Journal of the Nigerian Association of Mathematical Physics
Volume 10 (November 2006), **371 - 378**
© J of NAMP

Control approach to Queue Theory

J. O. Omolehin
Department of Mathematics,, University of Ilorin, Ilorin, Nigeria

.

Abstract

The rigid condition for simple queue problem is minimized by considering multiple channels through control approach. The result control problem is solved using Conventional Conjugate Gradient Method and the optimal system performance is obtained

Keywords: Queue; Gradient; Parameter; Minimize; Functional
C.R. Categories G.1.7

1.0 Introduction

For proper understanding of this work it is necessary to give a brief background to this work. Though, a general fundamental principle of queue theory was given in our earlier work in [12].
Recall, queues are common phenomenon or common day experience in which people or items in need of service are arranged in order of their arrival, which usually arises when the demand for service exceeds the capacity of the service facility.

Queuing is a major functions problem facing not only management of corporate organization. It also affects the socio-economic structure of any society. Therefore, it becomes imperative to find an efficient and precise way of solving queue problems. Many people have worked on queuing system with different approaches and outcomes. Such people like Murdoch J. [8] Cohen, S.S. [9] and Tah, H.A. [10] had worked on the development of queuing theory. As a common experience people queue at least once a day e.g.
a. People waiting to buy petrol.
b. People waiting in shops/banks/post office, for service at service point.
c. people waiting to board a bus etc.

All these problems can be addressed by queuing theory methods which is possible because of certain characteristics about the patterns of arrival and service of customers. These regularities can be described statistically and an analytical result (formulae) sometimes follows. In principle queuing theory can predict how a particular system may operate.

2.0 Measurement parameters

Certain important measures of performance –system parameters can be obtained, for example:
(a) Average waiting time;
(b) Expected length of queue;
(c) Average number of customers in the system;
(d) Probability of experiencing delay;
(e) Average service processing time.

The three main approaches of obtaining these parameters are:
* Analytical method.
* Real world experimentation
* Simulation.

In this work, analytical method will be employed in obtaining optimal system performance.

Queuing systems are broadly characterized by the differing nature of five factors;
(i). **Arrival pattern of customers:**

The order of arrival may be deterministic (e.g. items on a production flow line) or more usually random. Customers may arrive one at a time or even en masse: which may not depend on the state of the system (e.g. a customer may “balk”. If he sees a large queue). The arrival rate many be constant or vary over time.

(ii). **Queue discipline**

The simplest arrangement if FIFO (First-In-First-Out). There is also LIFO (Last-In-First-Out) as in the items drawn from stock or redundancies in a work force or random (As with a telephone caller trying to get a connection). There may be several queues. Here customers may change from one queue to another (jockey). Some customers may join a queue and leave before service (renege).

(iii). Service Mechanism

There may be one or more servers. This may differ in speed of service. The speed of service at any point may be constant or random may vary with the time of the day. Servers may be in parallel (as a supermarket) or in a serial (As in a self-service cafeteria).

(iv). **Capacitation**

Some system has a limit of the number of people it can contain at any time t. If there exist such a situation, it is said that the system is capacitated.

(v). **Population**:

The population or “The calling source” flow from which the customers arrive may be finite or infinite.

There are basically 2 types of queue i.e.

(i) Simple queue

(ii) Multiple service channels

We are going to consider simple queue in this work. Multiple service channels will be considered in my next work.

3.0 Simple queue

A simple queue is a single file approaching one service point. That is there is only one line and one server.

However, for a queue to be described simple, twelve conditions must be satisfied, i.e.

(i) Random pattern of arrivals.

(ii) Discrete customers (i.e. not continuous flow).

(iii) Large population (effectively infinite)

(iv) Random service time.

(v) A single queue.

(vi) No queue capacity limit.

(vii) FIFO queue discipline must apply.

(viii) No reneging (joining a queue and leaving before service).

(ix) No balking (decide not to join the queue, having seen it to be large

(x) One service point.

(xi) One person at a time service.

(xii) Mean rate of service ids greater than the number mean rate of arrival.

