ABSTRACT

A method to obtain numerical results directly by solving simplified equations of state, which utilise the uniform structure of subgroups in grading groups or link systems, is presented. It is shown that this method becomes a good practical approximation for a simple loss system grading group of both sequential and random hunting and also for a delay system. Computation time and programming are ascertained to be fairly reduced in comparison with simulations by use of the usual linear equation system solution program and iterations.

1. INTRODUCTION

Equations of state are well known as one of the most useful means for obtaining exact solution in traffic theory. However, when they are applied to practical systems other than full availability, e.g. grading groups or link systems, the equations could not be solved numerically because of the enormous number of states, even for the simplest cases, such as Poisson input and exponential holding time.

Those systems are generally composed of a number of subgroups and, in many cases, those subgroups are symmetrical in structure and uniformly loaded. The simplified equations of state method is devised to apply to those cases, utilizing the above-mentioned characteristics.

In this paper, this method is described by an application to a simple grading group. A great many works have already been published pertaining to solving the problem of grading groups (1). In comparison with those works, intentions of this method are as follows:

(1) To obtain numerical results directly by a digital computer, rather than analytical expression or manipulation.

(2) General application to both sequential and random hunting, also to delay systems, not confining to such grading patterns as ideal gradings, but suitable for simple progressive gradings.

(3) Less computation time and programming than simulations, intending direct calculation of traffic capacity tables.

2. PRINCIPLE AND CHARACTERISTICS

This method can be characterized by the following items.

(1) In usual equations of state, the state probabilities are obtained, taking into account the whole system concerned. However, in such a case with symmetrical and uniform subgroups, it is sufficient to evaluate congestion functions for an arbitrary subgroup. Therefore, the state probabilities may also be obtained by merely considering the state of this subgroup. Consequently, the variables (number of states) are remarkably reduced and the numerical results can easily be computed.

(2) In order to express transition probabilities (birth or death coefficients) for equations of state exactly, the states corresponding to the parts out of consideration may generally be required. However, if the above uniformity (symmetry) is utilised, they can be expressed by the state probabilities (or the information obtained from those probabilities) of the part under consideration. As a result, some approximations may usually be introduced.

(3) The equations of state must be solved simultaneously with the above relation introduced in the above process (2). Therefore, they do not become a linear equation system in general (even for stationary solutions which are exclusively concerned in this paper). Consequently, some
It must be remarked that the individual ideas mentioned in (1) and (3) have already been utilized (2), (3). Therefore, the feature of this method may be the idea in (2) combined with the following numerical solution method.

3. APPLICATION TO LOSS SYSTEM GRADING GROUP

3.1 TRAFFIC MODEL

The model treated in this paper is a simple grading group, as shown in Fig. 1, which may be sufficient to illustrate the calculation principle. (This model really occurred in coastal radio telephone service, where traffic capacity for random hunting and the difference between sequential and random hunting became in question.) The consideration can be confined to the inside of the frame shown in Fig. 1, in order to obtain state probabilities.

Let \( i,j \) be the state probability that \( i \) individual trunks are occupied among \( n \) and \( j \) common trunks among \( m \). Then, probability of loss is expressed by \( (n,m) \) for the subgroup under consideration, provided that Poisson input is assumed. Each state probability \( (i,j) \) and also \( (n,m) \) are the same for every subgroup because of the uniformity of model.

3.2 STATE TRANSITIONS FOR SEQUENTIAL HUNTING

If exponential holding time is assumed in addition to Poisson input, and if mean holding time is taken as a unit of time, the transition rates (birth or death coefficients) between states become as shown in Fig. 2. It should be noticed that unknown \( B_j \) is introduced, where \( B_j \) is the conditional probability that all \( n \) individual trunks are busy when \( j \) common trunks are occupied among \( m \).

In sequential hunting, calls are offered to the common trunks only from those subgroups wherein the individual trunks are all occupied. Therefore, the exact birth coefficient corresponding to transition \((j \rightarrow j+1)\) is expressed by

\[
\sum_{u=1}^{k} \delta(r_u=n) \frac{a}{(m-j)} \text{ if } r_u=n, \quad \delta(r_u=n) = \begin{cases} 1, & \text{if } r_u=n \\ 0, & \text{if } r_u \neq n, \end{cases}
\]

\[
\sum_{u=1}^{k} \delta(r_u=n) \frac{a}{(m-j)} \text{ if } r_u=n, \quad \delta(r_u=n) = \begin{cases} 1, & \text{if } r_u=n \\ 0, & \text{if } r_u \neq n, \end{cases}
\]

if the number of occupied individual trunks for each subgroup \( r_u \ (u=1, 2, \ldots, k) \) is given. Then, the equations of state become exact.

