Control and Cybernetics

vol. **45** (2016) No. 3

Testing of adaptive nonuniform sampling switch algorithm with real-time simulation-in-the-loop

by

Justyna Jańczak, Mirosław Kondratiuk, Ewa Pawłuszewicz

Bialystok University of Technology, Mechanical Faculty, Department of
Automatics and Robotics, Wiejska 45C, 15-351 Bialystok, Poland
j.janczak@doktoranci.pb.edu.pl, m.kondratiuk@pb.edu.pl,
e.pawluszewicz@pb.edu.pl

Abstract: The study is motivated by the considerations concerning implementation of nonuniform sampling regarding the real-life objects. It is shown that nonuniform adaptive sampling switch algorithm can ensure a reduction in the number of taken samples and does not exert negative influence on the quality of the system control. Experiment was performed with 24VDC permanent magnet motor with encoder as a plant. Signals from the encoder were used for rotational speed calculations. The results of experiment are included.

Keywords: nonuniform sampling, sampled-data control, adaptive nonuniform sampling

1. Introduction

Sampling theory has been intensively investigated since the 1960s, when, in particular, the Kalman filter demonstrated its possibilities. Over these years, sampling methods were divided, most generally, into two main modes – uniform and nonuniform sampling (see Ben-Romdhane et al., 2009; Davis et al., 2011). The difference between these main modes relies on determining the sampling time for each sample. Further, Additive Random Sampling (see Ben-Romdhane et al., 2009) and Jittered Random Sampling (see, as well, Ben-Romdhane et al., 2009) are the most common sampling schemes in nonuniform sampling research area.

In most cases of sampled-data control, sampling instants are assumed to be equally time spaced, where T is the basic sampling period, this being called uniform sampling. Since the maximum signal bandwith could be not a good measure of signal changes, in the case of uniform sampling, a part of taken samples may turn out to be redundant (see Bechir and Ridha, 2009). In order to eliminate this redundancy, nonuniform sampling can be applied. Nonuniform sampling can bring many benefits, namely, it can reduce the amount of measured data, decrease power consumption for computation, suppress aliasing effect (see Bilinkskis, 2007), etc. This leads to increased attention concerning practical applications of nonuniform sampling. For instance, measurements obtained from a lot of radar applications appear as nonuniform data. Various nonuniform sampling patterns are used to gain better performance, since the possible Nyquist frequency is much lower than required. Other examples of practical applications can be found, for example, in Eng (2007), Bechir and Ridha (2009), Murthy and Ahuja (2009). On the other hand, the main disadvantages of nonuniformly sampled control systems are complexity in implementation and difficulties in stability analysis.

The aim of this study is to investigate a real life object under *Adaptive Nonuniform Sampling Switch Algorithms* (ANSSAs) proposed in Czaczkowska et al. (2016), as a response to intensified investigations in this area. In order to identify the mathematical model of the object and to introduce a nonuniform sampling with respect to a real-life object, the Ethernet connection has been used. In comparison to Czaczkowska et al. (2016), the weight coefficient in the sampling time model was introduced. It was the effect of practical implementation of the algorithm developed and described in Czaczkowska et al. (2016).

The paper is organized as follows. In Section 2 the *Adaptive Nonuniform Sampling Switch Algorithms* (ANSSAs) are introduced. Section 3 contains the description, with technical details, of the investigated real-time control system. In Section 4 we present some results of tests conducted with the developed real-time system with ANSSA. The article is summarized in the conclusion section. Moreover, some future works are also mentioned.

2. The sampling framework

In the literature of the subject (see Ben-Romdhane et al., 2009; Feizi et al., 2010, 2011; Maalej et al., 2011), the sampling process is described as a process of establishing a discrete signal $y_k(t)$ on the basis of its continuous representation $y(t)$ in the following manner (see Ben-Romdhane et al., 2009; Maalej et al., 2011):

$$y_k\left(t\right) = \sum_{k=-\infty}^{\infty} y\left(kT\right)\delta\left(t-kT\right), \tag{1}$$

where k is the sample number and t_k is sampling time, T is the period of sampling, and $\delta(t)$ is the Dirac impulse. The time instant t_k is represented as $t_k = kT$ for uniform sampling and, of course, in general, $t_k \neq kT$ in case of nonuniform sampling.

