

ETH Lectures 401-0663-00L Numerical Methods for Computer Science
401-2673-00L Numerical Methods for CSE

Homework Problems and Projects

Prof. R. Hiptmair, SAM, ETH Zurich

Autumn Term 2021, Version of October 2, 2022
(C) Seminar für Angewandte Mathematik, ETH Zürich

[Link](#) to the current version of this homework collection

0.1 General Information

0.1.1 Weekly Homework Assignments

All problems will be published in this single “.pdf” file. Every week, we publish a list of problem numbers on the lecture Moodle page. These are the current assignments for that week. Details are given on the lecture Moodle page.

0.1.2 Importance of Homework

Homework assignments are **not** mandatory. However, it is **very important** that you constantly exercise with the material you learn. “Solving” the homework assignments one week before the main exam by looking at the solutions will likely result in failure.

We provide hints and solutions from the beginning. It is your responsibility to look at the solutions only if you are stuck with a difficult problem and you tried to find a solution for a sufficient amount of time.

Most homework problems come with an estimate for the time required for its solution. These times are what we expect a bright and well-prepared student to require for that problem during an exam.

0.1.3 Corrections and Grading of Assignments

Homework assignments will not be graded.

However, you may submit your handwritten solutions by handing them to your tutor directly or uploading them via the Moodle upload interface, see the lecture Moodle page for instructions. The assistants will

Problem 8-7: Nonlinear electric circuit

[Lecture → Ex. 2.1.0.3] discusses electric circuits with elements that give rise to linear voltage–current dependence, see [Lecture → Ex. 2.1.0.3] and [Lecture → Ex. 2.8.0.1]. The principles of nodal analysis were explained in these cases.

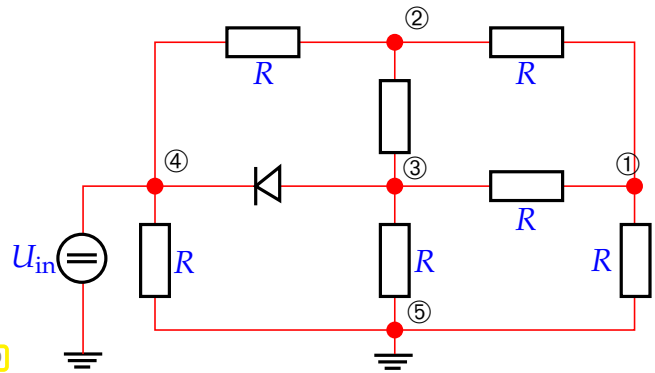
However, the electrical circuits encountered in practise usually feature elements with a *non-linear* current-voltage characteristic. Then nodal analysis leads to non-linear systems of equations as was elaborated in [Lecture → Ex. 8.1.0.1]. Please note that transformation to frequency domain is not possible for non-linear circuits so that we will always study the direct current (DC) situation.

In this problem we deal with a very simple non-linear circuit element, a diode. The current I through a diode to which a voltage U is applied can be modelled by the relationship

$$I = \alpha \left(e^{\beta \frac{U}{U_T}} - 1 \right)$$

with suitable parameters $\alpha, \beta > 0$ and the thermal voltage U_T .

Now we consider the circuit depicted in Fig. 90 and assume that all resistors have non-dimensional resistance $R = 1$.



The circuit is driven by imposing a voltage U_{in} through an ideal voltage source.

(8-7.a) □ (20 min.) Study again [Lecture → Ex. 2.1.0.3] and [Lecture → Ex. 8.1.0.1] to refresh your knowledge about the nodal analysis of electric circuits. ▲

(8-7.b) □ (20 min.) Carry out the nodal analysis of the electric circuit and derive the corresponding non-linear system of equations $\mathbf{F}(\mathbf{u}) = \mathbf{0}$ for the potentials in nodes ①, ② and ③, cf. [Lecture → Eq. (8.1.0.2)]. Note that the voltages in nodes 4 and 5 are known (given input voltage U_{in} and ground voltage 0).

SOLUTION of (8-7.b):

We consider Kirchhoff's law $\sum_{k,j} I_{kj} = 0$ for every node k . I_{kj} contributes to the sum if node j is connected to node k through a circuit element. In particular we use the constitutive relation for the Ohmic resistor, $I = U/R$, here with $R = 1$.

$$\begin{aligned} \textcircled{1} \quad & 3U_1 - 0 - U_2 - U_3 = 0, \\ \textcircled{2} \quad & 3U_2 - U_1 - U_3 - U_{in} = 0, \\ \textcircled{3} \quad & 3U_3 - 0 - U_1 - U_2 + \alpha \left(e^{\beta \left(\frac{U_3 - U_{in}}{U_T} \right)} - 1 \right) = 0. \end{aligned}$$

Thus the nonlinear system of equations reads

$$\mathbf{F}(\mathbf{u}) = \begin{bmatrix} 3U_1 - U_2 - U_3 \\ 3U_2 - U_1 - U_3 - U_{in} \\ 3U_3 - U_1 - U_2 + \alpha e^{-\frac{\beta}{U_T} U_{in}} \cdot e^{\frac{\beta}{U_T} U_3} - \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{for } \mathbf{u} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}.$$

**(8-7.c)** (20 min.) [depends on Sub-problem (8-7.b)]

In the file `nonlinear_circuit.hpp` write an EIGEN-based C++ function

```
void circuit(const alpha, double beta,
const Eigen::VectorXd &Uin, Eigen::VectorXd &Uout)
```

that computes the output voltages U_{out} , which is defined to be the voltage at node ① in Fig. 90 for a *sorted* vector of input voltages U_{in} (at node ④) and for a thermal voltage $U_T = 0.5$. The parameters α, β pass the (non-dimensional) diode parameters α, β .

