Chapter Four

An Application Of The New Gamma ARMA(1, 1) Model To Monthly Stream Flows

4.1 Introduction

As an application, the proposed Gamma *ARMA*(1, 1) model of Section 3.4.2 is fitted to the seasonally adjusted monthly river flows of Perak river in Malaysia. The seasonal effects of the original series are removed by using the classical method discussed in Section 2.2, i.e., subtract each observation from its corresponding monthly mean and then divide by its monthly standard deviation. By fitting a Gamma distribution to the above seasonally adjusted series, we find that the Chi-square statistic χ^2 is 18.749 with 15 degrees of freedom. Since $P(\chi^2 > 18.749) = 0.2253$, we conclude that the seasonally adjusted river flows of the Perak river follow a Gamma distribution.

The method of moments is used to estimate the model's parameters p_1, p_2, α, and k. As an alternative, we shall also estimate the parameter p_1 by using the least squares method (Lawrence and Paul, 1978) and the remaining parameters by moments method.

We shall investigate the performance of the proposed model by comparing the simulated sequence to the historical sequence in terms of means, variances, skewness and autocorrelation coefficients. We shall also compare the simulated results produced by the two different sets of estimated parameters, viz, the set of parameters estimated solely from the moments method and the set of parameters estimated from the combination of moments and least squares methods. The simulation results are discussed and displayed in Section 4.3.

4.2 Parameters Estimation

In order to preserve the variance s^2, skewness γ, lag-1 serial correlation r_1 and lag-2 serial correlation r_2 of the standardized historical data, the model's parameters k, α, p_1 and p_2 are estimated from the following system of equations,

$$s^2 = \begin{cases} k\theta^2(p_1+\nu^2\bar{p}_1), & \nu \neq 1 \\ k\theta^2, & \nu = 1 \end{cases} \tag{4.1}$$

$$r_2 = p_2 r_1 \tag{4.2}$$

$$\gamma = \begin{cases} \dfrac{2}{\sqrt{k}}\left[\dfrac{p_1 + \nu^3\bar{p}_1}{(p_1 + \nu^2\bar{p}_1)^{3/2}}\right], & \nu \neq 1 \\ \dfrac{2}{\sqrt{k}}, & \nu = 1 \end{cases} \tag{4.3}$$

$$r_1 = \begin{cases} \dfrac{\nu(\nu p_2\bar{p}_1 + p_1\bar{p}_2)}{p_1 + \nu^2\bar{p}_1} & \nu \neq 1 \\ (p_2\bar{p}_1 + p_1\bar{p}_2) & \nu = 1 \end{cases} \tag{4.4}$$

From the above formulas, we notice that when $\nu = 1$, the proposed model has Gamma distribution with shape parameter k and scale parameter θ^{-1}. We shall use this model for fitting the monthly standardized series of Perak river. The parameter k is first obtained from equation (4.3), which then enables us to obtain the value of θ from (4.1). We next obtain the value of p_2 from (4.2) and thus the value of α can be evaluated as

$\alpha = 1/\theta(1-p_2)$. Finally, the value of p_1 can be calculated from (4.4) and the mean of the standardized historical data is preserved by shifting the origin of the standardized historical data by a magnitude c, whose value is given by

$$\bar{x} = k\theta(p_1+\nu\bar{p}_1) + c. \tag{4.5}$$

Lawrence and Paul (1978) have developed an estimation procedure for dependent observations based on the minimization of a sum of squared deviations about conditional expectations. As an alternative, we shall use this approach to estimate the parameter p_1 in the proposed Gamma *ARMA*(1, 1) process. The conditional sum of squares $Q_n(\cdot)$ of the process (3.25) is given by

$$Q_n(p_1) = \sum_{t=0}^{n-1} [x_{t+1} - E(X_{t+1}|X_t = x_t)]$$

$$= \sum_{t=0}^{n-1}\left\{x_{t+1} - \frac{k}{\alpha(1-p_2)}\,[p_2(1-2p_1)+p_1] - (p_1\bar{p}_2+p_2\bar{p}_1)x_t\right\}^2,$$

where the conditional expectation $E(X_{t+1}|X_t)$ is given in Section 3.4.2.