A continuous-time object with discrete control signal is given by:

$$\left\{\begin{array}{c} \dot{x}(t) = Ax(t) + Bu(t) \\ u(t) = u(t_k),\ t \in (t_k, t_{k+1}) \\ y(t) = Cx(t) + Du(t) + Iz(t) \end{array}\right. , \tag{2}$$

where $x(t) \in R^n$ is a state vector, $u(t) \in R^m$ is a control input vector, $y(t) \in R^r$ is an output vector, $z(t) \in R^n$ is a disturbance vector, $A \in R^{nxn}$ is a stationary matrix, $B \in R^{nxm}$ is an input matrix, $C \in R^{rxn}$ is an output matrix, $D \in R^{rxm}$ is a feedforward matrix, $I \in R^{rxn}$ is an identity matrix, and k denotes a positive integer. The sampling pattern introduced into the control system (2) was developed in Czaczkowska et al. (2016).

The sample time instant t_i is defined by $|arctg(\beta \frac{dy}{dt})| \in \left(\frac{i\pi}{2}; \frac{(i+1)\pi}{2p}\right)$ for $i = 1,\ldots,k$-1, p is the number of chosen sampling times, which evaluates the speed of changes in the output signal y, β is weight coefficient introduced in order to properly scale the argument of arcus tangens function. In order to introduce the ANSSA, we define the sampling scheme in the following manner:

$$t_k = t_1 + \sum_{i=2}^{k} f\left(y\left(t_{i-1}\right)\right), \tag{3}$$

where

$$f_i\left(y\left(t_{i-1}\right)\right) = \left\{ \begin{array}{c} \tau_{i-1} fort \in \left(\frac{(i-2)\pi}{2p}; \frac{(i-1)\pi}{2p}\right] \\ \tau_i fort \in \left(\frac{(i-1)\pi}{2p}; \frac{i\pi}{2p}\right] \end{array} \right. . \tag{4}$$

The main idea of the proposed sampling time selection is symmetric division of *arctg* function value. For instance, for $p = 3$ (three sampling frequencies in ANSSA) algorithm uses three symmetric intervals between 0 and 2. Each interval is connected with other sampling frequency.

Equations (3) and (4) allow to implementation of two proposed adaptive nonuniform sampling switch algorithms (ANSSAs). The 1^{st} ANSSA shifts sampling interval straightaway after noticing explicit changes in output signal. Equation (4) describes rule of time intervals shifting. The 2^{nd} ANSSA is described by equations (3) and (4). The 2^{nd} ANSSA holds a sample to its end, while the 1^{st} ANSSA switches the sampling interval even if the signal value changes significantly in the arbitrary moment of current time interval. A detailed description of the proposed sampling algorithms can be found in Czaczkowska et al. (2016). In the present study the practical implementation of the 2^{nd} ANSSA for the DC motor rotational speed control system is described. In order to investigate such a system, the computer-based real-time model connected with the real world by Ethernet communication channel was built. Technical details of the developed system are presented in the next section.

3. Description of the real-time system

In order to practically investigate the developed 2^{nd} ANSSA, we have built a control system in the form, which is presented in Fig. 1.

