Simplified Algorithms for Removal of the Effect of Exponentially Decaying DC-Offset on the Fourier Algorithm

Yong Guo, *Member, IEEE*, Mladen Kezunovic, *Fellow, IEEE*, and Deshu Chen, *Senior Member, IEEE*

***Abstract*—The impact of exponentially decaying direct component on the Fourier algorithm is theoretically investigated first in this paper. A new algorithm to eliminate the error caused by this decaying component in the Fourier algorithm has been proposed. Furthermore, three simplified methods are proposed to alleviate the computation burden. The performance of the Fourier algorithm improved with these methods along with the least error squares algorithm is evaluated using a simple network and a real power system modeled by EMTP. The evaluation results are presented and discussed.**

***Index Terms*—Digital relaying, Fourier algorithm, least error squares algorithm, suppression of exponentially decaying DC offset effects.**

I. Introduction

The Fourier algorithm is one of the most popular algorithms used for variety of measurements in control and protection applications [1]. It is used to accurately extract the harmonic components if the following assumptions are satisfied.

- The highest frequency of input signal is less than $(N/2)f_1$, where N is the number of sampling points per fundamental frequency (f_1) cycle.
- There is no exponentially decaying direct component.

In practice, one can use a well-designed low-pass filter to meet the requirement of assumption 1, but for assumption 2, it is not likely to be satisfied in fault condition. The exponentially decaying dc offset in some of the signals introduces fairly large errors [2]. The calculated amplitude may deviate from the real value more than 15% in the worst case [3]. For a high performance digital relay, such a large relative error cannot be tolerated.

If both a constant and an exponentially decaying dc offset are present in the input signals, an algorithm on how to apply full-cycle discrete Fourier transformer for one cycle plus two samples to calculate and compensate for the dc offset is proposed [4]. Half cycle plus two samples are needed to remove the dc offset if there are no even harmonics in the input signals. The proposed algorithms demand a lot of calculation to remove the dc offset. The data window and amount of calculation are of concern when using this algorithm for the real-time application. Benmouyal proposed a digital mimic filter to suppress the effect of an exponentially decaying component over a broad range of time constants (0.5 to five cycles and more) [1]. This filter achieves the best performance once the time constant of this exponentially decaying component is equal to the time constant of the mimic filter. Another approach is to take the decaying dc component into account without knowing its time constant. In this case, the first two terms of the Taylor series expansion are used to represent the decaying direct component, then the least error squares (LES) curve fitting technique can be applied to estimate the fundamental and other harmonics [5]. The recursive least squares curve fitting algorithm can be introduced to reduce the computation burden [6], [7]. Another method was proposed to recognize the magnitude and time constant of the decaying dc offset term in [2]. In this method, the residual terms caused by some harmonics are ignored in the estimation procedure. The assumption that these residual terms are negligible should not be taken for granted, and needs to be investigated further. The performance of Kalman filters is evaluated in [1]. It was concluded that the third-order Kalman filters is sensitive to variations of the dc offset time constant. A Kalman filter should only be superior in removing a dc-offset if its time constant is the same as the one modeled in the state, transition matrix.

This paper presents a method to eliminate the influence of exponentially decaying direct component on the Fourier algorithm. Three simplified algorithms are proposed to alleviate the computation burden. The performance evaluation will focus on their immunity to dc-offset.

II. Impact of Exponentially Decaying DC Offset on the Fourier Algorithm

Several papers discuss the problem of how large the error caused by the exponentially decaying dc offset could be [1]–[3]; however, there is no consistent conclusion regarding this error. The different models and incompatible simulation results contribute to this inconsistency. The ideal network is widely used to reveal how a large decaying dc offset can cause the estimated magnitude of the Fourier algorithm to deviate from the real magnitude. In this paper, a theoretical investigation of this error is carried out for an ideal network shown in Fig. 1. The impact of different sampling rates on this error is also studied.

Manuscript received October 22, 2001; revised April 7, 2002.

Y. Guo was with the Department of Electrical Engineering, Texas A&M University, College Station, TX 77843-3128 USA. He is now with TDK Semiconductor Corp., Tustin, CA 92780 USA.

M. Kezunovic is with the Electrical Engineering Department of Texas A&M University, College Station, TX 77843-3128 USA.

D. Chen is with the Department of Electrical Engineering, Huazhong University of Science and Technology, Wuhan 430074, China.

Digital Object Identifier 10.1109/TPWRD.2003.813894

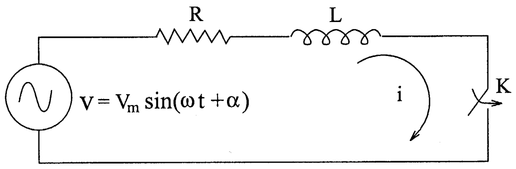

Fig. 1. Ideal network for Fourier algorithm error estimation.

In this ideal network, the equivalent source impedance will change with the power system operating condition, and the fault resistance is a variable as well. Consequently, the time constant (L/R) has to be considered as a variable.

