Modelling & Simulation Mathematical Models of Systems

Štefan Korečko
Department of Computers and Informatics
Faculty of Electrical Engineering and Informatics
Technical University of Košice
Slovak Republic

stefan.korecko@tuke.sk

2023

Contents

- Input-output modelling
- State space modelling
- Taxonomy of models (and systems)

Input-output modelling I

t - time (variable) $t_0 \leq t \leq t_f$

u - input variables $\{u_1(t), \ldots, u_p(t)\}$

- $\vec{u}(t) = \left[u_1(t), \ldots, u_p(t)\right]^T$ (vector form)

y - output variables $\{y_1(t), \ldots, y_m(t)\}$

- $\vec{y}(t) = [y_1(t), \ldots, y_m(t)]^T$
- suppressed output variables
  - not associated with either the input or the output

Input-output modelling II

- $t_0 \leq t \leq t_f$
- $\vec{u}(t) = \left[u_1(t), \dots, u_p(t)\right]^T$
- $\vec{y}(t) = [y_1(t), \dots, y_m(t)]^T$

g - function, defining mathematical relationship between the input and output

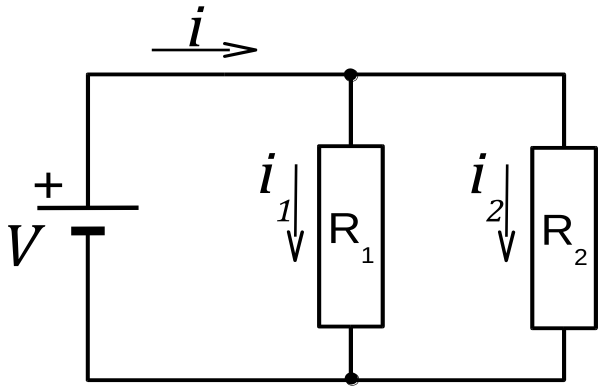
$$\vec{y}(t) = \vec{g}(\vec{u}(t)) = [g_1(u_1(t), \dots, u_p(t)) \dots g_m(u_1(t), \dots, u_p(t))]^T$$

$$y_i(t) = g_i\left(u_1(t), \dots, u_p(t)\right)$$

- in general, g can explicitly depend on t : $\vec{y}(t) = \vec{g}(\vec{u}(t), t)$

Input-output model example

- Current divider circuit



- From Kirchhoff's current law:
  - $i = i_1 + i_2$
- From Ohm's law:
  - $i_1 = \frac{V}{R_1}, \quad i_2 = \frac{V}{R_2}$
- Input-output model:
  - $u_1(t) = R_1, \quad y_1(t) = i = V\frac{R_1+R_2}{R_1 R_2}$

State space modelling I

- State of a system at time t_0
  - the information required at t_0 such that $\vec{y}(t)$, for all $t \geq t_0$, is uniquely determined from this information and from the input $\vec{u}(t), t \geq t_0$.
  - $\vec{x}$ - vector of state variables

$$\vec{x}(t) = [x_1(t), \dots, x_n(t)]^T$$

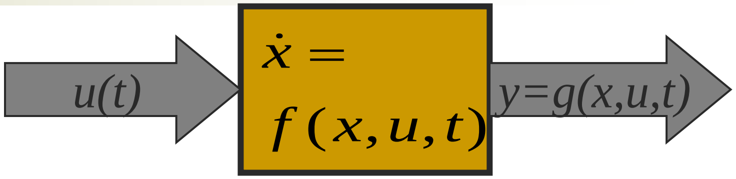
- X – State space of a system
  - the set of all possible values that the state may take.

State space modelling II

- State equations
  - the set of equations required to specify the state $\vec{x}(t)$ for all $t \geq t_0$ given $\vec{x}(t_0)$ and the function $\vec{u}(t)$, $t \geq t_0$.
  - usually differential equations of the form

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t), t)$$

State space modelling III



- State space model

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t), \vec{u}(t), t)$$
$$\vec{x}(t_0) = \vec{x}_0$$
$$\vec{y}(t) = \vec{g}(\vec{x}(t), \vec{u}(t), t)$$

