Computer Networks Performance Evaluation

Chapter 12
Single Class MVA

Performance by Design:
Computer Capacity Planning by Example

Daniel A. Menascé, Virgilio A.F. Almeida, Lawrence W. Dowdy
Prentice Hall, 2004

Chapter 12-Outlines

Introduction

- The Achilles' heel of Markov models is their susceptibility to state space explosion.
- In simple models, with a fixed number of identical customers, which the demands placed by each customer on each device are exponentially distributed, the number of states is given by the expression

$$\binom{N+K-1}{K-1} = \frac{(N+K-1)!}{(k-1)!\,N!}$$

- Where N is the number of customers and K is the number of devices.

Introduction.

- For small systems, such as the database server example in the previous chapter with $N = 2$ and $K = 3$, the number of states is 6.
- With 50 users and 50 workstations, the number of states is over 5×10^{28}.
- Since there is one linear equation (i.e., equating flow into the state to the flow out of the state) for every state, solving such a large number of simultaneous equations is infeasible.

Introduction..

- However, clever algorithms have been developed for a broad class of Markov models requiring no the explicit solution to a large number of simultaneous equations.

- One technique is Mean Value Analysis (MVA).
- Instead of solving a set of simultaneous linear equations to find the steady state probability of being in each system state, MVA calculates the performance metrics directly for a given number of customers, knowing only the performance metrics when the number of customers is reduced by one.

Introduction...

- All of the N customers are assumed to be identical, forming a single class of customers. Each of the K devices is assumed to be load independent.
- The demand placed on a device (the service required by a customer at a particular device) is assumed to be exponentially distributed.
- There are enhancements to MVA, removing these restrictions (i.e., allowing multi-class customers, allowing load dependent servers, and allowing non-exponential service).

Introduction….

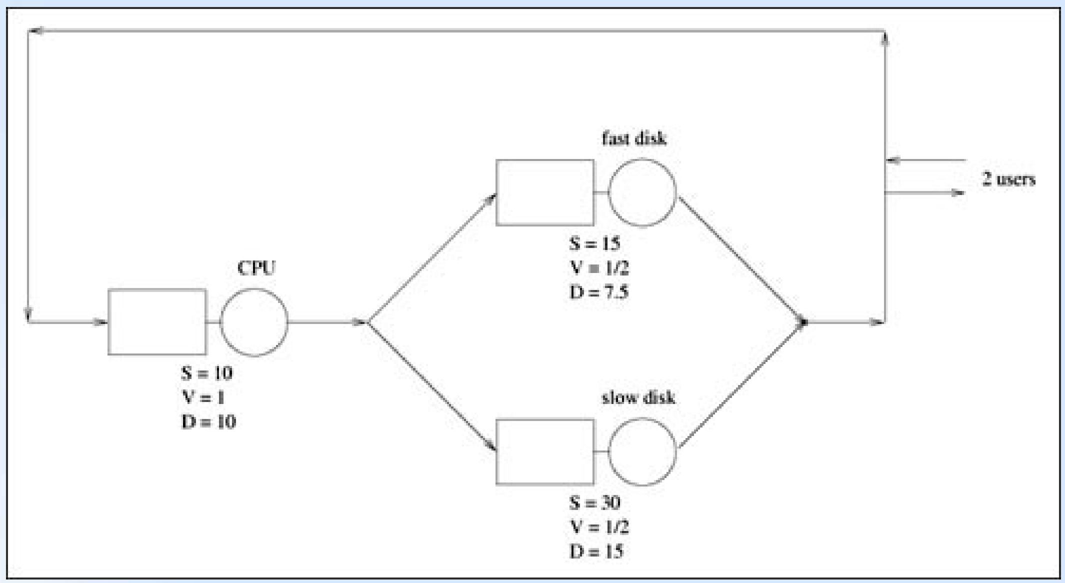
- This chapter is example based. In Section 12.2, the database server example from previous chapters is extended to develop the basic MVA algorithm. A concise, algorithmic description of MVA is given in Section 12.3. The special case of balanced systems is presented in Section 12.4. Section 12.5 describes extensions and limitations associated with MVA. The chapter concludes with a summary and relevant exercises.

Chapter 12-Outlines

- 12.1 Introduction
- 12.2 MVA Development
- 12.3 The MVA Algorithm
- 12.4 Balanced Systems
- 12.5 MVA Extensions and Limitations
- 12.6 Chapter Summary
- 12.7 Exercises

