Mathematics and Statistics 11(1): 34-50, 2023 http://www.hrpub.org
DOI: 10.13189/ms.2023.110105

Developing Average Run Length for Monitoring Changes in the Mean on the Presence of Long Memory under Seasonal Fractionally Integrated MAX Model

Wilasinee Peerajit

Faculty of Applied Science, King Mongkut's University of Technology North Bangkok (KMUTNB), Thailand

Received September 24, 2022; Revised December 7, 2022; Accepted December 22, 2022

Cite This Paper in the Following Citation Styles

(a): *[1] Wilasinee Peerajit , "Developing Average Run Length for Monitoring Changes in the Mean on the Presence of Long Memory under Seasonal Fractionally Integrated MAX Model," Mathematics and Statistics, Vol. 11, No. 1, pp. 34 - 50, 2023. DOI: 10.13189/ms.2023.110105.*

(b): *Wilasinee Peerajit (2023). Developing Average Run Length for Monitoring Changes in the Mean on the Presence of Long Memory under Seasonal Fractionally Integrated MAX Model. Mathematics and Statistics, 11(1), 34 - 50. DOI: 10.13189/ms.2023.110105.*

Abstract The cumulative sum (CUSUM) control chart can sensitively detect small-to-moderate shifts in the process mean. The average run length (ARL) is a popular technique used to determine the performance of a control chart. Recently, several researchers investigated the performance of processes on a CUSUM control chart by evaluating the ARL using either Monte Carlo simulation or Markov chain. As these methods only yield approximate results, we developed solutions for the exact ARL by using explicit formulas based on an integral equation (IE) for studying the performance of a CUSUM control chart running a long-memory process with exponential white noise. The long-memory process observations are derived from a seasonal fractionally integrated MAX model while focusing on X. The existence and uniqueness of the solution for calculating the ARL via explicit formulas were proved by using Banach's fixed-point theorem. The accuracy percentage of the explicit formulas against the approximate ARL obtained via the numerical IE method was greater than 99%, which indicates excellent agreement between the two methods. An important conclusion of this study is that the proposed solution for the ARL using explicit formulas could sensitively detect changes in the process mean on a CUSUM control chart in this situation. Finally, an illustrative case study is provided to show the efficacy of the proposed explicit formulas with processes involving real data.

Keywords CUSUM Control Chart, Exponential White Noise, Numerical Integral Equation (NIE) Method, SFIMAX(D, Q, X)$_L$

1. Introduction

The cumulative sum (CUSUM) control chart is an effective tool widely used in industrial and medical procedures for quality control [1,2]. The CUSUM control chart statistic accumulates past and current information about the process. It performs as well as the exponentially weighted moving-average (EWMA) control chart [3] and is superior to the Shewhart control chart [4] as it has shorter average run lengths (ARLs) that make it more sensitive for detecting small-to-moderate shifts in the process parameters. It is used in a variety of fields, such as finance, and economics, among others. For example, Hibbert [5] proposed using the CUSUM control chart in analytical laboratories. Kateman [6] suggested that the CUSUM control chart is more suited to the requirements of laboratory experiments than the Shewhart control chart. Moreover, Sheng-Shu [7] showed that the CUSUM control chart is more efficient for monitoring a wafer production process better than the EWMA control chart. The CUSUM control chart is reportedly the best for quality control in the healthcare domain [8]. The CUSUM control chart was used in the present study due to its capability of detecting small-to-moderate shifts in process parameters. Detailed discussions on taking advantage of the benefits of using an upper-sided CUSUM control chart are available in the literature [9–12].

Assessment of control charts is generally carried out under the assumption that observations are independent and identically distributed (i.i.d.) and normally distributed. However, in practical applications, the CUSUM control chart has often been used to monitor processes with autocorrelated observations (for reference, see [13–15]. The concept is to fit the autocorrelated data of the observed process into a time-series model, such that forecasting of each observation can be made by using the information from the previous observations and then applying the fitted data to the residuals of the control chart. Time-series observations can include trend, season, and autocorrelation aspects. Furthermore, the mean and variance of time-series observations fluctuate over a given time frame. For monitoring autocorrelated processes, the model must provide a framework for designating statistical control. Time-series models that many researchers have studied include autoregressive (AR), moving-average (MA), ARMA, AR integrated MA (ARIMA), and AR fractionally integrated MA (ARFIMA). The approach has also been applied to forecasting models with real observations in many fields, such as medical science, finance, and economics, among others. A time series demonstrating the properties of long-memory and periodic behavior with seasonality through the ARFIMA model is described as a long-memory seasonal ARFIMA (SARFIMA$(P,D,Q)_L$) process. The long-memory characteristics of a time series are captured by the fractional integration parameter [16], denoted by D for the SARFIMA process, where $0 < D < 0.5$. Long-memory processes are found in many fields, such as hydrology [17], or even in economics [18].

Most econometric models involve indicators or variables that affect economic forecasts. Exogenous variables, which are not affected by other variables in the system, are often used in econometric models. These include exchange rates, interest rates, inflation rates, etc., which can influence econometric models for forecasting future economic situations. Models with exogenous variables are often more accurate than those without them. When exogenous variables are included in long-memory SARFIMA$(P,D,Q)_L$, it becomes known as long-memory SARFIMAX$(P,D,Q,k)_L$. Monthly data in the context of seasonal fractional integration have been used [19, 20]. In the present work, we consider a long-memory process and chose the seasonal FIMAX model because it contains both a fractionally integrated and an MA component and includes exogenous variables. Hence, the effect of each type of parameter can be examined.

In process quality control, the most popular measure for the performance of a control chart is the average run length (ARL). The ARL computation for the CUSUM control chart has been achieved by applying various tools and techniques, such as the integral and Markov chain methods, among others. For example, Petcharat et al. [21] used explicit formulas via an integral equation (IE) method to derive the ARL for a CUSUM control chart running an MA process with exponential white noise. Recently, Phanthuna and Areepong [22] used the IE method to provide an exact solution for the ARL of a SAR$(P)_L$ process. Phanyaem [23] used the IE method to derive the ARL for a CUSUM control chart running a SARMA$(1,1)_L$ process. Finally, Suriyakat and Petcharat [24] derived explicit formulas for the ARL on an EWMA control chart running a stationary MAX process by using an IE. In addition, the existence and uniqueness of the ARL derived by using explicit formulas can be proved by using Banach's fixed-point theorem [25,26].

Consequently, the key aim of the present study is to derive the ARL for a CUSUM control chart running a long-memory seasonality fractionally integrated MA with an exogenous variable (SFIMAX) model with exponential white noise using explicit formulas. The rest of this research paper is organized as follows. A brief overview of the CUSUM control chart and the long-memory SFIMAX model is discussed in Section 2. In Sections 3, we provide solutions for the exact ARL in the above scenario using explicit formulas via an IE and an approximate solution using a numerical IE method, respectively. The numerical results of a comparison of the performances of both methods are reported in Section 4. In Section 5, a numerical example based on a real-life dataset is provided to illustrate the applicability of the proposed method. Conclusions of possible future work are summarized in Section 6. Finally, an appendix containing details of Banach's fixed-point theorem by verifying the existence and uniqueness of the ARL is provided.

2. The CUSUM Control Chart and the Process Model

This section provides a brief introduction to the CUSUM control chart and the long-memory SFIMAX model running on the control chart.

2.1. The CUSUM Control Chart

In this article, we focus on the one-sided upper CUSUM control chart with an upper control limit (UCL) for detecting shifts in the mean of a process [2]. Statistic S_t used for detecting upward shifts in the process mean on a CUSUM control chart can be expressed by using the following recursive equation:

$$S_t = \max\{S_{t-1} + Y_t - a,\ 0\}, \text{ for } t = 1,2,\dots, \qquad (1)$$

where Y_t is the sequence of a long-memory SFIMAX$(D,Q,k)_L$ process with exponential white noise, predetermined target or reference value a is a suitably chosen positive constant ($a \geq 0$), and S_0 is the starting value set between zero and *b*. $S_0 = \upsilon, (\upsilon \geq 0)$ must be satisfied for the CUSUM control chart [21].

The stopping time (τ_b) of the CUSUM control chart with a predetermined threshold of *b* (a constant parameter called the decision limit value or the upper control limit).

$$\tau_b = \inf\{t > 0;\ S_t > b\}, \text{ for } \upsilon < b. \qquad (2)$$

Note that a signal is given if S_t is above UCL (*b*) for detecting a shift from β_0 to β_1.

2.2. The Process Model

The seasonal FIMAX model was chosen because it contains both an FI and an MA component along with exogenous variables. Hence, the effect of each type of parameter can be examined. For the study, we assume that only one observation is available in each time period.

Seasonal time series can be analyzed by interweaving seasonal fluctuations in ARFIMA models in the form of parameters $(P,D,Q)_L$ Such models are represented as SARFIMA$(P,D,Q)_L$, where *P* and *Q* are the orders of the seasonal AR and seasonal MA components, respectively, while *D* represents the degree of seasonal fractional integration and *L* denotes the number of time periods in a year (e.g., 12 for monthly data). These models can be represented by a seasonal time series (Y_t) as follows:

$$\tau_b = \inf\{t > 0;\ S_t > b\}, \text{ for } \upsilon < b, \qquad (3)$$

where μ is a constant term for the time series, ε_t is a gaussian white noise process with mean 0 and variance σ_ε^2, B refers to the backward-shift operator where $B^{rL}Y_t = Y_{t-rL}, r \geq 0,$ and seasonal difference operator $(1-B^L)$ is raised to fractional power D, denoting the fractional order of integration. $\Phi_P(B^L)$ and $\Theta_Q(B^L)$ in the above model can be respectively expressed as

$$\Phi_P(B^L) = 1 - \Phi_1 B^L - \Phi_2 B^{2L} - \ldots - \Phi_P B^{PL}$$

$$\text{and } \Theta_Q(B^L) = 1 - \Theta_1 B^L - \Theta_2 B^{2L} - \ldots - \Theta_P B^{QL},$$

where Φ_i is the *i*-th seasonal autoregressive coefficient and Θ_j is the *j*-th seasonal MA coefficient.

