Modelling and Simulation ESS101
27 October 2023, Final Exam

This exam contains 11 pages (including this cover page) and 5 problems.

You are allowed to use the following material:

- *Modelling And Simulation, Lecture notes for the Chalmers course ESS101*, by S. Gros (with brief annotations, but cannot contain solutions to the exercises or previous exams)
- *Mathematics Handbook* (Beta)
- *Physics Handbook*
- Chalmers approved calculator
- Formula sheet, appended to the exam.

– Organize your work in a reasonably neat and coherent way. Work scattered all over the page without a clear ordering may receive less credit.

– Mysterious or unsupported answers will not receive credit, but an incorrect answer supported by substantially correct calculations and explanations will receive partial credit.

– None of the proposed questions require extremely long computations. If you get caught in endless algebra, you have probably missed the simple way of doing it.

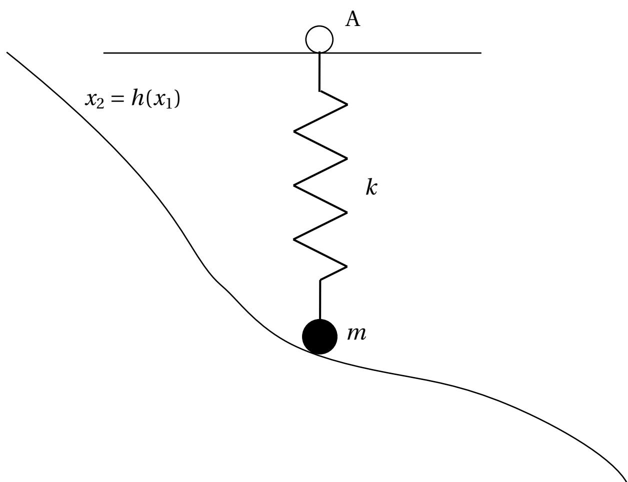
– The passing grade will be given at 20 points, grade 4 at 27 and the top grade at 34 points.

Problem	Points	Score
1	11	
2	6	
3	10	
4	7	
5	6	
Total:	40	

Best of luck to all !!

Examiner: Yasemin Bekiroglu, +46 70 148 72 71

1. Consider the mechanical system depicted below. A ball with (point-like) mass m has the position (x_1, x_2), where x_1 is the horizontal and x_2 the vertical coordinate. The ball is gliding without friction along a rail that is described by the relation $x_2 = h(x_1)$. Further, the ball is attached to one end of a spring, having the spring constant k. The other end of the spring (A) is gliding without friction along a horizontal rail, so that the spring is always vertical.
The forces acting on the ball are thus the spring force (assuming the neutral position of the force corresponds to $x_2 = 0$), gravity g, and the normal force from the rail.



(a) (4 points) Determine the Lagrange function for the system. *Reminder:* Potential energy of a spring is defined as $\frac{1}{2}kp^2$ where k is spring constant and p denotes spring displacement.

(b) (4 points) Derive a dynamic model of the system in DAE form.

(c) (3 points) Derive a standard state-space (ODE) model of the system.
Hint: Use the constraint equation to make substitutions.

Solution:

(a) Using $\mathbf{q} = \mathbf{p} = (x_1, x_2)$, the kinetic and potential energies of the system can be written:

$$T = \frac{1}{2}m(\dot{x}_1^2 + \dot{x}_2^2), \qquad V = mgx_2 + \frac{1}{2}kx_2^2 \tag{1}$$

With the constraint $c(\mathbf{q}) = x_2 - h(x_1) = 0$, the Lagrange function then reads as:

$$\mathcal{L} = T - V - zc = \frac{1}{2}m(\dot{x}_1^2 + \dot{x}_2^2) - mgx_2 - \frac{1}{2}kx_2^2 - z(x_2 - h(x_1)) \tag{2}$$

