Online appendices A and B to accompany:

Deboeck, P. R., & Preacher, K. J. (in press). No need to be discrete: A method for continuous time mediation analysis. *Structural Equation Modeling*.

Online Appendix A: Code for Plots in R
Online Appendix B: Proof for Equations 7 and 8

Appendix A: Code for Plots in R

The following code is provided to plot the discrete time parameters as a function of lag. The code is written for the statistical program R (R Development Core Team, 2013), and produces a figure similar to Figure 2. Two pieces of information are required to run this program: 1) A continuous time matrix (entered as `A`) and 2) the range of lags to be plotted (`minLag` and `maxLag`). This code is intended for use with a 3x3 continuous time matrix with zero in the upper triangle, consistent with a mediation model. The R function expm() can be used for solving the exponential of a matrix at a particular lag; for example expm(A*2) would solve for the discrete time matrix at a lag of 2, assuming the user has provided the continuous time matrix A.

```
rm(list=ls())
library(Matrix)

#CHANGE THIS TO YOUR “A” MATRIX
A <-   rbind(c(-.357,0,0),
             c(.771,-.511,0),
             c(-.450,.729,-.693))

#CHANGE FOR RANGE OF LAGS DESIRED
minLag <- 0
maxLag <- 8

#____________________________________________________________________
# Calculate Mediation Effects for Differing Lags

Lags <- seq(minLag,maxLag,length.out=500)
Estimates <- matrix(NA,length(Lags),11)
colnames(Estimates) <-
       c("direct","indirect",paste("a",c(1,2,3),rep(c(1,2,3),each=3),sep=""))

Adirect <- A
Adirect[2,1] <- 0
Adirect[3,2] <- 0

Aindirect <- A
Aindirect[3,1] <- 0

for(i in 1:length(Lags)) {
       Estimates[i,] <- c(expm(Adirect*Lags[i])[3,1][[1]],
             expm(Aindirect*Lags[i])[3,1][[1]],as.vector(expm(A*Lags[i])))
       }

#____________________________________________________________________
# Function for drawing minimum/maximum circles

DrawCircle <- function(Lags,y,lty,col,name) {
       if((which(y==max(y))!=1)&(which(y==max(y))!=length(y))) {
             points(Lags[which(y==max(y))],y[which(y==max(y))],col=col,lwd=3,cex=1.1)
             cat("Maximum of", name, "at",round(Lags[which(y==max(y))],2),"\n")
             }
       if((which(y==min(y))!=1)&(which(y==min(y))!=length(y))) {
```

```
            points(Lags[which(y==min(y))],y[which(y==min(y))],col=col,lwd=3,cex=1.1)
            cat("Minimum of", name, "at",round(Lags[which(y==min(y))],2),"\n")
            }
      }

#____________________________________________________________________
# Plot Effects as a function of lag

plot(Lags,Estimates[,5],type="l",main="Relationships Between Constructs",
      xlab="Lag",ylab="Discrete Time Parameter",
      ylim=range(Estimates[,c(1:2,4:5,8)]),
      xlim=c(minLag,maxLag*1.25),lwd=2,col=gray(0))
lines(Lags,Estimates[,8],lty=2,lwd=2,col=gray(.2))
lines(Lags,Estimates[,4],lty=3,lwd=2,col=gray(.4))
lines(Lags,Estimates[,1],lty=4,lwd=2,col=gray(.6))
lines(Lags,Estimates[,2],lty=5,lwd=2,col=gray(.7))

#____________________________________________________________________
# Draw the min/max circles

DrawCircle(Lags,Estimates[,5],lty=1,col=gray(.0),"X-Y total")
DrawCircle(Lags,Estimates[,8],lty=2,col=gray(.2),"M-Y")
DrawCircle(Lags,Estimates[,4],lty=3,col=gray(.4),"X-M")
DrawCircle(Lags,Estimates[,1],lty=4,col=gray(.6),"direct")
DrawCircle(Lags,Estimates[,2],lty=5,col=gray(.7),"indirect")

#____________________________________________________________________
# Draw the horizontal gray line (visual reference)

lines(c(-100,100),c(0,0),col=gray(.7))

#____________________________________________________________________
# Effect Names
# Changing cex=.8 (text size) or adding/subtracting a small constant to
# Estimates (e.g., 0.1+Estimates[]) can help if text is overlapping

text(maxLag,Estimates[dim(Estimates)[1],5],pos=4,"X->Y total",cex=.8)
text(maxLag,Estimates[dim(Estimates)[1],8],pos=4,"M->Y",cex=.8)
text(maxLag,Estimates[dim(Estimates)[1],4],pos=4,"X->M",cex=.8)
text(maxLag,Estimates[dim(Estimates)[1],1],pos=4,"X->Y direct",cex=.8)
text(maxLag,Estimates[dim(Estimates)[1],2],pos=4,"X->Y indirect",cex=.8)
```