The seasonal fractional differencing operator $(1-B^L)^D$ is defined by the binomial expansion as follows:

$$(1-B^L)^D = \sum_{r\geq 0} \binom{D}{r} (-1)^r B^{rL}$$

$$= 1 - DB^L + \frac{D(D-1)}{2!} B^{2L} - \frac{D(D-1)(D-2)}{3!} B^{2L} + \ldots, \qquad (4)$$

where differencing parameter D is non-integer in the range $-0.5 < D < 0.5,$ where $D < 0.5$ is stationarity condition and $D > -0.5$ is the invertibility condition: $-0.5 < D < 0$ indicates an intermediate-memory process, $D = 0$ indicates a short-memory (or short-range dependent) process equivalent to the classical ARIMA model [27], and $0 < D < 0.5$ indicates a long-memory (or long-range dependent) process.

A SARFIMA model can have an incorporated exogenous variable that affects structural changes in the analytical process. Therefore, the SARFIMAX$(P,D,Q,k)_L$. model can be expressed as

$$\Phi_P(B^L)(1-B^L)^D(Y_t - \mu) = \Theta_Q(B^L)\varepsilon_t + \sum_{j=1}^{k} \omega_j X_{jt}, \qquad (5)$$

where $X_{jt}, l = 1,2,\ldots,k$ are exogenous variables; *t* is the time; and $\omega_j, j = 1,2,\ldots,k$ are the coefficients corresponding to the k exogenous variables.

In this work, the SARFIMAX$(P,D,Q,k)_L$ model is restricted by imposing $P = 0$, such that

$$(1-B^L)^D(Y_t - \mu) = \Theta_Q(B^L)\varepsilon_t + \sum_{j=1}^{k} \omega_j X_{jt}. \qquad (6)$$

Subsequently, by substituting $(1-B^L)^D$ from (4) into (6), then the general form of the SFIMAX$(D,Q,k)_L$ model becomes

$$\begin{aligned} Y_t &= \mu + \varepsilon_t - \Theta_1\varepsilon_{t-L} - \Theta_2\varepsilon_{t-2L} - \ldots - \Theta_Q\varepsilon_{t-QL} \\ &\quad + \omega_1 X_{1t} + \omega_2 X_{2t} + \ldots + \omega_k X_{kt} \\ &\quad + DY_{t-L} - \frac{D(D-1)}{2} Y_{t-2L} + \frac{D(D-1)(D-2)}{6} Y_{t-2L} - \ldots, \end{aligned} \qquad (7)$$

where ε_t is exponential white noise ($\varepsilon_t \sim Exp(\beta)$); $\Theta_i, i = 1,2,\ldots,Q$ are the coefficients for the MA component ($-1 \leq \Theta_i \leq 1$); $\omega_j, j = 1,2,\ldots,k$ correspond to the k exogenous variables ($-1 \leq \omega_j \leq 1$); and the FI parameters are all nonzero. The initial values of $Y_{t-1}, Y_{t-2}, \ldots, X_{1t}, X_{2t}, \ldots, X_{rt}$ and ε_t are equal to 1. In this work, special attention is paid to a long-memory process with the SFIMAX model running on a CUSUM control chart defined as $0 < D < 0.5$.

The exact and approximate solutions for the ARL for detecting a shift in the mean shift of long-memory SFIMAX$(D,Q,k)_L$ process running on a CUSUM control chart are discussed in the next section.

3. ARL Derivations for a SFIMAX$(D,Q,k)_L$ Process Running on a CUSUM Control Chart

Several researchers have computed the ARL for a shift in the process mean on a CUSUM control chart by applying various tools and techniques [28,29]. The present work involves deriving the exact ARL based on explicit formulas and the approximate ARL based on the NIE method. Initially, the ARL calculation for an upper-sided CUSUM control chart running a long-memory SFIMAX$(D, Q, k)_L$ process is derived by rearranging (1) and (7) as

$$S_t = \max\left\{\begin{array}{l} S_{t-1}+\mu+\varepsilon_t-\Theta_1\varepsilon_{t-L}-\Theta_2\varepsilon_{t-2L}-... \\ -\Theta_Q\varepsilon_{t-QL}+\omega_1X_{1t}+\omega_2X_{2t}+...+\omega_kX_{kt} \\ +DY_{t-L}-\frac{D(D-1)}{2}Y_{t-2L} \\ +\frac{D(D-1)(D-2)}{6}Y_{t-2L}-...-a,\ 0 \end{array}\right\}, \quad (8)$$

where $\varepsilon_t \sim Exp(\beta)$.

Let $L(\upsilon)=\mathrm{E}_m(\tau_b)<\infty$ represent the ARL for an upper-sided CUSUM control chart with initialization $S_0=\upsilon$, and denote the probability measure and expectation corresponding to the initial value υ by $\mathbb{P}_S$ and $\mathbb{E}_S$, respectively, then the IE can be written in the form

$$L(\upsilon)=1+\mathbb{P}_s\{S_1=0\}L(0)+\mathbb{E}_s[I\{0<S_1<b\}L(S_1)], \quad (9)$$

where $I\{0<S_1<b\}$ is an indicator function. The ARL of the upper-sided CUSUM control chart in (8) and the IE in (9) derived from the Fredholm IE of the second kind can be written in the form

$$L(\upsilon)=1+F\left(a-\upsilon-Y_t\right)L(0)+\int_0^b L(g)f(a-\psi-Y_t+g)dg, \quad (10)$$

where $f(.)$ and $F(.)$ are the probability density function (pdf) and cumulative distribution function (cdf) of an exponential distribution, respectively, and $L(0)$ is a test starting at zero. Clearly, from (10), $L(\upsilon)$ must be known for υ, where $0\le\upsilon\le b$, , to find $L(0)$. Therefore, the equation (10) becomes

$$L(\upsilon)=1+(1-\exp\left\{\begin{array}{l} -\beta\big(a-\upsilon-\mu+\Theta_1\varepsilon_{t-L} \\ +\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL} \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L} \\ -\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ -\omega_1X_{1t}-\omega_2X_{2t}-...-\omega_kX_{kt}\big) \end{array}\right\})L(0)$$

$$+\beta(\exp\left\{\begin{array}{l} \beta\big(\upsilon-a+\mu-\Theta_1\varepsilon_{t-L}-\Theta_2\varepsilon_{t-2L}-...-\Theta_P\varepsilon_{t-QLkt} \\ +DY_{t-L}-\frac{D(D-1)}{2}Y_{t-2L}+\frac{D(D-1)(D-2)}{6}Y_{t-2L}-... \\ +\omega_1X_{1t}+\omega_2X_{2t}+...+\omega_kX\big) \end{array}\right\}) \times\int_0^b L(g)\exp\{-\beta g\}dg. \quad (11)$$

3.1. The Exact ARL Based on Explicit Formulas

The explicit formulas for studying the ARL of a CUSUM control chart running a long-memory process with exponential white noise are based on IEs. $(L(\upsilon))$ is derived as in (11) by defining $h=\int_0^b L(g)\exp\{-\beta g\}dg$, we can rearrange the result as follows:

$$L(\upsilon)=1+(1-\exp\left\{\begin{array}{l} -\beta\big(a-\upsilon-\mu_t+\Theta_1\varepsilon_{t-L} \\ +\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL} \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L} \\ -\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ -\omega_1X_{1t}-\omega_2X_{2t}-...-\omega_kX_{kt}\big) \end{array}\right\})L(0)$$

$$+h\beta(\exp\left\{\begin{array}{l} \beta\big(\upsilon-a+\mu-\Theta_1\varepsilon_{t-L} \\ -\Theta_2\varepsilon_{t-2L}-...-\Theta_P\varepsilon_{t-QLkt} \\ +DY_{t-L}-\frac{D(D-1)}{2}Y_{t-2L} \\ +\frac{D(D-1)(D-2)}{6}Y_{t-2L}-... \\ +\omega_1X_{1t}+\omega_2X_{2t}+...+\omega_kX\big) \end{array}\right\}). \quad (12)$$

After that, substitute $\upsilon=0$ in (12), we get

$$L(0)=1+(1-\exp\left\{\begin{array}{l} -\beta\big(a-\mu+\Theta_1\varepsilon_{t-L} \\ +\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL} \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L} \\ -\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ -\omega_1X_{1t}-\omega_2X_{2t}-...-\omega_kX_{kt}\big) \end{array}\right\})L(0)$$

$$+h\beta(\exp\left\{\begin{array}{l} \beta\big(\mu-a-\Theta_1\varepsilon_{t-L}-\Theta_2\varepsilon_{t-2L} \\ -...-\Theta_P\varepsilon_{t-QLkt}+DY_{t-L}-\frac{D(D-1)}{2}Y_{t-2L} \\ +\frac{D(D-1)(D-2)}{6}Y_{t-2L}-... \\ +\omega_1X_{1t}+\omega_2X_{2t}+...+\omega_kX\big) \end{array}\right\}).$$

$$L(0)=\exp\left\{\begin{array}{l} \beta\big(a-\mu+\Theta_1\varepsilon_{t-L}+\Theta_2\varepsilon_{t-2L} \\ +...+\Theta_P\varepsilon_{t-QL}-DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L} \\ -\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ -\omega_1X_{1t}-\omega_2X_{2t}-...-\omega_kX_{kt}\big) \end{array}\right\}+h\beta. \quad (13)$$

Subsequently, substitute $L(0)$ from (13) into (12), we get that

$$L(\upsilon)=1+h\beta-\exp\{\beta\upsilon\}$$

$$+\exp\left\{\begin{array}{l} \beta\big(a-\mu+\Theta_1\varepsilon_{t-L}+\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL} \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L}-\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ -\omega_1X_{1t}-\omega_2X_{2t}-...-\omega_kX_{kt}\big) \end{array}\right\}. \quad (14)$$

Subsequently, constant h can be defined as

$$h=\int_0^b L(g)\exp\{-\beta g\}dg$$

$$\therefore h = \frac{\exp\{\beta b\}}{\beta}\left(1-\exp\{-\beta b\}\right) \times \left(1+\exp\begin{Bmatrix} \beta\left(a-\mu+\Theta_1\varepsilon_{t-L}+\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL}\right. \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L}-\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ \left.-\omega_1 X_{1t}-\omega_2 X_{2t}-...-\omega_k X_{kt}\right) \end{Bmatrix}\right) - b\exp\{\beta b\}\Big). \quad (15)$$

By substituting constant h into (14), we obtain

$$L(\upsilon) = \exp\{\beta b\}\left(1+\exp\begin{Bmatrix} \beta\left(a-\mu+\Theta_1\varepsilon_{t-L}+\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL}\right. \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L}-\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ \left.-\omega_1 X_{1t}-\omega_2 X_{2t}-...-\omega_k X_{kt}\right) \end{Bmatrix} -\beta b\right) - \exp\{\beta\upsilon\};\ \upsilon \geq 0. \quad (16)$$

Therefore, the proof is complete.

