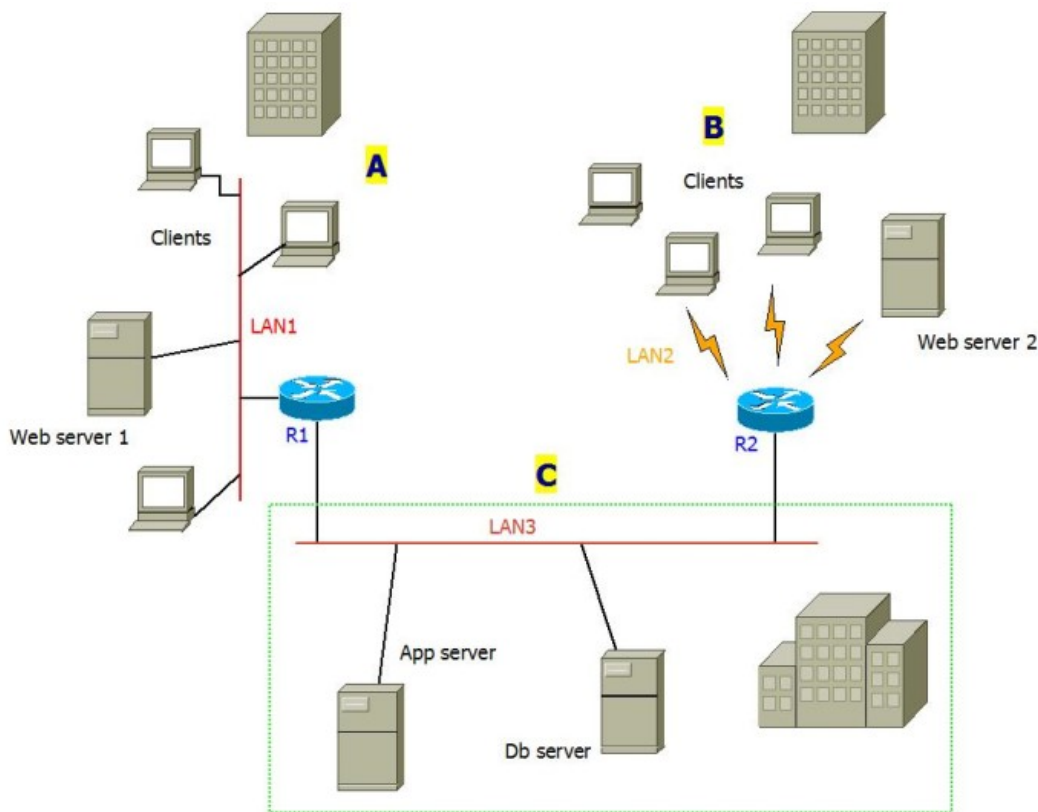


**Λύσεις Πρώτης Εργασίας
για το μάθημα
Επίδοση Υπολογιστικών Συστημάτων**
Εθνικό Μετσόβιο Πολυτεχνείο, Σχολή ΗΜΜΥ

Σε αυτό το θέμα θα μελετήσουμε την επίδοση της ιδιωτικής δικτύωσης (intranet) μίας εταιρίας. Η μορφή του υπολογιστικού συστήματος που περιγράφεται στην εκφώνηση είναι η ακόλουθη:



Σε αυτό το σύστημα έχουμε δύο ειδών αιτήσεις (CL1 και CL2) με διαφορετικούς μέσους χρόνους εξυπηρέτησης η κάθε μία σε κάθε μέρος του δικτύου. Στην πρώτη κατηγορία υπάρχουν 304 τερματικοί σταθμοί, ενώ στην δεύτερη κατηγορία υπάρχουν 240 σταθμούς εργασιών. Επειδή οι πελάτες που στέλνουν τις δύο ειδών αιτήσεις έχουν χρόνο σκέψης θα μοντελοποιήσουμε τους πελάτες ως σταθμούς καθυστέρησης με μέσο χρόνο εξυπηρέτησης ανά εργασία ίσο με τον χρόνο σκέψης των πελατών. Ιδιαίτερη αναφορά χρίζουν επίσης οι σταθμοί R1 και R2 που είναι επίσης σταθμοί καθυστέρησης (όπως αναφέρεται και στην εκφώνηση της ασκήσεως) καθώς και ο σταθμός Dd server CPU που είναι σταθμός που έχει ρυθμό εξυπηρέτησης εξαρτώμενο από το φορτίο. Οι υπόλοιποι σταθμοί θα είναι σταθμοί αναμονής με μέσο χρόνο εξυπηρέτησης ίσο με τον μέσο χρόνο εξυπηρέτησης που δίνεται στην εκφώνηση.

Ο ρυθμός εξυπηρέτησης του Dd Server CPU συναρτήσεται των εργασιών k που βρίσκονται σε αυτόν τον σταθμό (ανεξαρτήτως κατηγορίας) δίνεται προσεγγιστικά από την σχέση:

$$a(k) = \begin{cases} 0.4 + 0.6 \cdot k, & k \leq 64 \\ 38.8, & k > 64 \end{cases}, \text{ όπου } \mu(k) = a(k) \cdot \mu(1)$$

Με βάση τα παραπάνω και από τα δεδομένα της εκφώνησης για τους διάφορους σταθμούς του συστήματος η μέση συνολική απαίτηση εξυπηρέτησης σε msec είναι:

A/A Σταθμού	Σταθμός	CL1	CL2
1	Clients	25000	29000
2	LAN1	32	-
3	LAN2	-	90
4	LAN3	48	134
5	R1	25	-
6	R2	-	16

7	Web Server 1 CPU	59	-
8	Web Server 1 Disk	70	-
9	Web Server 2 CPU	-	28
10	Web Server 2 Disk	-	25
11	App Server CPU	48	58
12	App Server Disk	54	66
13	Dd Server CPU (όταν έχουμε μία εργασία)	69	106
14	Dd Server Disk 1	67	72
15	Dd Server Disk 2	88	96

Για τον σταθμό Dd Server CPU στον παραπάνω πίνακα αναφέρεται η μέση συνολική απαίτηση εξυπηρέτησης όταν έχουμε μόνο μία εργασία στον Dd Server CPU. Όπως προαναφέρθηκε αυτός ο σταθμός είναι Load Dependent. Για να βρούμε λοιπόν την μέση συνολική απαίτηση εξυπηρέτησης αυτού του σταθμού για οποιοδήποτε αριθμό εργασιών ακολουθούμε τα εξής:

$$D(k) = \frac{1}{\mu(k)} \Rightarrow D(k) = \frac{1}{\alpha(k) \cdot \mu(1)} \quad \text{και} \quad D(1) = \frac{1}{\mu(1)}, \quad \text{άρα}$$

$$D(k) = \frac{D(1)}{\alpha(k)} \quad (1)$$

όπου $D(1)=69\text{msec}$ για εργασίες της κατηγορίας CL1 και $D(1)=106\text{msec}$ για εργασίες της κατηγορίας CL2.

Με βάση λοιπόν τα παραπάνω μπορούμε να προχωρήσουμε στην μελέτη του συστήματος με το εργαλείο JMT αλλά και με προγραμματισμό δικού μας μοντέλου.

Χρήση του εργαλείου JMT/JMVA

Στο JMT θα χρησιμοποιήσουμε το αρχείο JMVA. Πρωτού προχωρήσουμε στην ανάλυσή μας θα πρέπει να αναφερθεί ότι επειδή το μοντέλο χρειάζεται μεγαλύτερο μέγεθος heap από αυτό που ξεκινάει by default το πρόγραμμα θα πρέπει να ξεκινήσουμε το πρόγραμμα με μεγαλύτερο μέγεθος heap. Για να το κάνουμε αυτό θα πρέπει να εντοπίσουμε το αρχείο JMT.jar στο σύστημά μας και στην συνέχεια θα πρέπει να εκτελέσουμε την εντολή

```
java -Xmx6144m -jar JMT.jar
```

όπου αυτή η εντολή αυξάνει το μέγεθος του heap σε 6GB.