Therefore, birth coefficients in Fig. 2 are an approximation, where \( \delta(i=n) \) is exactly expressed for the subgroup under consideration (because \( r_u=n \) is given) and, for the other subgroups, the mean value of \( \delta(r_u=n) \) is approximately substituted for on condition that \( j \) common trunks are occupied (i.e., \( B_j \)).

It will be evident from the above-mentioned uniformity that \( B_j \) is the same for every subgroup and that \( B_j \) can be expressed by the state probability \( (i,j) \) of the subgroup under consideration, as follows.

\[
B_j = \frac{\delta(i=n)}{\sum_{i=0}^{n} (i,j)}
\]

The solution is obtained by solving the equations of state determined from the coefficients in Fig. 2 and Eq. (2) simultaneously. Details of the equations are omitted, because they are not necessary to be written down in the following numerical solution method.

3.3 STATE TRANSITIONS FOR RANDOM HUNTING

The transition rates (birth or death coefficients) between states are as shown in Fig. 3, if the same traffic model and assumptions are used as before. \( R_j \) in Fig. 3 represents mean occupied individual trunks among \( n \) on condition that \( j \) common trunks are occupied among \( m \).

In random hunting, an arbitrary free trunk can be chosen and occupied with equal probability among all free trunks (regardless of individual or common) in a subgroup. Therefore, exact birth coefficient corresponding to transition \((j \rightarrow j+1)\) is expressed by

\[
\sum_{u=1}^{k} \frac{a}{(n-r_u+j)} \text{ if } n-r_u+j > 0,
\]

if the number of occupied individual trunks for each
subgroup \( R_i \) \((u=1, 2, \ldots, k)\) is given. Then, the equations of state become exact.

Birth coefficients in Fig. 3 are also an approximation, because \( R_i \) is expressed for merely the subgroup under consideration, while, for the other subgroups, the mean value of \( R_i \) is substituted for on condition that \( j \) common trunks are occupied \((i.e., R_j)\).

It is also evident as before that \( R_j \) is identical for every subgroup, and that \( R_j \) can be expressed by the state probability \((i,j)\) of the subgroup under consideration as follows.

\[
R_j = \sum_{i=0}^{n} \frac{n}{i} [1,1]^i \prod_{i=0}^{n} i
\]

(4)

### 3.4 SIMPLIFICATION EFFECT

The main purpose of simplified equations of state is to reduce the number of states (variables) and to make the calculation feasible for cases wherein calculation is impossible by the exact method. The total number of states becomes as follows for the model in Fig. 1.

1. Exact equations of state (general case with unequal subgroup traffic; where different probabilities of loss can be obtained for each subgroup)

\[
(n+1)^k (m+1)
\]

(5)

2. Exact equations of state (special case with equal subgroup traffic; where the number of states is reduced, taking into account the fact that probabilities of states obtained by any interchange of subgroups become equal)

\[
m+1 \times k (m+1) = \binom{n+k}{k} (m+1)
\]

(6)

3. Simplified equations of state

\[
(m+1)(m+1)
\]

(7)

Examples of their comparison are shown in Table 1. The effects of the simplification are evident. Consequently, it is confirmed that a linear equation system solution program in hand, which can handle about 200 unknowns without special modification, is sufficient even for the largest case required.

### Table 1 Simplification Effect

<table>
<thead>
<tr>
<th>Condition</th>
<th>Exact Equations of State</th>
<th>Simplified Equations of State</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>( m )</td>
<td>( k )</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>18</td>
<td>4</td>
<td>12</td>
</tr>
</tbody>
</table>

(Largest Case) 95

Notations are the same as in Fig. 1.

### 4. NUMERICAL SOLUTION METHOD

#### 4.1 ITERATION (CORRESPONDING TO SIMPLIFICATION)

It is seen from Sections 3.2 and 3.3 that, as a result of simplification, functions of state probabilities to be obtained are introduced in coefficients of equations of state. Therefore, the equation system do not become linear and, consequently, it is necessary to consider the calculation procedure (programming) according to each problem, in order to solve the equations directly.

