APPLYING NEURO-FUZZY DYNAMIC BUFFER TUNING TO MAKE WEB-BASED TELEMEDICINE SUCCESSFUL

Jackei H. K. Wong, Chen Ye Zhu, Wilfred W. K. Lin and Allan K. Y. Wong
Department of Computing, The Hong Kong Polytechnic University Hung Hom, Kowloon, Hong Kong

Keywords: Neuro-Fuzzy Logic Controller, web-based telemedicine, mobile Internet, fast system response, TCM (Traditional Chinese Medicine), dynamic buffer tuning.

Abstract: We propose to make web-based medical consultation successful by applying the Neuro-fuzzy Dynamic Logic Controller (NFLC). This is achieved for the NFLC shortens the service response time for the physician, who answers the patient requests pervasively, by dynamic buffer tuning. The physician carries a SFF (small form factor) mobile device (e.g. PDA) that provides the interface for interacting wirelessly with rest of the web-based telemedicine system (WTS) on the mobile Internet. The WTS in this paper caters to Traditional Chinese Medicine (TCM) and therefore called TCM-WTS or simply T-WTS. The T-WTS usability relies on various factors such as correct information exchange, and fast system response. This paper focuses on the second factor by exploiting real-time dynamic buffer tuning as a solution.

1 INTRODUCTION

We propose to apply the NFLC (Neuro-Fuzzy Logic Controller), which is a dynamic buffer tuner to quicken the response of the extant T-WTS (TCM (Traditional Chinese Medicine) Web-based Telemedicine System), which was operated by the Purapharm Group of the Hong Kong SAR. The T-WTS response is slow and unreliable under heavy traffic conditions. Our preliminary study showed that slow response and channel unreliability could be caused by overflows at the server side. Overflows cause widespread retransmission and long service roundtrip time (RTT) (i.e. slow service response).

TCM is enshrined in the Hong Kong local law. Its popularity invigorates the local drive to make TCM reach every SAR corner and eventually the rest of the world. The NFLC can contribute to make this goal a success.

The T-WTS is distributed on the mobile Internet and allows pervasive interaction between a mobile TCM physician and the dedicated surrogate node/server assigned to the smart space (Patterson et al., 2003). Before interaction takes place the physician must move into a smart space, which is a communication cell that seamlessly supports various wireless technologies. The interaction relationship is one-surrogate-to-many-clients (i.e. physicians) or asymmetric rendezvous.

A physician provides T-WTS based medical consultations anytime and over any geographical location via a portable SFF (small form factor) mobile device (e.g. mobile phone). The device hosts a logical agent to provide the interface for remote interaction with the rest of the distributed T-WTS. With the SFF patient records can be created, stored and retrieved remotely. The mechanism to support all these activities in the background is the dedicated surrogate server. A physician can dispense prescriptions in a remote fashion. Figure 1 shows the T-WTS infrastructure as follows: a) it is operating pervasively over the mobile Internet that supports both wireless and wireline communications; b) end-to-end client/server interaction can be wireless and wireline (server is surrogate (Patterson et al., 2003)); c) T-WTS has many surrogates that collaborate over a wireline high-speed network. Every surrogate server is assigned to serve at least one smart space and those physicians (i.e. clients) within; and d) the physicians interact with their surrogate via mobile SFF devices in a wireless manner, e) if a surrogate cannot serve a request it seeks help from others, in the cyber foraging mode.

Cyber foraging under Markovian conditions is the M/M/n (M for Markov) model; n is the number nodes/surrogates/information-stations) in collaboration.

$$S = \frac{(1-\delta/n)}{(1-\delta)}$$

H. K. Wong J., Ye Zhu C., W. K. Lin W. and K. Y. Wong A. (2008).
APPLYING NEURO-FUZZY DYNAMIC BUFFER TUNING TO MAKE WEB-BASED TELEMEDICINE SUCCESSFUL.
In *Proceedings of the International Conference on Wireless Information Networks and Systems*, pages 151-156
DOI: 10.5220/0002024101510156

The speedup produced by the distributed parallelism for n is , where δ is the surrogate utilization. The traffic stream between a physician and the surrogate may have a distinctive character (e.g. self-similar) at a specific period and change suddenly. Since the all the traffic streams from different physician merge at the SAP (service access point – Figure 1; "+" symbol means merging), the resultant traffic pattern into the surrogate's queue can be undefined. It is not uncommon for such merged traffic to surge the surrogate's request reception queue to overflow its buffer easily, causing widespread retransmissions and thus long service RTT (i.e. slow response) (Lin et al., 2006). In light of telemedicine (Kaar, 1999) for which T-WTS is an example, such overflows are not acceptable at all. A logical solution to prevent such mishaps is to make buffer always covers queue. This is exactly the principle for dynamic buffer tuning paradigm (Lin et al., 2006).

