Modelling and Simulation ESS101
22 August 2022 (Re-exam)

This exam contains 11 pages (including this cover page) and 4 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	11	
3	10	
4	8	
Total:	40	

Best of luck to all !!

Examiner: Yasemin Bekiroglu, +46 70 148 72 71
Teaching Assistant: Remi Lacombe, +46 70 716 34 71

1. Consider a mass "1" (of mass m) moving on a sphere of equation $\varphi(\mathbf{p}) = \frac{1}{2}\left(\mathbf{p}^\top\mathbf{p} - R^2\right) = 0$ and a mass "2" (also of mass m) connected to the mass "1" with a rigid, massless link of length L. The problem is illustrated in Fig. 1

   (a) (6 points) Write down the model equations of this system in the form of a semi-explicit index-3 DAE.

   Note: try to keep your notations compact. You do not need to provide $\frac{\partial \mathbf{C}}{\partial \mathbf{q}}$ explicitly (where $\mathbf{q}$ will be your set of generalised coordinates), but you need to detail the model enough that one would understand how to code it symbolically in the computer (i.e. using basic operations like Jacobians and matrix-vector multiplications)

   (b) (5 points) Propose an equivalent model in the form of a fully-implicit index-1 DAE. Specify its consistency conditions.

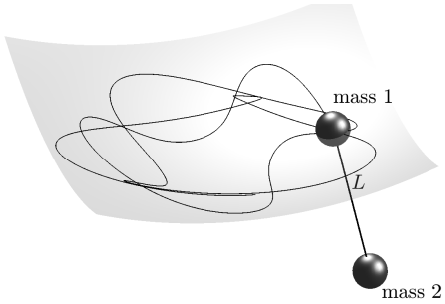

Figure 1: Illustration of the system. The surface φ is depicted as the see-through grey surface. The trajectory of the mass m_1 is depicted as a black trace on the surface φ.

Solution:

(a) Let us describe the system via the generalized coordinates (z-axis up):

$$\mathbf{q} = \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \end{bmatrix} \in \mathbb{R}^6 \tag{1}$$

specifying the position of each ball. The chain then has the kinetic and potential energy:

$$\mathcal{T} = \frac{1}{2}m\sum_{i=1}^{2}\dot{\mathbf{p}}_i^\top\dot{\mathbf{p}}_i \;= \frac{1}{2}m\dot{\mathbf{q}}^\top\dot{\mathbf{q}}, \qquad \mathcal{V} = mg\mathbf{a}^\top\mathbf{q} \tag{2}$$

where $\mathbf{a}^\top = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$. We have the constraint functions

$$\mathbf{C}(\mathbf{q}) = \begin{bmatrix} \varphi(\mathbf{p}_1) \\ \frac{1}{2}\left(\|\mathbf{p}_2 - \mathbf{p}_1\|^2 - L^2\right) \end{bmatrix} \tag{3}$$

The Lagrange function then reads as:

$$\mathcal{L}(\dot{\mathbf{q}}, \mathbf{q}, \mathbf{z}) = \frac{1}{2}m\dot{\mathbf{q}}^\top\dot{\mathbf{q}} - mg\mathbf{a}^\top\mathbf{q} - \mathbf{z}^\top\mathbf{C}(\mathbf{q}) \tag{4}$$

The Lagrange equation then provides the dynamics, using:

$$\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} = m\dot{\mathbf{q}}, \quad \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} = m\ddot{\mathbf{q}}, \quad \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = -mg\mathbf{a} - \mathbf{z}^\top \frac{\partial \mathbf{C}}{\partial \mathbf{q}} \tag{5}$$

Hence we have the index-3 DAE model:

$$\dot{\mathbf{q}} = \mathbf{v} \tag{6a}$$

$$\dot{\mathbf{v}} = -g\mathbf{a} - \frac{1}{m}\frac{\partial \mathbf{C}}{\partial \mathbf{q}}^\top \mathbf{z} \tag{6b}$$

$$0 = \mathbf{C}(\mathbf{q}) \tag{6c}$$

This result could also be obtained directly from (28) by observing that $W(\mathbf{q}) = I$. We can additionally further explicitly provide (though this is not required from the question):

$$\frac{\partial \mathbf{C}}{\partial \mathbf{q}}^\top = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_1 - \mathbf{p}_2 \\ 0 & \mathbf{p}_2 - \mathbf{p}_1 \end{bmatrix} \tag{7}$$

