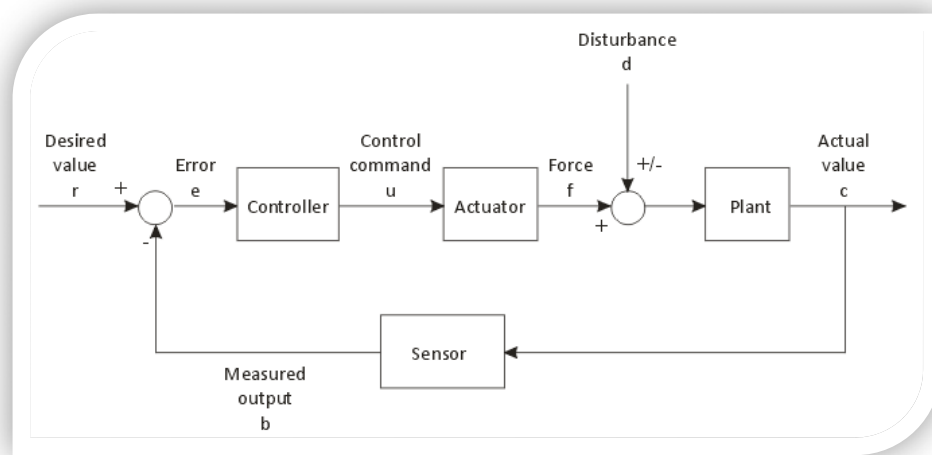# Mathematical Modelling Using Simulink

Control Systems Laboratory

Dr. Zaer Abo Hammour

1. Mathematical Model Definition

A **mathematical model** is a description of a system using mathematical concepts and language. The process of developing a mathematical model is termed **mathematical modeling**. Mathematical models are used in the natural sciences (such as physics, biology, earth science, meteorology) and engineering disciplines (such as computer science, artificial intelligence), as well as in the social sciences (such as economics, psychology, sociology, political science).

Physicists, engineers, statisticians, operations research analysts, and economists use mathematical models most extensively. A model may help to explain a system and to study the effects of different components, and to make predictions about behavior.

Mathematical models can take many forms, including but not limited to dynamical systems, statistical models, differential equations, or game theoretic models. These and other types of models can overlap, with a given model involving a variety of abstract structures. In general, mathematical models may include logical models. In many cases, the quality of a scientific field depends on how well the mathematical models developed on the theoretical side agree with results of repeatable experiments. Lack of agreement between theoretical mathematical models and experimental measurements often leads to important advances as better theories are developed.

2. Model Classifications in Mathematics

Mathematical models are usually composed of relationships and variables. Relationships can be described by operators, such as algebraic operators, functions, differential operators, etc. Variables are abstractions of system parameters of interest, that can be quantified. Several classification criteria can be used for mathematical models according to their structure:

- **Linear vs. nonlinear**: If all the operators in a mathematical model exhibit linearity, the resulting mathematical model is defined as linear. A model is considered to be nonlinear otherwise. The definition of linearity and nonlinearity is dependent on context, and linear models may have nonlinear expressions in them. For example, in a statistical linear model, it is assumed that a relationship is linear in the parameters, but it may be nonlinear in the predictor variables. Similarly, a differential equation is said to be linear if it can be written with linear differential operators, but it can still have nonlinear expressions in it. In a mathematical programming model, if the objective functions and constraints are represented entirely by linear equations, then the model is regarded as a linear model. If one or more of the objective functions or constraints are represented with a nonlinear equation, then the model is known as a nonlinear model. Nonlinearity, even in fairly simple systems, is often associated with phenomena such as chaos and irreversibility. Although there are exceptions, nonlinear systems and models tend to be more difficult to study than linear ones. A common approach to nonlinear problems is linearization, but this can be problematic if one is trying to study aspects such as irreversibility, which are strongly tied to nonlinearity.

- **Static vs. Dynamic:** A dynamic model accounts for time-dependent changes in the state of the system, while a static (or steady-state) model calculates the system in equilibrium, and thus is time-invariant. Dynamic models typically are represented by differential equations.

- **Explicit vs. Implicit:** If all of the input parameters of the overall model are known, and the output parameters can be calculated by a finite series of computations (known as linear programming, not to be confused with linearity as described above), the model is said to be explicit. But sometimes it is the output parameters which are known, and the corresponding inputs must be solved for by an iterative procedure, such as Newton's method (if the model is linear) or Broyden's method (if non-linear). For example, a jet engine's physical properties such as turbine and nozzle throat areas can be explicitly calculated given a design thermodynamic cycle (air and fuel flow rates, pressures, and temperatures) at a specific flight condition and power setting, but the engine's operating cycles at other flight conditions and power settings cannot be explicitly calculated from the constant physical properties.