The ARL measures how quickly a control chart responds to changes in the process mean. $L(\upsilon)$ in (16) is equivalent to ARL$_1$ (the out-of-control situation) depending on exponential parameter $(\beta = \beta_1)$, where $\beta_1 = \beta_0(1+\delta)$. The shift in the process mean is determined by (δ) and $\beta_0 = 1$. This situation shows that the calculation scheme can be completed in one stage as follows:

$$\text{ARL}_1 = \exp\{\beta_1 b\}\left(1+\exp\begin{Bmatrix} \beta_1\left(a-\mu+\Theta_1\varepsilon_{t-L}+\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL}\right. \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L}-\frac{D(D-1)(D-2)}{6}Y_{t-2L} \\ \left.+...-\omega_1 X_{1t}-\omega_2 X_{2t}-...-\omega_k X_{kt}\right) \end{Bmatrix} -\beta_1 b\right) - \exp\{\beta_1\upsilon\};\ \upsilon \geq 0. \quad (17)$$

This is the exact ARL derived by using explicit formulas. Subsequently, Banach's fixed-point theorem was applied to verify the existence and uniqueness of the explicit formulas (see Appendix). The results indicate that the ARL based on the explicit formulas for the CUSUM control chart running a long-memory SFIMAX process with exponential white noise exists and is unique.

3.2. The Approximate ARL Based on the NIE Method

The approximate ARL under an IE set via the NIE method is evaluated by using (11). In this section, the NIE method is used to evaluate the solution by using the Gauss-Legendre quadrature rule. This method was used to verify the accuracy of the exact ARL derived by using explicit formulas. Therefore, the approximate ARL based on the NIE method represented by $L_{NIE}(\upsilon)$ can be written as

$$L_{NIE}(\upsilon) = 1 + L_{NIE}(u_1)F\begin{pmatrix} a-\upsilon-\mu+\Theta_1\varepsilon_{t-L}+\Theta_2\varepsilon_{t-2L} \\ +...+\Theta_P\varepsilon_{t-QL}-DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L} \\ -\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ -\omega_1 X_{1t}-\omega_2 X_{2t}-...-\omega_k X_{kt} \end{pmatrix} + \sum_{j=1}^{m} w_j L_{NIE}(u_j) f\begin{pmatrix} a+u_j-\upsilon-\mu+\Theta_1\varepsilon_{t-L}+\Theta_2\varepsilon_{t-2L}+...+\Theta_P\varepsilon_{t-QL} \\ -DY_{t-L}+\frac{D(D-1)}{2}Y_{t-2L}-\frac{D(D-1)(D-2)}{6}Y_{t-2L}+... \\ -\omega_1 X_{1t}-\omega_2 X_{2t}-...-\omega_k X_{kt} \end{pmatrix}, \quad (18)$$

with $w_j = \frac{b}{m}$, and $u_j = \frac{b}{m}\left(j-\frac{1}{2}\right)$; $j = 1,2,\ldots,m.$

Details of the calculation and the results can be found in [22,30].

4. Performance Evaluation and Comparison

In this section, we describe the proposed exact ARL to compare with the approximate ARL on a CUSUM control chart running a long-memory SFIMAX$(D,Q,k)_L$ model. The ARL values on the CUSUM control chart are computed numerically by using the explicit formula and NIE methods with the codes for the Mathematica programming suite. An accurate assessment of the approximate ARL obtained by using the NIE method can be achieved with 800 division points. To verify the accuracy of the developed explicit formulas against the NIE method was evaluated in terms of the percentage accuracy and the computational time. The former is calculated as follows:

$$\%\,\text{Accuracy} = 100 - \left| \frac{L(\upsilon) - L_{NIE}(\upsilon)}{L(\upsilon)} \right| \times 100\%, \qquad (19)$$

where $L(\upsilon)$ and $L_{NIE}(\upsilon)$ are the ARL values obtained by using the explicit formulas and NIE method, respectively.

Because the one-sided upper CUSUM structure has only two design parameters: a and b, the value of a is first predefined as 3.5 while the value of b is calculated by using (16) and the predefined value of ARL_0. In this study, ARL_0 was set as 370 or 500, which are standard values [31]. For the in-control process, the ARL_0 value associated with no shift in the mean was determined for exponential white noise parameter β_0 = 1. Shifts in the process mean ($\beta_1 = \beta_0(1+\delta)$) were obtained for δ = 0.01, 0.05, 0.25, 0.50, 0.75, 1.00, or 2.00.

The results for calculating control limit b of the CUSUM control chart (with a = 3.5) when there is no shift in the process mean of the observations from a long-memory SFIMAX$(0.1,1,1)_{12}$, SFIMAX$(0.2,1,1)_{12}$, or SFIMAX$(0.4,1,1)_{12}$ for MA $\Theta_1 = \pm 0.1$, ± 0.5, ± 0.9 and $\omega_1 = 0.1$ are provided in Table 1. For example, the in-control process with design parameters a = 3.5 and b = 2.778292 for SFIMAX$(0.1,1,1)_{12}$ provided ARL_0 = 370 while a = 3.5 and b = 3.091097 for SFIMAX$(0.1,1,1)_{12}$ yielded ARL_0 = 500.

Table 1. Calculated control limit b values with corresponding reference parameter a=3.5 for various long-memory SFIMAX$(D,Q,k)_L$ processes running on a CUSUM control chart.

Coefficients of Long-memory		ARL_0 = 370			ARL_0 = 500		
		SFIMAX$(D, 1, 1)_{12}$			SFIMAX$(D, 1, 1)_{12}$		
ω_1	Θ_1	0.1	0.2	0.4	0.1	0.2	0.4
0.1	0.9	2.778292	2.954750	3.261840	3.091097	3.270346	3.583850
	0.5	2.823470	3.001381	3.312010	3.136937	3.317811	3.635306
	0.1	2.868985	3.048457	3.362916	3.183157	3.365778	3.687598
	-0.1	2.891880	3.072174	3.388665	3.206416	3.389960	3.714080
	-0.5	2.937951	3.119986	3.440800	3.253256	3.438750	3.767770
	-0.9	2.984427	3.168334	3.493850	3.300550	3.488146	3.822510

Table 2. The numerical results of using explicit formulas and the NIE method to calculate ARL_1 for a long-memory SFIMAX$(0.1,1,1)_{12}$ process with various choices of Θ_1

ARL_0	Coefficients of process			Methods	Shift size (δ) with $\beta_0 = 1$						
	D	Θ_1	ω_1		0.01	0.05	0.25	0.50	0.75	1.00	2.00
370	0.1	0.9	0.01	Explicit	348.106	275.871	107.493	47.030	26.096	16.824	6.167
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.644	275.517	107.347	46.989	26.074	16.811	6.165
				CPU_{NIE}	(1,350)	(1,353)	(1,356)	(1,389)	(1,350)	(1,358)	(1,356)
				%Acc	99.87	99.87	99.86	99.91	99.92	99.92	99.97
		0.5	0.01	Explicit	348.063	275.709	107.234	46.854	25.982	16.747	6.144
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.496	275.28	107.058	46.806	25.957	16.732	6.142
				CPU_{NIE}	(1,345)	(1,356)	(1,365)	(1,349)	(1,365)	(1,395)	(1,356)
				%Acc	99.84	99.84	99.84	99.90	99.90	99.91	99.97