(b) The index-reduced model is obtained by performing 2 time differentiations of $\mathbf{C} = 0$. They provide:

$$\dot{\mathbf{C}} = \begin{bmatrix} \mathbf{p}_1^\top \dot{\mathbf{p}}_1 \\ (\mathbf{p}_2 - \mathbf{p}_1)^\top (\dot{\mathbf{p}}_2 - \dot{\mathbf{p}}_1) \end{bmatrix} = 0 \tag{8a}$$

$$\ddot{\mathbf{C}} = \underbrace{\begin{bmatrix} \mathbf{p}_1^\top \ddot{\mathbf{p}}_1 \\ (\mathbf{p}_2 - \mathbf{p}_1)^\top (\ddot{\mathbf{p}}_2 - \ddot{\mathbf{p}}_1) \end{bmatrix}}_{=\frac{\partial \mathbf{C}}{\partial \mathbf{q}}\ddot{\mathbf{q}}} + \begin{bmatrix} \dot{\mathbf{p}}_1^\top \dot{\mathbf{p}}_1 \\ (\dot{\mathbf{p}}_2 - \dot{\mathbf{p}}_1)^\top (\dot{\mathbf{p}}_2 - \dot{\mathbf{p}}_1) \end{bmatrix} = 0 \tag{8b}$$

We can then assemble the semi-explicit index-1 DAE model e.g.

$$\begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ -g\mathbf{a} - \frac{1}{m}\frac{\partial \mathbf{C}}{\partial \mathbf{q}}^\top \mathbf{z} \end{bmatrix} \tag{9a}$$

$$0 = \ddot{\mathbf{C}} \tag{9b}$$

or in a complete, matrix form:

$$\dot{\mathbf{q}} = \mathbf{v} \tag{10a}$$

$$\begin{bmatrix} mI & \frac{\partial \mathbf{C}}{\partial \mathbf{q}} \\ \frac{\partial \mathbf{C}}{\partial \mathbf{q}}^\top & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \mathbf{z} \end{bmatrix} = - \begin{bmatrix} mg\mathbf{a} \\ \dot{\mathbf{p}}_1^\top \dot{\mathbf{p}}_1 \\ (\dot{\mathbf{p}}_2 - \dot{\mathbf{p}}_1)^\top (\dot{\mathbf{p}}_2 - \dot{\mathbf{p}}_1) \end{bmatrix} \tag{10b}$$

For simulation purposes, one would typically introduce $\mathbf{p}_1, \mathbf{p}_2, \dot{\mathbf{p}}_1, \dot{\mathbf{p}}_2, \mathbf{z}$ in the state-space. The consistency conditions require:

$$\mathbf{C}(\mathbf{q})|_{t=0} = 0, \qquad \dot{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\Big|_{t=0} = 0 \tag{11}$$

2. (a) (4 points) Perform an index reduction of the DAE:

$$\dot{\mathbf{x}} = A\mathbf{x} + \mathbf{b}z \tag{12a}$$

$$0 = \mathbf{g}(\mathbf{x}) \tag{12b}$$

where $z \in \mathbb{R}$ and function $\mathbf{g} : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R} \mapsto \mathbb{R}^m$. Assume that $\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\mathbf{b}$ is full rank.

(b) (7 points) Consider the differential equation:

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \dot{\mathbf{x}} = \mathbf{x} \tag{13}$$

1. (2 points) Is (13) an implicit ODE or a DAE? Justify.
2. (5 points) 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.*

Solution:

(a) We are dealing with a semi-explicit DAE, hence the index reduction requires time-differentiations of the algebraic equation (12b). The first step of the index reduction reads as:

$$\dot{\mathbf{x}} = A\mathbf{x} + \mathbf{b}z \tag{14a}$$

$$0 = \frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}}\dot{\mathbf{x}} = \frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}}(A\mathbf{x} + \mathbf{b}z) \tag{14b}$$

We note that $\frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}}\mathbf{b}$ is scalar here, and different than zero since it is full rank. It follows that (14) is of index 1. The consistency condition is simply:

$$\mathbf{g}(\mathbf{x}(0)) = 0 \tag{15}$$

DAE (12) is of index 2 if $\frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}}\mathbf{b} \neq 0$, if $\frac{\partial \mathbf{g}(\mathbf{x})}{\partial \mathbf{x}}\mathbf{b} = 0$, then it is of index at least 3.

