Symbolic Dynamics of Digital Signal Processing Systems

Dr. Bingo Wing-Kuen Ling

School of Engineering, University of Lincoln.

Brayford Pool, Lincoln, Lincolnshire, LN6 7TS, United Kingdom.

Email: wling@lincoln.ac.uk

My research interests

- Symbolic dynamics
  - Limit cycle, fractal and chaotic behaviors of digital filters with two's complement arithmetic, sigma delta modulators and perceptron training algorithms
- Optimization theory
  - Semi-infinite programming, nonconvex optimization and nonsmooth optimization with applications to filter, filter bank, wavelet kernel and pulse designs
- Time frequency analysis
  - Nonuniform filter banks, filter banks with block samplers to multimedia and biomedical signal processing
- Control theory
  - Impulsive control, optimal control, fuzzy control and chaos control for HIV model and avian influenza model

Acknowledgements

- Prof. Xinghuo Yu (Royal Melbourne of Institute of Technology)
- Dr. Joshua D. Reiss (Queen Mary University of London)
- Dr. Herbert Ho-Ching Iu (University of Western Australia)
- Dr. Hak-Keung Lam (King’s College London)

Outline

- Introduction
- Digital filters with two’s complement arithmetic
- Sigma delta modulators
- Perceptron training algorithms
- Conclusions
- Q&A Session

Introduction

- Definition
  - Symbolic dynamics is a kind of system dynamics which involves multi-level signals.
- Motivations
  - Many practical signal processing systems, such as digital filters with two's complement arithmetic, sigma delta modulators and perceptron training algorithms, are symbolic dynamical systems. They are found in almost everywhere.

Introduction

- Challenges
  - Symbolic dynamical systems could lead to chaotic, fractals and divergent behaviors.
  - Stability conditions are generally unknown.
  - Admissibility conditions are generally unknown.

Digital filters with two's complement arithmetic

- What are digital filters?
  - Digital filters are systems that are characterized in the frequency domain.

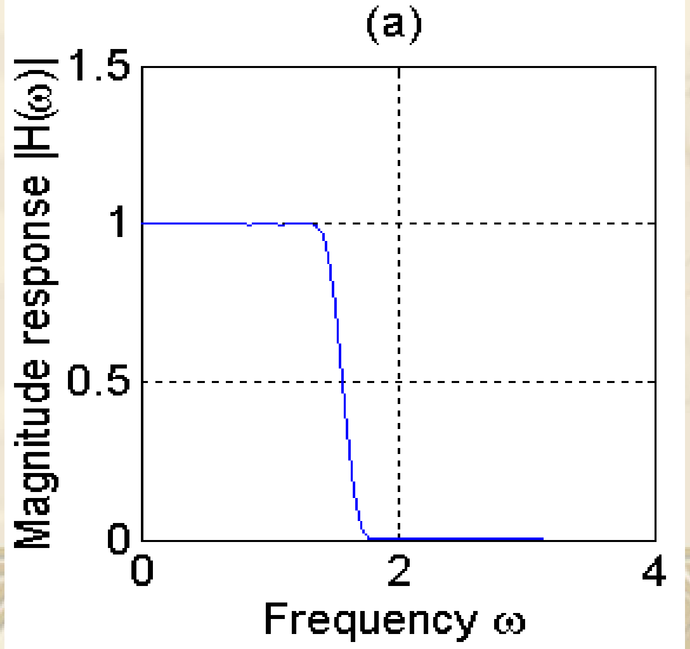
$$H(\omega) = \frac{Y(\omega)}{U(\omega)}$$

$$u(k) \longrightarrow \boxed{\mathrm{h(k)}} \longrightarrow y(k)$$

$$U(\omega) = \sum_{\forall k} u(k) e^{-j\omega k} \qquad H(\omega) = \sum_{\forall k} h(k) e^{-j\omega k} \qquad Y(\omega) = \sum_{\forall k} y(k) e^{-j\omega k}$$

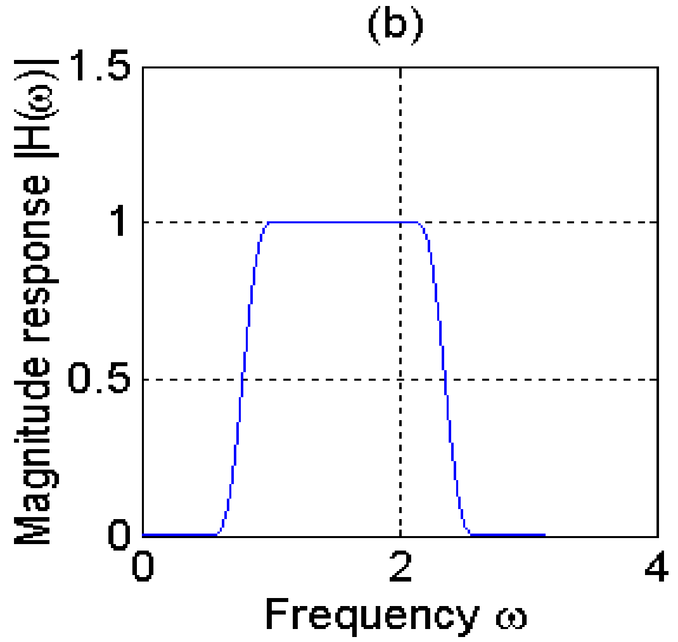
Digital filters with two's complement arithmetic

- Types of digital filters
  - Lowpass filters
    - Allow a low frequency band of a signal to pass through and attenuate a high frequency band.



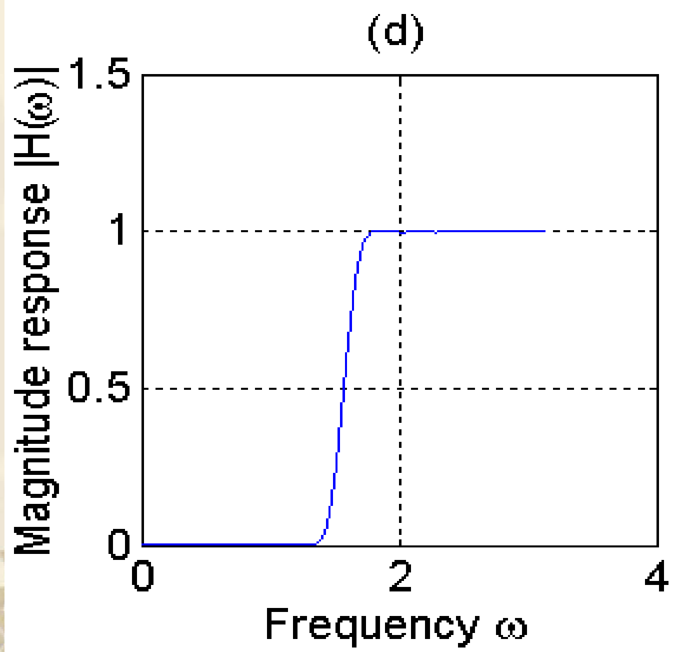
Digital filters with two's complement arithmetic

- Types of digital filters
  - Bandpass filters
    - Allow intermediate frequency bands of a signal to pass through and attenuate both low and high frequency bands.



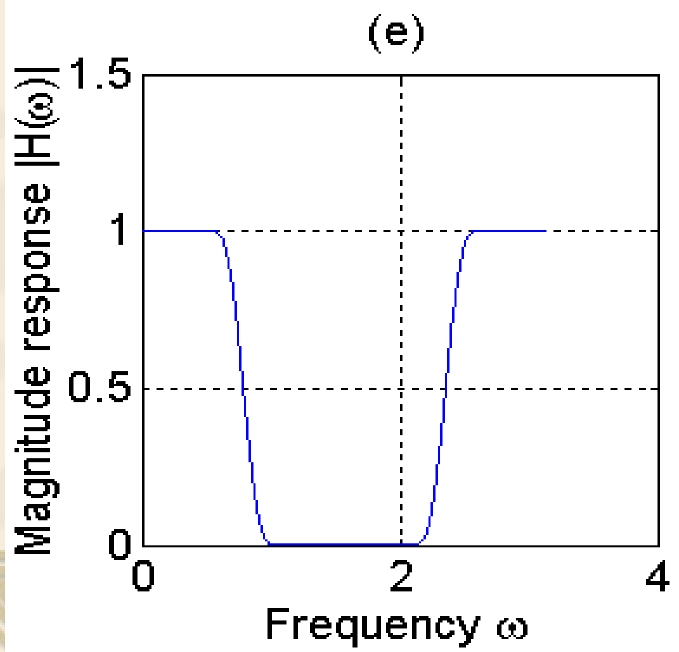
Digital filters with two's complement arithmetic

- Types of digital filters
  - Highpass filters
    - Allow a high frequency band of a signal to pass through and attenuate a low frequency band.



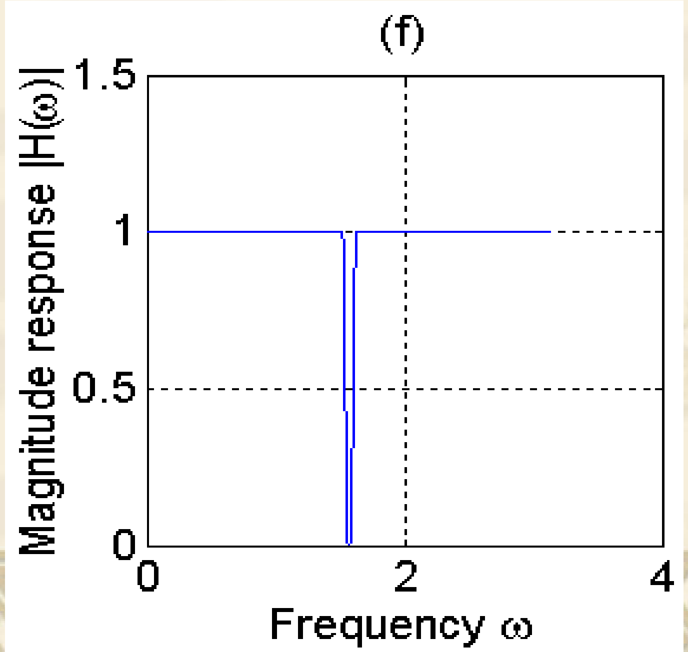
Digital filters with two's complement arithmetic

- Types of digital filters
  - Band reject filters
    - Allow both low and high frequency bands of a signal to pass through and attenuate intermediate frequency bands.



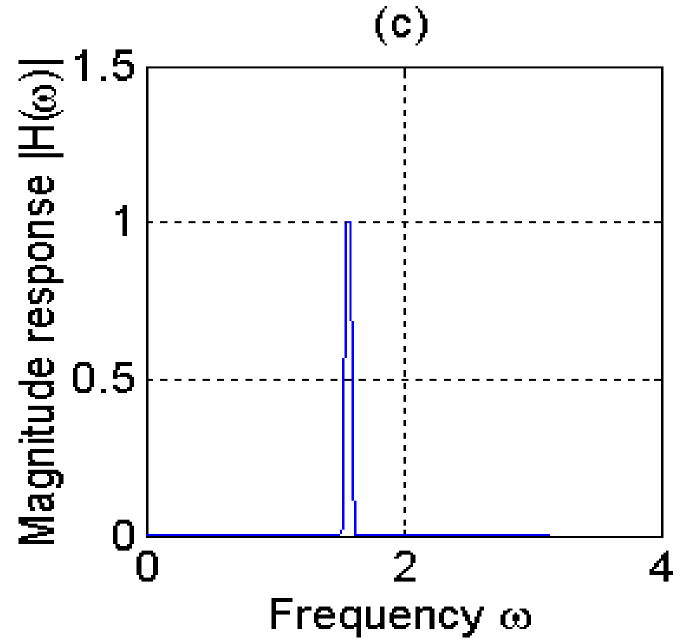
Digital filters with two's complement arithmetic

- Types of digital filters
  - Notch filters
    - Allow almost all frequency components to pass through but attenuate particular frequencies.



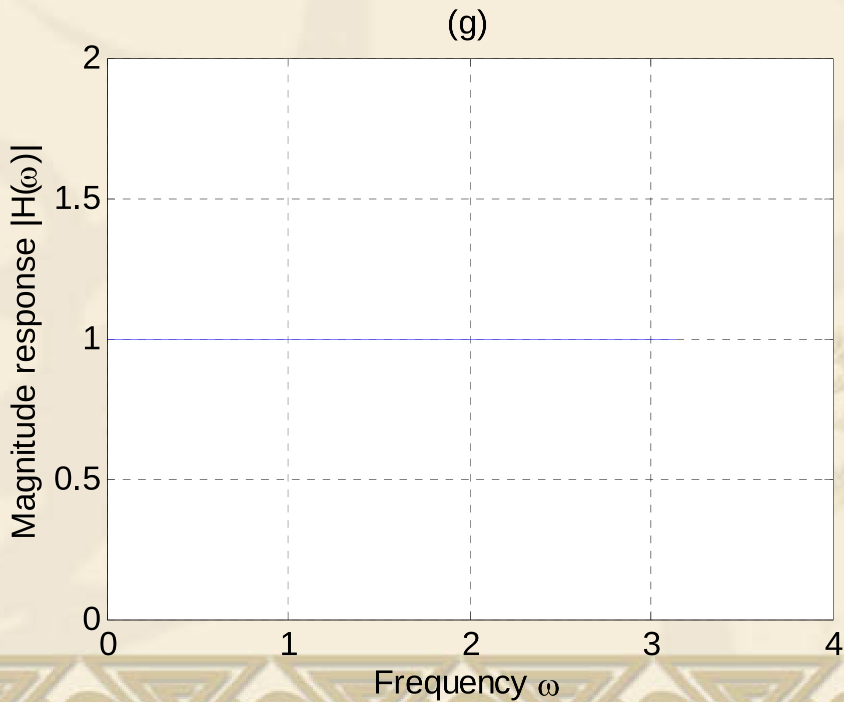
Digital filters with two's complement arithmetic

- Types of digital filters
  - Oscillators
    - Allow particular frequency components to pass through and attenuate almost all frequency components.



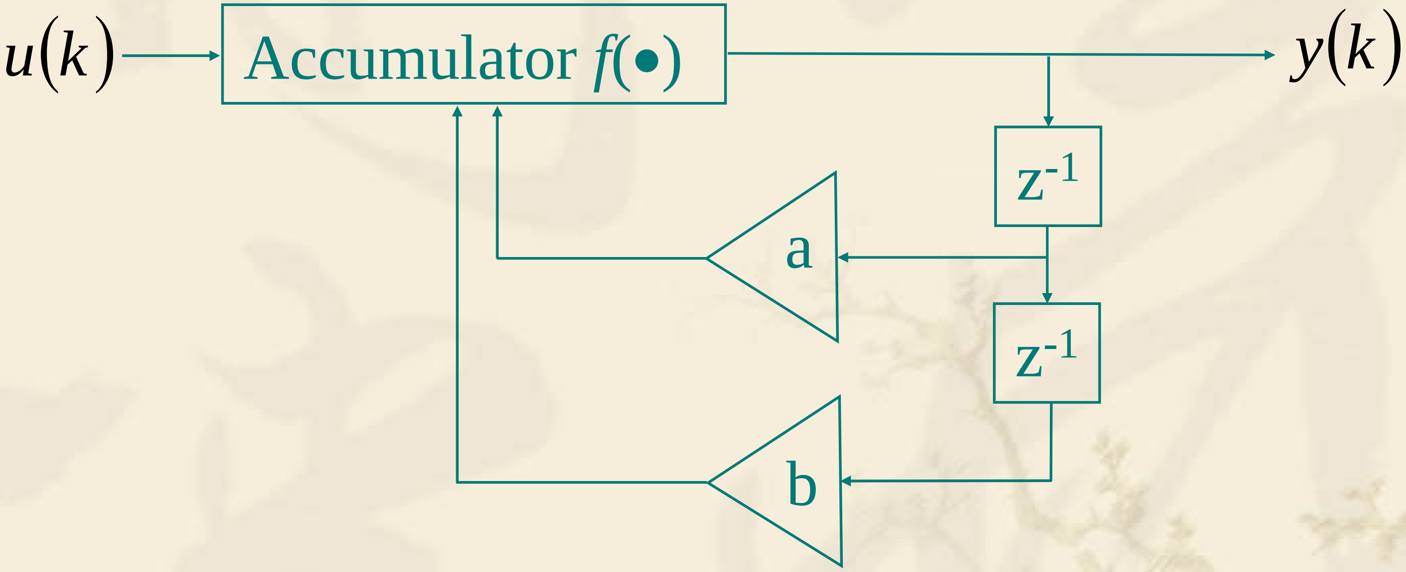
Digital filters with two's complement arithmetic

- Types of digital filters
  - Allpass filters
    - Allow all frequency components to pass through.



Digital filters with two's complement arithmetic

- Hardware schematic



Digital filters with two's complement arithmetic

- When no input is present, the filter can be described by the following nonlinear state space difference equation:

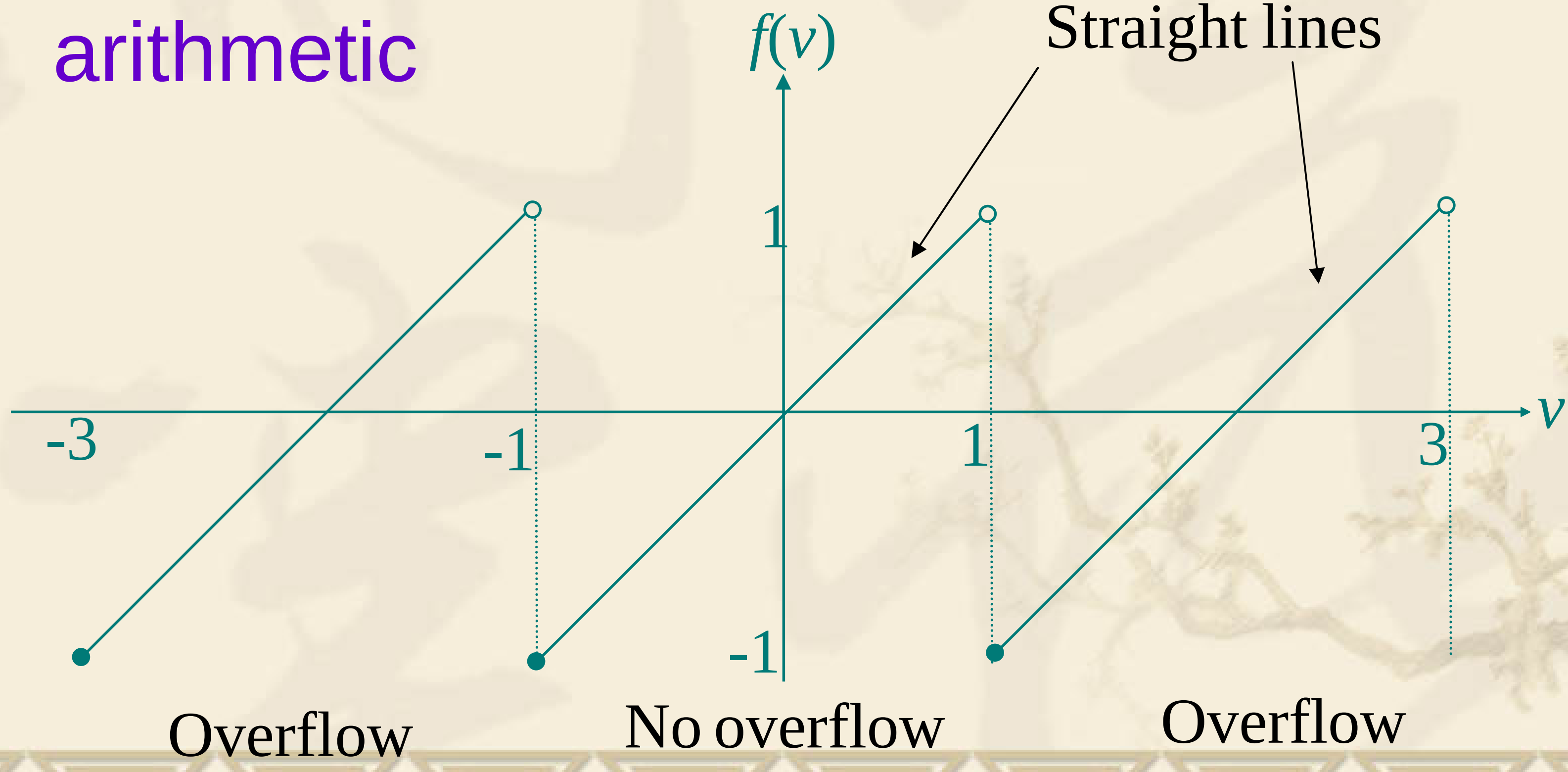
$$\mathbf{x}(k+1) \equiv \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \mathrm{F}(\mathbf{x}(k))$$

$$= \left(\begin{bmatrix} x_2(k) \\ f\left(b \cdot x_1(k) + a \cdot x_2(k)\right) \end{bmatrix} \right)$$