(b) The dynamics are constructed using:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}}^\top = m\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix}, \qquad \frac{\partial \mathcal{L}}{\partial \mathbf{q}}^\top = -mg\begin{bmatrix} 0 \\ 1 \end{bmatrix} - kx_2\begin{bmatrix} 0 \\ 1 \end{bmatrix} - z\begin{bmatrix} -h'(x_1) \\ 1 \end{bmatrix} \tag{3}$$

Adding the rail constraint, the model then follows from Euler-Lagrange's equation:

$$m\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + mg\begin{bmatrix} 0 \\ 1 \end{bmatrix} + z\begin{bmatrix} -h'(x_1) \\ 1 \end{bmatrix} + kx_2\begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0 \tag{4a}$$

$$x_2 - h(x_1) = 0 \tag{4b}$$

(The DAE model can be transformed into standard semi-explicit form, but that is not required in the problem formulation.)

(c) Differentiating the constraint equation gives

$$\dot{x}_2 = h'(x_1)\dot{x}_1 \tag{5a}$$

$$\ddot{x}_2 = h''(x_1)\dot{x}_1^2 + h'(x_1)\ddot{x}_1 \tag{5b}$$

Combining this with the 2nd row of (4a), we can solve for z:

$$z = -m\ddot{x}_2 - mg - kx_2 = -m\big(h''(x_1)\dot{x}_1^2 + h'(x_1)\ddot{x}_1\big) - mg - kx_2 \tag{6}$$

Inserting this expression into the first row of (4a) now gives a differential equation for x_1:

$$m\big(1 + h'(x_1)^2\big)\ddot{x}_1 + mh'(x_1)h''(x_1)\dot{x}_1^2 + (mg + kx_2)h'(x_1) = 0. \tag{7}$$

Using the state-variables x_1 and $v_1 = \dot{x}_1$, the following state-space model is finally obtained:

$$\dot{x}_1 = v_1 \tag{8a}$$

$$\dot{v}_1 = -\frac{1}{1 + h'(x_1)^2}\big(h'(x_1)h''(x_1)v_1^2 + (g + \frac{k}{m}x_2)h'(x_1)\big) \tag{8b}$$

By adding an "output equation" for x_2, the model completely describes the system:

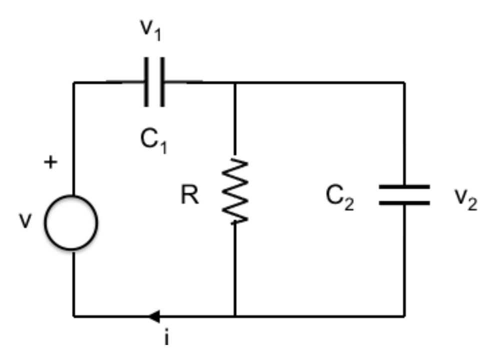
$$x_2 = h(x_1). \tag{9}$$

2. (a) (3 points) Consider the differential equation:

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \dot{\mathbf{x}} = \mathbf{x} \tag{10}$$

Show that we can rewrite this equation in a semi-explicit form having 2 algebraic variables and one differential variable. *Hint: you need to do algebraic manipulations and time-differentiations.*

(b) (3 points) Consider the electrical circuit depicted below, with a voltage source $v(t)$ driving a combination of two capacitors C_1, C_2 and a resistor R. The voltages over the capacitors are $v_1(t), v_2(t)$, and the total current is $i(t)$.



Given the corresponding DAE for the circuit as below, expressed in the variables v_1, v_2, and i, and with v as the input. What is the index of the DAE?

$$C_1 \frac{dv_1}{dt} = i$$
$$C_2 \frac{dv_2}{dt} = i - \frac{v_2}{R}$$
$$v = v_1 + v_2$$

Solution:

(a) We observe that DAE (10) reads as:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{11a}$$
$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{11b}$$
$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 + \dot{\mathbf{x}}_3 = \mathbf{x}_3 \tag{11c}$$

Subtracting (11a) and (11b) to (11c), we get:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{12a}$$
$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{12b}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{12c}$$

A time-differentiation of (12c) yields:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{13a}$$
$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{13b}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{13c}$$
$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 - \dot{\mathbf{x}}_3 = 0 \tag{13d}$$