4.0 System performance parameters

The most commonly used system performance parameter are [9, 10]:

(i) Traffic intensity $p = \frac{\lambda}{\mu}$

(ii) Probability of the system containing n customer $p_n = p^n(1-p)$

(iii) Probability that there are at least n customers in the system $p^{\lambda} = \left(\lambda/\mu\right)^n$

(iv) Average number of customers $= \frac{\lambda}{(\mu-\lambda)}$

(v) Average length of queue (excluding zero queues) $= \frac{\mu}{(\mu-\lambda)}$

(vi) Average system processing time ASPT) $=\dfrac{1}{(\mu-\lambda)}$

(vii) Average length of queue (number of items in the queue) $=\dfrac{\lambda^2}{\mu(\mu-\lambda)}$

(viii) Average queuing time $=\dfrac{\lambda}{(\mu(\mu-\lambda)}$

(ix) Average number of customers been served $=\dfrac{\lambda}{\mu}$

5.0 Literature review on conjugate gradient method

There are various types of algorithm for solving optimal control problems though there is no definite approach about which method to use in solving a particular class of problems. In choosing an implementable algorithm for solving a specific optimization problem the researcher attempts to evaluate the following aspect of performance. The first aspect is that of numerical accuracy, or sensitivity to errors. The second aspect is that of total time algorithm required to solve the specific problem at hand. The total time taken for the computation as we know depends largely on three factors; the specific computer been used not withstanding. The factors are [1] the rate of convergence of the method, the time taken per iteration and the competence of the programmer who writes the computer program.

Some of the well known descent methods include steepest descent method and conventional space algorithm (CFS). Though descent method is very popular but it has been confirmed that the method is not generally satisfactory [1]. The conventional function space algorithm (CFS) due to Di pillo et.al [1] that employs in approximate sense, the use of line search techniques has many problems, the penalty incurred in the attempt to facilitate the convergence rate of the algorithm (CFS) through many stringent conditions imposed on its minimization procedure is the enormous variety of cumbersome numerical calculations that is involved at each minimization step. This factor over the years has placed difficulty on the acceptability of this ingenuous computation method for handling penalized functional of optimal control problem. To circumvent this problem there came to existence many computational algorithms some of these are the conventional conjugate gradient method (CGM) and the extended conjugate method (ECGM) algorithm due to Ibiejugba.

6.0 Conjugate gradient method (CGM)

The CGM algorithm was originally developed by Hestenes and Stefel [4] to minimize quadratic functional. We will now give the description of the CGM algorithm now.

We begin by considering descent with Functional F on a Hilbert space H in which F is a Taylor series expansion truncated after the second order terms namely

$$F(x)=F_o+<a,x>_H+1/2<x,Ax>_H$$

where the scalar product is the scalar of H. We now consider what is termed conjugate descent with F conjugate descent it is assumed that a sequence

$$\{p_i\}\approx\lfloor p_o p_i\ldots,p_j\ldots,\rfloor$$

is available with the members of the sequence conjugate with respect to the positive definite linear matrix operator a.

By conjugate with respect to A is meant

$$<p_i,Ap_i>=0 \text{ if } i\neq j$$

In the case here A is assumed positive definite so

$$<p_i,Ap_i>>0$$

The CGM involves the following steps

Minimize $\quad F(x)=F_o(x)+<a,x>_H+1/2<x,Ax>_H$.

where A is nxn symmetric positive definite constant matrix operator on the Hilbert space H a is a vector in H and F_o is a constant term. Also $F(x)$ is a quadratic function.

The following steps are involved if $(H\equiv R)$

Step 1

The first element $X_o \in H$ of the descent sequence is guessed while the remaining members of the sequence are computed with the aid of the following formulae:

Step 2

$p_o = g_o - (a, Ax_o)$

(p_o is the descent direction and g_o is the gradient of $F(x)$ when $x = x_o$

Step 3

$$x_{i-1} = x_i + a_i p_i : a_i = < g_{i.} g_i >_H / < p_{i.} Ap_i >_H$$
$$p_{i-1} = -g_{i-1} + B_i P_i : B_i = < g_{i-1}, g_{i-i} > / < g_i, g_i >_H$$

Step 4

If $g_i = 0$ for some i terminate the sequence, else set $i = i + 1$ and go to step 3.

