J. Inst. Maths Applics (1978) **22**, 275–282

On the Fast Fourier Transform Inversion of Probability Generating Functions

J. K. CAVERS

Department of Systems Engineering and Computing Science, Carleton University, Ottawa, Canada K1S 5B6

[Received 19 October 1977]

The fast Fourier transform can be used to invert z transforms (including probability generating functions), but this application has received little attention or use. This correspondence makes a case for the FFT as a standard numerical tool in queuing and other statistical analyses in order to obtain probability density functions quickly and easily. Round-off and aliasing errors are discussed briefly for the queuing analyst without a signal processing background. Several variations are described which extend the accuracy and the utility of the method.

1. Motivation

IN MANY queuing and other statistical analyses, one finds it easier to work with the probability generating function (z-transform) of a probability density function (PDF) rather than with the PDF itself. There are at least two reasons for this. First, the PDF of a sum of independent random variables is the convolution of the individual PDF's, which implies the much simpler multiplication of generating functions. Second, most queuing systems have a structure such that the matrix equation describing the equilibrium probabilities is both sparse and repetitive.

The result of analysis is a transform. If only the first few moments of the PDF are desired, this should be sufficient. But if the entire PDF is required for, perhaps, a histogram of queue length or for overflow calculations, then the difficult question of inversion of the transform arises. Several inversion techniques are available.

Expansion in a power series is a formal method. But as the derivatives are at first tedious, then annoying and finally infuriating, one realizes that it will not pay off.

Partial fractions expansion after numerical calculation of the roots is more widely applicable. Its principal virtue is that the result is quasi-analytic; it has a known form with numerically determined parameters and any desired sample of the PDF can be obtained by simple substitution into the resulting formula. Its use, however, is restricted to rational polynomial generating functions. The transcendental functions so often encountered in queuing analysis cannot be handled easily.

A third, little used, technique is to perform a contour integration in the transform plane. No restrictions need be placed on the generating function as long as it corresponds to a valid PDF. This approach leads to use of the discrete Fourier

0020-2932/78/070275+08 $02.00/0

transform (DFT). Chu (1970) is an example of a mixed transform and matrix approach.

The FFT can also be used in the numerical inversion of Laplace transforms of continuous PDF's (Durbin, 1974). It is a somewhat less natural application, though, than z-transform inversion, since it introduces additional error by the replacement of an infinite integral with a finite one.

It is the purpose of this correspondence to point out the general applicability of the DFT, particularly in its fast Fourier transform (FFT) implementation, as a numerical tool in queuing and other statistical analyses, and to offer some techniques for extending its accuracy and utility. An algorithm based on the method is available separately (Cavers, 1978) in the form of a FORTRAN 4 program.

2. Inversion by DFT

If the PDF in question is $\{\ldots p_{-2}, p_{-1}, p_0, p_1, p_2 \ldots\}$ then the probability generating function $P(z)$ is defined as:

$$P(z) = \sum_{k=-\infty}^{\infty} p_k z^{-k} \tag{1}$$

in which negative instead of positive powers of z have been used in order to provide a more obvious mnemonic link with the well-developed theory of z transforms.

Most texts (e.g. De Russo, Roy & Close, 1965) derive the inversion integral

$$p_k = \frac{1}{2\pi j} \oint_C P(z) z^{k-1}\, dz,$$

where the closed curve C must enclose all poles of $P(z)$ for the one-sided transform ($p_k \equiv 0$ for $k < 0$), and C is the unit circle for the two-sided transform. If C is a circle of radius $R: z = Re^{j\theta}$, then

$$p_k = \frac{R^k}{2} \int_{-\pi}^{\pi} P(Re^{j\theta})\, e^{jk\theta}\, d\theta, \tag{2}$$

which is a Fourier series expansion. Finally, numerical integration by the trapezoidal (or piece-wise constant) rule yields the approximate result:

$$\hat{p}_k = \frac{R^k}{N} \sum_{m=0}^{N-1} P(Re^{j2\pi m/N}) e^{j2\pi km/N}, \tag{3}$$

where N is the number of distinct points. This is, of course, the inverse DFT, scaled by R^k.

The FFT offers a very fast way to obtain finite pieces (of length N samples, since $\hat{p}_k$ is periodic) of the approximate PDF. The approximation $\hat{p}_k \approx p_k$ is discussed in the next section.

3. Errors

Aliasing

Aliasing error is a property of the Fourier inversion itself, and is independent of the accuracy to which the numerical computations of (3) are performed. This phenomenon and others associated with the DFT have been described in tutorial papers (Bergland, 1969; Gentleman & Sande, 1966).