Table 2 Continued

		0.1	0.01	Explicit	348.018	275.538	106.963	46.671	25.864	16.668	6.121
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.556	275.184	106.817	46.63	25.842	16.655	6.119
				CPU_{NIE}	(1,330)	(1,340)	(1,341)	(1,353)	(1,350)	(1,353)	(1,359)
				%Acc	99.87	99.87	99.86	99.91	99.91	99.92	99.97
		-0.1	0.01	Explicit	347.996	275.449	106.823	46.577	25.804	16.628	6.109
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.429	275.02	106.647	46.529	25.779	16.613	6.107
				CPU_{NIE}	(1,348)	(1,341)	(1,348)	(1,342)	(1,350)	(1,340)	(1,365)
				%Acc	99.84	99.84	99.84	99.90	99.90	99.91	99.97
		-0.5	0.01	Explicit	347.947	275.266	106.533	46.383	25.679	16.545	6.084
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.28	274.768	106.338	46.333	25.654	16.53	6.082
				CPU_{NIE}	(1,343)	(1,351)	(1,353)	(1,349)	(1,346)	(1,348)	(1,345)
				%Acc	99.81	99.82	99.82	99.89	99.90	99.91	99.97
		-0.9	0.01	Explicit	347.896	275.073	106.231	46.181	25.551	16.459	6.059
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.191	274.55	106.032	46.133	25.527	16.445	6.057
				CPU_{NIE}	(1,351)	(1,349)	(1,346)	(1,350)	(1,345)	(1,343)	(1,346)
				%Acc	99.80	99.81	99.81	99.90	99.91	99.91	99.97
500	0.1	0.9	0.01	Explicit	468.846	366.852	135.723	56.749	30.504	19.217	6.706
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	468.079	366.274	135.494	56.69	30.473	19.199	6.703
				CPU_{NIE}	(1,348)	(1,347)	(1,350)	(1,345)	(1,351)	(1,352)	(1,349)
				%Acc	99.84	99.84	99.83	99.90	99.90	99.91	99.96
		0.5	0.01	Explicit	468.779	366.605	135.347	56.504	30.351	19.116	6.677
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.867	365.927	135.086	56.439	30.322	19.097	6.674
				CPU_{NIE}	(1,353)	(1,351)	(1,347)	(1,350)	(1,350)	(1,347)	(1,344)
				%Acc	99.81	99.82	99.81	99.88	99.90	99.90	99.96
		0.1	0.01	Explicit	468.712	366.346	134.955	56.250	30.192	19.012	6.648
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	468.045	365.848	134.76	56.2	30.167	18.997	6.646
				CPU_{NIE}	(1,346)	(1,350)	(1,343)	(1,342)	(1,347)	(1,348)	(1,344)
				%Acc	99.86	99.86	99.86	99.91	99.92	99.92	99.97
		-0.1	0.01	Explicit	468.675	366.211	134.752	56.119	30.110	18.959	6.633
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.822	365.574	134.504	56.056	30.078	18.94	7.08
				CPU_{NIE}	(1,340)	(1,346)	(1,350)	(1,344)	(1,351)	(1,349)	(1,352)
				%Acc	99.82	99.83	99.82	99.89	99.89	99.90	93.26
		-0.5	0.01	Explicit	468.601	365.932	134.332	55.850	29.943	18.850	6.602
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.689	365.254	134.071	55.785	29.914	18.831	6.599
				CPU_{NIE}	(1,346)	(1,345)	(1,351)	(1,347)	(1,344)	(1,347)	(1,350)
				%Acc	99.81	99.81	99.81	99.88	99.90	99.90	99.95
		-0.9	0.01	Explicit	468.523	365.638	133.894	55.569	29.769	18.737	6.571
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.513	364.896	133.622	55.505	29.738	18.719	6.568
				CPU_{NIE}	(1,344)	(1,345)	(1,346)	(1,350)	(1,349)	(1,346)	(1,348)
				%Acc	99.78	99.80	99.80	99.88	99.90	99.90	99.95

Note: The numerical results in parentheses are computational times in seconds

Table 3. The numerical results of using explicit formulas and the NIE method to calculate ARL_1 for a long-memory SFIMAX$(0.2,1,1)_{12}$ process process with various choices of Θ_1

ARL_0	Coefficients of process			Methods	Shift size (δ) with $\beta_0 = 1$						
	D	Θ_1	ω_1		0.01	0.05	0.25	0.50	0.75	1.00	2.00
370	0.2	0.9	0.01	Explicit	347.929	275.197	106.425	46.310	25.633	16.514	6.075
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.362	274.768	106.249	46.262	25.608	16.499	6.072
				CPU_{NIE}	(1,343)	(1,346)	(1,344)	(1,349)	(1,352)	(1,351)	(1,354)
				%Acc	99.84	99.84	99.83	99.90	99.90	99.91	99.95
		0.5	0.01	Explicit	347.877	275.001	106.117	46.106	25.503	16.482	6.049
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.415	274.647	105.971	46.065	25.48	16.468	6.047
				CPU_{NIE}	(1,346)	(1,345)	(1,343)	(1,351)	(1,347)	(1,349)	(1,352)
				%Acc	99.87	99.87	99.86	99.91	99.91	99.92	99.97
		0.1	0.01	Explicit	347.822	274.794	105.795	45.892	25.367	16.338	6.024
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.255	274.365	105.619	45.844	25.343	16.323	6.022
				CPU_{NIE}	(1,343)	(1,345)	(1,344)	(1,345)	(1,349)	(1,350)	(1,348)
				%Acc	99.84	99.84	99.83	99.90	99.91	99.91	99.97
		-0.1	0.01	Explicit	347.794	274.686	105.629	45.782	25.298	16.292	6.011
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.332	274.332	105.483	45.741	25.276	16.279	6.009
				CPU_{NIE}	(1,344)	(1,346)	(1,342)	(1,341)	(1,348)	(1,349)	(1,348)
				%Acc	99.87	99.87	99.86	99.91	99.91	99.92	99.97
		-0.5	0.01	Explicit	347.735	274.463	105.284	45.556	25.155	16.198	5.984
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.168	274.034	105.108	45.508	25.13	16.183	5.982
				CPU_{NIE}	(1,349)	(1,349)	(1,345)	(1,344)	(1,342)	(1,344)	(1,342)
				%Acc	99.84	99.84	99.83	99.89	99.90	99.91	99.97
		-0.9	0.01	Explicit	347.672	274.227	104.922	45.319	25.001	16.101	5.957
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	347.062	273.766	104.737	45.27	24.976	16.086	5.954
				CPU_{NIE}	(1,342)	(1,347)	(1,346)	(1,345)	(1,343)	(1,349)	(1,350)
				%Acc	99.82	99.83	99.82	99.89	99.90	99.91	99.95

Table 3 Continued

500	0.2	0.9	0.01	Explicit	468.573	365.827	134.176	55.749	29.880	18.809	6.591
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.666	365.152	133.915	55.685	29.847	18.791	6.587
				CPU_{NIE}	(1,351)	(1,350)	(1,350)	(1,345)	(1,341)	(1,341)	(1,350)
				%Acc	99.81	99.82	99.81	99.89	99.89	99.90	99.94
		0.5	0.01	Explicit	468.493	365.528	133.729	55.464	29.704	18.695	6.560
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.452	364.769	133.456	55.402	29.674	18.679	6.558
				CPU_{NIE}	(1,344)	(1,347)	(1,344)	(1,341)	(1,346)	(1,348)	(1,349)
				%Acc	99.78	99.79	99.80	99.89	99.90	99.91	99.97
		0.1	0.01	Explicit	468.409	365.213	133.262	55.168	29.522	18.577	6.527
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.842	364.784	133.086	55.12	29.497	18.562	6.525
				CPU_{NIE}	(1,350)	(1,349)	(1,346)	(1,344)	(1,343)	(1,342)	(1,341)
				%Acc	99.88	99.88	99.87	99.91	99.92	99.92	99.97
		-0.1	0.01	Explicit	468.366	365.049	133.019	55.015	29.428	18.516	6.511
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.379	364.321	132.747	54.95	29.397	18.498	6.508
				CPU_{NIE}	(1,352)	(1,351)	(1,352)	(1,348)	(1,346)	(1,345)	(1,344)
				%Acc	99.79	99.80	99.80	99.88	99.89	99.90	99.95
		-0.5	0.01	Explicit	468.274	364.708	132.519	54.699	29.235	18.392	6.477
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.233	363.949	132.246	54.637	29.205	18.376	6.475
				CPU_{NIE}	(1,350)	(1,349)	(1,345)	(1,345)	(1,344)	(1,342)	(1,342)
				%Acc	99.78	99.79	99.79	99.89	99.90	99.91	99.97
		-0.9	0.01	Explicit	468.178	364.349	131.994	54.371	29.035	18.264	6.443
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.069	363.572	131.748	54.326	29.015	18.254	6.442
				CPU_{NIE}	(1,352)	(1,352)	(1,352)	(1,349)	(1,347)	(1,345)	(1,342)
				%Acc	99.76	99.79	99.81	99.92	99.93	99.95	99.98

Note: The numerical results in parentheses are computational times in seconds

Table 4. The numerical results of using explicit formulas and the NIE method to calculate ARL_1 for a long-memory $SFIMAX(0.4,1,1)_{12}$ process with various choices of Θ_1

ARL_0	Coefficients of process			Methods	Shift size (δ) with $\beta_0 = 1$						
	D	Θ_1	ω_1		0.01	0.05	0.25	0.50	0.75	1.00	2.00
370	**0.4**	0.9	0.01	Explicit	347.545	273.744	104.186	44.842	24.709	15.907	5.904
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	346.878	273.246	103.991	44.792	24.684	15.892	5.902
				CPU_{NIE}	(1,346)	(1,343)	(1,345)	(1,348)	(1,347)	(1,344)	(1,345)
				%Acc	99.81	99.82	99.81	99.89	99.90	99.91	99.97
		0.5	0.01	Explicit	347.471	273.468	103.769	44.575	24.544	15.799	5.874
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	346.861	273.007	103.584	44.526	24.519	15.784	5.871
				CPU_{NIE}	(1,350)	(1,349)	(1,345)	(1,346)	(1,344)	(1,345)	(1,343)
				%Acc	99.82	99.83	99.82	99.89	99.90	99.91	99.95
		0.1	0.01	Explicit	347.393	273.177	103.333	44.296	24.372	15.688	5.844
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	346.748	272.693	103.142	44.247	24.346	15.673	5.842
				CPU_{NIE}	(1,346)	(1,345)	(1,345)	(1,346)	(1,342)	(1,345)	(1,346)
				%Acc	99.81	99.82	99.82	99.89	99.89	99.90	99.97
		-0.1	0.01	Explicit	347.353	273.025	103.106	44.152	24.283	15.631	5.829
				CPU_{Exp}	(0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	346.81	272.613	102.936	44.106	24.258	15.616	5.826
				CPU_{NIE}	(1,346)	(1,348)	(1,343)	(1,344)	(1,346)	(1,350)	(1,344)
				%Acc	99.84	99.85	99.84	99.90	99.90	99.90	99.95
		-0.5	0.01	Explicit	347.268	272.707	102.635	43.853	24.100	15.514	5.799
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	346.806	272.353	102.489	43.812	24.078	15.501	5.797
				CPU_{NIE}	(1,345)	(1,345)	(1,345)	(1,346)	(1,347)	(1,345)	(1,347)
				%Acc	99.87	99.87	99.86	99.91	99.91	99.92	99.97
		-0.9	0.01	Explicit	347.178	272.369	102.138	43.541	23.910	15.392	5.767
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	346.635	271.957	101.968	43.495	23.885	15.377	5.764
				CPU_{NIE}	(1,346)	(1,348)	(1,345)	(1,344)	(1,345)	(1,346)	(1,346)
				%Acc	99.84	99.85	99.83	99.89	99.90	99.90	99.95

