Operations Research Letters 44 (2016) 313–318

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

Fast Engset computation

P. Azimzadeh *, T. Carpenter

David R. Cheriton School of Computer Science, University of Waterloo, Waterloo ON, Canada N2L 3G1

ARTICLE INFO

Article history:
Received 20 April 2015
Received in revised form
23 February 2016
Accepted 23 February 2016
Available online 2 March 2016

Keywords:
Engset formula
Teletraffic
Ordinary hypergeometric function
Turán-type inequality

ABSTRACT

The blocking probability of a finite-source bufferless queue is a fixed point of the Engset formula, for which we prove existence and uniqueness. Numerically, the literature suggests a fixed point iteration. We show that such an iteration can fail to converge and is dominated by a simple Newton's method, for which we prove a global convergence result. The analysis yields a new Turán-type inequality involving hypergeometric functions, which is of independent interest.

1. Introduction

The Engset formula is used to determine the blocking probability in a bufferless queueing system with a finite population of sources. Applications to *bufferless optical networks* [6,20,12,14,13] have sparked a renewed interest in the Engset model and its generalizations [5]. Sztrik provides a literature review of applications [18], including *multiprocessor performance modeling* and the *machine interference problem*, in which machines request service from one or more repairmen. The analysis herein was inspired by a recent application in *sizing vehicle pools* for car-shares [4].

The queue under consideration is the $M/M/m/m/N$ queue [10]. This is a bufferless queue with N *sources* that can request service, provided by one of m identical *servers*. When all m servers are in use, incoming arrivals are *blocked* and leave the system without queueing. The *Engset formula* is used to determine the probability P that any random arrival is blocked. The Engset formula is [11, Equation (62)]

$$P=\underbrace{\lim_{P'\to P}\frac{\binom{N-1}{m}\left(M(P')\right)^m}{\sum_{X=0}^{m}\binom{N-1}{X}\left(M(P')\right)^X}}_{f(P)}\quad\text{where } M(P)=\frac{\alpha}{1-\alpha\,(1-P)} \qquad \text{(Engset formula)}$$

The number of sources N, the number of servers m, and the offered traffic *per-source* α are given as input. It is worthwhile to note that subject to some technical assumptions, the Engset formula remains valid under general distributions (i.e. $G/G/m/m/N$) [19, Section 5.4].

It is not obvious if any value of P satisfies the Engset formula, or if multiple values of P might satisfy it. To the authors' best knowledge, this work is the first to establish the existence and uniqueness of a solution (Section 2).

Remark. The limit appearing in the Engset formula is a technical detail to avoid (for ease of analysis) the removable discontinuity at $P = 1 - 1/\alpha$. We mention that f may admit nonremovable discontinuities at some negative values of P (at which the limit does not exist), though this does not affect the analysis.

Remark. Let λ be the *idle source initiation rate*, the rate at which a free source (i.e. one not being serviced) initiates requests, and $1/\mu$ be the *mean service time*. If P is the blocking probability, $M(P) = \lambda/\mu$. This substitution removes P from the right-hand side of the Engset formula [11, Equation (70)]. However, λ is often unknown in practice, and hence this method is only applicable in special cases, or subject to error produced from approximating λ.

2. Properties of the Engset formula

If the number of servers m is zero, any request entering the queue is blocked ($P = 1$). If there are at least as many servers as there are sources ($m \geq N$), any request entering the queue can

* Corresponding author.
E-mail addresses: pazimzad@uwaterloo.ca (P. Azimzadeh), tcarpent@uwaterloo.ca (T. Carpenter).

http://dx.doi.org/10.1016/j.orl.2016.02.011
0167-6377/

immediately be serviced ($P = 0$). Finally, the case of zero traffic ($\alpha = 0$) corresponds to a queue that receives no requests. We assume the following for the remainder of this work:

Assumption. m and N are integers with $0 < m < N$. α is a positive real number.

The following lemmas characterize f defined in the Engset formula and are used to establish several results throughout this work:

Lemma 1. *f is strictly decreasing on* $[0, \infty)$.

Lemma 2. *f is convex on* $[1 - 1/\alpha, \infty) \supset [1, \infty)$.