- **Discrete vs. Continuous**: A discrete model treats objects as discrete, such as the particles in a molecular model or the states in a statistical model; while a continuous model represents the objects in a continuous manner, such as the velocity field of fluid in pipe flows, temperatures and stresses in a solid, and electric field that applies continuously over the entire model due to a point charge.

- **Deterministic vs. Probabilistic (stochastic):** A deterministic model is one in which every set of variable states is uniquely determined by parameters in the model and by sets of previous states of these variables; therefore, a deterministic model always performs the same way for a given set of initial conditions. Conversely, in a stochastic model—usually called a "statistical model"—randomness is present, and variable states are not described by unique values, but rather by probability distributions.

- **Deductive, Inductive, or Floating:** A deductive model is a logical structure based on a theory. An inductive model arises from empirical findings and generalization from them. The floating model rests on neither theory nor observation, but is merely the invocation of expected structure. Application of mathematics in social sciences outside of economics has been criticized for unfounded models. Application of catastrophe theory in science has been characterized as a floating model.

3. Introduction to Simulink

3.1 Simulink Description:

Simulink® is a block diagram environment for multidomain simulation and Model-Based Design. It supports system-level design, simulation, automatic code generation, and continuous test and verification of embedded systems. Simulink provides a graphical editor, customizable block libraries, and solvers for modeling and simulating dynamic systems. It is integrated with MATLAB®, enabling you to incorporate MATLAB algorithms into models and export simulation results to MATLAB for further analysis.

3.2 Simulink Features:

- Graphical editor for building and managing hierarchical block diagrams
- Libraries of predefined blocks for modeling continuous-time and discrete-time systems
- Simulation engine with fixed-step and variable-step ODE solvers
- Scopes and data displays for viewing simulation results
- Project and data management tools for managing model files and data
- Model analysis tools for refining model architecture and increasing simulation speed
- MATLAB Function block for importing MATLAB algorithms into models
- Legacy Code Tool for importing C and C++ code into models

3.3 Open the Simulink Library Browser:

You need MATLAB® running before you can open the Simulink® Library Browser.

1. In the MATLAB Command Window, enter **simulink**.
   A short delay occurs the first time you open the Simulink Library Browser. The figure 1 shows the Library Browser with the Out1 block selected in the Simulink/Commonly Used Blocks sub library. A block description that appears when you hover over it.

2. The Library Browser keeps a repository of all the libraries it shows. If your library has missing repository information, a notification bar appears above the Libraries pane when you refresh the Library Browser. To prevent this notification from appearing again, click on Fix in the notification bar and choose Resave libraries in SLX file format. This saves all libraries in .slx format with Enable LBRepository property set to on. You can also open the Simulink Library Browser from the MATLAB Toolstrip, by clicking the Simulink Library button. To keep the Library Browser above all other windows on your desktop, in the toolbar, select the Stay on topbutton .
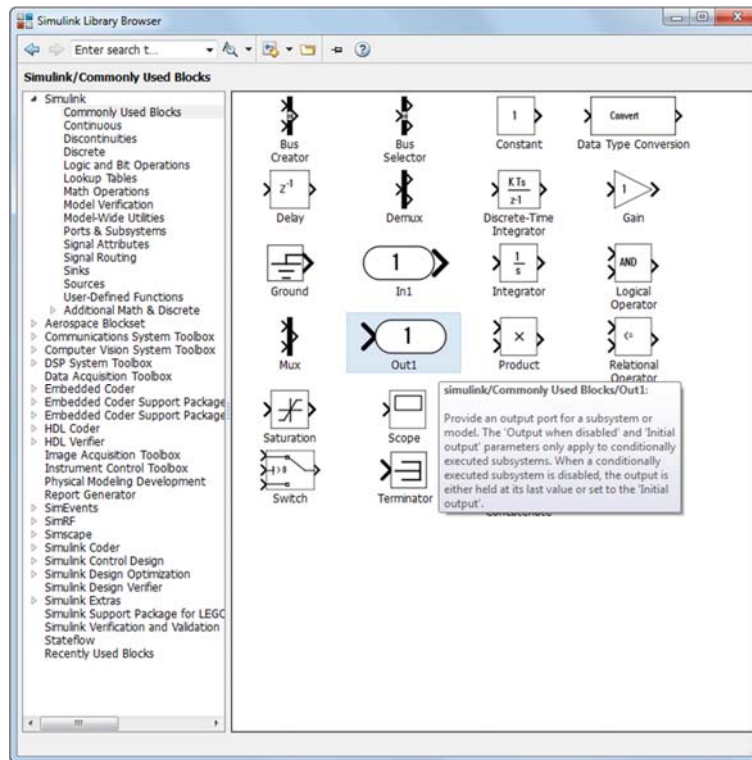
Figure 2.1: Simulink Library

3.4 Create a New Simulink Model

Create a new Simulink model from the Simulink Library Browser.