Θα ξεκινήσουμε λοιπόν ορίζοντας δύο κλάσσεις πελατών, που είναι κλειστοί, την κλάση CL1 και την κλάση CL2. Η κλάση CL1 έχει 304 πελάτες, ενώ η κλάση CL2 έχει 240 πελάτες.

JMVA - Product Form Queueing Network Solver - ep1v2.jmva

File Action Help

Algorithm: MVA

Classes Stations Service Times Visits Reference Station What-if Comment

Classes characteristics Number: 2

Number, customized name, type of classes and number of customers (closed class) or arrival rate (open class). Add classes one by one or define total number at once. Higher number means higher priority

*	Name	Type	Priority	No. of Customers	Arrival Rate (λ)
1	CL1	closed	0	304	
2	CL2	closed	0	240	

New Class

Στην συνέχεια θα ορίσουμε τους σταθμούς του συστήματος. Όπως προαναφέρθηκε θα μοντελοποιήσουμε και τους Clients ως ένα σταθμό καθυστέρησης αφού έχουν χρόνο σκέψης. Επίσης σταθμοί καθυστέρησης θα είναι και οι σταθμοί R1 και R2. Ο σταθμός Dd Server CPU είναι load dependent σταθμός και στο επόμενο στάδιο θα θέσουμε τις τιμές του ανάλογα με τον αριθμό των εργασιών σε αυτών όπως προαναφέρθηκε. Οι υπόλοιποι σταθμοί τίθενται ως load independent.

Number: 15

Stations characteristics
Number, customized name and type of stations. Add stations one by one or define the total number at once. Load Dependent stations necessarily require the use of MVA.

New Station

*	Name	Type	
1	Clients	Delay (Infinite Server)	X
2	LAN1	Load Independent	X
3	LAN2	Load Independent	X
4	LAN3	Load Independent	X
5	R1	Delay (Infinite Server)	X
6	R2	Delay (Infinite Server)	X
7	WS1 CPU	Load Independent	X
8	WS1 Disk	Load Independent	X
9	WS2 CPU	Load Independent	X
10	WS2 Disk	Load Independent	X
11	AppS CPU	Load Independent	X
12	AppS Disk	Load Independent	X
13	DdS CPU	Load Dependent	X
14	DdS Disk1	Load Independent	X
15	DdS Disk2	Load Independent	X

Στο επόμενο στάδιο θα θέσουμε τους χρόνους μέσης συνολικής απαίτησης εξυπηρέτησης. Οι μέσοι χρόνοι απαίτησης εξυπηρέτησης θα τεθούν σε όλα τα στάθμους οι μέσοι χρόνοι απαίτησης εξυπηρέτησης που αναγράφονται στα δεδομένα, εκτός φυσικά από τον σταθμό Dd Server CPU που θα αναφερθούμε στην συνέχεια. Επειδή δεν περνάνε και οι δύο κατηγορίες εργασιών από όλους τους σταθμούς (αυτό φαίνεται και από τα δεδομένα όπου ο μέσος χρόνος απαίτησης εξυπηρέτησης κάποιον εργασιών σε κάποιους σταθμούς έχει παύλα(-)) για τους σταθμούς που ένα είδος εργασίες δεν περνάει από αυτούς θα θέσουμε μέσο χρόνο απαίτησης εξυπηρέτησης για τις εργασίες αυτού του είδους 0msec.