As the plant, the 24VDC permanent magnet motor with encoder was used. In order to control its rotational speed, we applied simulation-in-the-loop (SIL)

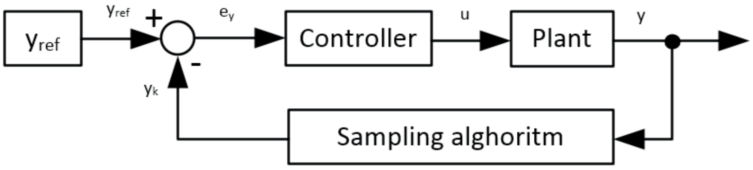

Figure 1. Scheme of the control system for testing ANSSA

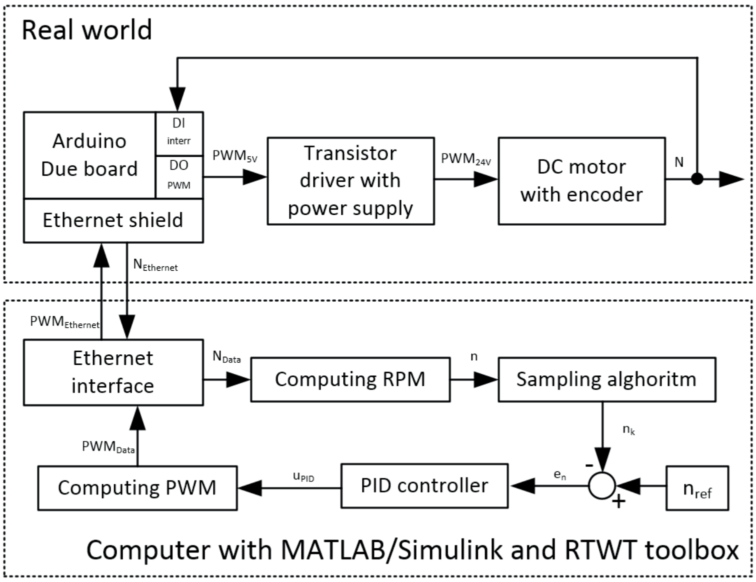

Figure 2. Scheme of the developed control system, where RPM means "rotations per minute"

technique. The corresponding scheme of the developed system is presented in Fig. 2.

In order to implement the real-time system it was necessary to create an efficient communication channel between PC and the motor driver. We use Arduino Due board based on 32-bit ARM core processor with 84MHz clock frequency, which is connected to the Ethernet Shield, this enabling us to apply communication with the User Datagram Protocole (UDP). The microcontroller has been receiving impulses from the motor encoder (N), counting them and transmitting their actual number via Ethernet Shield to PC ($N_{Ethernet}$). On the other hand, Arduino Due has been also receiving DC motor control signal from PC ($PWM_{Ethernet}$) and sending it via microcontroller PWM output (PWM_{5V}) to motor transistor driver. The driver was powered from laboratory power supply and delivered to the motor PWM power voltage signal of 24V (PWM_{24V}). The image of the developed system is presented in Fig. 3.

In order to carry out tests of nonuniformly sampled real-time control system, we employed MATLAB/Simulink Real-Time Windows Target toolbox. The system worked with 200 Hz frequency, and so the communication was performed at every 0.005s ($T_s = 0.005$s). Since Arduino has been sending only the number of pulses produced by the encoder, in PC environment the rotational speed of the motor would have to be computed. The encoder, which was incremental, had 825 gaps, and Arduino, in the interruption mode, increased the encoder accumulator for every rising and falling edge. Moreover, encoder was equipped with two optical gates, and so, one gap generated four pulses. This was advantageous, since in the case when microcontroller lost one pulse, the appearing angle error was divided by the factor of 4. Computation of rotational speed was done in the following way:

$$n\left(k\right) = \frac{0.25}{825\left[\frac{pulses}{rotation}\right]}\frac{\left(N\left(k\right) - N\left(k-20\right)\right)\left[pulses\right]}{20 \cdot T_s\left[s\right]}60\left[\frac{s}{min}\right], \tag{5}$$

where $n(k)$ is the rotational speed for the k-th sample, [rpm]; $N(k)$ is the number of pulses for the k-th sample, $N(k\text{-}20)$ is the number of pulses for 20 samples before the k-th sample, and T_s is the real-time sampling period, [s].

