EE263 Homework 2

Summer 2024

2.30. Some standard time-series models. A time series is just a discrete-time signal, *i.e.*, a function from $\mathbf{Z}_+$ into $\mathbb{R}$. We think of $u(k)$ as the value of the signal or quantity u at time (or *epoch*) k. The study of time series predates the extensive study of state-space linear systems, and is used in many fields (*e.g.*, econometrics). Let u and y be two time series (input and output, respectively). The relation (or *time series model*)

$$y(k) = a_0 u(k) + a_1 u(k-1) + \cdots + a_r u(k-r)$$

is called a *moving average (MA) model,* since the output at time k is a weighted average of the previous r inputs, and the set of variables over which we average 'slides along' with time. Another model is given by

$$y(k) = u(k) + b_1 y(k-1) + \cdots + b_p y(k-p).$$

This model is called an *autoregressive (AR) model*, since the current output is a linear combination of (*i.e.*, regression on) the current input and some previous values of the output. Another widely used model is the *autoregressive moving average (ARMA) model*, which combines the MA and AR models:

$$y(k) = b_1 y(k-1) + \cdots + b_p y(k-p) + a_0 u(k) + \cdots + a_r u(k-r).$$

Finally, the problem: Express each of these models as a linear dynamical system with input u and output y. For the MA model, use state

$$x(k) = \begin{bmatrix} u(k-1) \\ \vdots \\ u(k-r) \end{bmatrix},$$

and for the AR model, use state

$$x(k) = \begin{bmatrix} y(k-1) \\ \vdots \\ y(k-p) \end{bmatrix}.$$

You decide on an appropriate state vector for the ARMA model. (There are many possible choices for the state here, even with different dimensions. We recommend you choose a state for the ARMA model that makes it easy for you to derive the state equations.) **Remark:** multi-input, multi-output time-series models (*i.e.*, $u(k) \in \mathbb{R}^m$, $y(k) \in \mathbb{R}^p$) are readily handled by allowing the coefficients a_i, b_i to be matrices.

Solution. In this problem we should find matrices A, B, C and D such that

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned}$$

- *Moving average model.* We need to express $x(k+1)$ linearly in terms of $x(k)$ and $u(k)$. We have

$$x(k) = \begin{bmatrix} u(k-1) \\ u(k-2) \\ \vdots \\ u(k-r) \end{bmatrix}$$

and therefore

$$x(k+1) = \begin{bmatrix} u(k) \\ u(k-1) \\ \vdots \\ u(k+1-r) \end{bmatrix}.$$

Note that

$$x(k+1) = \begin{bmatrix} 0 \\ u(k-1) \\ \vdots \\ u(k+1-r) \end{bmatrix} + \begin{bmatrix} u(k) \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

but

$$\begin{bmatrix} 0 \\ u(k-1) \\ \vdots \\ u(k+1-r) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} u(k-1) \\ u(k-2) \\ \vdots \\ u(k-r) \end{bmatrix}}_{x(k)}$$

so

$$x(k+1) = \underbrace{\begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}}_{A} x(k) + \underbrace{\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{B} u(k).$$

$y(k)$ should be expressed in terms of $x(k)$ and $u(k)$. This is easy from the relation $y(k) = a_0 u(k) + a_1 u(k-1) + \cdots + a_r u(k-r)$ and we get

$$y(k) = \underbrace{\begin{bmatrix} a_1 & a_2 & \cdots & a_r \end{bmatrix}}_{C} x(k) + \underbrace{a_0}_{D} u(k).$$

(Note: the matrix A with ones on its subdiagonal is called a *shift matrix* because it shifts down the elements of the input vector.)

- *Autoregressive model.* In this case

$$x(k) = \begin{bmatrix} y(k-1) \\ y(k-2) \\ \vdots \\ y(k-p) \end{bmatrix}$$

so

$$x(k+1) = \begin{bmatrix} y(k) \\ y(k-1) \\ \vdots \\ y(k+1-p) \end{bmatrix} = \begin{bmatrix} 0 \\ y(k-1) \\ \vdots \\ y(k+1-p) \end{bmatrix} + \begin{bmatrix} y(k) \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Now

$$\begin{bmatrix} 0 \\ y(k-1) \\ \vdots \\ y(k+1-p) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} y(k-1) \\ y(k-2) \\ \vdots \\ y(k-p) \end{bmatrix}}_{x(k)}$$

