Networks II – Worksheet One

Richard G. Clegg,
richard@richardclegg.org

March 10, 2006

Basic Networking

Assume for simplicity throughout this section that 1KB = 1000 bytes (octets) – note that in fact, 1KB = 1024 bytes. Similarly assume 1MB = 1000KB, 1GB = 1000MB and so on. (Note that Kb refers to kilobits and KB refers to kilobytes).

Question 1. Yann and Richard want to copy 6GB of files from their homes to their office. Richard elects to use a modem and transfers the files at 512Kb/sec (assume that the modem operates at this data rate constantly and only data is sent — ignore the effects of packet headers and so on). Yann elects to copy the files to a writable CD (capacity 600MB). It takes him 15 minutes to travel home to his office, 15 minutes to write a CD (copying from files on disk) and 5 minutes to read a CD (copying files on the CD to the disk). If both start off in their office, how long do each take to copy all the data? What is the effective bandwidth in KB/sec for each?

Answer 1. 6GB = 6,000,000KB.

512Kb/sec (a typical speed for an ADSL modem) is 64KB/sec. Therefore it will take the modem 93,750 seconds (26 hours).

6GB will take 10 CDs to copy. The CD copying process takes 50 minutes (8 hours). The total time spent is 500 minutes. The bandwidth is therefore 200KB/sec.

Question 2. List two differences between the OSI reference model and the TCP/IP model. List two ways in which they are the same.

Answer 2. Many answers are possible — the two most obvious differences are the lack of session and presentation layers in TCP/IP and the combination of the Logical and Network layer into the Host-to-Network Layer. Perhaps the two most obvious similarities are the Transport layer and the Application layer.

Question 3. Which layer of the OSI model handles:

1. Breaking data into packets.
2. Determining which route to use through a subnet.

Answer 3. The transport layer and the network layer respectively.

Question 4. Consider the following hosts:

- `wobbegong.spurious.ac.uk 128.100.59.16`

- greatwhite.spurious.ac.uk 128.100.59.17
- cookiecutter.spurious.ac.uk 128.100.63.25
- mako.spurious.ac.uk 128.100.62.25
- tiger.spurious.ac.uk 128.100.1.52

Which hosts are on the same subnet given the following netmasks:

1. 255.255.255.224
2. 255.255.255.0
3. 255.255.254.0
4. 255.255.192.0
5. 255.255.193.0

Why is the last netmask illegal?

Answer 4. By far the least effort way to answer this question is with a C program (or similar in another language).

```
#include <stdio.h>

enum {
    NO_NETWORKS= 5,
    NO_NETMASKS= 5,
    BIN_IP_LEN= 33  /* Length of IP address in binary + 1 for '\0' */
};

void tobinary (char binno[BIN_IP_LEN], int ip_addr[]);
/* Convert an integer IP address to binary */

void binary_and (char answer[], char bin1[], char bin2[]);
/* Perform a bitwise AND on a network and a netmask */

int main()
{
    int i,j;
    char binmask[BIN_IP_LEN]; /* Holds a netmask in binary as a string */
    char binnet[BIN_IP_LEN]; /* Holds an IP in binary as a string */
    char bitand[BIN_IP_LEN]; /* Holds the and of the two in binary  */
    int networks[NO_NETWORKS][4]={  /* The networks to be looked at */
       128,100,59,16,
       128,100,59,17,
       128,100,63,25,
       128,100,62,25,
       128,100,1,52
    };
    int netmasks[NO_NETMASKS][4]= {  /* The netmasks to be used */
       255,255,255,224,
       255,255,255,0,
```

```
        255,255,254,0,
        255,255,192,0,
        255,255,193,0
    };
    for  (i= 0; i < NO_NETMASKS; i++) {
        tobinary (binmask, netmasks[i]);
        printf ("Using netmask %03d.%03d.%03d.%03d = %s\n",
          netmasks[i][0], netmasks[i][1],netmasks[i][2],
          netmasks[i][3], binmask);
        for (j= 0; j < NO_NETWORKS; j++) {
            tobinary(binnet, networks[j]);
            binary_and (bitand, binnet, binmask);
            printf ("      Network %03d.%03d.%03d.%03d = %s\n",
              networks[j][0], networks[j][1],
              networks[j][2],networks[j][3], binnet);
            printf ("                        becomes %s\n\n",
              bitand);
        }
    }
    return 0;
}

void tobinary (char binary[BIN_IP_LEN], int ip_addr[])
/* Convert a 4 integer IP address to a string of binary */
{
    int j=0;  /* Position within string */
    int i, k;
    int tmp_ip;
    for (i= 0; i < 4; i++) { /* loop over the 4 bytes of the IP address*/
        tmp_ip= ip_addr[i];
        for (k= 128; k >= 1; k= k/2) {  /* Subtract off powers in turn */
            if (tmp_ip >= k) {
                tmp_ip -= k;
                binary[j]= '1';
            } else {
                binary[j]= '0';
            }
            j++;
        }
    }
    binary[j]= '\0';
}

void binary_and (char answer[], char bin1[], char bin2[])
/* fills answer with a binary string which is a bitwise and of bin1 and bin2 */
{
    int i= 0;
    while (bin1[i] != '\0' && bin2[i] != '\0') {
        if (bin1[i] == '1' && bin2[i] == '1') {
            answer[i] = '1';
        } else {
            answer[i] = '0';
        }
```

```
        i++;
    }
    answer[i]= '\0';
}
```