We then do (13d) - (13a) + (13b) to get:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{14a}$$
$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{14b}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{14c}$$
$$0 = \mathbf{x}_1 - \mathbf{x}_2 \tag{14d}$$

A time-differentiation of (14d) yields:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{15a}$$
$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{15b}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{15c}$$
$$0 = \mathbf{x}_1 - \mathbf{x}_2 \tag{15d}$$
$$\dot{\mathbf{x}}_1 - \dot{\mathbf{x}}_2 = 0 \tag{15e}$$

We then do (15e)+(15a) to get the semi-explicit DAE:

$$\dot{\mathbf{x}}_1 = \frac{1}{2}\mathbf{x}_1 \tag{16a}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{16b}$$
$$0 = \mathbf{x}_1 - \mathbf{x}_2 \tag{16c}$$

(b) To determine the index, differentiate the algebraic equation to get

$$\dot{v} = \dot{v}_1 + \dot{v}_2 = \frac{1}{C_1}i + \frac{1}{C_2}i - \frac{1}{RC_2}v_2 = \left(\frac{1}{C_1} + \frac{1}{C_2}\right)i - \frac{1}{RC_2}v_2$$

Combining this with the two differential equations gives the model equations

$$\underbrace{\begin{bmatrix} C_1 & 0 & 0 \\ 0 & C_2 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{E_1} \begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{i} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -\frac{1}{R} & 1 \\ 0 & \frac{1}{RC_2} & -(\frac{1}{C_1} + \frac{1}{C_2}) \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \dot{v},$$

where the matrix E_1 is singular, i.e. the model is still a DAE. One more differentiation gives

$$\underbrace{\begin{bmatrix} C_1 & 0 & 0 \\ 0 & C_2 & 0 \\ 0 & -\frac{1}{RC_2} & (\frac{1}{C_1} + \frac{1}{C_2}) \end{bmatrix}}_{E_2} \begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{i} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -\frac{1}{R} & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \ddot{v},$$

where the matrix E_2 is now non-singular, i.e. the model is an ODE. Hence, the index is 2.

3. (a) (5 points) Consider the model fitting result in Figure 1, the data pairs are given as $\{[x(i), y(i)]\} = \{[3, 2], [2, 3], [-1, -1]\}$. Using a least squares approach capturing the relationship between x and y, $\hat{\mathbf{y}} = \boldsymbol{\theta}^\top \boldsymbol{\varphi}$ (where $\boldsymbol{\varphi}$ is the *regression vector* holding the *regressors* $(1, x, ...)$), find the values in the parameter vector θ that corresponds to the model fit in the figure.

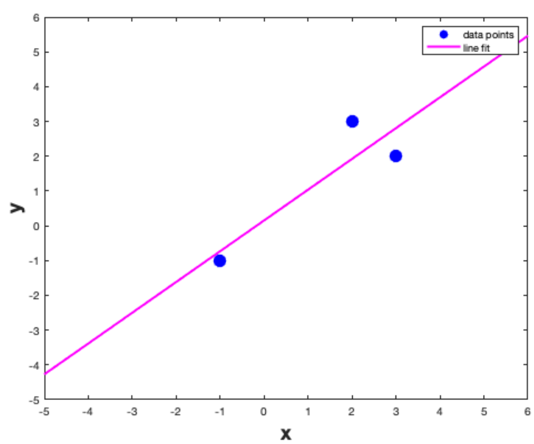


Figure 1: Illustration of three data points and least-squares based model fitting result.

(b) (1 point) Considering the results in Figure 2(a), comment on what is different between these models, and which model fit is better (explain why).

(c) (1 point) Considering the results in Figure 2(b), comment on which model fit might be worse. Comment on issues with worse model fitting results and how to overcome them.