On the other hand, equations of state are usually a simultaneous linear equation system with coefficients of definite regularity, and a general solution method utilising this characteristic is prepared. If this solution method is used, the procedure can be cut down because of its generality.

Therefore, an iteration method is employed, which assumes appropriate initial values for the unknowns that should be expressed by the relation of Eq.(2) or Eq.(4). Then, the simultaneous linear equation system is solved by the above-mentioned general solution method. The unknowns are modified by the obtained state probabilities (1st order approximations), using Eq.(2) or Eq.(4). Thus, by iterating the above procedure, the higher order approximation of state probabilities can be obtained.

#### 4.2 INITIAL VALUES

As the initial values of such unknowns seem to influence computation time (convergence speed), considerations in advance must be required for each model.

For the sequential hunting model of Section 3.2, the following value of \( R_j \) is used, assuming independence of \( j \), because \( R_j \) is the probability that all \( n \) individual trunks are busy.

\[
R_j = P_n(a)
\]

(8)

Where, \( P_n(a) \) is probability of loss by Erlang's B-formula in usual notation.

For the random hunting model of Section 3.3, the following value of \( R_j \) is used, assuming independence of \( j \). Because \( R_j \) is mean occupied individual trunks per subgroup, the corresponding values are obtained for sequential hunting from the individual trunk side (denoted by \( R_j \)) and from the common trunk
side (denoted by $R_C$), respectively, by Berkeley's method. They are averaged with weighting by number of individual trunks $n$ and common trunks $m$, as an approximation of random hunting.

$$R_j = \left(\frac{n R_1 + m R_C}{n+m}\right)$$

$$R_T = a\{1 - E_n(a)\}$$

$$R_C = a_0 \left\{E_n(a_0) - E_{n+m}(a_0)\right\},$$

where

$$a_0 \ E_n(a_0) = a \ E_n(ka).$$

problem involve only about 50 steps in FORTRAN, which can be completed immediately.

5. COMPARISON WITH SIMULATION RESULTS

Simulations have also been carried out for the same models as numerical calculations. Examples of their comparison are shown in Figs. 5 and 6. Although 5th order approximations of numerical calculation are shown in the figures, the differences among more than 2nd order approximations can scarcely be recognised in these figures. Simulation values are obtained by offering 20,000 to 30,000 calls.

The problem in question was as mentioned in Section 3.1. It is found, from these comparisons, that the simplified

Table 2 Examples of Iteration for Loss System

<table>
<thead>
<tr>
<th>Order</th>
<th>Probability of Loss</th>
<th>Random Hunting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$B_j$=0.2466</td>
<td>Initial $R_0$=0.7749</td>
</tr>
<tr>
<td></td>
<td>$B_0$ $B_1$ $B_2$</td>
<td>$R_0$ $R_1$ $R_2$</td>
</tr>
<tr>
<td>1</td>
<td>0.09341899 0.1927 0.2377 0.3105</td>
<td>0.11108668 0.6270 0.7079 0.8302</td>
</tr>
<tr>
<td>2</td>
<td>0.08787209 0.1915 0.2441 0.3166</td>
<td>0.10927572 0.6232 0.7056 0.8297</td>
</tr>
<tr>
<td>3</td>
<td>0.08869554 0.1915 0.2442 0.3152</td>
<td>0.10927411 0.6232 0.7056 0.8297</td>
</tr>
<tr>
<td>4</td>
<td>0.08873779 0.1915 0.2442 0.3152</td>
<td>0.10927413 0.6232 0.7056 0.8297</td>
</tr>
<tr>
<td>5</td>
<td>0.08872033 0.1915 0.2442 0.3152</td>
<td>0.10927411 0.6232 0.7056 0.8297</td>
</tr>
</tbody>
</table>

Examples of iteration (convergence aspect) are shown in Table 2. From those results, the initial value of $R_j$ by Berkeley's method is found to be of fairly good accuracy. Also, 2nd order approximation may be enough for capacity tables, if it is taken into account that simplification is itself only an approximation (for saving of computation time).

4.3 COEFFICIENTS PREPARATION (CORRESPONDING TO EQUATIONS OF STATE CHARACTERISTICS)

Solution methods for a simultaneous linear equation system have already been established, when the coefficient matrix is given. In this paper, a successive elimination method was used because of the programming simplicity. Equations of state are a kind of simultaneous linear equation system with coefficients of definite rule. Therefore, necessary procedure (programming) for each problem is to prepare the values of coefficient matrix.