Briefly, aliasing is a replication and summation of the true PDF $\{p_k\}$ to produce the periodic approximation $\{\hat{p}_k\}$. It arises in the following way. If the θ-dependent factor in the integrand of (2) is replaced with $\hat{P}(\theta) = P(\theta)G(\theta)$, then the multiplication/convolution relation gives:

$$\hat{p}_k = \sum_{m=-\infty}^{\infty} g_m p_{k-m}.$$

Therefore, if

$$G(\theta) = \frac{2\pi}{N} \sum_{n=0}^{N-1} \delta(\theta - n2\pi/N),$$

as in the trapezoidal rule, then

$$g_m = \begin{cases} 1 & m = rN, \\ 0 & \text{elsewhere}, \end{cases}$$

where r is any integer, and

$$\hat{p}_k = \sum_{r=-\infty}^{\infty} p_{k-rN}.$$

From this it can be seen that $\hat{p}_k$ is periodic with period N and consists of the desired component plus an infinite number of aliased components.

Since a PDF must sum to 1, its terms eventually decrease with k. In queuing analyses the decrease is usually exponential or faster, so that aliasing error can usually be reduced fairly quickly by increasing N, although long-tailed PDF's can give problems.

It is tempting to apply Simpson's rule weighting to the terms of (3) to produce an improved integration. This, however, merely increases the aliasing error:

$$g_m = \begin{cases} 1 & m = rN, \\ 1/3 & m = (r+1/2)N, \\ 0 & \text{elsewhere}, \end{cases}$$

because of the longer period of $G(\theta)$. Higher-order Newton–Cotes integration formulas are even worse. The reason is that these methods require the derivatives of the integrand to decrease with increasing order. In (2), of course, the derivatives actually increase because of the exponential factor.

Round-off Error

Distinct from aliasing error is the round-off error incurred by performing the arithmetic with a finite number of digits. In the following, we assume that real numbers are stored with t bit mantissas, and that the arithmetic unit performs binary arithmetic with rounding (rather than chopping) of excess digits.

Two cases must be distinguished. The first is direct computation of the DFT by the matrix multiplication

$$\hat{p}_k = \frac{1}{N}\sum_{m=0}^{N-1} W_{km}P_m$$

where

$$P_m = P(R\exp(j2\pi m/N) \quad \text{and} \quad W_{km} = \exp(j2\pi km/N).$$

The second is the FFT computation, which for N equal to a power of 2 is equivalent to $\log_2 N$ stages, each consisting of $N/2$ two-point transforms.

Gentleman & Sande (1966) evaluated the round-off error introduced by both the direct and FFT implementations. If we denote the absolute error by α_k, the difference between the computed and the actual value of $\hat{p}_k$, their analysis of the direct method produced (after transliteration):

$$\left(\sum_{k=0}^{N-1} |\alpha_k|^2\right)^{\frac{1}{2}} \leqslant 1{\cdot}06\sqrt{2}\,2^{-t+1}N\left(\sum_{m=0}^{N-1} |P_m|^2\right)^{\frac{1}{2}} \leqslant 3(2^{-t})N\sqrt{N}.$$

For the real part of the error in any single probability component $\hat{p}_k$, a similar derivation produces:

$$|\mathrm{Re}\,\alpha_k| \leqslant 1{\cdot}06(2^{-t+1})\sqrt{N}\left(\sum_{m=0}^{N-1} |P_m|^2\right)^{\frac{1}{2}} < 3(2^{-t})N.$$

In both of the above, the second inequality follows from the fact that p_k is a PDF. For the FFT version, the corresponding result is:

$$\left(\sum_{k=0}^{N-1} |\alpha_k|^2\right)^{\frac{1}{2}} \leqslant 1{\cdot}06(2^{-t+3})\frac{\log_2 N}{\sqrt{N}}\left(\sum_{m=0}^{N-1} |P_m|^2\right)^{\frac{1}{2}} \leqslant 8{\cdot}5(2^{-t})\log_2 N$$

so that FFT is more accurate, as well as faster, than the direct method.

The above bounds are somewhat unrealistic for two reasons. The first is that they bound the maximum possible error, which is attained under a highly unlikely combination of individual rounding errors. One is normally more interested in the probable range of the error. The second reason is the assumption that $\{P_m\}$ is supplied with no error. In the inversion of generating functions, however, the calculation of $\{P_m\}$ may itself involve significant round-off error. For these reasons we will bound the variance of the error in the direct calculation of p_k. Meaningful bounds for the FFT implementation are, unfortunately, much more difficult.

