Improving the Performance of DCH Timing Adjustment in 3G Networks

Gaspar Pedreño, Juan J. Alcaraz, Fernando Cerdán, and Joan García-Haro

Department of Information Technologies and Comunications
Polytechnic University of Cartagena (UPCT)
Plaza del Hospital, 1, 30202 Cartagena, SPAIN
{gaspar.pedreno,juan.alcaraz,fernando.cerdan,joang.haro}@upct.es

Abstract. The main goal of *Transport Channel Synchronization* is that frames sent by the RNC arrive in time to the Nodes B for transmission over the air interface. The 3GPP specifies a simple *Timing Adjustment* algorithm that tracks the delay of the *Iub* link (interface defined between RNC and Node B) by adding or subtracting a constant step. We show that this scheme reacts too slowly under abrupt delay variations and may become unstable in high delay scenarios. This paper proposes an alternative mechanism denominated *Proportional Tracking Algorithm* (PTA) which ensures stability in any scenario and improves performance compared to the classic one. The stability analysis of PTA is based on control theory and the performance is measured by means of an UTRAN simulator considering realistic traffic conditions in the Iub.

1 Introduction

In UMTS WCDMA access networks, link-layer communication between base stations (*Nodes B*) and *Radio Network Controllers* (RNCs) is governed by the *Frame Protocol* (FP), specified by the 3GPP in [1,2]. FP is responsible, among other things, of the *Transport Channel Synchronization* which is applied to dedicated channels and aims at delivering each frame to the Node B in time to be transmitted over the air interface within its corresponding *Transmission Time Instant* (TTI). This procedure, known as *Timing Adjustment*, basically involves adjusting the sending time instant of the data frames from the RNC to the Node B in order to make them arrive within a predefined receiving window. This receiving window is defined per each transport channel by two parameters (see Figure 1): *Time of Arrival Window Startpoint* (TOAWS) and *Time of Arrival Window Endpoint* (TOAWE). This window should not be too large in order to minimize the buffering time of the frames at the Node B. Frame queuing should be done in the RNC, where the scheduling and the resource management processes are located.

Figure 1 shows the possible arrival instants and temporal references. When a frame arrives outside its window, the Node B responds with a *Timing Adjustment* (TA) frame containing the *Time Of Arrival* (TOA) of the received frame. The RNC will use this feedback information to adjust the sending time instant of the

following frames, trying to steer the arrival time of the frames into the window. Therefore, for each TA, the RNC modifies the delay estimation (*offset*) of the corresponding FP flow. If a frame arrives after the *Last Time of Arrival* (LTOA) instant, the Node B will be unable to process it before the TTI of this frame, which is, therefore, discarded. For a more detailed explanation of the related signaling the reader is referred to [1].
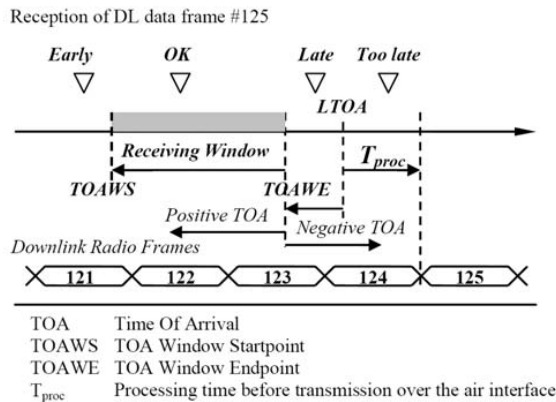

Fig. 1. Instants of frames reception at the Node B.

The work in this paper is motivated by the fact that UTRAN frame synchronization has not received much attention in scientific research. The scientific literature so far has mainly considered a simple algorithm consisting of adding or subtracting a constant step to the offset estimated at the RNC according to the TOA value contained in the *Timing Adjustment* frames (TAs):

$$t_{n+1} = t_n + t_{TTI} - K \quad if\ TOA < 0 \tag{1}$$
$$t_{n+1} = t_n + t_{TTI} + K \quad if\ TOA > 0$$

where t_{TTI} is the duration of a TTI and t_n is the transmission time instant of frame n. In the rest of this paper, we refer to this scheme as *classic algorithm*. In [3], the effect of the window parameters on the Iub link capacity for voice traffic is evaluated but authors assume that 3G nodes implement the classic algorithm and no improvement is suggested. The work presented in [4] is, to the best of our knowledge, the only one that presents an alternative timing adjustment method, and the approach suggested follows an additive increase, multiplicative decrease principle. The authors propose a modification of the classic algorithm by which the offset is decremented in the absence of TA frames, and the increment, applied upon a late frame arrival, is higher than the decrement. Therefore, in a scenario of steep delay changes its operation is analogue to the classic one.

