XVI IMEKO World Congress
Measurement - Supports Science - Improves Technology - Protects Environment ... and Provides Employment - Now and in the Future
Vienna, AUSTRIA, 2000, September 25-28

ON STABILITY OF MULTIPLE-FEEDBACK DELTA-SIGMA MODULATORS

K. Olejarczyk
Institute of Electronic Systems
Warsaw University of Technology, Poland

Abstract: In this paper a new method of proving stability of Delta-Sigma modulators in the way of bounding outputs of integrators has been described. The presented approach is theoretical as well as computational. The method has been tested exhaustively for second-order multiple-feedback modulators.

Keywords: Sigma-Delta, nonlinear dynamics, chaos.

1 INTRODUCTION

Although ΔΣ modulators have been successfully designed and manufactured for many years, there is a lack of solid theoretical foundation. Methods of design are based on empirical knowledge obtained mainly in the way of great amount of simulation. Stability, which is the basic parameter of modulators, can be obtained only by simulation in higher-order case. Only some ΔΣ modulator topologies have been analysed exactly [1]. First-order modulator has been described in many works, for example in [2]. Standard second-order modulators are the subject of theoretical work of Wang [3] as well as Hein and Zakhor [4]. The last work yields the method of determining convex trapping region with parabolic edges in the state space [4]. A computational approach to determining convex trapping region is presented by Schreier *et al.* [5]. This method produces tighter bounds then those of Hein and Zakhor [4]. The trapping region is a convex polygon (in the 2-D case; in 3-D case the region is a convex polyhedron). The advantage of this approach is that the method can be applied to higher-order modulators. The fully theoretical approach of Farrell and Feely [6] takes account of the jagged form of the outer boundary of the trapping region (which is no longer constrained to be convex). This method yields better results for standard second-order modulators then those of Schreier *et al.* [5].

The presented method is based on the similar idea as the approach of Farrell and Feely [6]. The method of Farrell and Feely is dedicated to standard second-order ΔΣ modulators (without any local feedback loops), and there was a need to find out more general method that can be applied to wider class of modulators. In this paper the new method will be described with an example of multiple-feedback second-order ΔΣ modulator (with single-bit quantiser) shown in Figure 1. The presented approach can be applied to higher-order modulators also.

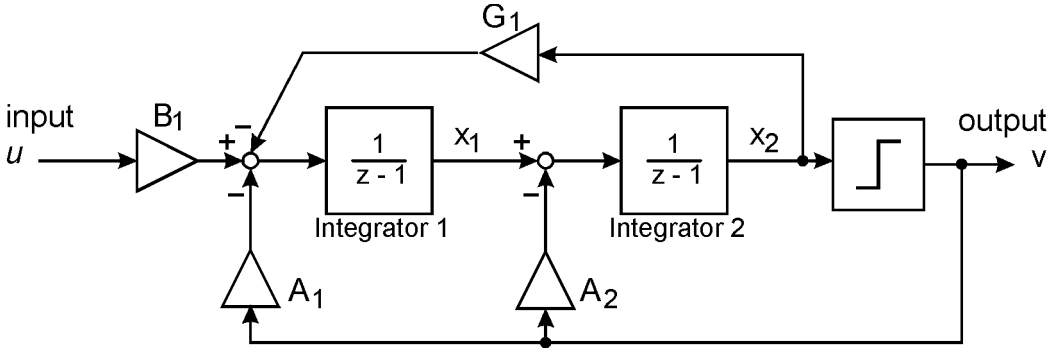

Figure 1. Multiple-feedback second-order ΔΣ modulator with single-bit quantiser.

2 EXAMPLE OF MULTIPLE-FEEDBACK DS MODULATOR

Consider the signal at the quantiser output. This signal is a sequence of +1s and –1s. (There is a convention that the quantiser outputs are normalised). Let all consecutive +1s and –1s be grouped and denoted as N^+ and N^- (these symbols will be called *numbers of positive and negative iterations*, respectively). For instance: (–1, +1, +1, +1, +1, +1, +1, +1) = (1^-, 7^+). This notation means also, that the modulator had to have at least one positive iteration before the sequence (1^-, 7^+), and there is at least one negative iteration after it.