Assume the switch K is closed at $t=0$, then the fault current $i(t)$ can be solved as

$$i(t)=A_0e^{-\frac{t}{\tau}}+A_1\sin(\omega t+\beta) \tag{1}$$

where $\tau=(L/R)$, $A_1=(v_m/\sqrt{R^2+(\omega L)^2})$, and $\beta=\alpha-\arctan(\omega L/R)$. Before switch K is closed, the current is zero. That yields

$$A_0=-A_1\sin(\beta). \tag{2}$$

Now we try to find a specific set of τ and β which causes the occurrence of the maximum amplitude error of the Fourier algorithm. Assume the sampling interval of a digital relay is (T_1/N), here T_1, is the period of the power system fundamental frequency f_1, and N is the number of sampled points per fundamental frequency cycle.

Uniformly sampling $i(t)$ every $\Delta T=(T_1/N)$, we obtain a set of discrete values $\{i_0,i_1,\ldots,i_N\}$ over a fundamental frequency cycle

$$i_k=A_0r^k+A_1\sin\left(\frac{2\pi}{N}k+\beta\right)\qquad k=0,\ldots,N \tag{3}$$

where $r=e^{-(\Delta T/\tau)}$.

The fundamental frequency components Y_{s1} and Y_{c1} can be obtained as follows after applying the Fourier algorithm to i_k

$$Y_{s1}=\frac{2}{N}\sum_{k=1}^{N-1}i_k\sin\left(\frac{2\pi}{N}k\right) \tag{4}$$

$$Y_{c1}=\frac{1}{N}\left[i_0+i_N+2\sum_{k=1}^{N-1}i_k\cos\left(\frac{2\pi}{N}k\right)\right]. \tag{5}$$

By substituting expression (3) into above equations, we can attain two concise formulas as follows:

$$Y_{s1}=\frac{2}{N}\sum_{k=1}^{N-1}A_0r^k\sin\left(\frac{2\pi}{N}k\right)+A_1\cos(\beta) \tag{6}$$

$$Y_{c1}=\frac{1}{N}\left[A_0+A_0r^N+2\sum_{k=1}^{N-1}A_0r^k\cos\left(\frac{2\pi}{N}k\right)\right]+A_1\sin(\beta). \tag{7}$$

The amplitude deviation of the Fourier algorithm can be observed from

$$f(r,N,\beta)=\frac{\sqrt{Y_{s1}^2+Y_{c1}^2}}{A_1}. \tag{8}$$

TABLE I
PEAK VALUE OF THE FOURIER ALGORITHM

N	8	12	16	20	24
Min	0.7988	0.8017	0.8026	0.8030	0.8032
Max	1.1548	1.1594	1.1631	1.1643	1.1648

By using the result from the Appendix and expression (2), the above equations can be simplified as

$$f^2(r,N,\beta)=\left[\frac{2r\sin\left(\frac{2\pi}{N}\right)}{N}\frac{(r^N-1)\sin(\beta)}{r^2-2r\cos\left(\frac{2\pi}{N}\right)+1}+\cos(\beta)\right]^2+\sin^2(\beta)\left[\frac{1-r^2}{N}\frac{r^N-1}{r^2-2r\cos\left(\frac{2\pi i}{N}\right)+1}+1\right]^2. \tag{9}$$

A numerical method is used to find the peak values of $f(r,N,\beta)$. At first, fix the sampling rate at 16 points a cycle, then change β from 1 to 360 °, and τ from 0.1 T_1 to 10 T_1. It is found that the peak value of the Fourier algorithm occurs around four points

- $f_{min}=0.8025$ at $r=0.75(\tau=0.217T_1)$, $\beta=60°$;
- $f_{min}=0.8025$ at $r=0.75(\tau=0.217T_1)$, $\beta=240°$;
- $f_{max}=1.0982$ at $r=0.93(\tau=0.861T_1)$, $\beta=139°$;
- $f_{max}=1.0982$ at $r=0.93(\tau=0.861T_1)$, $\beta=319°$.

From this result, we find out that the maximum deviation is with the minimum peak, which did not draw as much attention as the maximum peak did from the researchers in the past. The deviation of the minimum peak from the real amplitude is almost twice the one of the maximum peak in this case.

The maximum error is less than 10%, which is smaller than the 15% found in [1] and [3]. This is because the maximum and minimum value here are calculated from the first cycle of fault current. When the data window moves on, the calculated amplitude will vary with this window as well [3]. Sampling rate is another factor affecting the peak value and time at which the peak value occurs. The most commonly used sampling rates in digital relays are investigated here. For a specific τ, N and β, we will find out the peak values in the result from the Fourier algorithm for the first $(N/2)$ data window after the fault occurs. The study results are listed in Table I.

From this table, it can be observed that the peak value changes a little for different sampling rates. The position where the peak occurs will change with different τ, β, N and the data window position, although the variation of the peak value itself is very small.

From the evaluation using the ideal network, we can draw conclusion that the dc offset may have drastic impact on the Fourier algorithm. If no measurement correction is adopted, the relative error of the amplitude from the Fourier algorithm may reach 20%, which is purely caused by this decaying dc offset.

