Journal of the Korean
OR/MS Society
Vol. 22, No. 1, March 1997

An Algorithm for Computing the Fundamental Matrix of a Markov Chain †

Jeong-Soo Park* · Geon Cho**

Abstract

A stable algorithm for computing the fundamental matrix $(I-Q)^{-1}$ of a Markov chain is proposed, where Q is a substochastic matrix. The proposed algorithm utilizes the GTH algorithm (Grassmann, Taskar and Heyman, 1985) which is turned out to be stable for finding the steady state distribution of a finite Markov chain. Our algorithm involves no subtractions and therefore loss of significant digits due to cancellation is ruled out completely while Gaussian elimination involves subtractions and thus may lead to loss of accuracy due to cancellation. We present numerical evidence to show that our algorithm achieves higher accuracy than the ordinary Gaussian elimination.

Key Words : Fundamental Matrix, Gaussian Elimination, Markov Chain, Sub-stochastic Matrix.

1. Introduction

Let $Q=(q_{ij})$ be a $n\times n$ substochastic matrix (i.e., all entries are nonnegative, and $\sum_{j=1}^{n} q_{ij}\leq 1$ for all $i=1,2,\ldots,n$. Then we can enlarge Q to a $(n+1)\times(n+1)$ stochastic matrix by adding an nonnegative absorbing state vector, i.e.,

† This research was supported by Korean Science and Engineering Foundation Grant No. 941-0100-009-2, 1994, and The 1995-97 University R&D Grant Program of Cray Research, Inc. at SERI, Korea.
* Department of Statistics, Chonnam N. University
**Department of Business Administration, Chonnam N. University

$$P=\begin{pmatrix} & q_{1,n+1} \\ & q_{2,n+1} \\ Q & \vdots \\ & q_{n,n+1} \\ 0 & 1 \end{pmatrix},$$

where $q_{ij}\geq 0$ for all i and j, and $\sum_{j=1}^{n+1} q_{ij}=1$ for all $i=1,2,\ldots,n$.

Our objective in this paper is to develop an efficient and stable algorithm for computing the matrix $(I-Q)^{-1}$, where I is the $n\times n$ identity matrix. The matrix $(I-Q)^{-1}$ is called the fundamental matrix of an absorbing Markov chain, because it plays a vital role in the theory of finite Markov chains.

The inverse of $(I-Q)$ exists whenever Q^k goes to zero as k goes to infinity, and it can be expressed as an infinite series, i.e., $(I-Q)^{-1} = \sum_{k=0}^{\infty} Q^k$ (the proof is found, for example, in Seneta [1973]). Thus we assume that Q^k goes to zero as k goes to infinity. That is, we deal with only such a matrix Q assuring the existence of the inverse of $(I-Q)$. Actually the assumption is satisfied for any absorbing Markov chain (see Kemeny and Snell [1960]).

One usual way to compute the inverse matrix $(I-Q)^{-1}$ is to apply Gaussian elimination (LU-decomposition) to $I-Q$ and to solve equations by backward or forward substitutions (see Golub and Van Loan (1989) for details). However, Gaussian elimination (abbreviately, GE) involves subtractions, and thus may lead to loss of accuracy due to cancellation. Moreover, Gaussian elimination does not utilize the nonnegative substochastic property of Q.

Grassman, Taksar, and Heyman (1985) introduced a variant of GE for computing the steady-state distribution (or stationary distributions) of a Markov chain. Their algorithm, which is now known as the GTH algorithm, involves no subtractions and therefore loss of significant digits due to cancellation is ruled out completely. Empirical and theoretical evidences that the GTH algorithm computes steady-state probabilities with smaller relative error than the GE have been presented in Heyman (1987) and O'Cinneide (1992).

In this paper, we extend the key idea of GTH algorithm to propose an algorithm for computing the fundamental matrix of an absorbing Markov chain. Since our algorithm does not involve subtractions, it is clearly more accurate than the ordinary GE method.