Direct form

Digital filters with two's complement arithmetic

where $f(\bullet)$ is the nonlinear function associated with the two's complement arithmetic



Digital filters with two's complement arithmetic

$$\mathbf{x}(k+1) = \mathbf{A}\cdot\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \end{bmatrix}\cdot s(k)$$

where $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ b & a \end{bmatrix}$

$$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \in I^2 \equiv \{(x_1, x_2): -1 \le x_1 < 1, -1 \le x_2 < 1\}$$

$$s(k) \in \{-m, \cdots, -1, 0, 1, \cdots, m\}$$

and *m* is the minimum integer satisfying

$$-2\cdot m - 1 \le b\cdot x_1 + a\cdot x_2 \le 2\cdot m + 1$$

Digital filters with two's complement arithmetic

$s(k)$ is called symbolic sequences.

$$s(k)=1 \Rightarrow -3 \le b\cdot x_1(k)+a\cdot x_2(k) < -1$$
$$s(k)=0 \Rightarrow -1 \le b\cdot x_1(k)+a\cdot x_2(k) < 1$$
$$s(k)=-1 \Rightarrow 1 \le b\cdot x_1(k)+a\cdot x_2(k) < 3$$

Digital filters with two's complement arithmetic

For example: $\mathbf{x}(0)=[-0.6135 \quad 0.6135]^T, b=-1 \text{ and } a=0.5$

$$b\cdot x_1(0)+a\cdot x_2(0)=0.9203 \Rightarrow s(0)=0$$

$$b\cdot x_1(1)+a\cdot x_2(1)=-0.1534 \Rightarrow s(1)=0$$

Two's complement value

$$\vdots$$

$$b\cdot x_1(14)+a\cdot x_2(14)=1.0018 \Rightarrow s(14)=-1 \text{ and } x_2(15)=-0.9982$$

$$b\cdot x_1(15)+a\cdot x_2(15)=-0.7597 \Rightarrow s(15)=0$$

$$b\cdot x_1(16)+a\cdot x_2(16)=0.6184 \Rightarrow s(16)=0$$

$$b\cdot x_1(17)+a\cdot x_2(17)=1.0689 \Rightarrow s(17)=-1 \text{ and } x_2(18)=-0.9311$$

$$b\cdot x_1(18)+a\cdot x_2(18)=-1.0839 \Rightarrow s(18)=1 \text{ and } x_2(19)=0.9161$$

$$\vdots$$

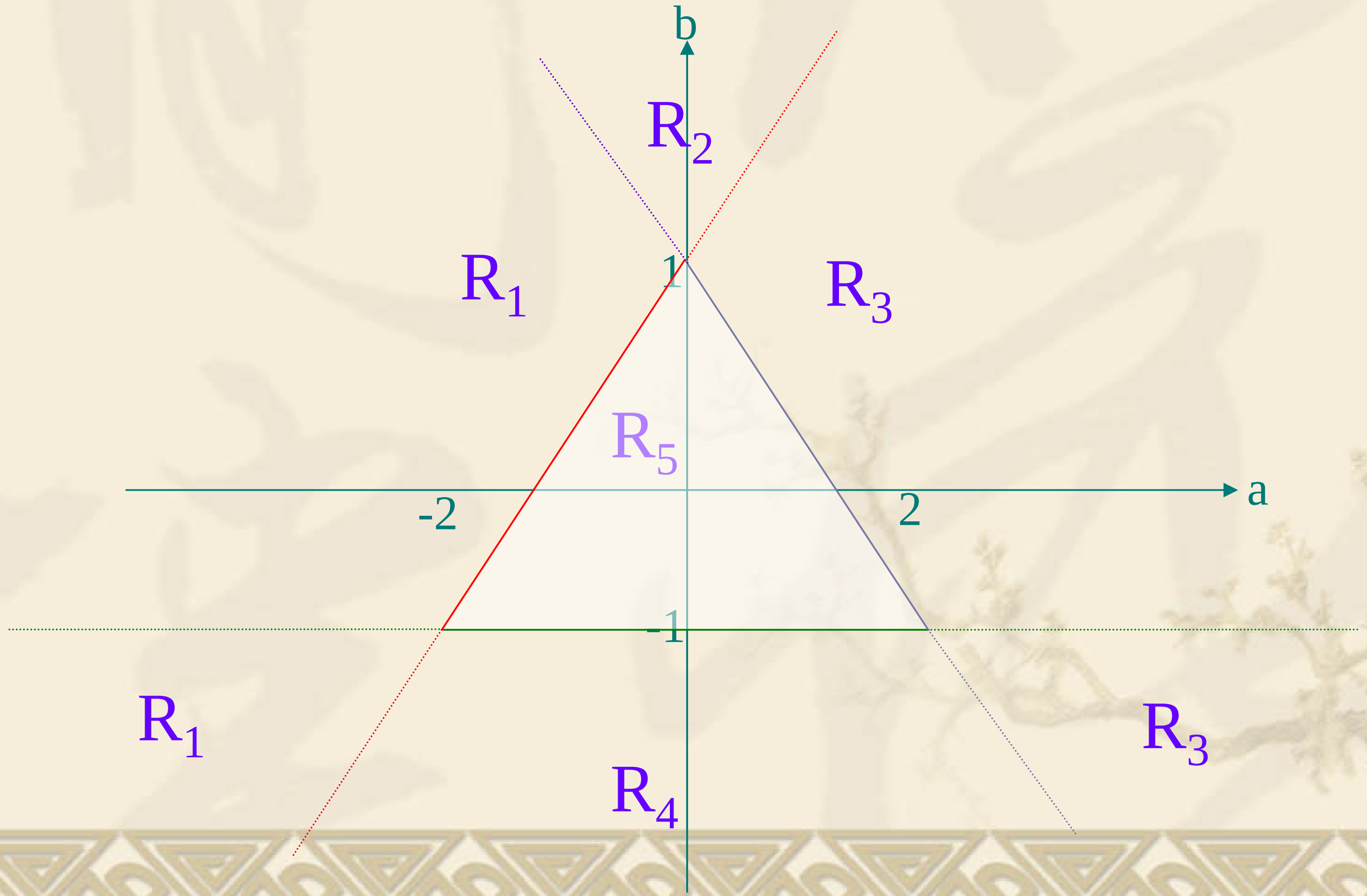
$$s=(0 \quad 0 \quad \cdots \quad 0 \quad -1 \quad 0 \quad 0 \quad -1 \quad 1 \quad \cdots)$$

Digital filters with two's complement arithmetic

- Stability analysis of the corresponding linear system
  - Eigenvalues of **A**

$$\lambda = \frac{a \pm \sqrt{a^2 + 4 \cdot b}}{2}$$

Digital filters with two's complement arithmetic



Digital filters with two's complement arithmetic

- **R_5** :Magnitudes of both eigenvalues <1.

  $\Rightarrow$The corresponding linear system is stable.

- **R_1** and **R_3** : Either one of the magnitudes of the eigenvalues < 1.

  $\Rightarrow$The corresponding linear system is unstable.

- **R_2** and **R_4** :Magnitudes of both eigenvalues are greater than 1.

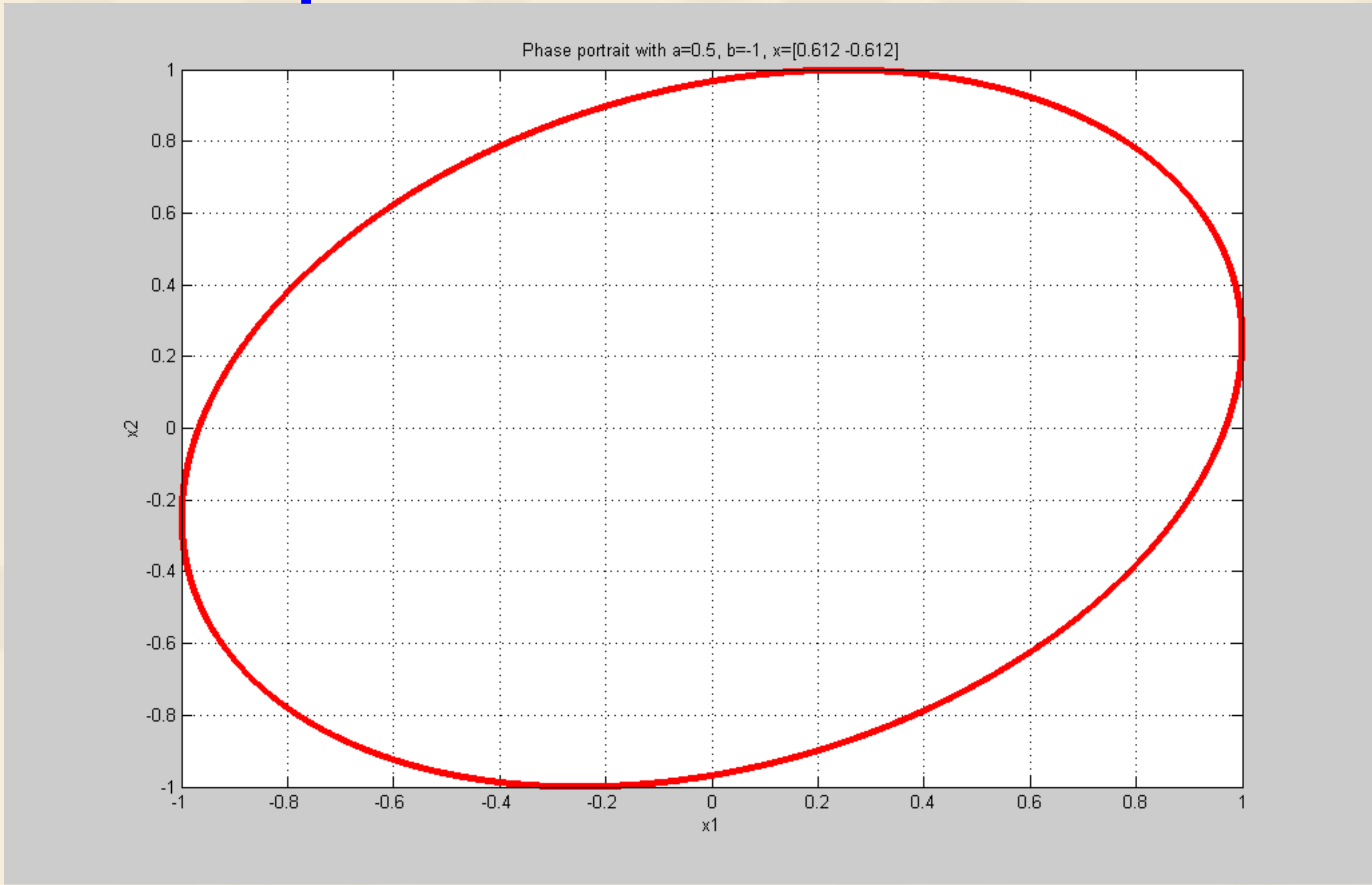
  $\Rightarrow$The corresponding linear system is unstable.

Digital filters with two's complement arithmetic

- Effects of different initial conditions when the eigenvalues of system matrix are on the unit circle
  - Type I trajectory
    - There is a single rotated and translated ellipse in the phase portrait.

$$a = 0.5, b = -1, x(0) = \begin{bmatrix} 0.612 \\ -0.612 \end{bmatrix}$$

Digital filters with two's complement arithmetic

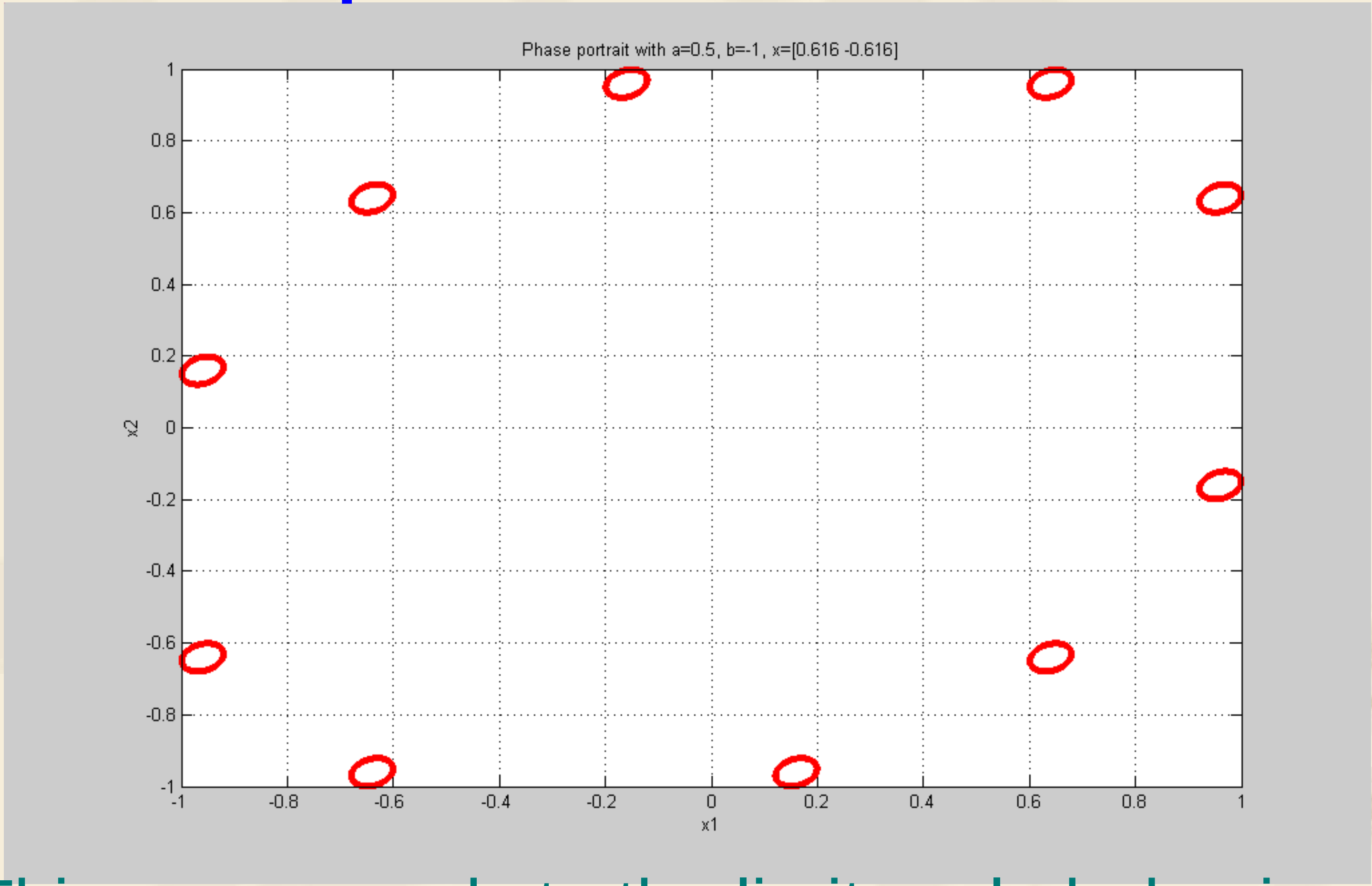


Digital filters with two's complement arithmetic

- Type II trajectory
  - There are some rotated and translated ellipses in the phase portrait.

$$a = 0.5, b = -1, x(0) = \begin{bmatrix} 0.616 \\ -0.616 \end{bmatrix}$$

Digital filters with two's complement arithmetic



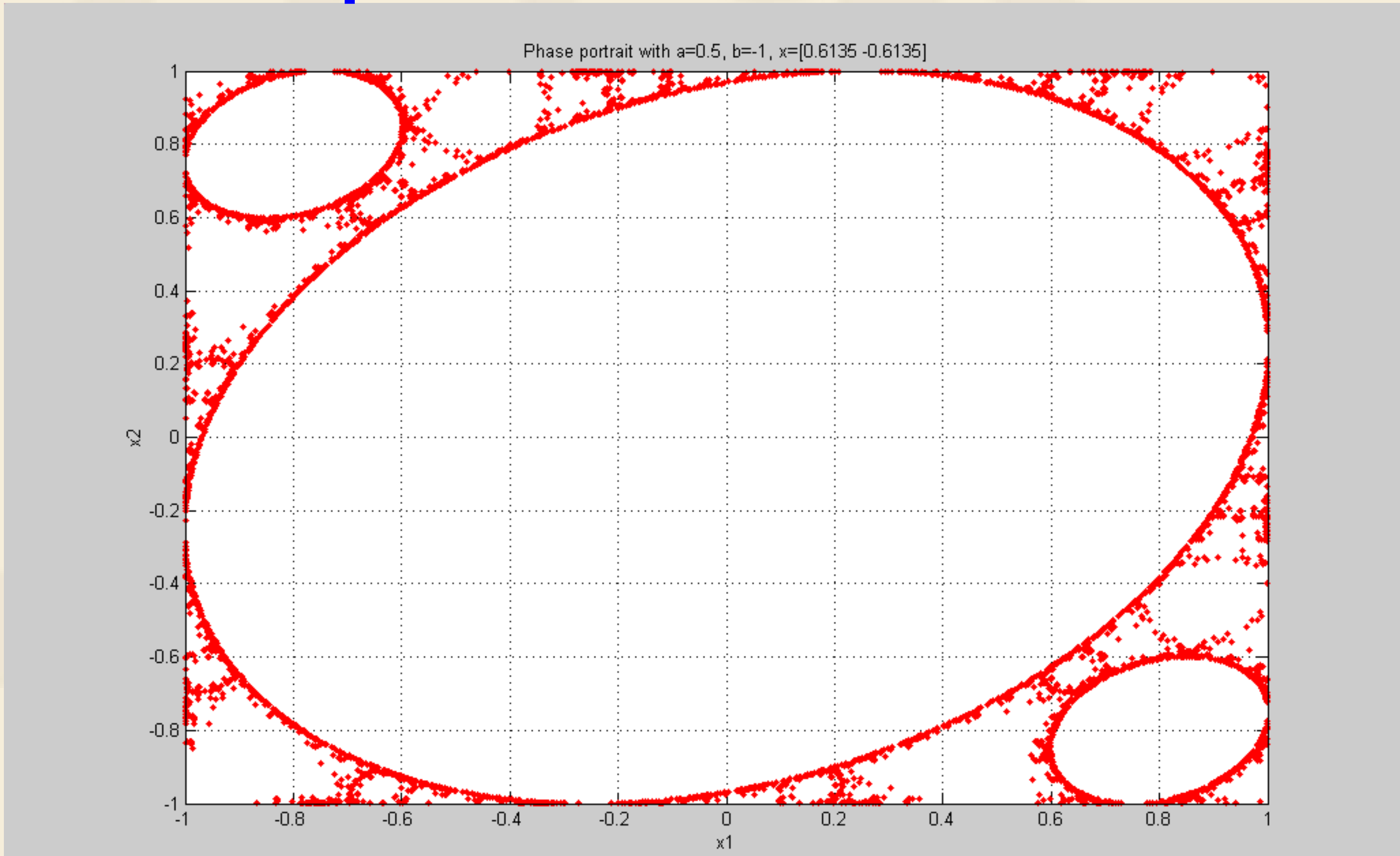
This corresponds to the limit cycle behavior.