(b)
1. Since matrix E is rank deficient (3^{rd} line is the sum of the 1^{st} and 2^{nd} lines), (13) is a DAE
2. We observe that DAE (13) reads as:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{16a}$$

$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{16b}$$

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 + \dot{\mathbf{x}}_3 = \mathbf{x}_3 \tag{16c}$$

Subtracting (16a) and (16b) to (16c), we get:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{17a}$$

$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{17b}$$

$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{17c}$$

A time-differentiation of (17c) yields:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{18a}$$

$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{18b}$$

$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{18c}$$

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 - \dot{\mathbf{x}}_3 = 0 \tag{18d}$$

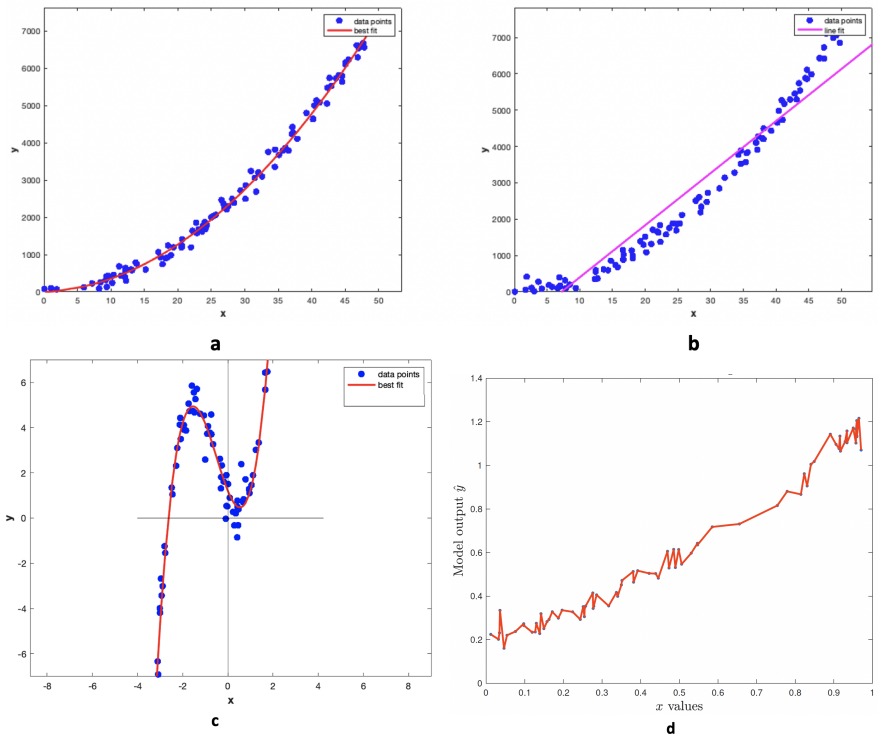

Figure 2: Linear regression examples based on least squares.

We then do (18d) - (18a) + (18b) to get:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{19a}$$
$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{19b}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{19c}$$
$$0 = \mathbf{x}_1 - \mathbf{x}_2 \tag{19d}$$

A time-differentiation of (19d) yields:

$$\dot{\mathbf{x}}_1 + \dot{\mathbf{x}}_2 = \mathbf{x}_1 \tag{20a}$$
$$\dot{\mathbf{x}}_3 = \mathbf{x}_2 \tag{20b}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{20c}$$
$$0 = \mathbf{x}_1 - \mathbf{x}_2 \tag{20d}$$
$$\dot{\mathbf{x}}_1 - \dot{\mathbf{x}}_2 = 0 \tag{20e}$$

We then do (20e)+(20a) to get the semi-explicit DAE:

$$\dot{\mathbf{x}}_1 = \frac{1}{2}\mathbf{x}_1 \tag{21a}$$
$$0 = \mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 \tag{21b}$$
$$0 = \mathbf{x}_1 - \mathbf{x}_2 \tag{21c}$$

3. (a) (3 points) Consider the curve fitting results in Figure 2a-d. Discuss the curve fitting results when Figure 2a and Figure 2b are compared, which one is preferable and why? What mainly changes in

the fitted curve models from Figure 2a to Figure 2c? Comment on the quality of the model fitting in Figure 2d, what could be potential problems and how they can be resolved?