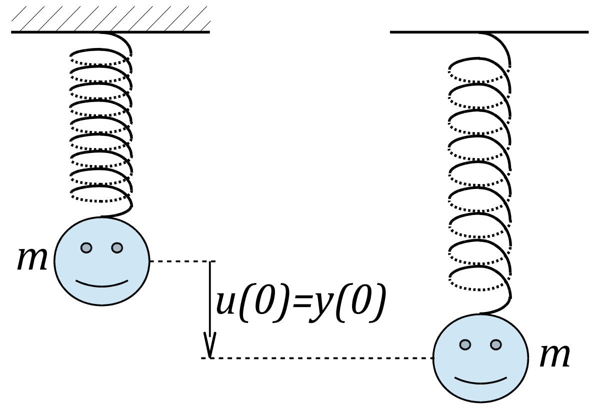
  - n state equations and initial conditions ($1 \leq i \leq n$):

$$\dot{x}_i(t) = f_i\big(x_1(t), \dots, x_n(t), u_1(t), \dots, u_p(t), t\big),$$
$$x_i(t_0) = x_{i0}$$

  - m output equations ($1 \leq j \leq m$):

$$y_j(t) = g_j\big(x_1(t), \dots, x_n(t), u_1(t), \dots, u_p(t), t\big)$$

State space model example



- Spring-mass system
  - a simple harmonic oscillator
- Hooke's law
  - $F = -ky$ or $m\ddot{y} = -ky$
- Initial conditions
  - $y(0) = u_0,\ \dot{y}(0) = 0$
- State space model ($x_1(t)$ instead of y)
  - $\dot{x_1}(t) = x_2(t)$
  - $\dot{x_2}(t) = -\frac{k}{m}x_1(t)$
  - $x_1(0) = u_0,\ \ x_2(0) = 0$
  - $y(t) = x_1(t)$

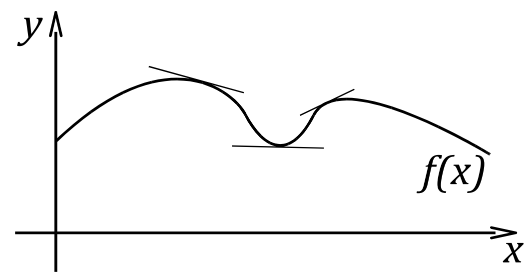
Note :: Derivative I

- a measure of how a function changes as its input changes

- differentiation
  - process of finding a derivative of a function
- antidifferentiation
  - = integration
  - reverse of differentiation

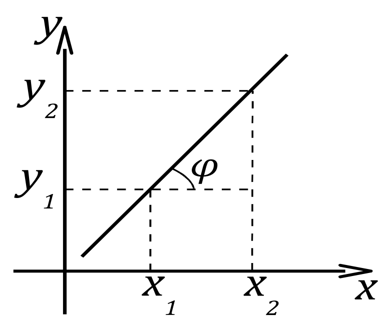
Note :: Derivative II

- derivative of a real function of a single variable (at a point)
  - slope of the tangent line to the graph of the function (at the point)



- slope (gradient) of a line *(smernica dotyčnice)*
  - = number that describes both the direction and the steepness of the line
  - usually denoted as m