Owing partly to Lemma 1, our first significant result is as follows:

Theorem 3. *There exists a unique probability $P^\star$ satisfying the Engset formula.*

Proofs of these results are given in the Appendix. The proof of Theorem 3 establishes that $f(0) - 0$ and $f(1) - 1$ have opposite signs. Therefore, $P^\star$ can be computed via the bisection method on the interval $[0, 1]$ applied to the map

$$P \mapsto f(P) - P. \quad (1)$$

3. Computation

3.1. Fixed point iteration

The literature suggests the use of a fixed point iteration [9, page 489]. This involves picking an initial guess P_0 for the blocking probability and considering the iterates of f evaluated at P_0. Specifically,

$$P_0 \in [0, 1]$$
$$P_n = f(P_{n-1}) \quad \text{for } n > 0. \qquad \text{(fixed point iteration)}$$

We characterize convergence in the following result:

Theorem 4. *If $\alpha \leq 1$ and $|f'(0)| < 1$, the fixed point iteration converges to $P^\star$.*

While the first inequality appearing above is a restriction on the per-source traffic, the second inequality is hard to verify, as it involves the derivative of f. This inspires the following:

Corollary 5. *If $\alpha \leq 1$ and $N \geq 2m$, the fixed point iteration converges to $P^\star$.*

The condition $N \geq 2m$ requires there to be twice as many sources as there are servers, satisfied in most (but not all) reasonable queueing systems.

Proofs of these results are given in the Appendix.

3.2. Newton's method

Newton's method uses first-derivative information in an attempt to speed up convergence. In particular,

$$P_0 \in [0, 1]$$
$$P_n = P_{n-1} - \frac{f(P_{n-1}) - P_{n-1}}{f'(P_{n-1}) - 1} \quad \text{for } n > 0. \qquad \text{(Newton's method)}$$

Often, convergence results for applications of Newton's method are *local* in nature: they depend upon the choice of initial guess P_0. By using the convexity established in Lemma 2, we are able to derive a *global* result for Newton's method:

Theorem 6. *If $\alpha \leq 1$, Newton's method converges to $P^\star$.*

Table 1
Comparison under $N = 20$ and $\alpha = \frac{1}{4}$.

Servers	Probability	Number of iterations	
m	$P^\star$	Fixed point	Newton
1	8.322e−01	6	3
2	6.725e−01	7	3
3	5.235e−01	7	3
4	3.879e−01	8	3
5	2.693e−01	9	3
6	1.714e−01	8	4
7	9.718e−02	8	4
8	4.753e−02	7	4
9	1.947e−02	6	4
10	6.554e−03	5	3
11	1.798e−03	4	3
12	4.005e−04	4	3
13	7.194e−05	3	3
14	1.028e−05	3	3
15	1.142e−06	3	3
16	9.518e−08	3	2
17	5.599e−09	2	2
18	2.074e−10	2	2
19	3.638e−12	2	2

Table 2
Comparison under $N = 20$ and $\alpha = \frac{1}{2}$.

Servers	Probability	Number of iterations	
m	$P^\star$	Fixed point	Newton
1	9.087e−01	7	3
2	8.187e−01	8	3
3	7.303e−01	9	3
4	6.436e−01	10	3
5	5.591e−01	11	3
6	4.773e−01	11	3
7	3.985e−01	14	3
8	3.235e−01	15	4
9	2.531e−01	16	4
10	1.885e−01	16	4
11	1.310e−01	14	4
12	8.259e−02	12	4
13	4.527e−02	10	4
14	2.041e−02	8	4
15	7.124e−03	6	4
16	1.827e−03	5	4
17	3.254e−04	4	3
18	3.623e−05	3	3
19	1.907e−06	3	3

A proof of this result is given in the Appendix. Superficially, Theorem 6 seems preferable to Corollary 5 as it does not place restrictions on N or m. In practice, we will see that Newton's method outperforms the fixed point iteration, and that it performs well even when $\alpha > 1$ (Section 4).

4. Comparison of methods

Tables 1–4 compare the performance the fixed point iteration and Newton's method for a queueing system with $N = 20$ sources (though we mention that the observed trends seem to hold independent of our choice of N). The initial guess used is $P_0 = \frac{1}{2}$. The stopping criterion used is $|P_{n+1} - P_n| \leq tol = 2^{-24}$.

Bisection halves the search interval at each step, so that the maximum possible error at the nth iteration is 2^{-n}. It follows that to achieve a desired error tolerance tol, bisection requires $\lceil -\lg(tol) \rceil = \lceil -\lg(2^{-24}) \rceil = 24$ iterations independent of the input parameters (for this reason, it is omitted from the tables). The fixed point iteration fails to converge or performs poorly (sometimes taking hundreds of iterations) precisely when the sufficient conditions of Corollary 5 are violated. Newton's method outperforms both algorithms by a wide margin, often converging in just a few iterations.