and

$$\begin{bmatrix} y(k) \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} b_1 & b_2 & \cdots & b_p \\ 0 & 0 & \cdots & 0 \\ \vdots & & & \\ 0 & 0 & \cdots & 0 \end{bmatrix} \underbrace{\begin{bmatrix} y(k-1) \\ y(k-2) \\ \vdots \\ y(k-p) \end{bmatrix}}_{x(k)} + \begin{bmatrix} u(k) \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Thus

$$x(k+1) = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} x(k) + \begin{bmatrix} b_1 & b_2 & \cdots & b_p \\ 0 & 0 & \cdots & 0 \\ \vdots & & & \\ 0 & 0 & \cdots & 0 \end{bmatrix} x(k) + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(k)$$

or

$$x(k+1) = \underbrace{\begin{bmatrix} b_1 & b_2 & b_3 & \cdots & b_p \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}}_{A} x(k) + \underbrace{\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{B} u(k)$$

and

$$y(k) = \underbrace{\begin{bmatrix} b_1 & b_2 & \cdots & b_p \end{bmatrix}}_{C} x(k) + \underbrace{1}_{D} u(k).$$

- *Autoregressive moving average model.* One simple choice for $x(k)$ is

$$x(k) = \left[\begin{array}{c} u(k-1) \\ u(k-2) \\ \vdots \\ u(k-r) \\ \hline y(k-1) \\ y(k-2) \\ \vdots \\ y(k-p) \end{array}\right]$$

and therefore

$$x(k+1) = \left[\begin{array}{c} 0 \\ u(k-1) \\ \vdots \\ u(k+1-r) \\ \hline 0 \\ 0 \\ \vdots \\ 0 \end{array}\right] + \left[\begin{array}{c} u(k) \\ 0 \\ \vdots \\ 0 \\ \hline 0 \\ 0 \\ \vdots \\ 0 \end{array}\right] + \left[\begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \\ \hline y(k) \\ y(k-1) \\ \vdots \\ y(k+1-p) \end{array}\right].$$

For similar reasons to the previous parts

$$\left[\begin{array}{c} 0 \\ u(k-1) \\ u(k-2) \\ \vdots \\ u(k+1-r) \\ \hline 0 \\ \vdots \\ 0 \end{array}\right] = \left[\begin{array}{ccccc|ccccc} 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 & \vdots & & & & \vdots \\ 0 & 1 & 0 & \cdots & 0 & \vdots & & & & \vdots \\ \vdots & & \ddots & \ddots & \vdots & \vdots & & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ \hline 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & & & & \vdots & \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \end{array}\right] x(k)$$

and

$$\left[\begin{array}{c} 0 \\ \vdots \\ 0 \\ \hline y(k) \\ y(k-1) \\ y(k-2) \\ \vdots \\ y(k+1-p) \end{array}\right] = \left[\begin{array}{ccccc|ccccc} 0 & 0 & & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & & & \cdots & 0 & \vdots & & & & \vdots \\ 0 & & & \cdots & 0 & 0 & 0 & & \cdots & 0 \\ \hline a_1 & a_2 & a_3 & \cdots & a_r & b_1 & b_2 & b_3 & \cdots & b_p \\ 0 & & & \cdots & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & & & \cdots & 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots & & \ddots & \ddots & \vdots & \\ 0 & & \cdots & 0 & 0 & 0 & 0 & \cdots & 1 & 0 \end{array}\right] x(k) + \left[\begin{array}{c} 0 \\ \vdots \\ 0 \\ \hline a_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{array}\right] u(k).$$

Thus

$$x(k+1) = \left[\begin{array}{ccccc|ccccc} 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 & \vdots & & & & \vdots \\ 0 & 1 & 0 & \cdots & 0 & \vdots & & & & \vdots \\ \vdots & & \ddots & \ddots & \vdots & \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & 0 & & \cdots & 0 \\ \hline a_1 & a_2 & a_3 & \cdots & a_r & b_1 & b_2 & b_3 & \cdots & b_p \\ 0 & & & \cdots & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & & & \cdots & 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots & & \ddots & \ddots & \vdots & \\ 0 & & \cdots & 0 & 0 & 0 & 0 & \cdots & 1 & 0 \end{array}\right] x(k) + \left[\begin{array}{c} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \hline a_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{array}\right] u(k)$$

and

$$y(k) = \underbrace{\left[\begin{array}{cccc|cccc} a_1 & a_2 & \cdots & a_r & b_1 & b_2 & \cdots & b_p \end{array}\right]}_{C} x(k) + \underbrace{a_0}_{D} u(k).$$

(Note: it is possible to give state-space models with the state dimension smaller than the ones given here. But our selection of state here makes writing the equations easier.)