It is assumed that the input signal u is constant. Let the input signal be equal u = 0.696, and coefficients of the modulator model (that can be seen in Figure 1) equal as follows: A1=0.215, A2=0.559, B1=0.215, G1=320356/1603602025≈0,00019977. The next assumption is that the modulator has performed following sequence of iterations: $(1^-, 7^+)$. In fact, the sequence $(1^-, 7^+)$, was obtained as a result of simulation which started with arbitrary initial conditions.

The modulator model can be described with following difference equation (given in matrix-vector notation):

$$\mathbf{x}(n+1) = f(\mathbf{x}(n)) = \begin{bmatrix} 1 & -G_1 \\ 1 & 1-G_1 \end{bmatrix} \cdot \mathbf{x}(n) + \begin{bmatrix} B_1 \cdot u \\ B_1 \cdot u \end{bmatrix} + \begin{bmatrix} -A_1 \\ -A_1 - A_2 \end{bmatrix} \cdot \operatorname{sgn}(x_2(n)), \quad (1)$$

where $\operatorname{sgn}(\cdot) = \begin{cases} 1 & \text{for argument } \geq 0 \\ -1 & \text{for argument } < 0 \end{cases}$, and $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$.

Let Γ_P and Γ_N be the positive and negative half-planes: $\Gamma_P = \{\mathbf{x}: x_2 \geq 0\}$, $\Gamma_N = \{\mathbf{x}: x_2 < 0\}$. It can be seen that the mapping f is piecewise-affine, and there exist such two affine mappings:

$f_P: \mathrm{R}^2 \to \mathrm{R}^2$ and $f_N: \mathrm{R}^2 \to \mathrm{R}^2$, that $f(\mathbf{x}) = \begin{cases} f_P(\mathbf{x}) & \text{for } \mathbf{x} \in \Gamma_P \\ f_N(\mathbf{x}) & \text{for } \mathbf{x} \in \Gamma_N \end{cases}$. (The symbol R denotes real numbers set).

After the modulator has performed the sequence $(1^-, 7^+)$ its state has to enter the quadrangle that is shown in Figure 2 C (the darkest region in the middle of the figure). It will be explained in several steps. Let define $\alpha=\{\mathbf{x}: x_2=0\}$, *i.e.* α is the horizontal axis.

Before the sequence $(1^-, 7^+)$, the modulator state had to be inside the positive half-plane. After the negative iteration (the first iteration of the sequence), the trajectory must enter the sector $\Gamma_N \cap f_P(\Gamma_P)$, shown in Figure 2 A (as the darkest region). Next, after the first positive iteration the modulator state enters the set $\Gamma_P \cap f_N(\Gamma_N) \cap f_N \circ f_P(\Gamma_P)$, that is shown in Figure 2 B (as a triangle and the darkest region). Finally, when the trajectory has performed the sequence $(1^-, 7^+)$ and additionally one negative iteration, the modulator state has to enter the set $\eta_1 = \Gamma_N \cap f_P(\Gamma_P) \cap f_P^7 \circ f_N(\Gamma_N) \cap f_P^7 \circ f_N \circ f_P(\Gamma_P)$, that can be seen in Figure 2 C (actually, this set is a quadrangle). There is following notation used: $f^n = \underbrace{f \circ \ldots \circ f}_{\text{n times}}$, *i.e.* the n-fold composition of f with itself. It is obvious that if a trajectory with an arbitrary initial conditions contains the sequence $(1^-, 7^+)$, then the set η_1 is not empty. In this case it is true, due to the fact, that the sequence was obtained as a result of simulation. (In order to avoid rounding errors, rational arithmetic was employed.) The set η_1 obtained as an intersection of a several half-planes is convex. The way of determining regions generated by sequences of the form (N_1^-, N_2^+) will be formulated for the general case in the next section.