By differentiating the conditional sum of squares with respect to p_1, we obtain the following relation

$$(p_1\bar{p}_2+p_2\bar{p}_1) = \frac{n\sum x_{t+1}x_t - \sum x_{t+1}\sum x_t}{n\sum x_t^2 - (\sum x_t)^2}, \tag{4.6}$$

where all the sums are run from $t = 0$ to $t = n-1$. Note that this relation is slightly different from (4.4).

With known value of p_2, the value of p_1 can be evaluated from (4.6). For the purpose of preserving the mean, variance, skewness, lag-1 and lag-2 serial correlations of the standardized series, the parameters k, α and p_2 are again estimated as before.

4.3 Simulation

An algorithm for generating a new Gamma *ARMA*(1, 1) sequence x_1, x_1, ..., x_n of Section 3.4.2 is as follows :

ALGORITHM

Generate Y_0 from Gamma($k\bar{p}_1$, θ^{-1}).

Set $i = 1$.

REPEAT

  Generate G from Gamma($k\bar{p}_1$, θ^{-1}).

  Generate B from Beta(kp_1, $k(1-2p_1)$).

  Compute $x_i = \nu y_{i-1} + BG + c$.

  Generate N from Poisson($\alpha p_2 y_{i-1}$).

  Set sum = 0, $j = 1$.

  REPEAT

    Generate W from Exp(α).

    Compute sum = sum + W, $j = j+1$.

  UNTIL $j > N$.

  Compute y_i = sum + $(1-p_2)G$.

  Compute $i = i + 1$.

UNTIL $i > n$.

The above algorithm is written in Fortran 77 and the program is attached in Appendix 4A.

The program is executed on the HP9000/320 workstation equipped with an MC68020 microprocessor and MC68881 coprocessor. The built-in pseudo-random number generator, RAN, of HP9000 Fortran 77 is used to generate the required Uniform $\mathcal{U}(0, 1)$ random variates. The independent $G(k, \beta)$ are generated by the rejection method of Ahrens and Dieter (1974). The Poisson(λ), $\lambda \le 60$ and the independent $Exp(\alpha)$ are generated by the widely used inverse transform method. For $\lambda > 60$, the Poisson(λ) variables are generated by the Normal approximation. The Normal(0, 1) variables are generated by the ratio method of Ripley (1984), and the independent $B(m, n)$ are generated by using the result that $X = Y_1/(Y_1+Y_2)$ is a $B(m, n)$ if $Y_1 + Y_2 \leq 1$, where $Y_1 = \mathcal{U}^{1/m}$, $Y_2 = \mathcal{U}^{1/n}$ and $\mathcal{U}_1$, $\mathcal{U}_2$ are two Uniform variates from $\mathcal{U}(0, 1)$ (Rubinstein, 1980, pg. 81).

Table 4.1 shows the sample autocorrelation r_k, partial autocorrelation $\hat{\phi}_{kk}$, mean, variance and skewness for the standardized data of Perak river. From these statistics, it is clearly seen that the standardized data is skewed and with long-term correlation structure.

In the simulation study, 100 sequences of the standardized monthly stream flows Y_t, each of them consisting 480 data, are simulated from the fitted Gamma $ARMA(1, 1)$ model of Section 3.4.2. For each of the 100 simulated sequences, the mean, variance, skewness and the autocorrelation coefficients r_j, $j = 1, 2, \ldots, 12$ are calculated. The original series X_t is then obtained as

$$X_t = m_\tau + s_\tau Y_t, \quad \tau = 1, 2, \ldots, 12$$

where m_τ and s_τ are respectively the mean and standard deviation of a given month τ of the original series. The simulated series in Table 4.2 are generated by using the parameters obtained solely from the method of moments. This table gives a comparison between characteristics of the historical data and characteristics of the simulated data for both the original and standardized series.

From the results shown in Table 4.2 and Table 4.3, we observe that the simulated series bears a reasonable resemblance to the historical series in terms of means, variances, skewness and autocorrelation coefficients.

Table 4.4 compares the performance of the model by using the two alternative sets of parameters. From this table, we can observe that the sample mean, skewness and serial correlations of the simulated series (except the sample variance and the third autocorrelation coefficient) have improved slightly if the combination of moments and least squares methods is used for simulation and furthermore, the standard errors (in parentheses) for most of the statistics have also been reduced slightly, which indicates that the simulation is more stable.