$$m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1} = tan\ (\varphi)$$

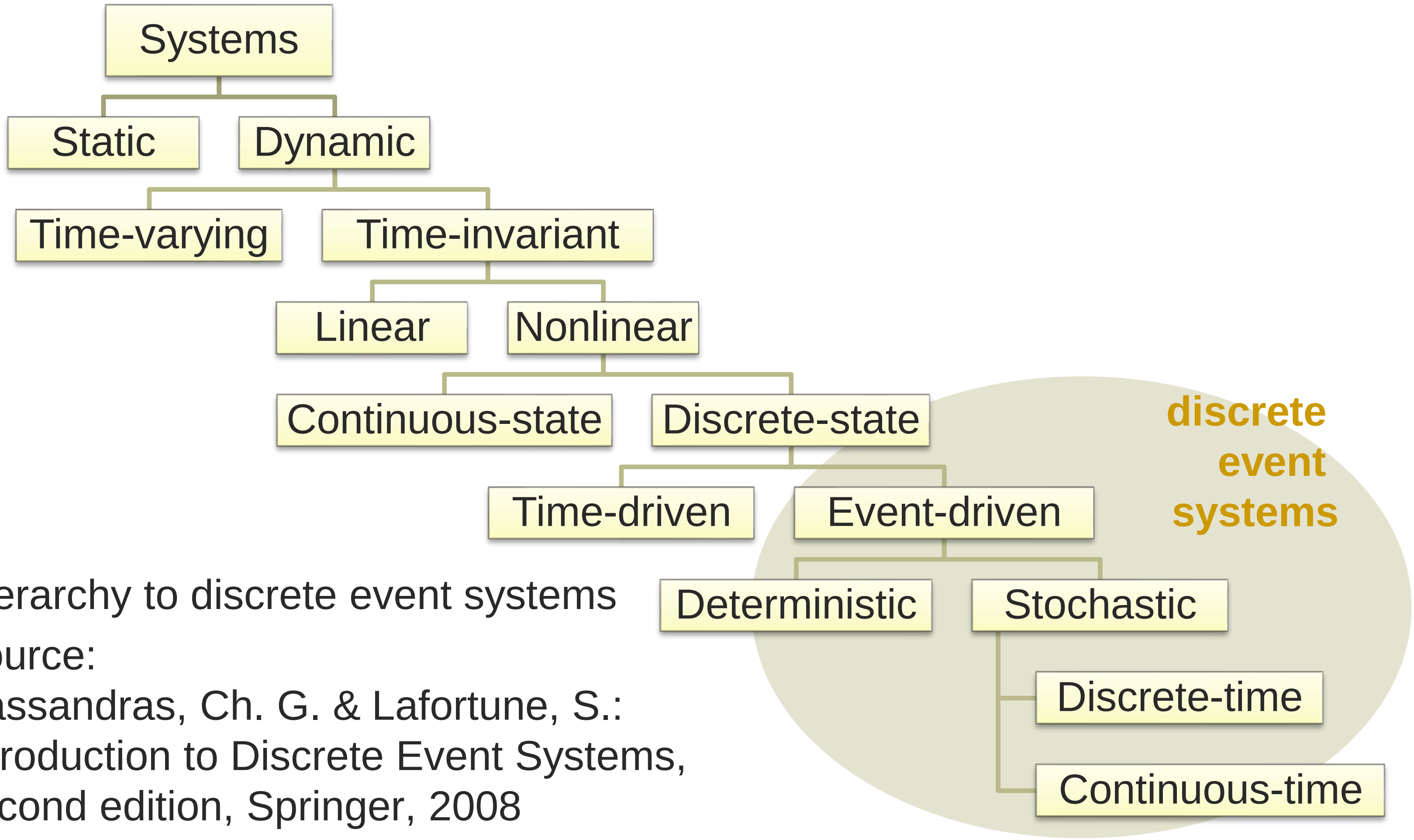


Note :: Derivative III

- Notations for derivative of $y = f(x)$
  - Lagrange's f' f'' f''' $f^{(n)}$
  - Leibniz's $\frac{dy}{dx}$ $\frac{df}{dx}(x)$ $\frac{d}{dx}f(x)$ $\frac{d^n y}{dx^n}$ $\frac{d^n f}{dx^n}(x)$ $\frac{d^n}{dx^n}f(x)$
  - Euler's $D_x y$ $D_x f(x)$ $D_x^n y$ $D_x^n f(x)$
  - Newton's $\dot{y}$ $\ddot{y}$
    - for time derivatives $y = f(t)$
- Definition

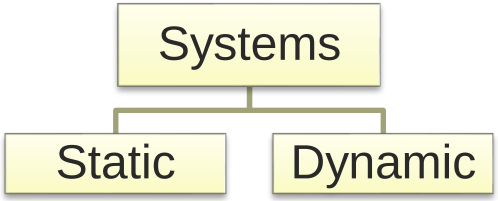
$$f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h} \qquad \dot{y}(t) = \lim_{\tau \to 0} \frac{f(t+\tau) - f(t)}{\tau}$$

Taxonomy of models and systems



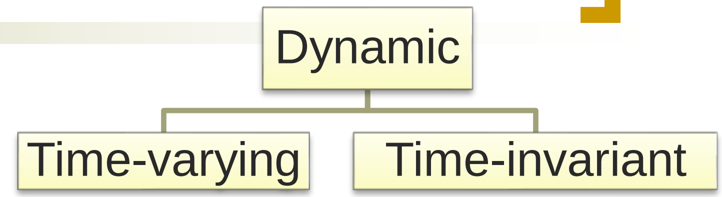
- Hierarchy to discrete event systems
- Source:
  Cassandras, Ch. G. & Lafortune, S.: Introduction to Discrete Event Systems, second edition, Springer, 2008

Static and Dynamic Systems



- Static
  - for all t the output $y(t)$ doesn't depend on past values of the input $u(t_b), t_b < t$.
  - the state is fixed $\dot{\vec{x}}(t) = \vec{0}$
- Dynamic
  - the output depends on some past values of the input.