3.260. Halfspace. Suppose $a, b \in \mathbb{R}^n$ are two given points. Show that the set of points in $\mathbb{R}^n$ that are closer to a than b is a halfspace, *i.e.*:

$$\{x \mid \|x-a\| \leq \|x-b\| \} = \{ x \mid c^\mathsf{T} x \leq d\}$$

for appropriate $c \in \mathbb{R}^n$ and $d \in \mathbb{R}$. Give c and d explicitly, and draw a picture showing a, b, c, and the halfspace.

Solution. It is easy to see geometrically what is going on: the hyperplane that goes right between a and b splits $\mathbb{R}^n$ into two parts; the points closer to a (than b) and the points closer to b (than a). More precisely, the hyperplane is normal to the line through a and b, and intersects that line at the midpoint between a and b. Now that we have the idea, let's try to derive it algebraically. Let x belong to the set of points in $\mathbb{R}^n$ that are closer to a than b. Therefore $\|x-a\| < \|x-b\|$ or $\|x-a\|^2 < \|x-b\|^2$ so

$$(x-a)^\mathsf{T}(x-a) < (x-b)^\mathsf{T}(x-b).$$

Expanding the inner products gives

$$x^\mathsf{T}x - x^\mathsf{T}a - a^\mathsf{T}x + a^\mathsf{T}a < x^\mathsf{T}x - x^\mathsf{T}b - b^\mathsf{T}x + b^\mathsf{T}b$$

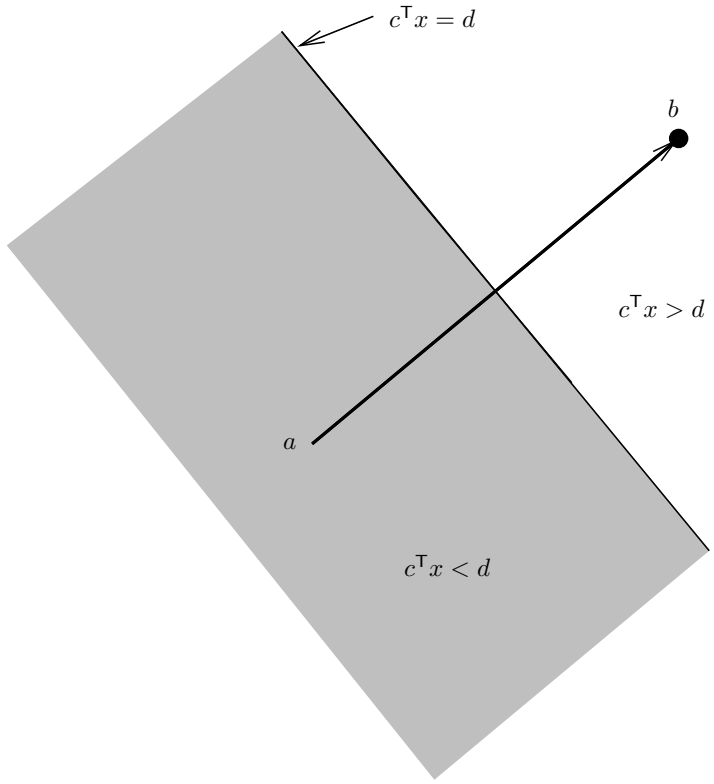
or

$$-2a^\mathsf{T}x + a^\mathsf{T}a < -2b^\mathsf{T}x + b^\mathsf{T}b$$

and finally

$$(b-a)^\mathsf{T}x < \frac{1}{2}(b^\mathsf{T}b - a^\mathsf{T}a). \tag{1}$$

Thus (**??**) is in the form $c^\mathsf{T} x < d$ with $c = b - a$ and $d = \frac{1}{2}(b^\mathsf{T} b - a^\mathsf{T} a)$ and therefore we have shown that the set of points in $\mathbb{R}^n$ that are closer to a than b is a halfspace. Note that the hyperplane $c^\mathsf{T} x = d$ is perpendicular to $c = b - a$.