Consider the inverse implication: what sequences of the form (N_1^-, N_2^+) may be generated, if trajectories start from the set η_1? It is easier to answer this question, if the set η_1 is close. The closure of η_1 will be denoted $\overline{\eta}_1$. Obviously, $\eta_1 \subset \overline{\eta}_1$. All the possible sequences of iterations can be found in the way of simulation. The polygon $\overline{\eta}_1$ is iterated with the mapping f. Because $\overline{\eta}_1$ lays in the negative half-plane, the whole polygon $\overline{\eta}_1$ will be transformed by affine mapping f_N (Figure 3 A). Affine mapping preserves structure of a set (straight lines maps to straight lines and the interior of a polygon maps to the interior of the polygon's image), so it is sufficient to map only the tops, in order to obtain the image of the polygon. During the iteration of the mapping, a polygon may cross the splitting line (*i.e.* α). In such a case the polygon is cut into pieces and the pieces are mapped further individually (Figure 3 A, B and C). In this way new tops appear, *i.e.* the split points (points where the line α intersect polygon edges). The iteration finishes when every pieces of the polygon $\overline{\eta}_1$ have crossed the splitting line α twice (because two-element sequences of iterations are considered).

An algorithm that derives automatically implications like this described above has been originally developed by the author. The algorithm is implemented in C++ language and uses rational arithmetic. For the case mentioned above following sequences have been obtained: $(1^-, 4^+)$, $(1^-, 5^+)$ and $(1^-, 6^+)$. It means, that after the sequence of iterations $(1^-, 7^+)$, one of these three above-mentioned sequences must follow. It doesn't mean however that *all* sequences obtained as a result

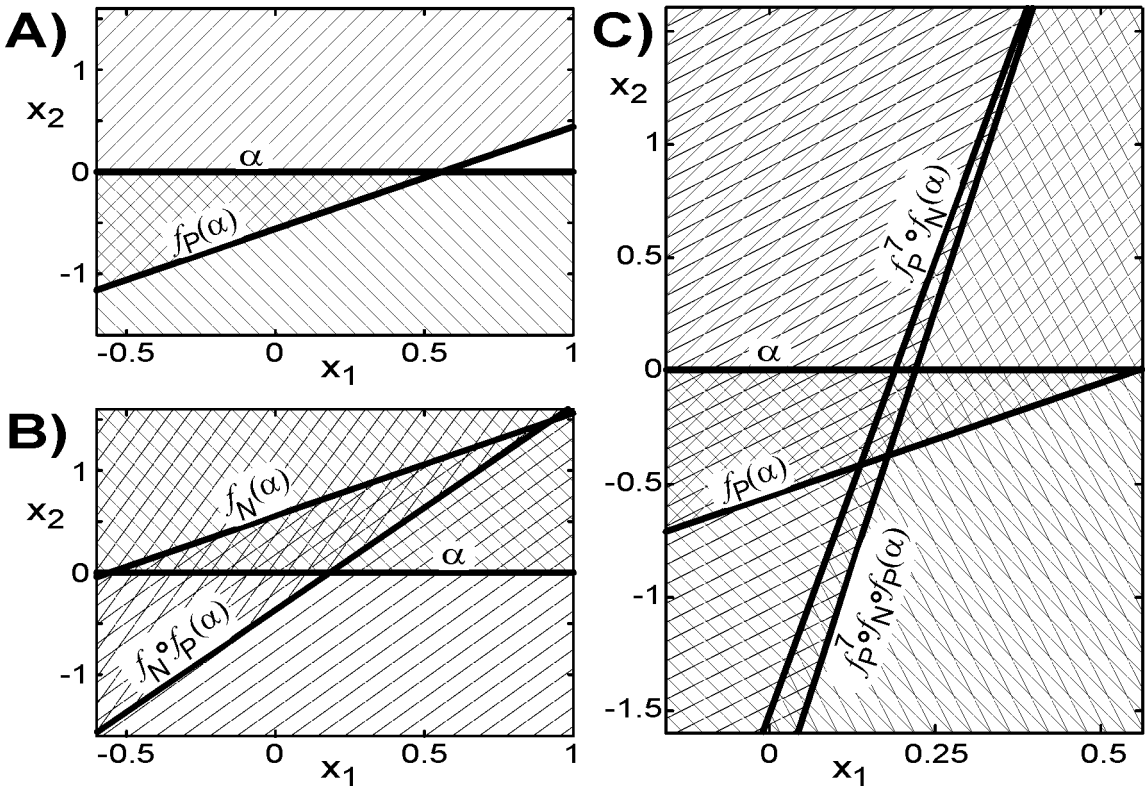

Figure 2. Determining a region in state space corresponding to the sequence of iterations (1^-, $7^{+\epsilon}$). A, B, C) - consecutive steps.