III. Improved Fourier Algorithm

In this paper, a new algorithm is proposed to eliminate the exponentially decaying direct component from the Fourier algorithm. The first assumption given in Section I as well as the assumption that there is no subharmonics in the input signal $Y(t)$ need to be satisfied.

Based on these assumptions, we can express the input signal $Y(t)$ as

$$Y(t) = A_0 e^{-\frac{t}{\tau}} + \sum_{m=1}^{\frac{N}{2}-1} A_m \sin(m\omega_1 t + \varphi_m) \tag{10}$$

where

A_0 magnitude of the dc offset;
τ time constant of the decaying component;
A_m amplitude of the m-th ac component;
φ_m initial angle of the m-th ac component.

Uniformly sampling $Y(t)$ every $\Delta T = (T_1/N)$, we obtain a set of discrete values $\{Y_0, Y_1, \ldots, Y_N\}$ in a fundamental frequency cycle. That is

$$Y_k = A_0 r^k + \sum_{m=1}^{\frac{N}{2}-1} A_m \sin\left(\frac{2\pi k}{N} m + \varphi_m\right). \tag{11}$$

In the above equation, $k = 0, 1, \ldots, N$, and $r = e^{-(\Delta T/\tau)}$.

Phadke, *et al.* used two partial sums to estimate A_0 and r [2]. However, a residual term exists in the partial sums, and this residual term may cause an error in the estimation of A_0 and r. In this paper, we define a new partial sum term PS_1 as follows:

$$\begin{aligned} PS_1 &= Y_1 + Y_3 + \ldots + Y_{N-1} \\ &= \sum_{i=1}^{\frac{N}{2}} A_0 r^{2i-1} + \sum_{i=1}^{\frac{N}{2}} \\ &\quad \times \left\{ \sum_{m=1}^{\frac{N}{2}-1} A_m \sin\left[\frac{2\pi(2i-1)}{N} m + \varphi_m\right] \right\}. \end{aligned} \tag{12}$$

From simple trigonometric relationship, we know that

$$\sum_{i=1}^{\frac{N}{2}} \left\{ \sum_{m=1}^{\frac{N}{2}-1} A_m \sin\left[\frac{2\pi(2i-1)}{N} m + \varphi_m\right] \right\} = 0. \tag{13}$$

Accordingly

$$PS_1 = A_0 \frac{r(r^N - 1)}{r^2 - 1}. \tag{14}$$

Similarly, we define another partial sum as

$$\begin{aligned} PS_2 &= Y_2 + Y_4 + \ldots + Y_N \\ &= A_0 \frac{r^2(r^N - 1)}{r^2 - 1}. \end{aligned} \tag{15}$$

From (14) and (15), we can solve the r and A_0 as

$$r = \frac{PS_2}{PS_1} \tag{16}$$

$$A_0 = \frac{r^2 - 1}{r(r^N - 1)} PS_1. \tag{17}$$

Once r and A_0 are obtained, the set of sampled values can be modified as

$$Y_{k\_new} = Y_k - A_0 r^k \quad k = 0, 1, \ldots, N. \tag{18}$$

This new set of sampled values no longer contains the exponentially decaying component. Applying the Fourier algorithm to this new set of samples, we can accurately extract the i-th harmonic component.

Denote the cosine and sine part of the new algorithm as Y_{si_new} and Y_{ci_new}, respectively, then

$$\begin{aligned} Y_{si\_new} &= \frac{2}{N} \sum_{m=1}^{N-1} Y_{k\_new} \sin\left(\frac{2\pi i}{N} k\right) \\ &= Y_{si} - \frac{2A_0}{N} \sum_{m=1}^{N-1} r^k \sin\left(\frac{2\pi i}{N} k\right) \end{aligned} \tag{19}$$

$$\begin{aligned} Y_{ci\_new} &= \frac{1}{N} \left[Y_{0\_new} + Y_{N\_new} + 2 \sum_{m=1}^{N-1} Y_{k\_new} \cos\left(\frac{2\pi i}{N} k\right) \right] \\ &= Y_{ci} - \frac{A_0}{N} \left[1 + r^N + 2 \sum_{m=1}^{N-1} r^k \cos\left(\frac{2\pi}{N} k\right) \right] \end{aligned} \tag{20}$$

where Y_{si} and Y_{ci} are the cosine and sine parts of the Fourier algorithm for the unmodified sampled set. By using the derivation in the Appendix and substituting $A_0 = [(r^2 - 1)/(r(r^N - 1))]PS_1$ into (19) and (20), finally a compact expression of Y_{si_new} and Y_{ci_new} is obtained

$$\begin{aligned} Y_{si\_new} &= Y_{si} + \frac{2 \sin\left(\frac{2\pi i}{N}\right)}{N} \frac{r^2 - 1}{r^2 - 2r \cos\left(\frac{2\pi i}{N}\right) + 1} PS_1 \\ &= Y_{si} + S(r) PS_1 \end{aligned} \tag{21}$$

$$\begin{aligned} Y_{ci\_new} &= Y_{ci} + \frac{1}{N} \frac{1 - r^2}{r} \frac{r^2 - 1}{r^2 - 2r \cos\left(\frac{2\pi i}{N}\right) + 1} PS_1 \\ &= Y_{ci} + C(r) PS_1. \end{aligned} \tag{22}$$