3.331. Proof of Cauchy-Schwarz inequality. You will prove the Cauchy-Schwarz inequality.

a) Suppose $a \geq 0$, $c \geq 0$, and for all $\lambda \in \mathbb{R}$, $a + 2b\lambda + c\lambda^2 \geq 0$. Show that $|b| \leq \sqrt{ac}$.

b) Given v, $w \in \mathbb{R}^n$ explain why $(v + \lambda w)^\mathsf{T}(v + \lambda w) \geq 0$ for all $\lambda \in \mathbb{R}$.

c) Apply (a) to the quadratic resulting when the expression in (b) is expanded, to get the Cauchy-Schwarz inequality:
$$|v^\mathsf{T} w| \leq \sqrt{v^\mathsf{T} v}\sqrt{w^\mathsf{T} w}.$$

d) When does equality hold?

Solution.

a) If the equation $a + 2b\lambda + c\lambda^2 = 0$ has no real roots (with odd degree) for λ then it never changes sign for $\lambda \in \mathbb{R}$. Since a and c are positive, the value of $a + 2b\lambda + c\lambda^2$ is non-negative at zero and infinity respectively, so the necessary and sufficient condition for $a + 2b\lambda + c\lambda^2$ to be non-negative is the condition for which $a + 2b\lambda + c\lambda^2 = 0$ has no (simple) real roots for λ. Therefore we should have
$$4b^2 - 4ac \leq 0$$
and since $a, c \geq 0$ this gives $|b| \leq \sqrt{ac}$.

b) Clearly $(v+\lambda w)^\mathsf{T}(v+\lambda w) = \|v+\lambda w\|^2$, and the norm of any vector (here $v+\lambda w$) is non-negative. Therefore $(v+\lambda w)^\mathsf{T}(v+\lambda w) \geq 0$ and equality holds when $v+\lambda w = 0$ or $v = -\lambda w$ (*i.e.*, v is a scalar multiple of w.)

c) From the previous part we know that $(v+\lambda w)^\mathsf{T}(v+\lambda w) \geq 0$ and since

$$(v+\lambda w)^\mathsf{T}(v+\lambda w) = v^\mathsf{T}v + 2(v^\mathsf{T}w)\lambda + (w^\mathsf{T}w)\lambda^2,$$

applying the result of problem (**??**) with $a = v^\mathsf{T}v \geq 0$, $b = v^\mathsf{T}w$ and $c = w^\mathsf{T}w \geq 0$ gives

$$|v^\mathsf{T}w| \leq \sqrt{v^\mathsf{T}v}\sqrt{w^\mathsf{T}w}.$$

d) According to (**??**), equality holds if and only if v is a scalar multiple of w. If v is a positive scalar multiple of w, then $v^\mathsf{T}w > 0$ so $|v^\mathsf{T}w| = v^\mathsf{T}w$ and we have $v^\mathsf{T}w = \sqrt{v^\mathsf{T}v}\sqrt{w^\mathsf{T}w}$. If v is a negative scalar multiple of w, then $v^\mathsf{T}w < 0$ and $|v^\mathsf{T}w| = -v^\mathsf{T}w$, so $v^\mathsf{T}w = -\sqrt{v^\mathsf{T}v}\sqrt{w^\mathsf{T}w}$.

3.410. Temperatures in a multi-core processor. We are concerned with the temperature of a processor at two critical locations. These temperatures, denoted $T = (T_1, T_2)$ (in degrees C), are affine functions of the power dissipated by three processor cores, denoted $P = (P_1, P_2, P_3)$ (in W). We make 4 measurements. In the first, all cores are idling, and dissipate 10W. In the next three measurements, one of the processors is set to full power, 100W, and the other two are idling. In each experiment we measure and note the temperatures at the two critical locations.

P_1	P_2	P_3	T_1	T_2
10W	10W	10W	27°	29°
100W	10W	10W	45°	37°
10W	100W	10W	41°	49°
10W	10W	100W	35°	55°

Suppose we operate all cores at the same power, p. How large can we make p, without T_1 or T_2 exceeding 70°?

You must fully explain your reasoning and method, in addition to providing the numerical solution.

Solution. The temperature vector T is an affine function of the power vector P, *i.e.*, we have $T = AP + b$ for some matrix $A \in \mathbb{R}^{2\times 3}$ and some vector $b \in \mathbb{R}^2$. Once we find A and b, we can predict the temperature T for *any* value of P.