Digital filters with two's complement arithmetic

- Type III trajectory
  - There is a fractal pattern on the phase portrait.

$$a = 0.5, b = -1, x(0) = \begin{bmatrix} 0.6135 \\ -0.6135 \end{bmatrix}$$

Digital filters with two's complement arithmetic



- Conclusion: Very sensitive to initial conditions

Digital filters with two's complement arithmetic

- Step response of second order digital filters with two's complement arithmetic
  - The system can be represented as:

$$\mathbf{x}(k+1)=\left(\begin{bmatrix} x_2(k) \\ f\left(b\cdot x_1(k)+a\cdot x_2(k)+u(k)\right)\end{bmatrix}\right)$$

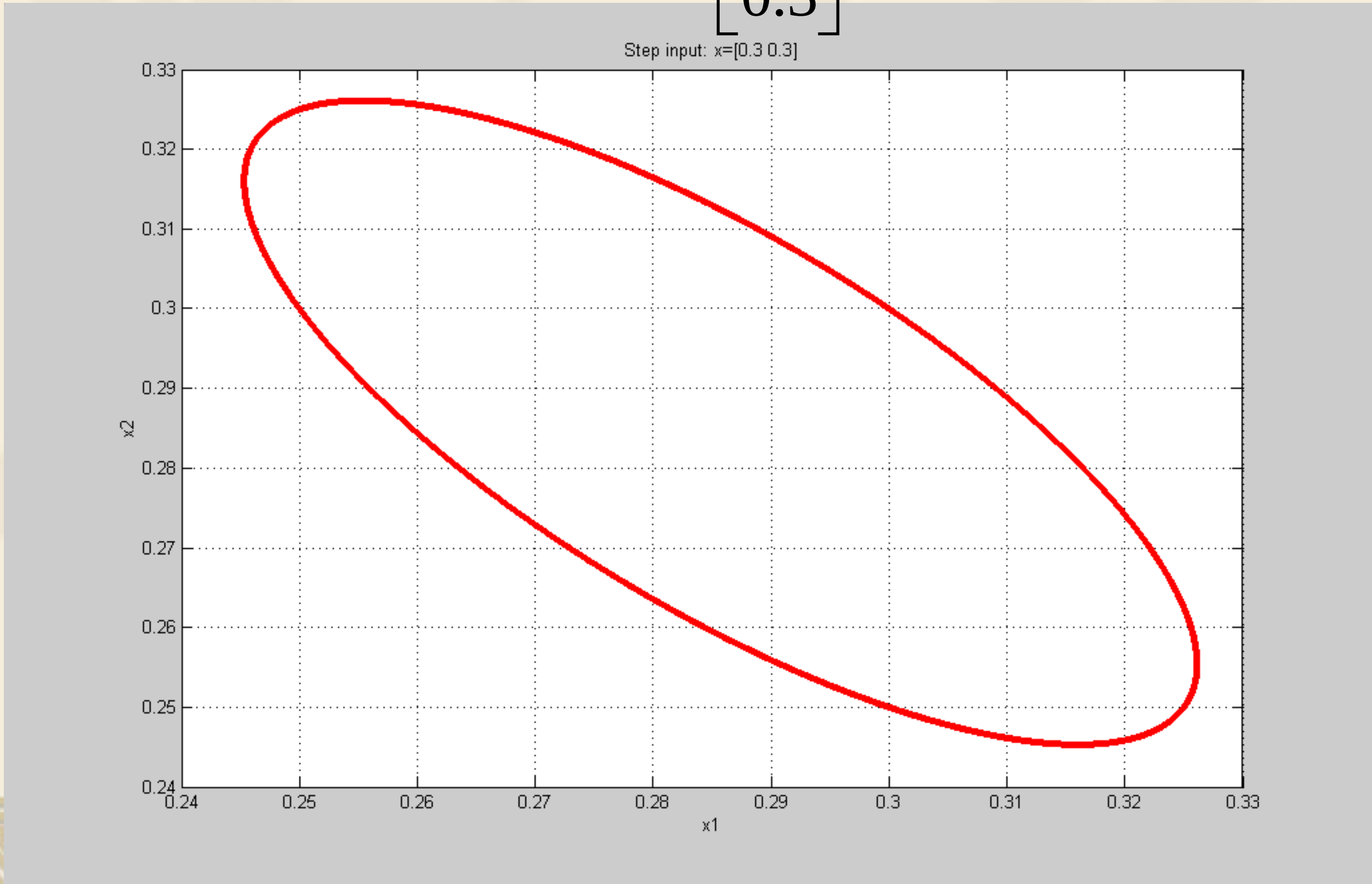
$$=\mathbf{A}\cdot\begin{bmatrix} x_1(k) \\ x_2(k)\end{bmatrix}+\mathbf{B}\cdot u(k)+\begin{bmatrix} 0 \\ 2\end{bmatrix}\cdot s(k)$$

where $\mathbf{A}=\begin{bmatrix} 0 & 1 \\ b & a\end{bmatrix}$ and $\mathbf{B}=\begin{bmatrix} 0 \\ 1\end{bmatrix}$

$\Rightarrow$DC offset on symbolic sequences.

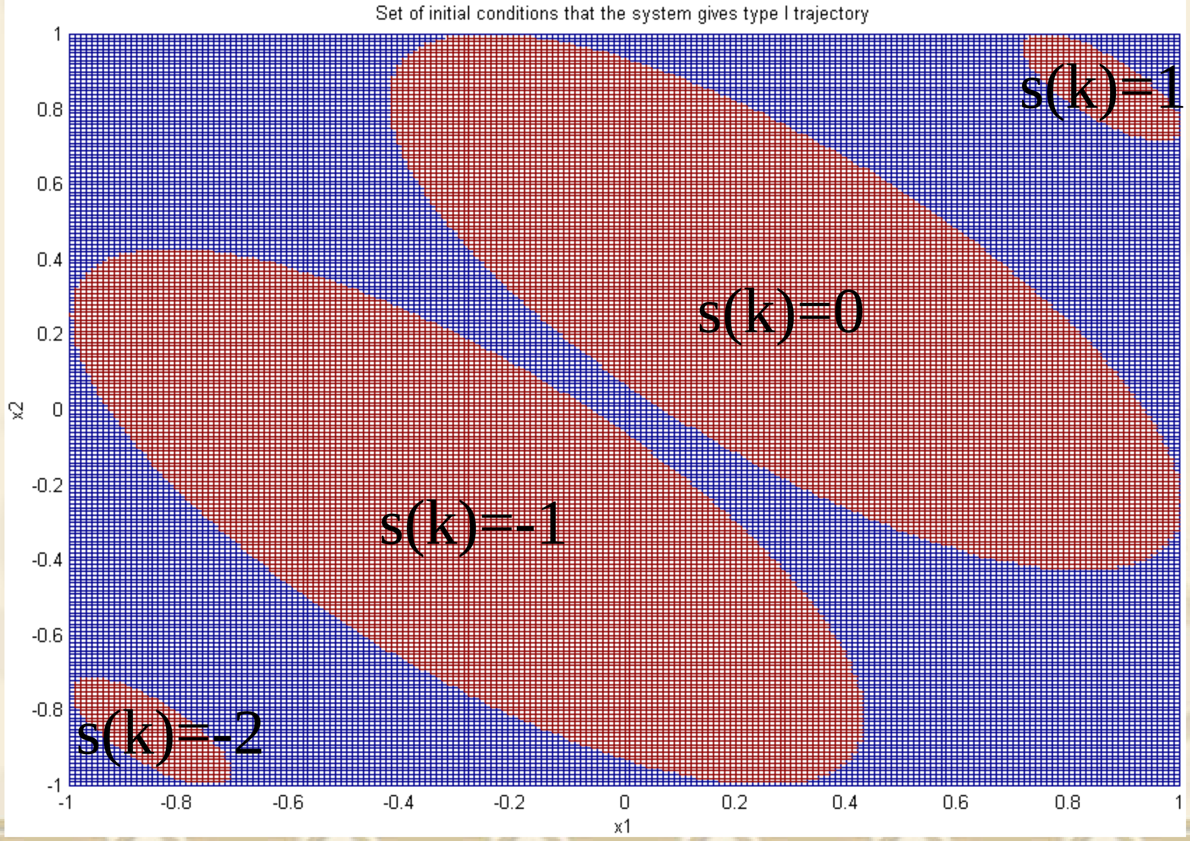
Digital filters with two's complement arithmetic

For $a=-1.5, b=-1, c=1, \mathbf{x}(0)=\begin{bmatrix}0.3\\0.3\end{bmatrix}$



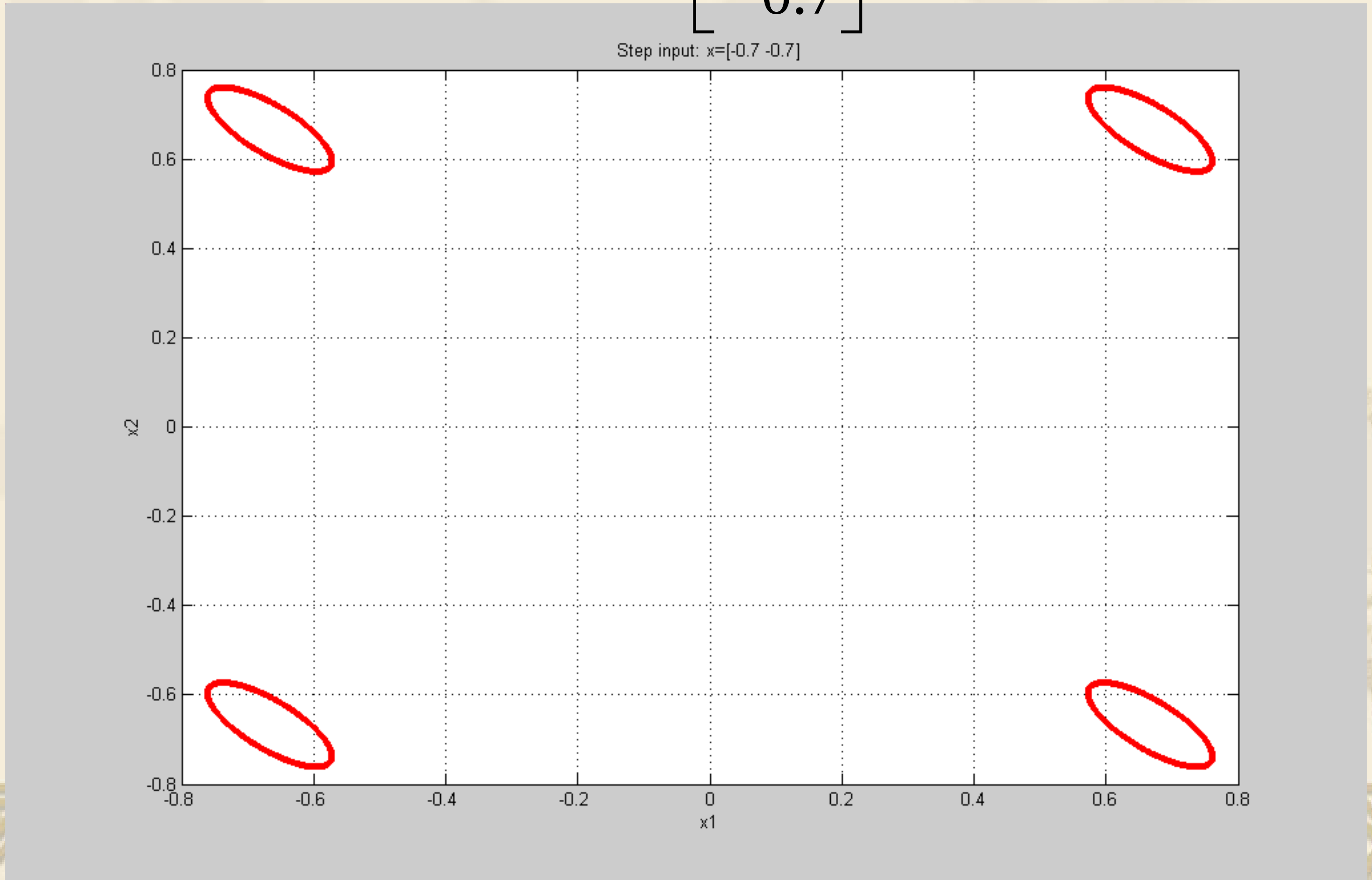
Digital filters with two's complement arithmetic

For $a = -1.5, b = -1 \text{ and } c = 1$



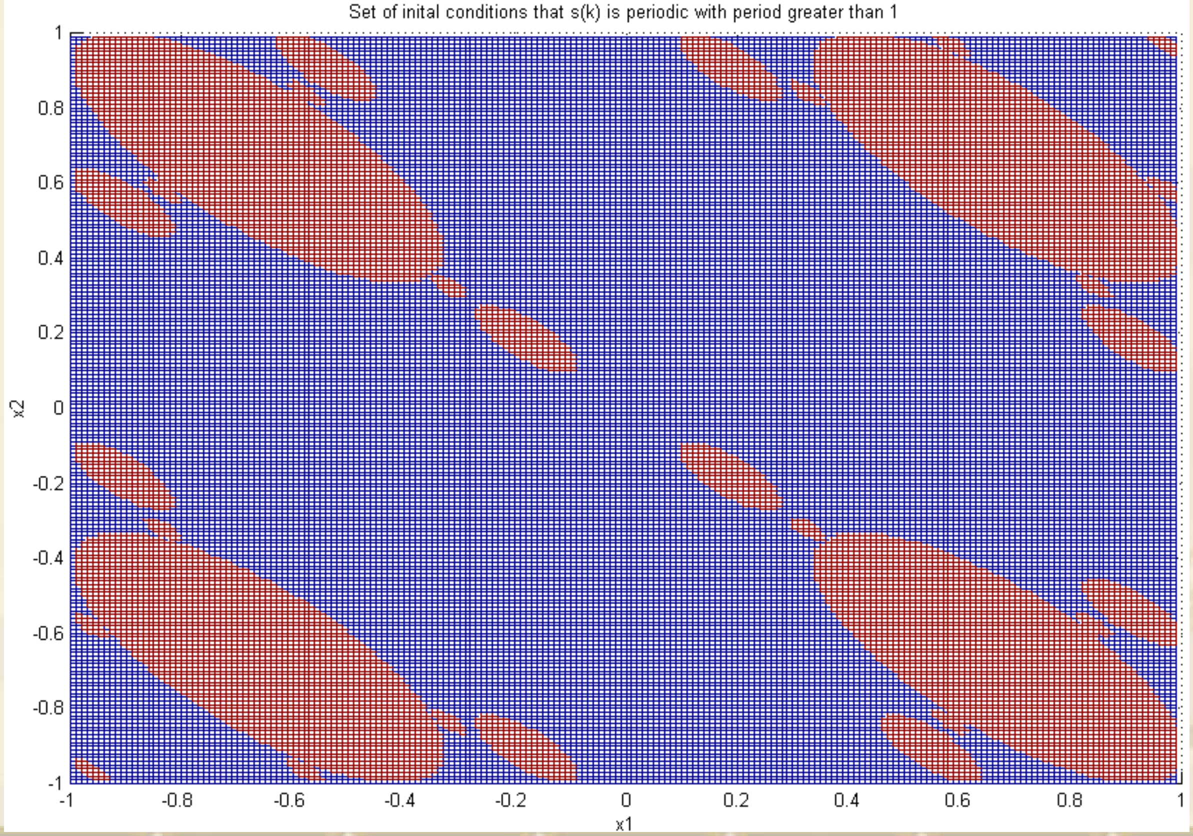
Digital filters with two's complement arithmetic

For $a = -1.5, b = -1, c = 1, \mathbf{x}(0) = \begin{bmatrix} -0.7 \\ -0.7 \end{bmatrix}, s = (0, -1, -1, 0, \cdots)$



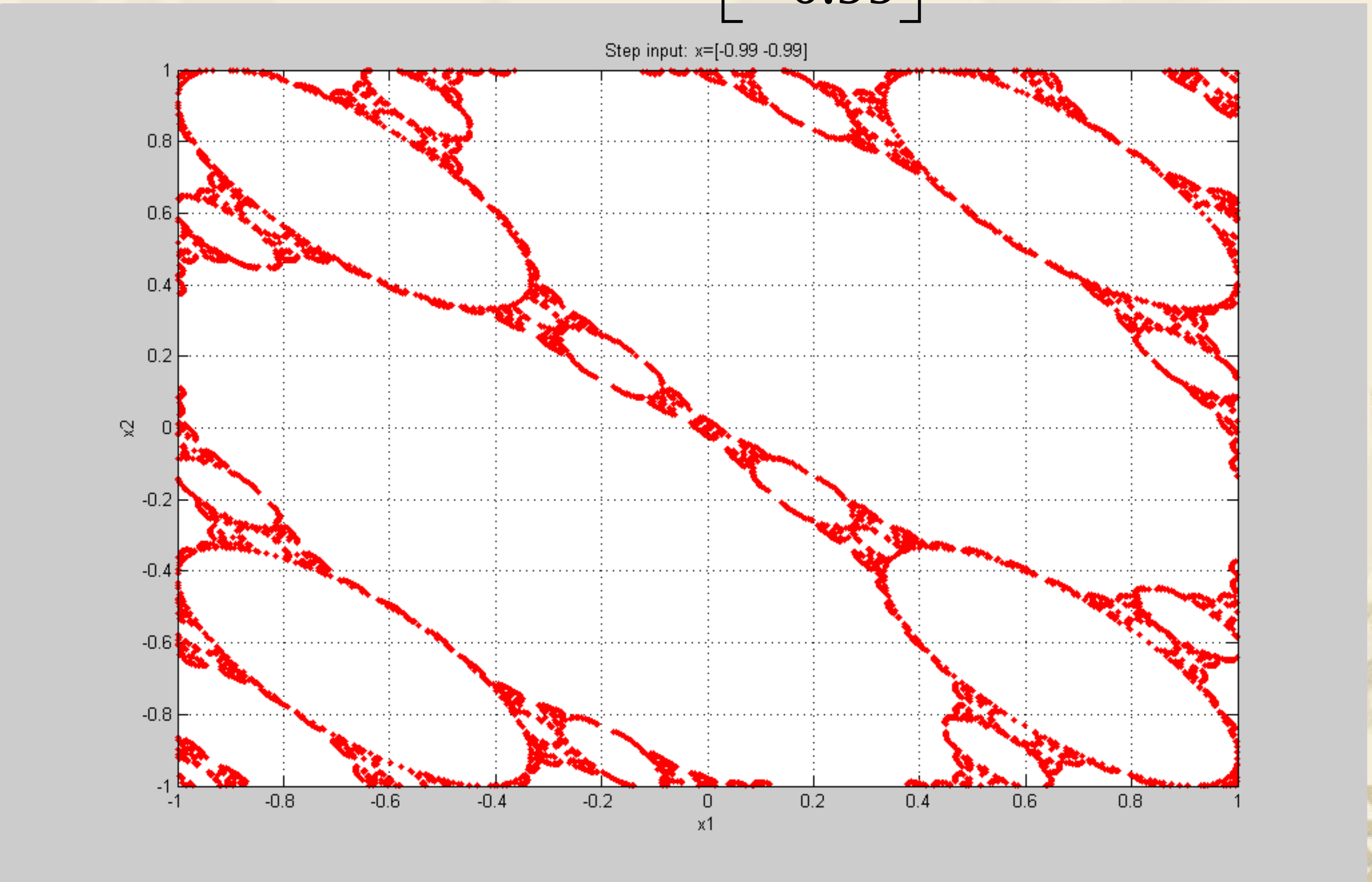
Digital filters with two's complement arithmetic

