Journal of the Korean
OR/MS Society
Vol. 16, No.2, Dec., 1991

Capacity Planning in a Closed Queueing Network

Juho Hahm*

Research Institute of Engineering Science

Seoul National University, Seoul, Korea

Abstract

In this paper, criteria and algorithms for the optimal service rate in a closed queueing network have been established. The objective is to minimize total cost. It is shown that system throughput is increasing concave over the service rate of a node and cycle time is increasing convex over the set of service times with a single class of customers. This enables developing an algorithm using a steepest descent method when the cost function for service rate is convex. The efficiency of the algorithm rests on the fact that the steepest descent direction is readily obtained at each iteration from the MVA algorithm. Sereral numerical examples are presented. The major application of this research is optimization of facility capacity in a manufacturing system.

1. Introduction

The capacity planning problem in a network of queues has been extensively studied in the last decade. The main objective of capapcity planning is to optimally determine the capacity of production facility to minimize the cost. Many manufacturing system can be modeled as a closed queueing network.

For a closed queueing network, Dallery and Grenoble [1] have solved the problem which optimally allocates servers to minimize the cost of servers satisfying the required throughput level. For open queueing network, Shanthikumar and Yao [5, 6, 7] have dealt with the problem which allocates given number of servers and buffers to minimize the cost of servers and throughput.

In this paper, the capacity is the service(production) rate(or time) of a production unit, and each production unit consists of a node in a closed queueing network.

* *Research Institute of Engineering Science, Seoul National University,*

Assumptions made in this paper are:

-Visit ratios are predetermined for all nodes

-Number of customers(pallets) in the system is predetermined.

-Service time of each node is exponentially distributed.

-Service policy is First Come First Serve(FCFS).

-Each node has infinite buffer space.

-Service time of each node is load-independent.

-Cost function associated with service rate is increasing convex, continuous, and differentiable.

-Profit function associated with system throughput is increasing linear.

-Cost function associated with cycle time is increasing linear.

2. Models

Notation

v_i : vist ratio of node j,

μ_i : service rate of node j,

s_i : service time of node j(i.e. $1/\mu_i$),

$\lambda_i(N)$: throughput of node j,

$\lambda(N)$: throughtput of a system

$CT(N)$: cycle time of a system

$Q_i(N)$: mean queue length of node j,

$T_i(N)$: second moment of queue length of node j,

$R_i(N)$: mean queue time of node j,

J : number of nodes in a system,

c_i : cost coefficient of service rate of node j,

p_i : exponent of service rate corresponding to cost(≥ 1),

c_o^t : profit associated with unit system throughput

c_o^c : cost associated with unit cycle time

Under the assumptions, the network satisfies the product-form solution for the equilibrium probability of the state(n_1, n_2, $\cdots$ n_J) :

$$P(n_1, n_2, \cdots n_J)=\frac{1}{G(N)}\left(\frac{v_1}{\mu_1}\right)^{n_1}\cdots\left(\frac{v_1}{\mu_i}\right)^{n_i}\cdots\left(\frac{v_1}{\mu_J}\right)^{n_J}$$

where n_j is the number of customers at queue j,

$$N=n_1+n_2+\cdots+n_J,$$

and $$G(N)=\sum_{n_1+n_2+\cdots+n_J=N}\left(\frac{v_1}{\mu_1}\right)^{n_1}\cdots\left(\frac{v_1}{\mu_i}\right)^{n_i}\cdots\left(\frac{v_1}{\mu_J}\right)^{n_J}$$

Then, the throughput of a system is expressed as :

$$\lambda(N)=\frac{G(N\text{-}1)}{G(N)}$$

Alternately, the throughput can be computed using the MVA algorithm.

MVA is based on the following recursive relations of the equilibrium quantities for queue j(j=1, 2, ···, J) :

$$R_i(N)=\frac{1}{\mu_i}[1+Q_i(N\text{-}1)]$$

$$CT(N)=\sum_{j=1}^{J}v_jR_j(N)$$

$$\lambda(N)=\frac{N}{CT(N)}$$

$$Q_i(N)=v_i\,\lambda(N)\,R_i(N)=\lambda_i(N)\,R_i(N)$$

3. Capacity Planning Problem

We now define the optimal capacity planning problems in a closed queueing network as follows :

Given : visit ratio [v_i], population N, and network topology,

Minimize : Total Cost[TC(N)]