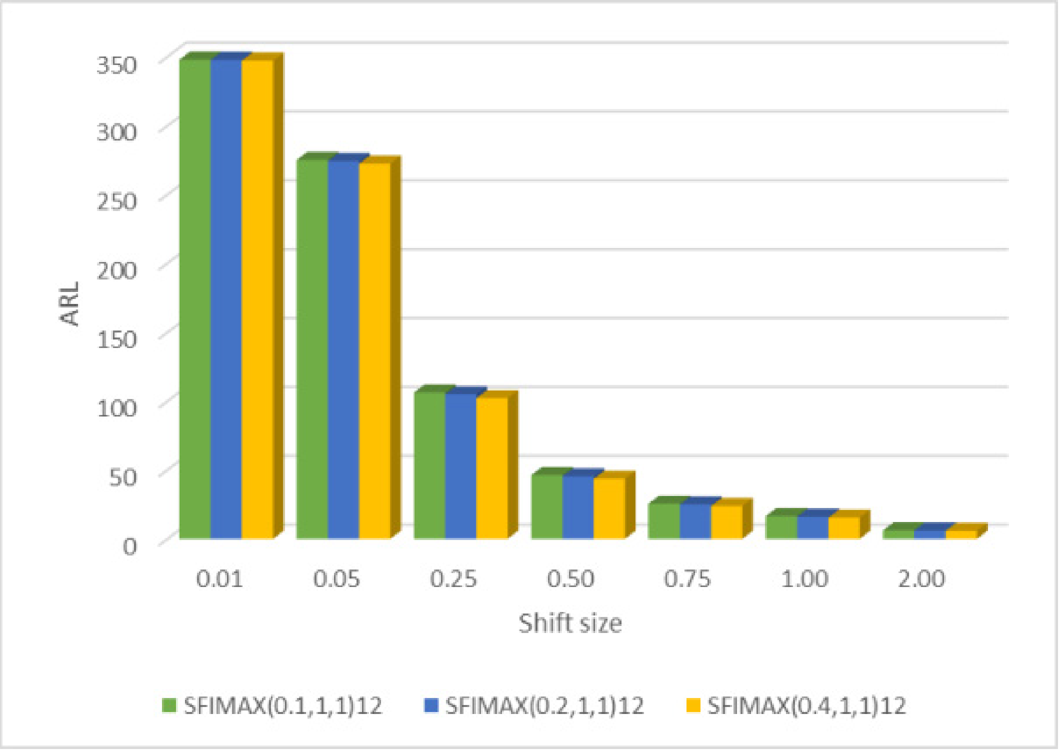
Table 4 Continued

500	0.4	0.9	0.01	Explicit	467.979	363.609	130.923	53.705	28.632	18.007	6.376
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	466.992	362.881	130.651	53.64	28.601	17.989	6.373
				CPU_{NIE}	(1,349)	(1,350)	(1,349)	(1,347)	(1,346)	(1,344)	(1,344)
				%Acc	99.79	99.80	99.79	99.88	99.89	99.90	99.95
		0.5	0.01	Explicit	467.866	363.188	130.317	53.332	28.408	17.865	6.340
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	467.013	362.551	130.069	53.269	28.376	17.846	6.337
				CPU_{NIE}	(1,352)	(1,350)	(1,350)	(1,345)	(1,348)	(1,344)	(1,346)
				%Acc	99.82	99.82	99.81	99.88	99.89	99.89	99.95
		0.1	0.01	Explicit	467.746	362.742	129.679	52.942	28.175	17.718	6.302
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	466.708	361.984	129.405	52.881	28.146	17.702	6.3
				CPU_{NIE}	(1,353)	(1,349)	(1,350)	(1,351)	(1,346)	(1,346)	(1,344)
				%Acc	99.78	99.79	99.79	99.88	99.90	99.91	99.97
		-0.1	0.01	Explicit	467.683	362.508	129.347	52.739	28.055	17.642	6.283
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	466.599	361.73	129.08	52.684	28.03	17.628	6.281
				CPU_{NIE}	(1,348)	(1,347)	(1,350)	(1,347)	(1,345)	(1,349)	(1,349)
				%Acc	99.77	99.79	99.79	99.90	99.91	99.92	99.97
		-0.5	0.01	Explicit	467.549	362.019	128.656	52.321	27.806	17.487	6.244
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	466.44	361.242	128.41	52.276	27.786	17.477	6.243
				CPU_{NIE}	(1,349)	(1,352)	(1,351)	(1,350)	(1,349)	(1,347)	(1,350)
				%Acc	99.76	99.79	99.81	99.91	99.93	99.94	99.98
		-0.9	0.01	Explicit	467.409	361.499	127.926	51.882	27.547	17.325	6.205
				CPU_{Exp}	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)	(<0.001)
				NIE	466.368	360.74	127.653	51.82	27.517	17.309	6.203
				CPU_{NIE}	(1,353)	(1,351)	(1,348)	(1,350)	(1,349)	(1,348)	(1,346)
				%Acc	99.78	99.79	99.79	99.88	99.89	99.91	99.97

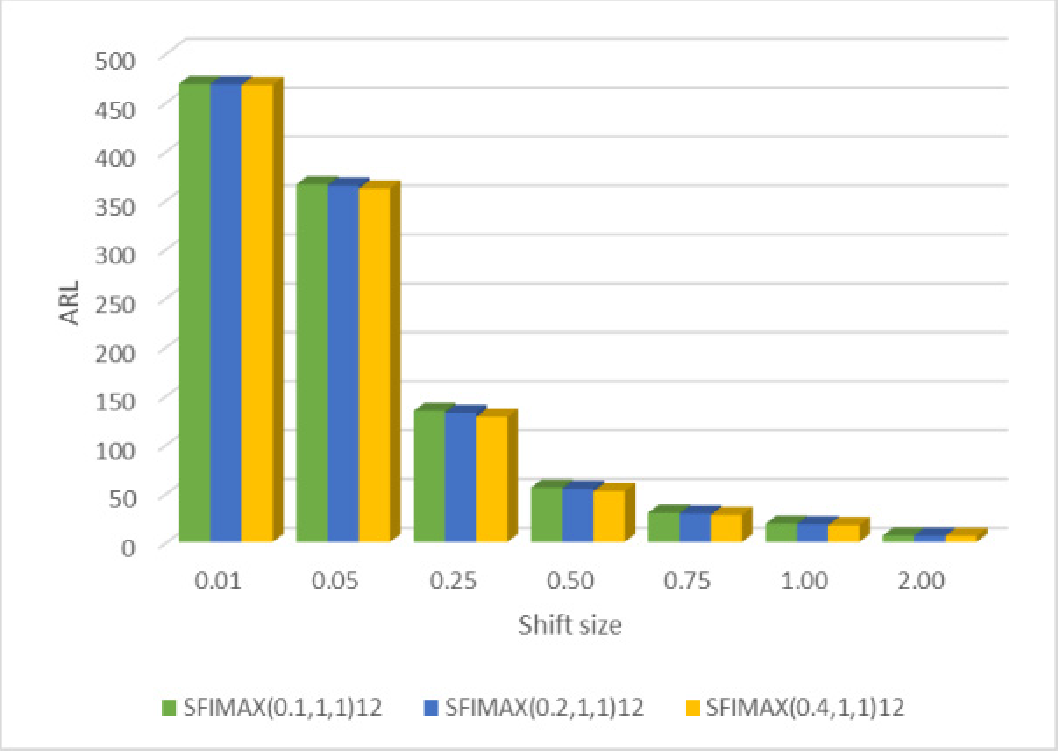
Note: The numerical results in parentheses are computational times in seconds

Comparison of the ARL performances for detecting shifts in the mean of a long-memory $SFIMAX(0.1,1,1)_{12}$, $SFIMAX(0.2,1,1)_{12}$, or $SFIMAX(0.4,1,1)_{12}$ process running on a CUSUM control chart is reported in Tables 2–4. The ARL_1 values were affected by control limit b because the out-of-control signals are obtained from the control limits (cf. Table 1). The ARL_1 values were also calculated after applying various shifts in the process mean (i.e., δ = 0.01 to 2.00). These three tables show that the lowest MA coefficient value resulted in the lowest ARL_1 value for all of the processes and magnitudes of process mean shift tested. For example, in Table 2, the ARL_1 values obtained by using the explicit formulas for long-memory $SFIMAX(0.1,1,1)_{12}$ with coefficients $D = 0.1$, $\Theta_1 = 0.9$, $\omega_1 = 0.1$ and δ = 0.01 yields ARL_1= 348.106, while varying MA: Θ_1 = 0.5, 0.1, -0.1, -0.5 or -0.9 provided ARL_1 = 348.063, 348.018, 347.996, 347.947 and 347.896, respectively, for the same shift size with ARL_0 = 370. Especially, MA = -0.9 provided the smallest ARL_1 value for each model and magnitude of process shift tested (Figure 1). This result is consistent with the numerical results of using explicit formulas in [24]. Moreover, the tables also show that ARL_1 decreased as the magnitude of the mean shift was increased. In particular, ARL_1 decreased sharply when the process mean shift size was small (δ = 0.01–0.50) to moderate (δ = 0.75–2.00). When considering the ARL_1 results obtained by using the NIE method for each of the long-memory processes and magnitudes of process mean shift tested, the ARL_1 results were in the same direction as those obtained by using the explicit formulas (cf. Tables 2–4). These comparisons show that, in all of the cases, the ARL_1 performance of the explicit formulas was slightly worse or very similar to that of the NIE method.