(d) (3 points) Consider the following model:

$$y(t) + 0.7y(t-1) = u(t-1) + 0.5u(t-2) + e(t) + 0.2e(t-1)$$

Find the corresponding plant and noise model transfer functions G and H, and give an *explicit*

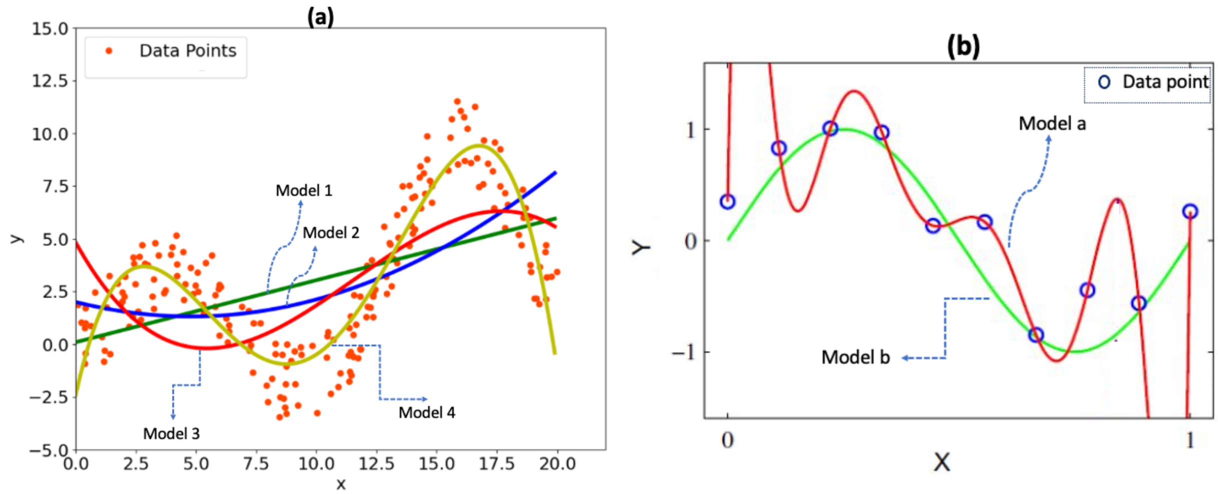


Figure 2: Illustration of raw data points and different model fitting results.

(with numeric coefficients) difference equation, showing how the one-step ahead prediction of the output is computed from data.

Solution:

(a) Using the data vectors $x = [3, 2, -1]$ and $y = [2, 3, -1]$ in the least-squares estimate where $k = 1$ for fitting a line:

$$\hat{\boldsymbol{\theta}}_N = R_N^{-1} f_N = \big(\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\varphi}(i)\boldsymbol{\varphi}^\top(i)\big)^{-1} \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\varphi}(i) y(i)$$

$$\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\varphi}(i)\boldsymbol{\varphi}^\top(i) = \frac{1}{N}\begin{bmatrix} N & \sum_{i=1}^{N} x(i) & \dots & \sum_{i=1}^{N} x^k(i) \\ \sum_{i=1}^{N} x(i) & \sum_{i=1}^{N} x^2(i) & \dots & \sum_{i=1}^{N} x^{k+1}(i) \\ \vdots & \vdots & & \vdots \\ \sum_{i=1}^{N} x^k(i) & \sum_{i=1}^{N} x^{k+1}(i) & \dots & \sum_{i=1}^{N} x^{2*k}(i) \end{bmatrix} \tag{17}$$

$$\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\varphi}(i) y(i) = \frac{1}{N}\begin{bmatrix} \sum_{i=1}^{N} y(i) \\ \sum_{i=1}^{N} x(i)y(i) \\ \vdots \\ \sum_{i=1}^{N} x^k(i)y(i) \end{bmatrix} \tag{18}$$

The resulting parameter estimate: $\hat{\boldsymbol{\theta}}_N = [a, b]^T$, $a = 0.1538$, $b = 0.8846$ in $\hat{\mathbf{y}} = a + bx = \boldsymbol{\theta}^\top \boldsymbol{\varphi}$.

(b) Model 4 is best, it leads to less mean squared error, it is a higher order polynomial.