Model(1) TC(N)=[cost of service]+[cost of cycle time]

Model(2) TC(N)=[cost of service]-[profit of throughput]

with respect to : service times[s_i]

under constraints : $\mu_i \geq 0 \ \forall$ j.

where cost of service is $\sum_{j=1}^{J}c_i\cdot(\mu_i)^{P_i}=\sum_{j=1}^{J}c_i\cdot\left(\frac{1}{S_i}\right)^{P_i}$,

profit of throughput is $c_o^t\cdot\lambda(N)$, and

cost of cycle time is $c_o^c\cdot CT(N)$

The first model deals with the situations when the number of throughputs needed per unit time is predetermined, hence the concern is the length of cycle time. Costs associated with cycle time are such as cost of money(interest) and utility costs, etc.

The second model deals with the situation when we can sell all units produced, therefore the concern in the number of throughputs per unit time.

The capacity planning problem is nonlinear programming problem and we use steepest descent method to solve this problem. The optimality of steepest descent method for Model(1) depends on the convexity of the objective function.

Lemma 1. The average delay $CT_N(\mathbf{s})$ is a convex function of service time vector s

where $\mathbf{s}=(s_1, s_2, \cdots, s_N)$.

Proof. The following inequality concerning a symmetric function was proved by Whitely [8] :

$$\frac{C_n(\mathbf{a}+\mathbf{b})}{C_{n-1}(\mathbf{a}+\mathbf{b})} \leq \frac{C_n(\mathbf{a})}{C_{n-1}(\mathbf{a})} + \frac{C_n(\mathbf{b})}{C_{n-1}(\mathbf{b})}. \tag{1}$$

where

$$C_n(\mathbf{x}) = \sum_{\substack{i_1+\cdots+i_m=n \\ i_i \geq 0}} x_1^{i_1} x_2^{i_1} x \cdots x_m^{i_m}. \tag{2}$$

and **a** and **b** are vectors of arbitrary nonnegative real numbers.

As we know by MVA,

$CT_N(\mathbf{s})=\frac{N \cdot G_N(\mathbf{s})}{G_{N-1}(\mathbf{s})}$ where $G_N(\mathbf{s})$ is the value of G(N) when service time vector is **s**.

Because $G_N(\mathbf{s})$ has the same form as that of $C_n(\mathbf{x})$, if eq. (1) to applied to $CT_N(\mathbf{s})$, we have :

$$CT_N(\alpha \mathbf{s}^1+(1-\alpha)\mathbf{s}^2) \leq \alpha\ CT_N(\mathbf{s}^1)+(1-\alpha)CT_N(\mathbf{s}^2). \tag{3}$$

Thus, function $CT_N(\mathbf{s})$ is convex.

It also can be proved that throughput is a concave function of service rate of a node.

Theorem 1. $\lambda(N)$ is a concave function of μ_i for any j, j=1 to N.

Proof. To prove Theorem 1, we need some preliminary results :

$$Q_i(N) \geq Q_i(N-1) \text{ for any j, } N \geq 1. \tag{4}$$

and $T_i(N)=\rho_i(N)[T_i(N-1)+2Q_i(N-1)+1]$ or (5)

$$=\rho_i(N)[T_i(N-1)+Q_i(N-1)]+Q_i(N) \text{ or} \tag{6}$$

$$=\rho_i(N)[T(N-1)-1]+2Q_i(N) \quad \forall_i,\ N\geq 1 \tag{7}$$

where $\rho_i(N)=\lambda_i(N)/\mu_i$.

The proofs of above equations are very straightforward and omitted here.

The second derivative of $\lambda(N)$ with respect to μ_i is

$$\frac{\partial^2}{\partial\mu_i^2}\lambda(N)=\frac{\lambda(N)}{\mu_i^2}\{[Q_i(N)-Q_i(N-1)]\cdot[2Q_i(N)-1]+T_i(N-1)-T_i(N)\}. \tag{8}$$

It needs to be shown that eq.(8) is nonnegative for any nonnegative integer N. Since $\frac{\lambda(N)}{\mu_i^2}$ is always nonnegative, let's look at only the value in a bracket in eq.(8) and let it be eq.(9).

