT = \sum_{j=1}^{N \rightarrow \infty} j[\rho^{j-1}(1-\rho)]$ or

$$ANT \approx \frac{1}{(1-\rho)}$$

Conceptually ρ encapsulates all the possible errors and faults such as buffer overflows, hardware glitches and software breakdowns. Such faults and errors can appear anytime because of the sheer size and heterogeneous nature of the Internet. Dynamic buffer tuning reduces ρ by stifling buffer overflow at the reception side [6]. Several dynamic buffer tuners have been proposed over the years. The FLC is chosen for improving the T-PDHS response because of its good qualities: a) it is fast, b) it unlikely produces buffer overflow while in control, and c) it locks no unused buffer space like the PIDC. Locking unused buffer space lowers the whole T-PDHS system performance because this deprives other tasks in the system of memory needed for smooth and efficient operations. Table 2.1 compares the different dynamic buffer tuners.

Tuner name (PIDC)	No. of Java lines	Average number of cycles to execute	Average number of TT cycles to settle to the steady state	Occasional overflow
Proportional Integral Derivative Controller (PIDC)	105	205	4725	Likely
Genetic Algorithm Controller (GAC); $GAC = PIDC + \{ \text{genetic algorithm} \}$	111	475	4995	Yes
Fuzzy Logic Controller (FLC); $GAC = PIDC + \{ \text{fuzzy logic} \}$	116	255	5220	No
Neural Network Controller (NNC); $NNC = \{0, \Delta\}^2 + \text{neural network}$	240	10800	10800	No

Table 2.1 Concise comparison of four dynamic buffer tuners known in the field

4. Experimental Results

The FLC usefulness for shortening the T-PDHS response time by eliminating surrogate buffer overflows was verified by simulations. These simulations were divided into two categories. In the first category known discrete waveforms or distributions (e.g. Poisson and self-similar) were generated to mimic the merge traffic (Figure 1) in light of the IAT (inter-arrival times) among the requests to the surrogate SAP. These waveforms verified that the “FLC + T-PDHS” combination indeed work stably in known traffic conditions. In the second category wireless traces collected in the Hong Kong Polytechnic University [11] were used to verify that the same combination indeed work for real environments. In all the simulations the same waveform would simultaneously excite the two dynamic buffer tuners running in parallel: PIDC and FLC. The aim is to compare the results by FLC and PIDC under the same traffic conditions. The comparison would

confirm that the FLC is indeed the right choice to yield a faster response for T-PDHS. Many simulations were conducted as follows: T-PDHS without dynamic tuner support, T-PDHS with FLC support, and T-PDHS with PIDC support only.

