Supplementary Material

Here, we provide the R codes related to the simulation study.

```
library(nleqslv)
n = 25 ; alpha = 2 ; theta = 2; sh1=sh2=sh3=sh4=sh5=sh6=0
aM=tM=aB=tB=aLBB=aUBB=tLBB=tUBB=aLpB=aUpB=tLpB=tUpB=c()
aLD=aUD=tLD=tUD=c(); q=1;Q=2000
while(q<=Q){
y=runif(n,0,1);x=c();j=1
while(j<=n){
model=function(u){
v=numeric(1)
v[1]=(1-(1+theta*u[1]/2)*exp(-theta*u[1]))^alpha-y[j]
v
}
  res=nleqslv(c(1),model)
if(abs(res$fvec)>=0.05) sh4=sh4+1
if(abs(res$fvec)< 0.05) { x=c(x,res$x);  j=j+1 }
}
#################
loglike = function(par){
  alpha = par[1];  theta = par[2]
l=n*log(par[1])+n*log(par[2])-par[2]*sum(x)+sum(log(1+par[2]*x))+
  (par[1]-1)*sum(log(1-(1+par[2]*x/2)*exp(-par[2]*x)))
  -l
}
opt=optim(par=c(0.9*alpha,0.9*theta),fn=loglike,method="BFGS",
control = c(maxit=1000000),hessian =TRUE)
if(opt$convergence!=0) sh1=sh1+1
if(opt$par[1]<=0 | opt$par[2]<=0) sh2=sh2+1
if(opt$par[1]>0  & opt$par[2]>0 & opt$convergence==0){
aM[q]=opt$par[1];tM[q]=opt$par[2]; fisher=solve(opt$hessian)
if(fisher[1,1]<=0 |fisher[2,2]<=0) sh3=sh3+1
if(fisher[1,1]>0 & fisher[2,2]>0){
aLD[q]= opt$par[1]*exp(-1.96*sqrt(fisher[1,1])/opt$par[1])
aUD[q]= opt$par[1]*exp( 1.96*sqrt(fisher[1,1])/opt$par[1])
tLD[q]= opt$par[2]*exp(-1.96*sqrt(fisher[2,2])/opt$par[2])
tUD[q]= opt$par[2]*exp( 1.96*sqrt(fisher[2,2])/opt$par[2])
#################################################
```

```
B=999;b=1;aBOOT=tBOOT=c();thetaB=tM[q];alphaB=aM[q]
while(b<=B){
y=runif(n,0,1);xx=c();j=1
while(j<=n){
model=function(u){
    v=numeric(1)
     v[1]=(1-(1+thetaB*u[1]/2)*exp(-thetaB*u[1]))^alphaB-y[j]
     v
  }
  res=nleqslv(c(1),model)
if(abs(res$fvec)>=0.05) sh4=sh4+1
if(abs(res$fvec)< 0.05) { xx=c(xx,res$x);  j=j+1 }
}
loglikeB = function(par){
  alpha = par[1];  theta = par[2]
l=n*log(par[1])+n*log(par[2])-par[2]*sum(xx)+sum(log(1+par[2]*xx))+
  (par[1]-1)*sum(log(1-(1+par[2]*xx/2)*exp(-par[2]*xx)))
  -l
}
optB=optim(par=c(0.9*alphaB,0.9*thetaB),fn=loglikeB,method="BFGS",
control = c(maxit=1000000),hessian =TRUE)
if(optB$convergence!=0) sh5=sh5+1
if(optB$par[1]<=0 | optB$par[2]<=0) sh6=sh6+1
if(optB$par[1]>0  & optB$par[2]>0 & optB$convergence==0){
aBOOT=c(aBOOT,optB$par[1]); tBOOT=c(tBOOT,optB$par[2])
b=b+1
}}
aBoot=sort(aBOOT);tBoot=sort(tBOOT)
aB[q]=mean(aBoot);tB[q]=mean(tBoot)
aLBB[q]=2*aM[q]-aBoot[975];aUBB[q]=2*aM[q]-aBoot[25]
tLBB[q]=2*tM[q]-tBoot[975];tUBB[q]=2*tM[q]-tBoot[25]
aLpB[q]=aBoot[25]; aUpB[q]=aBoot[975]
tLpB[q]=tBoot[25]; tUpB[q]=tBoot[975]
q=q+1
}}}
##################################################
table1=matrix(0,nrow=1,ncol=8)
rownames(table1)=c("")
colnames(table1)=c("MSE.alpha.MLE",  "BIAS.alpha.MLE",
```

```
                  "MSE.alpha.BOOT", "BIAS.alpha.BOOT",
                  "MSE.theta.MLE",  "BIAS.theta.MLE",
                  "MSE.theta.BOOT", "BIAS.theta.BOOT")
table1[1,1]=mean((aM-alpha)^2); table1[1,2]=mean((aM-alpha))
table1[1,3]=mean((aB-alpha)^2); table1[1,4]=mean((aB-alpha))
table1[1,5]=mean((tM-theta)^2); table1[1,6]=mean((tM-theta))
table1[1,7]=mean((tB-theta)^2); table1[1,8]=mean((tB-theta))
table2=matrix(0,nrow=2,ncol=6)
rownames(table2)=c("AW", "CP")
colnames(table2)=c("alpha.DACI",  "alpha.BBCI", "alpha.BpCI",
                  "theta.DACI",  "theta.BBCI", "theta.BpCI")
############################################################
table2[1,1]=mean(aUD-aLD);table2[1,2]=mean(aUBB-aLBB)
table2[1,3]=mean(aUpB-aLpB);table2[1,4]=mean(tUD-tLD)
table2[1,5]=mean(tUBB-tLBB);table2[1,6]=mean(tUpB-tLpB)
table2[2,1]=mean((aLD<alpha)*(alpha<aUD));
table2[2,2]=mean((aLBB<alpha)*(alpha<aUBB))
table2[2,3]=mean((aLpB<alpha)*(alpha<aUpB));
table2[2,4]=mean((tLD<theta)*(theta<tUD))
table2[2,5]=mean((tLBB<theta)*(theta<tUBB));
table2[2,6]=mean((tLpB<theta)*(theta<tUpB))
############################################################
round(table1,4);round(table2,4)
sh=c(sh1,sh2,sh3,sh4,sh5,sh6);sh;alpha;theta;n;q
```