Moreover, those values are easily produced in a computer by a simple program, because the transition rates are very simple functions of state variables, e.g. $i$ and $j$ in Figs. 2 and 3, and the coefficient matrix is produced by the following plain rule.

(1) A serial number is assigned to every state (from 1 to $(n+1)(m+1)$ in Fig. 2 or Fig. 3).

(2) The coefficient in $y$-th row and $x$-th column is the transition rate in Fig. 2 or Fig. 3 from state $x$ to state $y$, if $x \neq y$.

(A majority of them equals zero.)

(3) The diagonal coefficient in $y$-th row and $y$-th column is the minus signed sum of transition rates from state $y$ to every neighboring state.

4.4 PROGRAM

The flow chart of the program is as shown in Fig. 4. Necessary program to be written for individual

ITC 7
The equations of state method shows a sufficiently good matching with simulation results for the purpose of calculating practical traffic capacity tables in every condition, and that the differences between random and sequential hunting can be estimated with sufficient accuracy.

Moreover, numerical calculations need only about $1/20$ the computation time of the simulations, even if the largest case in Table 1 is included. Consequently, the above-mentioned traffic capacity tables with a fixed probability of loss was directly calculated by the numerical solutions of the simplified equations of state, including the necessary interpolation processes.

6. APPLICATION TO DELAY SYSTEM GRADING

6.1 STATE TRANSITIONS FOR DELAY SYSTEM

The same grading group as before, i.e., as shown in Fig. 1, is also considered. Delayed calls, offered from a specific subgroup and encountered all $n+m$ accessible trunks busy, are assumed to wait in front of this subgroup. Then, the state probability can also be denoted by $(i,j)$, where $i$ means the number of calls in the subgroup, occupying the individual trunks or waiting, and $j$ is the same as loss system.

Transitions between states become as shown in Fig. 7, because waiting calls can exist only when all $n+m$ trunks are busy. Thus, the number of waiting calls is $i-n$. It should be remarked that the transitions in the range $0 \leq i \leq n$ (of course $0 \leq j \leq m$) is quite the same as the loss system. Therefore, any one of the transitions corresponding to sequential hunting (Fig. 2) or random hunting (Fig. 3) can be substituted for in these range according to the hunting rule. (Consequently, details are omitted in Fig. 7.)

$$\mu_i = n + m \left( \frac{k-1}{k} \right) L_m (i-n)$$

6.2 DEATH COEFFICIENT IN CONGESTION STATE

The transition rate $\mu_i$ from $i (> n)$ to $i-1$ in Fig. 7 is obtained by the following considerations, assuming random service for waiting calls.

If the number of calls in each subgroup $r_k$ (the same meaning as above $i$) is given, then the exact death coefficient $\mu_i$ (corresponding to $k$-th subgroup) is expressed by

$$\mu_i = n + m \frac{r_k - n}{k} \sum_{u=1}^{\max( r_k-n, 0 )}$$

A waiting call in the subgroup is surely served when a call occupying an individual trunk in this subgroup terminates its holding time, while it can be served when a call occupying a common trunk terminates, provided that one of the waiting calls in the subgroup
Table 3 An Example of Iteration for Delay System

<table>
<thead>
<tr>
<th>Order</th>
<th>M(O)</th>
<th>( \frac{W}{h} )</th>
<th>( L_m )</th>
<th>( B_0 )</th>
<th>( B_1 )</th>
<th>( B_2 )</th>
<th>( B_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.07921407</td>
<td>0.01764467</td>
<td>0.6727</td>
<td>0.0992</td>
<td>0.1397</td>
<td>0.1956</td>
<td>0.2733</td>
</tr>
<tr>
<td>2</td>
<td>0.08314043</td>
<td>0.02021673</td>
<td>0.7256</td>
<td>0.0985</td>
<td>0.1556</td>
<td>0.2150</td>
<td>0.2793</td>
</tr>
<tr>
<td>3</td>
<td>0.08673774</td>
<td>0.02133070</td>
<td>0.7294</td>
<td>0.0984</td>
<td>0.1553</td>
<td>0.2099</td>
<td>0.2719</td>
</tr>
<tr>
<td>4</td>
<td>0.08586773</td>
<td>0.02113330</td>
<td>0.7337</td>
<td>0.0984</td>
<td>0.1554</td>
<td>0.2102</td>
<td>0.2734</td>
</tr>
<tr>
<td>5</td>
<td>0.08605632</td>
<td>0.02198777</td>
<td>0.7339</td>
<td>0.0984</td>
<td>0.1554</td>
<td>0.2101</td>
<td>0.2733</td>
</tr>
</tbody>
</table>

\( n, m, k, a : \) The same as Fig. 1, \( B_j : \text{Eq.}(12), \quad L_m : \text{Eq.}(11), \quad M(O) : \text{Eq.}(14), \quad W/h : \text{Eq.}(15). \)

\( \text{(numerator of Eq.}(10)\text{)} \) is selected with equal probability among all waiting calls (denominator of Eq.\((10)\)).