The encoder was delivering the number of pulses that was proportional to the angle of rotation of the motor. However, electronic imperfections caused that the microcontroller would sometimes lose a particular pulse, yielding thereby a small reading error. On the other hand, derivation of such readings introduced many more significant error pulses in the rotational speed. Introduction of the 20-sample delay in the discrete derivation process - equation (5) - enables us to reduce the amount of incorrect and distorted readings from the encoder. Moreover, in order to get a smoother signal, after derivation of rotation angle we applied low-pass filter with time constant of 5 T_s. We obtained the delay derivation and the filter constant during the tests in an experimental manner.

The 2^{nd} ANSSA require also second derivation of rotational angle. Rotational acceleration ($a(k)$) was computed thanks to the fallowing formula:

$$a\left(k\right) = \frac{\left(n\left(k\right) - n\left(k-20\right)\right)\left[\frac{rotation}{s}\right]}{20 \cdot T_s\left[s\right]}, \tag{6}$$

In the acceleration signal, a considerable 40-sample delay appeared. This was the only way to perform compensation for erroneous readings from the encoder. However, this signal was used only by ANSSA for switching the sampling frequencies, and so its delay did not have such a strong influence on the entire system behavior. The low-pass filter with the same time constant 5 T_s was also applied to the rotational acceleration signal.

In the SIL model we compared the measured motor speed n_k with reference speed n_{ref} both quantities having been expressed in units of rotations per minutes.

The control error e_n was delivered to the PID controller. The settings of the

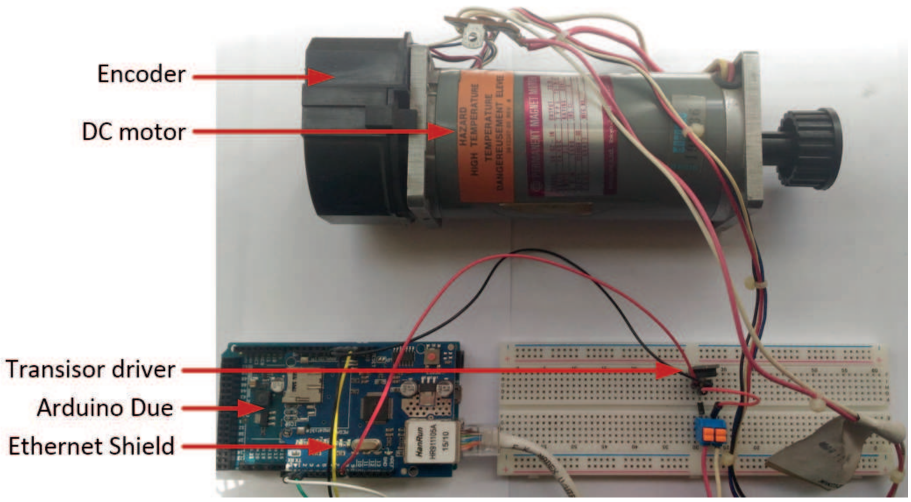

Figure 3. Picture of the developed control system

controller were obtained experimentally during tests and were the same for all simulations $K_P = 0.0007$, $K_I = 0.0015$, $K_D = 0$.

There is also an aspect, worth mentioning, concerning the signals transmitted through the UDP channel. The control signal u_{PID} was changing from 0 to 1. Arduino is equipped with the 10-bit PWM outputs, and so, in order to convert it to the form proper for Arduino board, we multiplied it by the factor of 255 and sent as one byte of binary data PWM_{Data}. It was previously mentioned that the encoder pulse number has been increasing its value when the rising or falling edges appeared on any of two Arduino Due interruption ports. Since the value of that accumulator was rapidly increasing, we were passing information about the pulse counter state using 4 bytes of data (32-bits) transmitted from Arduino to PC. Those data were decoded as the numbers of encoder pulses, written down as an unsigned integer number N_{Data}.