Time-varying and Time-invariant Systems



- Time-invariant
  - the output is always the same when the same input is applied

$$\dot{\vec{x}}(t) = \vec{f}\left(\vec{x}(t), \vec{u}(t)\right)$$

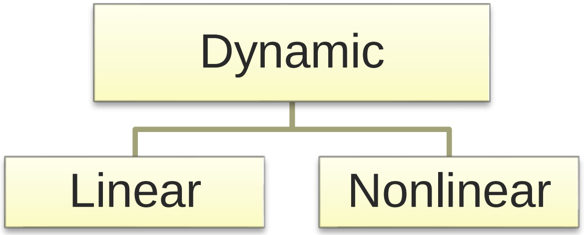
$$\vec{y}(t) = \vec{g}\left(\vec{x}(t), \vec{u}(t)\right)$$

- Time-varying
  - explicit time dependence

$$\dot{\vec{\mathrm{x}}}(t) = \vec{\mathrm{f}}(\vec{\mathrm{x}}(t), \vec{u}(t), \mathrm{t})$$

$$\vec{y}(t) = \vec{g}(\vec{\mathrm{x}}(t), \vec{u}(t), \mathrm{t})$$

Linear and non-linear systems



- Linear: $\vec{f}$ and $\vec{g}$ are linear
- Nonlinear: $\vec{f}$ or $\vec{g}$ is not linear
- The function $\vec{g}$ is linear if it satisfies the superposition principle

$$\vec{g}(a_1\overrightarrow{u_1} + a_2\overrightarrow{u_2}) = a_1\vec{g}(\overrightarrow{u_1}) + a_2\vec{g}(\overrightarrow{u_2})$$

- the state model of a linear system

$$\dot{\vec{x}}(t) = \mathrm{A}(t)\vec{x}(t) + \mathrm{B}(t)\vec{u}(t)$$
$$\vec{y}(t) = \mathrm{C}(t)\vec{x}(t) + \mathrm{D}(t)\vec{u}(t)$$

  - $\mathrm{A}(t)$ is an $n \times n$ matrix, $\mathrm{B}(t)$ is an $n \times p$ matrix, $\mathrm{C}(t)$ is an $m \times n$ matrix and $\mathrm{D}(t)$ is an $m \times p$ matrix.

Linear system example

$$\dot{\vec{x}}(t) = \mathrm{A}(t)\vec{x}(t) + \mathrm{B}(t)\vec{u}(t)$$
$$\vec{y}(t) = \mathrm{C}(t)\vec{x}(t) + \mathrm{D}(t)\vec{u}(t)$$

Spring – mass system

$$\dot{x}_1(t) = x_2(t)$$
$$\dot{x}_2(t) = -\frac{k}{m}x_1(t)$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\frac{k}{m} & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