In this paper, a new timing adjustment mechanism is presented, the *Proportional Tracking Algorithm* (PTA). Our proposal is based on correcting the offset

proportionally to the distance between the TOA and the center of the window. PTA is analyzed by means of discrete-time control theory which let us adjust the gain (K) of the system assuring stability for different delay situations. The behavior of the algorithm under different Iub delays and its response to sharp delay changes is studied and compared to the classic algorithm by means of computer simulation. The global performance is evaluated considering realistic traffic conditions in the Iub. The simulation results confirm that PTA appreciably improves performance (lower buffer levels at Nodes B, fewer frame losses and less signaling), adds robustness and simplifies configuration compared to the classic algorithm. It should be noted that our proposal does not require any changes in 3GPP specifications.

The rest of the paper is organized as follows. Section 2 describes our proposal theoretically as well as its modeling as a discrete-time control system including the stability analysis. Section 3 explains and shows the results of the simulation-based study conducted about the parameter configuration of each algorithm. In this section we compare both schemes under realistic traffic conditions, and evaluate their reaction to abrupt delay variations. Finally, Section 4 offers the conclusions of this work.

2   Proposed Algorithm

In this section, we introduce an alternative scheme based on a classical control model to perform *Transport Channel Synchronization*: the *Proportional Tracking Algorithm* (PTA).

As explained in Section 1, when a data frame arrives out of the receiving window, the Node B sends a signaling frame (*TA frame*) indicating the *Time Of Arrival* (TOA) of that data frame. Until now, no algorithm made use of this valuable information. Our proposal makes use of this TOA to schedule the transmission of the following data frames from the RNC. The goal of PTA is to place the arrival time of the frames in the center of the window. This instant is chosen because it minimizes the signaling and reduces excessive buffering time at the Node B. According to this principle, upon a *TA* frame arrival, the RNC makes the following correction in the sending time of the next frame:

$$t_{n+1} = t_n + t_{TTI} + K \cdot (TOA - TOAWS)/2) \tag{2}$$

That is, the next sending time will be advanced or delayed K times the difference between the center of the window and the instant of arrival. If the frame arrives after the end of the window ("late arrival"), $TOA - TOAWS/2$ is negative and, therefore, the next frame will be advanced. This advance involves a similar increment in the offset estimation performed at the RNC. On the other hand, if the frame arrives before the beginning of the window ("early arrival"), the following frame will be delayed. Whereas the frames arrive inside the window, there is no signaling sent and the RNC will continue transmitting frames with the same synchronization.

Figure 2 illustrates the operation of PTA and the delays involved. We define the *Round Trip Time* (RTT) of the FP control system as the time elapsed between the sending time of a frame and the moment when the corresponding *TA* frame is applied to another frame. The number of TTIs in the RTT is $r = \lceil RTT/t_{TTI} \rceil$. Defining j and m as the downlink and uplink delay in TTIs respectively, $r = j + m$. Therefore, we can express (2) as:

$$offset_{n+1} = offset_n + K \cdot e_n \qquad (3)$$

where e_n is the error between the offset applied to frame $n - r$ ($offset_{n-r}$) and the delay perceived in time-slot $n - m$ ($delay_{n-m}$), i.e. $e_n = delay_{n-m} - offset_{n-r}$. The delay r has a great impact on the stability and the performance of the system, as we explain later.
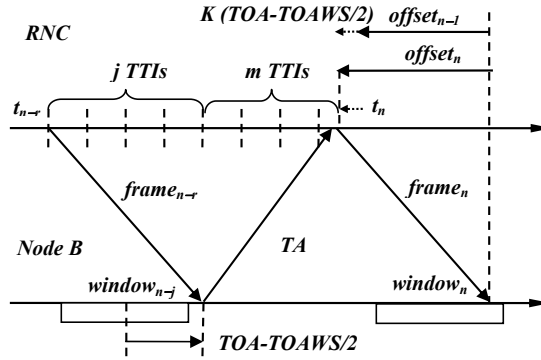

Fig. 2. Time diagram of the offset adjustment algorithm.

