Quantifying Performance Models Part 1

Dr. John Burns

Dept. of Computing
ITT Dublin

November 6, 2017

Flying Blind Without Instrumentation

To leverage your infrastucture in an efficient way, you need insight into your resources, e.g.:

1. How much of your infrastructure is actually being used?
2. Is your application's performance or availability being affected by a lack of sufficient capacity?
3. What is the capacity of your infrastructure?
4. When will you reach max. Throughput or Utilization?

Motivating Problem

This section presents the approach known as operational analysis, used to establish relationships among quantities based on measured or known data about computer systems. To see how the operational approach might be applied, consider the following motivating problem.

A CPU example

Suppose that during an observation period of 1 minute, a single-CPU is observed to be busy for 36 seconds. A total of 1800 transactions are observed to have arrived in the system. The total number of observed completions is also 1800 transactions [a]. What is the *performance* of the system? (ie, mean service time per transaction, CPU utilisation, system throughput)

[a] the so-called steady-state or equilibrium requirement

Notation

- $\mathcal{T}$: the length of time in the observation period.
- K: the number of resources in the system.
- B_i: the total busy time of the i-th resource in $\mathcal{T}$.
- A_i: the total number of requests to the i-th resource in $\mathcal{T}$.
- A_0: the total number of requests to the system in $\mathcal{T}$.
- C_i: the total number of completions by the i-th resource in $\mathcal{T}$.
- C_0: the total number of completions by the system in $\mathcal{T}$.

From these measurable quantities, a set of *derived* quantities can be obtained:

Derived Metrics

- S_i: The mean service time per completion at resource i. $S_i = B_i/C_i$
- U_i: The utilisation of resource i, given as: $U_i = B_i/\mathcal{T}$
- X_i: Throughput (completions per unit time) of resource i: $X_i = C_i/\mathcal{T}$
- λ_i: arrival rate (arrivals per unit time) at resource i: $\lambda_i = A_i/\mathcal{T}$
- X_0: system throughput: $X_0 = C_0/\mathcal{T}$.
- V_i: average number of visits (visit count) per request to resource i: $V_i = C_i/C_0$.

Restating the problem

Using the notation above, we can re-state the problem as follows:

- $\mathcal{T} = 60$ seconds
- $K = 1$ resource
- $B_1 = 36$ seconds
- $A_1 = A_0 = 1800$ transactions
- $C_1 = C_0 = 1800$ transactions

And thus, the derived quantities are:

Derived metrics

The dervied quantities can now be easily computed:

- $S_i = B_i/C_i = 36/1800 = 1/50$ second per transaction
- $U_1 = B_1/\mathcal{T} = 36/60 = 60\%$ busy
- $\lambda_1 = A_1/\mathcal{T} = 1800/60 = 30$ tps
- $X_0 = C_0/\mathcal{T}1800/60 = 30$ tps

Each of these metrics can be further devloped to derive *operational laws*:

Utilisation Law I

As mentioned, the utilisation of a resource is given by: $U_i = B_i/\mathcal{T}$. dividing the numerator and denominator by the number of completions C_i of the resource yields:

$$U_i = \frac{B_i}{\mathcal{T}} = \frac{B_i/C_i}{\mathcal{T}/C_i} \tag{1}$$

- The ratio B_i/C_i is simply the average time that the resource was busy for each completion from the resource, which is the average service time S_i per visit to the resource.
- The ratio $\mathcal{T}/C_i$ is the inverse of the resource throughput X_i. Thus, the relation known as the **utilisation law** can be written:

Utilisation Law II

Definition

Utilisation Law

$$U_i = S_i \times X_i \tag{2}$$

If the number of completions from resource i during the observation period $\mathcal{T}$ is equal to the number of arrivals in that period, ie, if

- $C_i = A_i$, then $X_i = \lambda_i$
- and the relationship given by the utilisation law is
- $U_i = S_i \times \lambda_i$

If resource i has m servers as in a multiprocessor, the utilisation law becomes $U_i = (S_i \times X_i)/m$ Consider the following Utilisation Law example:

Utilisation Law example

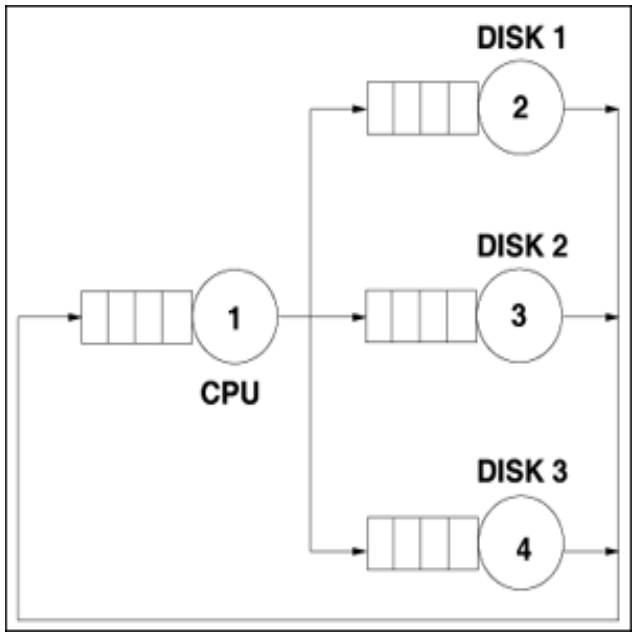
A computer system with one CPU and three disks used to support a database server. Assume that all database transactions have similar resource demands and that the database server is under a constant load of transactions [1]. (Fig. 1.) The CPU is an example of resource 1, and the three disks are resources 2,3,4. Measurements taken in 1 hour period provide the number of transactions executed (13,680), the number of reads and writes per second on each disk, and their utilisation, as indicated in Table 1. Calculate:

- The average service time per request on each disk
- The database servers throughput

[1] ie, there is no significant variance

Utilisation Law example I

Figure 1: Closed QN model of a database server



Utilisation Law example II

Table 1: DB Utilisation parameters

Disk	Total I/O /sec	Utilization
1	32	0.30
2	36	0.41
3	50	0.54

DB Utilisation Calculations

The throughput of each disk, denoted by X_i ($i = 2, 3, 4$), is the total number of I/Os per second, i.e., the sum of the number of reads and writes per second. This value is indicated in the fourth column of the table. Using the Utilization Law, the average service time is computed S_i as U_i/X_i.