Actually, it is not necessary to calculate $S(r)$ and $C(r)$ every iteration. Once r is estimated, $S(r)$ and $C(r)$ are two constants for this specific decaying waveform's time constant τ. Substituting these two constants to the next iteration will make the calculation for eliminating dc offset very simple. PS_1 and PS_2 can be calculated in a recursive fashion

$$PS_1^{new} = PS_2^{old} \tag{23}$$

$$PS_2^{new} = PS_1^{old} + Y_{N+1} - Y_1 \tag{24}$$

where superscript new stands for the data window for the set of samples $\{Y_1, Y_2, \ldots, Y_{N+1}\}$, and old stands for the previous window. The improved algorithm can also be implemented in a recursive fashion by combining the recursive Fourier algorithm with above formulas.

The above algorithm can totally eliminate the exponentially decaying component if the input signal can be described by (10). Even though every effort has been made to reduce the computational burden, it is still very difficult to accurately calculate the $S(r)$ and $C(r)$ for the microprocessor without the floating-point operation instruction. How to make a compromise between the calculation burden and accuracy is always a challenge for digital relay designer. A tradeoff has been made in the following three simplified algorithms.

IV. Simplified Algorithms

A. Simplified Algorithm 1

At first, a straight line is used to approximate the exponentially decaying dc offset. This straight line is determined by the first order Taylor-series expansion at $(T_1/2)$ over interval [0, T_1]. That is

$$\begin{aligned} A_0 e^{-\frac{t}{\tau}} &\cong C_0 + C_1\left(t - \frac{T_1}{2}\right) \\ &= B_0 + B_1 t. \end{aligned} \tag{25}$$

After this approximation, the two partial sums become

$$\begin{aligned} PS_1 &= \sum_{i=1}^{\frac{N}{2}} \left[B_0 + B_1 \Delta T(2i-1)\right] \\ &= \frac{N}{2} B_0 + \frac{N^2}{4} B_1 \Delta T \end{aligned} \tag{26}$$

$$\begin{aligned} PS_2 &= \sum_{i=1}^{\frac{N}{2}} [B_0 + B_1 \Delta T 2i] \\ &= \frac{N}{2} B_0 + \frac{N(N+2)}{4} B_1 \Delta T. \end{aligned} \tag{27}$$

From these two equations, we solve

$$B_1 \Delta T = \frac{2}{N}(PS_2 - PS_1). \tag{28}$$

Since the Fourier algorithm has the ability to filter out the nondecaying direct component, it is not necessary to subtract B_0 from the sampled values. Therefore, the set of samples can be modified as

$$Y_{k\_new} = Y_k - B_1 \Delta T_k \quad k = 0, 1, \ldots, N. \tag{29}$$

After the Fourier algorithm is applied to this new value, a compact form of this simplified algorithm is obtained

$$Y_{si\_1} = Y_{si} + \frac{2}{N}\coth\left(\frac{\pi i}{N}\right)(PS_2 - PS_1) \tag{30}$$

$$Y_{ci\_1} = Y_{ci}. \tag{31}$$

Obviously, there is no modification for Y_{ci_1}, and the constant $(2/N)\coth((\pi i)/(N))$ can be precalculated. Also, $PS_2 - PS_1$ calculation can be implemented in a recursive fashion

$$(PS_2 - PS_1)^{new} = Y_{N+1} - Y_1 - (PS_2 - PS_1)^{old}. \tag{32}$$

B. Simplified Algorithm 2

It is expected that a higher accuracy in the approximation can be achieved by expanding $A_0 e^{-(t/\tau)}$ into the second order Taylor series at $(T_1/2)$ over interval [0, T_1]. That is

$$A_0 e^{-\frac{t}{\tau}} \cong B_0 + B_1 t + B_2 t^2. \tag{33}$$

This quadratic fit needs one more equation to solve for B_0, B_1, and B_2. A new partial sum is defined as

$$PS_0 = \sum_{i=0}^{\frac{N}{2}-1} Y_{2i}. \tag{34}$$

Similarly, three linear equations can be obtained, and we can solve

$$B_1 \Delta T = \frac{1}{N}(PS_2 - PS_0) - (PS_2 - 2PS_1 + PS_0) \tag{35}$$

$$B_2 \Delta T^2 = \frac{1}{N}(PS_2 - 2PS_1 + PS_0). \tag{36}$$

By modifying the set of sampled values with

$$Y_{k\_new} = Y_k - B_1 \Delta T k - B_2 \Delta T^2 k^2 \tag{37}$$

and after applying the Fourier algorithm to this new value, the following concise formulas are derived:

$$Y_{si\_2} = Y_{si} + \frac{1}{N}\coth\left(\frac{\pi i}{N}\right)(PS_2 - PS_0) \tag{38}$$

$$Y_{ci\_2} = Y_{ci} - \frac{1}{N}\csc^2\left(\frac{\pi i}{N}\right)(PS_2 - 2PS_1 + PS_0). \tag{39}$$

$PS_2 - PS_0$ can be simplified as $Y_N - Y_0$. The modification term of Y_{ci_2} can be implemented in a recursive fashion to reduce the computational burden.