2.1 Stability analysis of the algorithm

In this section we use a discrete-time control theory approach to study the stability of the proposed algorithm and to adjust gain K according to a desired performance of the step response. The simulation results shown in Section 3 confirm the benefits of our scheme compared to the classic algorithm.

Both PTA and the classic algorithm are systems where the signals involved, i.e. the delay, the error and the offset, take one value per frame. Therefore, we can model this algorithm as a discrete-time control system making the following assumptions:

1. The RNC transmits one frame per TTI, i.e. there are no idle times for data traffic in the period under study.
2. The RNC has TOA information in every time slot. This implies that the window size ($TOAWS$) is $w = 0$ (see Fig. 2).

3. The inter-departure time between frames in the RNC is constant and equal to t_{TTI}. Therefore, the model neglects the variation in the TTI duration caused by the offset correction each time slot.

Figure 3 shows a diagram of the control system for the PTA mechanism, where the sampling frequency is $1/t_{TTI}$ and the blocks $1/z^j$ and $1/z^m$ represent respectively the downlink and uplink delays. The signal $u(n)$, corresponds to $delay_n$ and acts as the control signal, while $x(n)$ is $offset_n$.
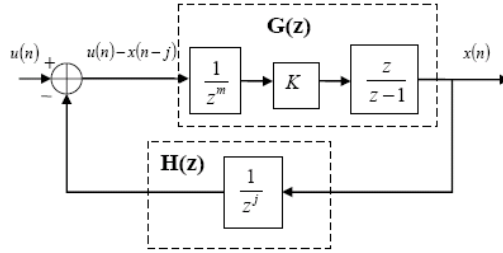

Fig. 3. Discrete-time control model of the Adaptive Proportional Tracking Algorithm (PTA).

Expressing (3) in terms of the signals in Figure 3 we have:

$$x(n) = x(n-1) + K(u(n-m) - x(n-r)) \tag{4}$$

Taking the z-transform in (4) we get:

$$X(z) = z^{-1}X(z) + K \cdot z^{-m}U(z) - K \cdot z^{-r}X(z) \tag{5}$$

Resulting in the following transfer function:

$$\frac{X(z)}{U(z)} = \frac{K \cdot z^j}{z^r - z^{r-1} + K} \tag{6}$$

In order to adjust the gain K, we use a typical performance parameter in classic control design [5], the overshoot percentage of the first peak of the response, M_p. Considering an step input of the form $u(n) = C$ for $n > 0$, and an output which first peak equals (A_{max}), the definition is:

$$M_p(\%) = \frac{A_{max}}{C} \cdot 100 \tag{7}$$

The above equation is related to the damping ratio (ζ) by

$$\frac{A_{max}}{C} = e^{-\zeta\pi/\sqrt{1-\zeta^2}} \tag{8}$$

Using (8) we obtain the ratio (ζ) for a given overshoot. The gain K is obtained solving the characteristic equation $1 + G(z) \cdot H(z) = 0$, which can be expressed as $F(z) = -1$, where $F(z) = G(z) \cdot H(z)$. Identifying $G(z)$ and $H(z)$ in our system (see Figure 3) we obtain:

$$F(z) = \frac{K \cdot z}{z^r (z - 1)} = -1 \tag{9}$$

If $\zeta < 0$, the poles of the system can be written as $z = exp(-2\pi\zeta\omega/\sqrt{1 - \zeta^2})$ according to [5], where $\omega = \omega_d/\omega_s$ and ω_s is the sampling frequency in rad/s. In our case $\omega_s = 2\pi/t_{TTI}$, where t_{TTI} is fixed to 10 ms in the channels under consideration. Given that $F(z)$ is a complex magnitude, (9) can be split into two equations, one for the phase of $F(z)$, equal to π radians, and the other for the module that must be equal to 1. The phase condition yields the equation:

$$(r - 1)2\pi\omega + arg(C^\omega e^{j2\pi\omega} - 1) = \pi \tag{10}$$

Where $C = e^{-2\pi\zeta/\sqrt{1-\zeta^2}}$. From (10) we obtain ω which is applied to the module condition to obtain K:

$$K = C^{\omega(r-1)} \left| C^\omega e^{j2\pi\omega} - 1 \right| \tag{11}$$

For the particular case of $\zeta = 0$, we get the critical K value, i.e. the gain above which the system becomes unstable. These values can be checked with the Jury stability test [5] for discrete time systems. Table 1 summarizes K for different overshoots (in percentage) and delay values (r, in number of TTIs).