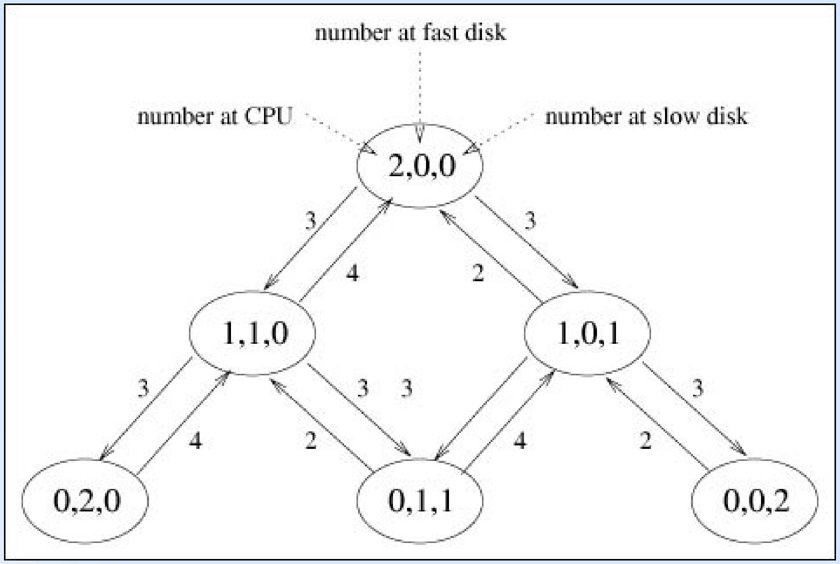
A Typical Problem

- Previous Paradigm Revisited: Reconsider the database server from the previous chapter, whose diagram is reproduced in Figure 12.1.



Problem Parameters

- S: mean service time per visit,
- V: average number of visits per transaction
- D = S x V: total demand per transaction The underlying Markov model is reproduced in Figure 12.2.



Problem Solution

- By solving the six balance equations, the steady state probabilities were found to be:

$$P_{(2,0,0)} = \frac{16}{115} = 0.1391$$
$$P_{(1,1,0)} = \frac{12}{115} = 0.1043$$
$$P_{(1,0,1)} = \frac{24}{115} = 0.2087$$
$$P_{(0,2,0)} = \frac{9}{115} = 0.0783$$
$$P_{(0,1,1)} = \frac{18}{115} = 0.1565$$
$$P_{(0,0,2)} = \frac{36}{115} = 0.3131$$

Testing the Solution

- From these probabilities, other performance metrics can be derived. For example, the average number of customers at the CPU is a simple weighted sum of those probabilities.
- Therefore, the average number of customers at the CPU is:

$$2 \times P_{(2,0,0)} + 1 \times P_{(1,1,0)} + 1 \times P_{(1,0,1)} = \frac{68}{115} = 0.5913 \text{ customers.}$$

- Similarly, the average number of customers at the fast disk is:

$$1 \times P_{(1,1,0)} + 2 \times P_{(0,2,0)} + 1 \times P_{(0,1,1)} = \frac{48}{115} = 0.4174 \text{ customers}$$

Testing the Solution.

- The average number of customers at the slow disk is:

$$1 \times P_{(1,0,1)} + 1 \times P_{(0,1,1)} + 2 \times P_{(0,0,2)} = \frac{114}{115} = 0.9913 \text{ customers.}$$

- The sum of these three numbers, 0.5913+0.4174+0.9913 = 2.0000, accounts for the two customers in the system.

CPU Utilization

- The utilization of each device can be easily calculated knowing the steady state probabilities. For instance, the utilization of the CPU is:

$$P_{(2,0,0)} + P_{(1,1,0)} + P_{(1,0,1)} = \frac{16}{115} + \frac{12}{115} + \frac{24}{115} = \frac{52}{115} = 0.4522.$$

Disks Utilization

- Likewise, the utilization of the fast disk is:

$$P_{(1,1,0)} + P_{(0,2,0)} + P_{(0,1,1)} = \frac{12}{115} + \frac{9}{115} + \frac{18}{115} = \frac{39}{115} = 0.3391$$

- and the utilization of the slow disk is:

$$P_{(1,0,1)} + P_{(0,1,1)} + P_{(0,0,2)} = \frac{24}{115} + \frac{18}{115} + \frac{36}{115} = \frac{78}{115} = 0.6783.$$

- [Important side note: Device utilizations are in the same ratio as their service demands, regardless of number of customers in the system (i.e., the system load).]

Throughputs

- Knowing utilizations, device throughputs follow from the Utilization Law in Chapter 3.
- Device i's throughput is given by: $X_i = U_i / S_i$.
- Thus, the throughput of
  - the CPU is 0.4522/10 = 0.0452 customers per second, or 2.7130 customers per minute.
  - of each disk is 1.3565 customers per minute. This is consistent since the throughput of the CPU is split evenly between the two disks.

Residence Time

- Knowing
  - 1) the average number of customers, n_i, at each device and
  - 2) the throughput, X_i, of each device,
  - the response time, per visit to each device is: (via Little's Law) n_i/X_i .
  - Since V=1 then the residence time= response time
- Thus, the response times R_i
  - Of the CPU is 13.08 seconds
  - Of the fast disk is 18.46 seconds, and
  - Of the slow disk is 43.85 seconds.

Transaction Response Time

- A typical customer's transaction visits
  - the CPU once and
  - only one of the disks (with equal likelihood),
- Overall response time of a transaction is a weighted sum of the individual device residence times. Thus, a transaction's response time is :

  $1 \times 13.08 + 1/2 \times 18.46 + 1/2 \times 43.85 = 44.24$ seconds.
- A summary of the relevant performance measures is presented in Table 12.1.