For clarity, we verified the accuracy of the developed explicit formulas against the NIE method as the percentage accuracy. Obviously, the percentage accuracy was greater than 99% for the mean process shifts in all of the models. The values in parentheses in each row are the computational times for calculating ARL_1. For $ARL_0 = 370$ or 500, the computational time for the NIE method was 1,300 – 1,400 seconds while that of the explicit formula was <0.001 seconds.



(A)



(B)

Figure 1. Out-of-control ARL values for a SFIMAX $(D,1,1)_{12}$ process running on a CUSUM control chart with $\Theta_1 = -0.9$ where $ARL_0 = 370$ in Part A and $ARL_0 = 500$ in Part B

The graphs of the ARL_1 values for the $SFIMAX(0.1,1,1)_{12}$, $SFIMAX(0.2,1,1)_{12}$, and $SFIMAX(0.4,1,1)_{12}$ models with the lowest MA coefficient value (Θ_1 = -0.9) and various process mean shift values for ARL_0 = 370 or 500 obtained by using explicit formulas are displayed in Figure 1, respectively. These results are consistent with those in [25]. Moreover, the figures also show that ARL_1 decreased as the magnitude of the process mean shift was increased. In particular, ARL_1 decreased sharply when the process mean shift was small and slightly less so for moderate shifts. Moreover, ARL_1 for detecting changes in the process mean tended to decrease when the process mean shift size was increased in the order of $SFIMAX(0.1,1,1)_{12}$ > $SFIMAX(0.2,1,1)_{12}$ > $SFIMAX(0.4,1,1)_{12}$. Therefore, the ARL on a CUSUM control chart obtained by using explicit formulas worked best for the long-memory $SFIMAX(0.4,1,1)_{12}$ process with white noise process following an exponential distribution for ARL_0 = 370 or 500.

To summarize the above discussion, the sensitivity of the CUSUM control chart running a long-memory $SFIMAX(D,Q,k)_L$ model is indicated by the ARL_1 results using explicit formulas to detect changes in the process mean. The sensitivity of the explicit formulas is a good alternative to the NIE method due to the significantly reduced computation time. Similar outcomes can also be produced by considering other reference values. For this reason, the explicit formula method is recommended for this scenario.

5. The Practicability of the Proposed Exact ARL Derived by Using Explicit Formulas

This was demonstrated by monthly observations of seasonally fitted real-life data. There are two case studies of processes running on a one-sided CUSUM control chart. The first case consists of 85 observations of the monthly export and import of agricultural products (per million baht) from January 2013 to January 2020 and the second case comprises 72 observations of the monthly value of rice exports and the Euro/Baht exchange rate from January 2012 to December 2017. Both datasets were downloaded from the Office of Agricultural Economics website (https://oae.go.th). The monthly exchange rate was the exogenous variable used in the study, data for which were downloaded from https://th.investing.com.

The correct and estimated model parameters were obtained by using the Eviews 10 statistical software package. In addition, the SPSS software package was used to test whether the residuals from both case studies were exponentially distributed.

Table 5. The coefficient values for the long-memory SFIMAX model used for case study 1

Case study 1: $SFIMAX(0.311179, 1, 1)_{12}$				
Variable	Coefficient	Std. Error	t-Statistic	P-value
D	0.311179	0.101812	3.056415	0.003*
MA(12)	0.558560	0.121233	4.607323	0.000*
IMPORT	2.629366	0.042013	62.58451	0.000*
One-sample Kolmogorov-Smirnov test				
Exponential parameter				8,807.2371
Kolmogorov-Smirnov Z				0.992
Asymp. Sig. (2-tailed)				0.279

*A significance level of 0.05.

The suitability of the dataset for case study 1 was tested. The t-test statistic values were 3.056415, 4.607323, and 62.58451 with p-values of 0.003, 0.000, and 0.000 for coefficients D = 0.311179, Θ_1 = 0.558560, and ω_1 = 2.629366, respectively. Furthermore, Table 5 reports that the white noise follows an exponential distribution (Kolmogorov-Smirnov test = 0.992 and p-value = 0.279) and was statistically nonsignificant (p-value < 0.05) with a mean of 8,807.2371. Hence, the fitted $SFIMAX(0.311179,1,1)_{12}$ model is

$$Y_t = \mu + \varepsilon_t - 0.558560\varepsilon_{t-12} + 0.311179Y_{t-12} + 0.107173Y_{t-24} + 0.0603322Y_{t-36} + 2.629366X_{1t}, \quad (20)$$

where $\varepsilon_t \sim Exp(\beta)$.

Hence, the exact ARL from (16) for case study 1 running the long-memory model from (20) on a CUSUM control chart is

$$L(\upsilon) = \exp\{\beta b\}\left(1 + \exp\begin{Bmatrix}\beta\big(a - \mu + 0.558560\varepsilon_{t-12} \\ -0.311179Y_{t-12} - 0.107173Y_{t-24} \\ -0.0603322Y_{t-36} - 2.629366X_{1t}\big)\end{Bmatrix} - \beta b\right) - \exp\{\beta\upsilon\}, \quad (21)$$

where $\beta = \beta_0 = 8{,}807.2371$ when the process is in-control (ARL_0), which was predefined as 370. When a = 3.0, initial upper bound $b = 13{,}493.42$ was computed by using (21). In addition, for the approximate ARL using (18), we obtain

$$L_{NIE}(\upsilon) = 1 + L_{NIE}(u_1)F\begin{pmatrix}a - \upsilon - \mu + 0.558560\varepsilon_{t-12} \\ -0.311179Y_{t-12} - 0.107173Y_{t-24} \\ -0.0603322Y_{t-36} - 2.629366X_{1t}\end{pmatrix}$$

$$+\sum_{j=1}^{m} w_j L_{NIE}(u_j) f\begin{pmatrix}a + u_j - \upsilon - \mu + 0.558560\varepsilon_{t-12} \\ -0.311179Y_{t-12} - 0.107173Y_{t-24} \\ -0.0603322Y_{t-36} - 2.629366X_{1t}\end{pmatrix}.$$

Table 6. The coefficient values for the long-memory SFIMAX model used for case study 2

Case study 2: SFIMAX(0.389213, 1, 1)$_{12}$				
Variable	Coefficient	Std. Error	t-Statistic	P-value
D	0.389213	0.172045	2.262277	0.0268*
SMA(12)	0.449405	0.140000	3.210033	0.0020*
EUR/THB	34366383	690878.6	49.74302	0.0000*
One-sample Kolmogorov-Smirnov test				
Exponential parameter				132,271,702.9518
Kolmogorov-Smirnov Z				1.045
Asymp. Sig. (2-tailed)				0.224

*A significance level of 0.05.

From Table 6, the data for case study 2 fit a long-memory SFIMAX(0.389213,1,1)$_{12}$ model since all of the parameters were statistically significant (p-value < 0.05). Therefore, the coefficient values for the model were $D = 0.389213$, $\Theta_1 = 0.449405$ and $\omega_1 = 34366383$. Clearly, the white noise was exponentially distributed (Kolmogorov-Smirnov test = 1.045 and p-value = 0.224) and was statistically nonsignificant (p-value < 0.05) with a mean of $132,271,702.9518$. The exchange rate impacted monthly rice exports, with the general form of the model on the CUSUM control chart being

$$Y_t = \mu + \varepsilon_t - 0.449405\varepsilon_{t-12} + 0.389213Y_{t-12} + 0.118863Y_{t-24} + 0.063821Y_{t-36} + 34,366,383X_{1t}, \quad (22)$$

where $\varepsilon_t \sim Exp(132,271,702.9518)$. Moreover, we used 132,271,702.9518 for the in-control process. Hence, the exact ARL for case study 2 taking the long-memory model from (22) is

$$L(\upsilon) = \exp\{\beta b\}\left(1 + \exp\left\{\begin{matrix}\beta(a - \mu + 0.449405\varepsilon_{t-12} \\ -0.389213Y_{t-12} - 0.118863Y_{t-24} \\ -0.063821Y_{t-36} - 34,366,383X_{1t})\end{matrix}\right\} - \beta b\right) - \exp\{\beta\upsilon\}, \quad (23)$$

and approximated the ARL from (18) is

$$L_{NIE}(\upsilon) = 1 + L_{NIE}(u_1)F\begin{pmatrix}a - \upsilon - \mu + 0.449405\varepsilon_{t-12} \\ -0.389213Y_{t-12} - 0.118863Y_{t-24} \\ -0.063821Y_{t-36} - 34,366,383X_{1t}\end{pmatrix} + \sum_{j=1}^{m} w_j L_{NIE}(u_j) f\begin{pmatrix}a + u_j - \upsilon - \mu + 0.449405\varepsilon_{t-12} \\ -0.389213Y_{t-12} - 0.118863Y_{t-24} \\ -0.063821Y_{t-36} - 34,366,383X_{1t}\end{pmatrix},$$

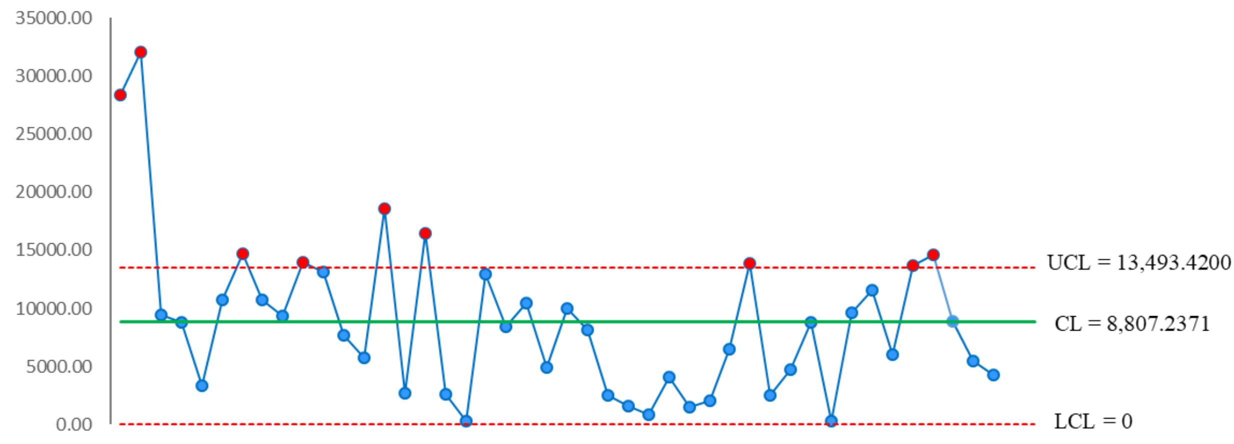
for a = 3.0, ARL$_0$ = 370, and $b = 163,009,100.00$.