Table 1. Gain K for different overshoot percentages.

r	2%	5%	10%	15%	20%	$\zeta \to \infty$
2	0.313	0.346	0.389	0.428	0.464	1
3	0.186	0.207	0.233	0.257	0.280	0.618
4	0.133	0.140	0.167	0.184	0.200	0.445
5	0.103	0.114	0.129	0.143	0.156	0.347
6	0.084	0.094	0.106	0.117	0.127	0.285
7	0.071	0.079	0.089	0.099	0.108	0.241

From this theoretical analysis we deduce that the gain K should be adjusted depending on the measured offset in the RNC in order to keep the stability of the algorithm. Therefore, the algorithm has to update K automatically according to the downlink delay estimation performed in the RNC. This operation requires the FP to keep a set of pre-calculated values of K. In our implementation we take the set corresponding to a 10% overshoot. The value of K is automatically selected from this set according to the Iub delay estimated by the RNC (*offset*), resulting in a dynamic adjustment.

3 Simulation Results

In this section, performance results obtained by computer simulation are discussed. To test both algorithms over Dedicated Channels (DCHs), we have developed a detailed implementation of the Frame Protocol, according to 3GPP specifications [1,2], using OMNeT++, an event-driven C++ simulation tool [6].

Figure 4 illustrates the simulated network topology, consisting of a point to point link between the RNC and the Node B. The bandwidth of the *Iub* interface was set to 1.7 Mbit/s, roughly equal to the net rate of an E1 PDH link subtracting 10% for signaling and control traffic.
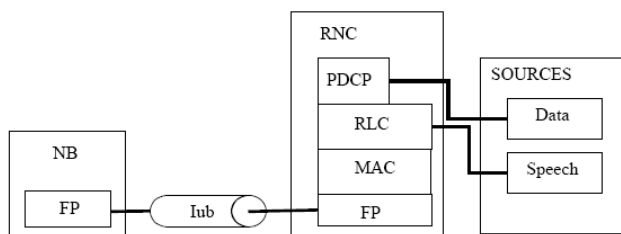

Fig. 4. Simulated network topology.

Both voice and data traffic are enabled in the simulations. Speech sources are based on *Adaptive Multi-Rate* (AMR) codecs while each data source is governed by a WWW application. For a more detailed description of the used data and speech sources the reader is referred to [7]. Each of these sources uses a DCH at a constant rate of 384 Kbps.

3.1 Response of both Algorithms to Abrupth Delay Changes

Foremost, we study the behavior of both algorithms when the Iub delay undergoes a sharp delay change. Given this situation, the simulation results show that PTA reacts faster than the classic algorithm avoiding excessive frame losses and signaling and, simultaneously, keeping buffer levels low. Besides, the classic algorithm can become unstable depending on the value of the gain K, requiring a cautious configuration. These dramatic changes of the Iub delay characterize realistic situations that may be caused by several reasons:

- A simultaneous connection/disconnection of several sources in a Node B causing an abrupt increase/decrease of traffic toward that Node B.
- An intermediate node in the UTRAN transport network handling a large amount of traffic.
- When traffic engineering or path restoration mechanisms re-route Iub flows. See [8] and the references therein.

In the examples provided in [9], one-way delay of different Iub branches ranged from 12 to 31 ms considering voice traffic and ATM transport. This gives an idea of the delay differences within the network connecting 3G access nodes. If data services and IP transport are considered, these delays can be larger. In this paper, we considered delay changes ranging from 10 ms to 60 ms although our algorithm can adapt to even longer delays.