Service Times
Input service times of each station for each class.
If the station is "Load Dependent" you can set the service times for each number of customers by double-click on "LD Settings..." button.
Press "Service Demands" button to enter service demands instead of service times and visits.
MULTICLASS MODELS: when for a station the per-class service times are different, the results are correct ONLY if its scheduling discipline is assumed Processor Sharing (PS) and not FCFS (See BCMP Theorem).

*	CL1	CL2
Clients	25000.0000	29000.0000
LAN1	32.0000	0.0000
LAN2	0.0000	90.0000
LAN3	48.0000	134.0000
R1	25.0000	0.0000
R2	0.0000	16.0000
WS1 CPU	59.0000	0.0000
WS1 Disk	70.0000	0.0000
WS2 CPU	0.0000	28.0000
WS2 Disk	0.0000	25.0000
AppS CPU	48.0000	58.0000
AppS Disk	54.0000	66.0000
DdS CPU	LD Settings...	LD Settings...
DdS Disk1	67.0000	72.0000
DdS Disk2	88.0000	96.0000

Service Demands

Επίσης πατάμε το κουμπί service demands (που φαίνεται στην παραπάνω εικόνα κάτω αριστερά) καθώς γνωρίζουμε τις μέσες συνολικές απαιτήσεις εξυπηρέτησης κάθε σταθμού για κάθε κατηγορία εργασιών και όχι τον μέσο χρόνο επίσκεψης και τον αριθμό των επισκέψεων σε κάθε σταθμό για κάθε κατηγορία εργασιών.

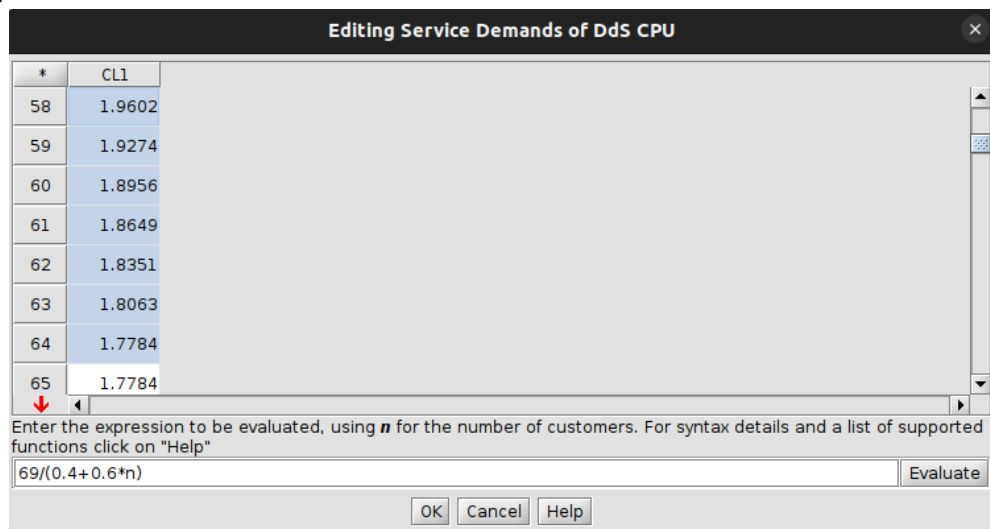
Έπειτα θα υπολογίσουμε τις μέσες συνολικές απαιτήσεις εξυπηρέτησης για τον χρόνο Dd Server CPU που είναι Load Dependent. Για τον υπολογισμό της μέσης συνολικής απαίτησης εξυπηρέτησης συναρτήσει του αριθμού των εργασιών εντός του σταθμού Dd Server CPU θα χρησιμοποιήσουμε την εξίσωση (1) που βρέθηκε παραπάνω. Έτσι για την κλάση εργασιών CL1 έχουμε ότι η μέση συνολική απαίτηση εξυπηρέτησης στο σταθμό Dd Server CPU συναρτήσει των εργασιών εντός του σταθμού είναι:

$$D(k) = \begin{cases} \frac{69}{0.4 + 0.6 \cdot k}, & \text{για } k \leq 64 \\ 1.7784, & \text{για } k > 64 \end{cases}$$