Activities inside a channel for end-to-end communication are considered at the system level. Normally 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 and too fast by choke packets. This throttling process is called active queue management (AQM). It does not, however, reduce overflow due to merged traffic at the user level (i.e. surrogates and clients) (Lin et al., 2006).

Figures 2 and 3 are screen captures for the following wireless operations via the T-WTS respectively: a) login request by a TCM physician; and b) request to the surrogate for retrieving a patient record. The control bar shows some of the T-WTS icons, namely, Login, Patient Record, Prescription, and Dispensing. Each icon can be exploded for more detailed operations. Field tests of the *basic* T-WTS prototype with no dynamic tuning support indicated that its response time could vary significantly over 24 hours. Our analysis indicates that one cause of the variations was the transient mass transit population through a smart space (Jamioom 2004). This concurs with the findings by others (e.g. (Kaar, 1999)). The mass transit can seriously increase the traffic volume between SFF mobile clients and the surrogate at peak hours. One solution to lessen the congestion is setting a maximum number of SFF-surrogate connections in a smart space. This solution, however, cannot prevent surrogate buffer overflow caused by traffic ill effects. This paper only focuses on how to apply the NFLC to deal with traffic volume.

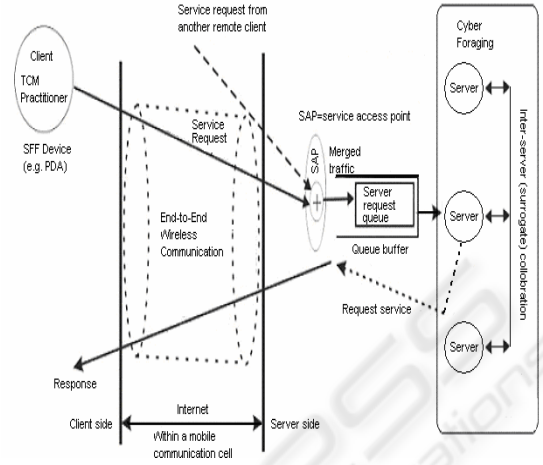

Figure 1: Pervasive T-WTS infrastructure.

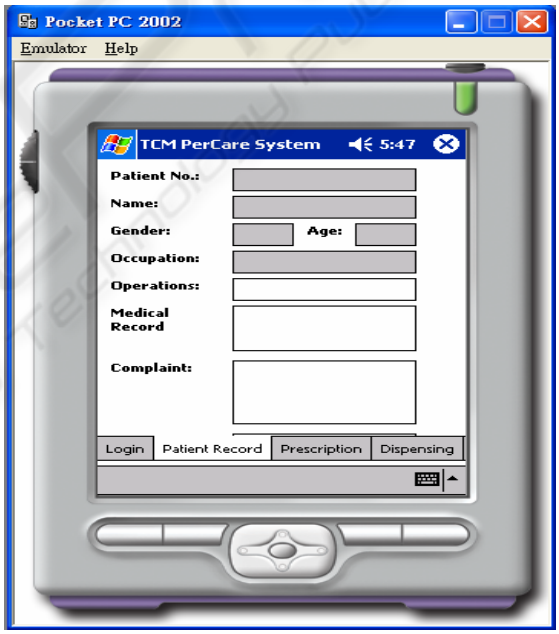

Figure 2: Patient record retrieval.