This produces the following output:

```
Using netmask 255.255.255.224 = 11111111111111111111111111100000
      Network 128.100.059.016 = 10000000011001000011101100010000
                        becomes 10000000011001000011101100000000

      Network 128.100.059.017 = 10000000011001000011101100010001
                        becomes 10000000011001000011101100000000

      Network 128.100.063.025 = 10000000011001000011111100011001
                        becomes 10000000011001000011111100000000

      Network 128.100.062.025 = 10000000011001000011111000011001
                        becomes 10000000011001000011111000000000

      Network 128.100.001.052 = 10000000011001000000000100110100
                        becomes 10000000011001000000000100100000

Using netmask 255.255.255.000 = 11111111111111111111111100000000
      Network 128.100.059.016 = 10000000011001000011101100010000
                        becomes 10000000011001000011101100000000

      Network 128.100.059.017 = 10000000011001000011101100010001
                        becomes 10000000011001000011101100000000

      Network 128.100.063.025 = 10000000011001000011111100011001
                        becomes 10000000011001000011111100000000

      Network 128.100.062.025 = 10000000011001000011111000011001
                        becomes 10000000011001000011111000000000

      Network 128.100.001.052 = 10000000011001000000000100110100
                        becomes 10000000011001000000000100000000

Using netmask 255.255.254.000 = 11111111111111111111111000000000
      Network 128.100.059.016 = 10000000011001000011101100010000
                        becomes 10000000011001000011101000000000

      Network 128.100.059.017 = 10000000011001000011101100010001
                        becomes 10000000011001000011101000000000

      Network 128.100.063.025 = 10000000011001000011111100011001
                        becomes 10000000011001000011111000000000

      Network 128.100.062.025 = 10000000011001000011111000011001
                        becomes 10000000011001000011111000000000
```

```
       Network 128.100.001.052 = 10000000011001000000000100110100
                         becomes 10000000011001000000000000000000

Using netmask 255.255.192.000 = 11111111111111111100000000000000
       Network 128.100.059.016 = 10000000011001000011101100010000
                         becomes 10000000011001000000000000000000

       Network 128.100.059.017 = 10000000011001000011101100010001
                         becomes 10000000011001000000000000000000

       Network 128.100.063.025 = 10000000011001000011111100011001
                         becomes 10000000011001000000000000000000

       Network 128.100.062.025 = 10000000011001000011111000011001
                         becomes 10000000011001000000000000000000

       Network 128.100.001.052 = 10000000011001000000000100110100
                         becomes 10000000011001000000000000000000

Using netmask 255.255.193.000 = 11111111111111111100000100000000
       Network 128.100.059.016 = 10000000011001000011101100010000
                         becomes 10000000011001000000000100000000

       Network 128.100.059.017 = 10000000011001000011101100010001
                         becomes 10000000011001000000000100000000

       Network 128.100.063.025 = 10000000011001000011111100011001
                         becomes 10000000011001000000000100000000

       Network 128.100.062.025 = 10000000011001000011111000011001
                         becomes 10000000011001000000000000000000

       Network 128.100.001.052 = 10000000011001000000000100110100
                         becomes 10000000011001000000000100000000
```

We can see that under mask 1, wobbegong and greatwhite are the same.