Table 12.1. Performance Metrics for the Database Server Example (2 customers)

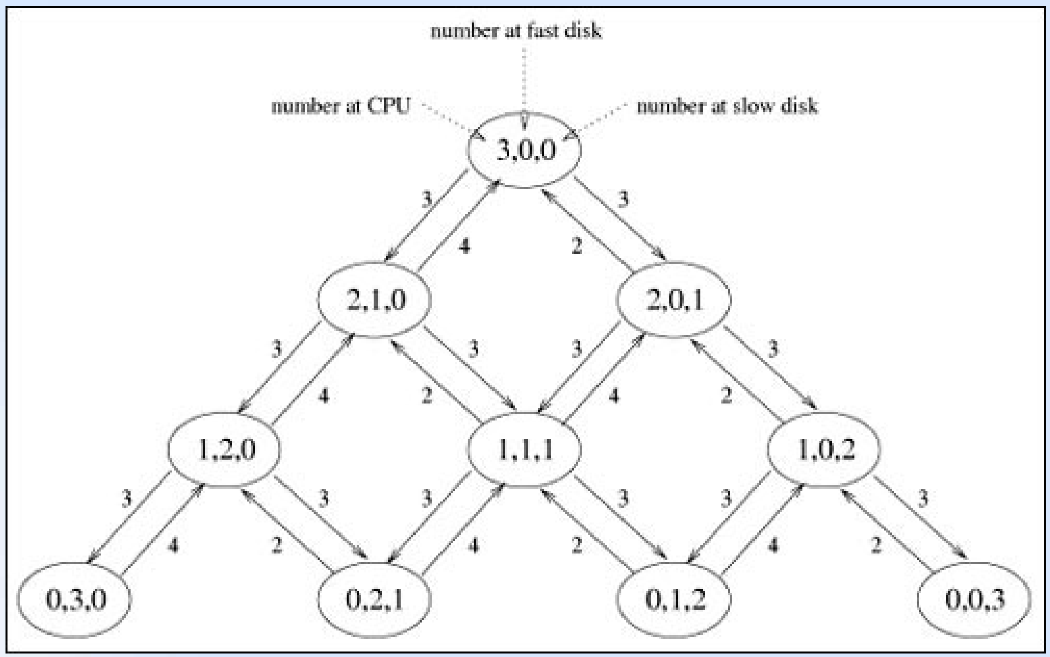
Average Number of Customers	
CPU	0.5913
Fast disk	0.4174
Slow disk	0.9913
Utilizations (%)	
CPU	45.22%
Fast disk	33.91%
Slow disk	67.83%
Throughputs (customers per minute)	
CPU	2.7130
Fast disk	1.3565
Slow disk	1.3565
Residence Times (seconds)	
CPU	13.08
Fast disk	9.23
Slow disk	21.93
Response Time (seconds)	44.24

3 Customers in System

- Consider the same database server example, with three customers. The associated Markov model is illustrated in Figure 12.3.
- The ten balance equations are shown in Table 12.2.
- The steady state solution to the balance equations are given in Table 12.3 and
- the associated performance metrics are given in Table 12.4.
- These are straight-forward extensions of the case with two customers and are left as exercises for the reader.

MVA Development

- Figure 12.3. Markov model of the database server example (3 customers).



Balance Equations

Table 12.2. Balance Equations for the Database Server Example (3 customers)

$$
\begin{aligned}
(4 \times P_{(2,1,0)}) + (2 \times P_{(2,0,1)}) &= 6 \times P_{(3,0,0)} \\
(3 \times P_{(3,0,0)}) + (4 \times P_{(1,2,0)}) + (2 \times P_{(1,1,1)}) &= 10 \times P_{(2,1,0)} \\
(3 \times P_{(3,0,0)}) + (4 \times P_{(1,1,1)}) + (2 \times P_{(1,0,2)}) &= 8 \times P_{(2,0,1)} \\
(3 \times P_{(2,1,0)}) + (4 \times P_{(0,3,0)}) + (2 \times P_{(0,2,1)}) &= 10 \times P_{(1,2,0)} \\
(3 \times P_{(2,1,0)}) + (3 \times P_{(2,0,1)}) + (4 \times P_{(0,2,1)}) + (2 \times P_{(0,1,2)}) &= 12 \times P_{(1,1,1)} \\
(3 \times P_{(2,0,1)}) + (4 \times P_{(0,1,2)}) + (2 \times P_{(0,0,3)}) &= 8 \times P_{(1,0,2)} \\
3 \times P_{(1,2,0)} &= 4 \times P_{(0,3,0)} \\
(3 \times P_{(1,2,0)}) + (3 \times P_{(1,2,1)}) &= 6 \times P_{(0,2,1)} \\
(3 \times P_{(1,1,1)}) + (3 \times P_{(1,0,2)}) &= 6 \times P_{(0,1,2)} \\
3 \times P_{(1,0,2)} &= 2 \times P_{(0,0,3)} \\
P_{(3,0,0)} + P_{(2,1,0)} + P_{(2,0,1)} + P_{(1,2,0)} + P_{(1,1,1)} & \\
+P_{(1,0,2)} + P_{(0,3,0)} + P_{(0,2,1)} + P_{(0,1,2)} + P_{(0,0,3)} &= 1.0
\end{aligned}
$$