For $a = -1.5, b = -1 \text{ and } c = 1$



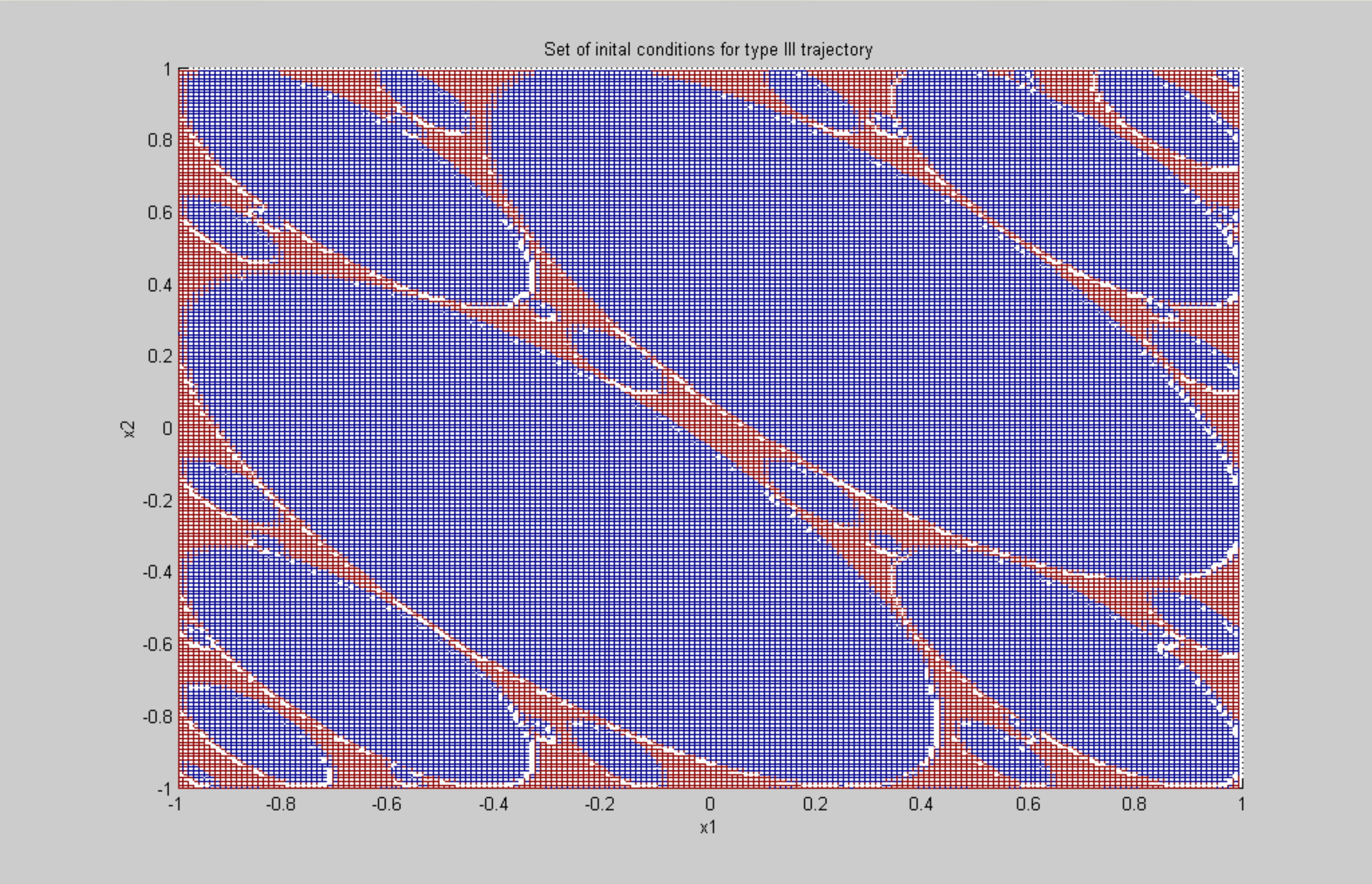
Digital filters with two's complement arithmetic

For $a = -1.5, b = -1, c = 1, \mathbf{x}(0) = \begin{bmatrix} -0.99 \\ -0.99 \end{bmatrix}$



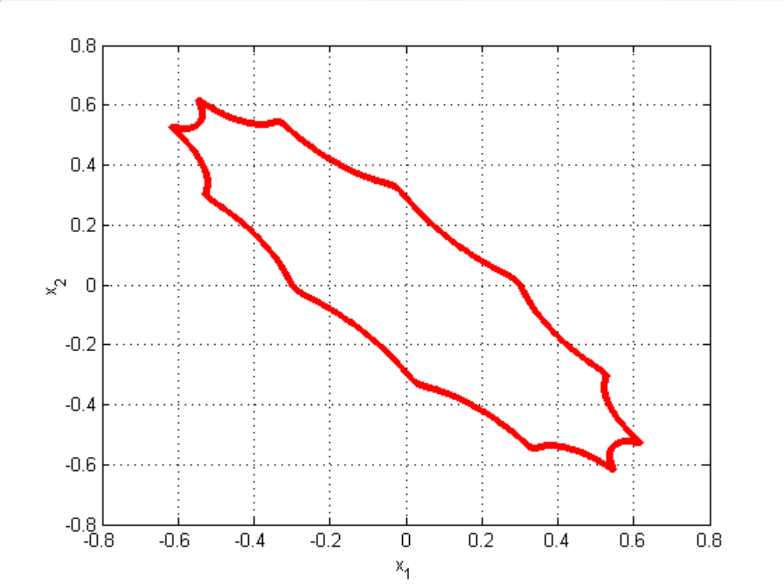
Digital filters with two's complement arithmetic

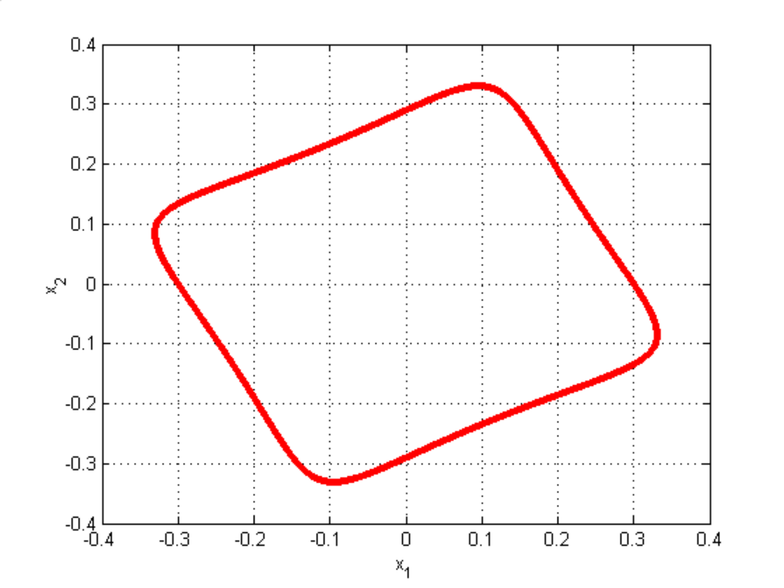
For $a = -1.5, b = -1$ and $c = 1$



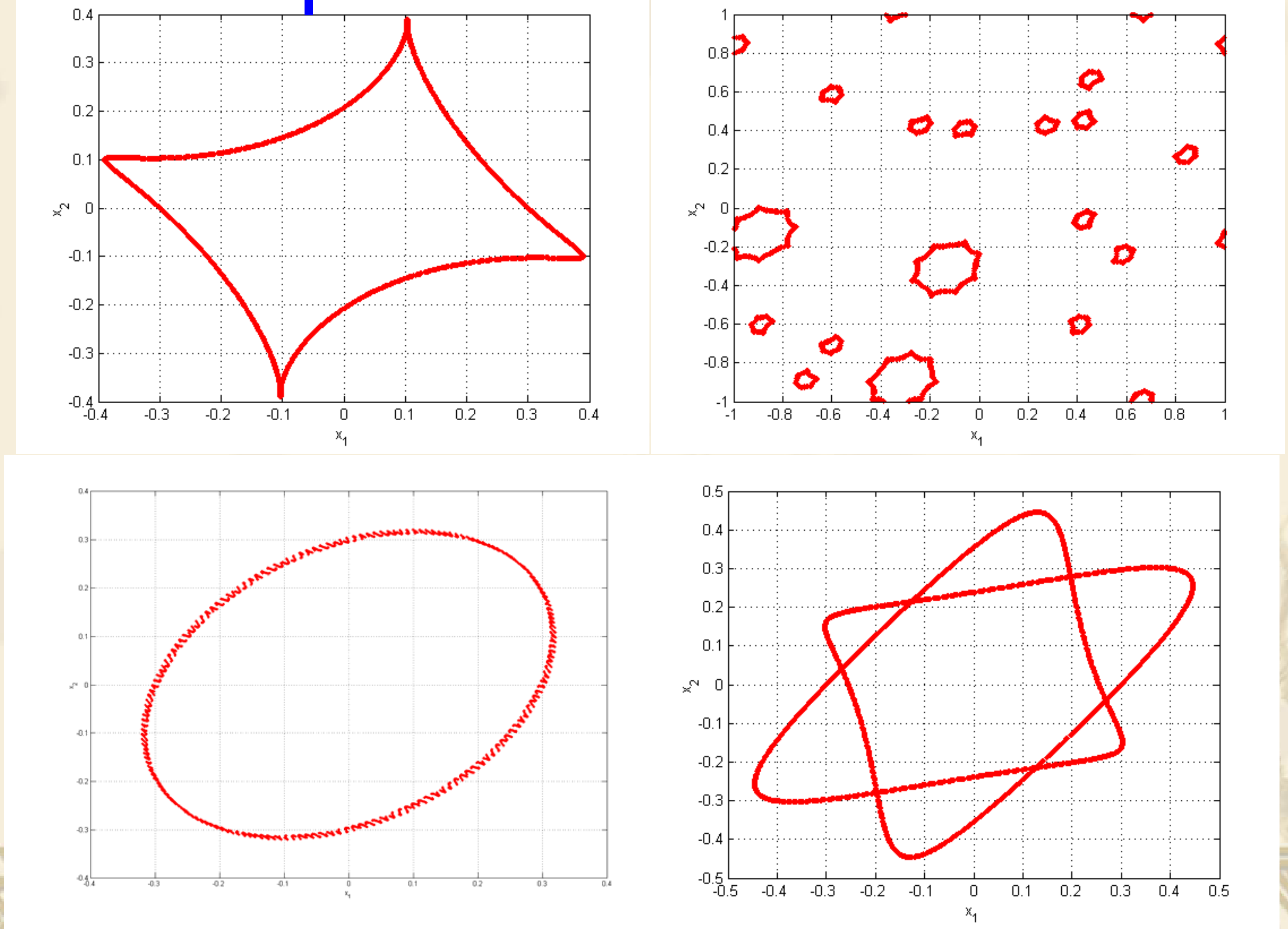
Digital filters with two's complement arithmetic

- Sinusoidal response of second order digital filters with two's complement arithmetic



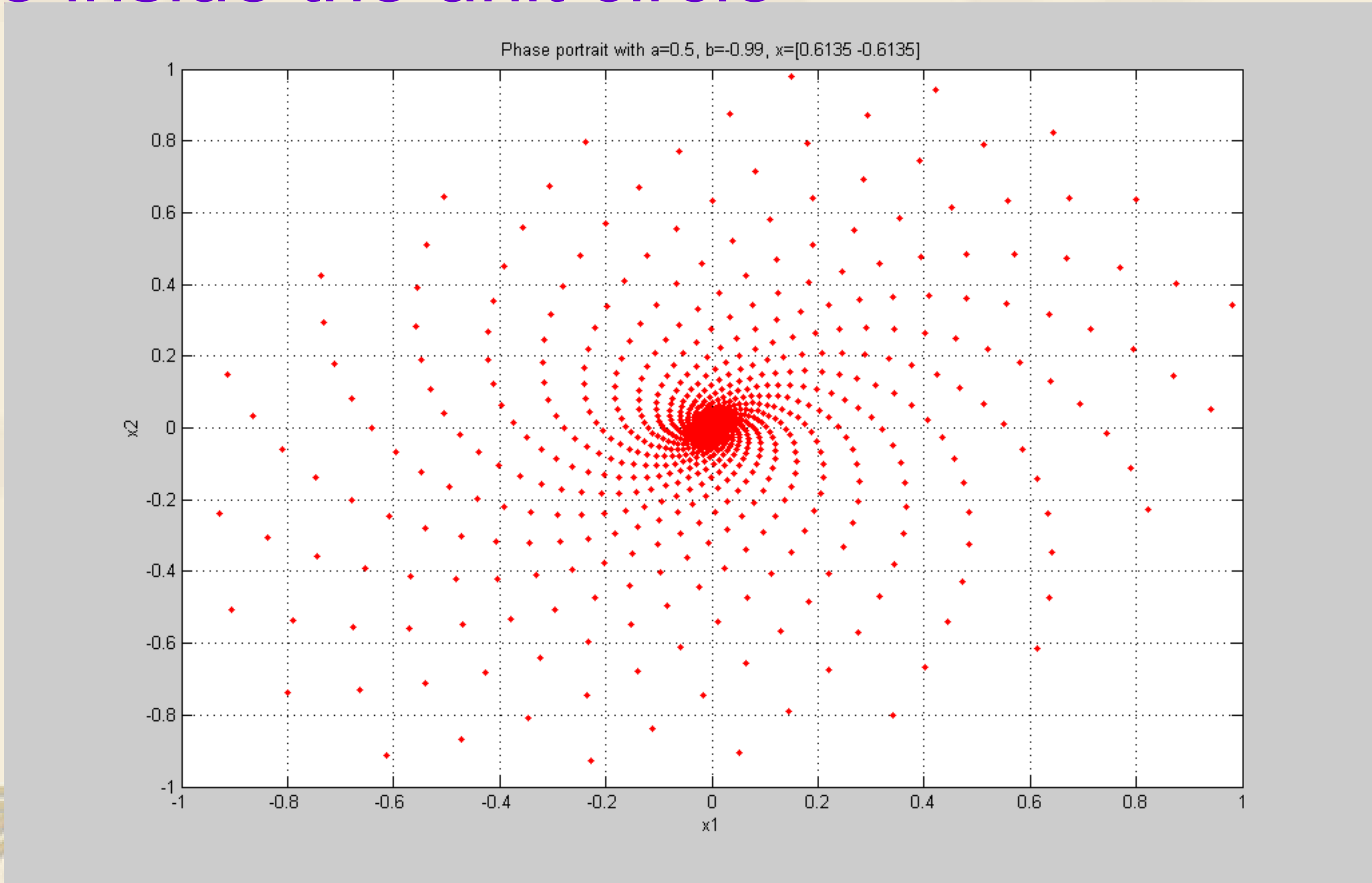


Digital filters with two's complement arithmetic

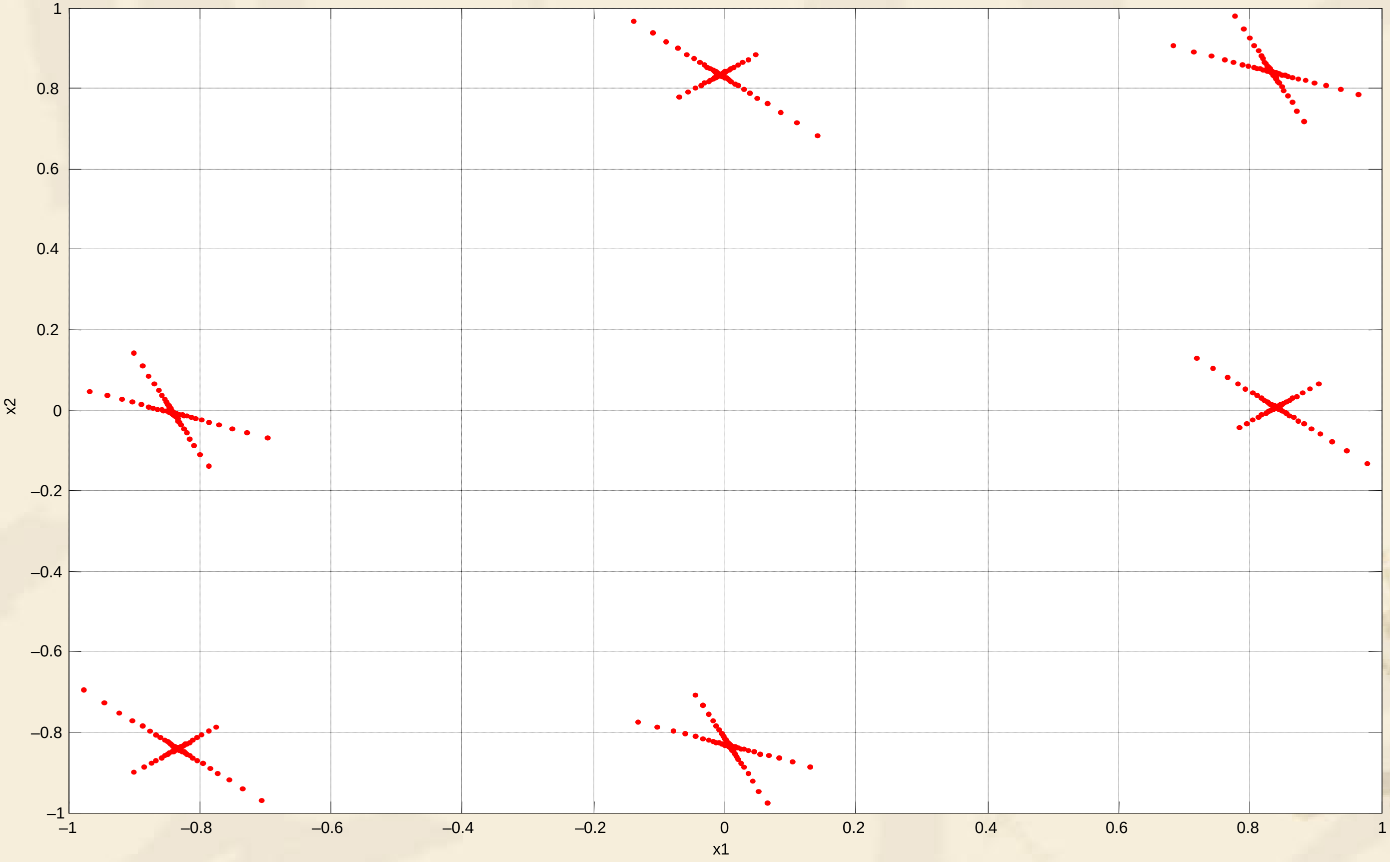


Digital filters with two's complement arithmetic

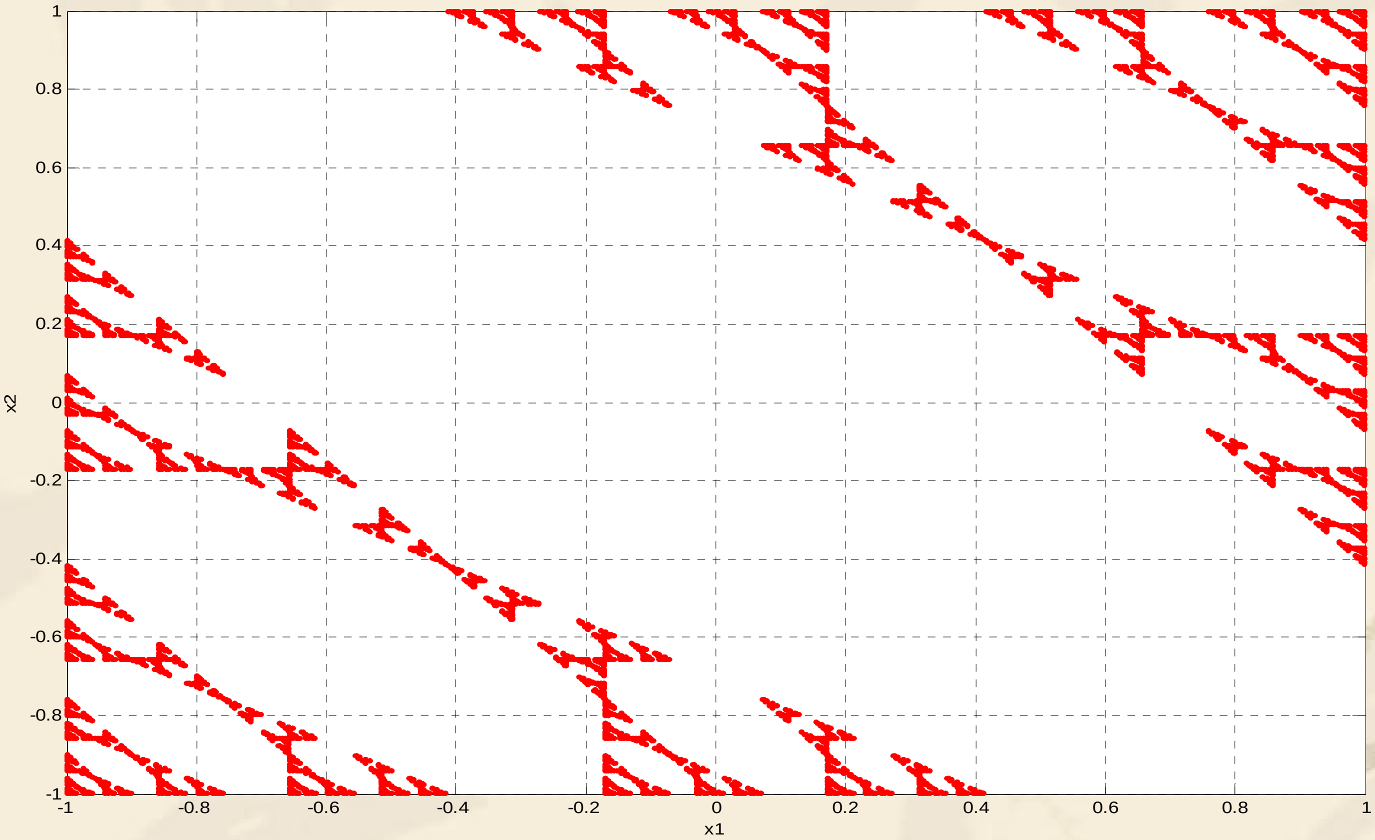
- Autonomous case when the eigenvalues of **A** are inside the unit circle