Table 4.1

Sample Autocorrelation r_k, Partial Autocorrelation $\hat{\phi}_{kk}$, Mean, Variance And Skewness Of The Standardized Data Of Perak River.

	k							
	1	2	3	4	5	6	7	8
r_k	0.642	0.546	0.4641	0.429	0.333	0.281	0.329	0.238
$\hat{\phi}_k$	0.642	0.229	0.084	0.093	-0.051	-0.007	0.171	-0.096
	k							
	9	10	11	12	13	14	15	16
r_k	0.174	0.102	0.114	0.090	0.134	0.160	0.167	0.127
$\hat{\phi}_k$	-0.070	-0.076	0.038	0.032	0.132	0.052	0.022	-0.056
	k							
	17	18	19	20	21	22	23	24
r_k	0.120	0.133	0.091	0.084	0.059	0.066	0.108	0.017
$\hat{\phi}_k$	0.016	0.032	-0.053	-0.033	-0.055	0.003	0.168	-0.140
Mean = 0.0000, Variance = 1.0000, Skewness = 0.9061								

Table 4.2

Mean, Variance, Skewness And Autocorrelation Coefficients r_j, j = 1, 2, 3, ..., 12 of The Historical Stream Flows And The Synthetic Stream Flows Generated By The Gamma ARMA(1, 1) Model.

monthly streamflows statistics	Original Data		Standardized Data	
	Historical	Simulated	Historical	Simulated
Mean	2.732	2.781 (0.192)	0.000	0.040 (0.161)
Variance	2.700	2.750 (0.451)	1.000	1.002 (0.154)
Skewness	1.569	1.508 (0.249)	0.921	0.855 (0.211)
r_1	0.632	0.619 (0.039)	0.642	0.628 (0.055)
r_2	0.373	0.362 (0.052)	0.546	0.529 (0.064)
r_3	0.173	0.133 (0.064)	0.464	0.445 (0.072)
r_4	0.064	0.004 (0.064)	0.429	0.371 (0.079)
r_5	0.035	0.006 (0.063)	0.333	0.308 (0.083)
r_6	0.030	-0.002 (0.063)	0.281	0.256 (0.089)
r_7	0.027	-0.041 (0.062)	0.329	0.212 (0.086)
r_8	-0.081	-0.093 (0.059)	0.238	0.174 (0.089)
r_9	-0.032	-0.029 (0.062)	0.174	0.141 (0.085)
r_{10}	0.113	0.125 (0.058)	0.102	0.116 (0.085)
r_{11}	0.296	0.296 (0.056)	0.114	0.090 (0.086)
r_{12}	0.383	0.397 (0.065)	0.090	0.070 (0.086)

*Means and standard errors (in parentheses) of the 100 simulated series.

TABLE 4.3

Means And Standard Deviations Of Each Of The 12 Calendar Months Of The Historical And Simulated Data.

Month	Means		Standard Deviations	
	Historical	Simulated	Historical	Simulated
Jan.	3.612	3.710 (0.396)	2.090	2.080 (0.266)
Feb.	2.230	2.267 (0.203)	1.122	1.094 (0.169)
March	1.887	1.934 (0.185)	0.955	0.946 (0.148)
April	2.076	2.102 (0.174)	0.827	0.798 (0.121)
May	2.689	2.714 (0.183)	0.934	0.904 (0.136)
June	1.950	1.975 (0.135)	0.675	0.659 (0.111)
July	1.763	1.782 (0.085)	0.445	0.442 (0.059)
Aug.	1.966	2.012 (0.253)	1.326	1.305 (0.918)
Sept.	2.040	2.082 (0.140)	0.686	0.689 (0.107)
Oct.	3.774	3.848 (0.315)	1.690	1.680 (0.229)
Nov.	4.600	4.656 (0.347)	1.790	1.779 (0.268)
Dec.	4.207	4.284 (0.356)	1.958	1.910 (0.274)

*Means and standard errors (in parentheses) of the 100 simulated series.