Appendix B: **Proof for Equations 7 and 8.**

In Equation 4 it was written that

$$\boldsymbol{A}(\Delta t_i) = \boldsymbol{e}^{\boldsymbol{A}\times\Delta t} \tag{4, repeated}$$

where $\boldsymbol{A}(\Delta t)$ is the discrete time (CLPM) coefficient matrix, $\mathbf{A}$ is the continuous time coefficient matrix, and Δt_i is the lag. As:

$$\exp(\boldsymbol{A}) = \sum_{i=0}^{\infty} \frac{1}{i!}\boldsymbol{X}^i \tag{A1}$$

$$\exp(\boldsymbol{A} \times \Delta t) = \sum_{i=0}^{\infty} \frac{\Delta t^i}{i!}\boldsymbol{A}^i \tag{A2}$$

Abadir and Magnus (2005, p. 368) prove the following for any square matrix $\mathbf{A}$:

$$\frac{\partial}{\partial \Delta t}\exp(\boldsymbol{A} \times \Delta t) = \boldsymbol{A}\exp(\boldsymbol{A} \times \Delta t) \tag{A3}$$

Given a 3x3 **A** matrix, solving for the first derivative yields:

$$\boldsymbol{A}\boldsymbol{e}^{\boldsymbol{A}\times\Delta t} = \begin{bmatrix} a_{xx} & 0 & 0 \\ a_{mx} & a_{mm} & 0 \\ a_{yx} & a_{ym} & a_{yy} \end{bmatrix} \exp\left(\begin{bmatrix} a_{xx} & 0 & 0 \\ a_{mx} & a_{mm} & 0 \\ a_{yx} & a_{ym} & a_{yy} \end{bmatrix} \Delta t\right) \tag{A4}$$

$$= \begin{bmatrix} a_{xx} & 0 & 0 \\ a_{mx} & a_{mm} & 0 \\ a_{yx} & a_{ym} & a_{yy} \end{bmatrix} \begin{bmatrix} e^{a_{xx}\Delta t} & 0 & 0 \\ \dfrac{a_{mx}(e^{a_{mm}\Delta t} - e^{a_{xx}\Delta t})}{a_{mm} - a_{xx}} & e^{a_{mm}\Delta t} & 0 \\ \zeta_{31} & \dfrac{a_{ym}\left(e^{a_{mm}\Delta t} - e^{a_{yy}\Delta t}\right)}{a_{mm} - a_{yy}} & e^{a_{yy}\Delta t} \end{bmatrix}$$

$$\zeta_{31} = \frac{a_{mx}a_{ym}e^{a_{mm}\Delta t}}{(a_{mm} - a_{xx})(a_{mm} - a_{yy})} + \frac{(-a_{mx}a_{ym} + a_{mm}a_{yx} - a_{xx}a_{yx})e^{a_{xx}\Delta t}}{(a_{mm} - a_{xx})(a_{xx} - a_{yy})} + \frac{(-a_{mx}a_{ym} + a_{mm}a_{yx} - a_{yy}a_{yx})e^{a_{yy}\Delta t}}{(a_{xx} - a_{yy})(a_{yy} - a_{mm})}$$

Note that the equation for ζ_{31} corresponds to element 3,1 in the matrix exponentiation of **A** times Δt. The (2,1) element, the relationship between two variables, is:

$$a_{mx}e^{a_{xx}\Delta t} + \frac{a_{mm}a_{mx}\left(e^{a_{mm}\Delta t} - e^{a_{xx}\Delta t}\right)}{a_{mm} - a_{xx}} \tag{A5}$$

Setting this equal to zero and solving for Δt,

$$0 = a_{mx}e^{a_{xx}\Delta t} + \frac{a_{mm}a_{mx}\left(e^{a_{mm}\Delta t} - e^{a_{xx}\Delta t}\right)}{a_{mm} - a_{xx}} \tag{A6}$$

$$0 = \frac{a_{mx}(a_{mm}e^{a_{mm}\Delta t} - a_{xx}e^{a_{xx}\Delta t})}{a_{mm} - a_{xx}} \tag{A7}$$

$$a_{mm}e^{a_{mm}\Delta t} = a_{xx}e^{a_{xx}\Delta t} \tag{A8}$$

$$\ln[a_{mm}] + a_{mm}\Delta t = \ln[a_{xx}] + a_{xx}\Delta t \tag{A9}$$

$$\Delta t = \frac{-\ln[a_{xx}/a_{mm}]}{a_{xx} - a_{mm}} \tag{A10}$$

This result is reported as Equation 7. Equation 8 is the result of substituting Equation 7 into Equation 4 and extracting element (2,1).