MVA Development

- Table 12.3. Solution for the Database Server Example (3 customers)

$$
\begin{aligned}
P_{(3,0,0)} &= \tfrac{64}{865} = 0.0740\\
P_{(2,1,0)} &= \tfrac{48}{865} = 0.0555\\
P_{(2,0,1)} &= \tfrac{96}{865} = 0.1110\\
P_{(1,2,0)} &= \tfrac{36}{865} = 0.0416\\
P_{(1,1,1)} &= \tfrac{72}{865} = 0.0832\\
P_{(1,0,2)} &= \tfrac{144}{865} = 0.1665\\
P_{(0,3,0)} &= \tfrac{27}{865} = 0.0312\\
P_{(0,2,1)} &= \tfrac{54}{865} = 0.0624\\
P_{(0,1,2)} &= \tfrac{108}{865} = 0.1249\\
P_{(0,0,3)} &= \tfrac{216}{865} = 0.2497
\end{aligned}
$$

Table 12.4. Performance Metrics for the Database Server Example (3 customers)

Average Number of Customers	
CPU	0.8462
Fast disk	0.5653
Slow disk	1.5885
Utilizations (%)	
CPU	53.18%
Fast disk	39.88%
Slow disk	79.77%
Throughputs (customers per minute)	
CPU	3.1908
Fast disk	1.5954
Slow disk	1.5954
Residence Times (seconds)	
CPU	15.91
Fast disk	10.63
Slow disk	29.87
Response Time (seconds)	56.41

Side notes

- As a consistency check on the performance metrics given in Table 12.4, the sum of the average number of customers at the devices equals the total number of customers in the system (i.e., three).
- Also, the utilization of the CPU is (2/3) of the slow disk, and the utilization of the slow disk is twice that of the fast disk (i.e.,utilizations remain in the same ratio as their service demands).
- The throughputs of the disks are identical and sum to that of the CPU.

Need for a New Paradigm

- This technique of going from the two customer case to the three customer case (i.e., state space extension, balance equation derivation, solution of the linear equations, interpretation of the performance metrics) does not scale as the number of devices and the number of customers increases.
- A new paradigm of analyzing the relationships between the performance metrics is required.

Arrival Theorem: Example

- Consider the relationship between the residence time at the CPU with three customers (i.e., 15.91 seconds) to the average number of customers at the CPU with two customers (i.e., 0.5913).
- Three customers in the network:
  - at the instant when a customer arrives at the CPU, the average number of customers that the arriving customer sees already at the CPU is precisely the average number of customers at the CPU with two customers in the network. (This is an important result known as the "Arrival Theorem".)
  - Therefore, an arriving customer at the CPU will expect to see 0.5913 customers already there.

Arrival Theorem: Example.

- Thus, the time it will take for the newly arriving customer to complete service and leave the CPU (i.e., its residence time) will be
  - the time it takes to service those customers already at the CPU
  - plus the time it takes to service the arriving customer.
- Since the average service time per customer at the CPU is 10 seconds,
  - it will take an average of 10×0.5913 seconds to service those customers already at the CPU,
  - plus an additional 10 seconds to service the arriving customer.
  - Therefore, the residence time is $10(1+0.5913) = 15.91$ seconds.

Arrival Theorem

- As a general relationship, Letting
- $R_i\,(n)$ be the average response time per visit to device i when there are n customers in the network, and
- S_i be the average service time of a customer at device i, and
- $\overline{n}_i(n-1)$ be the average number of customers at device i when there are a total of n–1 customers in system,
- the following relationship exists:

$$\boxed{R_i(n) = S_i[1 + \overline{n}_i(n-1)]}$$

Arrival Theorem: Example..

- Response time at the fast disk, when there are three customers in the network, is the product of its service time (i.e., 15 seconds) and the number of customers at the disk: $15(1 + 0.4174) = 21.36$ seconds.

- Likewise, the residence time at the slow disk is : $30(1 + 0.9913) = 59.74$ seconds.

Overall Response Time

- Now the overall response time, $R_0(n)$, is the sum of the residence times (i.e. $R_i'(n)$).

- In database server with three customers, the residence times at CPU, fast disk, and slow disk are 15.91, 21.26,and 59.74 seconds.
- The number of visits to these devices per transaction is 1.0, 0.5, and 0.5.
- Thus the overall response time is (1.0 x 15.91) + (0.5 x 21.26) + (0.5 x 59.74) = 56.41 seconds.