TABLE 4.4

Mean, Variance, Skewness And Autocorrelation Coefficients r_j, $j = 1, 2, \ldots, 12$ *Of The Two Synthetic Stream Flows Generated By The Gamma ARMA(1, 1) Model With Different Sets Of Parameters.*

Monthly Streamflows Statistics	Standardized Data Historical	Simulated Data Moments method	Combination
Mean	0.000	0.040 (0.161)	0.022 (0.145)
Variance	1.000	1.002 (0.154)	0.983 (0.154)
Skewness	0.921	0.855 (0.211)	0.858 (0.211)
r_1	0.642	0.628 (0.055)	0.631 (0.055)
r_2	0.546	0.529 (0.064)	0.533 (0.068)
r_3	0.464	0.445 (0.072)	0.442 (0.068)
r_4	0.429	0.371 (0.079)	0.373 (0.072)
r_5	0.333	0.308 (0.083)	0.314 (0.075)
r_6	0.281	0.256 (0.089)	0.264 (0.081)
r_7	0.329	0.212 (0.086)	0.219 (0.081)
r_8	0.238	0.174 (0.089)	0.181 (0.082)
r_9	0.174	0.141 (0.085)	0.147 (0.084)
r_{10}	0.102	0.116 (0.085)	0.121 (0.078)
r_{11}	0.114	0.090 (0.086)	0.100 (0.078)
r_{12}	0.090	0.070 (0.086)	0.081 (0.080)

*Means and standard errors (in parentheses) of the 100 simulated series.