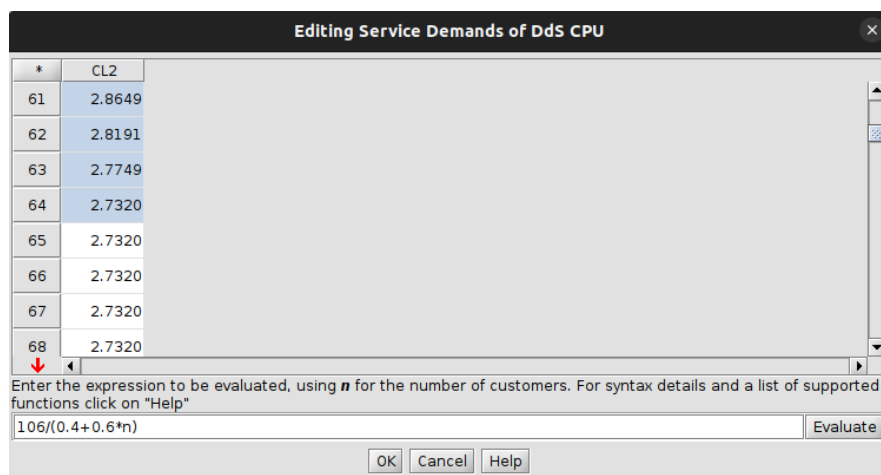
ενώ για εργασίες της κλάσης CL2 είναι:

$$D(k) = \begin{cases} \frac{106}{0.4 + 0.6 \cdot k}, & \text{για } k \leq 64 \\ 2.7320, & \text{για } k > 64 \end{cases}$$

Στην συνέχεια πατάμε για κάθε μία από τις δύο εργασίες πατάμε στο LD Settings κουμπί που βρίσκεται στην στήλη της κάθε κλάσης και κάνουμε evaluate τις δύο αυτές συναρτήσεις. Έτσι το evaluation της μέσης συνολικής απαίτησης της κλάσης εργασιών CL1 στον σταθμό Dd Server CPU είναι:



ενώ για την κλάση εργασιών CL2 είναι:



Οι σταθερές τιμές μου φαίνονται στις δύο παραπάνω δίκλαδες εξισώσεις υπολογισμού των μέσων συνολικών απαιτήσεων εξυπηρέτησης φτάνουν μέχρι και την τιμή 544, αφού στον σταθμό αυτό μπορούν να είναι το πολύ όλες οι εργασίες του δικτύου.

Αφού έχουμε κάνει όλα τα παραπάνω και επιλέγοντας τον αλγόριθμο MVA μπορούμε να δούμε τα αποτελέσματα της προσομοίωσής μας.

Ξεκινώντας με το throughput το αποτέλεσμα είναι:

Throughput			
Throughput of each class for each station. System Throughput is the completion rate at the Reference Station .			
*	Aggregate	CL1	CL2
System	0.0110	0.0065	0.0045
Clients	0.0110	0.0065	0.0045
LAN1	0.0110	0.0065	0.0045
LAN2	0.0110	0.0065	0.0045
LAN3	0.0110	0.0065	0.0045
R1	0.0110	0.0065	0.0045
R2	0.0110	0.0065	0.0045
WS1 CPU	0.0110	0.0065	0.0045
WS1 Disk	0.0110	0.0065	0.0045
WS2 CPU	0.0110	0.0065	0.0045
WS2 Disk	0.0110	0.0065	0.0045
AppS CPU	0.0110	0.0065	0.0045
AppS Disk	0.0110	0.0065	0.0045
DdS CPU	0.0110	0.0065	0.0045
DdS Disk1	0.0110	0.0065	0.0045
DdS Disk2	0.0110	0.0065	0.0045

Έπειτα ο μέσος χρόνος απόκρισης υπολογίστηκε:

Residence Times			
Total time spent by each customer class summed across all visits to a station.			
*	Aggregate	CL1	CL2
Aggregate	--	--	--
Clients	26631.5...	25000.0...	29000.0...
LAN1	23.9020	40.3669	0.0000
LAN2	61.2495	0.0000	150.1654
LAN3	890.8089	519.9695	1429.15...
R1	14.8030	25.0000	0.0000
R2	6.5261	0.0000	16.0000
WS1 CPU	56.5269	95.4654	0.0000
WS1 Disk	75.7828	127.9857	0.0000
WS2 CPU	13.0493	0.0000	31.9929
WS2 Disk	11.4758	0.0000	28.1353
AppS CPU	121.2937	111.8063	135.0667
AppS Disk	166.0486	152.2743	186.0448
DdS CPU	101.2514	83.0840	127.6251
DdS Disk1	283.5151	275.1306	295.6870
DdS Disk2	21189.3...	20424.2...	22300.0...

Συνεχίζοντας ο μέσος αριθμός εργασιών σε κάθε σταθμό του συστήματος είναι:

Number of Customers			
Average number of customers for each class at each station.			
*	Aggregate	CL1	CL2
Aggregate	544.0000	304.0000	240.0000
Clients	291.8106	162.2014	129.6092
LAN1	0.2619	0.2619	0.0000
LAN2	0.6711	0.0000	0.6711
LAN3	9.7609	3.3736	6.3873
R1	0.1622	0.1622	0.0000
R2	0.0715	0.0000	0.0715
WS1 CPU	0.6194	0.6194	0.0000
WS1 Disk	0.8304	0.8304	0.0000
WS2 CPU	0.1430	0.0000	0.1430
WS2 Disk	0.1257	0.0000	0.1257
AppS CPU	1.3291	0.7254	0.6037
AppS Disk	1.8195	0.9880	0.8315
DdS CPU	1.1094	0.5391	0.5704
DdS Disk1	3.1066	1.7851	1.3215
DdS Disk2	232.1788	132.5137	99.6651

Τέλος ο βαθμός χρησιμοποίησης των σταθμών του συστήματος είναι:

Utilization			
Utilization of a customer class at the selected station. The utilization of a delay station is the average number of customers greater than 1).			
*	Aggregate	CL1	CL2
Aggregate	--	--	--
Clients	291.8106	162.2014	129.6092
LAN1	0.2076	0.2076	0.0000
LAN2	0.4022	0.0000	0.4022
LAN3	0.9103	0.3114	0.5989
R1	0.1622	0.1622	0.0000
R2	0.0715	0.0000	0.0715
WS1 CPU	0.3828	0.3828	0.0000
WS1 Disk	0.4542	0.4542	0.0000
WS2 CPU	0.1251	0.0000	0.1251
WS2 Disk	0.1117	0.0000	0.1117
AppS CPU	0.5706	0.3114	0.2592
AppS Disk	0.6453	0.3504	0.2950
DdS CPU	0.4737	0.0000	0.4737
DdS Disk1	0.7565	0.4347	0.3218
DdS Disk2	1.0000	0.5709	0.4291

Έχοντας λοιπόν υπολογίσει όλα τα παραπάνω μπορούμε να προχωρήσουμε στον προγραμματισμό της δικής μας προσομοίωσης και να διασταυρώσουμε τα αποτελέσματα.

Προγραμματισμός Προσεγγιστικής Μεθόδου MVA

Σε αυτό το σημείο της εργασίας θα προγραμματίσουμε σε C++ την προσεγγιστική μέθοδο MVA ώστε να υπολογίσουμε όσα υπολογίσαμε και προηγουμένως και να συγκρίνουμε τα αποτελέσματα που υπολογίστηκαν με τις δύο μεθόδους.