1. From the Simulink Library Browser toolbar, click the New Model button . An empty model diagram figure 2.2 opens in the Simulink Editor.
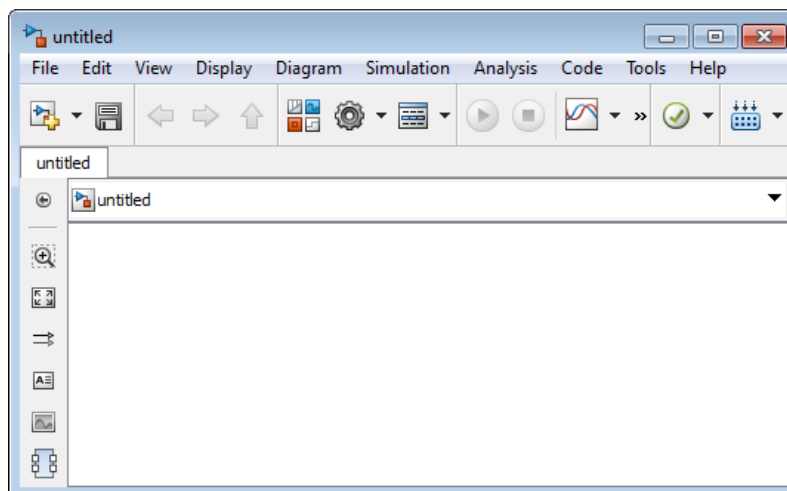

Figure 2.2: Simulink New Model

Example 1: Model the differential Equation (Time Domain)

$$\dot{x}(t) = 2x(t) + u(t) \tag{1}$$

where $u(t)$ is a square wave with an ***amplitude of 1 and a frequency of 1 rad/sec***, use an integrator block and a gain block.

The Integrator block integrates its input $\dot{x}(t)$ to produce $x(t)$. Other blocks needed in this model include a Gain block and a Sum block. To generate a square wave, use a Signal Generator block and select the Square Wave form but change the default units to radians/sec. Again, view the output using a Scope block. Gather the blocks and define the gain.
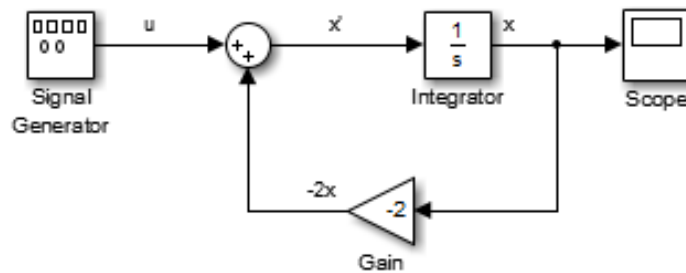

Figure 2.3: Blocks representation of equation 1

An important concept in this model is the loop that includes the Sum block, the Integrator block, and the Gain block.

In this equation, $x(t)$ is the output of the Integrator block. It is also the input to the blocks that compute $\dot{x}(t)$, on which it is based. This relationship is implemented using a loop.

The Scope displays $x(t)$ at each time step. For a simulation lasting 10 seconds, the output shows in figure 4:
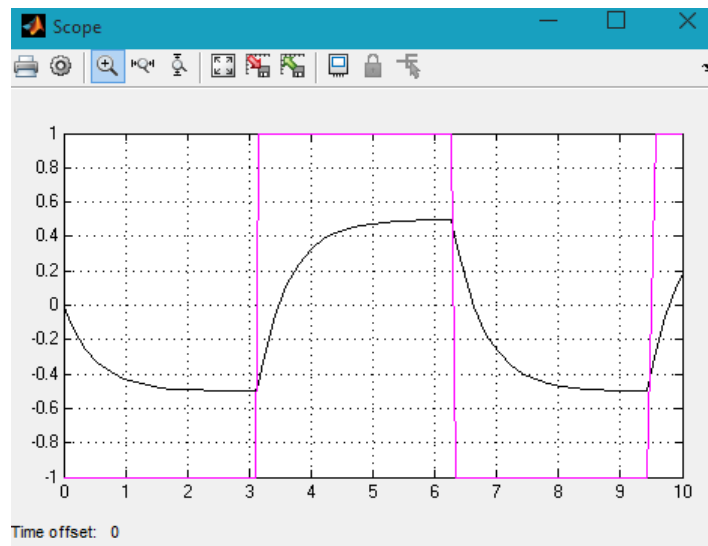

Figure 2.4: The simulation result of $x(t)$

Example .2: Model the differential Equation (Frequency Domain- Transfer Function)

$$\dot{x}(t) = 2x(t) + u(t) \tag{2}$$

The equation you modeled in this example can also be expressed as a transfer function. The model uses the **Transfer Fcn block**, which accepts u as input and outputs $x(t)$.