In this section, we evaluate two scenarios where the Iub connection experiences a sharp delay change (an increment and a decrement). For the experiments performed one active speech source is considered and the parameters *TOAWS* and *TOAWE* have been set to typical values [3], 10 ms and 5 ms respectively.
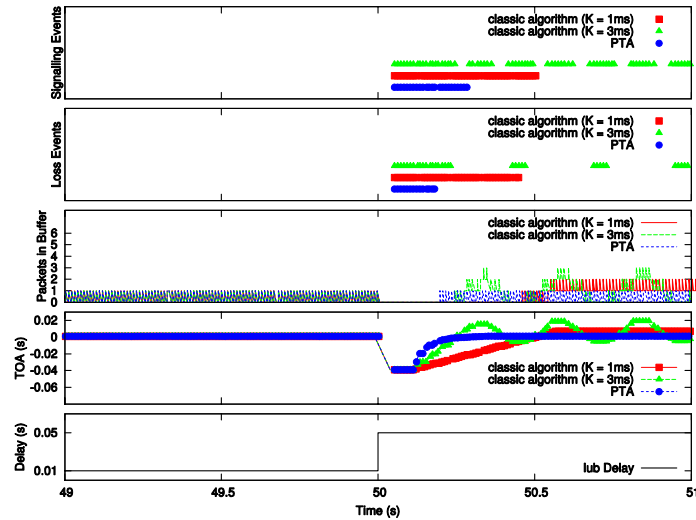

Fig. 5. Reaction of both algorithms to a delay increment

Figure 5 illustrates the operation of PTA and the classic algorithm with $K = 1$ ms and $K = 3$ ms in the delay increase scenario. The *Time Of Arrival* (TOA) of the FP flow is also shown. The Iub interface is initially configured with a propagation delay of 10 ms. At simulation time $t = 50$ ms, the Iub delay switches to 50 ms. This change implies that frames start to arrive after the receiving window causing the transmission of TA frames (*signaling events*) and the discarding of the frames arriving too late (*loss events*), as shown in Figure 5. Consequently, the buffer in the Node B empties until the sending time of new frames is adjusted in the RNC so that they arrive on time to be transmitted over the air interface.

These traces show that PTA behaves clearly better and stabilizes much faster than the classic algorithm for any of its configurations. With $K = 3$ ms (appropriate for an Iub delay of 10 ms) the classic mechanism looses stability after the

delay increase and, with $K = 1$ ms (appropriate for an Iub delay of 50 ms), the response of the classic algorithm is appreciably slow compared to PTA.
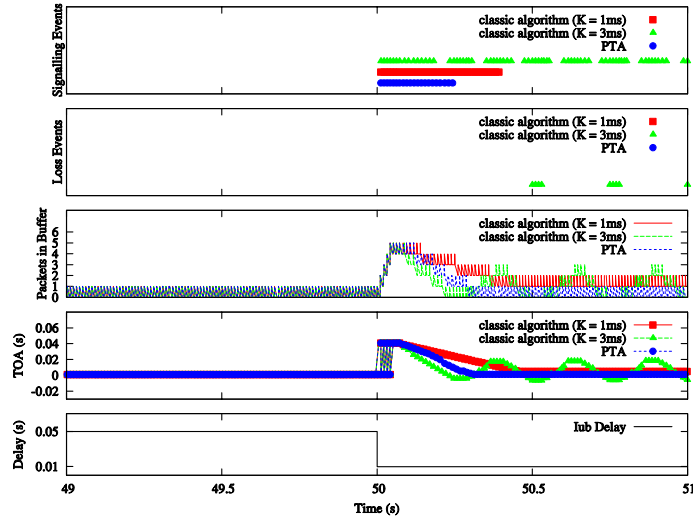

Fig. 6. Reaction of both algorithms to a delay decrement.

The results of the delay decrease experiment are depicted in Figure 6. The Iub interface is initially configured with a propagation delay of 50 ms and, at simulation time $t = 50$ ms, it switches to 10 ms. After this change, frames start to arrive before the receiving window and, therefore, they generate signaling but are not discarded. However, a delay decrease has an undesirable effect: overbuffering in the Node B. For this reason, the timing adjustment algorithm should react as quickly as possible to avoid an excessive buffering but, at the same time, assuring a stable response of the system. As we can see in Figure 6, the classic algorithm with K = 3ms is the fastest in terms of buffer draining but it becomes unstable. On the other hand, the classic algorithm with $K = 1$ ms remains stable but its response is too slow. PTA presents the most balanced behavior. It is able to follow the delay variation quickly thanks to the proportional strategy while the dynamic variation of K (based on Table 1 values for 10% overshoot) assures stability. Compared to the classic algorithm, PTA shows a better behavior in every measurement given a delay change (lower buffer levels, fewer frame losses and less signaling).