This paper is organized as follows. In Section 2, we review the GTH algorithm for computing steady-state probabilities. The proposed algorithm for computing the fundamental matrix is described in Section 3. In Section 4, a numerical example is presented to illustrate the process of the algorithm.

Finally in Section 5, numerical experiments are given to show that our algorithm is superior to the ordinary GE method.

2. THE GTH ALGORITHM

In order to introduce the GTH algorithm, we formally state the problem for computing steady-state distribution of a finite, discrete, irreducible Markov chain. Let P be the transition probability matrix of an irreducible Markov chain with states $0, 1, 2, \ldots, n$. The problem is to compute the steady-state probability vector $\pi = (\pi_0, \pi_1, \ldots, \pi_n)$, which is detremined by

$$\pi P = \pi, \qquad \sum_{j=0}^{n} \pi_j = 1 .$$

The computation of π is of widespread interest (Kemeny and Snell (1960), and Paige, Styan and Wachter (1975), for example). To solve $\pi P = \pi$ (i.e., $\pi(P-I)=0$), we first find LU-decomposition of $(P-I)$ by using Gaussian elimination, and the backward substitution is used to get the answer. However, if some components of π are very small, then GE may give inaccurate results due to subtracting operations (see Harrod and Plemmons (1984), and Heyman (1987) for details).

The GTH algorithm is a modification of the Gaussian elimination procedure with the pivot element substituted by the sum of entries in the right-hand side of the row containing the pivot element (see Grassman, Taksar, and Heyman (1985)). Note that this substitution does not make the computation incorrect because the row sum is always zero in any step during the GE process. Now we present the GTH algorithm originally presented by Grassman, Taksar, and Heyman (1985).

Algorithm GTH

for $k = n$ down to 1 **do**

$S := \sum_{j=0}^{k-1} p_{kj}$;

for $i = k-1$ down to 1 **do**

$p_{ik} := p_{ik} / S$;

for $j = k-1$ down to 1 **do**

$p_{ij} := p_{ij} + p_{ik} p_{kj}$;

```
            end
        end
    end
    Tot := 1;
    π_0 := 1;
    for j=1 up to n do
        π_j := p_0j + Σ_{l=1}^{j-1} π_l p_lj ;
        Tot := Tot + π_j ;
    end
    for j=0 up to n do
        π_j := π_j / Tot ;
    end
```

Note that all of the arithmetic operations use only nonnegative numbers, and that there are no subtractions. Grassmann (1983) and O'Cinneide (1992) showed that this kind of nonnegative arithmetic algorithm is extremely resistant to rounding errors. This algorithm requires about $2n^3/3$ operations ($n+1$ is the number of states). Moreover, note that we are assuming that every pivot element is nonzero during the whole procedure and therefore no pivoting occurs in this algorithm. Actually Harrod and Plemmons (1984) showed that this kind of nonnegative arithmetic algorithm is stable even without pivoting. Thus no pivoting strategy also occurs in our proposed algorithm which is given in the next section.

3. PROPOSED ALGORITHM

To find an inverse matrix $(I-Q)^{-1}$of $(I-Q)$, we use the following steps.

Step 1: Obtain an LU-decomposition of $(I-Q)$, where L is a unit lower triangular matrix and U is an upper triangular matrix.

Step 2: Find a matrix $X \in R^{n \times n}$ such that

$$(I-Q)X = (LU)X = I.$$

To get an LU-decomposition of $(I-Q)$ in Step 1, we use only the crucial idea of GTH algorithm which substitutes the pivot element by the sum of the remaining entries of the row containing the pivot element. Note that the GTH algorithm deals with a stochastic matrix and thus we enlarge the substochastic matrix Q to a stochastic matrix P. In fact, our algorithm utilizes the stochastic property of P to find $(I-Q)^{-1}$.