(b) (3 points) Consider a two-dimensional dataset $\{x(i), y(i)\}, i = 1, ...N$. Using a least squares approach capturing the relationship between x and y,

$$\hat{\mathbf{y}} = a + bx + cx^2 + dx^3 + ex^4 + fx^5 = \boldsymbol{\theta}^\top \boldsymbol{\varphi},$$

where $\boldsymbol{\theta}$ is the parameter vector and $\boldsymbol{\varphi}$ is the *regression vector* (holding the *regressors* $1, x, ...$), write out all the components in the R_N and f_N matrices in the least squares estimate defined as

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

(c) (2 points) What is a loss function, give an example, how is it used in model fitting? How do loss function values change with respect to increasing model order?

(d) (2 points) Discuss what overfitting to training data means and discuss an approach to avoid overfitting.

Solution:

(a) Model fit in Figure 2a is better, in comparison to Figure 2b, as the fitting error is less. Model fit quality can be determined by a loss function based on the differences between the predictions and the ground truth values. Model in Figure 2c has a higher order, it is a more complex model. Figure 2d shows overfitting which is problematic as it leads to bad generalization performance, i.e. model predictions on unseen data would not be of good quality. Cross-validation can help to find a good model order to avoid overfitting.

(b) The matrices are written as below where $k = 5$

$$\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\varphi}(i)\boldsymbol{\varphi}^\top(i) = \frac{1}{N}\begin{bmatrix} N & \sum_{i=1}^{N} x(i) & ... & \sum_{i=1}^{N} x^k(i) \\ \sum_{i=1}^{N} x(i) & \sum_{i=1}^{N} x^2(i) & ... & \sum_{i=1}^{N} x^{k+1}(i) \\ . & . & . & \\ . & . & . & \\ . & . & . & \\ \sum_{i=1}^{N} x^k(i) & \sum_{i=1}^{N} x^{k+1}(i) & ... & \sum_{i=1}^{N} x^{2*k}(i) \end{bmatrix} \tag{22}$$

$$\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\varphi}(i)y(i) = \frac{1}{N}\begin{bmatrix} \sum_{i=1}^{N} y(i) \\ \sum_{i=1}^{N} x(i)y(i) \\ . \\ . \\ . \\ \sum_{i=1}^{N} x^k(i)y(i) \end{bmatrix} \tag{23}$$

(c) Loss function is used to evaluate how well a model fits the data. A basic example is mean squared error based on the difference between predictions and the ground truth data. Loss function values decrease as the model fits well to the data, i.e., typically when the model order is incrreased.

(d) Overfitting occurs when the model parameters are too fine tuned wrt training data and the model complexity is too high, this means the model predictions do not perform well when there is unseen test data. We can choose model order when the loss does not drop significantly any more - corresponding to the knee in the figure.

4. (a) (1 point) How can one spot from the Butcher tableau if the RK method it describes is explicit or implicit?

   (b) (1 point) Specify what information the Butcher tableau readily provides on the resulting RK scheme, and what information is not obviously available (answers can be short but must be specific).

   (c) (4 points) 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:

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

   1. Is the RK scheme explicit or implicit? How many stages are there?
   2. Write the equations describing an update of the solution sequence $\{x_k\}$.
   3. Determine the stability function.
   4. Is the scheme A-stable?

   (d) (1 point) The Newton methods aims at solving a set of equation $\mathbf{r}(\mathbf{x}) = 0$ numerically. To that end, iterates the recursion:

$$\frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} \Delta \mathbf{x} + \mathbf{r}(\mathbf{x}) = 0 \tag{24a}$$

$$\mathbf{x} \leftarrow \mathbf{x} + \alpha \Delta \mathbf{x} \tag{24b}$$

   where $\alpha \in]0, 1]$ is the step-size. Explain in words what condition(s) is (are) required for Newton to converge with $\alpha = 1$.

   (e) (1 point) Why do we need α and how should it be chosen?

Solution:

(a) Butcher tableaus describing explicit methods have to be (lower) triangular. If they are not then they describe an implicit method.