This CGM algorithm is very simple and precise. It has a well worked out convergence rate. It has been proved that it has a quadratic rate of convergence. That is for any problem in R^n the method converges in at most n iteration.

The Extended Conjugate Gradient Method (ECGM) algorithm due to Ibiejugba is a modified version of the conventional conjugate Gradient Method (CGM). That is the ECGM is based on the formalism of CGM algorithm. Recall CGM can only solve class of problems of quadratic functional in the form:

$$F(x) = F_o + < a, x >_H + 1/2 < x, Ax >_H$$

Where A is a constant matrix operator. But the ECGM can solve the class of problems of quadratic functional with constant matrix operator A and it can also solve the class of continuous optimal cost functional of the form

Minimize $j(u.v) = \int_0^t [au^2(t) + bv^2(t)]dt$

Subject to $u'(t) = cu(t) + dv(t).a > 0.\ b > 0$

The point ECGM is the construction of a control operator. Though the construction of the control operator A is complicated but the joy is that it was calculated in the exact form. That is there is no approximation involved. The ECGM is a very powerful algorithm. For detail see Ref.[1]. The algorithm has been used to solve many problems. In [12], Omolehin used it to solve Diffusion, problem.

7.0 Main result

In this work simple queue is considered. Queue problem is an every day's event and the need to adopt a new approach to solving queue problems can not be over emphasized. The new approach must be simple, precise and effective. Based on the above assumption: the elegant Conjugate Gradient Method algorithm (CGM) is employed in this study. Firstly, we revisit CGM algorithm briefly. We define a quadratic functional as:

$$F(x) = F_o + < a, x >_H + 1/2 < x, Ax >_H$$

where A is an nxn symmetric positive definite Matrix operator on the Hilbert space H, a is a vector in H.

If $H \equiv R^n$ the CGM algorithm is described as follows:

Step 1

Te first element $x_o \in H$ of the sequence is guessed will the remaining members of the sequence are computed with the aid of the formulae.

Step 2

$p_o = -g_o = -(a + Ax_o)$. p_o is the descent direction, g_o is the gradient of $F(x)$ when $x = x_0$.

Step3:

$$x_{i-1} = x_i + a_i p_i, a_i = < g_i.g_i >_H / < p_i.Ap_i >_H$$

$g_{i-1} = g_i + a_i Ap_i$, a is the step length

$$p_{i-1} = -g_{i-1} + \beta_i p_i, \beta_i = < g_{i-1}, g_{i-1} >_H \ / < g_i, g_i >_H$$

Step 4

If $g_i = 0$, for some i, terminate the sequence, else set $i = i+1$ and go to step 3.

Recall the CGM algorithm has a well defined convergence profile which has been extensively considered by Ibiejugba [1].

We now formulate our control problem in this form:

Minimize $F(x) = F_o + < a.x >_H + 1/2 < x.Ax >_H$ where $F(x)$ is the cost, F_o is the fixed cost. $x = (x_1, x_2, x_3)^T$ is the minimizing vector and our matrix A is such that the entries are represented by the system parameter of simple queue system i.e.

$$A = \begin{bmatrix} \dfrac{\lambda^2}{\mu(\mu-\lambda)} & \dfrac{\lambda}{\mu} & \dfrac{\lambda}{(\mu-\lambda)} \\ \dfrac{\lambda}{\mu} & \dfrac{\lambda}{\mu(\mu-\lambda)} & \dfrac{\lambda}{c\mu} \\ \dfrac{\lambda}{(\mu-\lambda)} & \dfrac{\lambda}{c\mu} & \dfrac{1}{(\mu-\lambda)} \end{bmatrix}$$

were $\lambda^2/\mu(\mu-\lambda)$ is the average length of queue λ/μ is the traffic intensity.

$\lambda/(\mu-\lambda)$ is the average number of customers, $\lambda/(c\mu)$ is the average number of customers being served and $1/(\mu-\lambda)$ is the average system processing time.

The following parameters are to be noted also:

μ = mean rate of service

λ = mean rate of arrival

c = service point.