The notation $\mathrm{fl}(\cdot)$ will be used to indicate the floating point evaluation of the expression given as the argument. The direct version of the DFT can therefore be written as:

$$\hat{p}_k = \mathrm{Re}\left[\mathrm{fl}\left(\sum_{m=0}^{N-1} w_{km}P_m\right)\right] = \mathrm{fl}\left(\mathrm{Re}\left[\sum_{m=0}^{N-1} w_{km}P_m\right]\right)$$
$$= \mathrm{fl}\left(\sum_{r=0}^{2N-1} u_{kr}Q_r\right),$$

where u_{kr} is alternately $\cos(2\pi k[r/2]/N)/N$ and $-\sin(2\pi k[r/2]/N)/N$, and Q_r is alternately $\mathrm{Re}[P_{[r/2]}]$ and $-j\,\mathrm{Im}[P_{[r/2]}]$.

We will consider each Q_r to have inherited some relative error ε_r from the evaluation of the z-transform. The coefficients u_{kr}, by comparison at least, will have negligible error. Making use of a lemma by Wilkinson (1963, pp. 18–19) on the floating point computation of inner products, we obtain

$$\hat{p}_k = \sum_{r=0}^{2N-1} u_{kr} Q_r (1+\varepsilon_r)(1+\xi_r)\left(\prod_{i=\max(r,1)}^{2N-1} (1+\eta_i)\right)$$

for some choice of multiplication errors ξ_r and addition errors η_k, all in the range $[-2^{-t}, 2^{-t}]$. If all errors are small, then the additive (absolute) error α_k in $\hat{p}_k$ satisfies

$$\alpha_k \approx \sum_{r=0}^{2N-1} u_{kr} Q_r (\varepsilon_r + \xi_r) + \sum_{i=1}^{2N-1} \eta_i \sum_{r=0}^{i} u_{kr} Q_r,$$

where the summations in the second term have been interchanged.

Making the natural assumption that all errors are independent, and the less defensible assumption that they are zero mean, we obtain the variance of α_k:

$$\sigma_{\alpha k}^2 = (\sigma_\varepsilon^2 + \sigma_\xi^2) \sum_{r=0}^{2N-1} u_{kr}^2 Q_r^2 + \sigma_\eta^2 \sum_{i=1}^{2N-1} \left(\sum_{r=0}^{i} u_{kr} Q_r\right)^2.$$

This expression is readily bounded. Applying the Schwartz inequality and the fact that $u_{kr} \leqslant 1/N$, we obtain

$$\sigma_\alpha^2 \leqslant (\sigma_\varepsilon^2 + \sigma_\xi^2) \sum_{r=0}^{2N-1} Q_r^2/N^2 + \sigma_\eta^2 \sum_{i=1}^{2N-1} \frac{i+1}{2N^2} \sum_{r=0}^{i} Q_r^2.$$

Running the rightmost summation up to $2N-1$, instead of i, and noting that, for a PDF,

$$\sum_{r=0}^{2N-1} Q_r^2 \leqslant N,$$

we finally obtain the bound

$$\begin{aligned} \sigma_\alpha^2 &\leqslant (\sigma_\varepsilon^2 + \sigma_\xi^2)/N + \sigma_\eta^2 \frac{(2N-1)(N+1)}{2N} \\ &\leqslant (\sigma_\varepsilon^2 + \sigma_\xi^2)/N + N\sigma_\eta^2. \end{aligned} \tag{4}$$

The interesting conclusion is that there is a range in which round-off error actually decreases with increasing N. In effect, the errors ε and ξ are averaged. As N becomes large, the addition errors η dominate and the standard deviation σ_α increases as $\sqrt{N}$, in accordance with intuition.

From (4) we see that if accuracy is required for probabilities of the order of 10^{-6} and lower, double precision must be employed in all calculations, particularly those involved in the preparation of the generating function components P_m. A useful observation is that the imaginary parts of the probabilities $\hat{p}_k$ are due only to additive error with the same variance (4) as that on the real part, thereby providing a check on the accuracy of the inversion.

4. Variations

Skipping Samples

So far an N-point transform has been needed to find the approximate probabilities $\hat{p}_0, \ldots, \hat{p}_{N-1}$. In the case of a long-tailed PDF, N can become quite large, and one begins to question whether transforms of that length are necessary. If, for example, one is interested only in the shape of the PDF and would be happy with every second or every fourth $\hat{p}_k$, then it is a waste of computation to generate the intermediate samples. Alternatively, if the FFT implementation is limited to, perhaps, 256 points, then it would be useful to extend the range of probabilities calculated out to p_{510} or p_{1020} or farther.