C. Simplified Algorithm 3

The above two simplified algorithms based on the Taylor expansion are a good approximation only around expansion point, and the error may be large over the whole interval [0, T_1]. The problem of approximating a continuous function by a finite linear combination of given functions can be approached in various ways. For the purpose of representing arbitrary continuous functions by elementary functions (e.g. polynomials), it is natural to use the maximum deviation of the approximation as a measure of the quality of approximation [8]. To make this approximation feasible, the best uniform approximation linear polynomial $\tilde{u} = \tilde{B}_0 + \tilde{B}_1 t$ is used here, and it is defined as

$$\max_{0 \le t \le T_1}\left|A_0 e^{-\frac{t}{\tau}} - \tilde{u}\right| = \inf_{u \in H_1} \max_{0 \le t \le T_1}\left|A_0 e^{-\frac{t}{\tau}} - u\right| \tag{40}$$

where H_1 is the subset of arbitrary polynomials whose order is less than or equal to 1, and the element u in the subset H_1 can be expressed as

$$u = B_0 + B_1 t \tag{41}$$

where B_0 and B_1 are arbitrary real numbers. In other words, the best uniform approximation polynomial in $\tilde{u} = \tilde{B}_0 + \tilde{B}_1 t$ is the one whose deviation is the smallest one of any linear polynomial approximations of $u = B_0 + B_1 t$ for $A_0 e^{-(t/\tau)}$ over interval [0, T_1].

The continuous function is the exponential function $f(t) = A_0 e^{-(t/\tau)}$, and its second derivative does not change sign over interval [0, T_1]. In this case, a three point alternant is given by $0 = t_1 < t_2 < t_3 = T_1$, where t_2 is chosen so that $f'(t_2) = ((f(t_3) - f(t_1))/(t_3 - t_1))$. Then, the best uniform approximation linear polynomial is

$$\tilde{u} = \frac{f(t_3) - f(t_1)}{t_3 - t_1}\left(t - \frac{t_1 + t_2}{2}\right) + \frac{1}{2}\left[f(t_1) + f(t2)\right]. \tag{42}$$

Fig. 2 shows the solution of the best uniform approximation linear polynomial $\tilde{u}$.

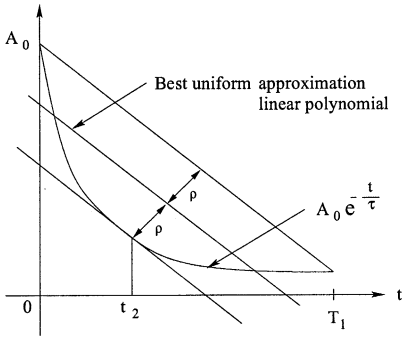

Fig. 2. Best uniform approximation method.

Since $Y_N - Y_0 = f(t_3) - f(t_1)$, therefore

$$\tilde{B}_1 = \frac{f(t_3) - f(t_1)}{t_3 - t_1} = \frac{Y_N - Y_0}{T_1}. \tag{43}$$

The new algorithm can be obtained after modifying the set of sampled values like the simplified algorithm 1

$$Y_{si\_3} = Y_{si} + \frac{1}{N}\coth\left(\frac{\pi i}{N}\right)(Y_N - Y_0) \tag{44}$$

$$Y_{ci\_3} = Y_{ci}. \tag{45}$$

It is of interest to note that the cosine part Y_{si_3} is exactly the same one as in Y_{si_2}. That means Y_{si_3} has the ability to eliminate the second order polynomial component, and we expect that the second order polynomial can approximate function $f(t) = A_0 e^{-(t/\tau)}$ more accurately.

V. Performance Evaluation of the Simplified Algorithms Using the Ideal Network

In this section, the ideal network is used to evaluate the performance of the three simplified algorithms and LES algorithm. Two performance indices are proposed in our study.

The first index PI_1 is used to find the peak value of an algorithm over a range of time constants from 0.5 to five cycles. This index demonstrates the worst case an algorithm may encounter in service, and it corresponds to particular values of the dc offset time constants τ and β. The value of β is determined by the fault inception angle, and the time constant τ is also a variable, which depends upon the system configuration and fault resistance.

The second index PI_2 is introduced to evaluate the algorithm over the same range of time constants. It is defined as follows: for every time constant, an algorithm can reach its peak (minimum and maximum value) at a particular β. By averaging the whole set of these peak values, we get the second index PI_2 which indicates the average minimum and maximum value of the algorithm over the above range.

These two indices are very similar to the indices proposed in [1]. The Fourier algorithm performance is also evaluated by using these two indices. To do so, we can use the Fourier algorithm performance index as a bridge to compare the performance of the three simplified algorithms with the performance of other algorithms which have been studied in detail in [1].