Using equations (5), (6), (7) and mean value theorem in Kant [2], eq.(9) can be rewritten as :

$$[Q_i(N)-Q_i(N-1)]-[2Q_i(N)-1]+T_i(N-1)-T_i(N)$$

$$=(1-\rho_i(N))\{T_i(N-1)+Q_i(N-1)-2Q_i(N)\cdot[1+Q_i(N-1)]\} \tag{9}$$

An induction will be used to show that eq.(9) is nonpositive.

i) For N=1, eq.(9) is $-2(1-\rho_i(1))\cdot Q_i(1)$. Since $\rho_i(N)$ is always less than or equal to 1 and nonnegative for any N(refer to Kant[2]), this value is obviously nonpositive.

ii) For N=n, suppose eq.(9) is nonpositive, that is :

$$2Q_i(n)\cdot[1+Q_i(n-1)]\geq T_i(n-1)+Q_i(n-1)$$

$$=[T_i(n)-Q_i(n)]/\rho_i(n) \qquad \text{by eq.(6).}$$

Therefore,

$$T_i(n)+Q_i(n)\leq 2\rho_i(n)\cdot Q_i(n)\cdot(1+Q_i(n-1))+2Q_i(n)$$

$$=2Q_i(n)[\rho_i(n)\cdot(1+Q_i(n-1))]+2Q_i(n)$$

$$=2Q_i(n)\cdot Q_i(n)+2Q_i(n)$$

$$=2Q_i(n)\cdot(1+Q_i(n))$$

$$\leq 2Q_i(n+1)\cdot(1+Q_i(n)) \qquad \text{vy eq.(4).}$$

iii) For N=n+1, eq.(9) becomes :

$$(1-\rho_i(n+1))\ \{T_i(n)+Q_i(n)-2Q_i(n+1)\cdot[1+Q_i(n)]\} \tag{10}$$

In (ii), it is shown that :

$$T_i(n)+Q_i(n)\leq 2Q_i(n+1)\cdot(1+Q_i(n)).$$

Therefore, eq.(10) is also nonpositive.

As a result, the second derivative of $\lambda(N)$ with respect to μ_i is always nonpositive and thus, $\lambda(N)$ is a

concave function of service rate of node j.

But Theorem 1 does not necessarily mean that the throughput is a concave function over the set of service rates of all nodes. It is extremely hard to show that the throughput is a concave function over the set of service rates of all rodes. Therfore, it is not guaranteed that the the local minimum in Model(2) is a global minimum. However, the fact that the throughput is a concave function over the service rate of a node, suggest that it is quite likely that the throughput is a concave function over the set of service rates of all nodes. Even if this is not the case, it would not be meaningless to find an local minimum around the initially given service rate vector in a sense of real world. This motivates the development of an algorithm for Model(2).

In this paper, it is assumed that the cost of service times are given as convex function. Therefore, the steepest descent method leads to a global minimum in Model(1) and to a local minimum in Model(2). Moreover, if the problem is to find the best service rate of a node of a node while the service rates of other nodes are fixed, it is guaranteed that Model(2) finds a global minimum.

Next, the computation of steepest descent direction wiube discussed.

4. Steepest Descent Direction

To compute the steepest descent direction, we first compute the gradient of the normalization function G(N) with respect to μ : must be computed first

$$\frac{\partial G(N)}{\partial \mu_i} = \sum \frac{-v_i \cdot n_i}{\mu_i^2} \left(\frac{v_1}{\mu_1}\right)^{n_1} \cdots \left(\frac{v_i}{\mu_i}\right)^{n_i - 1} \cdots \left(\frac{v_J}{\mu_J}\right)^{n_J}$$

$$= \frac{-1}{\mu_i} \sum n_i \cdot \left(\frac{v_1}{\mu_1}\right)^{n_1} \cdots \left(\frac{v_i}{\mu_i}\right)^{n_i} \cdots \left(\frac{v_J}{\mu_J}\right)^{n_J}$$

$$= \frac{-G(N)}{\mu_i} \sum n_i \, P(n_1, n_2, \cdots, n_J)$$

$$= \frac{-G(N) \cdot Q_i(N)}{\mu_i}. \tag{11}$$

Analogously,

$$\frac{\partial \lambda(N)}{\partial s_i} = \frac{G(N) \cdot Q_i(N)}{s_i}. \tag{12}$$