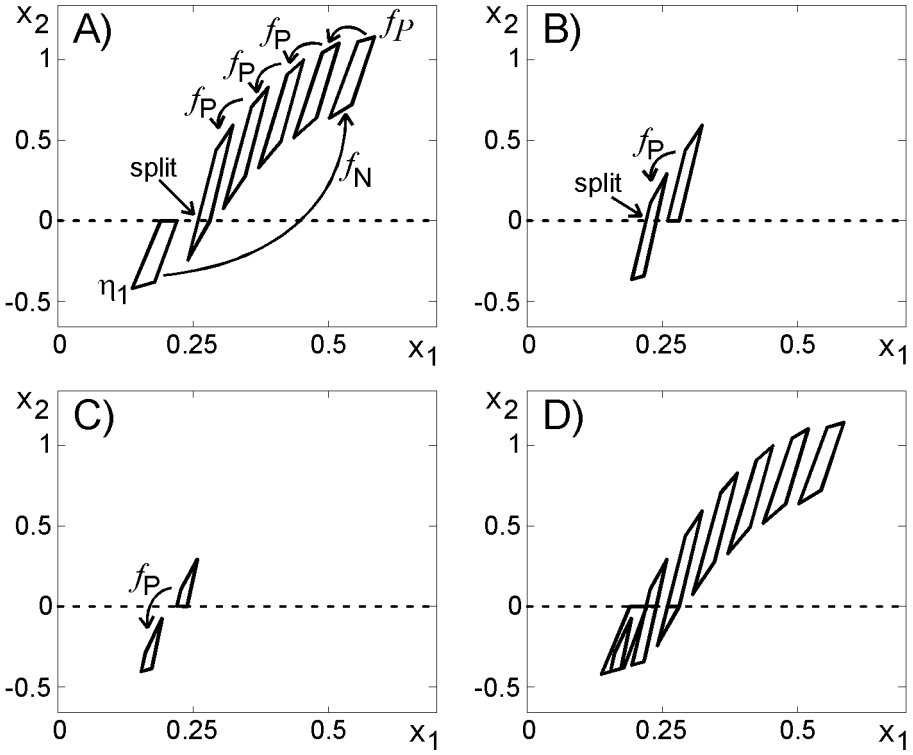

Figure 3. Polygons obtained during the process of determining all possible sequences of iterations which correspond to the polygon η_1. A) The polygon η_1 and its images before the first split. Obtained result: (1^-, 4^+). B) The new polygon and its image. Obtained result: (1^-, 5^+). C) The second new polygon and its image. Obtained result: (1^-, 6^+). D) All polygons obtained during the process.

of concatenation *i.e.* $(1^-, 7^+, 1^-, 4^+)$, $(1^-, 7^+, 1^-, 5^+)$ and $(1^-, 7^+, 1^-, 6^+)$ must be acceptable, it means only that a sequence which follows the sequence $(1^-, 7^+)$ has to be a member of the set $\{(1^-, 4^+), (1^-, 5^+), (1^-, 6^+)\}$. Such a relationship will be called *sequence relationship*, and will be denoted symbolically as follows: $(1^-, 7^+) \rightarrow \{(1^-, 4^+), (1^-, 5^+), (1^-, 6^+)\}$.

It would be useful to find a set of sequences which may follow each new (just obtained) sequence, *i.e.* $(1^-, 4^+)$, $(1^-, 5^+)$ and $(1^-, 6^+)$. For the case mentioned above following sequence relationships have been obtained: $(1^-, 7^+) \rightarrow \{(1^-, 6^+), (1^-, 4^+), (1^-, 5^+)\}$, $(1^-, 6^+) \rightarrow \{(1^-, 6^+), (1^-, 4^+), (1^-, 5^+), (1^-, 7^+)\}$, $(1^-, 4^+) \rightarrow \{(1^-, 6^+), (1^-, 4^+), (1^-, 5^+), (1^-, 7^+), (1^-, 3^+)\}$, $(1^-, 5^+) \rightarrow \{(1^-, 6^+), (1^-, 4^+), (1^-, 5^+), (1^-, 7^+)\}$, $(1^-, 3^+) \rightarrow \{(1^-, 6^+), (1^-, 4^+), (1^-, 5^+), (1^-, 7^+), (1^-, 3^+), (1^-, 8^+)\}$, $(1^-, 8^+) \rightarrow \{(1^-, 6^+), (1^-, 4^+), (1^-, 5^+)\}$.