Appendix 4A

Below are programs written for the algorithm given in Section 4.3.

```
      PROGRAM MAIN
c     ---------------------------------------------------------
c     This program is written for generating monthly river flows
c     using the algorithm discussed in Section 4.3.  This
c     program returns the vital statistics, such as the mean,
c     variance, skewness, autocorrelation coefficients r_j,
c     (j = 1, 2, ..., 12) and the monthly mean and standard
c     deviation for each of the 12 calendar months of the
c     generated series.  These statistics can then be used to
c     evaluate the performance of the proposed model.
c     ---------------------------------------------------------
c     The following subroutines/functions are directly called
c     by this program:
c     momen
c     stat
c     stat1
c     gama
c     beta
c     expon
c     poi
c     poisso
c     normal
c     acf
c     ---------------------------------------------------------
c     Required inputs   Corresponding parameter in (3.25)
c     ***************   *********************************
c
c          alpha                        α
c          P1                           p_1
c          P2                           p_2
c          gk                           k
c          v                            ν
c     ************************************************************
c
c     imonth(i) = mean of the τ-th month of the original series
c     isd(i)      = standard deviation of the τ-th month of the
c                   original series
c               c = magnitude of the shifted origin as given in
c                   (4.5)
c     ---------------------------------------------------------
c
c     Constants used in this program:
c     n1 = number of samples
c     n2 = number of data in a sample
c     ---------------------------------------------------------
c
c     Declarations
c     ---------------------------------------------------------
c
      real month,imonth,isd
      integer poisso,poi,count1,count2,count3
      dimension imonth(12),isd(12),month(12),sd(12)
```

```
      dimension ar(12),sdar(12),r1(12)
      dimension,rmonth(100,12),rsd(100,12),r2(100,12)
      dimension y(601),x(601)
c     --------------------------------------------------
      read *, alpha,P1,P2,gk,v,c
      teta=alpha(1-P2)
      do k=1,12
          read *,imonth(k),isd(k)
      end do
c     --------------------------------------------------
      n1=100
      n2=480
c     --------------------------------------------------
      write (6,994) alpha,P1,P2,gk,teta,v
c     ------------------------------------------------------------
c               beta2 = shape parameter of Gamma(beta2, teta)
c     beta1 and beta3 = parameters of Beta(beta1, beta3)
c     ------------------------------------------------------------
      beta1=gk*P1
      beta2=gk*(1-P1)
      beta3=gk*(1-(2*P1))
      write (6,997) beta1,beta2,beta3,teta
c     ------------------------------------------------------------
c     Initializations
c     ------------------------------------------------------------
      ta1=0.
      ta2=0.
      tv1=0.
      tv2=0.
      ts1=0.
      ts2=0.
      iseed=113
c     ------------------------------------------------------------
      write (6,9999)
      do l=1,n1
         y(1)=gama(beta2,teta,iseed)
         do i=2,n2+1
            b=Beta(beta1,beta3,iseed)
            g=gama(beta2,teta,iseed)
            x(i)=v*y(i-1)+(b*g)+c
            t=P2*alpha*y(i-1)
            if (t .gt. 60) then
              m=poi(t,iseed)
            else
              m=poisso(t,iseed)
            endif
            sum=0.
            do j=1,m
               w=expon(alpha,iseed)
               sum=sum+w
            end do
            y(i)=sum+(1-P2)*g
         end do
c        ------------------------------
c        Compute the historical series
```

```
c      ----------------------------
         count2=0
         do while (count2 .le. n2-12)
           do count1=1,12
              count3=count2+count1
              x(count3)=x(count3+1)*isd(count1)+imonth(count1)
           end do
           count2=count2+12
         end do
c      ----------------------------------------------------------
c      Compute and display the mean, variance and skewness  of
c      each set of the generated   data.  The autocorrelation
c      coefficients r_j (j = 1, 2, ..., 12), the monthly mean  and
c      standard deviation  for each of the 12 calendar months are
c      also computed; they are kept in the arrays r2(n1, 12),
c      rmonth(n1, 12) and rsd(n1, 12) respectively.
c      ----------------------------------------------------------
          call momen(x,averx,varx,skew,r1,n2)
          call stat(x,month,sd,n2)
          do n=1,12
             r2(l,n)=r1(n)
             rmonth(l,n)=month(n)
             rsd(l,n)=sd(n)
          end do
          write (6,999)l,averx,varx,skew
          ta1=ta1+averx
          ta2=ta2+(averx**2)
          tv1=tv1+varx
          tv2=tv2+(varx**2)
          ts1=ts1+skew
          ts2=ts2+(skew**2)
       end do
c      ----------------------------------------------
c      Compute overall average, variance and skewness
c      ----------------------------------------------
       aa=ta1/n1
       va=(ta2/n1)-(aa**2)
       sda=sqrt(va)
       av=tv1/n1
       vv=(tv2/n1)-(av**2)
       sdv=sqrt(vv)
       as=ts1/n1
       vs=(ts2/n1)-(as**2)
       sds=sqrt(vs)
       write (6,995) aa,av,as
       write (6,998) sda,sdv,sds
       write (6,9998)
c      ----------------------------------------------------------
c      Compute  and  display  the  overall  average  and  standard
c      error  for  each  of  the  12  sample autocorrelation
c      coefficients r_j (j = 1, 2, ..., 12) of the n1 samples.
c      ----------------------------------------------------------
       call stat1(r2,ar,sdar,n1)
       do i=1,12
```

```
        write (6,996) i,ar(i),sdar(i)
      end do
c     -------------------------------------------------------
c     Compute and display the overall average and standard error
c     of the monthly mean and standard deviation for each of the
c     12 calendar months of the n1 samples.
c     -------------------------------------------------------
      write (6, 9996)
      call stat1(rmonth,month,sd,n1)
      call stat1(rsd,ar,sdar,n1)
      do k=1,12
        write(6,9997) k,month(k),sd(k),ar(k),sdar(k)
      end do
c     -------
c     Formats
c     -------
994   format (6f10.4/)
995   format (//1x,4haver,3f10.3/)
996   format (i5,2f10.3)
997   format (4f10.4//)
998   format (2x,4hs.d.,3f10.3)
999   format (i5,3f10.3)
9996  format (//2x,7hmonth  ,6hmeans ,6hs.d.   ,11hstand.dev. 4hs.d.)
9997  format (i5,6f7.3)
9998  format (//2x,4hlag ,13hsamp.autocor ,12h stand.devia/)
9999  format (//10x,7haver   ,4x,6hvar   ,3x,4hskew/)
      stop
      end