Under mask 2 wobbegong and greatwhite are the same.

Under mask 3 wobbegong and greatwhite are the same and cookiecutter and mako are the same.

Under mask 4 all machines are the same.

Under mask 5 all machines but mako are the same. Mask 5 is illegal because it has holes in it (it is not a continuous sequence of 1s followed by 0s).

Question 5. A 10MB message is transferred using TCP/IP. The maximum packet size is 1KB including headers. How many packets must be sent? What fraction of the bandwidth was wasted on headers?

Answer 5. From lecture notes, the total header size is 48 bytes (TCP and IP header) therefore the data size is 952 bytes. 10,505 packets must be sent. Therefore a total of 10,000,000 bytes of data are sent and 10,505,000 bytes in total (data plus headers). Wasted bandwidth is 4.8% (although this could have been calculated simply by observing that in the average packet there is 4.8% ofthe packet as a header.

Note: In fact, it has been pointed out to me that the total header size of 48 bytes assumes TCP and IP header both have 4 bytes of options. Whereas my lecture notes make it seem like this is compulsary, in fact it is more common that there are no options and therefore each header would be 20 bytes and the total header length would be 40 bytes. I have marked this answer correct too.

Basic Queuing Theory and Poisson Processes

Question 6. Two communication nodes 1 and 2 send files to another node 3. Files from 1 and 2 require, on average, R_1 and R_2 time units for transmission respectively. Node 3 processes a file from node $i (i = 1, 2)$ in an average of P_i time units and then requests another file from either node 1 or 2 according to some rule (which is left unspecified). If λ_i is the throughput of node i in files sent per unit time then what is the region of all feasible throughput pairs (λ_1, λ_2)? (That is to say, which values of λ_1 and λ_2 will allow the network to continue to function in the way described

Answer 6. Consider the system from when node 3 requests a file to when a file has been processed by node 3 and exits the system (that is, ignore files waiting to be requested and processed). The total time spent from being requested to exiting node three is $R_1 + P_1$ for files from node 1 and $R_2 + P_2$ for files from node 2. The proportion of files from node 1 is $\lambda_1/(\lambda_1 + \lambda_2)$. Therefore, it is clear that the average time T from request to exit for all files through node 3 is given by:

$$T = \frac{\lambda_1(R_1 + P_1) + \lambda_2(R_2 + P_2)}{\lambda_1 + \lambda_2}$$

The total input to node 3 is $\lambda_1 + \lambda_2$. Therefore, from Little's theorem we get:

$$N = \lambda T = (\lambda_1 + \lambda_2)\frac{\lambda_1(R_1 + P_1) + \lambda_2(R_2 + P_2)}{\lambda_1 + \lambda_2} = \lambda_1(R_1 + P_1) + \lambda_2(R_2 + P_2)$$

From the specification of node 3, however, we know that $0 < N \leq 1$ since node 3 only requests and processes one file at a time. Therefore:

$$1 \geq \lambda_1(R_1 + P_1) + \lambda_2(R_2 + P_2)$$

which defines a feasible region for λ_1 and λ_2.

Question 7. Consider K independent sources of packets where the interarrival times of each source are exponentially distributed (that is each source is a Poisson process) with the kth source having mean λ_k. If these packet streams are merged (assuming no delay in doing so), prove that the K independent sources form a Poisson process with mean $\lambda = \lambda_1 + \lambda_2 + \cdots + \lambda_K$.

Answer 7. It is sufficient to prove that two processes with means λ_1 and λ_2 sum to a single process with mean $\lambda_1 + \lambda_2$ since the general result follows trivially from this (we could add a third process λ_3 to the combined process and so on).