$$A = \begin{bmatrix}0 & 1\\ -\frac{k}{m} & 0\end{bmatrix}, \qquad B = 0_{2,2}$$

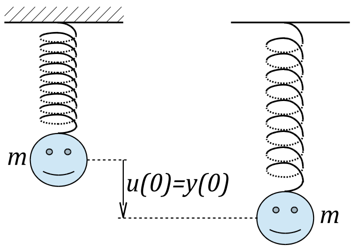
$$y(t) = x_1(t)$$

$$y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

$$C = \begin{bmatrix}1 & 0\end{bmatrix}, \qquad D = 0_{1,2}$$

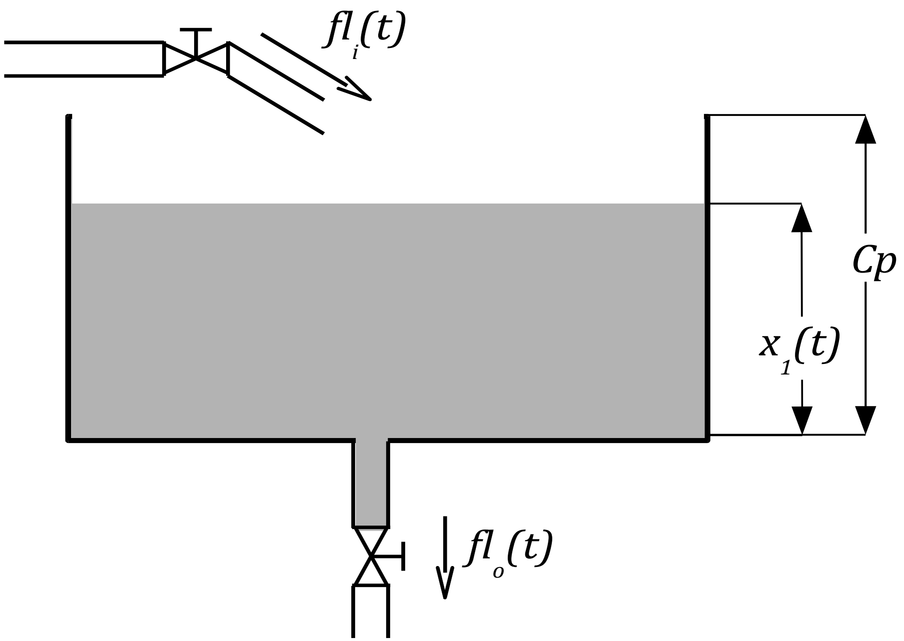
$$x_1(0) = u_0,$$
$$x_2(0) = 0$$

$$\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix} = \begin{bmatrix}u_0\\ 0\end{bmatrix}$$



Nonlinear system example

Flow system



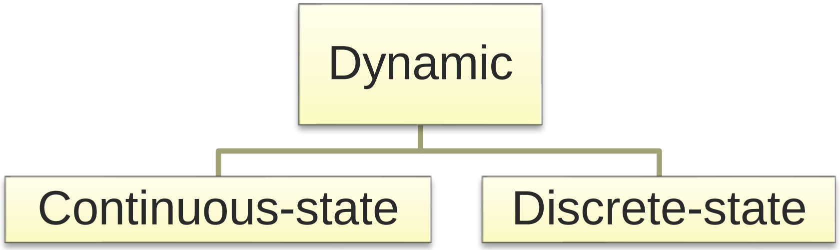
$u_1(t) = fl_i(t)$
$u_2(t) = fl_o(t)$

$y_1(t) = x_1(t)$

$x_1(0) = 0$

$$\dot{x}_1(t) = \begin{cases} 0 & (x_1(t) = 0 \wedge fl_i(t) \leq fl_o(t)) \vee \\ & \vee \ (x_1(t) = Cp \wedge fl_i(t) \geq fl_o(t)) \\ \mathrm{fl_i}(t) - fl_o(t) & otherwise \end{cases}$$

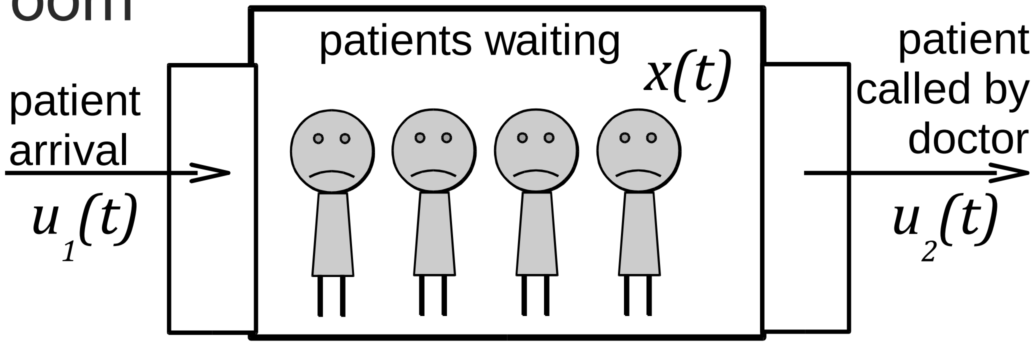
Continuous and Discrete-State



- Continuous-State
  - state variables can generally take on any real (or complex) value
- Discrete-State
  - state variables are elements of a discrete set (e.g., the non-negative integers)