Let x_k and e_k be the k^{th} column vectors of X and I, respectively. If we set $G=I-Q$, then our algorithm for finding $G^{-1}=(I-Q)^{-1}$ is as follows:

Algorithm FUNDM

Step 1. { Begin with G and end with $L-I+U$}

```
for k=1 up to n-1 do
    s := 0;
    for j=k+1 up to n do
        s := s + G(k,j);
    end
    G(k,k) := q_{k,n+1} - s;
            { no subtraction occurs since s is non-positive }
    for i=k+1 up to n do
        G(i,k) := G(i,k)/G(k,k);
        q_{i,n+1} := q_{i,n+1} - G(i,k) q_{k,n+1}
            { no subtraction occurs since G(i,k) is negative }
        for j=k+1 up to n do
            if (i≠j) then
                G(i,j) := G(i,k) - G(i,k)G(k,j);
            else if (i=j=n) then
                G(i,j) := q_{i,n+1};
            endif
        end
    end
end
```

Step 2. { Find $X=(I-Q)^{-1}$ by solving $(LU)X=I$}

for $k=1$ up to n **do**

solve $Ly=e_k$ to get $y=L^{-1}e_k$

solve $Ux_k=y$ to get $x_k=U^{-1}y$

end

To reduce the space complexity, $G(1:k,k)$ is overwritten by $U(1:k,k)$ for $k=1,2,\ldots,n$, and $G(k+1:n,k)$ is overwritten by $L(k+1:n,k)$ for $k=1,2,\ldots,n-1$ at the end of Step 1. Thus $G(1:k,k)$ and $G(k+1:n,k)$ are actually used instead of U and L in Step 2, respectively.

Note that the key idea of GTH algorithm is applied to find LU-decomposition of G in Step 1. That is, to find the pivot element $G(k,k)$ in the GE procedure, we simply use the key idea of the GTH algorithm discussed before instead of using the ordinary GE procedure including subtractions. This is what distinguishes our algorithm from the standard GE procedure.

Note that no cancellation error due to subtraction occurs in the algorithm because of utilizing the GTH algorithm in Step 1 and the following reasons in Step 2. Since the diagonal entries of G are nonnegative and the off-diagonal entries are negative, we can solve $Ly=e_k$ by forward substitution without cancellation for each $k=1,2,\ldots,n$. In fact, the solution y is always nonnegative for each $k=1,2,\ldots,n$. Therefore we can solve $Ux_k=y$ by backward substitution without cancellation for each $k=1,2,\ldots,n$.

4. AN ILLUSTRATION

To illustrate how our algorithm really works, we consider a substochastic matrix

$$Q=\begin{pmatrix} \frac{1}{4} & \frac{1}{4} \\ 1 & 0 \end{pmatrix}.$$

Then Q can be enlarged to a stochastic matrix

$$P=\begin{pmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Let

$$G = I - Q = \begin{pmatrix} \frac{3}{4} & -\frac{1}{4} \\ -1 & 1 \end{pmatrix}.$$

Then to find G^{-1}, we set $s=0$ and find the sum of entries in the right-hand side of the row of G, which is equal to $-1/4$. Then the pivot element $G(1,1)$ is replaced by $q_{13}-s = 1/2-(-1/4) = 3/4$, which is equal to the sum of entries in the right-hand side of the first row of P.

Now we evaluate the multiplier $G(2,1)/G(1,1)$ in Gauss transformation, which is -4/3, and store it in $G(2,1)$. We also update q_{23} by $q_{23}-G(2,1)q_{13}$, which is $0-(-4/3)(1/2) = 2/3$. Similarly, the pivot element $G(2,2)$ is replaced by the sum of entries in the right-hand side of the second row of P, which is equal to $q_{23}=2/3$. Therefore we have actually found the LU-decomposition of G from the algorithm FUNDM, which is stored in G as

$$G=\begin{pmatrix} 3/4 & -1/4 \\ -4/3 & 2/3 \end{pmatrix}.$$

In fact, since the returned matrix G from FUNDM is equal to $L-I+U$, the unit lower triangular matrix L and the upper triangular matrix U can be obtained by

$$\begin{pmatrix} 1 & 0 \\ -4/3 & 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 3/4 & -1/4 \\ 0 & 2/3 \end{pmatrix},$$

respectively. Consquently, by solving $Ly=e_k$ and $Ux_k=y$ for $k=1,2$, we are able to find

$$G^{-1}=\begin{pmatrix} 2 & 1/2 \\ 2 & 3/2 \end{pmatrix}.$$

5. NUMERICAL EXPERIMENTS

We compare the accuracy of our proposed algorithm to the GE, using test matrices. The IMSL (International Mathematics and Statistical Library, 1987) is used for the GE procedure.