Call the two processes X_1 and X_2 where, in a time period t we have:

$$Pr\{X_1 = k\} = \frac{(\lambda_1 t)^k e^{-\lambda_1 t}}{k!}$$

and a similar equation for X_2 but with parameter λ_2. Now, for the combined process $X = X_1 + X_2$ we must have:

$$Pr\{X = k\} = \sum_{i=0}^{k} Pr\{X_1 = i\} Pr\{X_2 = k - i\}$$

since, if $X = k$ and $X_1 = i$ $(0 \leq i \leq k)$ then $X_2 = k - i$. Substituting from above the expressions for Poisson Processes we get:

$$Pr\{X = k\} = \sum_{i=0}^{k} \frac{(\lambda_1 t)^i e^{-\lambda_1 t} (\lambda_2 t)^{(k-i)} e^{-\lambda_2 t}}{i!(k-i)!}$$

First take out the constant terms and dividing top and bottom by $k!$:

$$Pr\{X = k\} = \frac{e^{-\lambda_1 t} e^{-\lambda_2 t}}{k!} \sum_{i=0}^{k} \frac{k!(\lambda_1 t)^i (\lambda_2 t)^{(k-i)}}{i!(k-i)!}$$

Now we note that $\frac{k!}{i!(k-i)!} = \binom{k}{i}$ and therefore:

$$\mathbb{P}\left[X = k \right] = \frac{e^{-(\lambda_1 + \lambda_2)t}}{k!} \sum_{i=0}^{k} (\lambda_1 t)^i (\lambda_2 t)^{(k-i)} \binom{k}{i}$$

now we can see that the sum is simply the binomial expansion of $(\lambda_1 t + \lambda_2 t)^k$. Therefore:

$$\mathbb{P}\left[X = k \right] = \frac{e^{-(\lambda_1 + \lambda_2)t} (\lambda_1 t + \lambda_2 t)^k}{k!}$$

which is the Poisson process with mean $\lambda_1 + \lambda_2$ as required.

Question 8. A packet source either emits or does not emit a single packet every microsecond with the probability of emiting a packet being p. Let $(X_j)_{j \geq 0}$ be a timeseries (of zeros and ones) representing this packet stream. A counter records the number of packets seen every N microseconds — $S_N = X_1 + X_2 + \cdots + X_N$. If N has a Poisson distribution with mean λ show that the number of packets counted (S_N) has a Poisson distribution with mean λp.

Answer 8. First, let us write down the distribution for N.

$$\mathbb{P}\left[N = k \right] = \frac{e^{-\lambda} \lambda^k}{k!} \tag{1}$$

We want to find the probability that $S_N = k$. In this case, we cannot find it directly but must use the various probabilities for N.

$$\mathbb{P}\left[S_N = k \right] = \sum_{i=k}^{\infty} \mathbb{P}\left[S_N = k, N = i \right] = \sum_{i=k}^{\infty} \mathbb{P}\left[S_N = k | N = i \right] \mathbb{P}\left[N = i \right]. \tag{2}$$

Now, if we know that $N = i and S_N = k$ where, naturally, $(i \geq k)$ then we know that, of i possible packets, k packets were generated and $i - k$ were not.

$$\mathbb{P}[\, S_N = k | N = i\,] = \binom{i}{k} p^k (1-p)^{i-k} \tag{3}$$

where $\binom{i}{k}$ is the standard expression for i choose k. Substituting what we have so far we get:

$$\mathbb{P}[\, N = k\,] = \sum_{i=k}^{\infty} \binom{i}{k} p^k (1-p)^{i-k} \frac{e^{-\lambda}\lambda^i}{i!} \tag{4}$$

now expanding $\binom{k}{i}$ as the binomial coefficient in the standard way and, using our usual $q = (1-p)$ and taking constants out of the sum we get

$$= p^k e^{-\lambda} \sum_{i=k}^{\infty} \frac{i! q^{i-k} \lambda^i}{k!(i-k)!i!} \tag{5}$$

Taking some more factors out we get

$$= \frac{p^k e^{-\lambda} \lambda^k}{k!} \sum_{i=k}^{\infty} \frac{q^{i-k}\lambda^{i-k}}{(i-k)!} \tag{6}$$