The first approach is to (somewhat laboriously) write equations describing the measurements in terms of the elements of A. Let a_{ij} denote the (i,j) entry of A. We can write out

the relations $T = AP + b$ for the 4 experiments listed above as the set of 8 equations

$$\begin{aligned}
10a_{11} + 10a_{12} + 10a_{13} + b_1 &= 27, \\
10a_{21} + 10a_{22} + 10a_{23} + b_2 &= 29, \\
100a_{11} + 10a_{12} + 10a_{13} + b_1 &= 45, \\
100a_{21} + 10a_{22} + 10a_{23} + b_2 &= 37, \\
10a_{11} + 100a_{12} + 10a_{13} + b_1 &= 41, \\
10a_{21} + 100a_{22} + 10a_{23} + b_2 &= 49, \\
10a_{11} + 10a_{12} + 100a_{13} + b_1 &= 35, \\
10a_{21} + 10a_{22} + 100a_{23} + b_2 &= 55.
\end{aligned}$$

Next, we define a vector of unknowns, $x = (a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, b_1, b_2) \in \mathbb{R}^8$. We rewrite the 8 equations above as $Cx = d$, where

$$C = \begin{bmatrix} 10 & 10 & 10 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 10 & 10 & 10 & 0 & 1 \\ 100 & 10 & 10 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 100 & 10 & 10 & 0 & 1 \\ 10 & 100 & 10 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 10 & 100 & 10 & 0 & 1 \\ 10 & 10 & 100 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 10 & 10 & 100 & 0 & 1 \end{bmatrix} \qquad d = \begin{bmatrix} 27 \\ 29 \\ 45 \\ 37 \\ 41 \\ 49 \\ 35 \\ 55 \end{bmatrix}.$$

We solve for x as $x = C^{-1}d$. (It turns out that C is invertible.) Putting the entries of x into the appropriate places in A and b, we have

$$A = \begin{bmatrix} 0.200 & 0.156 & 0.089 \\ 0.089 & 0.222 & 0.289 \end{bmatrix}, \qquad b = \begin{bmatrix} 22.6 \\ 23.0 \end{bmatrix}.$$

At this point we can predict T for any P (assuming we trust the affine model).

Substituting $P = (p, p, p)$ into $T = AP + b$, we get

$$T_1 = 0.444p + 22.6, \qquad T_2 = 0.600p + 23.0.$$

Both of these temperatures are increasing in p (it would be quite surprising if this were not the case). The value of p for which $T_1 = 70$ is $p = (70 - 22.6)/0.444 = 106.8$W. The value of p for which $T_2 = 70$ is $p = (70 - 23)/0.6 = 78.3$W. Thus, the maximum value of p for which both temperatures do not exceed 70° is $p = 78.3$W.

```
%problem data
C = [10 10 10 0 0 0 1 0
0 0 0 10 10 10 0 1
100 10 10 0 0 0 1 0
0 0 0 100 10 10 0 1
10 100 10 0 0 0 1 0
```

```
0 0 0 10 100 10 0 1
10 10 100 0 0 0 1 0
0 0 0 10 10 100 0 1];
d = [27 29 45 37 41 49 35 55]';

%Find affine model
x = C\d;
A = reshape(x(1:6), 3, 2)';
b = x(7:8)';

%Find maximum power
p1 = (70 - b(1))/sum(A(1,:));
p2 = (70 - b(2))/sum(A(2,:));
p = min(p1, p2)
```

Alternative solution. Another way of solving this problem is to directly exploit the fact that T is an affine function of P. This means that if we form any linear combination of the power vectors used in the experiment, *with the coefficients summing to one*, the temperature vector will also be the same linear combination of the temperatures.

By averaging the last three experiments we find if the powers are $P = (40, 40, 40)$, then the temperature vector is $T = (40.33, 47.00)$. (Note that this is really a prediction, based on the observed experimental data and the affineness assumption; it's not a new experiment!)