2 RELATED WORK

We observed that slow T-WTS system response could be caused by frequent buffer overflow at the reception side. This cannot be resolved by AQM (active queue management) alone (Braden et al., 1998). It is naturally to augment the AQM with user-level reception buffer overflow by dynamic buffer

Figure 3: Prescription preparation.

tuning ((Lin et al., 2006), (Lin et al., 2007)). For this reason propose to apply the novel NFLC (Neuro-Fuzzy Logic Controller) dynamic buffer tuner, which we developed for telemedicine systems. Our simulations showed that the NFLC had outperformed other extant dynamic buffer tuners (e.g. FLC (Fuzzy Logic Controller) (Lin et al., 2006)). The NFLC proposal here is partially based on a theoretical controller (Wang et al., 2001), which could not be realized due to absence of details such as: how to train the neural network part; how to specify the fault tolerance; and how to avoid diving by zero. The NFLC carefully addresses these shortcomings in light of usability.

3 THE NEURO-FUZZY LOGIC CONTROLLER

The novel Neuro-Fuzzy Logic Controller (NFLC) shortens the T-WTS response time by reducing the channel error probability ρ that encapsulates various hardware and software errors. One of the contributors to ρ is buffer overflow at the reception side (e.g. the surrogate server in the T-WTS setup). The ρ value affects the average number of trials (ANT) to get a successful transmission, for:

$$ANT = \sum_{j=1}^{N \to \infty} j[\rho^{j-1}(1-\rho)] \approx 1/(1-\rho) \cdot$$

Dynamic buffer tuning shrinks ρ because it eliminates reception buffer overflow at the user level (i.e. servers outside the channel domain – e.g. surrogates). The theoretical foundation of the NFLC can be summarized by the equations (3.1), (3.2) and (3.3), for which the parameters include: i) k for the current control cycle; ii) CA for Convergence Algorithm which is equation (3.3); the basis of integral (I) control of k^{th} cycle that involves the current sample of size f as well as the last predicted mean (i.e. M_{k-1}) as feedback;

$$|\sum_{n=1}^{n \to \infty} RIC_k| / n$$

iii) RIC_k which is the ratio using the typical/mode value of the f data points (i.e. s_k^j) in cycle k as the reference, can be positive or negative; and iv) QL is queue length. The actual NFLC output is $delB_n$; n for the current n^{th} control cycle since the buffer tuning process had started. The factor shows the integral nature of the $delB_n$ calculation.

$$RIC_k = (QL_{CA}^k - QL_{typical}^k) / QL_{typical}^k \quad (3.1)$$

$$delB_n = QL_{CA}^k [|\sum_{n=1}^{n \to \infty} RIC_k| / n] \quad (3.2)$$

$$QL_{CA}^k = M_k = (M_{k-1} + \sum_{j=1}^{f} s_k^j) / (f+1) \quad (3.3)$$

The NFLC has two main control parts: fuzzy logic (FL), and artificial neural network (ANN). The FL leverages two parameters, namely, the QOB (queue length over buffer length) ratio and the rate of changes in the queue length - dQ/dt . The FL output in the i^{th} cycle is a sign, $\sigma(\bullet)_i = \{+,-,?\}$ (i.e. add, subtract, or uncertain) for buffer adjustment, as depicted by equation (3.4). The $\sigma(\bullet)_i$ is the input to the ANN part that ascertains if "?" should be plus (+) or minus (-) so that the buffer adjustment size $delB_i$ can be properly computed as shown by equation (3.5).

$$FL_i(QOB_i, (dQ/dt)_i) = \sigma(\bullet)_i = \{+,-,?\} \quad (3.4)$$

$$delB_i = \sigma(\bullet)_i QL_{CA}^k (|\sum_{n=1}^{n \to \infty} RIC_k| / n) \quad (3.5)$$

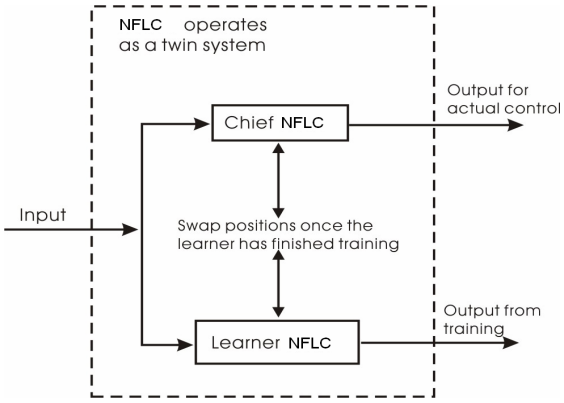

Figure 4: NFLC is a swapping twin system.

Table 1: Concise 7-step NFLC pipeline.