Ο αλγόριθμος που χρησιμοποιήθηκε είναι ο προσεγγιστικός αλγόριθμος που περιγράφεται στην σελίδα 106 του βιβλίου που περιλαμβάνει και τον αλγόριθμο Bard-Schweitzer που περιγράφεται στην σελίδα 100 του βιβλίου και τον χρησιμοποιούμε για να υπολογίσουμε το R και το Q των σταθμών αναμονής και καθυστέρησης (δηλαδή τους σταθμούς delay και load independent). Ο

αλγόριθμος αυτός θα εκτελείται επαναληπτικά μέχρι το throughput και για τις δύο κλάσεις μεταξύ δύο διαδοχικών επαναλήψεων να διαφέρει λιγότερο από 0.0000001.

Ο κώδικας γράφτηκε για την προσομοίωση του συστήματος που ζητείται στην εκφώνηση. Επομένως θεωρεί ότι μόνο ένας σταθμός είναι LD και είναι ο 12 και ότι έχουμε μόνο 2 διαφορετικές κατηγορίες εργασιών. Ο κώδικας της προσομοίωσης που γράψαμε βρίσκεται εντός του zip που υποβλήθηκε.

Σχετικά με τον κώδικα, ένα run του approximate MVA γίνεται τρέχοντας την συνάρτηση `appgMVA()`. Επομένως για να διαφέρει το throughput μεταξύ δύο διαδοχικών runs και για τις δύο κλάσεις λιγότερο από 0.0000001, όπως αναφέρθηκε προηγουμένως, τρέχουμε συνεχώς την συνάρτηση `appgMVA()` μέχρι να επιτευχθεί αυτό. Όταν αυτό επιτευχθεί εκτυπώνουμε τα αποτελέσματα. Οι συναρτήσεις `a(int n)` και `rhoCalc()` αποτελούν βοηθητικές συναρτήσεις για τον υπολογισμό του R των Load Dependent σταθμών. Τέλος η συνάρτηση `printSeperator(int length)` χρησιμεύει στην εκτύπωση των αποτελεσμάτων.

Τρέχοντας τον αλγόριθμο με τα δεδομένα της άσκησης το αποτέλεσμα είναι το ακόλουθο:

Throughput				
+-----+				
	CL1:	0.00647		
	CL2:	0.00446		
+-----+				
Residence time				
+-----+				
	Station	CL1	CL2	
+-----+				
	Clients	25000.00000	29000.00000	
	LAN1	40.32441	0.00000	
	LAN2	0.00000	149.90553	
	LAN3	518.38757	1445.04419	
	R1	25.00000	0.00000	
	R2	0.00000	16.00000	
	WS1 CPU	95.25636	0.00000	
	WS1 Disk	127.64019	0.00000	
	WS2 CPU	0.00000	31.97542	
	WS2 Disk	0.00000	28.12169	
	AS CPU	111.17706	134.33148	
	AS Disk	151.10333	184.66776	
	DdS CPU	83.05237	127.58769	
	DdS Disk1	271.59698	291.89133	
	DdS Disk2	20545.94336	22415.77148	
+-----+				
Number of Customers				
+-----+				
	Station	Aggregate	CL1	CL2
+-----+				
	Clients	291.11438	161.80717	129.30722
	LAN1	0.26099	0.26099	0.00000
	LAN2	0.66841	0.00000	0.66841
	LAN3	9.79842	3.35515	6.44326
	R1	0.16181	0.16181	0.00000
	R2	0.07134	0.00000	0.07134
	WS1 CPU	0.61653	0.61653	0.00000
	WS1 Disk	0.82612	0.82612	0.00000
	WS2 CPU	0.14257	0.00000	0.14257
	WS2 Disk	0.12539	0.00000	0.12539
	AS CPU	1.31854	0.71957	0.59897
	AS Disk	1.80139	0.97798	0.82341
	DdS CPU	1.10644	0.53754	0.56890
	DdS Disk1	3.05936	1.75785	1.30151
	DdS Disk2	232.92825	132.97925	99.94901
+-----+				
Utilization				
+-----+				
	Station	Aggregate	CL1	CL2
+-----+				
	Clients	291.11438	161.80717	129.30722
	LAN1	0.20711	0.20711	0.00000
	LAN2	0.40130	0.00000	0.40130
	LAN3	0.90816	0.31067	0.59749
	R1	0.16181	0.16181	0.00000
	R2	0.07134	0.00000	0.07134
	WS1 CPU	0.38186	0.38186	0.00000
	WS1 Disk	0.45306	0.45306	0.00000
	WS2 CPU	0.12485	0.00000	0.12485
	WS2 Disk	0.11147	0.00000	0.11147
	AS CPU	0.56928	0.31067	0.25861
	AS Disk	0.64379	0.34950	0.29429
	DdS CPU	0.91923	0.44659	0.47264
	DdS Disk1	0.75468	0.43364	0.32104
	DdS Disk2	0.99761	0.56956	0.42805
+-----+				