4. Tests

During the tests, we set the reference rotational speed $n_{ref} = 1000$rpm and compared the step responses of the three systems: the first – with 200Hz constant sampling (reference system with maximal real-time frequency), the second – with 25Hz constant sampling, and the third one – including the 2^{nd} ANSSA with three arbitrarily chosen sampling frequencies – 10Hz, 25Hz and 50Hz. In the 2^{nd} ANSSA, the frequencies were switched according to formula (3). The coefficient β, which was necessary to scale the argument of *arctg* function, was experimentally obtained and equaled 0.1.

The output and error signals of the developed real-time system were computed for 3 seconds ($t_{sim} = 3$s). Error signals of the particular sampling algorithms for the k-th sample were obtained according to the following equations:

$$e_{200Hz}(k) = n_{ref} - n_{200Hz}(k), \tag{7}$$

$$e_{25Hz}(k) = n_{ref} - n_{25Hz}(k), \tag{8}$$

$$e_{ANSSA}(k) = n_{ref} - n_{ANSSA}(k), \tag{9}$$

where: e_{200Hz} – control error of the reference system, e_{25Hz} – control error of the system with 25Hz constant sampling, e_{ANSSA} – control error of the system with ANSSA, n_{200Hz} – rotational speed of the reference system, n_{25Hz} – rotational speed of the system with 25_{Hz} constant sampling, n_{ANSSA} – rotational speed of the system with ANSSA. The results of simulations are presented in Fig. 4.

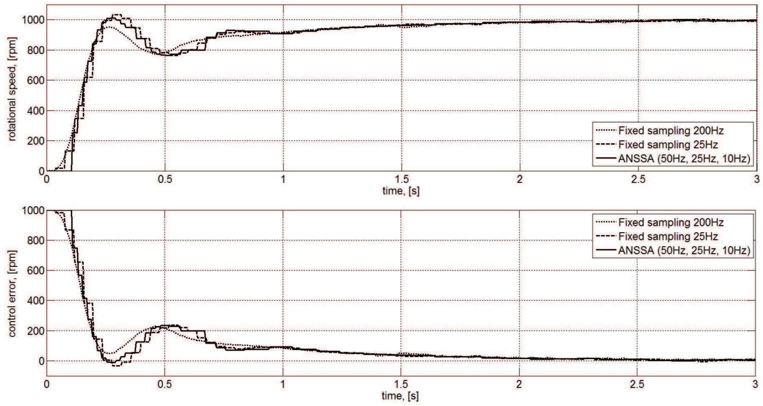

Figure 4. Output and error signals for the reference model (200Hz), model with 25Hz sampling and model with 2nd ANSSA

It was relatively difficult to make a reasonable comparison of signals, as expressed in time domain, and so we proposed to use the quality functions defined, respectively, as:

$$q_{200Hz} = \frac{\sum\limits_{k=1}^{t_{sim}/T_s} e_{200Hz}(k)}{\sum\limits_{k=1}^{t_{sim}/T_s} e_{200Hz}(k)} = 1, \tag{10}$$

$$q_{25Hz} = \frac{\sum\limits_{k=1}^{t_{sim}/T_s} e_{25Hz}(k)}{\sum\limits_{k=1}^{t_{sim}/T_s} e_{200Hz}(k)} < 1, \tag{11}$$

$$q_{ANSSA} = \frac{\sum\limits_{k=1}^{t_{sim}/T_s} e_{ANSSA}(k)}{\sum\limits_{k=1}^{t_{sim}/T_s} e_{200Hz}(k)} < 1, \tag{12}$$

The values of these quality functions represent similarity of the systems with 25Hz constant sampling and with the 2^{nd} ANSSA to the reference system with 0.005 s constant sample time (200Hz sampling). The value of the quality function q_{200Hz} is always equal to 1, because it is our reference signal sampled with maximal frequency. After 3-second simulations we obtained $q_{25Hz} = 0.9119$ and $q_{ANSSA} = 0.8833$. Lower value of q_{ANSSA} than of q_{25Hz} implies that ANSSA gives relatively worse results than the approach with fixed step. However, when we compare the number of output signal samples used in feedback loop, a considerable advantage of ANSSA is revealed. The reference system generated 600 samples, the system with constant 25Hz frequency required 75 samples, and ANSSA generated only 47 samples. These significant differences in sample numbers could even increase when the tested system would control the rotational speed of the motor for a longer time. When the output signal is stabilized on the reference level, then its derivative is close to zero. For very small derivative value, ANSSA switches the sampling frequency to the highest value, what generates much less samples.