- Overall Response Time Formula:

$$\boxed{R_0(n) = \sum_{i=1}^{K} R_i'(n) = \sum_{i=1}^{K} [V_i \times R_i(n)].}$$

System Throughput

- From Little's Law ,the average number of customers in the system ,n, is the product of system throughput, $X_0(n)$, and system response time, $R_0(n)$ .Thus,

$$X_0(n) = \frac{n}{R_0(n)}$$

- The individual device throughputs can be found using the Forced Flow Law,

$$X_i(n) = V_i \times X_0(n).$$

Utilization

- The device utilizations follow from the device throughputs via the Utilization Law,

$$U_i(n) = S_i \times X_i(n).$$

Customer Number

- finally average number of customers at each device when there are n customers in the system follows directly from Little's Law applied to each individual device,

$$\bar{n}_i(n) = X_i(n) \times R_i(n).$$

- But, from the Forced Flow Law, $X_i(n) = V_i \times X_0(n)$. Thus,

$$\bar{n}_i(n) = X_0(n) \times V_i \times R_i(n) = X_0(n) \times R_i'(n).$$

Example

- In the database server with three customers, overall system throughput is
  $X_0(3) = 3/R(3) = 3/56.41 = 0.0532$ customers/second (3.1908 customers/minute) which yields 0.8462 customers at cpu.
- And the individual device throughputs are 0.0532, 0.0266, and 0.0266 customers per seconds.
- Average number of customers at the fast disk when there are three customers in the system is $0.0266 \times 21.26 = 0.5653$ customers. At the slow disk, there are $0.0266 \times 59.74 = 1.5885$ customers.

A Conclusion

- Knowing only the average number of customers at each device with *two customers*, the device residence times when there are *three customers* in the network can be quickly derived.
- Knowing these residence times leads directly to the overall response time.
- The initialization of the iterative process is resolved by noting that when *no customers* are in system, the average number of customers at each device is zero.
- Thus, when $n = 0,\ \bar{n}_i(0) = 0$ for all devices i.

A Conclusion.

- Given the average number of customers at each device with $n-1$ customers in the system, the device residence times when there are n customers in the network can be derived.
- Knowing these residence times leads to the overall response time, which, in turn, leads to the system and individual device throughputs.
- The device throughputs lead to
  - the device utilizations and to
  - the average number of customers at each device with n customers in the system.
- Knowing these, the iteration continues to derive the performance metrics with $n+1$ customers in the system, and, in general, to any desired number of customers ... all without formulating and solving any of the underlying steady state balance equations.

Chapter 12-Outlines

- 12.1 Introduction
- 12.2 MVA Development
- 12.3 The MVA Algorithm
- 12.4 Balanced Systems
- 12.5 MVA Extensions and Limitations
- 12.6 Chapter Summary
- 12.7 Exercises

The MVA Algorithm

- The MVA algorithm is given concisely in Table 12.5 for any single class network with N customers and K devices. The average service time of a customer at device i is S_i and the average number of visits that a customer makes to device i is V_i.
- For all customer populations n $(1 \leq n \leq N)$, the algorithm finds the following performance metrics:
  - the average residence time at each device,
  - the overall system response time,
  - the overall system throughput,
  - the individual device throughputs,
  - the device utilizations, and
  - the average number of customers at each device.

Table 12.5. The MVA Algorithm

1. Initialize the average number of customers at each device i : $\bar{n}_i(0) = 0$
2. For each customer population $n = 1, 2, \ldots N$,

   Calculate the average residence time for each device i: $R'_i(n) = V_i\ S_i\,[1 + \bar{n}_i(n-1)] = D_i\,[1 + \bar{n}_i(n-1)]$

   Calculate the overall system response time:
   $$R_0(n) = \sum_{i=1}^{K} [V_i \times R_i(n)] = \sum_{i=1}^{K} R'_i(n)$$

   Calculate the overall system throughput:
   $$X_0(n) = \frac{n}{R_0(n)}$$

   Calculate the throughput for each device i:
   $$X_i(n) = V_i \times X_0(n)$$

   Calculate the utilization for each device i:
   $$U_i(n) = S_i \times X_i(n)$$

   Calculate the average number of customers at each device i:
   $$\bar{n}_i(n) = X_0(n) \times R'_i(n).$$

Example

- Applied to the database server example, where the average service times are 10 seconds, 15 seconds, and 30 seconds, respectively, for the CPU (cp), fast disk (fd), and slow disk (sd), and where the average number of visits to each device are 1.0, 0.5, and 0.5, the MVA iteration proceeds as follows: $\bar{n}_i$

- Initialize the average number of customers at each device i: $\bar{n}_i(0) = 0$

$$\bar{n}_{cp}(0) = 0.0000 \text{ customers}$$
$$\bar{n}_{fd}(0) = 0.0000 \text{ customers}$$
$$\bar{n}_{sd}(0) = 0.0000 \text{ customers}$$

Example n=1

- For customer population n = 1, calculate the average residence time for each device i:

$$R_i'(n) = V_i \; S_i \, [1 + \bar{n}_i(n-1)] = D_i \, [1 + \bar{n}_i(n-1)]$$

$$\begin{aligned} R_{cp}'(1) &= 10 \times 1 \; (1 + 0.0000) = 10.00 \text{ seconds} \\ R_{fd}'(1) &= 15 \times 0.5 \; (1 + 0.0000) = 7.50 \text{ seconds} \\ R_{sd}'(1) &= 30 \times 0.5 \; (1 + 0.0000) = 15.00 \text{ seconds} \end{aligned}$$

- Calculate the overall system response time:

$$(R_0(n) = \textstyle\sum_{i=1}^{K} R_i'(n))$$

$$R_0(1) = 10.00 + 7.50 + 15.00 = 32.50 \text{ seconds}$$

- Calculate the overall system throughput:

$$(X_0(n) = \tfrac{n}{R(n)}).$$

$$X_0(1) = \frac{1}{32.50} = 0.0308 \text{ customers per second}$$

Example n=1.