The evaluation results for sampling rate $\mathrm{N} = 16$ are listed in Table II, where *Fou* stands for the Fourier algorithm, *Sim* for the simplified algorithm, and *LES* for the least error squares algorithm.

TABLE II
Algorithm Amplitude Indices Over 0.5 to Five Cycles

Alg.	PI_1		PI_2	
	Min.	**Max.**	**Min.**	**Max.**
Fou.	0.83793	1.16335	0.93864	1.09670
Sim1	0.91911	1.00127	0.98748	1.00060
Sim2	0.98391	1.01732	0.99858	1.00115
Sim3	0.91746	1.00202	0.98741	1.00009
LES	0.91215	1.00161	0.98660	1.00007

TABLE III
Algorithm Amplitude Indices Over 0.1 to Five Cycles

Alg.	PI_1		PI_2	
	Min.	**Max.**	**Min.**	**Max.**
Fou.	0.80247	1.16335	0.92866	1.09756
Sim1	0.82201	1.02764	0.97768	1.00107
Sim2	0.95004	1.38682	0.99595	1.00874
Sim3	0.79557	1.06854	0.97693	1.00142
LES	0.78740	1.04615	0.97550	1.00106

From the simulation results, we observe that

- Simplified algorithm 1 narrows the amplitude deviation. Specially for maximum peak, it reduces the overshoot from 1.161 09 to 1.001 27.
- Simplified algorithm 2 is the best algorithm among these five algorithms. Its peak deviation is limited to 1.732% of the real amplitude.
- Simplified algorithm 3 has almost the same performance as the simplified algorithm 1 and LES algorithm. The overshoot of the simplified algorithm 2 is the largest one when compared to the simplified algorithms and LES algorithm. In this sense, to use the index PI_1 as the only indication in the performance evaluation is not fair, since we are dealing with nonlinear peak-value solution problem with multiple variables.
- From the index PI_2, we can see that the simplified algorithm 2 is the best algorithm among these five algorithms, and its deviation is less than 0.15%.

As it was pointed out earlier, the results presented in Table II are evaluated over the range of time constants from 0.5 to five cycles. If this interval extends from 0.1 to five cycles, the minimum value of the Fourier algorithm will be exactly the same as the result shown in Table I. The data window is moving from the first fault cycle to 1.5 cycles, so the maximum value in the Fourier algorithm can be found. Reference [3] points out that such period (1 to 1.25 cycles after the occurrence of the dc offset) can be used to find out the maximum and minimum error for the Fourier algorithm. The study results are shown in Table III after the time constants range is extended from [0.5, 5] cycles to [0.l, 5] cycles.

We can see that the performance of the simplified algorithms and LES algorithm deteriorates very dramatically as τ changes over the period 0.1 to 0.5 cycles. In this period, the dc offset

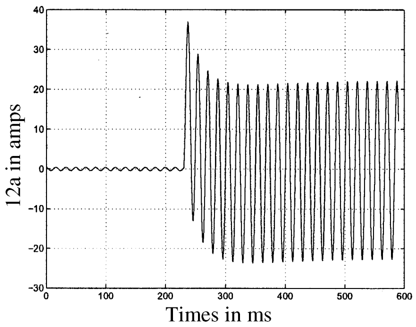

Fig. 3. Filter input waveform of I_{2a}.

TABLE IV
AMPLITUDE VARIATION OF DIFFERENT ALGORITHMS BASED ON THE SAMPLES TAKEN AFTER THE FILTERS

Alg.	I_{2a}		I_{2b}		I_{2c}	
	A^{pu}_{min}	A^{pu}_{max}	A^{pu}_{min}	A^{pu}_{max}	A^{pu}_{min}	A^{pu}_{max}
Fou.	0.9102	1.1276	0.9990	1.0010	0.9199	1.1139
Sim1	0.9913	1.0123	0.9946	1.0199	0.9837	1.0108
Sim2	0.9894	1.0492	0.9870	1.0399	0.9249	1.0490
Sim3	0.9821	1.0119	0.9925	1.0064	0.9844	1.0107
Imp.	0.9952	1.0162	0.9983	1.0105	0.9961	1.0140
LES	0.9823	1.0127	0.9974	1.0024	0.9826	1.0114

decays very fast. It is very hard to compensate it, not to mention the calculation error for this compensation in a real digital relay implementation. The overshoot of the simplified algorithm 2 is quite large over this period. The index PI_2 indicates an improved performance in the simplified algorithms and LES algorithm over the Fourier algorithm. The simplified algorithms 1 and 3 as well as the LES algorithm have very similar performance in this case. The simplified algorithms 1 and 3 are computationally less demanding than the LES algorithm. For N $=$ 16, the simplified algorithm 3 needs 9 multiplication and 28 addition/subtraction operation to calculate Y_{si_3} and Y_{ci_3}. For LES algorithm, 17 multiplication and 31 addition/subtraction operation are needed to obtain Y_{si_les} and Y_{ci_les}. Obviously, a significant computation reduction is achieved for the simplified algorithm 3 over the LES algorithm.