Now we form a new power vector of the form

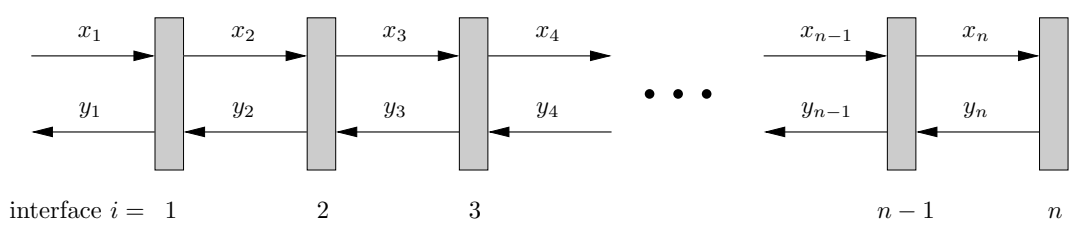
$$P = (1-\theta)(10, 10, 10) + \theta(40, 40, 40) = (10 + 30\theta, 10 + 30\theta, 10 + 30\theta),$$

where $\theta \in \mathbb{R}$. The coefficients $1 - \theta$ and θ sum to one, so since T is affine, we find that the corresponding temperature vector is

$$T = (1-\theta)(27, 29) + \theta(40.33, 47.00) = (27 + 13.33\theta, 29 + 18\theta),$$

just as above. The first coefficient hits 70 at $\theta = 3.226$; the second coefficient hits 70 at $\theta = 2.278$. Thus, θ can be as large as $\theta = 2.278$. This corresponds to the powers $P = (78.3, 78.3, 78.3)$.

3.490. Layered medium. In this problem we consider a generic model for (incoherent) transmission in a layered medium. The medium is modeled as a set of n layers, separated by n dividing interfaces, shown as shaded rectangles in the figure below.



We let $x_i \in \mathbb{R}$ denote the right-traveling wave amplitude in layer i, and we let $y_i \in \mathbb{R}$ denote the left-traveling wave amplitude in layer i, for $i = 1, \ldots, n$. The right-traveling wave in the

first layer is called the incident wave, and the left-traveling wave in the first layer is called the reflected wave. The scattering coefficient for the medium is defined as the ratio $S = y_1/x_1$ (assuming $x_1 \neq 0$).

The right- and left-traveling waves on each side of an interface are related by transmission and reflection. The right-traveling wave of amplitude x_i contributes the amplitude $t_i x_i$ to x_{i+1}, where $t_i \in [0, 1]$ is the transmission coefficient of the ith interface. It also contributes the amplitude $r_i x_i$ to y_i, the left-traveling wave, where $r_i \in [0, 1]$ is the reflection coefficient of the ith interface. We will assume that the interfaces are symmetric, so the left-traveling wave with amplitude y_{i+1} contributes the wave amplitude $t_i y_{i+1}$ to y_i (via transmission) and wave amplitude $r_i y_{i+1}$ to x_{i+1} (via reflection). Thus we have

$$x_{i+1} = t_i x_i + r_i y_{i+1}, \quad y_i = r_i x_i + t_i y_{i+1}, \quad i = 1, 2, \ldots, n-1.$$

We model the last interface as totally reflective, which means that $y_n = x_n$.

a) Explain how to find the scattering coefficient S, given the transmission and reflection coefficients for the first $n-1$ layers.

b) Carry out your method for a medium with $n = 20$ layers, and $t_i = 0.96$, $r_i = 0.02$ for $i = 1, \ldots, n-1$. Plot the left- and right-traveling wave amplitudes x_i, y_i versus i, and report the value of S you find.

   Hint: You may find the matlab function `diag(x,k)` useful.

c) *Fault location.* A fault in interface k results in a reversal: $t_k = 0.02$, $r_k = 0.96$, with all other interfaces having their nominal values $t_i = 0.96$, $r_i = 0.02$. You measure the scattering coefficient $S = S^{\text{fault}}$ with the fault (but you don't have access to the left- or right-traveling waves with the faulted interface). Explain how to find which interface is faulted. Carry out your method with $S^{\text{fault}} = 0.70$. You may assume that the last (fully reflective) interface is not faulty. Be sure to give the value of k that is most consistent with the measurement.

Solution.

a) The equations are homogeneous in x and y, so we can find the scattering coefficient by fixing the incident wave amplitude as $x_1 = 1$; the scattering coefficient is then $S = y_1/x_1 = y_1$. The transmission and reflection conditions lead to the following set of $2n$ (homogeneous) equations:

$$\begin{aligned}
x_1 &= 1 \\
x_2 &= t_1 x_1 + r_1 y_2 \\
x_3 &= t_2 x_2 + r_2 y_3 \\
&\vdots \\
x_{n-1} &= t_{n-2} x_{n-2} + r_{n-2} y_{n-1} \\
x_n &= t_{n-1} x_{n-1} + r_{n-1} y_n \\
y_1 &= r_1 x_1 + t_1 y_2
\end{aligned}$$