Now we change the initial point of the sum and add a factor $e^{-q\lambda}$ on the top and bottom of our fraction

$$= \frac{p^k e^{-\lambda} \lambda^k}{k! e^{-q\lambda}} \sum_{i=0}^{\infty} \frac{e^{-q\lambda}(q\lambda)^i}{i!} \tag{7}$$

We recognise instantly that the part inside the sum is a Poisson distribution with parameter $q\lambda$ and therefore the sum from 0 to ∞ is 1. Tidying up we get

$$= \frac{(p\lambda)^k e^{-\lambda} e^{q\lambda}}{k!} \tag{8}$$

And finally substituting back $q = 1 - p$ we get

$$= \frac{(p\lambda)^k e^{-\lambda} e^{(\lambda - p\lambda)}}{k!} \tag{9}$$

finally giving

$$\mathbb{P}[\, S_N = k\,] = \frac{(p\lambda)^k e^{-p\lambda}}{k!} \tag{10}$$

which is a Poisson process with parameter $p\lambda$ as required.

Question 9. The input to a router is a Poisson Process with parameter λ. The router sends packets down one of its K output streams. Each packet arriving at the router is assigned without delay to a stream chosen at random with a probability p_i that the packet is assigned to stream i (naturally, $\sum_{i=1}^{K} p_i = 1$). Prove that the ith stream is Poisson with parameter $\lambda_i = p_i\lambda$.

Answer 9. This is answered in exactly the same way as the previous question. We simply notice that the situation is that in the next question. Let us assume that N is the number of packets produced at the router in some period τ. Clearly N will have a Poisson distribution (it is a sample from a Poisson process with mean $\lambda\tau$. The probability that any of these packets are observed at the ith stream is p_i. From this we can calculate that the distribution of N_i the number of packets seen at the ith stream in the period τ:

$$\mathbb{P}[N_i = k] = \sum_{i=k}^{\infty} \mathbb{P}[N_i = k, N = i] = \sum_{i=k}^{\infty} \mathbb{P}[N_i = k | N = i] \mathbb{P}[N = i] \tag{11}$$

the proof now follows as above with p_i taking the role of p.

Question 10. Packets arrive at a router at a rate of 25 per second. Packets take 5 milliseconds to process. 50 percent of packets are considered urgent and immediately forwarded without queuing. The remaining 50 percent are queued for an average of 20 milliseconds before forwarding. What is the average number of packets in the system (both being processed and being queued)?

Answer 10. The average time in the system $T = 0.5(20 + 5) + 0.5(5) = 15$ milliseconds. The average arrival rate $\lambda = 0.025$ packets/millisecond. Therefore, from Little's Theorem the average number of packets in the system is $N = \lambda T = 0.375$ packets.

Basic Markov Chains

Question 11. Suppose that $(X_n)_{n\geq 0}$ is Markov $(\lambda, \mathbf{P})$. If $Y_n = X_{kn}$ where $(k \in \mathbb{N})$ then show that $(Y_n)_{n\geq 0}$ is Markov $(\lambda, \mathbf{P}^k)$.

Answer 11. Firstly, we have:

$$Y_0 = X_0 = \lambda$$

Next we can say that since $X_{n+k} = \mathbf{P}^k X_n$ we have:

$$Y_{n+1} = X_{k(n+1)} = X_{kn+k} = \mathbf{P}^k X_{kn}$$

And therefore:

$$Y_{n+1} == \mathbf{P}^k Y_n$$

Therefore, $(Y_n)_{n\geq 0}$ is Markov $(\lambda, \mathbf{P}^k)$ as required.

Question 12. Show that a point inside or on an equilateral triangle of unit height can be used to represent a distribution in a three state Markov chain in the sense that the sum of the distances of a given point in the triangle to the three sides must always equal unity.

Answer 12. There are many ways to answer this question, the simplest is a graphical solution from similar triangles.

Question 13. A flea hops about the vertices of a triangle. It is twice as likely to hop clockwise as anti-clockwise. Write down $\mathbf{P}$ for the Markov chain. What is the probability that it returns to its starting point after n hops?