3.2 Performance Comparison under Realistic Traffic Conditions

In this Section, both algorithms are studied and evaluated using realistic traffic (30 speech sources and 6 data sources). The performance figures measured are

the frame loss ratio, the signaling ratio and the maximum buffer level at Node B. The signaling ratio is the ratio between the number of TA frames sent by the Node B and the total number of data frames received at the Node B. The frame loss ratio is the ratio between the number of frames arriving "too late" and the total number of frames received at the Node B. Both ratios are expressed in percentage while the buffer levels are expressed in number of frames.
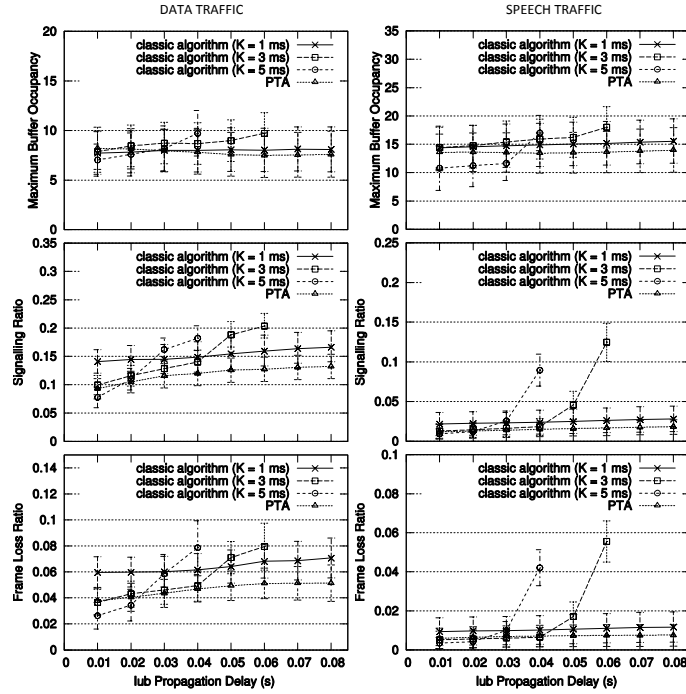

Fig. 7. Performance of both algorithms under realistic traffic conditions for different Iub delays.

In the first test we compare both algorithms given different Iub propagation delays, set at the beginning of each simulation and remaining constant during the simulation time. The parameters $TOAWS$ and $TOAWE$ have been established to 20 ms and 5 ms respectively. Performance results are depicted in Figure 7. Each value is obtained by averaging 10 simulation runs and the error bars correspond to a confidence interval of 95% according to a t-student distribution.

From Figure 7, it can be seen that the classic algorithm with high K values performs similarly to PTA in case of low Iub delays (10 or 20 ms) but, however, it becomes unstable with larger Iub delays. On the other hand, the classic algorithm with $K = 1$ ms remains stable for a wide range of Iub delays but its performance