$$
\begin{aligned}
y_2 &= r_2 x_2 + t_2 y_3 \\
y_3 &= r_3 x_3 + t_3 y_4 \\
&\vdots \\
y_{n-1} &= r_{n-1} x_{n-1} + t_{n-1} y_n \\
y_n &= x_n
\end{aligned}
$$

for the $2n$ unknowns $x_1, \ldots, x_n, y_1, \ldots, y_n$. These can be rewritten in block matrix form as

$$
\underbrace{\begin{bmatrix} A & B \\ C & D \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{z} = \underbrace{\begin{bmatrix} e_1 \\ 0 \end{bmatrix}}_{b},
$$

where $e_1, 0 \in \mathbb{R}^n$ and $A, B, C, D \in \mathbb{R}^{n \times n}$ are defined as

$$
A = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ t_1 & -1 & 0 & \cdots & 0 & 0 \\ 0 & t_2 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 0 \\ 0 & 0 & 0 & \cdots & t_{n-1} & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & r_1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & r_1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & r_{n-2} & 0 \\ 0 & 0 & 0 & \cdots & 0 & r_{n-1} \end{bmatrix},
$$

$$
C = \begin{bmatrix} r_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & r_2 & 0 & \cdots & 0 & 0 \\ 0 & 0 & r_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & r_{n-1} & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} -1 & t_1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & t_2 & \cdots & 0 & 0 \\ 0 & 0 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & t_{n-1} \\ 0 & 0 & 0 & \cdots & 0 & -1 \end{bmatrix}.
$$

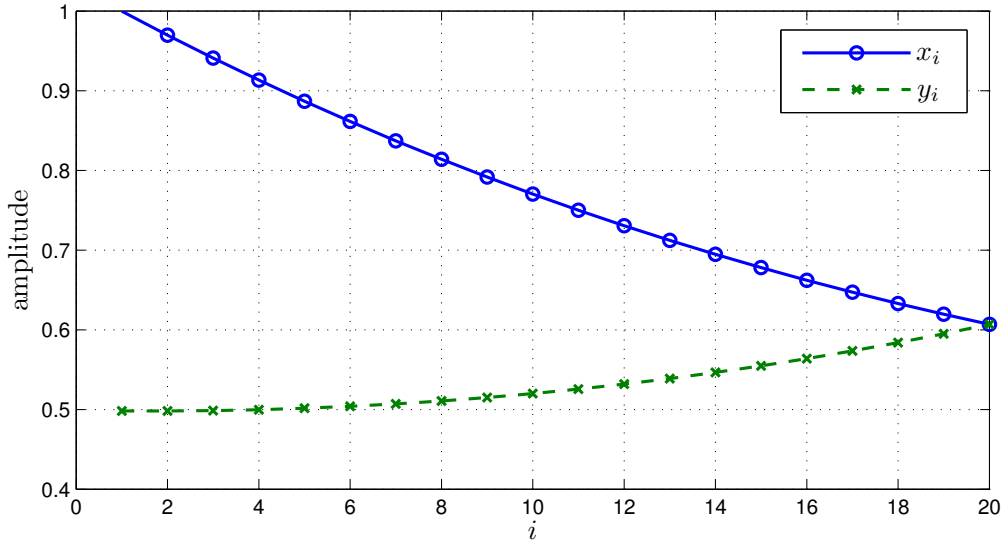
To find the scattering coefficinent, we solve the system $Fz = b$ for $z \in \mathbb{R}^{2n}$, and let $S = z_{n+1}$.

b) The following code solves the system and predicts $S = 0.4982$.

```
n = 20;
t = 0.96*ones(n-1,1);
r = 0.02*ones(n-1,1);

A = diag([1; -ones(n-1,1)]) + diag(t, -1);
B = diag([0; r]);
C = diag([r; 1]);
D = diag(-ones(n,1)) + diag(t, 1);
b = [1; zeros(2*n-1,1)];
F = [A B; C D];
z = F \ b;
S = z(n+1);
```

```
plot(1:n, z(1:n), 'o-', 1:n, z(n+1:end), 'x--');
xlabel('i'); ylabel('amplitude');
legend('xi', 'yi');
grid on;
```