Use Newton's method to solve $\mathbf{F}(\mathbf{u}) = \mathbf{0}$ with a relative tolerance of $\tau = 10^{-6}$.

SOLUTION of (8-7.c):

According to [Lecture → Section 8.5.1] the iteration of the multi-dimensional Newton method is

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} - (\mathbf{D}\mathbf{F}(\mathbf{u}^{(k)})^{-1} \mathbf{F}(\mathbf{u}^{(k)}).$$

The components of the Jacobian $\mathbf{D}\mathbf{F}$ are defined as $DF(\mathbf{u})_{ij} := \frac{\partial F_i(\mathbf{u})}{\partial u_j}$. For the function obtained in a) we get

$$\mathbf{D}\mathbf{F}(\mathbf{u}) = \begin{bmatrix} 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 + \frac{\alpha\beta}{U_T} e^{\beta \frac{U_3 - U_{in}}{U_T}} \end{bmatrix}.$$

We use EIGEN's built-in Gaussian elimination solver, see [Lecture → § 2.5.0.4].

C++-code 8.7.1: Solution

```
2  void circuit(const double& alpha, const double& beta,
3              const Eigen::VectorXd& Uin, Eigen::VectorXd& Uout) {
4      constexpr double Ut = 0.5;
5      constexpr double tau = 1e-6;
6      const unsigned int n = Uin.size();
7
8      // TODO: (8-7.b) Compute the output voltages for the given circuit
9      // START
10     double Uin_; // Uin of current node
11     // lambda function for evaluation of F
12     auto F = [alpha, beta](const Eigen::VectorXd& U, const double Uin_) {
13         Eigen::VectorXd f(3);
14         f << 3. * U(0) - U(1) - U(2), 3. * U(1) - U(0) - U(2) - Uin_,
15             3. * U(2) - U(0) - U(1) +
16             alpha * (std::exp(beta * (U(2) - Uin_) / Ut) - 1);
17         return f;
18     };
19
20     Eigen::MatrixXd J(3, 3); // the Jacobian
21     J << 3, -1, -1, -1, 3, -1, -1, -1, 0; // dummy in J(2,2)
22     Eigen::VectorXd f(3); // the function
23
24     for (unsigned int i = 0; i < n; ++i) {
25         Uin_ = Uin(i);
26         Eigen::VectorXd U = Eigen::VectorXd::Random(3); // random initial guess
27         Eigen::VectorXd h = Eigen::VectorXd::Ones(3);
```

```

28 while (h.cwiseAbs().maxCoeff() > tau * U.norm()) {
29     J(2, 2) = 3. + (alpha * beta) / Ut * std::exp(beta * (U(2) - Uin_) / Ut);
30     f = F(U, Uin_);
31     h = J.partialPivLu().solve(f);
32     U -= h;
33 }
34 Uout(i) = U(0);
35 }
36 // END
37 }

```

Get it on [GitLab](#) (nonlinear_circuit.hpp).



(8-7.d) 🕒 (15 min.) [depends on Sub-problem (8-7.c)]

We are interested in the nonlinear effects introduced by the diode. Using **MATPLOTLIBCPP** implement a C++ function

```
void plotU(void);
```

that creates a plot of the output voltage $U_{\text{out}} = U_{\text{out}}(U_{\text{in}})$ as a function of the variable input voltage $U_{\text{in}} \in [0, 20]$ (for non-dimensional parameters $\alpha = 8$, $\beta = 1$ and for a thermal voltage $U_T = 0.5$). How can you discern the non-linearity of the circuit in the plot?

SOLUTION of (8-7.d):

The nonlinear effects can be observed by calculating the differences between the solutions U_{out} : see file `nonlinear_circuit.hpp`. The relationship $U_{\text{out}} = U_{\text{out}}(U_{\text{in}})$ is obviously not affine-linear.

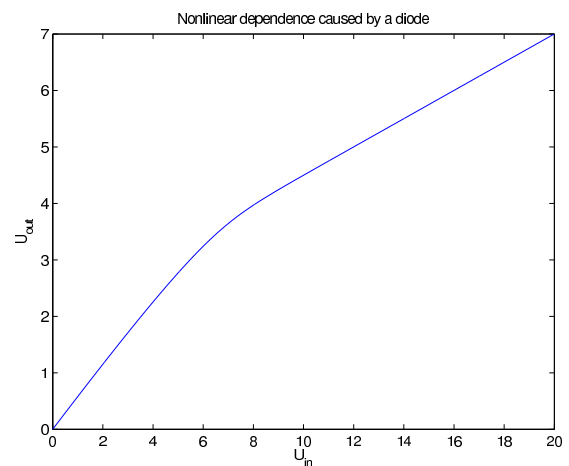


Fig. 91

C++-code 8.7.2: Solution

```

2 void plotU() {
3     constexpr unsigned int n = 20; // number of grid points
4     constexpr double alpha = 8;
5     constexpr double beta = 1;
6
7     plt::figure();
8
9     // TODO: (8-7.c) Plot Uout as a function of Uin with matplotlibcpp
10    // START
11    Eigen::VectorXd Uin = Eigen::VectorXd::LinSpaced(n, 0, 20);
12    Eigen::VectorXd Uout(n);

```

```
13 circuit(alpha, beta, Uin, Uout);  
14  
15 plt::plot(Uin, Uout);  
16 plt::title("Nonlinear dependence caused by a diode");  
17 plt::xlabel("U_in");  
18 plt::ylabel("U_out");  
19 // END  
20 plt::savefig("./cx_out/Udiode.png");  
21 }
```

Get it on 🐙 [GitLab](#) (nonlinear_circuit.hpp).



End Problem 8-7, 75 min.