is lower than that of PTA in every scenario, because of its slow response. Speeding up this response requires a higher K, which causes instability at higher delays. These facts illustrate that the main drawback of the classic algorithm is the trade-off between responsiveness and robustness against oscillations.
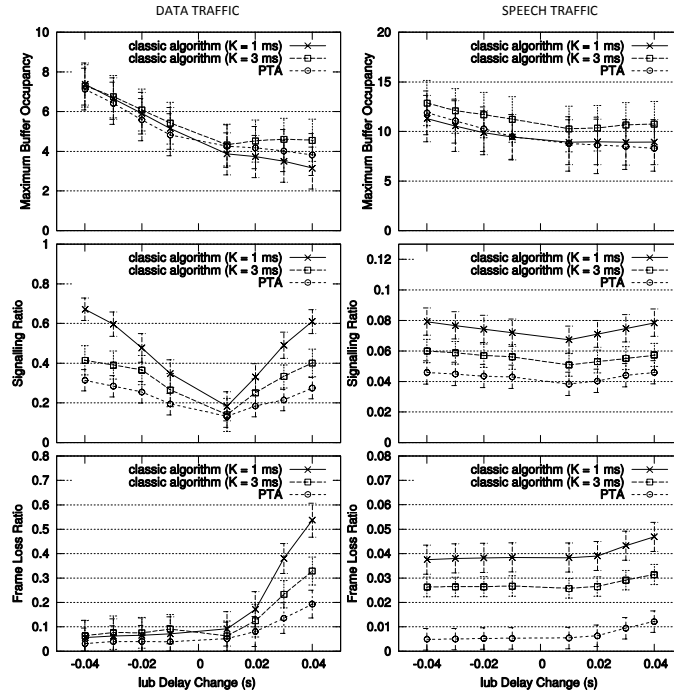

Fig. 8. Performance of both algorithms in a realistic traffic scenario with sharp Iub delays changes.

Next, we study a situation where the Iub delay experiences an abrupt change. The traffic considered is the same as in the previous scenario. Figure 8 illustrates the performance figures for Iub delay increases (positive sign) and decreases (negative sign) of different magnitude, ranging from 10 ms to 40 ms. The *TOAWS* is set to 10 ms and *TOAWE* equals 5 ms. Each performance figure is measured during the period of 60 seconds after the delay change. The average values and the confidence intervals (95%) are obtained from 10 simulation runs.

The performance figures shown in Figure 8 are in accordance with the results observed in Section 3.1, where only a single FP flow was considered. It is shown that, even under realistic traffic conditions, PTA achieves better results than the classic algorithm given Iub delay increments and decrements. The buffer occupancy figure of PTA is sometimes higher than the classic algorithm because

of the lower frame loss ratio of PTA, which implies that the Node B has to store frames that, with the classic algorithm, would have been lost.

4   Conclusions

In this paper, we propose a new timing adjustment algorithm for the *Transport Channel Synchronization* mechanism in 3G access networks. This algorithm, the *Proportional Tracking Algorithm* (PTA), has been analyzed from a control theory point of view and evaluated by computer simulation under realistic traffic conditions showing better performance than the classic algorithm in scenarios with Iub delay increments and decrements. The most important conclusions are that PTA has a faster response to abrupt delay changes than classic algorithm, reducing frame losses, and its operation is stable regardless of the total Iub delay.

In our opinion, QoS management as well as congestion control algorithms in UTRAN transport networks should be revisited with the perspective of a faster and more efficient synchronization procedure as the one proposed in this paper, especially with the gradual substitution of ATM by IP in the transport infrastructure giving support to the mobile access network.

References

1. 3GPP; Technical Specification Group RAN.: Synchronization in UTRAN Stage 2, 7.2.0. 3GPP TS 25.402 (2006)
2. 3GPP; Technical Specification Group RAN.: Iub/Iur Interface User Plane Protocol for DCH Data Streams, 7.3.0. 3GPP TS 25.427 (2006)
3. S. Abraham et al.: Effect of Timing Adjust Algorithms on Iub Link Capacity for Voice Traffic in W-CDMA Systems. In: Proc. IEEE VTC, vol. 1, pp. 311–315 (2002)
4. M. Sagfors et al.: Transmission Offset Adaptation in UTRAN. In: Proc. IEEE VTC, vol. 7, pp. 5240–5244 (2004)
5. K. Ogata.: Discrete Time Control Systems. Prentice Hall Inc. (1994).
6. OMNeT++ Discrete Event Simulation System, http://www.omnetpp.org
7. 3GPP; Technical Specification Group RAN.: IP Transport in UTRAN, 5.4.0. 3GPP TS 25.933 (2003)
8. Tian Bu, Mun Choon Chan, and Ram Ramjee.: Connectivity, Performance, and Resiliency of IP-Based CDMA Radio Access Networks. IEEE Trans. on Mobile Computing, vol. 5, no. 8 (2006)
9. 3GPP; Technical Specification Group RAN.: Delay Budget within the Access Stratum, 4.0.0, 3GPP TR 25.853 (2004)