(b) The Butcher tableau specifies: the number of stages of the RK method, whether the method is implicit or explicit and enough information to code the RK scheme. It does not (readily) provide the order of the integration method, as the order depends on the specific entries used in the tableau. To assess the order from the tableau, one needs to perform (possibly) involved computations.

(c) 1. The RK scheme is explicit and has 2 stages.

2.

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

3. 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:

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

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

(d) Full Newton steps are guaranteed to converge in a neighborhood of a solution only. The "size" of that neighborhood depends on how nonlinear $\mathbf{r}(\mathbf{x})$ is, and the Jacobian $\frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}}$ must be full rank throughout this neighborhood.

(e) The step-size α allows the Newton iteration to converge even if $\mathbf{x}$ is not close to the solution $\mathbf{x}_\star$. It is typically chosen so as to ensure that

$$\|\mathbf{r}(\mathbf{x} + \alpha\Delta\mathbf{x})\| < \|\mathbf{r}(\mathbf{x})\| \tag{25}$$

Finding α is the role of a (small) computer code usually labelled "line-search".

Appendix: some possibly useful formula

- Lagrange mechanics is built on the equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{Q}, \qquad \mathcal{L}\left(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{z}\right) = \mathcal{T} - \mathcal{V} - \mathbf{z}^\top \mathbf{C}, \qquad \mathbf{C} = 0, \qquad \langle \delta\mathbf{q},\, \mathbf{Q}\rangle = \delta W, \forall\, \delta\mathbf{q} \tag{26}$$

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

$$\mathcal{T} = \frac{1}{2} m \dot{\mathbf{p}}^\top \dot{\mathbf{p}}, \qquad \mathcal{V} = mg\mathbf{p}_3 \tag{27}$$

  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:

$$\dot{\mathbf{p}} = \mathbf{v} \tag{28a}$$

$$W\dot{\mathbf{v}} + \frac{\partial \mathbf{C}}{\partial \mathbf{q}}^\top \mathbf{z} = \mathbf{Q} - \frac{\partial \mathcal{V}}{\partial \mathbf{q}}^\top \tag{28b}$$

$$0 = \mathbf{C}\left(\mathbf{q}\right) \tag{28c}$$

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

$$\mathbf{r}\left(\mathbf{y}, \mathbf{z}\right) = 0 \tag{29}$$

  "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

$$\mathbf{r}\left(\mathbf{y}, \phi\left(\mathbf{y}\right)\right) = 0 \tag{30}$$

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

$$\frac{\partial \mathbf{z}}{\partial \mathbf{y}} = -\frac{\partial \mathbf{r}}{\partial \mathbf{z}}^{-1} \frac{\partial \mathbf{r}}{\partial \mathbf{y}} \tag{31}$$

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

$$\mathbf{x} \leftarrow \mathbf{x} - \alpha \frac{\partial \mathbf{r}}{\partial \mathbf{x}}^{-1} \mathbf{r} \tag{32}$$

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

- Runge-Kutta methods are described by:

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

$$\mathbf{K}_j = \mathbf{f}\left(\mathbf{x}_k + \Delta t \sum_{i=1}^{s} a_{ji}\mathbf{K}_i,\, \mathbf{u}\left(t_k + c_j \Delta t\right)\right), \quad j = 1, \dots, s \tag{33a}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t \sum_{i=1}^{s} b_i \mathbf{K}_i \tag{33b}$$

- 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:s

$$\dot{\mathbf{x}}(t_k + \Delta t \cdot \tau) \approx \dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau) = \sum_{i=1}^{s} \mathbf{K}_i \ell_i(\tau), \quad \tau \in [0,\, 1] \tag{34}$$

$$\mathbf{x}(t_k + \Delta t \cdot \tau) \approx \hat{\mathbf{x}}(t_k + \Delta t \cdot \tau) = \mathbf{x}_k + \Delta t \sum_{i=1}^{s} \mathbf{K}_i L_i(\tau) \tag{35}$$

where the Lagrange polynomials are given by:

$$\ell_i(\tau) = \prod_{j=1, j\neq i}^{s} \frac{\tau - \tau_j}{\tau_i - \tau_j}, \quad \text{and} \quad L_i(\tau) = \int_0^{\tau} \ell_i(\xi) \mathrm{d}\xi \tag{36}$$