In addition, equations (21) and (23) define $\beta = \beta_1$ for the out-of-control process for δ = 0.01, 0.05, 0.25, 0.50, 0.075, 1.00, or 2.00. The numerical results for the explicit formulas and NIE methods for the out-of-control ARL for detecting shifts in the process mean in terms of percentage accuracy and computational time are reported in Table 7. They are clearly consistent with those in Tables 2–4 and confirm that the proposed explicit formulas are sensitive to shifts in the process mean and are as good at detecting them as the NIE method. Moreover, it is easier to calculate the ARL with them than with the NIE method. This means that using explicit formulas for calculating the ARL is a good alternative to the NIE method for measuring the performance of a process on a CUSUM control chart for both of the scenarios studied.

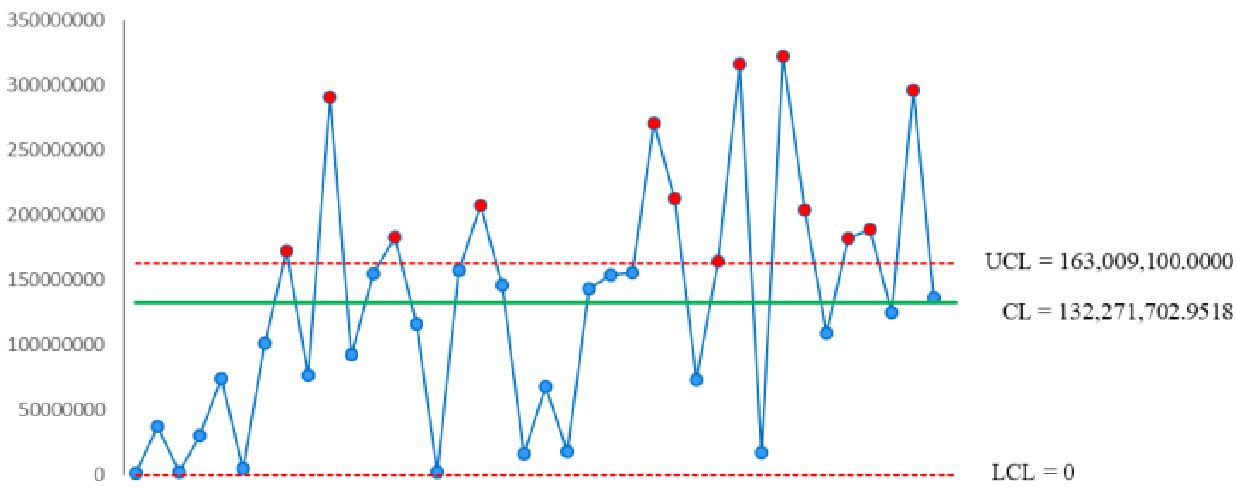
Table 7. The numerical results of explicit formulas and NIE method with real datasets for ARL$_0$ = 370.

Shift size (δ)	Case study 1					Case study 2				
	$a = 3.0, b = 13,493.42$				%Acc	$a = 3.0, b = 163,009,100.00$				%Acc
	Explicit		NIE			Explicit		NIE		
	ARL	CPU$_{Exp}$	ARL	CPU$_{NIE}$		ARL	CPU$_{Exp}$	ARL	CPU$_{NIE}$	
0.01	348.835	(<0.001)	348.510	(2036.47)	99.91	348.890	(<0.001)	348.627	(2057.81)	99.92
0.05	278.706	(<0.001)	278.457	(1974.72)	99.91	278.923	(<0.001)	278.721	(2087.81)	99.93
0.25	112.465	(<0.001)	112.382	(1930.80)	99.93	112.864	(<0.001)	112.797	(2086.94)	99.94
0.50	50.800	(<0.001)	50.770	(1938.76)	99.94	51.111	(<0.001)	51.0862	(2073.34)	99.95
0.75	28.801	(<0.001)	28.787	(1963.56)	99.95	29.025	(<0.001)	29.0133	(2009.84)	99.96
1.00	18.828	(<0.001)	18.821	(1912.14)	99.96	18.993	(<0.001)	18.9862	(2040.06)	99.96
2.00	7.017	(<0.001)	7.009	(1924.26)	99.89	7.079	(<0.001)	7.07776	(2013.94)	99.98

Note: The numerical results in parentheses are computational times in seconds.



(A)



(B)

Figure 2. The control limits for the proposed explicit formulas derivation of the ARL of real-data processes running on a CUSUM control chart for ARL_0= 370: (A) case study 1 and (B) case study 2

Figure 2 shows the performance of the CUSUM control chart proposed to have sensitivity to detect shifts in the process mean for this scenario. The process running on a CUSUM control chart in Figure 2(A) comprises a real-life dataset of monthly export and import figures of agricultural products; CUSUM statistic S_t is plotted on the control chart along with LCL = 0, CL = 8,807.2371, and UCL = 13,493.4200. The results show that the statistical values of the control chart can detect an abrupt shift in the process mean at the 1st observation. Moreover, Figure 2(B) shows monthly rice product export figures with the exchange rate as the exogenous variable, which provides LCL = 0, CL = 132,271,702.9518, and UCL = 163,009,100. The results show that the statistical values exceeded the bound for the first time at the 8th observation. It is evident that for any of the shifts in the process mean tested, the proposed explicit formulas solution for the ARL performed well for the detection of a shift in the process mean of a long-memory SFIMAX process running on a CUSUM control chart.

6. Conclusions

The performance of a CUSUM control chart running various long-memory SFIMAX$(D,Q,k)_L$ models with exponential white noise was evaluated in terms of the ARL derived by using explicit formulas, which was then compared with the approximate ARL derived by using the NIE method. The results reveal that the out-of-control ARL using explicit formulas performed very well for small-to-moderate shifts in the process mean in terms of percent accuracy and computational time. Therefore, it is plausible to use the explicit formulas as an alternative approach for deriving the ARL for a shift in the mean of a long-memory SFIMA process with exogenous variables running on a CUSUM control.

The major advantage of the proposed explicit formulas is their simplicity in design and ease of use, which were demonstrated by the numerical and real-life data examples presented herein. The scope of this study could be extended to other control charts such as EWMA and modified EWMA.

Appendix

Theorem 1 (Banach's fixed-point theorem).

Let $\mathcal{K}=(\mathcal{K},d)$ be a complete metric space, then mapping $T:\mathcal{K}\to\mathcal{K}$ is said to be a contraction mapping on $\mathcal{K}$ if there exists real number $\rho;\ 0\le\rho<1$ such that

$$d\left(T(L_1),T(L_2)\right)\le\rho d\left(L_1,L_2\right) \text{ for } L_1,\ L_2\in\mathcal{K}.$$

Subsequently, T has a precisely unique fixed point (e.g. unique $L(.)\in\mathcal{K}$ such that $T(L)=L$).

Theorem 2: Suppose that $L(\upsilon)$ in Theorem 1, the ARL obtained from explicit formula corresponding to the CUSUM control chart for a long-memory SFIMAX$(D,Q,k)_L$ process exists and is unique.

***Proof*:** To prove the existence of the ARL.

Let T be a contraction in complete metric space $(\mathcal{K},d)$, $\mathbf{C}[0,b]$ be a set of continuous functions of the ARL on interval $[0,b]$ and ARL_0 be an arbitrary but fixed element in $\mathcal{K}$. Define a sequence of iterates $\{L_n\}_{n\ge0}$ in $\mathcal{K}$ by $L_{n+1} = T(L_n)$, for all $n\ge1$. Since T is a contraction, then

$$d\left(L_2,L_1\right)=d\left(T\left(L_1\right),T\left(L_0\right)\right)\le\rho d\left(L_1,L_0\right),\ 0\le\rho<1.$$

Continuing inductively, we obtain

$$d\left(L_{n+1},L_n\right)\le\rho d\left(L_n,L_{n-1}\right)\le\rho^2 d\left(L_{n-1},L_{n-2}\right)\le...\le\rho^n d\left(L_1,L_0\right).$$

Repeatedly applying the triangle inequality into this formula when $n<m$ implies that

$$d\left(L_n,L_m\right)\le d(L_n,L_{n+1})+...+d(L_{m-1},L_m),$$

it follows that

$$d\left(L_n,L_m\right)\le(\rho^n+\rho^{n+1}+...+\rho^{m-1})d(L_1,L_0).$$

Using the property of sum is a geometric series in ρ, we obtain

$$d\left(L_n,L_m\right)\le\frac{\rho^n}{1-\rho}d(L_1,L_0).$$

From above, $0\le\rho<1$ implies that $\rho^n/(1-\rho)\to0$ as $n\to\infty$. Hence, $\{L_n\}_{n\ge0}$ is a Cauchy sequence. There is a limit point of ARL in $\mathcal{K}$ because $(\mathcal{K},d)$ is complete metric space. Hence, there exists a unique point $L\in\mathcal{K}$ such that

$$T(L)=\lim_{n\to\infty}T\left(L_n\right)=\lim_{n\to\infty}L_{n+1}=L.$$

This completes the proof.