A particular term p is now introduced to the system. T represents the uncertainty with the system

$$A = \begin{bmatrix} p\lambda^2 & \dfrac{\lambda}{\mu} & \dfrac{\lambda}{(\mu-\lambda)} \\ \dfrac{\lambda}{\mu} & \dfrac{\lambda}{\mu(\mu-\lambda)} & \dfrac{\lambda}{c\mu} \\ \dfrac{\lambda}{(\mu-\lambda)} & \dfrac{\lambda}{c\mu} & \dfrac{1}{(\mu-\lambda)} \end{bmatrix}$$

Our matrix A now becomes

For detail see my earlier work [12].

Recall system parameter used in queue theory as stated in the previous section.

Where λ is the arrival rate (i.e. is the average number of customer entering the system per unit time). μ is the service rate (i.e. the average number of customers been served) and c is the number of service channels.

Our objective function is to minimize cost subject to the condition of the system parameters for queuing system. This can be modeled in the form of unconstrained objective quadratic function by incorporating the system parameters into the entries of our control matrix operator. That is we use the system parameter for queuing system to form the entries of our control matrix operator A.

Denote our new parameter by $e, P_o.P_w W_s.L_s$

Our matrix becomes

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} L_g & P_w & L_q \\ P_w & W_s & W_q \\ L_q & W_q & E \end{bmatrix}$$

where

$$A_{11} = e(ec)^c p_o / \{c!(1-e)^2\} + ec$$
$$A_{12} = (ec)^c p_o / c!(1-e)$$
$$A_{21} = A_{12}$$
$$A_{22} = [(ec)^c p_o / \{c!(1-e)^2 c\mu\}] + \{1/\mu\}$$
$$A_{23} = (ec)^c p_o / c!(1-e)^2 c\mu$$
$$A_{31} = A_{13}$$
$$A_{32} = A_{23}$$
$$A_{33} = \lambda/\{c\mu\}$$

8. 0 Numerical Results

What follows is the numerical investigation of our new method. Towards this end we are going to use three permutations of A to perform our matrix experiment.
The three permutations are as follows

$$A = \begin{bmatrix} L_g & P_w & L_q \\ P_w & W_s & W_q \\ L_q & W_q & e \end{bmatrix}$$

when the queue pattern is perfect

$$B = \begin{bmatrix} e & W_q & L_q \\ W_q & L_s & P_w \\ L_q & P_w & W_s \end{bmatrix}$$

when the queue pattern is deterministic

$$C = \begin{bmatrix} W_s & P_w & L_q \\ P_w & e & W_q \\ L_q & W_q & L_s \end{bmatrix},$$

when the queue pattern is random

In all the permutations the requirement for symmetric property of control matrix operator A, B, C is satisfied.

9.0 Table for the numerical results

Table 1: Matrix A: Perfect Pattern

Cases	1	2	3	4	5
	$\lambda = 6$	$\lambda = 35$	$\lambda = 32$	$\lambda = 36$	$\lambda = 3$
	$\mu = 15$	$\mu = 12$	$\mu = 13$	$\mu = 13$	$\mu = 13$
	$NQ = 8$	$NQ = 6$	$NQ = 7$	$\mu NQ = 10$	$NQ = 10$
Input variables	$c = 3$	$\lambda = 32$	$c = 5$	$c = 6$	$c = 7$
Iterations			Gradients		

0	1.15395800	6.590370	0.07504624	0.252389300	0.0280783
1	0.6324594	0.1490880	1.0472870E-002	0.0745777	4.7831090E-006
2	7.4683400E-004	2.03850990E-004	1.2039170E-008	0.061367600	1.9223240E-010
3	3.32150E-018	6.6894130E-014	2.44491290E-016	1.4416190E-012	1.2638370E-016

Table 2] Matrix B: Deterministic Pattern

Cases	**1**	**2**	**3**	**4**	**5**
Input variables	$\lambda = 3$ $\mu = 13$ $NQ = 3$ $c = 3$	$\lambda = 36$ $\mu = 10$ $NQ = 12$ $\lambda = 3$	$\lambda = 30$ $\mu = 6$ $NQ = 2$ $c = 3$	$\lambda = 10$ $\mu = 8$ $NQ = 3$ $c = 3$	$\lambda = 30$ $\mu = 18$ $NQ = 5$ $c = 4$
Iterations			Gradients		
0	0.1544008	11.5074200	2.9364350	1.1931870	1.0991380
1	1.220595E-004	1.0780350	0.9612817	0.0301381	0.0097441
2	4.8857220E-010	6.0151800E-012	5.7734950E-012	1.695660E-013	1.3754550E-013
3	1.3449670E-014	8..9719080E-015	8.971908E-015	1.563478E-016	1.2315673E-016