Digital filters with two's complement arithmetic

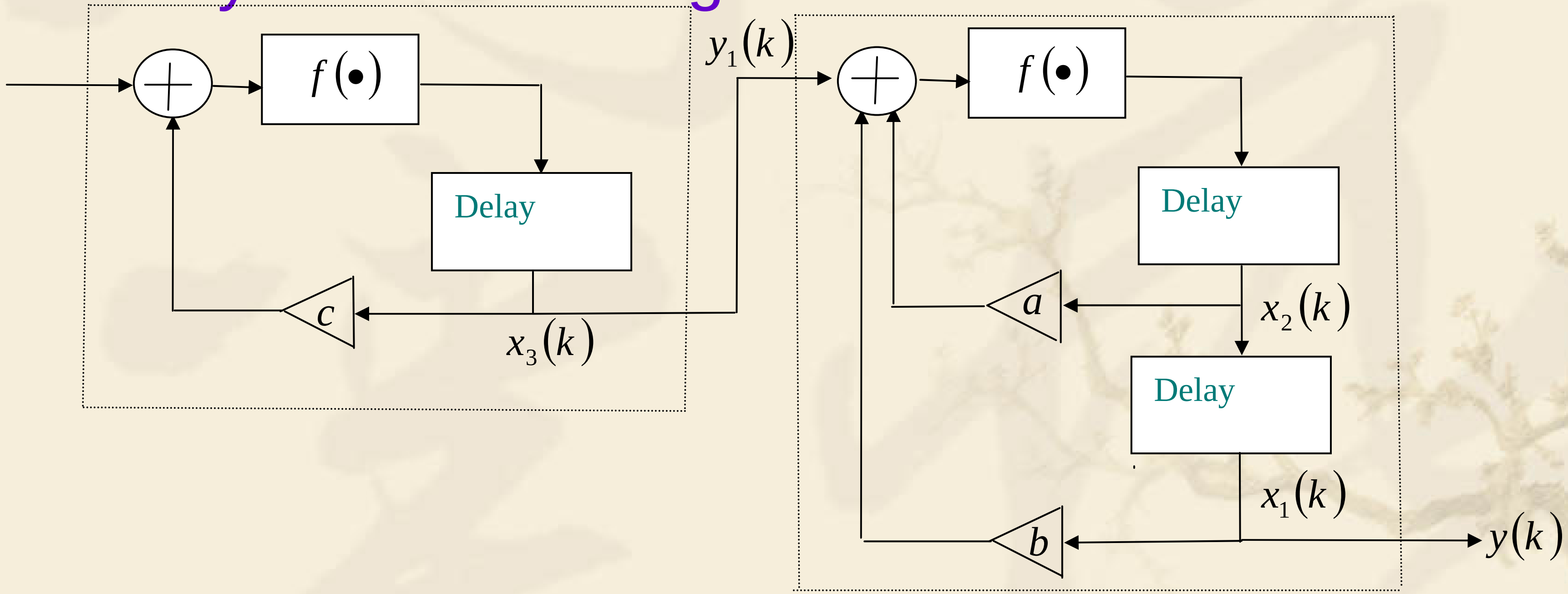


Digital filters with two's complement arithmetic

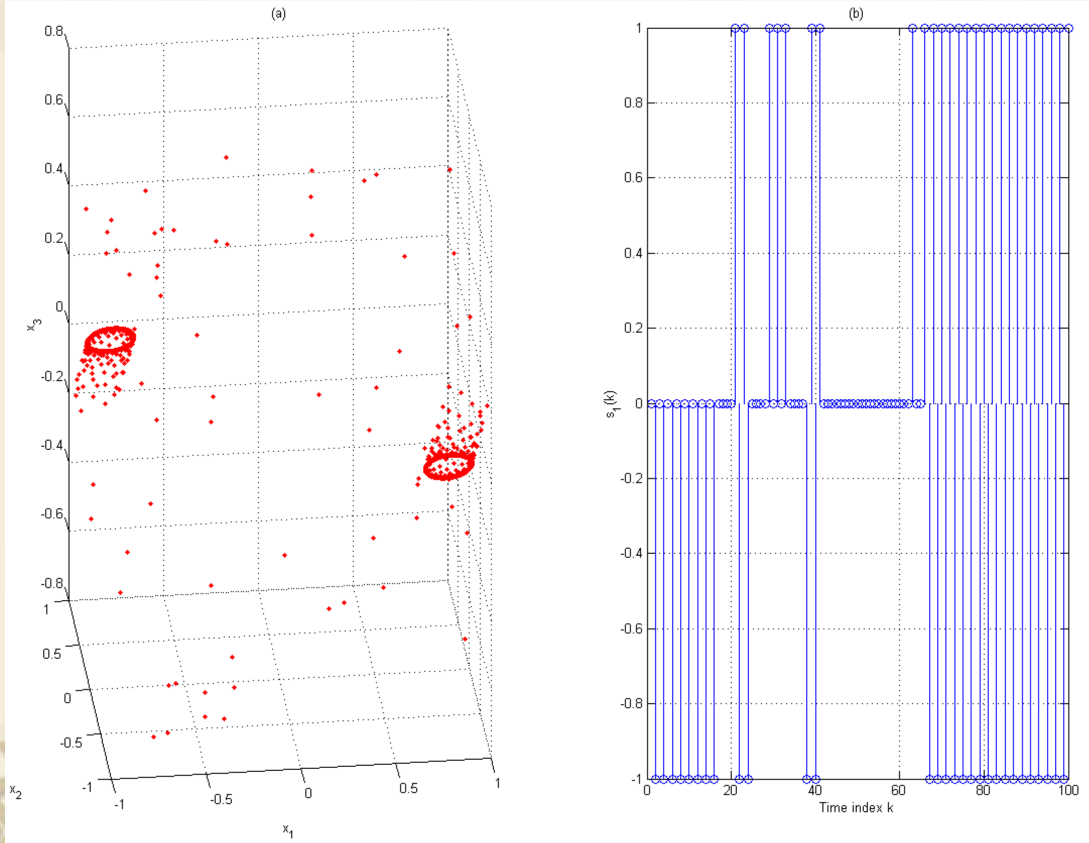


Digital filters with two's complement arithmetic

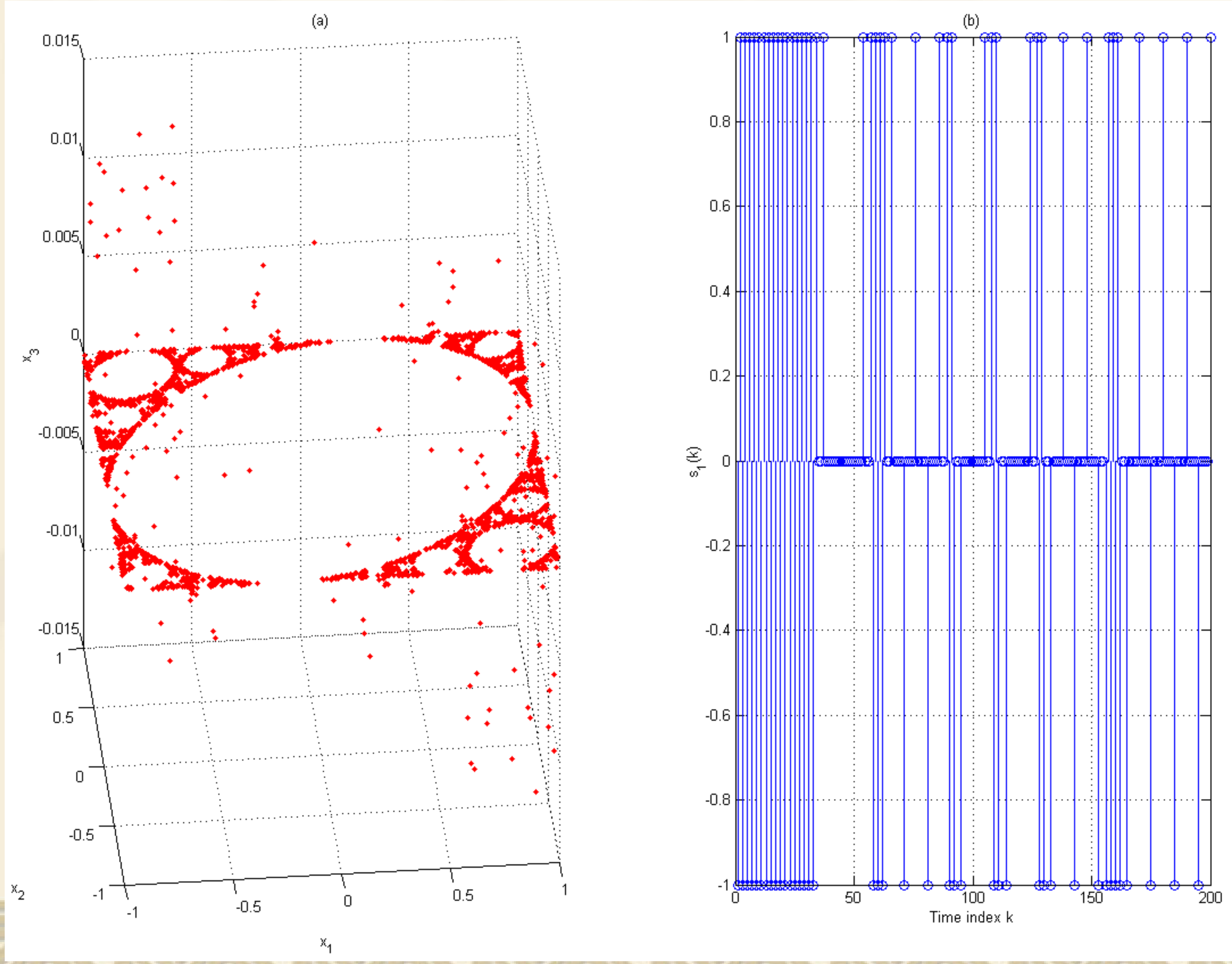
- Autonomous case for third order digital filters with two's complement arithmetic realized in cascade form
  - System block diagram:



Digital filters with two's complement arithmetic

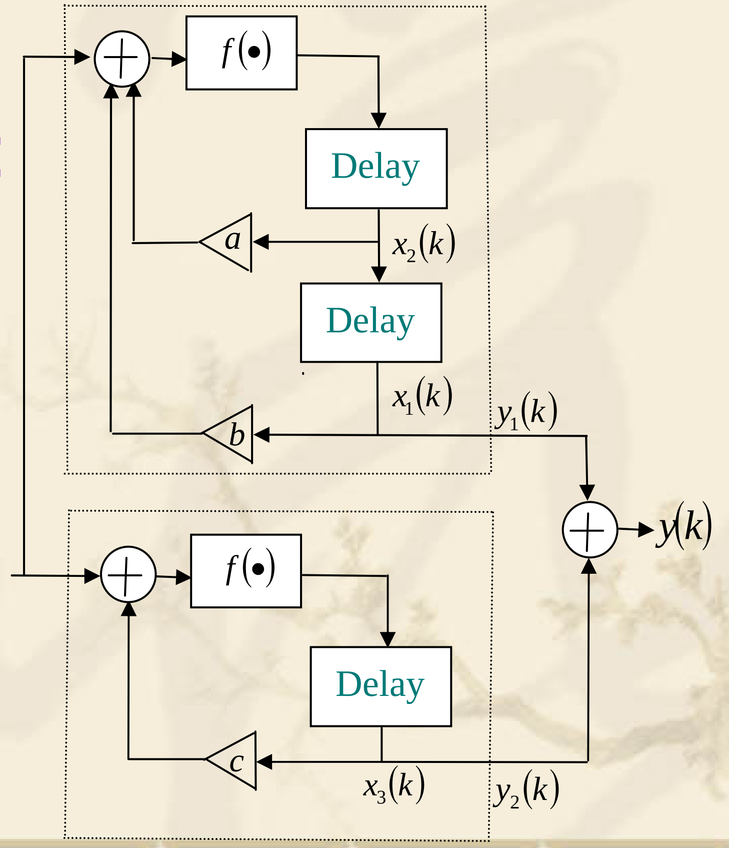


Digital filters with two’s complement arithmetic

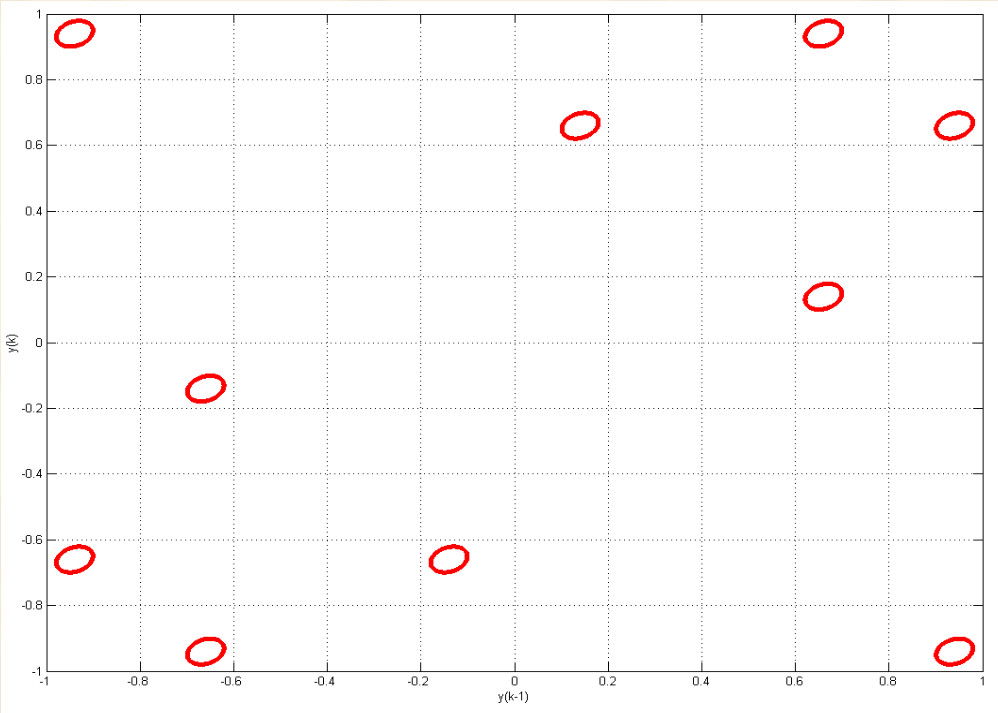


Digital filters with two's complement arithmetic

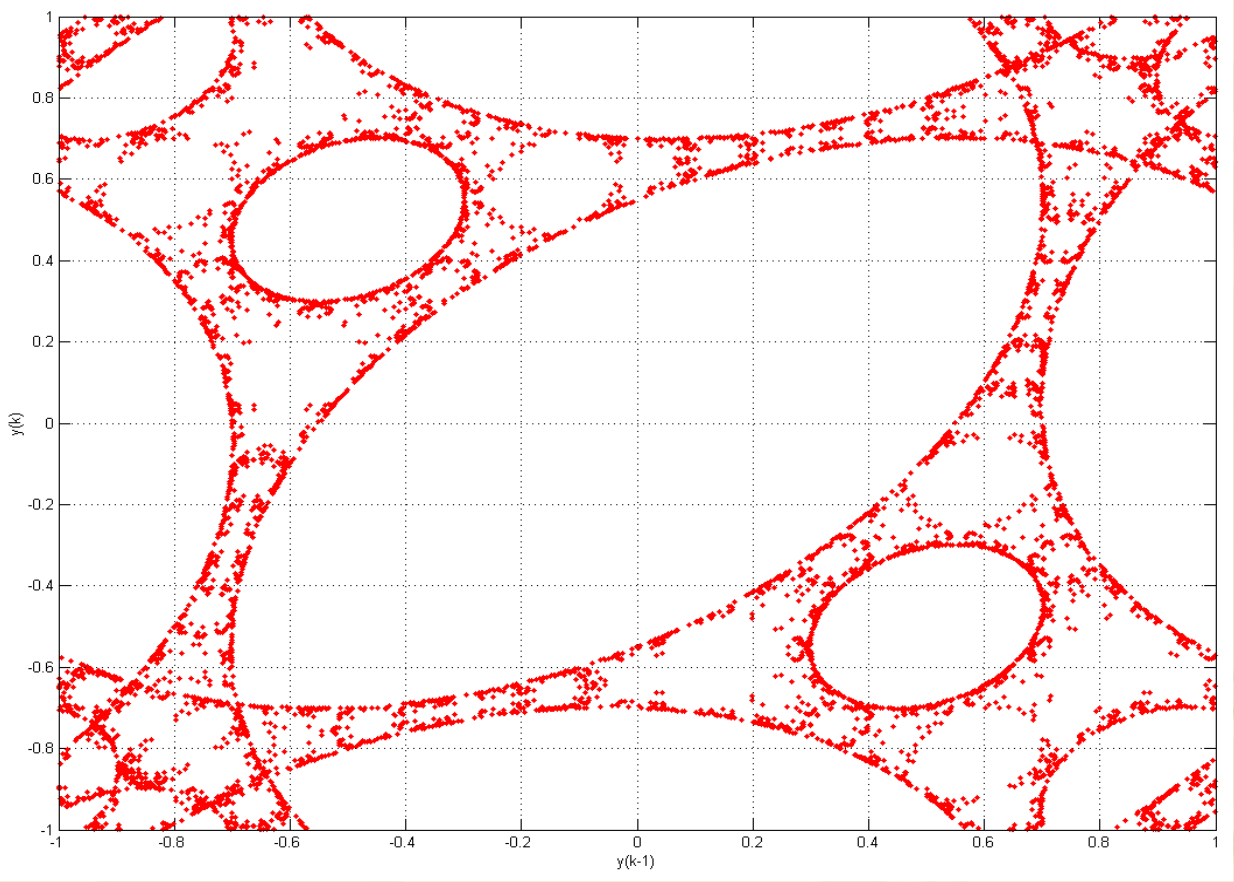
- Autonomous case for third-order digital filters with two's complement arithmetic realized in parallel form
  - System block diagram:



Digital filters with two's complement arithmetic



Digital filters with two's complement arithmetic



Digital filters with two's complement arithmetic

- Admissibility of second order digital filters with two's complement arithmetic
  - Admissible set of periodic symbolic sequences is defined as a set of periodic symbolic sequences such that there exists an initial condition that produces the symbolic sequences.