When each \( r_u \) cannot be given excluding \( r_u = i \), it is reasonable to substitute its mean value, i.e., mean number of waiting calls per subgroup on condition that all common trunks are occupied \( (L_m) \), similarly to the preceding cases. Thus, the death coefficient \( \mu_i \) in Fig. 1 is obtained, which is also an approximation.

It will be evident as before that \( L_m \) is identical for every subgroup, and that it can be expressed by the state probabilities under consideration as follows.

\[
L_m = \sum_{i=n}^{\infty} \frac{[i,m]}{[i,m]} \quad \text{\((11)\)}
\]

### 6.3 NUMERICAL CALCULATION

It is easily seen from state transitions in Fig. 7 that state probability \( [i,m] \) can be expressed by

\[
[i,m] = \frac{a_{i-n}}{\mu_u} \quad \text{\((12)\)}
\]

Consequently, it is sufficient to solve a simultaneous linear equation system only for the range \( 0 \leq i \leq n \), \( 0 \leq j \leq m \) (i.e., equal in size to Section 3.4) by quite the same procedure as the loss system. Then, state probabilities \( [i,m] \) for \( i>n \) can be calculated by Eq.\((12)\).

Those state probabilities are not normalized. Therefore, every state probability obtained above must be multiplied with the following normalizing factor.

\[
\left\{ \sum_{i=n+1}^{\infty} [i,m] \right\}^{-1} \quad \text{\((13)\)}
\]

Probability of delay \( M(O) \) and mean waiting time \( W \) is obtained by those corrected probabilities, as follows.

\[
M(O) = \sum_{i=n}^{\infty} [i,m] \quad \text{\((14)\)}
\]

\[
W = h \sum_{i=n}^{\infty} (i-n) [i,m] \quad \text{\((15)\)}
\]

\( h : \text{Mean holding time} \)

If unlimited waiting facilities are concerned, it may practically be substituted for by sufficiently large limitation, neglecting those probabilities \( [i,m] \) for larger value of \( i \) (as in simulation).

It may be remarked that, in order to obtain higher order approximation, modification of the value \( L_m \) by Eq.\((11)\) is required together with \( B_j \) or \( R_j \) (of Sections 3.2 or 3.3) which is necessary to solve the linear equation system. Those \( B_j \) and \( R_j \) differ from the case of loss system when \( j=m \). However, those values for \( j=m \) are not available for transition rates in the equation system. Therefore, correction is not needed.

As an initial value of \( L_m \), the following expression is used for simplicity.

\[
L_m = \frac{E_n(a)}{n+m/k-a} \quad \text{\((16)\)}
\]

Therefore, convergence is somewhat slower than the loss system, as seen from an example in Table 3.

---

**Fig. 8** Comparison with simulation (probability of delay).
6.4 **COMPARISON WITH SIMULATION RESULTS**

The accuracy of this method has also been ascertained by comparing with a simulation result for sequential hunting random service delay system grading group, which was published by Gambe at the 3rd ITC (4). The results are shown in Figs. 8 and 9.

The simulation values correspond to the average for 20,000 calls of each condition, where confidence intervals were not obtained. Numerical calculation values correspond to 5th order approximation. It may be concluded from these figures that this method also shows a fairly good approximation for delay systems.

![Fig. 9 Comparison with simulation (mean waiting time).](image)

**REFERENCES**


7. **CONCLUSION**

A method to obtain numerical results directly from simplified equations of state applicable to grading or link systems has been described and proved to show a good practical approximation for the simple grading groups (both loss and delay system). It has been ascertained that computation time and programming are fairly reduced in comparison with simulations in these cases, and that traffic capacity tables with fixed probability of loss can directly be calculated.

This method can also be applied to other systems, provided that their structures are not so complicated and consist of uniform subgroups. However, further studies may be required concerning generality of accuracy and numerical calculation problems, such as initial values and convergence speed of iterations.