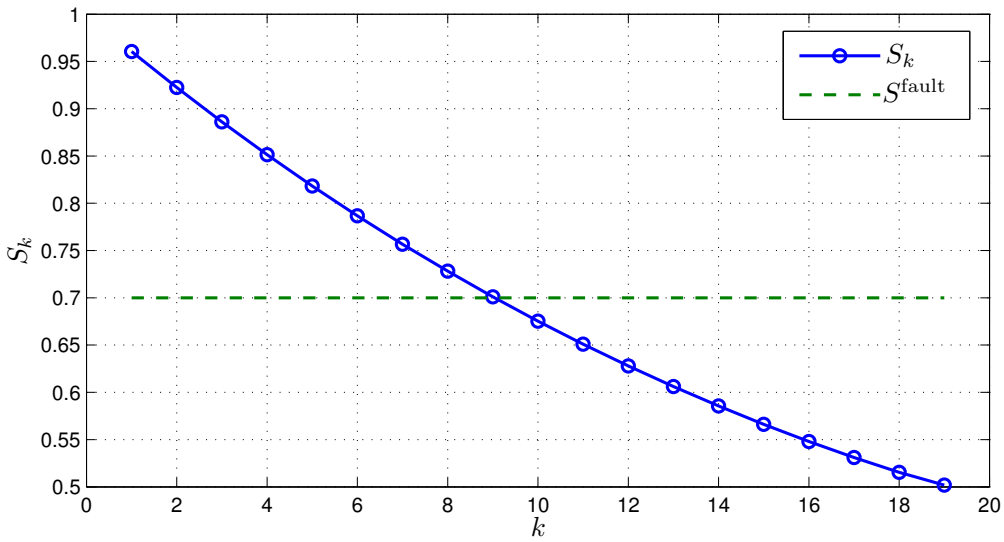
c) To figure out which interface is faulty, we find S_k, the reflection coefficient resulting from setting $t_k = 0.02$, $r_k = 0.96$ for each $k = 1, \ldots, n-1$. The following code computes and plots S_k for each k. From the graph, interface $k = 9$ is most consistent with the measurement $S^{\text{fault}} = 0.70$.

```
n = 20;
Sfault = 0.70;
S = zeros(n-1,1);

for k=1:n-1
    t = 0.96*ones(n-1,1);
    r = 0.02*ones(n-1,1);
    t(k) = 0.02;
    r(k) = 0.96;

    A = diag([1; -ones(n-1,1)]) + diag(t, -1);
    B = diag([0; r]);
    C = diag([r; 1]);
    D = diag(-ones(n,1)) + diag(t, 1);
    b = [1; zeros(2*n-1,1)];
    F = [A B; C D];
    z = F \ b;
    S(k) = z(n+1);
end
```

```
plot(1:n-1, S, 'o-', 1:n-1, Sfault*ones(n-1,1), '--');
xlabel('k'); ylabel('Sk');
legend('Sk', 'Sfault');
grid on;
```



Alternative solution. Some people found an equivalent recursive way to compute the left- and right- traveling wave amplitudes, for which we gave full credit. From the given homogeneous relations among the x_i and y_i, we rewrite the x_i and y_i as a function of x_{i+1} and y_{i+1}.

$$\begin{aligned} x_{i+1} &= t_i x_i + r_i y_{i+1} \\ y_i &= r_i x_i + t_i y_{i+1}, \quad i = 1, \ldots, n-1. \end{aligned}$$

Solving the first equation for x_i and substituting into the second equation, gives the following recursion:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{1}{t_i} & -\frac{r_i}{t_i} \\ \frac{r_i}{t_i} & \frac{t_i^2 - r_i^2}{t_i} \end{bmatrix}}_{A_i} \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix}, \quad i = 1, \ldots, n-1,$$

where we assume that $t_i \neq 0$, $i = 1, \ldots, n-1$ and the arbitrary initial condition $x_n = y_n = 1$. Expanding the recursion gives

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = A_1 A_2 \cdots A_{n-1} \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$

from where we get $S = y_1/x_1$. Finally, to find the faulted interface, we replace A_k by A^{fault}, where

$$A^{\text{fault}} = \begin{bmatrix} \frac{1}{(0.02)} & -\frac{(0.96)}{(0.02)} \\ \frac{(0.96)}{(0.02)} & \frac{(0.02)^2-(0.96)^2}{(0.02)} \end{bmatrix},$$

and compute the corresponding reflection coefficient S_k. The value of k for which the corresponding $|S_k - S^{\text{fault}}|$ has the smallest value is the faulted interface.