- $S_2 = U_2/X_2 = 0.30/32 = 0.0094$ sec,
- $S_3 = U_3/X_3 = 0.41/36 = 0.0114$ sec, and
- $S_4 = U_4/X_4 = 0.54/50 = 0.0108$ sec.
- The throughput, X0, of the database server is given by $X_0 = C_0/\mathcal{T} = 13,680$ transactions/3,600 seconds $= 3.8$ tps.

Service Demand Law I

- Service demand is a fundamental concept in performance modeling.
- The notion of service demand is associated both with a resource and a set of requests using the resource.
- The service demand, denoted D_i, is defined as the total average time spent by a typical request of a given type obtaining service from resource i.
- Throughout its existence, a request may visit several devices, possibly multiple times. However, for any given request, its service demand is the sum of all service times during all visits to a given resource.

Service Demands I

- Fortunately, there is an easy way to obtain service demands from resource utilizations and system throughput.
- By multiplying the utilization U_i of a resource by the measurement interval T one obtains the total time the resource was busy.
- If this time is divided by the total number of completed requests, C_0, the average amount of time that the resource was busy serving each request is derived.

Definition

Service Demand

$$D_i = \frac{U_i \times \mathcal{T}}{C_0} = \frac{U_i}{C_0/\mathcal{T}} = \frac{U_i}{X_0} \tag{3}$$

Service Demands

- This relationship (Eqn.(3)) is called the **Service Demand Law**, which can also be written as $D_i = V_i \times S_i$, by definition of the service demand (and since
$D_i = U_i/X_0 = (B_i/\mathcal{T})/(C_0/\mathcal{T})$
$= B_i/C_0 = (C_i \times S_i)/C_0 = (C_i/C_0) \times S_i = V_i \times S_i$).
- In many cases, it is not easy to obtain the individual values of the visit counts and service times.
- However, Eqn.(3) indicates that the service demand can be computed directly from the device utilization and system throughput.

Example I

Example

A Web server is monitored for 10 minutes and its CPU is observed to be busy 90% of the monitoring period. The Web server log reveals that 30,000 requests are processed in that interval. What is the CPU service demand of requests to the Web server?

- The observation period $\mathcal{T} = 600 (= 10 \times 60)$ seconds.
- The Web server throughput, X_0, is equal to the number of completed requests C_0 divided by the observation interval;
- $X_0 = 30,000/600 = 50$ requests/sec.
- The CPU utilization is $U_{CPU} = 0.9$.
- Thus, the service demand at the CPU is $D_{CPU} = U_{CPU}/X_0 = 0.9/50 = 0.018$ seconds/request.

Example

Further Example

What are the service demands at the CPU and the three disks for the database server of the previous example, assuming that the CPU utilization is 35% measured during the same one-hour interval?

- Remember that the database server's throughput was computed to be 3.8 tps.
- Using the Service Demand Law and the utilization values for the three disks shown in Table 1 yields:
  $D_{CPU} = 0.35/3.8 = 0.092$ sec/transaction,
  $D_{disk1} = 0.30/3.8 = 0.079$ sec/transaction,
  $D_{disk2} = 0.41/3.8 = 0.108$ sec/transaction, and
  $D_{disk3} = 0.54/3.8 = 0.142$ sec/transaction.

The Forced Flow Law I

- There is an easy way to relate the throughput of resource i, X_i, to the system throughput, X_0.
- Assume for the moment that every transaction that completes from the database server of Example 1 performs an average of two I/Os on disk 1.
- That is, suppose that for every one visit that the transaction makes to the database server, it visits disk 1 an average of two times.
- What is the throughput of that disk in I/Os per second?
- Since 3.8 transactions complete per second (i.e., the system throughput, X_0) and each one performs two I/Os on average on disk 1, the throughput of disk 1 is 7.6($= 2.0x3.8$) I/Os per second.

The Forced Flow Law II

- In other words, the throughput of a resource (X_i) is equal to the average number of visits (V_i) made by a request to that resource multiplied by the system throughput (X_0).
- This relation is called the Forced Flow Law:

Definition

Forced Flow Law

$$X_i = V_i \times X_0 \tag{4}$$

The multiclass version of the Forced Flow Law is $X_{i,r} = V_{i,r} \times X_{0,r}$

The Forced Flow Law III

Example

What is the average number of I/Os on each disk in Example 1? The value of V_i for each disk i, according to the **Forced Flow Law**, can be obtained as X_i/X_0. The database server throughput is 3.8 tps and the throughput of each disk in I/Os per second is given in the fourth column of Table 1. Thus,
$V_1 = X_1/X_0 = 32/3.8 = 8.4$ visits to disk 1 per database transaction. Similarly, $V_2 = X_2/X_0 = 36/3.8 = 9.5$ and
$V_3 = X_3/X_0$