Discrete-state system example

- Doctor's waiting room



$$u_1(t) = \begin{cases} 1 & \textit{if a patient arrives at time } t \\ 0 & \textit{otherwise} \end{cases}$$

$$u_2(t) = \begin{cases} 1 & \textit{if a patient is called at time } t \\ 0 & \textit{otherwise} \end{cases}$$

$$x(t+1) = \begin{cases} x(t)+1 & u_1(t) = 1 \wedge u_2(t) = 0 \\ x(t)-1 & u_1(t) = 0 \wedge u_2(t) = 1 \wedge x(t) > 0 \\ x(t) & \textit{otherwise} \end{cases}$$

Continuous and discrete-time

- Continuous-time
  - all input, state, and output variables are defined for all possible values of time
- Discrete-time
  - one or more of these variables are defined at discrete points in time only (usually as the result of some sampling process)
  - time is a **discrete** variable

- each real system is a continuous-time system
  - inputs and outputs can be observed at any time instant
  - their models can be discrete-time

Discrete-time systems I

- Good for models
  - where output and state variables can change only at exactly defined time instants
    - digital circuits
  - based on a finite set of data, recorded at certain time moments
    - a model is discrete-time because we cannot construct a continuous-time one

- Continuous-time systems become de-facto discrete-time systems when simulated on digital computers

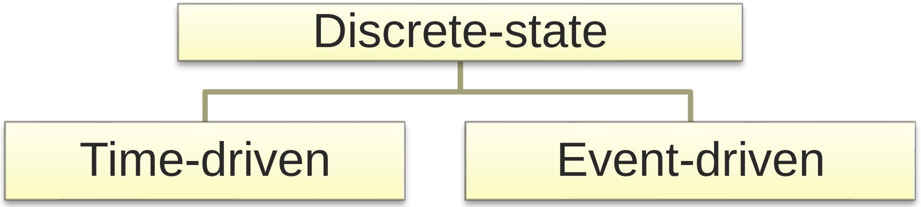
Discrete-time systems II

- Time = sequence of time values $t_0, t_1, \dots, t_k, \dots,$
  - $\forall i (i \geq 0): t_i < t_{i+1}$
- T – sampling interval $\forall i (i \geq 0): t_{i+1} = t_i + T$
- State space model uses *difference equations*

$$\vec{x}(k+1) = \vec{f}(\vec{x}(k), \vec{u}(k), k)$$
$$\vec{x}(0) = x_0$$
$$\vec{y}(k) = \vec{g}(\vec{x}(k), \vec{u}(k), k)$$

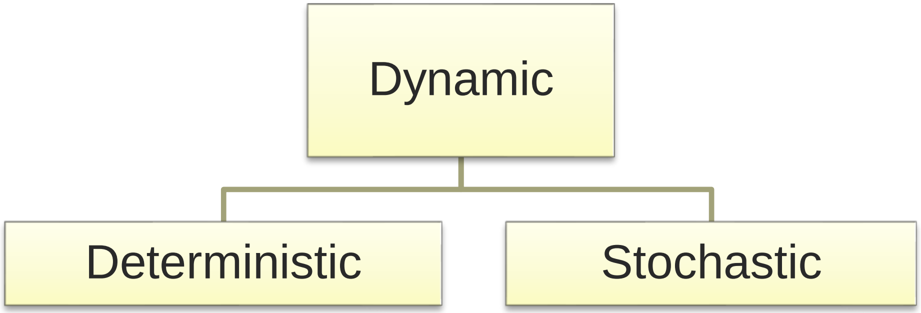
- k – integer variable, replaces t
  - number of intervals of the length T that passed since the initial time point t_0

Time-driven vs. Event-driven



- Time-driven
  - state continuously changes as time changes
  - time advance causes state change
- Event-driven
  - only an occurrence of an asynchronously generated discrete event forces instantaneous state transition.
  - event occurrence causes state change

Stochastic vs. Deterministic



- Stochastic
  - one or more of its output variables is a random variable
- Deterministic
  - no random variable

Discrete Event Systems