- Calculate the throughput for each device i:
  ($X_i(n) = V_i \times X_0(n)$).

$$X_{cp}(1) = 1.0(0.0308) = 0.0308 \text{ customers per second}$$
$$X_{fd}(1) = 0.5(0.0308) = 0.0154 \text{ customers per second}$$
$$X_{sd}(1) = 0.5(0.0308) = 0.0154 \text{ customers per second}$$

- Calculate the utilization for each device i:
  ($U_i(n) = S_i \times X_i(n)$).

$$U_{cp}(1) = 10.00(0.0308) = 0.3077 = 30.77\%$$
$$U_{fd}(1) = 15.00(0.0154) = 0.2308 = 23.08\%$$
$$U_{sd}(1) = 30.00(0.0154) = 0.4615 = 46.15\%$$

- Calculate the average number of customers at each device i: $\overline{n}_i(n) = X_0(n) \times R'_i(n)$

$$\bar{n}_{cp}(1) = 0.0308(10.00) = 0.3077 \text{ customers}$$
$$\bar{n}_{fd}(1) = 0.0308(7.50) = 0.2310 \text{ customers}$$
$$\bar{n}_{sd}(1) = 0.0308(15.00) = 0.4620 \text{ customers}$$

Example n=2

- For customer population n = 2, calculate the average residence time for each device i:

$$R_i'(n) = V_i \; S_i \, [1 + \bar{n}_i(n-1)] = D_i \, [1 + \bar{n}_i(n-1)]$$

$$R_{cp}'(2) = 10(1 + 0.3077) = 13.08 \text{ seconds}$$
$$R_{fd}'(2) = 7.5(1 + 0.2308) = 9.23 \text{ seconds}$$
$$R_{sd}'(2) = 15(1 + 0.4615) = 21.93 \text{ seconds}$$

- Calculate the overall system response time:

$$(R_0(n) = \textstyle\sum_{i=1}^{K} R_i'(n))$$

$$R_0(2) = 13.08 + 9.23 + 21.93 = 44.24 \text{ seconds}$$

- Calculate the overall system throughput:

$$(X_0(n) = \tfrac{n}{R(n)}).$$

$$X_0(2) = \frac{2}{44.24} = 0.0452 \text{ customers per second}$$

Example n=2.

- Calculate the throughput for each device i:
  ($X_i(n) = V_i \times X_0(n)$).

$$X_{cp}(2) = 1.0(0.0452) = 0.0452 \text{ customers per second}$$
$$X_{fd}(2) = 0.5(0.0452) = 0.0226 \text{ customers per second}$$
$$X_{sd}(2) = 0.5(0.0452) = 0.0226 \text{ customers per second}$$

- Calculate the utilization for each device i:
  ($U_i\ (n) = S_i \times X_i\ (n)$).

$$U_{cp}(2) = 10.00(0.0452) = 0.4522 = 45.22\%$$
$$U_{fd}(2) = 15.00(0.0226) = 0.3391 = 33.91\%$$
$$U_{sd}(2) = 30.00(0.0226) = 0.6783 = 67.83\%$$

- Calculate the average number of customers at each device i: $\bar{n}_i(n) = X_0(n) \times R(n)$

$$\bar{n}_{cp}(2) = 0.0452(13.08) = 0.5913 \text{ customers}$$
$$\bar{n}_{fd}(2) = 0.0452(9.23) = 0.4174 \text{ customers}$$
$$\bar{n}_{sd}(2) = 0.0452(21.93) = 0.9913 \text{ customers}$$

Example n=3

- For customer population n = 3, calculate the average residence time for each device i:

$$R_i'(n) = V_i\ S_i\,[1 + \bar{n}_i(n-1)] = D_i\,[1 + \bar{n}_i(n-1)]$$

$$R_{cp}'(3) = 10(1 + 0.5913) = 15.91 \text{ seconds}$$
$$R_{fd}'(3) = 7.5(1 + 0.4174) = 10.63 \text{ seconds}$$
$$R_{sd}'(3) = 15(1 + 0.9913) = 29.87 \text{ seconds}$$

- Calculate the overall system response time:

$$(R_0(n) = \textstyle\sum_{i=1}^{K} R_i'(n))$$

$$R_0(3) = 15.91 + 10.63 + 29.87 = 56.41 \text{ seconds}$$

- Calculate the overall system throughput:

$$(X_0(n) = \tfrac{n}{R(n)}).$$

$$X_0(3) = \frac{3}{56.41} = 0.0532 \text{ customers per second}$$

Example n=3.