***Proof*:** To prove the uniqueness of the ARL.

Let L_1 and L_2 be two arbitrary functions in $\mathbf{C}[0,b]$. The common term for complete metric space is $\left(\mathbf{C}[0,b],\ \|.\|_\infty\right)$. That is to say, a set of continuous functions of the ARL defined on $[0,b]$, and $\mathbf{C}[0,b]$ becomes norm space if we define

$$\|L\|_\infty=\sup_{\upsilon\in[0,b]}\left|\int_0^b k(\upsilon,g)dg\right|,$$

for all functions $k(\upsilon,g)\in\mathbf{C}[0,b]$, where $k(\upsilon,g)$ is a kernel function of the IE for the ARL based on explicit formulas obtained by using Theorem 1:

$$\|T(L_1)-T(L_2)\|_\infty$$

$$=\sup_{\upsilon\in[0,b]}\left|\int_0^b\beta\left(\exp\left\{\begin{matrix}\beta(\upsilon-a+\mu-\Theta_1\varepsilon_{t-L}\\-\Theta_2\varepsilon_{t-2L}-...-\Theta_P\varepsilon_{t-QL}\\+DY_{t-L}-\frac{D(D-1)}{2}Y_{t-2L}\\+\frac{D(D-1)(D-2)}{6}Y_{t-2L}-...\\+\omega_1X_{1t}+\omega_2X_{2t}+...+\omega_kX_{kt})\end{matrix}\right\}\right)\left|L_1(g)-L_2(g)\right|dg\right|$$

$$\le\sup_{\upsilon\in[0,b]}\int_0^b\left|\beta\left(\exp\left\{\begin{matrix}\beta(\upsilon-a+\mu-\Theta_1\varepsilon_{t-L}\\-\Theta_2\varepsilon_{t-2L}-...-\Theta_P\varepsilon_{t-QL}\\+DY_{t-L}-\frac{D(D-1)}{2}Y_{t-2L}\\+\frac{D(D-1)(D-2)}{6}Y_{t-2L}-...\\+\omega_1X_{1t}+\omega_2X_{2t}+...+\omega_kX_{kt})\end{matrix}\right\}\right)\right|dg\left\|L_1(g)-L_2(g)\right\|_\infty.$$

Hence, we obtain

$$\left\|T(L_1)-T(L_2)\right\|_\infty=\rho\left\|L_1(g)-L_2(g)\right\|_\infty,$$

where $\rho=\sup_{\upsilon\in[0,b]}\int_0^b\left|\beta\left(\exp\left\{\begin{matrix}\beta(\upsilon-a+\mu-\Theta_1\varepsilon_{t-L}-\Theta_2\varepsilon_{t-2L}\\-...-\Theta_P\varepsilon_{t-QL}+DY_{t-L}-\frac{D(D-1)}{2}Y_{t-2L}\\+\frac{D(D-1)(D-2)}{6}Y_{t-2L}-...\\+\omega_1X_{1t}+\omega_2X_{2t}+...+\omega_kX_{kt})\end{matrix}\right\}\right)\right|dg<1.$

This completes the proof.

Therefore, the ARL based on explicit formulas for the CUSUM control chart for a long-memory SFIMAX$(D,Q,k)_L$ process exists and is unique.

Acknowledgments

The author gratefully acknowledges the editor and referees for their valuable comments and suggestions which greatly improve this paper. This research was funded by King Mongkut's University of Technology North Bangkok, Contract no. KMUTNB-65-BASIC-23.

REFERENCES

[1] M.R. Abujiya, M. Riaz, M.H. Lee. Enhanced Cumulative Sum Charts for Monitoring Process Dispersion, PLoS ONE, Vol.10, No.4, 1-22, 2015.

[2] E.S. Page, Continuous Inspection Schemes, Biometrika, Vol.41, 100-114, 1954.

[3] S.W. Roberts. Control Chart Tests Based on Geometric Moving Averages, Technometrics, Vol.1, No.3, 239–50, 1959.

[4] W.A. Shewhart. Economic control of quality of manufactured product, American Society for Quality Control, Milwaukee, Wis, 1980.

[5] D.B. Hibbert. Quality Assurance in the Analytical Chemistry Laboratory, Oxford University Press, 2007.

[6] G. Kateman, L. Buydens. Quality control in analytical chemistry, Wiley, New York, 1993.

[7] S.S. Cheng, F.J Yu. A CUSUM control chart to monitor wafer quality, Journal of Information and Optimization Sciences, Vol.35, 483-501, 2013.

[8] N.M. Novoa, G. Varela. Monitoring surgical quality: the cumulative sum (CUSUM) approach, Mediastinum 4, 4–4, 2020.

[9] W.H. Woodall. On the markov chain approach to the two-sided CUSUM procedure, Technometrics, Vol.26, No.1, 41-46, 1984.

[10] W.D. Ewan, K.W. Kemp. Sampling inspection of continuous processes with no autocorrelation between successive results, Biometrika, Vol.47, No.3/4, 1960.

[11] D. Brook, D.A. Evans. An approach to the probability distribution of cusum run length, Biometrika, Vol.59, No.3, 539-549, 1972.

[12] S. Zacks. The probability distribution and the expected value of a stopping variable associated with one-sided cusum procedures for non-negative integer valued random variables, Communications in Statistics - Theory and Methods, Vol.10, No.21, 2245-2258, 1981.

[13] L.N. Vanbrackle, M.R. Reynolds. EWMA and CUSUM control charts in the presence of correlation, Communications in Statistics - Simulation and Computation, Vol.26, No.3, 979-1008, 1997.

[14] O.O. Atienza, L.C. Tang, B.W. Ang. A CUSUM scheme for autocorrelated observations, Journal of Quality Technology, Vol.34, No.2, 187–199, 2002.

[15] Y.M. Chang, T.L. Wu. On average run lengths of control charts for autocorrelated processes, Methodology and Computing in Applied, Vol.13, No.2, 419-431, 2011.

[16] J.R.M. Hosking. Fractional differencing, Biometrika, Vol.68, No.1, 165–176, 1981.

[17] H.E. Hurst. Long-term storage capacity of reservoirs, Transactions of the American Society of Civil Engineers, Vol.116, No.1, 770–799, 1951.

[18] C.W.J. Granger, R. Joyeux. An Introduction to long memory time series models and fractional differencing, Journal of Time Series Analysis, Vol.1, No.1, 15-29, 1980.

[19] L.A. Gil-Alana. Testing fractional integration with monthly data, Economic Modelling, Vol.16, No.4, 613–629, 1999.

[20] M. Ooms, P.H. Franses. A seasonal periodic long memory model for monthly river flows, Environmental Modelling & Software, Vol.16, No.6, 559–569, 2001.

[21] K. Petcharat, S. Sukparungsee, Y. Areepong. Exact solution of the average run length for the cumulative sum chart for a moving average process of order q, ScienceAsia, Vol.41, No.2, 141-147, 2015.

[22] P. Phanthuna, Y. Areepong. Analytical solutions of ARL for $SAR(p)_L$ model on a Modified EWMA chart, Mathematics and Statistics, Vol.9, No.5, 685–696, 2021. DOI: 10.13189/ms.2021.090508.

[23] S. Phanyaem. Explicit formulas and numerical integral equation of arl for SARX(P,r)L model based on CUSUM chart, Mathematics and Statistics, Vol.10, No.1, 88–99, 2022. DOI: 10.13189/ms.2022.100107.

[24] W. Suriyakat, K. Petcharat. Exact run length computation on ewma control chart for stationary moving average process with exogenous variables, Mathematics and Statistics, Vol.10, No.3, 624–635, 2022. DOI: 10.13189/ms.2022.100319.

[25] R. Sunthornwat, Y. Areepong, and S. Sukparungsee. Average Run Length with a practical investigation of estimating parameters of the EWMA control chart on the long memory AFRIMA Process, Thailand Statistician, Vol.16, No.2, 190-202, 2018.

[26] Y. Areepong, W. Peerajit. Integral equation solutions for the average run length for monitoring shifts in the mean of a generalized seasonal ARFIMAX(P, D, Q, r)s process running on a CUSUM control chart, PLoS ONE, Vol.17, No.2, e0264283, 2022.

[27] G.E.P. Box, G.M. Jenkins, G.C. Reinsel, G.M. Ljung. Time series analysis: forecasting and control, Fifth edition, John Wiley & Sons, Inc, Hoboken, New Jersey, 2016.

[28] C.W. Champ, S.E. Rigdon. A a comparison of the markov chain and the integral equation approaches for evaluating the run length distribution of quality control charts, Communications in Statistics - Simulation and Computation, Vol.20, No.1, 191-204, 1991.

[29] B.V. Rao, R.L. Disney, J.J. Pignatiello. Uniqueness and convergence of solutions to average run length integral equations for cumulative sum and other control charts, IIE Transactions, Vol.33, No.6, 463–469, 2001.

[30] D. Bualuang, W. Peeraji. Performance of the CUSUM Control Chart Using Approximation to ARL for Long-Memory Fractionally Integrated Autoregressive Process with Exogenous Variable, Applied Science and Engineering Progress, Vol.16, No.2, 2022.

[31] D.C. Montgomery. Introduction to statistical quality control, 5th ed, John Wiley, Hoboken, N.J, 2005.