(c) Model b is better, as model a tends to overfit. When there is a new previously unseen test point, it might lead to larger error.

(d) The plant and noise model transfer functions are

$$G(q) = \frac{B(q)}{A(q)} = \frac{q^{-1} + 0.5q^{-2}}{1 + 0.7q^{-1}} \qquad H(q) = \frac{C(q)}{A(q)} = \frac{1 + 0.2q^{-1}}{1 + 0.7q^{-1}}$$

and the one-step ahead prediction is calculated from

$$C(q)\hat{y}(t|t-1) = B(q)u(t) + \big(C(q) - A(q)\big)y(t),$$

giving the explicit expression

$$\hat{y}(t|t-1) = -0.2\hat{y}(t-1|t-2) + u(t-1) + 0.5u(t-2) - 0.5y(t-1)$$

4. (a) (5 points) Consider the function $\mathbf{f} : \mathbb{R}^2 \mapsto \mathbb{R}^2$, $f(x,y) = \begin{bmatrix} x^2 + y^2 - 4 \\ xy - 1 \end{bmatrix}$, for which we construct approximate solutions to the equation $f(x,y) = [0,0]^T$ using (full step) Newton method and the initial guess $\mathbf{x} = [x_0, y_0] = [2, 0.5]$. Calculate the resulting solution from applying Newton iteration only once, i.e. $[x_1, y_1]$.

   (b) (2 points) Comment on the limitations of the Newton method. When would it fail to converge to a solution?

Solution:

(a) Plugging in the following in the update formula: $\partial f(x,y) = \begin{bmatrix} 2x & 2y \\ y & x \end{bmatrix}$, $det\partial f(x,y) = 2x^2 - 2y^2$, $\partial f(x,y)^{-1} = \begin{bmatrix} \frac{x}{2x^2-2y^2} & \frac{-y}{x^2-y^2} \\ \frac{-y}{2x^2-2y^2} & \frac{x}{x^2-y^2} \end{bmatrix}$

Thus, $\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \end{bmatrix} - \begin{bmatrix} \frac{x_{k-1}}{2x_{k-1}^2-2y_{k-1}^2} & \frac{-y_{k-1}}{x_{k-1}^2-y_{k-1}^2} \\ \frac{-y_{k-1}}{2x_{k-1}^2-2y_{k-1}^2} & \frac{x_{k-1}}{2x_{k-1}^2-2y_{k-1}^2} \end{bmatrix} \cdot \begin{bmatrix} x_{k-1}^2 + y_{k-1}^2 - 4 \\ x_{k-1}y_{k-1} - 1 \end{bmatrix}$

Beginning with $x_0 = 2, y_0 = 0.5$, $x_1 = 1.93$, $y_1 = 0.51$

(b) Sensitivity to initial solution and the choice of the full step vs reduced. When the Jacobian is singular, it fails.

5. (a) (1 point) Why are IRK methods with a large number of stages not favoured in practice?

   (b) (1 point) Why are high-order explicit RK methods often not the optimal choice?

   (c) (1 point) Consider a Runge-Kutta scheme for integration of an ODE $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, defined by the following Butcher array:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 3/4 & 3/4 & 0 \\ \hline & 1/3 & 2/3 \end{array}$$

   Is the RK scheme above explicit or implicit? How many stages are there?

   (d) (1 point) For the RK scheme above, write the equations describing an update of the solution sequence $\{x_k\}$.

   (e) (1 point) For the RK scheme above, determine the stability function.

   (f) (1 point) For the RK scheme above, is the scheme A-stable?

Solution:

(a) IRK methods suffer from the complexity of factorizing the Jacobian matrices involved in the Newton method underlying the integration scheme. A large number of stages provides a very high order, but requires also a heavy linear algebra. The trade off between having a high order and taking fewer steps, or having a lower order but taking more steps is not straightforward, but it tends to favor fairly low order methods.