c      SUBROUTINE momen(x,averx,varx,skew,r1,n2)
c     -------------------------------------------------------
c     Subroutine momen is used to compute the mean, variance,
c     skewness   and   autocorrelation   coefficients   r_j
c     (j = 1, 2, ..., 12) for each set of the generated data.
c     -------------------------------------------------------
c         x = generated series
c     averx = average of the series
c     varx  = variance of the series
c     skew  = skewness of the series
c     r1    = autocorrelation coefficients of the series
c     n2    = number of data in a sample
c     -------------------------------------------------------
      dimension x(601),r1(12)
      tol1=0.
      tol2=0.
      tol3=0.
      do i=1,n2
        tol1=tol1+x(i)
        tol2=tol2+x(i)**2
        tol3=tol3+x(i)**2
      end do
      averx=tol1/n2
      varx=tol2/n2-(averx**2)
      skew=tol3-(3*tol2*averx)+(3*tol1*averx**2)
      skew=skew/n2-(averx**3)
```

```
      var2=sqrt(varx)**3
      skew=skew/var2
      call acf(x,r1,averx,varx,n2)
      return
      end

      SUBROUTINE stat(x,month,sd,n2)
c     ---------------------------------------------------------
c     Subroutine stat is used to compute the monthly mean and
c     standard deviation for each of the 12 calendar months of
c     every generated series.
c     month = monthly mean of the generated series
c     sd    = monthly standard deviation of the generated
c             series
c     n2    = number of data in a sample
c     ---------------------------------------------------------
      real month,t1,t2
      integer count1,count2,count3
      dimension month(12),sd(12)
      dimension x(601)
      n4=n2/12
      do count1=1,12
        t1=0.
        t2=0.
        count2=0
        do while (count2 .le. n2-12)
          count3=count2+count1
          t1=t1+x(count3)
          t2=t2+x(count3)**2
          count2=count2+12
        end do
        month(count1)=t1/n4
        sd(count1)=t2/n4-month(count1)**2
        sd(ccunt1)=sqrt(sd(count1))
      end do
      return
      end

      SUBROUTINE stat1(r2,ar,sdar,n1)
c     ---------------------------------------------------------
c     Subroutine stat1 is used to compute the overall average
c     and standard error for the autocorrelation coefficients
c     r_j (j = 1, 2, ..., 12) of the n1 samples.  The monthly
c     mean  and monthly standard deviation  for each of the
c     12 calendar months of the n1 samples are also computed
c     by this subroutine.
c     ar    = average of the statistics
c     sdar  = standard error of the statistics
c     n1    = number of samples
c     ---------------------------------------------------------
      dimension r2(100,12),ar(12),sdar(12)
      do i=1,12
        tr1=0.
        tr2=0.
        do j=1,n1
```

```
    tr1=tr1+r2(j,i)
    tr2=tr2+(r2(j,i)**2)
  end do
  ar(i)=tr1/n1
  sdar(i)=(tr2/n1)-(ar(i)**2)
  sdar(i)=sqrt(sdar(i))
end do
return
end

FUNCTION gama(alpha,lamda,iseed)
c     ------------------------------------------------------------
c     Function gama is used to generate the independent G(k, β).
c     It uses the rejection method of Ahrens and Dieter
c     (1974).
c     ------------------------------------------------------------
real alpha,lamda
m=int(alpha)
a=alpha-m
yy=1.
do i =1,m
   yy=yy*ran(iseed)
end do
yy=-alog(yy)
if (a .ne. 0.) then
  b=1.-a
  u=2.
  do while (u .gt. 1.)
    u1=ran(iseed)**(1./a)
    u2=ran(iseed)**(1./b)
    u=u1+u2
  end do
  w=u1/u
  v=-alog(ran(iseed))
else
  w=0.
  v=0.
endif
gama=(yy+v*w)/lamda
return
end

FUNCTION beta(q1,q2,ip3)
c     ------------------------------------------------------------
c     Function beta is used to generate the independent B(m, n).
c     The Beta variates are generated by using the result
c     (Rubinstein, 1980, pg. 81) that if Y1+Y2 ≤ 1, then
c     X=Y1/(Y1+Y2) is a B(m, n), where Y1=U1^(1/m), Y2=U2^(1/n) and
c     U1, U2 are two Uniform variates from U(0, 1).
c     ------------------------------------------------------------
real init
z12=2.
```

```fortran
      do while (z12 .ge. 1)
         init=ran(ip3)
         z1=init**(1/q1)
         init=ran(ip3)
         z2=init**(1/q2)
         z12=z1+z2
      end do
      beta=z1/z12
      return
      end

      FUNCTION expon(t,ip3)
c     ------------------------------------------------------------
c     Function Expon is used to generate the independent Exp(α).
c     It uses the inverse transform method.
c     ------------------------------------------------------------
      real init
      init=ran(ip3)
      expon=-alog(init)/t
      return
      end

      INTEGER FUNCTION poi(lamda,iseed)
c     ------------------------------------------------------------
c      Function poi is used to generate the independent Poi(λ)
c     variates when λ > 60.   The Poisson distribution is
c     approximated by the Normal distribution.
c     ------------------------------------------------------------
      real lamda
      e1=exp(1.)
      e2=sqrt(2./e1)
      call normal(x,e2,iseed)
      sx=sqrt(lamda)
      poi=int(lamda+x*sx+.5)
      if(poi .le. 0) poi=0
      return
      end

      INTEGER FUNCTION poisso(lamda,iseed)
c     ------------------------------------------------------------
c     Function poisso is used to generate the independent
c     Poisson variates when λ ≤ 60.   It uses the inverse
c     transform method.
c     ------------------------------------------------------------
      real lamda
      c=exp(-lamda)
      b=c
      k=0
      a=ran(iseed)
      do while(a .gt. b)
        c=c*lamda/(k+1)
        b=b+c
        k=k+1
      end do
      poisso=k
```

```
      return
      end

      SUBROUTINE normal(x,e2,iseed)
c     ------------------------------------------------------------
c     Subroutine normal is used to generate the independent
c     Normal(μ, σ^2) variates.  It uses the ratio method of
c     Ripley (1983).
c     ------------------------------------------------------------
      tw=1.
      do while (tw .gt. 0.)
         u=ran(iseed)
         u1=ran(iseed)
         v=(2.*u1-1.)*e2
         x=v/u
         z=(x**2.)/4.
         if (z .le. 1.-u) then
           tw=-1.
         else
           if (z .le. -alog(u)) then
             tw=-1.
           endif
         endif
      end do
      return
      end

      SUBROUTINE acf(xx,rr,rev1,rev2,ms)
c     ------------------------------------------------------------
c     Subroutine acf is used to compute the 12 autocorrelation
c     coefficients r_j (j = 1, 2, ..., 12) of a given series.
c     ------------------------------------------------------------
      dimension rr(12),xx(600)
      do j=1,12
         sum1=0.
         j1=ms-j
         do i=1,j1
            sum1=sum1+(xx(i)-rev1)*(xx(i+j)-rev1)
         end do
         sum1=sum1/ms
         rr(j)=sum1/rev2
      end do
      return
      end
```