Digital filters with two’s complement arithmetic

For example, when $a=0.5, b=-1, c=0,$

M	Admissible s	Not admissible s
1	$s=(0,\cdots)$	$s=(1,\cdots)$ $s=(-1,\cdots)$
2	$s=(-1,1,\cdots)$ $s=(1,-1,\cdots)$	$s=(1,0,\cdots)$ $s=(0,1,\cdots)$
		$s=(-1,0,\cdots)$ $s=(0,-1,\cdots)$
3	$s=(0,0,1,\cdots)$ $s=(0,1,0,\cdots)$	$s=(0,1,1,\cdots)$ $s=(0,1,-1,\cdots)$
	$s=(1,0,0,\cdots)$ $s=(-1,0,0,\cdots)$	$s=(0,-1,1,\cdots)$ $s=(0,-1,-1,\cdots)$
	$s=(0,0,-1,\cdots)$ $s=(0,-1,0,\cdots)$	$s=(1,0,1,\cdots)$ $s=(1,0,-1,\cdots)$
		$s=(1,-1,0,\cdots)$ $s=(1,-1,0,\cdots)$
		$\cdots$ $\cdots$
15	Not found	

Digital filters with two's complement arithmetic

A periodic sequence s with period M is admissible if and only if for $i = 0,1,\cdots,M-1$

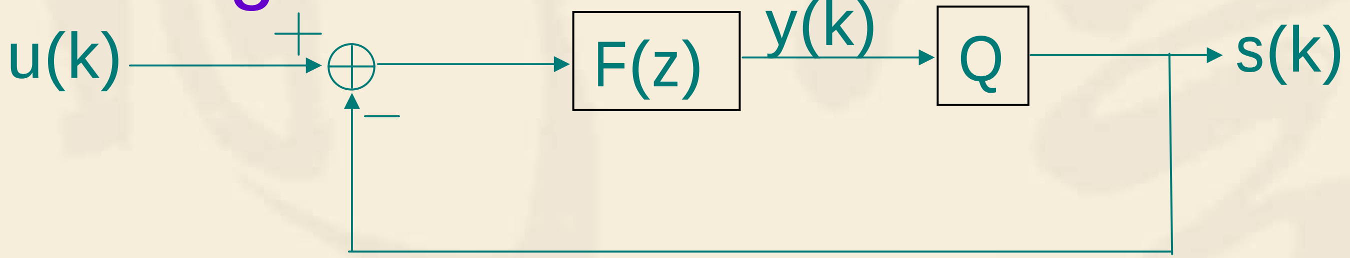
$$-1 \le \frac{\sum_{j=0}^{M-1} s(\mathrm{mod}(i+j,M)) \cdot \cos\left(\left(\frac{M}{2} - j - 1\right) \cdot \theta\right)}{\sin\left(\frac{M \cdot \theta}{2}\right) \cdot \sin\theta} < 1$$

Sigma delta modulators

- What is sigma delta modulators?
  - Sigma delta modulators are devices implementing sigma delta modulations and are widely used in analogue-to-digital conversions.
  - The input signals are first oversampled to obtain the inputs of the sigma delta modulators. The loop filters are to separate the quantization noises and the input signals so that very high signal-to-noise ratios could be achieved at very coarse quantization schemes.

Sigma delta modulators

- Block diagram



$$\frac{S(z)}{U(z)} = \frac{F(z)}{1+F(z)}$$

$$\frac{S(z)}{N(z)} = \frac{1}{1+F(z)}$$

Sigma delta modulators

- Nonlinear state space dynamical model:

$$\mathrm{F}(z) = \frac{2\cos\theta\, z^{-1} - z^{-2}}{1 - 2\cos\theta\, z^{-1} + z^{-2}}$$

$$\Rightarrow y(k) - 2\cos\theta\, y(k-1) + y(k-2)$$

$$= 2\cos\theta\big(u(k-1) - Q(y(k-1))\big) - \big(y(k-2) - Q(y(k-2))\big)$$

$$\Rightarrow y(k) = 2\cos\theta\, y(k-1) - y(k-2) +$$

$$2\cos\theta\big(u(k-1) - Q(y(k-1))\big) - \big(y(k-2) - Q(y(k-2))\big)$$

$$\mathbf{u}(k) \equiv \begin{bmatrix} u(k-2) & u(k-1) \end{bmatrix}^T$$

$$\mathbf{x}(k) \equiv \begin{bmatrix} y(k-2) & y(k-1) \end{bmatrix}^T$$

$$\mathbf{s}(k) \equiv \begin{bmatrix} Q(y(k-2)) & Q(y(k-1)) \end{bmatrix}^T$$

Sigma delta modulators

$$\mathbf{A} \equiv \begin{bmatrix} 0 & 1 \\ -1 & 2\cos\theta \end{bmatrix} \quad \mathbf{B} \equiv \begin{bmatrix} 0 & 0 \\ -1 & 2\cos\theta \end{bmatrix}$$

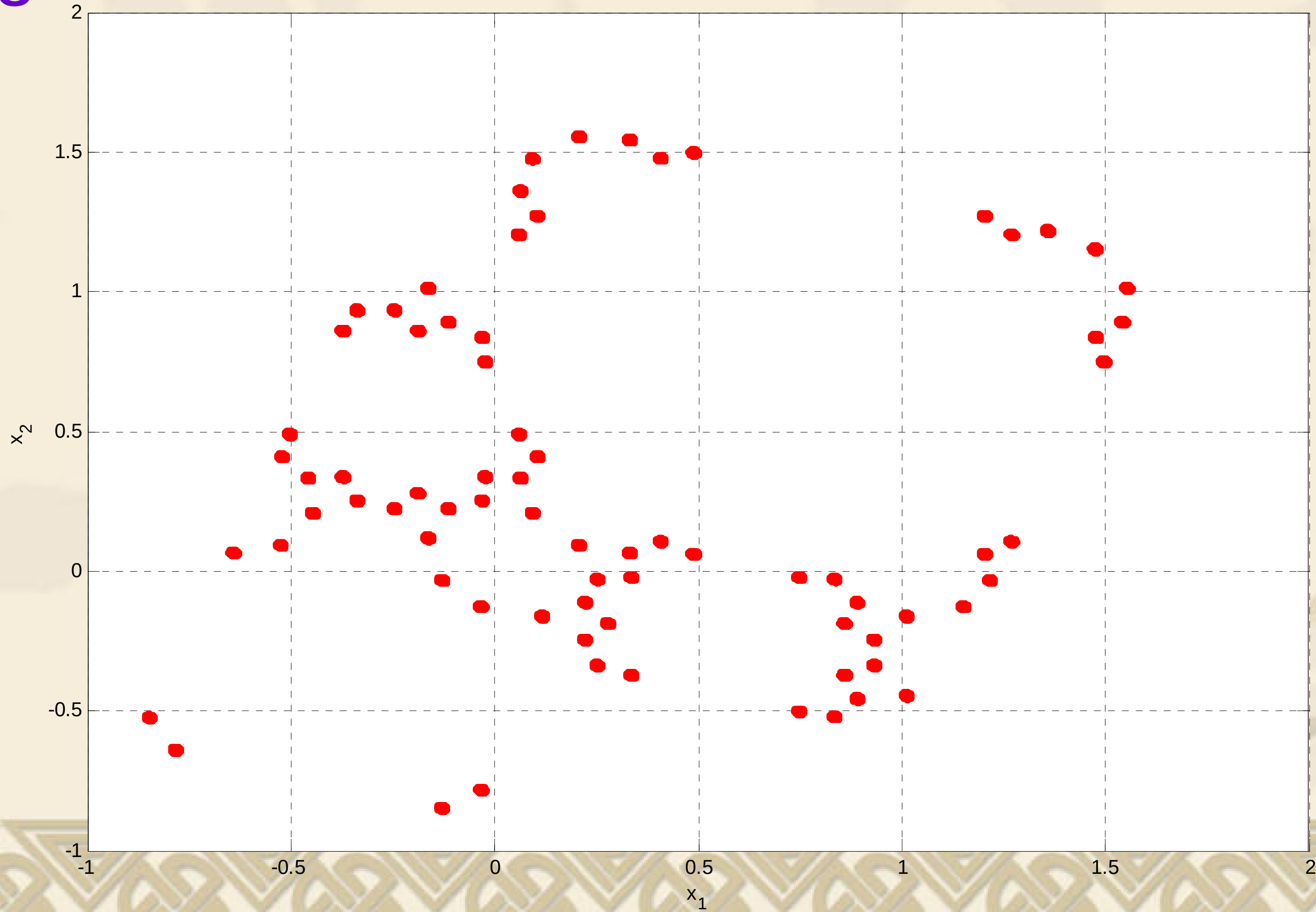
$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}(\mathbf{u}(k) - \mathbf{s}(k))$$

$$Q(y) \equiv \begin{cases} 1 & y \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

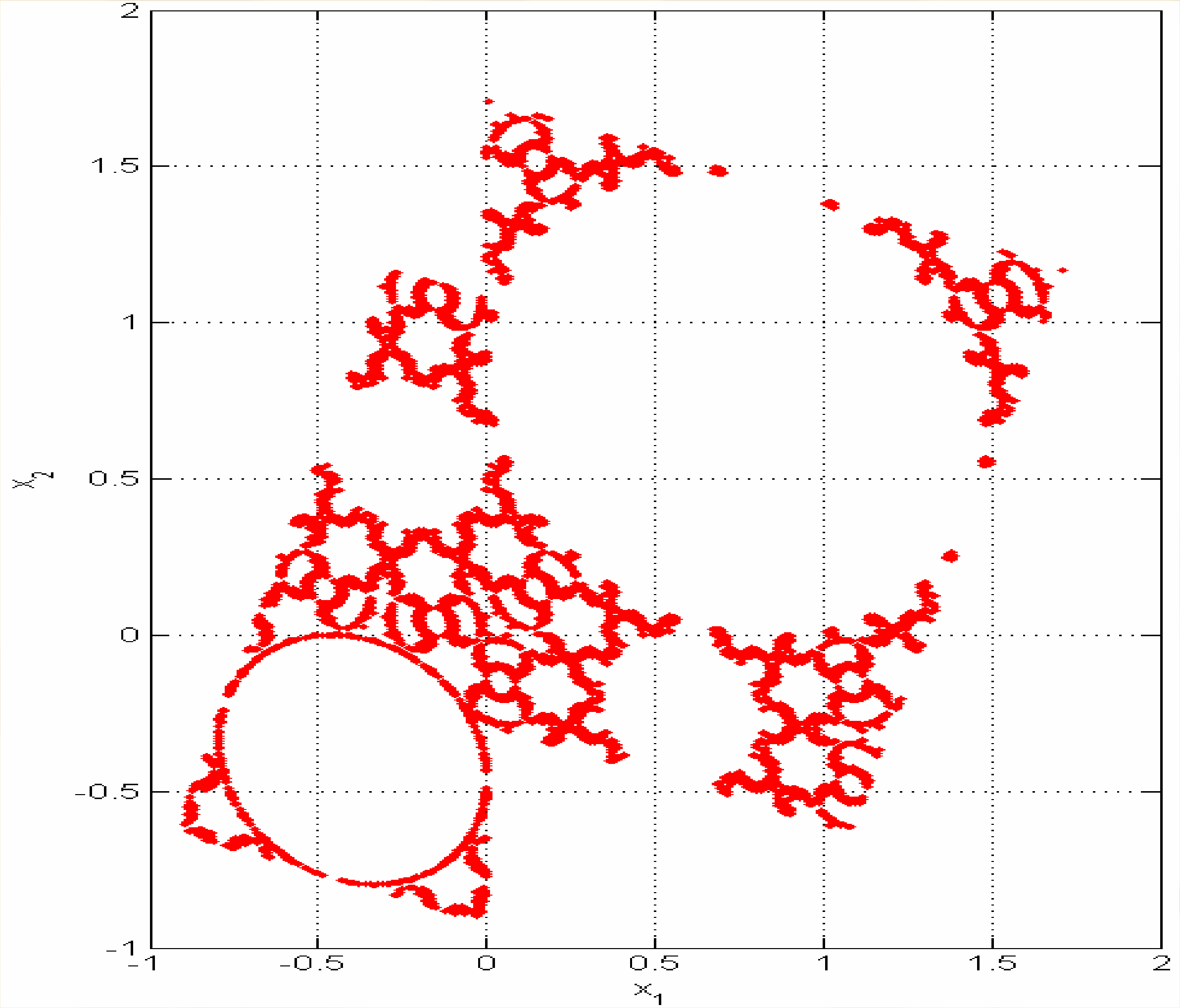
- There are only finite numbers of possibilities of **s**(k). Hence, **s**(k) can be viewed as symbol and **s**(k) is called a symbolic sequence.