Let define $\Omega = \{(1^-, 3^+), (1^-, 4^+), (1^-, 5^+), (1^-, 6^+), (1^-, 7^+), (1^-, 8^+)\}$. Consider sequence of iterations $(N_1^-, N_2^+, N_3^-, N_4^+)$. It can be seen that if $(N_1^-, N_2^+) \in \Omega$, than $(N_3^-, N_4^+) \in \Omega$. It follows that the set Ω is a trapping set of sequences. If Ω is finite, and its elements contain finite numbers of iterations, then the modulator is stable (for the given value of the constant input signal). It follows that stability of the modulator is proven for the above-mentioned example.

3 GENERAL FORMULA OF THE ALGORITHM

In this section general formula of the relationship between sequence of numbers of iterations and a region in state space will be given. Consider a D-order modulator, $D \geq 2$, which can be described with a mapping f: $R^D \rightarrow R^D$. Let define Γ_P and Γ_N as follows: $\Gamma_P = \{\mathbf{x}: x_D \geq 0\}$, $\Gamma_N = \{\mathbf{x}: x_D < 0\}$. There is an assumption that f is piecewise-affine, so that it can be expressed as:

$$f(\mathbf{x}) = \begin{cases} f_P(\mathbf{x}) & \text{for } \mathbf{x} \in \Gamma_P \\ f_N(\mathbf{x}) & \text{for } \mathbf{x} \in \Gamma_N \end{cases} \tag{2}$$

where the mappings f_P and f_N are affine. It is assumed that the trajectory of the modulator includes a M-element sequence of iterations $(\xi_1, \xi_2, \xi_3, \ldots \xi_M)$, where M is even, and ξ_k are positive integers, k=1, … M. In the above notation of sequence of iterations super-script-signs have been omitted. Let S be the initial state of the quantiser, $S \in \{-1, +1\}$. Without loss of generality it can be assumed that the sequence starts in zero time instant. The quantiser outputs form the following sequence: $v_0, v_1, \ldots, v_{\xi_1+\xi_1+\ldots+\xi_M-1} = \underbrace{S,\ldots S}_{\xi_1}, \underbrace{-S,\ldots -S}_{\xi_2}, \underbrace{S,\ldots S}_{\xi_3}, \ldots, \underbrace{-S,\ldots -S}_{\xi_M}$. Let define a set η in state space, in the following way:

$$\mu_1 = \begin{cases} f_N(\Gamma_N) \cap f_N^{\xi_1} \circ f_P(\Gamma_P) & \text{for } S = +1 \\ f_P(\Gamma_P) \cap f_P^{\xi_1} \circ f_N(\Gamma_N) & \text{for } S = -1 \end{cases} \tag{3}$$

$$\mu_{i+1} = \begin{cases} f_P^{\xi_{i+1}}(\mu_i) \cap f_P(\Gamma_P) & \text{if } (S = +1 \wedge i\text{-odd}) \vee (S = -1 \wedge i\text{-even}) \\ f_N^{\xi_{i+1}}(\mu_i) \cap f_N(\Gamma_N) & \text{if } (S = +1 \wedge i\text{-even}) \vee (S = -1 \wedge i\text{-odd}) \end{cases} \tag{4}$$

i = 1, … M–1.

$$\eta = \begin{cases} \mu_M \cap \Gamma_P & \text{if } (S = +1 \wedge i\text{-odd}) \vee (S = -1 \wedge i\text{-even}) \\ \mu_M \cap \Gamma_N & \text{if } (S = +1 \wedge i\text{-even}) \vee (S = -1 \wedge i\text{-odd}) \end{cases} \tag{5}$$