Using the above results, we can express the derivative of $\lambda(N)$ as a function of G and Q :

$$\frac{\partial G(N)}{\partial \mu_i} = \frac{G(N)\cdot\frac{\partial G(N-1)}{\partial \mu_i} - G(N-1)\cdot\frac{\partial G(N)}{\partial \mu_i}}{[G(N)]^2}$$

$$= \frac{-G(N)\cdot G(N-1)\cdot Q_i(N-1) + G(N-1)\cdot G(N)\cdot Q_i(N)}{\mu_i\cdot[G(N)]^2}$$

$$= \frac{\lambda(N)}{\mu_i}\,[Q_i(N) - Q_i(N-1)]. \tag{13}$$

Aslso,

$$\frac{\partial CT(N)}{\partial s_i} = \frac{CT(N)}{s_i}\,[Q_i(N) - Q_i(N-1)]. \tag{14}$$

The last equation is computationally very convenient since every value on the right-hand side can be directly obtained from MVA algorithm. And it can be proved that equations(13) and (14) are always nonnegative using eq.(4). Therefore, the cycle time is an increasing function of service times and the throughput is an increasing function of service rates.

Using equations (13) and (14), we can easily get the steepest descent direction for Model(1) and Model(2).

5. Algorithm

Algorithm for Model 1.

Step 1. Set $n=0$ and let $\mathbf{s}^0$ be the initial service time vetor.

Step 2. Compute $\frac{\partial}{\partial s_i}\,CT(N)$ using MVA method.

Compute $\frac{\partial}{\partial s_i}\,TC(N)$ using $\frac{\partial}{\partial s_i}\,CT(N)$.

Let $\mathbf{d}^n=(d_i^n)$ where $d_i^n=\frac{\partial}{\partial s_i}\,TC(N)$.

*Step 3.*Using $\mathbf{d}$, find the best α^* using MVA such that

$$TC(N:\alpha^*) = \min_{\alpha} TC(N:\alpha)$$

Step 4. If $\| \nabla TC(N) \| \le \epsilon$, stop.

Otherwise $n=n+1$, let $\mathbf{s}^n=\mathbf{s}^n+\alpha^*\mathbf{d}^n$ and go to Step 2.

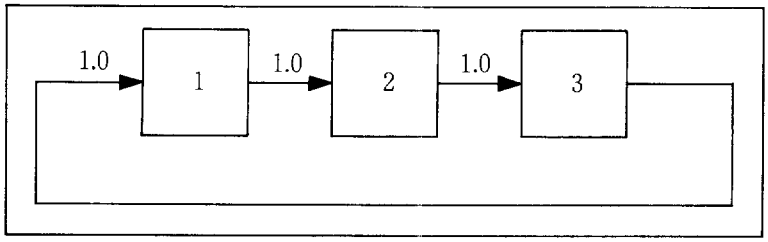
For Model(2), replace CT(N) with $\lambda(N)$ and, s^n with μ^n.

6. Numerical Examples

Here, algorithms are applied to several network examples. For each example, both Model(1) and Model(2) are applied to two cases where the cost of service rate increases linearly (Case 1) and quadratically (Case 2). Algrithms have been programmed in FORTRAN and was run on IBM-XT.

6.1 Example 1

The first example network has a following structure:



Costs

Case 1 : Model(1):$400 \cdot CT(N) + 20 \cdot \mu_1 + 20 \cdot \mu_2 + 20 \cdot \mu_3$

Model(2):$400 \cdot \lambda(N) + 20 \cdot \mu_1 + 20 \cdot \mu_2 + 20\mu_3$

Case 2 : Model(1):$400 \cdot CT(N) + 20 \cdot \mu_1^2 + 20 \cdot \mu_2^2 + 20 \cdot \mu_3$

Model(2):$400 \cdot \lambda(N) + 20 \cdot \mu_1^2 + 20 \cdot \mu_2^2 + 20 \cdot \mu_3^2$