VI. Transient Performance Evaluation

The ideal network evaluation method focuses on the error purely caused by the exponentially decaying dc offset. In a real power system, the transient signals may contain not only the decaying component, but also high frequency components and other noise. In this study, a real power system (a section from CenterPoint Energy Company's 345-kV transmission system) is modeled to generate waveforms for transient performance evaluation.

The short transmission line (10.14 mi) between NBELT and KING modeled by EMTP is used for the transient performance evaluation [9]. A three phase fault is applied at the middle of this transmission line. Simulation results show that there is no significant exponentially decaying dc offset in voltage signals. As expected, quite large exponentially decaying dc offset does exist in some current signals as shown in Fig. 3 for the KING side current in phase A.

In this study, 960-Hz sampling rate is used (i.e. 16 samples per cycle) [the output frequency of EMTP waveform is $6 \times 960 =$ 5760 Hz]. A fifth-order active low-pass Butterworth filter with cutoff frequency 360 Hz at -20 dB is employed here.

Determination of the magnitude variation for different measurement algorithms is of most concern during a disturbance. The steady state magnitude calculation for a fault current (ten cycles after the fault happens) is used as a reference to measure the variation of the magnitude calculation. As the data window advances, the magnitude computed by an algorithm will oscillate around the steady state magnitude A_{ss}. The oscillation in the results caused by the exponentially decaying dc component and other noise fades out and finally the results settle down to the steady state magnitude. The calculated maximum and minimum magnitude (A_{max} and A_{min}) occur between the first and second cycle of sampled data following the fault inception. Based on this observation, we define two indices for performance evaluation as

$$A^{pu}_{max} = \frac{A_{max}}{A_{ss}} \quad A^{pu}_{min} = \frac{A_{min}}{A_{ss}}. \tag{46}$$

Obviously, the smaller the difference between these two values, the better the performance of an algorithm.

The decaying dc offset and high frequency components play a major role in the transient performance evaluation for different algorithms. The low-pass filter reduces the high frequency components more than 20 dB before the signal reaches the A/D converter of the digital relay. The simulation result listed in Table IV shows the impact of the decaying dc offset for the algorithms under evaluation.

In this table, the subscript 2 represents the KING side signal, and *Imp* stands for the improved algorithm defined by (21) and (22). From these simulation results, we can conclude that

- There is a large error in the Fourier algorithm. That means the impact of the decaying dc offset on the Fourier algorithm cannot be ignored.
- For the Fourier algorithm, the maximum magnitude deviation is with I_{2a}, which has a large decaying dc offset shown in Fig. 3. In this case, all the proposed algorithms have a better performance than the Fourier algorithm.
- Not surprisingly, the overshoot of simplified algorithm 2 is the largest one. The undershoot of simplified algorithm 2 is also the largest one, which is not seen in the evaluation using the ideal network.
- The simplified algorithm 1 is very susceptible to the high frequency components. If the sampled data of I_{2a} before the low-pass filter is applied to this algorithm, the deviation becomes 0.9422 $\sim$ 1.0521. That also happens to the improved algorithm: the deviation is 0.9294 $\sim$ 1.0683. The simplified algorithm 3 has almost the same performance in this case: the deviation is 0.9844 $\sim$ 1.0122.
- The performance difference among the simplified algorithm 1, 3 and LES algorithm is negligible (the deviation is 0.9837 $\sim$ 1.0199, 0.9821 $\sim$ 1.0119 and 0.9823 $\sim$ 1.0127, respectively).

- The good performance of the simplified algorithm 3 comes from its ability to suppress not only the decaying dc component, but also the high frequency harmonics. Another advantage of the simplified algorithm 3 is that the modification is very simple: only one multiplication and one subtraction are needed, and no recursive computation is involved.
- The improved algorithm has the best performance among these six algorithms. Its deviation varies only from 0.9952 to 1.0162. However, this algorithm is computationally quite demanding and may not be suitable for real-time applications.

VII. Conclusion

- Exponentially decaying dc offset has a significant impact on the Fourier algorithm. An ideal network consisting of the lumped R-L circuits is used to reveal this impact. In the worst case, the deviation may be over 20% of the real magnitude.
- An improved algorithm by using a partial summation technique is proposed to eliminate the influence of the decaying dc offset on the Fourier algorithm.
- Three simplified algorithms are proposed to compromise between the computational burden and accuracy.
- The simplified algorithm performance evaluation based on the current signals from the ideal network demonstrates the significant performance improvement over the Fourier algorithm.
- A transmission line in a 345-kV power system modeled by EMTP is utilized to evaluate the algorithm transient performance. Surprisingly, the simplified algorithm 2 has the worst performance under this study.
- The performance of the improved algorithm is very impressive if the signal fits into the assumed model,otherwise its performance degrades. The tradeoff is that this algorithm is computationally quite demanding.
- The simplified algorithms 1, 3 and LES algorithm have almost the same performance using the ideal and actual network evaluation. By comparing the modification simplicity and improved performance over the Fourier algorithm, the simplified algorithm 3 is considered to be the best candidate to replace the Fourier algorithm for the processing of the current signal.