Hint: $-1/2 \pm \frac{i}{2.\sqrt{3}} = \left(-\frac{1}{\sqrt{3}}\right) e^{\pm \frac{i\pi}{6}}$

Answer 13. The transition matrix is:

$$\mathbf{P} = \begin{bmatrix} 0 & \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & 0 & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} & 0 \end{bmatrix}$$

Therefore to get eigenvalues we must solve:

$$\begin{vmatrix} -\lambda & \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & -\lambda & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} & -\lambda \end{vmatrix} = 0$$

This gives us:

$$-\lambda^3 + \frac{2}{3}\lambda + \frac{1}{3} = 0$$

It is clear we can extract $\lambda = 1$ as a root so removing a factor of $(\lambda - 1)$ we are left with:

$$\lambda^2 + \lambda + 1/3 = 0$$

Using the quadratic formula we get two solutions with the form:

$$\lambda = -1/2 \pm \frac{i}{2.\sqrt{3}} = \left(-\frac{1}{\sqrt{3}}\right) e^{\pm \frac{i\pi}{6}}$$

Using the rule that complex conjugate pairs appear in the general form solution as $AC^n \cos(n\theta) + BC^n \sin(n\theta)$, this leads us to the general form for answers:

$$p_{ij}^{(n)} = A(1)^n + B\left(-\frac{1}{\sqrt{3}}\right)^n \cos(n\pi/6) + C\left(-\frac{1}{\sqrt{3}}\right)^n \sin(n\pi/6)$$

From observation of the matrix we can see that $p_{11}^{(0)} = 1$, $p_{11}^{(1)} = 0$ and $p_{11}^{(2)} = 4/9$. This gives us three simultaneous equations:

$$\begin{aligned} p_{11}^{(0)} &= 1 &&= A + B \\ p_{11}^{(1)} &= 0 &&= A - \frac{1}{2}B - \frac{1}{2\sqrt{3}}C \\ p_{11}^{(2)} &= \tfrac{4}{9} &&= A + \frac{1}{6}B + \frac{1}{2\sqrt{3}}C \end{aligned}$$

Solving gives us $A = 1/3$, $B = 2/3$ and $C = 0$. So our final solution is:

$$p_{11}^{(n)} = \frac{1}{3} + \frac{2}{3}\left(-\frac{1}{\sqrt{3}}\right)^n \cos(n\pi/6)$$

Question 14. An octopus is trained to choose object A from a pair of objects A and B by repeated trials. The octopus maybe in one of three states of mind:

1. It is untrained and picks A or B at random
2. It is trained and always picks A but may forget its training.
3. It is trained and always picks A and will never forget its training.

After each trial it is rewarded for success and may change its state accordingly. The transition matrix between states is given by:

$$\mathbf{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{12} & \frac{5}{12} \\ 0 & 0 & 1 \end{bmatrix}$$

Assuming that the octopus is in state 1 before trial 1. What is the probability that it is in state 1 before trial $n+1$? What is the probability that it correctly picks item A on trail $n+1$?

The suggestion is made that a two state Markov chain with constant transition probabilities is sufficient to describe the octopus. Discuss briefly this possibility with reference to your calculated value for the probability of picking A on the $n+1$th trial.

Answer 14. First we must find the eigenvalues using:

$$\begin{vmatrix} \frac{1}{2} & \frac{1}{2}-\lambda & 0 \\ \frac{1}{2} & \frac{1}{12}-\lambda & \frac{5}{12} \\ 0 & 0 & 1-\lambda \end{vmatrix} = 0$$

Giving us the equation:

$$(1/2-\lambda)(1/12-\lambda)(1-\lambda) - 1/4(1-\lambda)$$

This has solutions: $\lambda_1 = 1$, $\lambda_2 = 5/6$ and $\lambda_3 = -1/4$. (The $\lambda_1 = 1$ solution can be seen instantly the others come from factorising the remainder of the equation).