Table 3 Matrix C: Random Pattern

Cases	**1**	**2**	**3**	**4**	**5**
Input variables	$\lambda = 4$ $\mu = 10$ $NQ = 8$ $c = 3$	$\lambda = 4$ $\mu = 8$ $NQ = 8$ $c = 3$	$\lambda = 3$ $\mu = 13$ $NQ = 7$ $c = 5$	$\lambda = 36$ $\mu = 4$ $NQ = 6$ $c = 3$	$\lambda = 2$ $\mu = 5$ $NQ = 6$ $c = 3$
Iterations			Gradients		
0	0.1538738	0.2071431	0.2087523	0.2121790	0.1558337
1	0.0944581	0.1363207	0.1354510	0.1337684	0.0949267
2	0.011294	0.0019609	0.0014544	0.0007132	4.3970430E-004
3	7.1302460E-006	4.488010E-006	3.0691240E-006	1.6087750E-006	2.940619E-007

10.0 Analysis of numerical result

Normally, the optimal value of an optimization problem is found when the gradient is zero or when the tolerance of the gradient is satisfied. A close look at the Tables 1 - 3 show that the optimal vales are attained at the third iteration. In Table1, the optimal value will be obtained when $\lambda = 6$, $\mu = 15$, $NQ = 8$, $c = 3$.This can be found at the first column of the Table. Table [2] shows that the optimal result is obtained when $\lambda = 30$, $\mu = 18$ $NQ = 5$, $c = 4$. The result for the random pattern is obtained at the last column of the table. These results are interesting and more research will be carried out.

11.0 Conclusion

This work addressed the usual calculation complication associated with queue problems. The focal point is to put the system performance parameters in matrix form to the extent that our arrangement should satisfy the positive definiteness of matrix operator in the Conventional Conjugate Gradient Method algorithm. This method is simpler than the usual long calculations involved in the existing method. See [8,10]. All need be done is just to vary your parameter and determine the optimal set.

References

[1] Ibiejugba. M.A. Computing Method in Optimal Control. University of Leeds. Leeds England. PhD Thesis. 1980.

[2] Omolehin. J.O. Numerical Experiment with the Extended Conjugate Gradient Method Algorithm. University of Ilorin. Nigeria. Msc. Thesis 1985.

[3]. Glad. T. Constrained Optimization using multiplier methods with Applications to Control Problems. University of Lund. Sweden. Report No. 7603. 1976.

[4] Hestenes, M. and Steifel, E.: Method of Conjugate Gradient for solving linear system, J. Res. Nat. Bus. Standards, Vol.49, pp409-436, 1952.

[5] Ibiejugba. M.A. On Ritz-penalty for solving the Control of a Diffusion Equation. Journal of Optimization Theory and Application. Vol. 39. pp 431-449. 1983.

[6]. Hill. J.M On the solution of Reaction Diffusion Equation. IMA Journal of Applied Mathematics (1981) 27, 177-194.

[7]. Ibiejugba. M.A and Onumanyi. P. On a Control Operator and some of its Applications Journal of Mathematical Analysis and Applications. Vol.103. Pp31-47. 1981.

[8] Murdoch, J., 1978, Queuing Theory (Macmillan)

[9]. Cohen, S.S, 1995, Operational Research (Edward Arnold)

[10]. Tah, H.A, 1987, Operational Research (4th ed) (collan Macmillan)

[11]. Cook, T.M and Russell, D. 1981, Introduction to Management Science (2nd ed) (Prentice Hall).

[12]. J.O.Omolehin, K.Rauf and D.J.Evans, 2005, Conjugate gradient method approach to queue theory, Int.Jour. Of Comp. Maths., Vol.82, No.7, 829-835.