The element-wise comparison is done for the following two problems. The elements $(\hat{x}_{ij})$ of fundamental matrix are computed in double-precision arithmetic and they are compared with the true answers (x_{ij}).

The absolute errors ($|x_{ij}-\hat{x}_{ij}|$) and relative errors ($|x_{ij}-\hat{x}_{ij}|/|x_{ij}|$) are computed, and their maximum and average are compared. For each test problem, we present the maximum error (MaxErr), the maximum relative error (MaxRe), the average of error (AveErr) and the average relative error (AveRe) for both algorithms.

Test Matrix 1.

$$Q=\begin{bmatrix} 1.0-10^{-6} & 10^{-7} \\ 10^{-5} & 1.0-10^{-4} \end{bmatrix},$$

$$(I-Q)^{-1}=\frac{1}{0.99\times 10^{-10}}\times\begin{bmatrix} 10^{-4} & 10^{-7} \\ 10^{-5} & 10^{-6} \end{bmatrix}.$$

Table 1. The accuracy measures of the GE and the FUNDM for test matrix 1.

	GE	FUNDM
MaxErr	2.93×10^{-5}	1.46×10^{-11}
AveErr	8.07×10^{-6}	3.67×10^{-12}
MaxRe	2.90×10^{-11}	1.44×10^{-16}
AveRe	2.18×10^{-11}	$6.42 * 10^{-17}$

Test Matrix 2.

$$Q=\begin{bmatrix} 0 & 0 & 10^{-5} \\ 0 & 1.0-10^{-5} & 0 \\ 10^{-5} & 0 & 0 \end{bmatrix},$$

$$(I-Q)^{-1}=\begin{bmatrix} 1.0+10^{-10} & 0 & 10^{-5}/(1-10^{-10}) \\ 0 & 10^{5} & 0 \\ 10^{-5}/(1-10^{-10}) & 0 & 1.0+10^{-10} \end{bmatrix}.$$

Table 2. The accuracy measures of the GE and the FUNDM for test matrix 2.

	GE	FUNDM
MaxErr	4.55×10^{-10}	1.48×10^{-11}
AveErr	5.06×10^{-8}	1.62×10^{-12}
MaxRe	1.00×10^{-10}	1.00×10^{-10}
AveRe	4.09×10^{-11}	4.00×10^{-11}

As shown in Table 1 and 2, the FUNDM algorithm produces 10-11 significant decimal digits while the GE produces only 4-7 digits. Thus FUNDM algorithm provides higher accuracy than the GE in the above two test matrices.

The following test problems are modified from the matrices considered in Heyman (1987). The fundamental matrix is first computed in double-precision arithmetic, and the element of the matrix, x_{ij}, is regarded as the "true" value, because of the difficulty of finding the true value. Then $\hat{x}_{ij}$ is computed in single-precision arithmetic. Now the relative error is $|x_{ij} - \hat{x}_{ij}|/|x_{ij}|$. This kind of comparison by the relative error provides information on the stability of algorithms.

Test Matrix 3.

$$Q=\begin{bmatrix} .999999 & 10^{-7} & 2\times10^{-7} & 3\times10^{-7} & 4-7 \\ 0.1 & 0 & 0.1 & 0.1 & 0.1 \\ 5\times10^{-7} & 0 & 0 & .555555 & 5\times10^{-7} \\ 5\times10^{-7} & 0 & .999999 & 0 & 5\times10^{-7} \\ 2\times10^{-7} & 3\times10^{-7} & 10^{-7} & 4^{-7} & .999999 \end{bmatrix}.$$

Table 3. The accuracy measures of the GE and the FUNDM for test matrix 3.