Additionally, we compared the reference system (200Hz), the system with ANSSA (10Hz, 25Hz, 50Hz) to the fixed-sampled systems with 15Hz and 16Hz, which gives 45 and 48 samples respectively. The results are presented in Fig. 5.

In Fig. 5, there are no significant differences between 15Hz and 16Hz sampling. However, when we compared quality functions, after the 3-second simulations, we obtained $q_{15Hz} = 0.8357$ and $q_{16Hz} = 0.8405$. These values are smaller than $q_{ANSSA} = 0.8833$, what implies that for a similar number of samples ANSSA gives better control quality than constant sampling. The values of quality functions for 15Hz and 16Hz sampling were calculated similarly as for 25Hz sampling according to equation (11). Moreover, additional tests enable us to establish that for the system with fixed sampling, the number of samples, which gives a similar value of the control quality function as ANSSA (10Hz, 25Hz, 50Hz) is approximately equal to 66. This corresponds to fixed sampling with frequency of 22Hz.

5. Conclusions

In this paper, a sampled-data control system with a real-life plant was tested. We employed MATLAB/Simulink software with Real-Time Windows Target toolbox. Moreover, we created the communication channel between the PC and the microcontroller Arduino Due board. This allowed us for controlling the rotational speed of the DC motor and for testing nonuniform sampling algorithms with a physical object and a virtual controller. Further, the responses of the closed-loop system for both nonuniform sampling algorithms and uniform

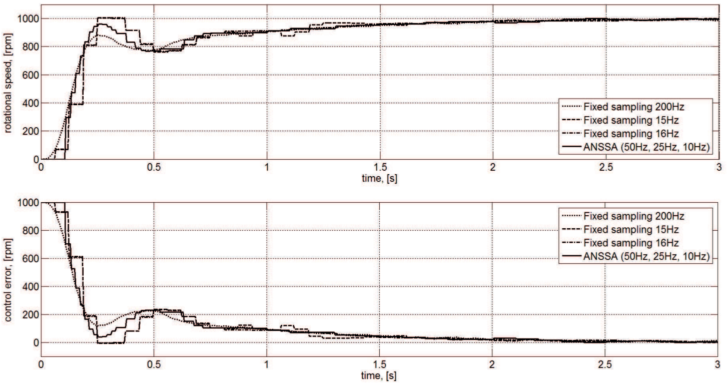

Figure 5. Output and error signals for reference model (200Hz), model with 15Hz sampling and model with 2nd ANSSA

sampling algorithm were compared. The tests have shown that nonuniform sampling algorithm, with available frequencies 10Hz, 25Hz, 50Hz, generates lower numbers of samples than the uniform sampling with 25Hz frequency. On the other hand, the system with uniform sampling provides a slightly better control quality than the system with ANSSA. This is connected with the fact that the sampling intervals designed persist even up to the instant, when signal starts to change values rapidly. In order to more precisely compare fixed sampling and the proposed ANSSA, additional tests were conducted. We compared quality function values for the systems which gave similar number of samples as ANSSA. Moreover, we obtained fixed sampling frequency, for which the investigated system had the similar quality function value as ANSSA with 10Hz, 25Hz and 50Hz.