Sigma delta modulators

- Second order marginally stable bandpass sigma delta modulators

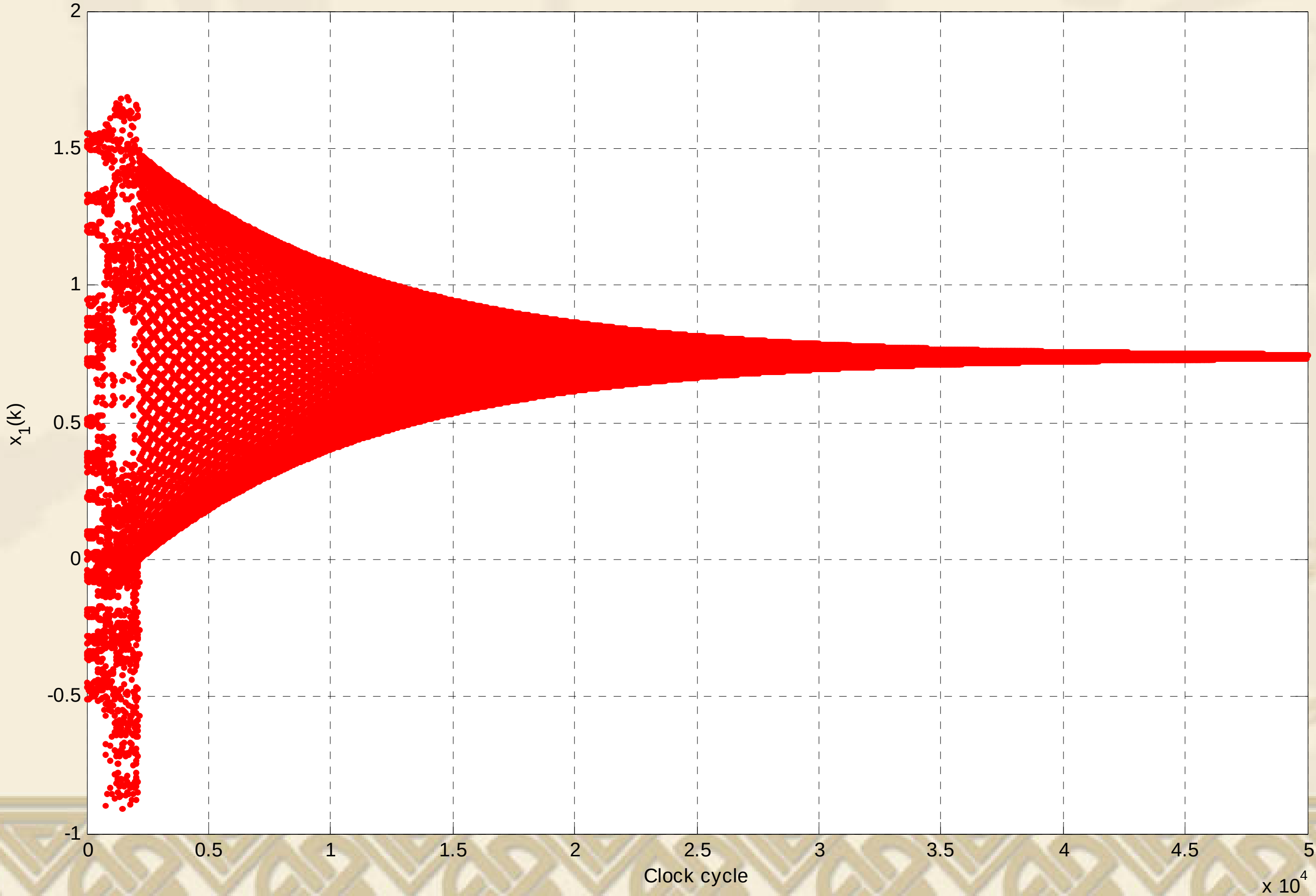


Sigma delta modulators

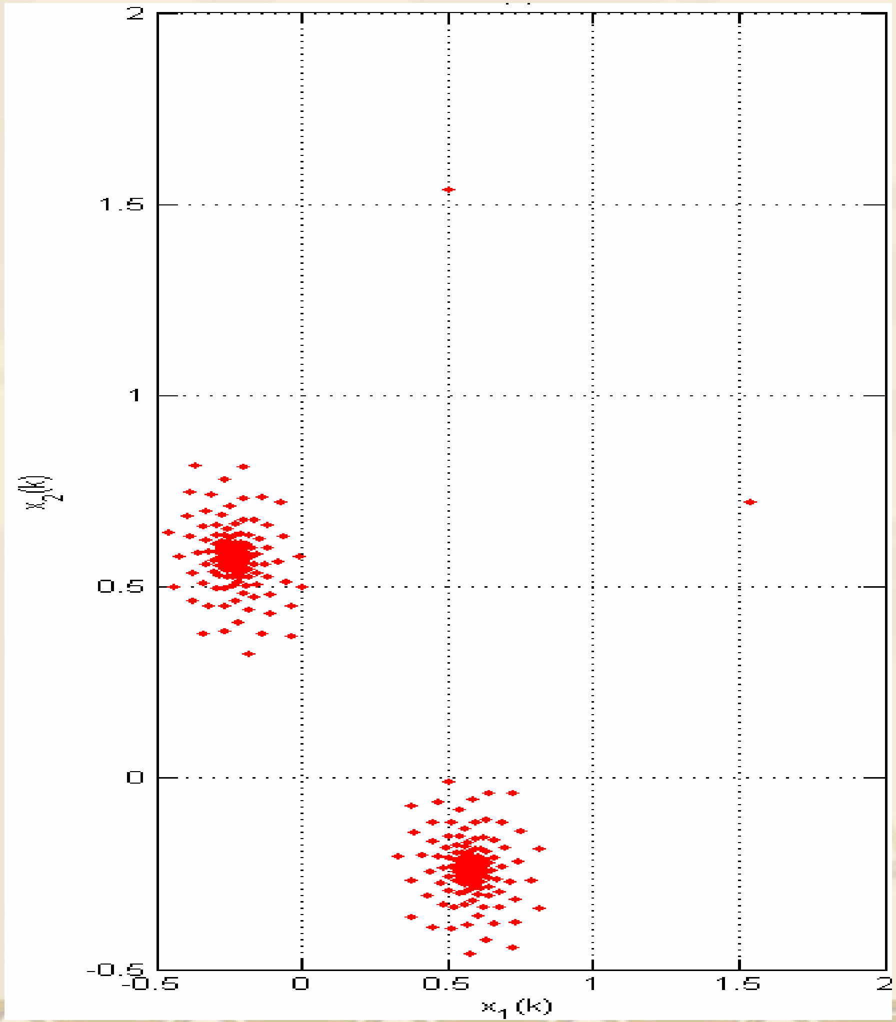


Sigma delta modulators

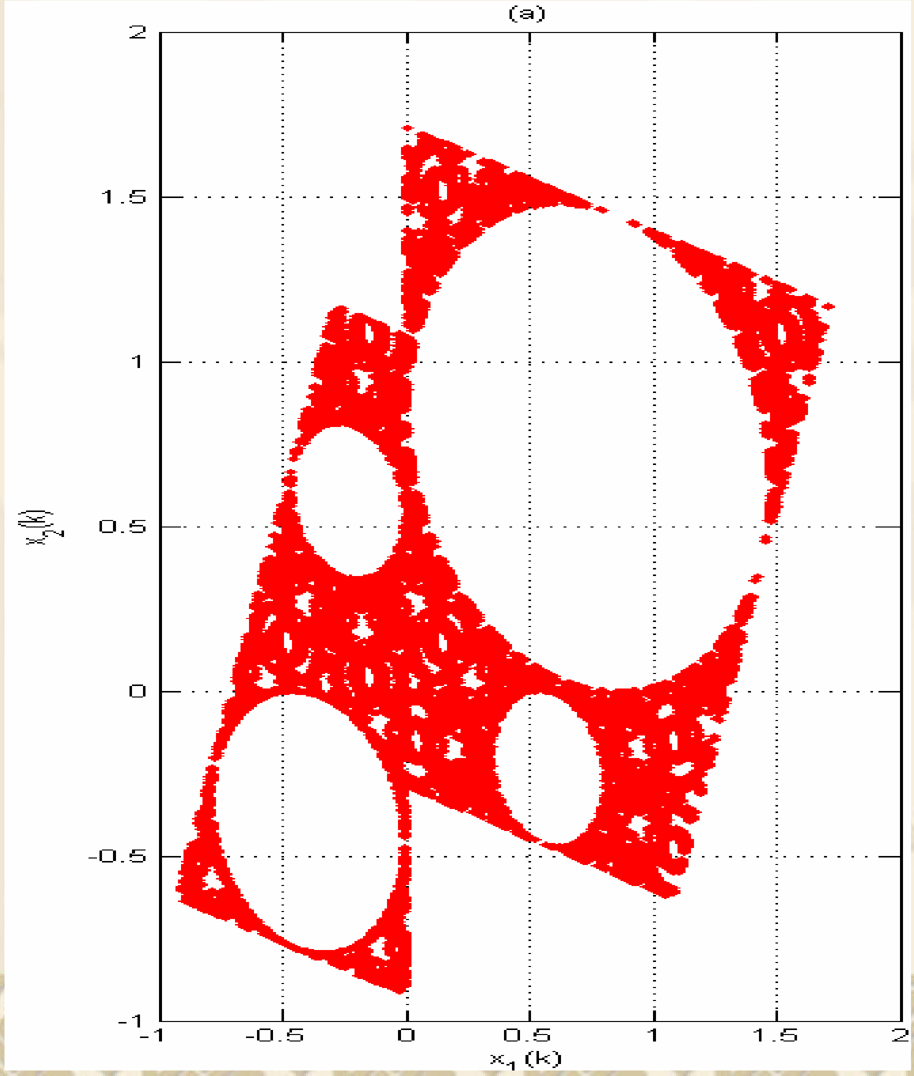
- Second order strictly stable bandpass sigma delta modulators



Sigma delta modulators

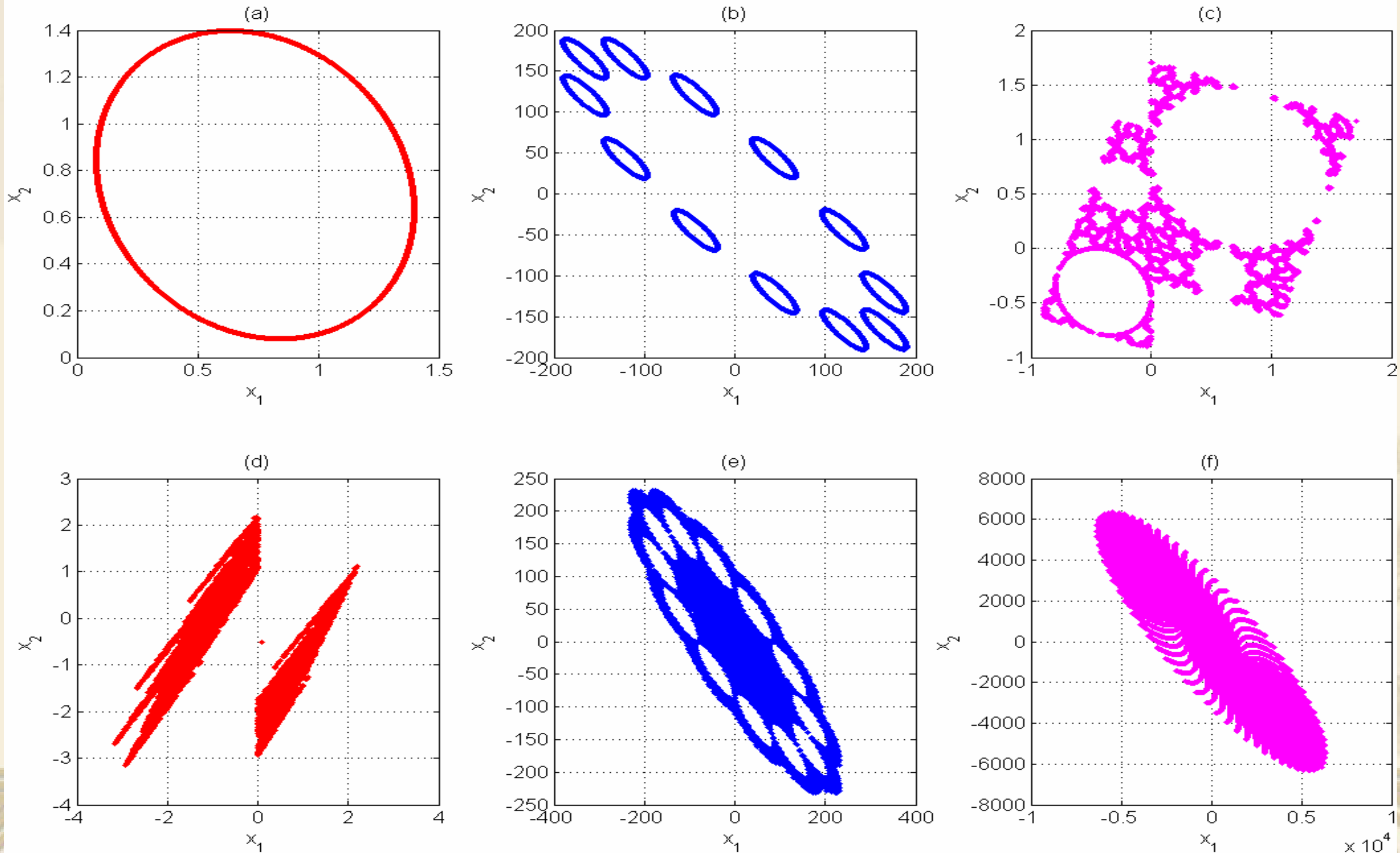


Sigma delta modulators



Sigma delta modulators

- General second order bandpass sigma delta modulators



Sigma delta modulators

- Admissibility of periodic sequence

$$\zeta \equiv [s_1(0), \quad s_2(0), \quad \cdots \quad s_1(M-1), \quad s_2(M-1)]^T$$

$$\mathbf{D}_j \equiv \begin{bmatrix} d \sin j\theta & c \sin j\theta \\ d \sin(j+1)\theta & c \sin(j+1)\theta \end{bmatrix}$$

$$\mathbf{D} \equiv \begin{bmatrix} \mathbf{D}_{M-1} & \mathbf{D}_{M-2} & \cdots & \mathbf{D}_1 & \mathbf{D}_0 \\ \mathbf{D}_0 & \mathbf{D}_{M-1} & \ddots & & \mathbf{D}_1 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \mathbf{D}_{M-1} & \mathbf{D}_{M-2} \\ \mathbf{D}_{M-2} & \cdots & \cdots & \mathbf{D}_0 & \mathbf{D}_{M-1} \end{bmatrix}$$

Sigma delta modulators

$$\mathbf{K} \equiv diag\left(\left(\mathbf{I} - \mathbf{A}^M\right)^{-1}, \quad \cdots, \quad \left(\mathbf{I} - \mathbf{A}^M\right)^{-1}\right)$$

$$\boldsymbol{\upsilon} \equiv \left[\frac{(d+c)\bar{u}}{\sin\theta}\sum_{j=0}^{M-1}\sin j\theta, \quad \frac{(d+c)\bar{u}}{\sin\theta}\sum_{j=1}^{M}\sin j\theta\right]$$

$$\boldsymbol{\tau} \equiv \left[\boldsymbol{\upsilon}, \quad \cdots, \quad \boldsymbol{\upsilon}\right]^T$$

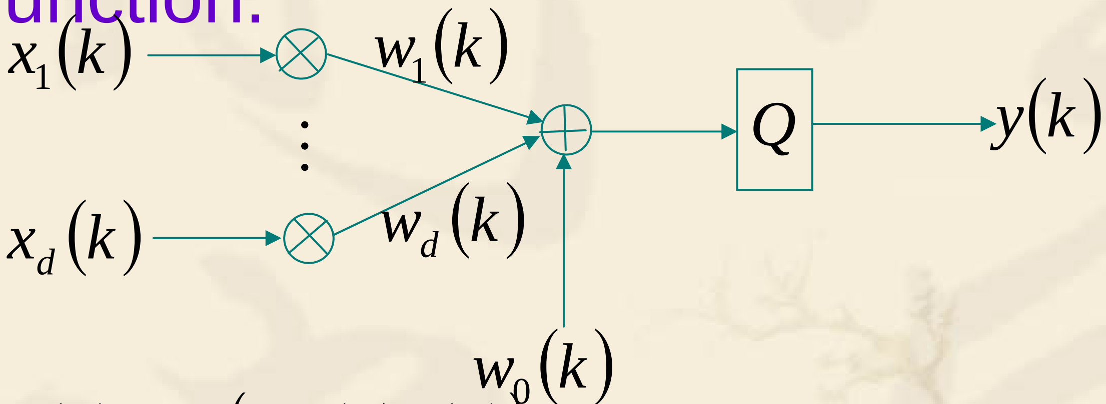
$$\mathbf{s} \equiv \left(\mathbf{s}(0), \mathbf{s}(1), \cdots, \mathbf{s}(M-1)\right)$$

- Σ be the admissible set of periodic output sequences with period M

$$\Sigma = \boxed{\left\{\mathbf{s} : \mathbf{Q}\left(\mathbf{K}\left(\mathbf{D}\zeta + \boldsymbol{\tau}\right)\right) = \zeta\right\}}$$

Perceptron training algorithms

- What is a perceptron training algorithm?
  - A perceptron is a single neuron that employs a single bit quantization function as its activation function.



$$y(k) = Q\left(\mathbf{w}^T(k)\mathbf{x}(k)\right)$$

where
$$\mathbf{x}(k) \equiv [1, \quad x_1(k), \quad \cdots, \quad x_d(k)]^T$$
$$\mathbf{w}(k) \equiv [w_0(k), \quad w_1(k), \quad \cdots, \quad w_d(k)]^T$$
$$Q(z) \equiv \begin{cases} 1 & z \ge 0 \\ -1 & z < 0 \end{cases}$$

Perceptron training algorithms

- What is a perceptron training algorithm?
  - The weights of a perceptron is usually found by the perceptron training algorithm.

$$\mathbf{w}(k+1)=\mathbf{w}(k)+\frac{t(k)-y(k)}{2}\mathbf{x}(k)$$

    where t(k) is the desired output corresponding to **x**(k).
  - If **w**(k) converges, then the steady state value of **w**(k) could be employed as the weights of the perceptron.

Perceptron training algorithms

$$\mathbf{w}(k+1)=\mathbf{w}(k)+\frac{t(k)-y(k)}{2}\mathbf{x}(k)$$

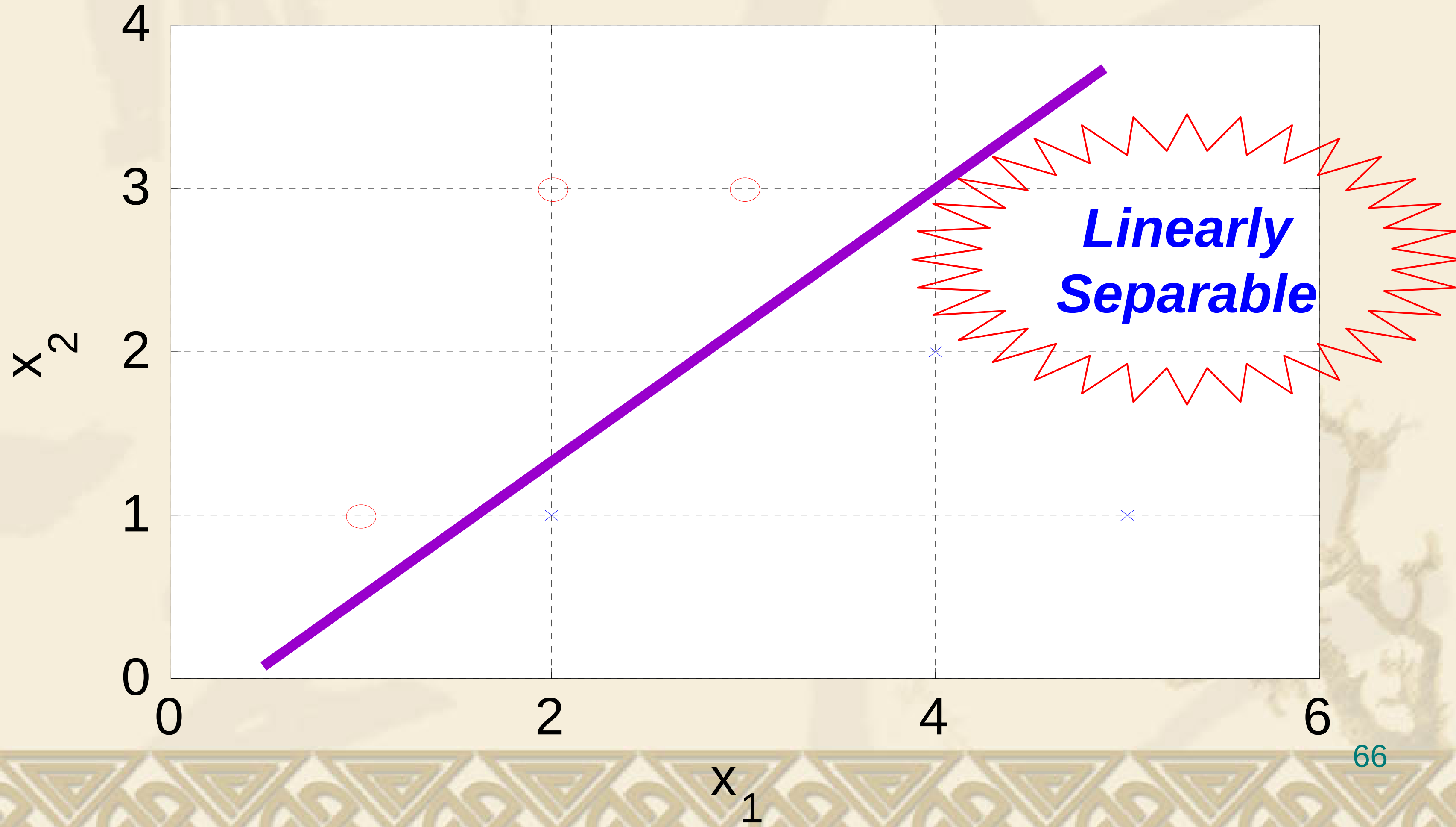
- Example 1

Iteration index k	$\mathbf{w}_{old}(k)$	$\mathbf{x}(k)$	$t(k)$	$y(k)$	$\mathbf{w}_{new}(k)$
0	$[0\ 0\ 0]^T$	$[1\ 5\ 1]^T$	-1	1	$[-1\ -5\ -1]^T$
1	$[-1\ -5\ -1]^T$	$[1\ 2\ 1]^T$	-1	-1	$[-1\ -5\ -1]^T$
2	$[-1\ -5\ -1]^T$	$[1\ 1\ 1]^T$	1	-1	$[0\ -4\ 0]^T$
3	$[0\ -4\ 0]^T$	$[1\ 3\ 3]^T$	1	-1	$[1\ -1\ 3]^T$
4	$[1\ -1\ 3]^T$	$[1\ 4\ 2]^T$	-1	1	$[0\ -5\ 1]^T$
5	$[0\ -5\ 1]^T$	$[1\ 2\ 3]^T$	1	-1	$[1\ -3\ 4]^T$
6	$[1\ -3\ 4]^T$	$[1\ 5\ 1]^T$	-1	-1	$[1\ -3\ 4]^T$
7	$[1\ -3\ 4]^T$	$[1\ 2\ 1]^T$	-1	-1	$[1\ -3\ 4]^T$
8	$[1\ -3\ 4]^T$	$[1\ 1\ 1]^T$	1	1	$[1\ -3\ 4]^T$
9	$[1\ -3\ 4]^T$	$[1\ 3\ 3]^T$	1	1	$[1\ -3\ 4]^T$
10	$[1\ -3\ 4]^T$	$[1\ 4\ 2]^T$	-1	-1	$[1\ -3\ 4]^T$
11	$[1\ -3\ 4]^T$	$[1\ 2\ 3]^T$	1	1	$[1\ -3\ 4]^T$