It can be seen that the modulator state after the sequence of iterations, $\mathbf{x}(\xi_1 + \xi_2 + \ldots + \xi_M)$, belongs to the set η. In order to apply the method of determining a region in state space corresponding to a sequence of iterations, a closed-form expressions for f_P^n (the n-fold composition of f_P with itself) and f_N^n, for any $n \geq 0$, are needed. For multiple-feedback higher-order modulators (with single quantiser) such expressions can be obtained in the way described in [7]. For example following formulas correspond to the difference equation (1) and Figure 1:

$$f_{P,N}^n(\mathbf{x}) = \mathbf{M} \cdot \left(\begin{bmatrix} \cos(n \cdot \alpha) & -\sin(n \cdot \alpha) \\ \sin(n \cdot \alpha) & \cos(n \cdot \alpha) \end{bmatrix} \cdot \left(\mathbf{M}^{-1} \cdot \mathbf{x} + \mathbf{R}_{P,N} \right) - \mathbf{R}_{P,N} \right) \tag{6}$$

where $\alpha = a\sin\left(-\frac{1}{2} \cdot \sqrt{G_1 \cdot (4 - G_1)}\right)$, $\mathbf{M} = \begin{bmatrix} \frac{\sqrt{2}}{2} \cdot G_1 & \frac{\sqrt{2}}{2} \cdot \sqrt{G_1 \cdot (4 - G_1)} \\ \sqrt{2} & 0 \end{bmatrix}$,

$$\mathbf{R}_P = \begin{bmatrix} -\frac{\sqrt{2}}{2} \cdot \frac{B_1 \cdot u - A_1}{G_1} \\ \frac{\sqrt{2}}{2} \cdot \frac{B_1 \cdot u - A_1 - 2 \cdot A_2}{\sqrt{G_1 \cdot (4 - G_1)}} \end{bmatrix}, \text{ and } \mathbf{R}_N = \begin{bmatrix} -\frac{\sqrt{2}}{2} \cdot \frac{B_1 \cdot u + A_1}{G_1} \\ \frac{\sqrt{2}}{2} \cdot \frac{B_1 \cdot u + A_1 + 2 \cdot A_2}{\sqrt{G_1 \cdot (4 - G_1)}} \end{bmatrix}.$$

Notation $\mathbf{M}^{-1}$ means the inverse matrix of **M**. It can be seen that the trajectory of the modulator corresponds to a circle movement in certain basis. Examples of formulas corresponding to other topologies can be found in [7].

In the general case determining all possible sequences of iterations which correspond to a convex polytope (the D-dimensional analogue of 2-dimensional polygon and 3-dimensional polyhedron) is very similar to the exemplary 2-dimensional case described above: it is sufficient to calculate images of the tops to obtain an image of the polytope. In D-dimensional state space the splitting line α (defined above for 2-dimensional case) has to be redefined as a hyperplane: $\alpha=\{\mathbf{x}: x_D=0\}$. When a D-dimensional convex polytope crosses the hyperplane α, many new tops may appear (they would be points where the hyperplane α intersect the polytope edges). Obviously. calculating these split points is much more difficult in the higher-order case.

4 RESULTS OF TESTS

The presented algorithm was tested for different values of the length of sequence of iterations M, and for different values of constant input signal *u*. A sum of all polygons obtained during the process of derivation sequence relationships forms a positively invariant set (PIS) in state space. The computer program that realises the above-mentioned algorithm calculates minimal and maximal values of co-ordinates x_1 and x_2 for each positively invariant set. It enables an analysis of the bounds of the PIS versus the input signal value *u*. The results of the computations are shown in Figure 4.