Procedure	Input	Output
Sample queue length Q and inter-arrival times (IAT) for requests	Time series of Q and the IAT among requests	Expected Q computed for the interval
Normalize Q and dQ/dt	Q and its rate of change dQ/dt	Normalized Q and dQ/dt
Compute fuzzy set; membership functions for Q & dQ/dt	Normalized Q and dQ/dt	Fuzzy set of Q & dQ/dt or membership functions ($2x3$ outputs)
Train/learn by ANN for $\{0,\Delta\}^2$	Fuzzy Q and dQ/dt (6 values)	Predicted next normalized Q
de-normalize predicted Q	Predicted next normalized Q	Predicted next de-normalized Q
Compute $delB_i$	Predicted next de-normalized Q	Predicted next de-normalized buffer length B ($B_{\min imun}$ considered)
Tune buffer size by $delB_i$ and for the ascertained $\sigma(\bullet)_i$	Predicted next de-normalized buffer length B ($B_{\min imun}$ considered)	Fulfilling the equations (3.4) and (3.5)

In operation, the three basic NFLC modules (i.e. *Chief* NFLC, *Learner* NFLC and CA) are running in parallel. The CA execution time has no impact on the execution time NFLC for timing analyses of by using the *Intel's VTune Performance Analyzer* showed that on average the NFLC needed 9800 clock cycles to execute and CA needs only 300 clock cycles.

If the *Chief* needs M_k (equation (3.3) for its computation, it simply fetches the current value directly from the much faster CA entity. The importance of the fast CA is to instantaneously capture system changes. This is essential for accurate and qualitative control. The *Chief* and *Learner* modules form a twin system (Figure 4). While the *Chief* NFLC performs actual control the *Learner* acquires new knowledge by supervised training. The teacher signal is the given safety margin Δ of the $\{0,\Delta\}^2$ objective function, which was absent in the concept proposed in (Lin et al., 2006). Training completes once if the output from the *Learner* is within the $\pm\Delta$ band. The current *Chief* swaps position with the *Learner* that has just completed its training cycle.

The internal NFLC dynamics goes through a pipeline of 7 steps: i) sample queue length Q and IAT (inter-arrival times) of requests; ii) normalize Q and dQ/dt ; iii) compute fuzzy set; iv) train/learn by ANN; v) de-normalize the predicted Q ; vi) compute $delB_i$; and vii) tune buffer size by $delB_i$ with respect to the sign ascertained for $\sigma(\bullet)_i = \{+,-,?\}$; Table 1.

To recap, NFLC is represented by the following elements: i) the fuzzy logic (FL) to determine the sign of dynamic buffer adjustment (i.e. to elongate or shorten the buffer length); ii) if the FL could not decide the sign for $\sigma(\bullet)_i$ then the ANN downstream would ascertain it before computing the dynamic buffer size $delB_i$ for the current i^{th} adjustment/control cycle; and iii) the above operations can be summarized as a 7-step pipeline procedure.

4 EXPERIMENTAL RESULTS

The NFLC power in shortening the T-WTS response time by eliminating surrogate buffer overflows was verified by simulations. These simulations are separated into two categories. In the first category known discrete waveforms or distributions (e.g. Poisson and self-similar) were used to mimic the merge traffic in light of the IAT (inter-arrival times)

among the requests to the surrogate SAP. These waveforms verified that the "NFLC + T-WTS" combination indeed worked stably in known traffic conditions. In the second category wireless traces collected in the Hong Kong Polytechnic University (Lin) were used to verify that the same combination indeed worked for real situations. In all simulations the same waveform would simultaneously excite the two dynamic buffer tuners running in parallel: PIDC and NFLC. The aim was to compare the results by NFLC and PIDC under the same traffic conditions. The comparison should confirm if the NFLC was the right choice to yield a faster T-WTS response. Many simulations were conducted: i) T-WTS with no dynamic buffer tuner support; (ii) T-WTS with NFLC, and iii) T-WTS with PIDC. The preliminary results indicate that NFLC converges faster to the steady state reference than PIDC. There was less oscillation in the control process as well. When dynamic buffer tuning was absent the T-WTS produced frequent surrogate buffer overflows.