Converge

Perceptron training algorithms

- Example 1



Perceptron training algorithms

- Example 2

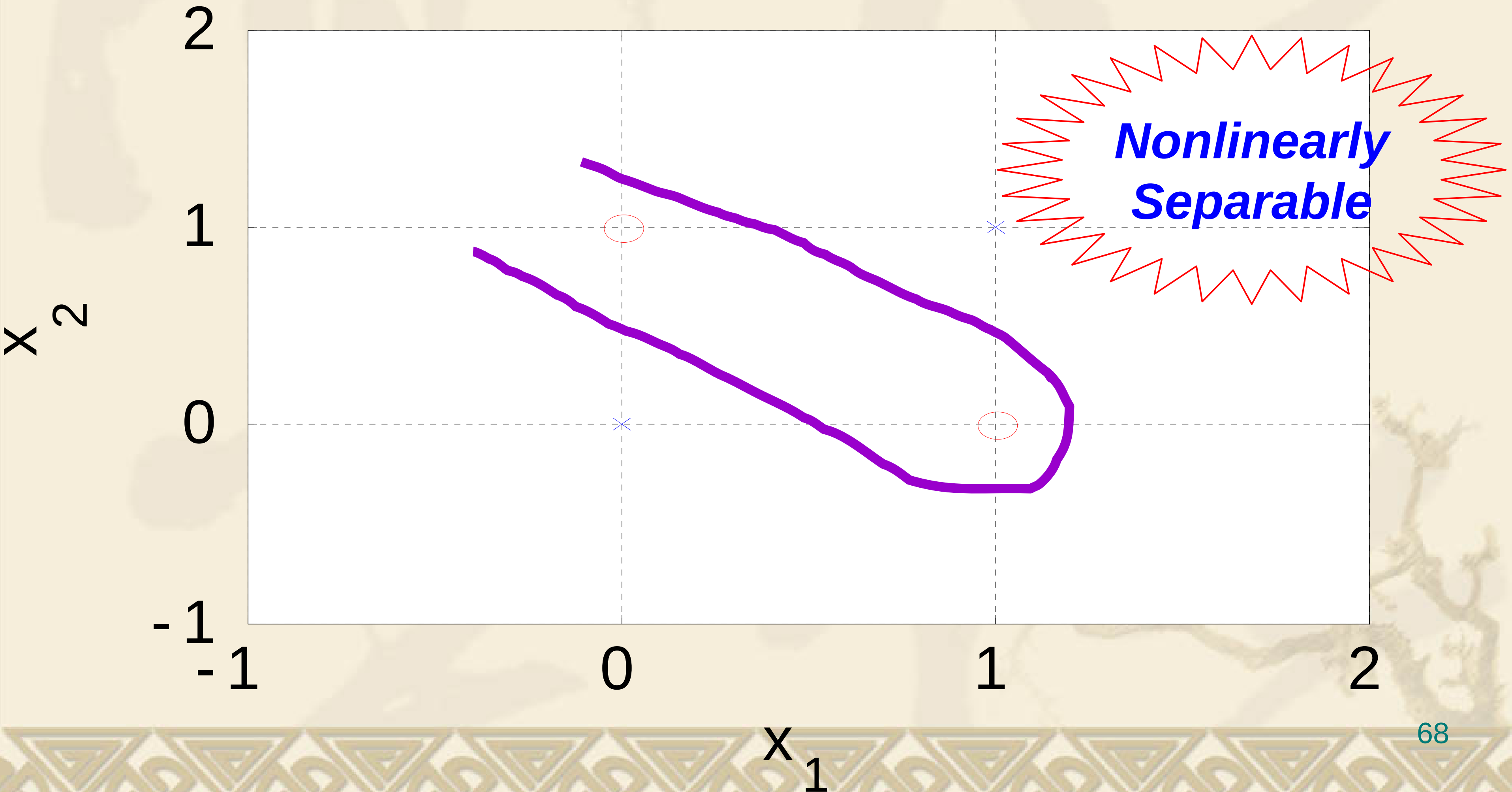
$$\mathbf{w}(k+1)=\mathbf{w}(k)+\frac{t(k)-y(k)}{2}\mathbf{x}(k)$$

Iteration index k	$\mathbf{w}_{\mathbf{old}}(k)$	$\mathbf{x}(k)$	$t(k)$	$y(k)$	$\mathbf{w}_{\mathbf{new}}(k)$
0	$[0\ 0\ 0]^T$	$[1\ 0\ 0]^T$	-1	1	$[-1\ 0\ 0]^T$
1	$[-1\ 0\ 0]^T$	$[1\ 0\ 1]^T$	1	-1	$[0\ 0\ 1]^T$
2	$[0\ 0\ 1]^T$	$[1\ 1\ 0]^T$	1	1	$[0\ 0\ 1]^T$
3	$[0\ 0\ 1]^T$	$[1\ 1\ 1]^T$	-1	1	$[-1\ -1\ 0]^T$
4	$[-1\ -1\ 0]^T$	$[1\ 0\ 0]^T$	-1	-1	$[-1\ -1\ 0]^T$
5	$[-1\ -1\ 0]^T$	$[1\ 0\ 1]^T$	1	-1	$[0\ -1\ 1]^T$
6	$[0\ -1\ 1]^T$	$[1\ 1\ 0]^T$	1	-1	$[1\ 0\ 1]^T$
7	$[1\ 0\ 1]^T$	$[1\ 1\ 1]^T$	-1	1	$[0\ -1\ 0]^T$
8	$[0\ -1\ 0]^T$	$[1\ 0\ 0]^T$	-1	1	$[-1\ -1\ 0]^T$
9	$[-1\ -1\ 0]^T$	$[1\ 0\ 1]^T$	1	-1	$[0\ -1\ 1]^T$
10	$[0\ -1\ 1]^T$	$[1\ 1\ 0]^T$	1	-1	$[1\ 0\ 1]^T$
11	$[1\ 0\ 1]^T$	$[1\ 1\ 1]^T$	-1	1	$[0\ -1\ 0]^T$

Limit Cycle

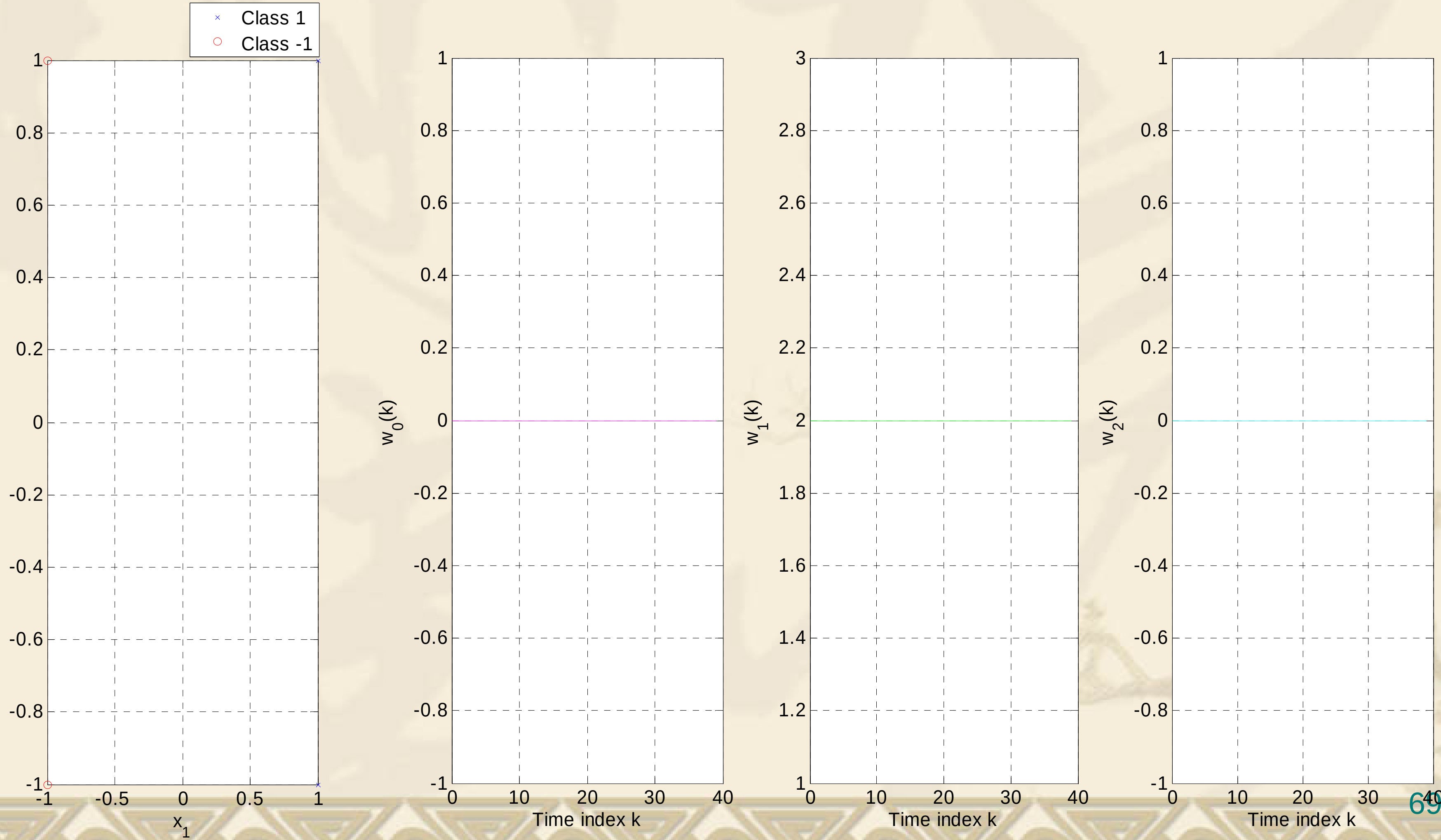
Perceptron training algorithms

- Example 2



Perceptron training algorithms

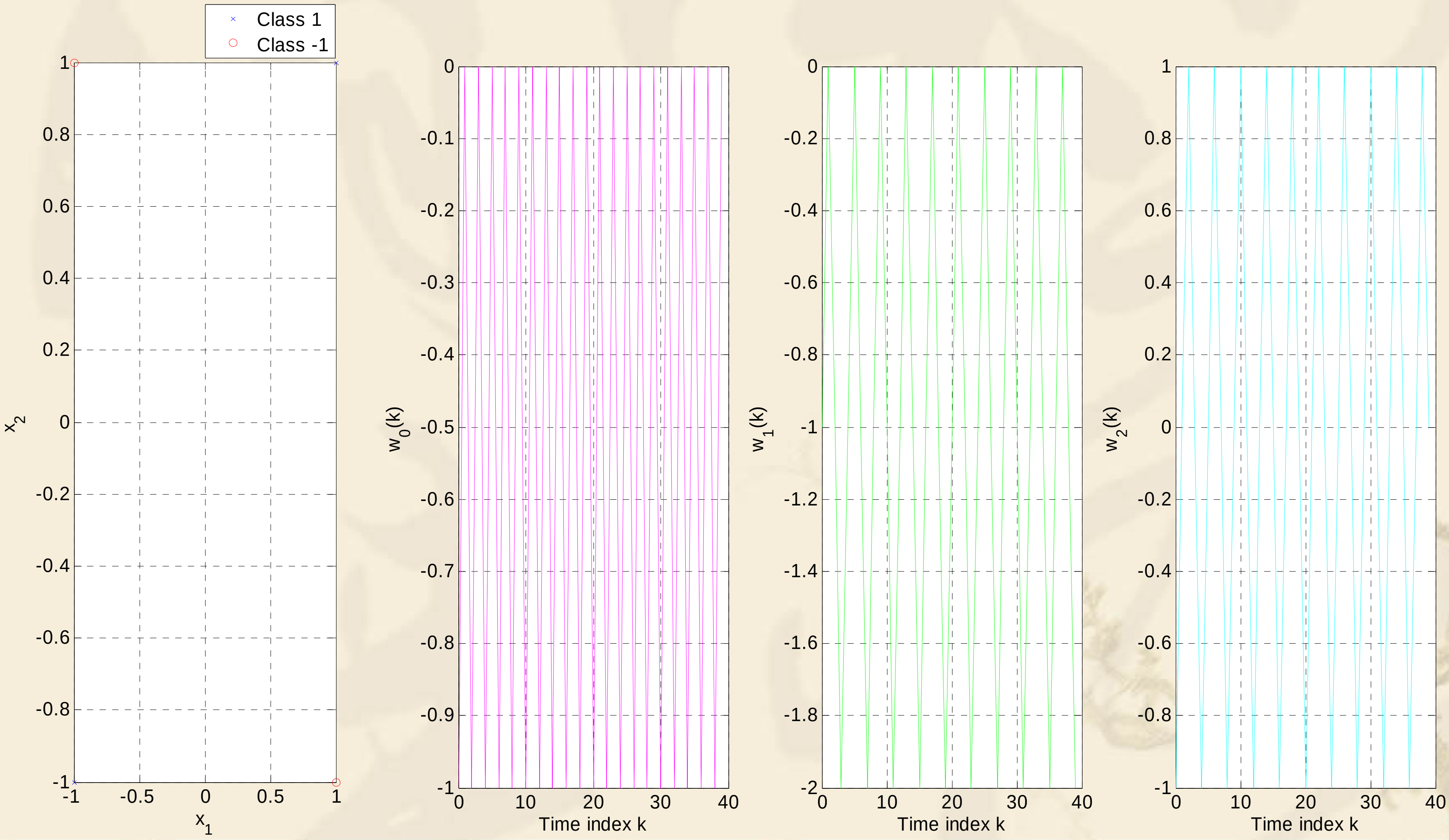
- Example 3



Perceptron training algorithms

- Example 4
  - The set of training feature vectors $\left\{\begin{bmatrix}1\\1\\1\end{bmatrix}, \begin{bmatrix}1\\1\\-1\end{bmatrix}, \begin{bmatrix}1\\-1\\-1\end{bmatrix}, \begin{bmatrix}1\\-1\\1\end{bmatrix}\right\}$
  - Desirable output $\{1,-1,1,-1\}$
  - Initial weight $\mathbf{w}(0) = [-1, \quad -1, \quad -1]^T$
  - Result: $\mathbf{w}(k) = [0, \quad 2, \quad 0]^T \quad \forall k \in Z$
  - Set of weights of the perceptron: $\left\{\begin{bmatrix}-1\\-1\\-1\end{bmatrix}, \begin{bmatrix}0\\0\\0\end{bmatrix}, \begin{bmatrix}-1\\-1\\1\end{bmatrix}, \begin{bmatrix}0\\-2\\0\end{bmatrix}\right\}$
  - Invariant set of weights of the perceptron: $\wp = \left\{[-1, \quad -1, \quad -1]^T\right\}$
  - Invariant map: $\Im^F : \left\{[-1, \quad -1, \quad -1]^T\right\} \to \left\{[-1, \quad -1, \quad -1]^T\right\}$
  - Note: $\tilde{\Im}^F : \Re^3 \to \Re^3$ is not bijective because $\|\mathbf{x}(k)\|^2 > \left|\mathbf{w}^T(k)\mathbf{x}(k)\right|$ $\forall k \in Z$

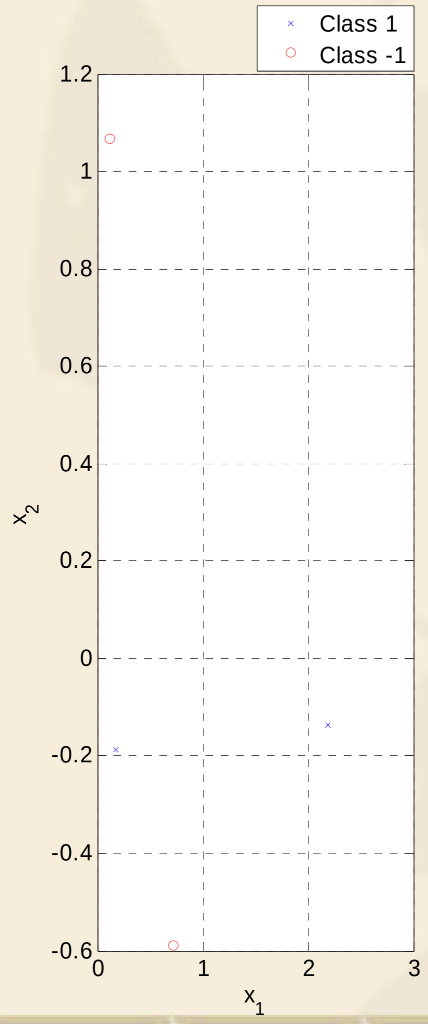
Perceptron training algorithms

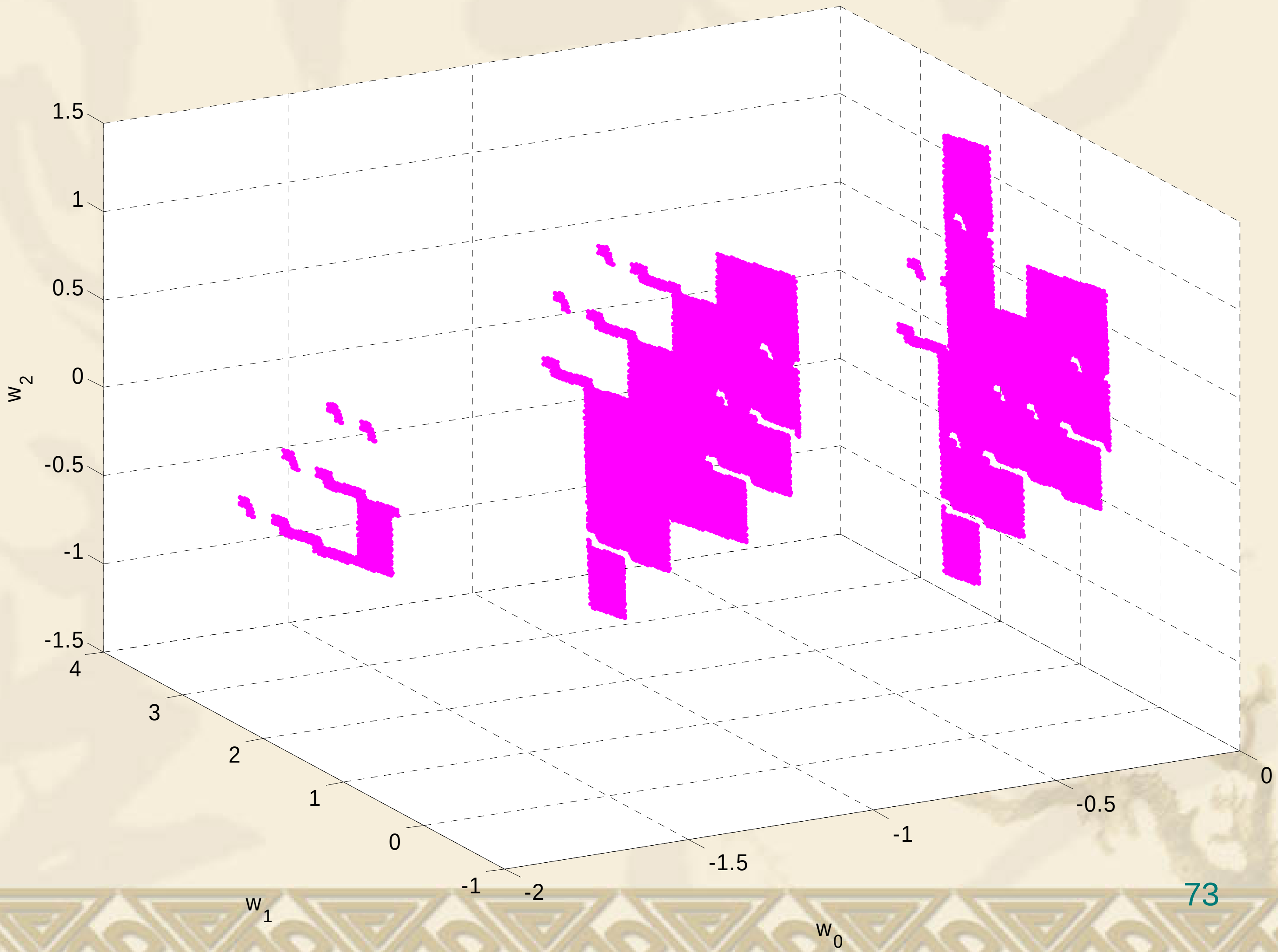


Perceptron training algorithms

- Example 5
  - The set of training feature vectors $\left\{ \begin{bmatrix} 1 \\ 0.1746 \\ -0.1867 \end{bmatrix}, \begin{bmatrix} 1 \\ 0.7258 \\ -0.5883 \end{bmatrix}, \begin{bmatrix} 1 \\ 2.1832 \\ -0.1364 \end{bmatrix}, \begin{bmatrix} 1 \\ 0.1139 \\ 1.0668 \end{bmatrix} \right\}$
  - Desirable output $\{1, -1, 1, -1\}$
  - Initial weight $\mathbf{w}(0) = [-1, \quad 0.7923, \quad -0.2133]^T$
  - Invariant set of the weights of the perceptron consists of three hyperplanes.
  - Note: $\tilde{\Im}^F : \Re^3 \to \Re^3$ is not bijective because $\exists k \in Z$ $\|\mathbf{x}(k)\|^2 > \left|\mathbf{w}^T(k)\mathbf{x}(k)\right|$

Perceptron training algorithms





Conclusions

- Many digital signal processing systems are symbolic dynamical systems.
- These symbolic dynamical systems could exhibit fractal, chaotic and divergent behaviors.
- By investigating the properties of these symbolic dynamical systems, unwanted behaviors could be avoided.

Q&A Session

Thank you!

Let me think…