(b) Up to order $o = 4$, ERK methods require $s = o$ stages, hence $s = o$ evaluations of the model equations. Each extra function evaluation readily delivers an extra order of accuracy, and allows for reducing the total number of function evaluation required. This trend is broken for $o > 4$. At higher orders, the required number of stages (and hence the number of function evaluations) progresses faster than o. Then the overall computational cost of obtaining a given accuracy tends to not improve (or even increase) for higher orders.

(c) The RK scheme is explicit and has 2 stages.

(d)

$$\begin{aligned}
\mathbf{K}_1 &= \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}(t_k)\right) \\
\mathbf{K}_2 &= \mathbf{f}\big(\mathbf{x}_k + \frac{3\Delta t}{4} \cdot \mathbf{K}_1, \mathbf{u}(t_k + \frac{3\Delta t}{4})\big) \\
\mathbf{x}_{k+1} &= \mathbf{x}_k + \frac{\Delta t}{3}\mathbf{K}_1 + \frac{2\Delta t}{3}\mathbf{K}_2
\end{aligned}$$

(e) Denoting the Butcher array as

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

the stability function is given by $R(\mu) = 1 + \mu b^T (I - \mu A)^{-1}\mathbf{1}$, where $\mu = \lambda \Delta t$ and $\mathbf{1}$ is a column vector with all entries equal to 1. Thus:

$$R(\mu) = 1 + \mu \begin{bmatrix} 1/3 & 2/3 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -3/4\mu & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = 1 + \mu + \mu^2/2$$

(f) Since $|R(\mu)|$ increases for large $|\mu|$, the scheme is not A-stable (this is true for all explicit RK schemes).

Appendix: some possibly useful formula

- Lagrange mechanics is built on the equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{Q}, \qquad \mathcal{L}\left(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{z}\right) = \mathcal{T} - \mathcal{V} - \mathbf{z}^\top \mathbf{C}, \qquad \mathbf{C} = 0, \qquad \langle \delta \mathbf{q}, \mathbf{Q} \rangle = \delta W, \forall\, \delta \mathbf{q} \tag{19}$$

  The kinetic and potential energy of a point mass are given by:

$$\mathcal{T} = \frac{1}{2} m \dot{\mathbf{p}}^\top \dot{\mathbf{p}}, \qquad \mathcal{V} = mg\mathbf{p}_3 \tag{20}$$

  respectively, where $\mathbf{p} \in \mathbb{R}^3$ is the position of the mass in a cartesian reference frame having the third coordinate as the vertical axis pointing up. The generalized forces are identical to the external forces applied to a point mass if the position of that point is expressed in cartesian coordinates in the generalized coordinates $\mathbf{q}$.

- In the case $\mathcal{T} = \frac{1}{2} m \dot{\mathbf{q}}^\top W \dot{\mathbf{q}}$ with W constant $\mathcal{V} = \mathcal{V}\left(\mathbf{q}\right)$ and $\mathbf{C} = \mathbf{C}\left(\mathbf{q}\right)$, the Lagrange equations simplify to the dynamics in the semi-explicit index-3 DAE form:

$$\dot{\mathbf{p}} = \mathbf{v} \tag{21a}$$

$$W\dot{\mathbf{v}} + \frac{\partial \mathbf{C}}{\partial \mathbf{q}}^\top \mathbf{z} = \mathbf{Q} - \frac{\partial \mathcal{V}}{\partial \mathbf{q}}^\top \tag{21b}$$

$$0 = \mathbf{C}\left(\mathbf{q}\right) \tag{21c}$$

- The Implicit Function Theorem (IFT) guarantees that a nonlinear set of equations

$$\mathbf{r}\left(\mathbf{y}, \mathbf{z}\right) = 0 \tag{22}$$

  "can be solved" in terms of $\mathbf{z}$ for a given $\mathbf{y}$ iff the Jacobian $\frac{\partial \mathbf{r}(\mathbf{y},\mathbf{z})}{\partial \mathbf{z}}$ is full rank at the solution. More specifically, it guarantees that there is a function $\phi\left(\mathbf{y}\right)$ such that

$$\mathbf{r}\left(\mathbf{y}, \phi\left(\mathbf{y}\right)\right) = 0 \tag{23}$$

  holds in the neighborhood of the point $\mathbf{y}$ where the Jacobian is evaluated. Furthermore, the IFT specifies that:

$$\frac{\partial \mathbf{z}}{\partial \mathbf{y}} = -\frac{\partial \mathbf{r}}{\partial \mathbf{z}}^{-1} \frac{\partial \mathbf{r}}{\partial \mathbf{y}} \tag{24}$$