So, the block implements $(s)/U(s)$. If you substitute $sX(s)$ for $\dot{x}(t)$ in the above equation, you get

$$sX(s) = -2X(s) + U(s)$$

Solving for $X(s)$ gives

$$X(s) = \frac{U(s)}{s+2}$$

Or

$$\frac{X(s)}{U(s)} = \frac{1}{s+2}$$

The **Transfer Fcn** block uses parameters to specify the numerator and denominator coefficients. In this case, *the numerator is 1 and the denominator is s+2*. Specify both terms as vectors of coefficients of successively decreasing powers of s

In this case the numerator is [1] (or just 1) and the denominator is [1 2].
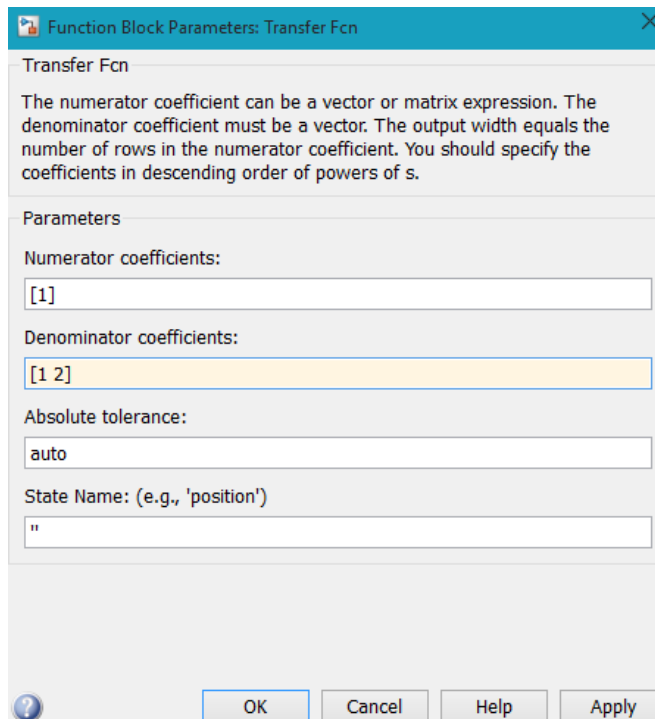

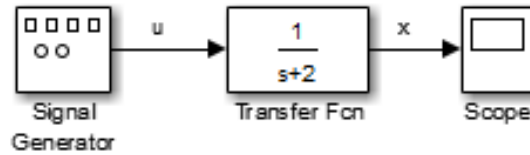**Figure 2.5**:Transfer Fcn block parameter

Figure 2.6: Transfer function representation of equation 2

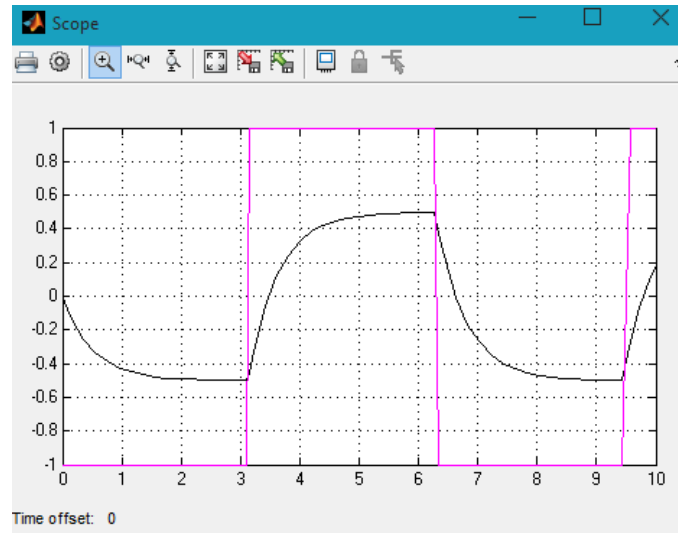**Figure 2.7:** The simulation result of $x(t)$

Example 3:

Find the Solution of the following system

$$\frac{dx_1(t)}{dt} = x_2(t)$$
$$\frac{dx_2(t)}{dt} = -2x_1(t) - 3x_2(t) + u(t)$$

Where $x_1(0) = 1. x_2(0) = 0$

For the input function is: step, ramp, impulse, sinusoidal

References . . . .

1) http://www.mathworks.com/help/index.html
2) Farid Golnaraghi, Benjamin C.Kuo; Automatic Control Systems; Ninth Edition
3) Norman S.Nise; Control Systems Engineering; Sixth Edition
4) Richard C.Dorf, Robert H.Bishop; Modern Control Systems; Twelfth Edition.