- Calculate the throughput for each device i:
  ($X_i(n) = V_i \times X_0(n)$).

$$X_{cp}(3) = 1.0(0.0532) = 0.0532 \text{ customers per second}$$
$$X_{fd}(3) = 0.5(0.0532) = 0.0266 \text{ customers per second}$$
$$X_{sd}(3) = 0.5(0.0532) = 0.0266 \text{ customers per second}$$

- Calculate the utilization for each device i:
  ($U_i(n) = S_i \times X_i(n)$).

$$U_{cp}(3) = 10.00(0.0532) = 0.5318 = 53.18\%$$
$$U_{fd}(3) = 15.00(0.0266) = 0.3988 = 39.88\%$$
$$U_{sd}(3) = 30.00(0.0266) = 0.7977 = 79.77\%$$

- Calculate the average number of customers at each device i: $\bar{n}_i(n) = X_0(n) \times R(n)$

$$\bar{n}_{cp}(3) = 0.0532(15.91) = 0.8462 \text{ customers}$$
$$\bar{n}_{fd}(3) = 0.0532(10.63) = 0.5653 \text{ customers}$$
$$\bar{n}_{sd}(3) = 0.0532(29.87) = 1.5885 \text{ customers}$$

Example Conclusion

- These performance metrics found via MVA for two and three customers (i.e., when $n = 2$ and when $n = 3$) correspond directly to those found from first principles (i.e., by constructing the Markov model, forming the balance equations, solving the balance equations, and interpreting the results)

Chapter 12-Outlines

- 12.1 Introduction
- 12.2 MVA Development
- 12.3 The MVA Algorithm
- 12.4 Balanced Systems
- 12.5 MVA Extensions and Limitations
- 12.6 Chapter Summary
- 12.7 Exercises

Balanced Systems

- The MVA iteration starts once *the customer distribution among the devices is known*. That is, knowing how n – 1 customers are distributed among the devices, the performance measures when there are n customers in the system follow directly, as seen from the MVA algorithm given in Table 12.5.
- Now consider a balanced system. A system is considered to be balanced if a typical customer places the same average Demand (D) on each of the devices. This implies that all devices are equally utilized.

Performance Upper Band

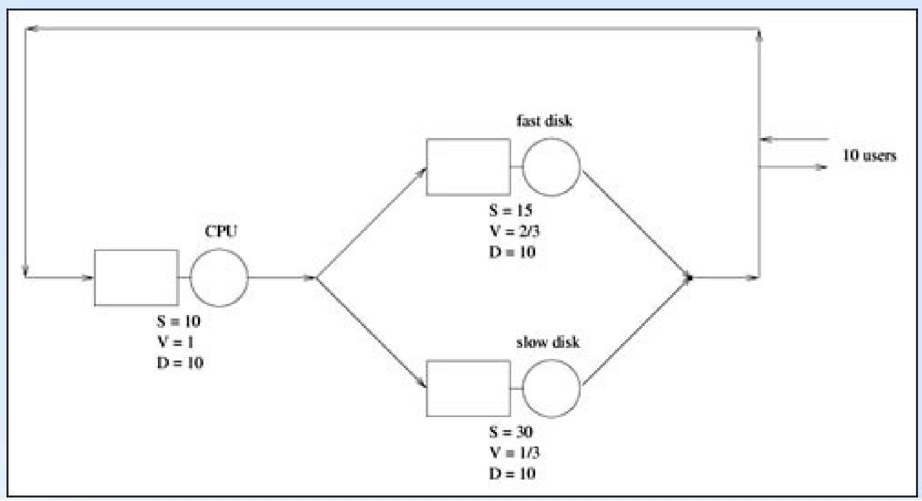
- A balanced system is not one where all devices are the same speed, only that the faster devices are either visited more often or the demand per visit to them is higher.
- A balanced system implies that there is no single bottleneck in the system.
- Balanced systems are important to consider, since they provide an upper bound on performance, a gold standard toward which to aspire.

Example

- For example, reconsider the database server example. From Tables 12.1 and 12.4 , the slow disk has the highest utilization and is the bottleneck .So the system is not balanced.
- Because the slow disk is over-utilized compared to the other devices, one way to improve performance would be to move some of the files from the slow disk to the fast disk. This has the effect of reducing the load (and utilization) of the slow disk and increasing the load (and utilization) of the fast disk.

Example'

- By moving disk files so that the fast disk is visited twice as often as the slow disk, the overall system becomes balanced. Now consider this balanced system with 10 customers in the system, as shown in Figure 12.4. All device demands (D's) are equal.
- Instead of running 10 iterations of MVA, because the system is balanced, only one MVA step is required;



One Iteration

- Recall that the only thing necessary (the iteration basis) for finding the performance measures for 10 customers is for MVA to know the average number of customers at each device when there are only 9 customers in the system ($\tilde{n}_i(9)$ for each device i).
- Since the system is balanced, the 9 customers are equally distributed among devices with 3 customers being at each of the 3 devices.
- Knowing that $\tilde{n}_i(9) = 3$ for each i, from the MVA algorithm in Table 12.5, it follows that the average residence time for each device i is : $R_i'(n) = D_i\left[1 + \bar{n}_i(n-1)\right]$