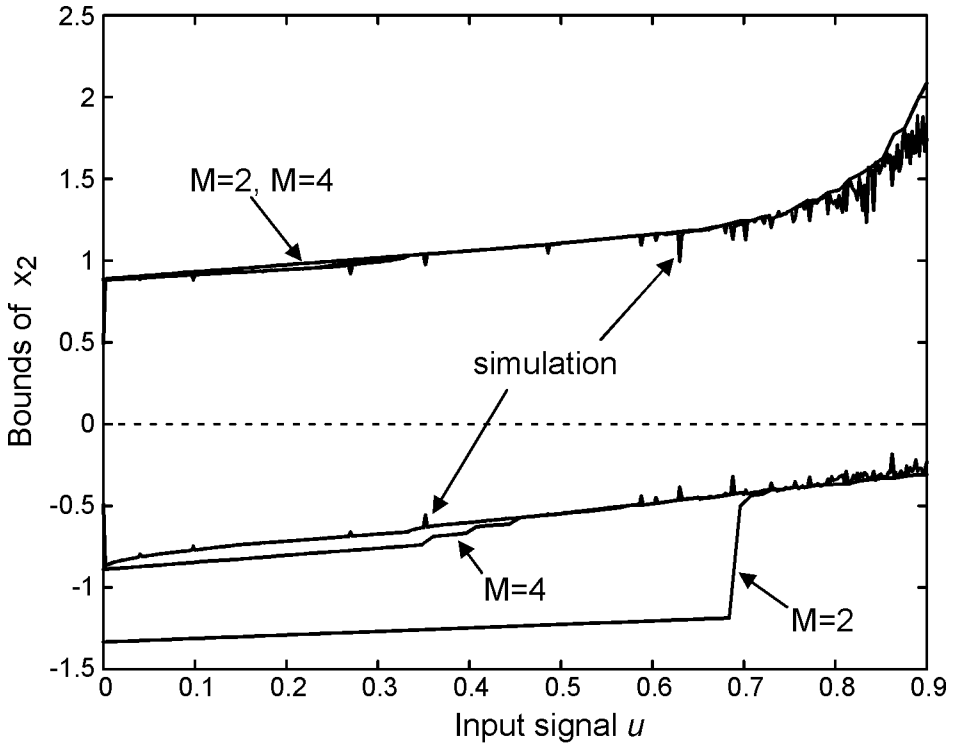

Figure 4. Bounds (the maximal and minimal values) of the signal x_2. The outer bounds have been obtained by the algorithm for M=2, the tighter bounds have been calculated for M=4. Bounds obtained as a result of long simulations are denoted as "S" and are tightest.

For parameter M equal two the algorithm of determining PIS has been run 132 times for different values of *u* ranging from 0 to 0.9. For M=4 the algorithm has been run for 76 constant input values equally spaced between 0 and 0.9. The simulation has been performed 451 times with input values

equally spaced between 0 and 0.9 also. Each simulation contained 1179648 time steps, and initial 65536 time steps were discarded.

5 CONCLUSIONS

The new method of determining positively invariant set in state space has been developed. The method enables calculation of bounds of integrators outputs and leads to proving stability of modulator for certain value of constant input signal. The method has been implemented as a computer program and verified. The advantage of the algorithm is that it can be applied to multiple-feedback higher-order $\Delta\Sigma$ modulators and the bounds obtained for M=4 are tight. The disadvantage of the method is that the algorithm is time-consuming especially for values of input signal close to +1. Tests for greater values of parameter M has been made also. The result is that the bounds of signals x_1 and x_2 obtained for M $\geq$ 6 are almost the same as those obtained for M=4 (but the speed of calculations falls down).

REFERENCES

[1] S. Norsworthy, R. Schreier, G. Temes, *Delta-Sigma Data Converters*, IEEE Press, 1997.
[2] O. Feely, L. Chua, Nonlinear dynamics of a class of analogue-to-digital converters, *International Journal of Bifurcations and Chaos*, **2** (2) (1992) 325-340.
[3] H. Wang, A geometric view of $\Sigma\Delta$ modulators, *IEEE Transactions on Circuits and Systems II*, **39** (6) (1992) 402-405.
[4] S. Hein, A. Zakhor, On the stability of Sigma Delta modulators, *IEEE Transactions on Signal Processing*, **41**, (7) (1993) 2322-2348.
[5] R. Schreier, M. Goodson, B. Zhang, An algorithm for computing convex positively invariant sets for Delta-Sigma modulators, *IEEE Transactions on Circuits Systems I*, **44** (1) (1997) 38-44.
[6] R. Farrell, O. Feely, Bounding the integrator outputs of second-order Sigma-Delta modulators, *IEEE Transactions on Circuits and Systems II*, **45** (6) (1998) 691-702.
[7] K. Olejarczyk, A geometric view of multiple feedback $\Delta\Sigma$ modulator, in *Proceedings of Third International Conference on Advanced A/D and D/A Conversion Techniques and their Applications*, (Glasgow, 26-28. July 1999), Glasgow, Great Britain, 1999, pp. 66-69.

AUTHOR: M. Sc. Krzysztof OLEJARCZYK, Institute of Electronic Systems, Department of Electronics and Information Technology, Warsaw University of Technology, ul. Nowowiejska 15/19, 00-665 Warsaw, Poland, Phone Int +48 22 6607634, Fax Int +48 22 252300, E-mail: kpole@ise.pw.edu.pl