Παρατηρούμε λοιπόν ότι τα αποτελέσματα του προσεγγιστικού αλγόριθμου που γράψαμε είναι πολύ κοντά με τα δεδομένα που υπολογίστηκαν με τον αλγόριθμο MVA μέσω του JMT. Οι μικρές διαφορές που παρατηρούμε είναι πιθανό να οφείλονται στο ότι στην δικιά μας υλοποίηση χρησιμοποιήσαμε την προσεγγιστική μέθοδο, ενώ στο JMT/JMVA χρησιμοποιήθηκε η ακριβής μέθοδος MVA. Επίσης βλέπουμε ότι το JMT/JMVA είναι πιο αργό στην εκτέλεση για την εύρεση των αποτελεσμάτων σε σχέση με το πρόγραμμα που γράψαμε εμείς (το JMT χρειάζεται περίπου 5 sec, ενώ το πρόγραμμα που γράψαμε χρειάζεται περίπου 0.05sec), ωστόσο ο χρόνος προετοιμασίας του JTM/JMVA είναι πολύ μικρότερος σε σχέση με τον χρόνο για να γράψουμε τον δικό μας κώδικα για τον προσεγγιστικό αλγόριθμο MVA.

Σύνοψη Αποτελεσμάτων

Από τα αποτελέσματα που προέκυψαν από τον προσεγγιστικό αλγόριθμο MVA και από το πρόγραμμα JMT/JMVA καταλήγουμε στο ότι η επίδοση του συστήματος είναι κακή, καθώς το throughput του συστήματος είναι πολύ χαμηλό. Παρατηρούμε επίσης ότι κάποιοι σταθμοί έχουν πολύ υψηλό utilization σε σύγκριση με άλλους πράγμα που δείχνει ότι έχουμε bottleneck και δεν εκμεταλλευόμαστε βέλτιστα όλους τους πόρους που διαθέτουμε για το σύστημα.

Για να βελτιώσουμε αυτή την επίδοση θα προτείναμε την αντικατάσταση του Dd Server Disk 2 με έναν πιο γρήγορο καθώς βλέπουμε ότι υπάρχουν πολλές εργασίες σε αυτόν κατά μέσο όρο και ο μέσος χρόνος απόκρισής του είναι πολύ μεγάλος ή την αντικατάστασή του με περισσότερους του ενός δίσκους ώστε να μοιραστεί η δουλειά που κάνει. Επίσης θα προτείναμε την αύξηση της ταχύτητας του LAN3 με έναν πιο γρήγορο καθώς και αυτό έχει μεγάλο μέσο αριθμό εργασιών εργασιών και μεγάλο μέσο χρόνο απόκρισης. Μία τελευταία αναβάθμιση που θα βοηθούσε θα ήταν η αντικατάσταση του Dd Server Disk 1 με έναν πιο γρήγορο για παρόμοιους λόγους με τις δύο προηγούμενες αναβαθμίσεις.