Appendix

$$\begin{aligned}\sum_{k=1}^{N-1} r^k e^{j\frac{2\pi i}{N}k} &= \sum_{k=1}^{N-1}\left(re^{j\frac{2\pi i}{N}}\right)^k \\ &= \frac{r^N e^{j\frac{2\pi i}{N}N} - re^{j\frac{2\pi i}{N}}}{re^{j\frac{2\pi i}{N}} - 1} \\ &= \frac{r^N - 1}{r^2 - 2r\cos\left(\frac{2\pi i}{N}\right) + 1} \\ &\quad \times \left[r\cos\left(\frac{2\pi i}{N}\right) - 1 - jr\sin\left(\frac{2\pi i}{N}\right)\right] - 1\end{aligned} .$$

That is

$$\begin{aligned}\sum_{k=1}^{N-1}\left[r^k \sin\left(\frac{2\pi i}{N}k\right)\right] &= \frac{r(1-r^N)\sin\left(\frac{2\pi i}{N}\right)}{r^2 - 2r\cos\left(\frac{2\pi i}{N}\right) + 1} \\ \sum_{k=1}^{N-1}\left[r^k \cos\left(\frac{2\pi i}{N}k\right)\right] &= \frac{(r^N - 1)\left[r\cos\left(\frac{2\pi i}{N}\right) - 1\right]}{r^2 - 2r\cos\left(\frac{2\pi i}{N}\right) + 1} - 1\end{aligned} .$$

References

[1] G.Gabriel Benmouyal, “Removal of DC-offset in current waveforms using digital mimic filtering,” *IEEE Trans. Power Delivery*, vol. 10, pp. 621–630, Apr. 1995.

[2] A. G. Phadke, T. Hlibka, and M. Ibrahim, “A digital computer system for EHV substation: analysis and field tests,” *IEEE Trans. Power Apparat. Syst.*, vol. PAS-95, pp. 291–301, Jan./Feb. 1976.

[3] D. Chen and X. Yin, *The Principle and Technology of Computer Relay*. Beijing, China: Hydraulics and Electrical Power Publication, Nov. 1992.

[4] J.-C. Gu and S.-L. Yu, “Removal of DC-offset in current and voltages signals using a novel Fourier filter algorithm,” *IEEE Trans. Power Delivery*, vol. 15, pp. 73–79, Jan. 2000.

[5] M. S. Sachdev and M. A. Baribeau, “A new algorithm for digital impedance relays,” *IEEE Trans. Power Apparat. Syst.*, vol. 98, pp. 2232–2240, Nov./Dec. 1979.

[6] A. Isaksson, “Digital protective relaying through recursive least-squares identification,” *Proc. Inst. Elect. Eng.—Gen., Transm., Dist.*, pt. C, vol. 135, pp. 441–449, 1988.

[7] M. S. Sachdev and M. Nagpal, “A recursive least error squares algorithm for power system relaying and measurement applications,” *IEEE Trans. Power Delivery*, vol. 6, pp. 1008–1015, July 1991.

[8] G. Hämmerlin and K. Hoffmann, *Numerical Mathematics*. New York: Springer-Verlag, 1991.

[9] M. Kezunovic and A. Abur, Protective Relay Workstation Applications of Digital Simulator for Protective Relay Studies: System Requirement Specifications, Final Rep., EPRI RP 3192-01, Phase 1, EPRI TR 102781, Oct. 1993.

Yong Guo (M’00) received the B.S. and M.S. degrees in electrical engineering from Huazhong University of Science and Technology, Wuhan, China, in 1984 and 1987, respectively. He received the Ph.D. degree from Texas A&M University, College Station, in 1997.

Mladen Kezunovic (S’77–M’80–SM’85–F’99) received the Dipl. Ing. degree in electrical engineering from the University of Sarajevo, Bosnia-Herzegovina, in 1974, and the M.S. and Ph.D. degrees in electrical engineering from the University of Kansas, Lawrence, in 1977 and 1980, respectively.

Currently, he is the Eugene E. Webb Professor and Director of Electric Power and Power Electronics Institute at Texas A&M University, College Station. He was with Westinghouse Electric Corporation, Pittsburgh, PA, and the Energoinvest Company, Sarajevo, Bosnia-Herzegovina. He was also with the University of Sarajevo, Bosnia-Herzegovina, and a Visiting Associate Professor at Washington State University, Pullman, from 1986 to 1987. His main research interests are digital simulators and simulation methods for relay testing as well as application of intelligent methods to power system monitoring, control, and protection.

Dr. Kezunovic is also a Fellow of IEEE and a member of CIGRE-Paris.

Deshu Chen (SM’87) graduated in electrical engineering with a B.S. degree from Huazhong University of Science and Technology (HUST), Wuhan, China, in 1952, and the Ph.D. degree in protective relaying from Harbin Institute of Technology, Harbin, China, in 1955.

Currently, he is a Professor with HUST, where he has been since 1952.