Input/Output

- Program is in running state unit it makes an I/O request.
- The I/O Request is a service request and is handled by the Operating System.
- Once the request has been made, the program waits for the I/O request to be fulfilled.
- Once the request is fulfilled, an I/O Exception signifies to the hardware that the I/O has completed and the program may continue.
CPU sends I/O request Blocks to controller, which in turn sends signals to the Read/Write Head to retrieve information from the disk
• Disks are divided into tracks, (concentric disks on the surface of the media)
• It is possible to arrange data in such a manner as to optimize the movement of the R/W Head across the tracks (i.e., rather than visiting tracks in the order 3-7-2, visit in the order 2-3-7 to allow one smooth motion)
• When the information has been retrieved, the controller sends an I/O Exception to the CPU
The queue is the final order of I/O’s that is to occur. It may be reordered when new requests come in to make best use of the R/W Head.
After a certain period of build-up the system works into an equilibrium, shown by the IORB x Time graph to the right, and the diagram above. In each state, the Flow of incoming IORB equals the Flow of completed IORB’s.
Some Definitions

A = Number of IO Request arrivals
C = Number of completed IO Requests
T = Length of time system is observed
B = Length of time the device is busy
S = Service time (per request)

Arrival Rate: \( \lambda = \frac{A}{T} \)
Throughput: \( x = \frac{C}{T} \)
Utilization: \( U = \frac{B}{T} = \frac{\lambda}{\mu} \)
Service Time: \( s = \frac{1}{\mu} \)
Probabilities of IO System

\( P_0 \)  Probability of having 0 I/O requests in the system

\( P_1 \)  Probability of having 1 I/O requests in the system

\( P_2 \)  Probability of having 2 I/O requests in the system

\( P_i \)  Probability of having \( i \) I/O requests in the system

Assuming an equal flow-in rate and flow-out rate we get the following...

\[
\mu P_i = \lambda P_{i-1} \Rightarrow P_i = \frac{\lambda}{\mu} P_{i-1}
\]

\[
\begin{align*}
\mu P_1 &= \lambda P_0 \Rightarrow P_1 = \frac{\lambda}{\mu} P_0 \\
\mu P_2 &= \lambda P_1 \Rightarrow P_2 = \frac{\lambda}{\mu} P_1 \\
&\vdots \\
\mu P_i &= \lambda P_{i-1} \Rightarrow P_i = \frac{\lambda}{\mu} P_{i-1}
\end{align*}
\]

Note:

\[
\frac{\lambda}{\mu} = U
\]

where \( U \) is the utilization of IO system
Finding $P_i$

Given that:

$$P_2 = \frac{\lambda}{\mu} P_1$$

substituting the following $P_1 = \frac{\lambda}{\mu} P_0$ and we get

$$P_2 = \left(\frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right) P_0 = \left(\frac{\lambda}{\mu}\right)^2 P_0$$

Continue the pattern for $P_i$ we get:

$$P_i = \left(\frac{\lambda}{\mu}\right)^i P_0 = U^i P_0$$

where $i$ is any given state
Sum of Probabilities

\[ P_0 + P_1 + P_2 + \ldots + P_k = \sum_{i=0}^{k} P_i = 1 \]

substituting \( P_i = U^i P_0 \) we get

\[ \sum_{i=0}^{k} U^i P_0 = 1 \Rightarrow P_0 = \frac{1}{\sum_{i=0}^{k} U^i} \]
Geometric Series

Please recall the following about a geometric series

\[ \sum_{i=0}^{\infty} r^i = \frac{1}{1-r} \quad \text{where } |r| < 1 \]

Assume there exists an infinite number of states

\[ P_o = \frac{1}{\sum_{i=0}^{k} U^i} = \frac{1}{1-U} = 1-U \quad \text{where } |U| < 1 \]

In other words, the utilization, U, must always be less than 100% or else the system will not function.
Average System State

\( \bar{N} \) is the average system state at a finite small amount of time

\[ \bar{N} = \lim_{t \to 0} \sum_{i=0}^{k} iP_i = \lim_{t \to 0} \sum_{i=0}^{k} i(1-U)U^i \]

Noting that the first derivative of \( U^i = iU^{i-1} \) we can substitute this into the equation

\[ \bar{N} = \lim_{t \to 0} U(1-U) \sum_{i=0}^{k} iU^{i-1} \]

\[ iU^{i-1} = \frac{\partial (U^i)}{\partial U} \]

\[ \bar{N} = \lim_{t \to 0} U(1-U) \sum_{i=0}^{k} \left( \frac{\partial U^i}{\partial U} \right) \]
Solving the for the geometric sum and the derivative of the result we get the following

\[
\overline{N} = \lim_{t \to 0} U(1-U) \left( \frac{1}{1-U} \right) = \lim_{t \to 0} \frac{U(1-U)}{(1-U)^2}
\]

Which, in turn, gives us the final equation:

\[
\overline{N} = \lim_{t \to 0} \frac{U}{(1-U)}
\]