

# 1 TKT 4150 Biomechanics: Problem set 9

For an optimal experience visit "NTNU's jupyter-server": "<https://10.212.25.27/hub>" and login with your NTNU credentials or visit "tmpnb.org": "<http://tmpnb.org>" and upload "windkessels.ipynb", "Windkessel Schematics.jpg", and '55arteryNetwork\_age19.csv' and '55arteryNetwork\_age75.csv'. Then open windkessels.ipynb on that server to continue.

**Introduction.** This is an IPython notebook, for a more general introduction please see this "introduction": "<https://nbviewer.jupyter.org/github/jupyter/notebook/blob/master/docs/source/exam>

## Save your work!

- If you are running this on "tmpnb.org": "<http://tmpnb.org>", remember to save your work by downloading the notebook via clicking **File**->**Download As** in the toolbar.

## Exercise 1: Deriving the Windkessel Model

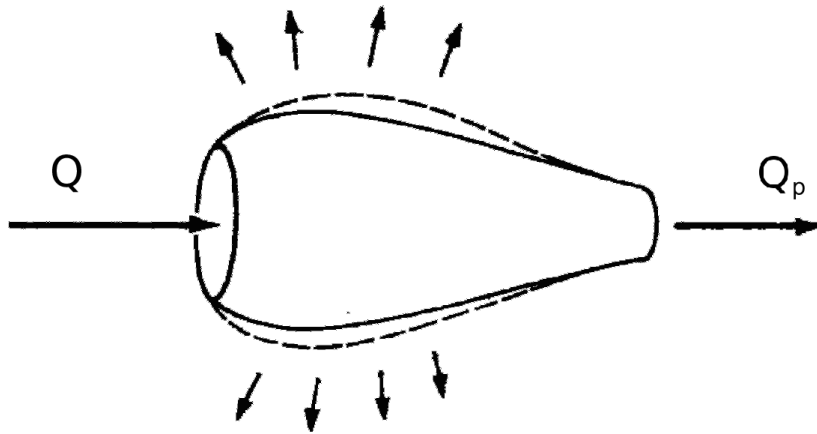


Figure 1: Diagram of mass balance in a section of compliant pipe.  $Q$  is  $Q_1$ ,  $Q_p$  is  $Q_2$ .

**a)** Use the principle of conservation of mass to derive the differential equation for pressure corresponding to the 2 element Windkessel by assuming a compliant tube with inflow  $Q_1$  such the  $A(P) = CP$  where the distal (downstream) flow out of the tube  $Q_2$  is driven across a fluid resistance  $R_p$  such that  $P = RQ_2$ .  $Q_2$  is the flow out of compliant arteries over the peripheral (resistance) vessels.

**b)** Derivation of 4-element Windkessel Model via averaging of the 1-D Navier Stokes === Consider the governing equations for flow through a compliant tube

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0 \quad (1)$$

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left( \frac{Q^2}{A} \right) + \frac{A}{\rho} \frac{\partial P}{\partial x} + K_R \frac{Q}{A} = 0 \quad (2)$$

completed with the constitutive equation  $P = \beta(\sqrt{A} - \sqrt{A_0})$

Show that the average values of  $P$ ,  $A$  and  $Q$ , i.e.  $\bar{P} = \frac{1}{l} \int_0^l P dx$  may be related as

$$k_1 l \frac{d\bar{P}}{dt} = Q_1 - Q_2 \quad (3)$$

and

$$\frac{d\bar{Q}}{dt} \frac{\rho l}{A_0} + \frac{K_R l \rho}{A_0^2} \bar{Q} = P(0) - P(l) \quad (4)$$

by integrating the partial differential equations from  $x = 0$  to  $x = l$ , which are essentially the 4-element Windkessel's equations. To complete the system assume a resistance,  $R_p$ , between  $P(l)$  and venous pressure  $P_v = 0$ .  $K_R$  is the coefficient of viscous friction for a given velocity profile and  $A_0$  is the characteristic cross sectional area.