	GE	FUNDM
MaxErr	1.00×10^{-1}	5.93×10^{-2}
AveErr	1.30×10^{-2}	4.27×10^{-3}
MaxRe	2.76×10^{-2}	1.31×10^{-7}
AveRe	4.01×10^{-3}	5.28×10^{-8}

Test Matrix 4.

$$Q=\begin{bmatrix} .099 & .3 & .1 & .2 & .3 & .001 & 0 & 0 & 0 & 0 \\ .2 & .2 & .2 & .2 & .2 & 0 & 0 & 0 & 0 & 0 \\ .1 & .2 & .2 & .4 & .1 & 0 & 0 & 0 & 0 & 0 \\ .4 & .2 & .1 & .2 & .1 & 0 & 0 & 0 & 0 & 0 \\ .2 & .2 & .2 & .2 & .2 & 0 & 0 & 0 & 0 & 0 \\ .001 & 0 & 0 & 0 & 0 & .099 & .2 & .2 & .4 & .1 \\ 0 & 0 & 0 & 0 & 0 & .2 & .2 & .1 & .3 & .2 \\ 0 & 0 & 0 & 0 & 0 & .1 & .3 & .2 & .2 & .1 \\ 0 & 0 & 0 & 0 & 0 & .2 & .2 & .2 & .2 & .2 \\ 0 & 0 & 0 & 0 & 0 & .1 & .2 & .2 & .3 & .2 \end{bmatrix}.$$

Table 4. The accuracy measures of the GE and the FUNDM for test matrix 4.

	GE	FUNDM
MaxErr	2.41×10^{-1}	1.73×10^{-4}
AveErr	5.32×10^{-2}	1.42×10^{-5}
MaxRe	2.18×10^{-4}	2.09×10^{-7}
AveRe	1.64×10^{-4}	5.36×10^{-8}

As shown in Table 3 and 4, the FUNDM algorithm produces 2-4 significant decimal digits while the GE produces only 1-2 digits. Thus the FUNDM algorithm provides higher accuracy than GE at least in the above test matrices.

The computer program for FUNDM is available from the authors upon request.

REFERENCES

[1] Golub, G.H., and Van Loan, C.F. (1989). ***Matrix Computation.*** The Johns Hopkins University Press, Baltimore.

[2] Grassmann, W.K. (1983), "Rounding Errors in Some Recursive Methods Used in Computational Probability", Technical Report 73, Dept. of Operations Research, Stanford University.

[3] Grassmann, W.K., Taksar, M.I., and Heyman, D.P. (1985), "Regenerative Analysis and Steady State Distributions for Markov Chains", ***Operations Research,*** Vol. 33, 1107-1116.

[4] Harrod, W.J., and Plemmons, R.J. (1984), "Comparison of Some Direct Methods for Computing Stationary Distributions of Markov Chains", ***SIAM Journal of Scientific Computing,*** Vol. 5,

453-569.

[5] Heyman, D.P. (1987), "Further Comparisons of Direct Methods for Computing Stationary Distributions of Markov Chains", ***SIAM Journal of Algebraic and Discrete Mathematics,*** Vol. 8, 226-232.

[6] Kemeny, J.G., and Snell, J.L. (1960), ***Finite Markov Chains,*** Van Nostrand, Princeton, NJ.

[7] IMSL, Inc. (1987). ***User's Manual,*** Version 1.0, Houston, TX.

[8] O'Cinneide, C.A. (1992), "Entrywise Perturbation Theory and Error Analysis for Markov Chains", ***Numerische Mathematik,*** Vol. 66, pp. 109-120.

[9] Paige, C.C., Styan, P.H., and Wachter, P.G. (1975), "Computation of The Stationary Distribution of A Markov Chain", ***Journal of Statistical Computation and Simulation,*** Vol. 4, 173-186.

[10] Seneta, E. (1973), ***Non-negative Matrices,*** George Allen & Unwin Ltd., London.