Calculations

- The overall system response time is:

$$R'_{cp}(10) = 10(1 + 3.0000) = 40.00 \text{ seconds}$$
$$R'_{fd}(10) = (2/3)\ 15(1 + 3.0000) = 40.00 \text{ seconds}$$
$$R'_{sd}(10) = (1/3)\ 30(1 + 3.0000) = 40.00 \text{ seconds}$$

- The overall system throughput is:

$$(R_0(n) = \sum_{i=1}^{K} R'_i(n))$$
$$R_0(10) = 40.00 + 40.00 + 40.00 = 120.00 \text{ seconds}$$

$$(X_0(n) = \frac{n}{R(n)}).$$
$$X_0(10) = \frac{10}{120.00} = 0.0833 \text{ customers per second}$$

Calculations.

- The throughput for each device i ($X_i(n) = V_i \times X_0(n)$) is:

$$
\begin{aligned}
X_{cp}(10) &= 1.0(0.0833) = 0.0833 \text{ customers per second} \\
X_{fd}(10) &= \frac{2}{3}(0.0833) = 0.0556 \text{ customers per second} \\
X_{sd}(10) &= \frac{1}{3}(0.0833) = 0.0278 \text{ customers per second}
\end{aligned}
$$

- The utilization for each device i ($U_i(n) = S_i \times X_i(n)$) is:

$$
\begin{aligned}
U_{cp}(10) &= 10.00(0.0833) = 0.8333 = 83.33\% \\
U_{fd}(10) &= 15.00(0.0556) = 0.8333 = 83.33\% \\
U_{sd}(10) &= 30.00(0.0278) = 0.8333 = 83.33\%
\end{aligned}
$$

Calculations..

- Finally, the average number of customers at each device i ($\bar{n}_i(n) = X_0(n) \times R_i'(n)$) is:

$$
\begin{aligned}
\bar{n}_{cp}(10) &= 0.0833(40.00) = 3.3333 \text{ customers} \\
\bar{n}_{fd}(10) &= 0.0833(40.00) = 3.3333 \text{ customers} \\
\bar{n}_{sd}(10) &= 0.0833(40.00) = 3.3333 \text{ customers}
\end{aligned}
$$

Balanced Calculation

- In balanced systems, all the device demands (D_i's) are equivalent. Let this common device demand be D. Therefore, finding the overall system response time can be simplified to:

$$
\begin{aligned}
R_0(n) &= \sum_{i=1}^{K}\left[V_i \times R_i'(n)\right] \\
&= \sum_{i=1}^{K}\left[V_i S_i\left[1 + \bar{n}_i(n-1)\right]\right] \\
&= \sum_{i=1}^{K}\left[D_i\left[1 + \bar{n}_i(n-1)\right]\right] \\
&= \sum_{i=1}^{K}\left[D\left[1 + \frac{n-1}{K}\right]\right] \\
&= KD\left[1 + \frac{n-1}{K}\right] \\
&= KD + Dn - D \\
&= D(K + n - 1)
\end{aligned}
$$

Balanced System Equation

- Overall system throughput is simply:

$$X_0(n) = \frac{n}{R_0(n)} = \frac{n}{D(K+n-1)}.$$

- As a verification in the balanced database server example, where

  n = 10, D = 10, and K = 3,

  the overall system response time is $R_0(10) = 10(3 + 10 – 1) = 120$ seconds and the overall system throughput is

  $X_0\ (10) = 10/120 = 0.0833$ customers/second.

Chapter 12-Outlines

MVA Extensions and Limitations

- MVA algorithm has been the focus of much researchs. These include:

1. Multi-class networks
2. Networks with load dependent servers
3. Networks with open and closed classes of customers

- The extension of MVA to product form, load-independent, multi-class networks is the topic of Chapter 13 . Chapter 14 extends MVA and the treatment of multiclass open QNs to the load-dependent case. Approximations to deal with non-product form QNs are presented in Chapter 15.

limitations and shortcomings surrounding MVA

1. MVA does not provide the steady state probabilities of individual system states.
2. MVA does not provide transient analysis information.
3. MVA does not model state dependent behavior.
4. MVA solves product form networks. As a result, MVA is not directly applicable to non-product form situations.

Chapter 12-Outlines

- 12.1 Introduction
- 12.2 MVA Development
- 12.3 The MVA Algorithm
- 12.4 Balanced Systems
- 12.5 MVA Extensions and Limitations
- 12.6 Chapter Summary
- 12.7 Exercises

Chapter Summary

- The Mean Value Analysis technique is arguably one of the most significant contributions to the field of performance evaluation within the past 25 years. It is the primary solution engine behind the large majority of state-of-the-art analytical solution packages currently in use. MVA is intuitive, elegant, and simple.
- This chapter first motivates, then develops, then summarizes, then applies, and finally qualifies MVA. Examples from the database server example introduced in previous chapters are used to demonstrate MVA. The following exercises are intended to reinforce and to broaden the reader's understanding and range of applicability of MVA.