Conclusion

The main aim of this study is to develop suitable probabilistic models for the stochastic component of hydrologic time series, which is independent of seasonal movement and trend cycle. By means of simulation, we then generate synthetic series for the stochastic, seasonal and trend components, and combine these generated series to get an artificial series that can be used in various water resource projects.

In order to separate the stochastic component from its original series, we have attempted to use the decomposition techniques such as the classical non-parametric method, the Fourier series approach, the Census X-11 procedure and the *ARIMA*-model-based approach. We have also appraised the applicability of these techniques in hydrologic study. From the results obtained in Chapter Two, techniques like the Census X-II procedure and the *ARIMA*-model-based approach, which have been developed for decomposing economic and business time series into trend, seasonal and irregular components, are not appropriate for serving the main aim of this study.

In the second part of this study, we have proposed a first-order *ARMA* model that can be used to generate correlated Gamma-like processes. By comparing the simulated results of the proposed model (as shown in Tables 4.2 and 4.3) to those of the *MGARMA*(1, 1) model of Sim (1987a), we notice that the performance of the proposed model is not as good as the latter. However, unlike the newly proposed model, the parameters of the *MGARMA*(1, 1) model have to be obtained by solving a set of two simultaneous equations, which does not always promise a solution. Therefore in practice, the proposed model can be used as an alternative to the *MGARMA*(1, 1) model when the restrictions discussed in Section 3.4.1 of Chapter Three are encountered in parameters estimation.