The general solution for $p_{ij}^{(n)}$ therefore has the form

$$p_{ij}^{(n)} = A + B\left(\frac{5}{6}\right)^n + C\left(-\frac{1}{4}\right)^n.$$

We now get simultaneous equations by inspecting the chain to get various values for $p_{11}^{(n)}$. So

$$\begin{aligned} p_{11}^{(0)} &= & 1 = & \quad A + B + C \\ p_{11}^{(1)} &= & 1/2 = & \quad A + (5/6)B - (1/4)C \\ p_{11}^{(2)} &= & 1/2 = & \quad A + (25/36)B + (1/16)C \end{aligned}$$

Solving we get $A = 0$, $B = 9/13$ and $C = 4/13$. Therefore

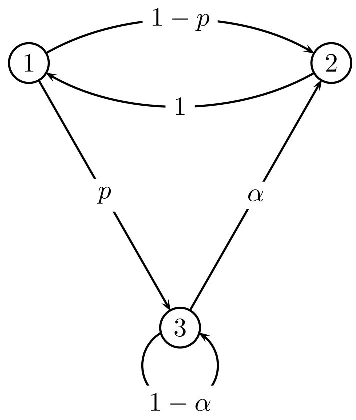
$$p_{11}^{(n)} = \frac{9}{13}\left(\frac{5}{6}\right)^n + \frac{4}{13}\left(-\frac{1}{4}\right)^n$$

The octopus can only pick the wrong object if it is state 1. If it is in state 1 it picks the wrong object 50% of the time. Therefore the probability of it picking the right object on the $n+1$th trial is

$$\mathbb{P}[\text{Right object picked on trial } n+1] = 1 - \mathbb{P}[\text{Wrong object picked on trial } n+1] = 1 - \frac{1}{2}p_{11}^{(n)}$$

We note that the expression for the octopus picking the correct object depends on only two eigenvalues. This suggests the possibility of a two state model which could work in place of the three state model. However, a Markov chain will always have one as an eigenvalue – the answer here would require two eigenvalues neither of which are one. Therefore a three state model is the minimum.

Question 15. Consider the Markov chain shown below.



1. Write down $\mathbf{P}$.
2. Under what conditions (if any) will the chain be irreducible, aperiodic, ergodic?
3. Find the equilibrium probability vector $\boldsymbol{\pi}$.
4. What is the mean recurrence time for state 2.
5. Find values of α and p such that $\pi_1 = \pi_2 = \pi_3$.

Answer 15. 1. The transition matrix between states is given by:

$$\mathbf{P} = \begin{bmatrix} 0 & 1-p & p \\ 1 & 0 & 0 \\ 0 & \alpha & 1-\alpha \end{bmatrix}$$

2. The chain is irreducible if $\alpha > 0$ and $p > 0$. (If $\alpha > 0$ then state 2 can be reached from state 3. If $p > 0$ then state 3 can be reached from state 1. State 1 can always be reached from state 2.

   The chain is periodic if $\alpha = 1$ and $p = 1$ (period 3) or $p = 0$ (period 2) — note that strictly in the last case, only states 1 and 2 are periodic since the chain is reducible. It is aperiodic otherwise.

   Since it is finite, the chain is ergodic if it is irreducible and aperiodic.

3. The equilibrium probability vector can be obtained from the balance equations.

$$\begin{aligned} \pi_1 &= \pi_2 \\ \pi_2 &= (1-p)\pi_1 + \alpha\pi_3 \\ \pi_3 &= p\pi_1 + (1-\alpha)\pi_3 \\ \pi_1 + \pi_2 + \pi_3 &= 1 \end{aligned}$$

   From the first and second equations we can get:

$$\pi_1 = \pi_2 = \frac{1}{2 + p/\alpha}$$

   From the fourth equation then:

$$\pi_3 = \frac{p/\alpha}{2 + p/\alpha}$$

This gives us $\boldsymbol{\pi} = (\frac{1}{2+p/\alpha}, \frac{1}{2+p/\alpha}, \frac{p/\alpha}{2+p/\alpha})$.

4. The mean return time of state two is simply $1/\pi_2$ which is $2 + p/\alpha$.
5. Finally, $\pi_1 = \pi_2 = \pi_3$ iff $p/\alpha = 1$ that is $p = \alpha$.