This is a WinBUGS Codes for the artificial example in Chapter 8, Section 8.4.3.

Model: Nonlinear Structural Equation Model with covariates
Data Set Name: YO.dat
Sample Size: N=500

```
model{
    for(i in 1:N){
        for(j in 1:10){y[i,j] <- yy[i,j]}
        x[i] <- yy[i,11]
        # measurement equation model
        for(j in 1:10){
            y[i,j]~dnorm(mu[i,j],psi[j])
            ephat[i,j]<-y[i,j]-mu[i,j]
        }
        mu[i,1]<- u[1]+eta[i]
        mu[i,2]<- u[2]+lam[1]*eta[i]                # lam[1]=lam[2,1]
        mu[i,3]<- u[3]+lam[2]*eta[i]                # lam[2]=lam[3,1]
        mu[i,4]<- u[4]+xi[i,1]
        mu[i,5]<- u[5]+lam[3]*xi[i,1]               # lam[3]=lam[5,2]
        mu[i,6]<- u[6]+lam[4]*xi[i,1]               # lam[4]=lam[6,2]
        mu[i,7]<- u[7]+lam[5]*xi[i,1]               # lam[5]=lam[7,2]
        mu[i,8]<- u[8]+xi[i,2]
        mu[i,9]<- u[9]+lam[6]*xi[i,2]               # lam[6]=lam[9,3]
        mu[i,10]<- u[10]+lam[7]*xi[i,2]             # lam[7]=lam[10,3]
        # structural equation model
        xi[i,1:2] ~ dmnorm(u0[1:2],ph[1:2,1:2])   # u0=[0 0]^T  is a fixed constant vector
        eta[i] ~ dnorm(nu[i], psd)
        nu[i]<-gam[1]*x[i]+gam[2]*xi[i,1]+gam[3]*xi[i,2]+gam[4]*xi[i,1]*xi[i,2]
              +gam[5]*xi[i,1]*xi[i,1]+gam[6]*xi[i,2]*xi[i,2]
                    dthat[i]<-eta[i]-nu[i]
    }
    # priors on loadings and coefficients
    for(j in 1:10){ u[j] ~ dnorm(0.0,1.0) }
    lam[1] ~ dnorm(0.9, psi[2])   lam[3] ~ dnorm(0.9, psi[5])  lam[6] ~ dnorm(0.9, psi[9])
    lam[2] ~ dnorm(0.7, psi[3])   lam[4] ~ dnorm(0.7, psi[6])  lam[7] ~ dnorm(0.7, psi[10])
    lam[5] ~ dnorm(0.5, psi[7])
    gam[1]~dnorm(0.5,psd)   gam[2]~dnorm(0.4,psd)   gam[3]~dnorm(0.4,psd)
    gam[4]~dnorm(0.3,psd)   gam[5]~dnorm(0.2,psd)   gam[6]~dnorm(0.5,psd)
    # priors on precisions
    for(j in 1:10){psi[j] ~ dgamma(9,4)
                    sigma[j]<-1/psi[j]}
    psd ~ dgamma(9,4)
    sigd<-1/psd
    ph[1:2,1:2] ~ dwish(R[1:2,1:2],4)
    phx[1:2,1:2] <- inverse(ph[1:2,1:2])
    # output of parameters
    for(j in 1:10){alpha[j]<- u[j]}
    lambda[2,1]<- lam[1]   lambda[3,1]<- lam[2]    lambda[5,2]<- lam[3]     lambda[6,2]<- lam[4]
    lambda[7,2]<- lam[5]   lambda[9,3]<- lam[6]    lambda[10,3]<- lam[7]
    beta[1]<- gam[1]    for(j in 1:5){ gamma[j]<- gam[1+j] }
    for(j in 1:10){ psiepsilon[j]<- sigma[j] }
    psidelta<- sigd
    phi[1,1]<- phx[1,1]
```

```
    phi[1,2]<- phx[1,2]
    phi[2,2]<- phx[2,2]
}#end of model
```

Data

```
list(N=500, u0=c(0,0),
   R=structure(.Data= c(1.0989, -0.3297, -0.3297, 1.0989),  .Dim= c(2,2)),
   yy=structure(.Data= c(paste YO.dat here), .Dim= c(500,11)))
```

Three different initial values

```
list(u=c(0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1), lam=c(0.7,0.5,0.7,0.5,0.3,0.7,0.5),
      psi=c(0.1, 0.1, 0.1, 0.3, 0.3, 0.3, 0.3, 0.2, 0.2,0.2), psd=0.15,
      gam=c(0.3,0.2,0.2,0.1,0.0,0.1),  ph=structure(.Data=c(0.9, 0.1,0.1,0.9),.Dim= c(2,2)))

list(u=c(0.2,0.2,0.2,0.2,0.2,0.2,0.2,0.2,0.2,0.2), lam=c(0.5,0.3,0.5,0.3,0.1,0.4,0.3),
      psi=c(0.4, 0.4, 0.4, 0.6, 0.6, 0.6, 0.6, 0.5, 0.5,0.5), psd=0.5,
      gam=c(0.6,0.5,0.5,0.4,0.3,0.4),    ph=structure(.Data=c(0.7, 0.4,0.4,0.7), .Dim= c(2,2)))

list(u=c(-0.2,-0.2,-0.2,-0.2,-0.2,-0.2,-0.2,-0.2,-0.2,-0.2), lam=c(1.2,0.9,1.2,0.9,0.7,1.2,0.9),
      psi=c(0.7, 0.7, 0.7, 0.9, 0.9, 0.9, 0.9, 0.8, 0.8,0.8), psd=0.8,
      gam=c(0.8,0.7,0.7,0.6,0.5,0.6),    ph=structure(.Data=c(1.2, 0.5,0.5,1.2), .Dim= c(2,2)))
```