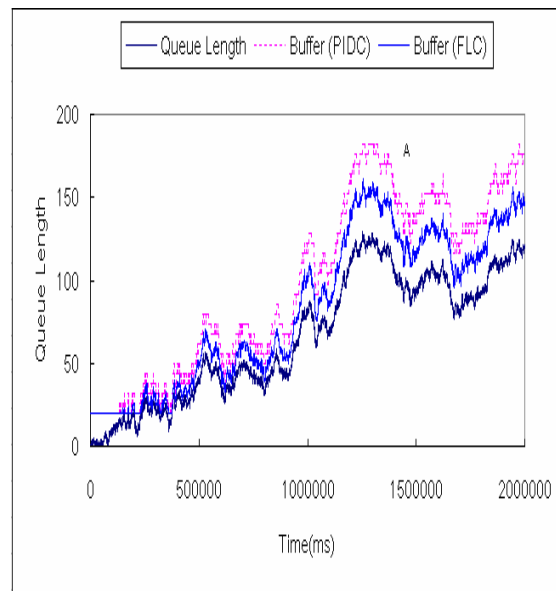


Figure 6. Unlike PIDC FLC locks no unused

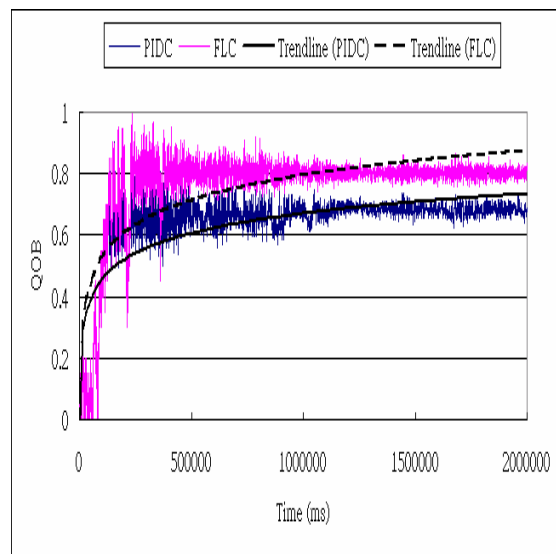


Figure 7. Faster FLC convergence to buffer space the steady state than PIDC

The preliminary results indicate that FLC converges faster to the steady state than PIDC. There is also less oscillation in the convergence process. Without dynamic buffer tuning support the basic T-PDHS had frequent surrogate buffer overflows. Figure 6 shows how FLC rectifies the PIDC problem of locking too much buffer memory

even when it is no longer needed (as shown by point A). This PIDC shortcoming would lower the overall T-PDHS performance because it deprives other T-PDHS tasks of needed memory. The FLC rectifies this PIDC shortcoming by consistently maintaining the prescribed safety margin between the buffer length and the queue length in a dynamic fashion. As a result unused buffer space would be recycled to benefit other tasks in the systems. Figure 7 shows that FLC has a much faster and less oscillatory convergence towards the steady state reference, $QOB = 0.8$ (i.e. 80%) or a safety margin of $\Delta = 0.2$ (20%). The FLC trend-line settled quickly to $QOB = 0.8$ after some overshoots but the PIDC trend-line would reach this reference asymptotically in theory. This explains why practically the FLC is a much more accurate and faster dynamic buffer tuner than PIDC. The advantage of settling quickly to the steady state is lessening the chance for surrogate buffer overflow. This leads to more effective pervasive consultations in telemedicine over the web.

5. Conclusion

This paper proposes to apply the Fuzzy Logic Controller (FLC) to effectively support more dependable client/server interactions in the T-PDHS. As a result, the improved channel dependability attained through dynamic buffer tuning shortens the T-PDHS response time. TCM physicians in a smart space can hook onto the same dedicated surrogate to conduct pervasive medical consultations. Any transient mass traffic through the smart space, however, can create an unpredictable volume and pattern for the merged traffic into the surrogate's queue. This kind data volume and traffic unpredictability would surge the surrogate queue to overflow its buffer if dynamic buffer tuning is absent. If this happens, the clients/physicians in the smart space would lose their chance to benefit from the cyber foraging, which speeds up the response even more due to the intrinsic distributed parallelism in the pervasive T-PDGS computing infrastructure. The FLC can help prevent surrogate buffer overflow by ensuring that the buffer always cover the queue by the given Δ safety margin. This makes the channel more dependable, as confirmed by the simulation results. Our analysis of these results indicated that FLC accuracy and convergence smoothness were actually affected by traffic patterns in different ways as well. In fact, at anytime Internet traffic can be stationary or chaotic (for unstable conditions). Stationary traffic can be SRD (short-

range-dependence) and LRD (long-range dependence). SRD includes Markovian patterns and LRD include self-similar and heavy-tailed distributions. Yet, observations from the "T-PDHS + FLC" simulations indicate that each traffic pattern might produce a specific and distinctive ill effect on the FLC accuracy and convergence smoothness. Thus, the next step planned for the research is to investigate what kind of correlations are there among traffic patterns and ill effects; for example, oscillatory convergence is considered an ill effect in the context of dynamic buffer tuning. If such correlations could be defined, then new ways could be proposed for FLC to shrug off traffic ill effects to the benefit of the T-PDHS response even more.