It is possible to use an N point transform to calculate every Kth sample $\hat{p}_0, \hat{p}_K, \hat{p}_{2K} \ldots p_{(N-1)K}$. Define $P_m = P(Re^{j2\pi m/KN})$, $0 \leqslant m \leqslant KN-1$ and its "pre-aliased" version:

$$Q_m = \frac{1}{K} \sum_{r=0}^{K-1} P_{m+r}, \quad 0 \leqslant m \leqslant N-1. \tag{5}$$

Then the N-point transform of Q,

$$q_i = \frac{1}{N} \sum_{m=0}^{N-1} Q_m e^{j2\pi im/N}, \quad 0 \leqslant i \leqslant N-1,$$

can easily be shown to be the vector of probability samples:

$$q_i = \hat{p}_{iK} \quad 0 \leqslant i \leqslant N-1. \tag{6}$$

The technique of skipping samples has reduced the amount of computation in the transform by a factor of $K(1+\log N/\log K)$. It does not, however, reduce computation in the preparation of the vector of generating function samples. As shown by (5), it is still necessary to compute KN of these samples P_m; this stage may in fact be the most time consuming of the entire procedure.

Choice of Integration Radius

Equation (2) shows that the integration radius R does not affect the inversion. In practice, however, round-off effects due to finite precision number representation produce some variation with R.

For one-sided transforms, at least, there is a rough guide to selection of R. We can think of $P(z)$ as being written in factored form:

$$P(z) = \frac{(z-\alpha_1)(z-\alpha_2)}{(z-\beta_1)(z-\beta_2)},$$

where only two poles and two zeros have been shown since this is sufficient to demonstrate the numerical effects. If R is increased it becomes more difficult to resolve the separate contributions from the poles and zeros of $P(z)$; from a distance they all appear to have approximately the same angle. If R is decreased to place the circle very close to the outermost pole, then digits are lost in the subtraction $z-\beta_{max}$. A reasonable compromise might make R 10% larger than the modulus of the maximum pole.

PDF Tails

It often happens that the primary quantity of interest is a function of the tail of the PDF. This is particularly true of buffer overflow and lost traffic calculations. An obvious example is the complementary cumulative distribution function (CCDF):

$$\sigma_k = \sum_{i=k}^{\infty} p_k.$$

Another example is the expected number of lost customers given that there is room for only k of them:

$$c_k = \sum_{i=1}^{\infty} i p_{k+i},$$

where $\{p_k\}$ is interpreted as the arrivals PDF.

In these cases the direct route of inverting $P(z)$ and performing the required computations on the resulting $\{\hat{p}_k\}$ is unattractive because of the computational effort and the lack of accuracy. Fortunately an alternative exists for the simpler tail functions. The z transforms of the tail function can be related to the original PDF. Thus the transform of $\{\sigma_k\}$ is

$$S(z) = (z-P(z))/(z-1)$$

and the transform of $\{c_k\}$ (Cavers & Woodside, 1978) is

$$C(z) = z(1+\lambda-S(z))/(z-1),$$

where $\lambda = -P'(1)$, the expected number of arrivals.

These new z transforms can, of course, be inverted by the DFT to yield the desired tail functions. The earlier discussion need be modified only slightly by the fact that they are not probability generating functions. Specifically, the inequality leading to the error bound (4) no longer applies, although similar bounds are not difficult to obtain. The other change is in the treatment of pole-zero pairs at $z = 1$. Instead of substituting $P(1) = 1$, it may be necessary to use L'Hôpital's rule manually to obtain the transform value at that point.

5. Summary

The FFT provides an economical method of inverting probability generating functions and other transforms encountered in queuing analysis. Although it provides only numerical results, as compared with the quasi-analytic form of a partial fractions expansion, it is more widely applicable. It also requires much less effort, and can be performed automatically.

This work was supported by the National Research Council of Canada.

References

BERGLAND, G. D. 1969 A guided tour of the fast Fourier transform. *IEEE Spectrum* **6,** 41–52.

CAVERS, J. K. 1978 A computer program for inversion of probability generating functions by the discrete Fourier transform, *Comput. J.*, to appear.

CAVERS, J. K. & WOODSIDE, C. M. 1978 Buffer requirements for packet transmission over a fading radio link. *IEEE Trans. Communs Tech.*, to appear.

CHU, W. W. 1970 Buffer behaviour for batch Poisson arrivals and single constant output. *IEEE Trans. Communs Tech.* **COM-18,** 613–618.

DE RUSSO, P. M., ROY, R. & CLOSE, C. M. 1965 *State Variables for Engineers*. New York: John Wiley.

DURBIN, F. 1974 Numerical inversion of Laplace transforms: an efficient improvement to Dubner and Abate's method, *Comput. J.* **17,** 371–376.

GENTLEMAN, W. M. & SANDE, G. 1965 Fast Fourier transforms—for fun and profit. *AFIPS Proc.* **29,** 563–578.

WILKINSON, J. H. 1963 *Rounding Errors in Algebraic Processes*. London: H.M.S.O.