**Hint.** To find  $\frac{d\bar{P}}{dt}$  consider the chain rule.

Assume the convective term  $\frac{\partial}{\partial x} \left( \frac{Q^2}{A} \right)$  is negligible and that spatial variations in  $A$  are much smoother than variations in  $P$  and  $Q$ , i.e. treat  $A$  as a constant over space.

## Exercise 2: Frequency domain impedance analysis comparison of Windkessel models

Windkessel models are often used to interpret pressure and flow signals in the frequency domain. In this domain the relationship between pressure and flow is given as  $\hat{p} = Z\hat{q}$  where  $\hat{p}$  and  $\hat{q}$  are the fourier transform of  $p$  and  $q$ . Impedances are linear and thus may be added according to the formula  $Z_{eq} = \sum Z_i$  for impedances in series, i.e. same flow through a sequence of elements, and  $Z_{eq}^{-1} = \sum Z_i^{-1}$  for impedances in parallel, i.e. same pressure across a sequence of elements by split of flow. Capacitance has an impedance  $Z_C = \frac{1}{j\omega C}$ , resistance has impedance  $Z_R = R$ , and inductance has impedance  $Z_L = j\omega L$  where  $\omega = 2\pi f$  is the angular frequency.

**a)** Derive the impedance for the 2-Element, 3-Element and 4-Element Windkessel models shown in Figure 2 Using the above relations find equivalent impedances for each of the schematics. Compare the input impedances over frequencies of inputs. What interpretation can you make about the impedances at low and high frequencies?

b) Compare how well these relations fit some data. Below you will find implementation of the functions necessary to fit the impedance of the 2 element Windkessel model to a data set, e.g. './55arteryNetwork\_age19.csv' and './55arteryNetwork\_age75.csv'. If you prefer to use Matlab the attached files have code to get you started. The same python code is also available as a python script.