6. Acknowledgement

The authors thank the Hong Kong Polytechnic University for the A-PA9H and ZW93 research grants.

7. References

- [1] Jakob E. Bardram and Henrik B. Christensen, Pervasive Computing Support for Hospitals: An overview of the Activity-Based Computing Project, IEEE Pervasive Computing, 6(1), 44 - 51
- [2] D. Garlan, D.P. Siewiorek, A. Smailagic and P. Steenkiste, Project Aura: Toward Distraction-free Pervasive Computing, IEEE Pervasive Computing, 1(2), 2002, 22 - 31
- [3] C.A. Patterson, R.R. Muntz and C.M. Pancake, Challenges in Location-aware Computing, IEEE Pervasive Computing, 2(2), 2003, 80 - 89
- [4] H. Jamjoom, P. Pillai and K.G. Shin, Resynchronization and Controllability of Bursty Service Requests, IEEE/ACM Transactions on Networking, 14(4), August 2004, 582 - 594
- [5] T. Lewis, The Next 10,000 Years: Part 1, IEEE Computer, 29(4), 1996, 64 - 70
- [6] Wilfred W.K. Lin, Allan K.Y. Wong and Tharam S. Dillon, Application of Soft Computing Techniques to Adaptive User Buffer Overflow Control on the Internet, IEEE Transactions on Systems, Man and Cybernetics, Part C, 36(3), May 2006, 397 - 410
- [7] Wilfred W.K. Lin and Allan K.Y. Wong, Tharam S. Dillon, Elizabeth Chang, Detection of Fractal Breakdowns by the Novel Real-Time Pattern Detection Model (Enhanced-RTPD+Holder Exponent) for Web Applications, Proceedings of the 10th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing, 2007 (ISORC '07), 7-9 May 2007, 79 - 86

- [8] A. Malla, M. El-Kadi, S. Olariu and P. Todorova, A Fair Resource Allocation Protocol for Multimedia Wireless Networks, IEEE Transactions on Parallel and Distributed Systems, 14(1), January 2003, 63-71
- [9] A. Medina, I. Matta and J. Byers, On the Origin of Power Laws in Internet Topologies, ACM SIGCOMM, 30(2), 2000, 18-28
- [10] S. Molnar, T.D. Dang and A. Vidacs, Heavy-Tailedness, Long-Range Dependence and Self-Similarity in Data Traffic, Proceedings of 7th International Conference on Telecommunication Systems, Modelling and Analysis, March 1999, Nashville, USA,18-21
- [11] The Wireless LAN Traces, Department of Computing, Hong Kong Polytechnic University, <http://www4.comp.polyu.edu.hk/~cswklin/research/traces/wireless/>
- [12] M. Weiser, The Computer for the Twenty-First Century, Scientific American, September 1991, 94-104
- [13] Allan K.Y. Wong, Tharam S. Dillon, Wilfred W.K. Lin and May T.W. Ip, M²RT: A Tool Developed for Predicting the Mean Message Response Time for Internet Channels, Journal of Computer Networks, vol. 36, 2001, 557-577
- [14] Allan K.Y. Wong and Joseph H.C. Wong, A Convergence Algorithm for Enhancing the Performance of Distributed Applications Running on Sizeable Networks, the International Journal of Computer Systems, Science & Engineering, 16(4), 2001, 229-236
- [15] J.F. Kaar, International Legal Issues Confronting Telehealth Care, Telemedicine Journal, March 1999
- [16] Allan K.Y. Wong, May T.W. Ip and Tharam S. Dillon, M³RT: An Internet End-to-End Performance Measurement Approach for Real-Time Applications with Mobile Agents, Proc. of the ISPAN'2002, 2002, 119-124