- For solving a problem $\mathbf{r}\left(\mathbf{x}\right) = 0$, Newton iterates:

$$\mathbf{x} \leftarrow \mathbf{x} - \alpha \frac{\partial \mathbf{r}}{\partial \mathbf{x}}^{-1} \mathbf{r} \tag{25}$$

  until $\mathbf{r}\left(\mathbf{x}\right) \approx 0$ where $\alpha \in [0, 1]$

- Runge-Kutta methods are described by:

$$\begin{array}{c|ccc} c_1 & a_{11} & \dots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_s & a_{s1} & \dots & a_{ss} \\ \hline & b_1 & \dots & b_s \end{array}$$

$$\mathbf{K}_j = \mathbf{f}\left(\mathbf{x}_k + \Delta t \sum_{i=1}^{s} a_{ji} \mathbf{K}_i,\, \mathbf{u}\left(t_k + c_j \Delta t\right)\right), \quad j = 1, \dots, s \tag{26a}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t \sum_{i=1}^{s} b_i \mathbf{K}_i \tag{26b}$$

- For ERK methods, the relationship between the (minimum) number of stages s to the order o is given by:

s	1	2	3	4	6	7	9	11	...
o	1	2	3	4	5	6	7	8	...

Table 1: Stage to order of ERK methods

- Collocation methods use:

$$\dot{\mathbf{x}}(t_k + \Delta t \cdot \tau) \approx \dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau) = \sum_{i=1}^{s} \mathbf{K}_i \ell_i(\tau), \quad \tau \in [0,\, 1] \tag{27}$$

$$\mathbf{x}(t_k + \Delta t \cdot \tau) \approx \hat{\mathbf{x}}(t_k + \Delta t \cdot \tau) = \mathbf{x}_k + \Delta t \sum_{i=1}^{s} \mathbf{K}_i L_i(\tau) \tag{28}$$

where the Lagrange polynomials are given by:

$$\ell_i(\tau) = \prod_{j=1, j\neq i}^{s} \frac{\tau - \tau_j}{\tau_i - \tau_j}, \quad \text{and} \quad L_i(\tau) = \int_0^{\tau} \ell_i(\xi)\mathrm{d}\xi \tag{29}$$

The Lagrange polynomials satisfy the conditions of

$$\text{Orthogonality:} \quad \int_0^1 \ell_i(\tau)\ell_j(\tau)\,\mathrm{d}\tau = 0 \quad \text{for} \quad i \neq j \tag{30a}$$

$$\text{Punctuality:} \quad \ell_i(\tau_j) = \begin{cases} 1 & \text{if} \quad j = i \\ 0 & \text{if} \quad j \neq i \end{cases} \tag{30b}$$

and enforce the collocation equations (for $j = 1, \ldots, s$):

$$\dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau_j) = \mathbf{f}\left(\hat{\mathbf{x}}(t_k + \Delta t \cdot \tau_j),\, \mathbf{u}\left(t_k + \Delta t \cdot \tau_j\right)\right), \quad \text{in the explicit ODE case} \tag{31a}$$

$$\mathbf{F}\left(\dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau_j), \hat{\mathbf{x}}(t_k + \Delta t \cdot \tau_j),\, \mathbf{u}\left(t_k + \Delta t \cdot \tau_j\right)\right) = 0, \quad \text{in the implicit ODE case} \tag{31b}$$

$$\mathbf{F}\left(\dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau_j), \hat{\mathbf{z}}_j, \hat{\mathbf{x}}(t_k + \Delta t \cdot \tau_j),\, \mathbf{u}\left(t_k + \Delta t \cdot \tau_j\right)\right) = 0, \quad \text{in the fully-implicit DAE case} \tag{31c}$$