The Lagrange polynomials satisfy the conditions of

$$\text{Orthogonality:} \quad \int_0^1 \ell_i(\tau)\ell_j(\tau)\,\mathrm{d}\tau = 0 \quad \text{for} \quad i \neq j \tag{37a}$$

$$\text{Punctuality:} \quad \ell_i(\tau_j) = \begin{cases} 1 & \text{if} \quad j = i \\ 0 & \text{if} \quad j \neq i \end{cases} \tag{37b}$$

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

$$\dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau_j) = \mathbf{f}\left(\hat{\mathbf{x}}(t_k + \Delta t \cdot \tau_j),\, \mathbf{u}\left(t_k + \Delta t \cdot \tau_j\right)\right), \qquad \text{in the explicit ODE case} \tag{38a}$$

$$\mathbf{F}\left(\dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau_j), \hat{\mathbf{x}}(t_k + \Delta t \cdot \tau_j),\, \mathbf{u}\left(t_k + \Delta t \cdot \tau_j\right)\right) = 0, \qquad \text{in the implicit ODE case} \tag{38b}$$

$$\mathbf{F}\left(\dot{\hat{\mathbf{x}}}(t_k + \Delta t \cdot \tau_j), \hat{\mathbf{z}}_j, \hat{\mathbf{x}}(t_k + \Delta t \cdot \tau_j),\, \mathbf{u}\left(t_k + \Delta t \cdot \tau_j\right)\right) = 0, \qquad \text{in the fully-implicit DAE case} \tag{38c}$$

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

$$P_s\left(\tau\right) = \frac{1}{s!}\frac{\mathrm{d}^s}{\mathrm{d}\tau^s}\left[\left(\tau^2 - \tau\right)^s\right] \tag{39}$$

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

- Maximum-likelihood estimation is based on

$$\max_{\boldsymbol{\theta}} \quad \mathbb{P}\left[e_k = y_k - \hat{y}_k \quad \text{for} \quad k = 1, \ldots, N \mid \boldsymbol{\theta}\right] \tag{40}$$

If the noise sequence is uncorrelated, then

$$\mathbb{P}\left[e_k = y_k - \hat{y}_k \quad \text{for} \quad k = 0, \ldots, N \mid \boldsymbol{\theta}\right] = \prod_{k=1}^{N} \mathbb{P}\left[e_k = y_k - \hat{y}_k \,|\, \boldsymbol{\theta}\,\right] \tag{41}$$

- The solution of a linear least-squares problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2}\left\|A\boldsymbol{\theta} - \mathbf{y}\right\|^2_{\Sigma_e^{-1}} \tag{42}$$

reads as:

$$\hat{\boldsymbol{\theta}} = \left(A^\top \Sigma_e^{-1} A\right)^{-1} A^\top \Sigma_e^{-1} \mathbf{y} \tag{43}$$

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

$$\Sigma_{\hat{\boldsymbol{\theta}}} = \left(A^\top \Sigma_e^{-1} A\right)^{-1} \tag{44}$$

- 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

$$H(z)\hat{y}(z) = G(z)u(z) + (H(z) - 1)y(z) \tag{45}$$

- The Gauss-Newton approximation in an optimization problem

$$\min_{\mathbf{x}} \quad J(\mathbf{x}) = \frac{1}{2} \|\mathbf{R}(\mathbf{x})\|^2 \tag{46}$$

uses the approximation:

$$\frac{\partial^2 J}{\partial \mathbf{x}^2} \approx \frac{\partial R}{\partial \mathbf{x}}^\top \frac{\partial R}{\partial \mathbf{x}} \tag{47}$$

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

$$\mathbf{x}(t) = e^{At}\mathbf{x}(0) + \int_0^t e^{A(t-\tau)} B\mathbf{u}(\tau) \mathrm{d}\tau \tag{48}$$

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

$$G(s) = C\left(sI - A\right)^{-1} B + D \tag{49}$$

- $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$, $det(A) = ad - bc$
- $A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$, $det(A) = a.det(\begin{bmatrix} e & f \\ h & i \end{bmatrix}) - b.det(\begin{bmatrix} d & f \\ g & i \end{bmatrix}) + c.det(\begin{bmatrix} d & e \\ g & h \end{bmatrix})$
- $\alpha = \mathbf{x}^T A \mathbf{x}$, 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, $\frac{\partial \alpha}{\partial \mathbf{x}} = 2\mathbf{x}^T A$.