Figure 5 shows how NFLC rectified the PIDC problem, which was locking too much buffer memory even when it was no longer needed. This PIDC problem lowered T-WTS performance because it deprived other T-WTS tasks of needed memory. The NFLC achieved this by rigorously maintaining the given safety margin Δ between the buffer length and the queue length on the fly. Surely, the NFLC is a more accurate, smoother, faster, and usable dynamic buffer tuner than PIDC. The benefit of settling quickly to the steady state is less or no buffer overflow and thus shorter roundtrip time (RTT) (i.e. quicker system response).

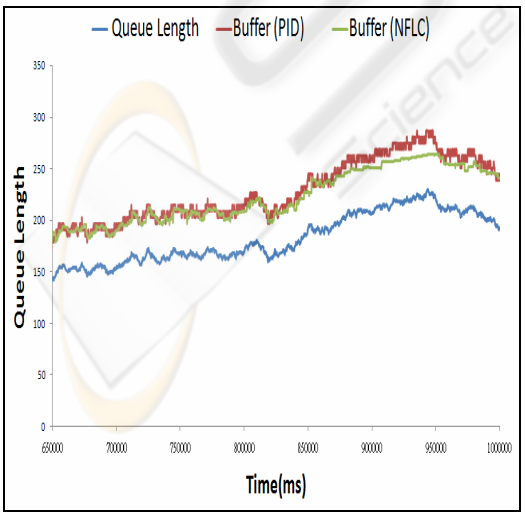

Figure 5: NFLC unlocked unused buffer space.

5 CONCLUSIONS

The NFLC (Neuro-Fuzzy Logic Controller) is proposed to yield shorter T-WTS response. This is achieved for NFLC produces more dependable client/server interaction over an end-to-end channel. This advantage becomes obvious if NFLC performance is compared to the PIDC's. TCM physicians in a smart space need to hook onto the dedicated surrogate to conduct pervasive medical consultations. The response, however, can be seriously affected by the transient mass through the smart space. This mass can create unpredictable traffic volume and pattern for the merged traffic that enters the queue of the dedicated surrogate. If dynamic buffer tuning is absent, the merge traffic could surge the queue to overflow the surrogate buffer. If this happens, the clients in the smart space would suffer from long service RTT and their chance to benefit from the cyber foraging in the pervasive computing infrastructure. The NFLC prevents surrogate buffer overflow by ensuring that the buffer always cover the queue by the given Δ safety margin. This makes the channel dependable, as confirmed by the simulation results. The focus of this research is to explore if the proposed NFLC can indeed prevent user-level buffer overflow effectively. But, we analyzed the simulation results as well for any possible correlation between traffic patterns and NFLC accuracy. Our analysis indicated that such a correlation exists. Internet traffic aggregates (time series) can be stationary or chaotic (unstable), and stationary traffic is either SRD (short-range-dependence) or LRD (long-range dependence). SRD includes Markovian traffic time series and LRD has self-similar and heavy-tailed patterns. As observed from the "T-WTS + NFLC" simulations, each traffic pattern could produce distinctive ill effect on NFLC control accuracy and convergence smoothness to the steady state. Therefore, the next logical step for the research is to explore and establish the correlation between traffic patterns and their ill effects (e.g. oscillatory convergence).

ACKNOWLEDGEMENTS

The authors thank the Hong Kong Polytechnic University for the A-PA9H research grant.

REFERENCES

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

J.F. Kaar, International Legal Issues Confronting Telehealth Care, Telemedicine Journal, March 1999

Lin, The Wireless LAN Traces, Department of Computing, Hong Kong Polytechnic University, http://www4.comp.polyu.edu.hk/~cswklin/research/traces/wireless/

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), 2006, 397-410

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, Proc. 10th IEEE Int'l Symposium on Object and Component-Oriented Real-Time Distributed Computing, May 2007, 79 - 86

C.A. Patterson, R.R. Muntz and C.M. Pancake, Challenges in Location-aware Computing, IEEE Pervasive Computing, 2(2), April-June 2003, 80-89

Braden et.al., Recommendations on Queue Management and Congestion Avoidance in the Internet, RFC 2309, April 1998

D. Wang, Allan K.Y. Wong, and Tharam S. Dillon, Heuristic Rule Based Neuro-Fuzzy Approach for Adaptive Buffer Management for Internet-based Computing, FUZZ-IEEE, 2001 .