Use this to compare compliance, characteristic aortic impedance, and characteristic aortic intertance between the 19 year old and the 75 year old.

```
import numpy as np
import scipy as sp
import scipy.optimize
from scipy import io as sio
from matplotlib import pyplot as plt

unit_Pa_to_mmHg = 1/133.32
unit_m3_to_ml = 1e6

# This code may be used to import ./55arteryNetwork_age19.csv or ./55arteryNetwork_age75.csv
csv = np.genfromtxt('./55arteryNetwork_age19.csv', delimiter=";", names=True)
p = csv["P_Pa"]*unit_Pa_to_mmHg
q = csv["Q_m3s"]*unit_m3_to_ml
t = csv["t_s"]
dt = t[1]-t[0]
N = len(t)
T = N*dt
w = np.arange(N//2+1)*2*np.pi/T # Frequencies corresponding to an FFT of the data.

def ZWK2(R,C,w):
    """ Calculate and return impedance of a 2elt WK
    Args:
        R Resistance
        C Compliance
        w the frequencies at which the impedance is calculated.
    """
    Z=R/(1+1j*w*R*C)
    return Z

def WKresidual(C,R,p,q,t):
    """ Calculates error between Windkessel model and Data
    Args:
        C - compliance
        R - Resistance
        p - pressure measurements in time domain
        Qfft - Discrete fourier transform of flows corresponding to p
        w - frequencies of DFT
        t - times of p
    Returns:
        r : the sum of squared errors between pwk and p
    """
    N=len(q)
    dt=t[1]-t[0]
    T = N*dt;
    w = np.arange(N//2+1)*2*np.pi/T
    Qfft = np.fft.rfft(q)
    Pfft = np.fft.rfft(p)
    Zdata=Pfft/Qfft
    Z=ZWK2(R,C,w)
    Pwk_fft=Z*Qfft
    r=np.sum(np.abs(Pfft-Pwk_fft)**2)/N
    return r
```

```

=====
# Guess a compliance value
=====
R=np.mean(p)/np.mean(q) # Total arterial reistance is given by the ratio of
                        # average pressure to average flow. This is not always
                        # the same as the peripheral resistance
C_guess=0.9

#---Compute the pressure pwk0 from impedance based on the guessed compliance value
Qfft=np.fft.rfft(q)
Z = ZWK2(R,C_guess,w)
pwk0 = np.fft.irfft(Qfft*Z,len(q)) #the len(q) has has to be specified when len(q) is odd

def residual(opt_args):
    """Transform WKresidual to a residual function of C,
    which is the only scalar argument"""
    return WKresidual(opt_args[0], R, p, q,t)

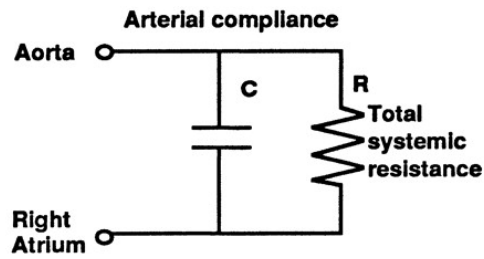
=====
# Estimate the compliance
=====
import scipy.optimize

res = scipy.optimize.minimize(residual, [C_guess,])
#res = scipy.optimize.minimize_scalar(lambda C: WKresidual(C, R, p, q,t))
C=res.x[0]
print ('C=%f'%C)

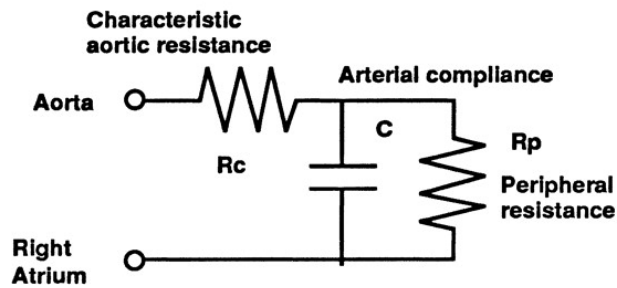
Z = ZWK2(R,C,w)
pwk = np.fft.irfft(Z*Qfft,len(q))

=====
# Plot solutions
=====
plt.figure()
plt.title('Predicted pressures based on estimated compliances')
_ =plt.plot(t,pwk0, label="C guess")
_ =plt.plot(t,pwk, label="C estimated fft")
_ =plt.plot(t,p,'k', label="data")
_ =plt.legend()
plt.xlabel("Time")
plt.ylabel("Pressure")

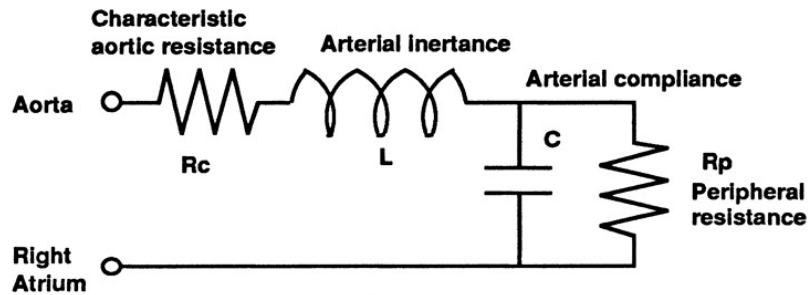
```



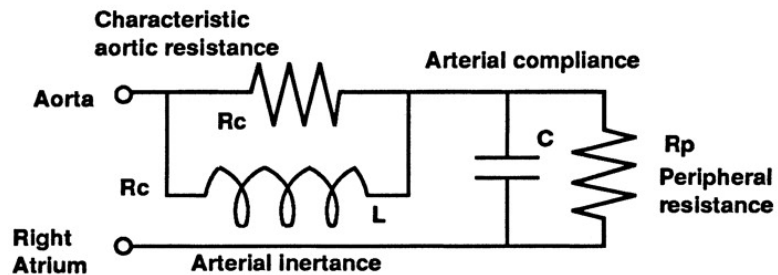
**RC Model**



**RCR Model**



**RLRC Model**



**RLRC2 Model**

Figure 2: Diagrams of various Windkessel type models.