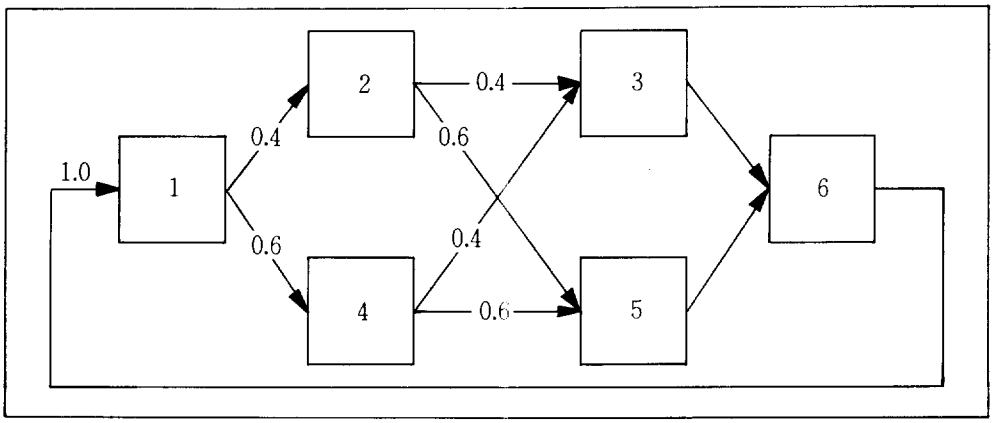
The service rate of node 3(μ_3) is fixed as 5. The problem is to determine the values of μ_1 and μ_2 to minimize cost. The initial values of μ_1 and μ_2 are given as 1 for cases.

Results

The summary of the results is as follows:

Case	Model	Optimal Service Rate		Minimum Cost
		Node 1	Node 2	
1	1	6.488408	6.488408	1204.723
	2	7.472469	7.472469	-1555.205
2	1	3.711713	3.711713	2271.477
	2	3.879365	3.879365	-261.836

2. Example 2



$c_0^c = c_0^l = 400$, $c_1 = 20$, $c_2 = 8$, $c_3 = 9$, $c_4 = 13$, $c_5 = 15$, $c_6 = 23$.

Service Rate for node 1 and node 6 are predetermined as 5.

Results

Case	Model	Optimal Service Rate				Minimum Cost
		Node 2	Node 3	Node 4	Node 5	
1	1	3.456572	3.368892	4.438494	4.243135	2118.308
	2	3.051215	3.411430	4.395514	4.286566	-1461.828
2	1	2.591291	2.546569	3.272574	3.28697	3328.507
	2	2.558162	2.514355	3.235014	3.182167	-251.316

For example 1, algorithm finds optimal solutions at one iteration and the computing time is less than 5 seconds for all problems.

For example 2, the algorithm finds solutions which give the total cost within 0.01% error bound in 6-7 iterations and the computing time is less than 15 seconds for all problems.

For both examples, the number of MVA visited at each iteration is around 30-40 with 10^{-16} error allowance in the search method which used the golden ratio.

7. Conclusion

Algorithms for the optimization of capacity planning in a single-chain closed network. Algorithms are based on the fact that the throughput is an increasing concave function with respect to service rates and the cycle time is an increasing convex function with repect to service times, and the proofs of these have been provided. Efficiency of these methods are based on the fact that the derivatives are easily obtained as a byproduct of the MVA algorithm.

The main application that motivated this work is the optimization of capacity in point of view of cost, that is, to maximize the profit (or minmize cost) by optimally designing a system.

Further study might include the number of customers in a system as a decision variable. Since the number of customers is not a continuous variable, the approach should be modified to accommodate this fact. And to improve efficiency of algorithms, it is necessary to study on the better search technique and error allowance(ε).

8. References

[1] Dallery, Y., and Frein, Y., *Flexible Manufacturing System*: An efficient Method ot Determine the Optimal Configuration of a Flexible Manufacturing System. Elsvier, Amsterdam, 1986.

[2] Kant, K., *Introduction to Computer System Performance Evaluation*, 1988, To be published.

[3] Kobayashi, H., and Gerla, M., Optimal Routing in Closed Queueing Networks. *ACM Transactions on Computer Science*, Vol 1, No 4, 294-310, 1983.

[4] Shanthikumar, G., and Yao, D. D., Stochastic Monotonicity of the Queue Lengths in Closed Queueing Networks. *Operations Research*, Vol 35, No 4, 583-588, 1987.

[5] _____, Optimal Server Allocation in a System of Multi-Server Stations, 1986.

[6] _____, Optimal Allocation of Buffers in a System of Manufacturing Cells, 1986.

[7] _____,On Server Allocation in Multiple Center Manufacturing Systems, 1986.

[8] Whitely, J. N., Some Inequalities Concerning Symmetric Forms, *Mathematica*, Vol 5, No 9, 49-57, 1958.