- Gauss-Legendre collocation methods select the set of points $\tau_{1,\ldots,s}$ as the zeros of the (shifted) Legrendre polynomial:

$$P_s(\tau) = \frac{1}{s!} \frac{\mathrm{d}^s}{\mathrm{d}\tau^s} \left[\left(\tau^2 - \tau\right)^s\right] \tag{32}$$

They achieve the order $\|\mathbf{x}_N - \mathbf{x}(t_\mathrm{f})\| = \mathcal{O}\left(\Delta t^{2s}\right)$.

- Maximum-likelihood estimation is based on

$$\max_{\boldsymbol{\theta}} \quad \mathbb{P}\left[e_k = y_k - \hat{y}_k \quad \text{for} \quad k = 1, \ldots, N \mid \boldsymbol{\theta}\right] \tag{33}$$

If the noise sequence is uncorrelated, then

$$\mathbb{P}\left[e_k = y_k - \hat{y}_k \quad \text{for} \quad k = 0, \ldots, N \mid \boldsymbol{\theta}\right] = \prod_{k=1}^{N} \mathbb{P}\left[e_k = y_k - \hat{y}_k \mid \boldsymbol{\theta}\right] \tag{34}$$

- The solution of a linear least-squares problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2} \|A\boldsymbol{\theta} - \mathbf{y}\|^2_{\Sigma_e^{-1}} \tag{35}$$

reads as:

$$\hat{\boldsymbol{\theta}} = \left(A^\top \Sigma_e^{-1} A\right)^{-1} A^\top \Sigma_e^{-1} \mathbf{y} \tag{36}$$

and the covariance of the parameter estimation based is given by the formula:

$$\Sigma_{\hat{\boldsymbol{\theta}}} = \left(A^\top \Sigma_e^{-1} A\right)^{-1} \tag{37}$$

- In system identification, given the a plant $G(z)$ and a noise $H(z)$ model description, the one-step-ahead predictor $\hat{y}(k|k-1)$ can be retrieved with

$$H(z)\hat{y}(z) = G(z)u(z) + (H(z) - 1)y(z) \tag{38}$$

- The Gauss-Newton approximation in an optimization problem

$$\min_{\mathbf{x}} \quad J(\mathbf{x}) = \frac{1}{2}\left\|\mathbf{R}(\mathbf{x})\right\|^2 \tag{39}$$

uses the approximation:

$$\frac{\partial^2 J}{\partial \mathbf{x}^2} \approx \frac{\partial R}{\partial \mathbf{x}}^\top \frac{\partial R}{\partial \mathbf{x}} \tag{40}$$

- The solution to an LTI system $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ is given by:

$$\mathbf{x}(t) = e^{At}\mathbf{x}(0) + \int_0^t e^{A(t-\tau)} B\mathbf{u}(\tau)\mathrm{d}\tau \tag{41}$$

and the transformation state-space to transfer function is given by:

$$G(s) = C\left(sI - A\right)^{-1} B + D \tag{42}$$

- $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$, $det(A) = ad - bc$
- $A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$, $det(A) = a.det(\begin{bmatrix} e & f \\ h & i \end{bmatrix}) - b.det(\begin{bmatrix} d & f \\ g & i \end{bmatrix}) + c.det(\begin{bmatrix} d & e \\ g & h \end{bmatrix})$
- $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$, $det(A) = ad - bc$, $A^{-1} = \frac{1}{det(A)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$
- $\alpha = \mathbf{x}^T A\mathbf{x}$, where A is a symmetric matrix and $\mathbf{x}$ is $n \times 1$, A is $n \times n$, and A does not depend on $\mathbf{x}$, then, $\frac{\partial \alpha}{\partial \mathbf{x}} = 2\mathbf{x}^T A$.