The investigations, carried out, showed that the developed ANSSA can be used with control systems characterized by much shorter transition process than steady states. The examples can be constituted by systems of valve position control, switching power relays, speed control in industrial transport lines, holding fluid level or fluid pressure in tanks, etc. In that kind of systems, introducing nonuniform sampling allows for decreasing the required computational power and data processing. Moreover, since the control system with ANSSA uses much less data when plant is close to steady state, there is also a possibility of optimizing the controller power consumption and save energy, necessary in control processes.

Acknowledgment

The research work of Justyna Jańczak has been performed in the framework of project No. MB/WM/3/2016 and financed from the funds for science by the Polish Ministry of Science and Higher Education.

The research of Mirosław Kondratiuk and Ewa Pawłuszewicz has been carried out in the framework of project No. S/WM/1/2016 and financed from the funds for science by the Polish Ministry of Science and Higher Education.

References

Albertos, P. and Crespo, A. (1999) Real-time control of non-uniformly sampled systems. *Control Engineering Practice* **7**(4), 445-458.

Bechir, D. M. and Ridha, B. (2009) Non-uniform sampling schemes for RF bandpass sampling receiver. In: *Proceedings of the 2009 International Conference on Signal Processing Systems ICSPS'2009*, Singapore, 15-17 May, 2009, 13-17.

Ben-Romdhane, M., Rebai, C., Ghazel, A., Desgeys, P. and Loumeau, P. (2006) Non-uniform sampling schemes for IF sampling radio receiver. In: *Proceedings of the International Conference on Design and Test of Integrated Systems in Nanoscale Technology*, Tunis, Tunisia, 5-7 September, 2006, 15-20.

Czaczkowska, J., Kondratiuk, M. and Pawłuszewicz, E. (2016) Control system with adaptive nonuniform sampling switch algorithms. In: *Proceedings of the 17th International Carpathian Control Conference*, Slovakia, 29 May – 1 June, 2016, 128–133.

Davis, J. M., Gravagne, I. A. and Ramos, A. A. (2010) Stability of switched linear systems on non-uniform time domains. In: *Proceedings of the 42nd South Eastern Symposium on System Theory (SSST)*, 7-9 March 2010, Texas, USA, 127-132.

Ding, F., Chen, T. and Iwai, Z. (2007) Adaptive digital control of Hammerstein nonlinear systems with limited output sampling. *SIAM Journal on Control and Optimization* **45**(6), 2257-2276.

Eng, F. (2007) *Non-uniform sampling in statistical signal processing.* PhD Thesis. Linköpings University, Sweden.

Feizi, S., Goyal, V. and Medard, M. (2010) Locally adaptive sampling. In: *Proceedings of the 48th Annual Allerton Conference on Communication, Control and Computing*, 29 September - 1 October, 2010, 152-159.

Feizi, S., Angelopoulos, G., Goyal, V. and Medard, M. (2011) Energy-efficient time-stampless adaptive nonuniform sampling. In: *Proceedings of the 10th IEEE Conference on Sensors Location*, 28-31 October, 2011, Limerick, Ireland, 912-915.

Guo, G. (2005) System with nonequidistant sampling. Controllable? Observable? Stable? *Asian Journal of Control* **7**, 455-461.

Khan, S., Goodall, R. M. and Dixon, R. (2010) Design and analysis of non-uniform rate digital controllers. In: *Proceedings of the UKACC International Conference on Control*, 7-10 September, 2010, United Kingdom, 1-6.

Luo, C. (2012) *Non-uniform sampling algorithms and architectures.* PhD thesis. Georgia Institute of Technology, December.

MAALEJ, A., BEN-ROMDHANE, M., REBAI, C., DESGREYS, P., LOUMEAU, P. and GHAZEL, A. (2011) Non uniform sampling for power consumption reduction in SDR receiver baseband stage. In: *Proceedings of the 30th General Assembly and Scientific Symposium URSI*, 13-20 August, 2011, Turkey, DOI:10.1109/URSIGASS.2011.6050507.

MURTHY, G. R. and AHUJA, N. (2009) Non-uniform sampling: a novel approach. In: *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 19-24 April, 2009, USA, 3229-3232.