# Unstable States: Community dynamics in a chaotic world

M.K. Lau

June 19, 2014

## 1 19 Jun 2014

The pitcher plant respiration model:

Terms:

- t = time
- x = oxygen
- A = environmental factor promoting oxygenation (i.e. PAR = light)
- f(w,x) = loss/decay of oxygen
- g(x) = recovery of oxygen (augmented by mineralized nutrients)
- w = prey mass

$$\frac{dx}{dt} = A - f(w, x) + g(x) \tag{1}$$

$$x_{t+1} = a_t * sin(2\pi ft) - (m + a_t\frac{w_{t-1}}{K_w + w_{t-1}}) + D_t(x_t, a'_t) \tag{2}$$

```
>   A <- function(a0,f){
+
+   }
```

(3)

```
Here is a link to the pitcher-plant tipping point dropbox. To
start, go to ../Model sensitivity analysis and second paper/data/ and
read the metadata.csv file. The various raw and average files are the
ones to think about plotting. We can talk about it later today.
```

## 2 18 Jun 2014

Question: Are transitions between chaotic states detectable?

Question: Are there classes of models that possess detectable chaotic transition warning signals?

Modeling Issues:

1. a in S2.2 can be confused with a in S2.1

2. What is the exponent in S2.2?

3. Why is b (S2.2) in days rather than minutes?

4. Why is w(t-1) used in S2.3?

```
>   ##Initial stab at the spo2 model
>   a0 <- 10
> f <- 1/(60*24) #total minutes in a day
> t <- rep((1:(60*24)),11)
> At <- a0 * sin(2*pi*f*t)
> At[t>720] <- 0
> plot(At~I(1:length(t)),type='l')
> a <- 20
> b <- 4
> wt <- 20
> wt <- a*exp(1)^(-b*wt)
> spo2 <- function(x,a,f,t,m,w,Kw,s,d){
+   At <- a0 * sin(2*pi*f*t)
+
+
+ }
>
```

# 3 17 Jun 2014

Chatted with Aaron

The data don't support a general pattern of alternative states.

If systems are inherently chaotic or stochastic, then what?

The data:

- Climate isn't stable
- Fossil records don't show stability
- Tropical systems don't show stability
- Even temperate systems break the rule of stability

Two goals:

1. Can we tell when we transition between states?
2. Is there an alternative philosphy? Mathematical framework?

Run simulation increasing r with variance over the chaos threshold

```
>   ##hold ni constant
>   ##hold sd constant at 0.1
>   ##record r
>   ##record n
>   ##record ews
>   source('../src/unstable_states.R')
> library(earlywarnings)
> cb8.16 <- 2.57 #choatic boundary, 8-16 cycles
> ri <- (cb8.16)-(0.02/2)
> rf <- (cb8.16)+(0.02/2)
> dmc <- list()
> ews <- list()
> mcn <- 100
```

```
> for (i in 1:mcn){
+   print(i)
+   dmc[[i]] <- disrupt.mc(N=10,sd=0.01,ri=ri,rf=rf,dump=TRUE)
+   ews[[i]] <- generic_ews(dmc[[i]]$N)
+   dev.off()
+ }
> ## dput(dmc,'../results/dmc.out')
> ## dput(ews,'../results/ews.out')
> ###
> ###
> ###
> dmc <- dget('../results/dmc.out')
> ews <- dget('../results/ews.out')
> ##
> yl <- apply(do.call(rbind,ews),2,min)
> yu <- apply(do.call(rbind,ews),2,max)
> ##
> par(mfrow=c(2,(ncol(ews[[1]])-1)/2))
> for (i in 2:ncol(ews[[1]])){
+   for (j in 1:length(ews)){
+     if (j==1){
+       plot(ews[[j]][,i]~ews[[j]][,1],ylim=c(yl[i],yu[i]),
+            ylab=colnames(ews[[1]])[i],xlab='t',
+            type='l',lwd=0.25)
+     }else{
+       lines(ews[[j]][,i]~ews[[j]][,1],lwd=0.25)
```

```
+     }
+   }
+ }
> ###Average plots
> ews. <- do.call(rbind,ews)
> t <- do.call(rbind,dmc)$t
> r <- do.call(rbind,dmc)$r
> r. <- r
> r[r.>=cb8.16] <- 2
> r[r.<cb8.16] <- 1
> r <- r[t>=ews[[1]][,1][1]]
> #pairs(ews.,cex=0.05,pch=19,col=r)
> par(mfrow=c(1,1))
> plot(ews.[,c(3,4)],pch=19,col=r,cex=(0.01+(0.5*(ews.[,1]/max(ews.[,1])))))
> unique(ews.[ews.[,3]>30.15&ews.[,4]>0.185,1])
> ews. <- apply(ews.,2,function(x,t) tapply(x,t,mean),t=ews.[,1])
> pairs(ews.,cex=0.10,pch=19)
>
```

Reading Hastings Reading Sheffer Reading Dakos

# 4 16 Jun 2014

- the distribution of the average of r is uniform
- ensemble distribution is normal
- EWS stats not correlated between 8to16 and 16to8

- Phase (Ni to Nf) spaces for EWS correlations
- EWS stats intercorrelations show correlations and break points

```
> ###Run repeated simulations for ews time series
> source('../src/unstable_states.R')
> library(earlywarnings)
> cb8.16 <- 2.57 #choatic boundary, 8-16 cycles
> ri <- (cb8.16)-(0.02/2)
> rf <- (cb8.16)+(0.02/2)
> ##Visualizing the error in r
> r8.16 <- list()
> r16.8 <- list()
> for (i in 1:138){
+   print(i)
+   r8.16[[i]] <- disrupt.mc(N=i,sd=0.01,ri=ri,rf=rf,dump=TRUE)$r
+   r16.8[[i]] <- disrupt.mc(N=i,sd=0.01,ri=rf,rf=ri,dump=TRUE)$r
+ }
> rmu8.16 <- apply(do.call(rbind,r8.16),2,mean)
> rmu16.8 <- apply(do.call(rbind,r16.8),2,mean)
> par(mfrow=c(1,2))
> plot(density(rmu8.16),main='',xlab='r')
> for (i in 1:length(r8.16)){
+   lines(density(r8.16[[i]]),col='grey',lwd=0.5)
+ }
> abline(v=cb8.16,lty=2);abline(v=mean(rmu8.16),lty=2,col='darkgrey')
> plot(density(rmu16.8),xlab='r',main='')
```

```
> for (i in 1:length(r16.8)){
+   lines(density(r16.8[[i]]),col='grey',lwd=0.5)
+ }
> abline(v=cb8.16,lty=2);abline(v=mean(rmu16.8),lty=2,col='darkgrey')
> ###Determine average threshold
> ##What is the point at which rbar has crossed the threshold?
> ##Directionality depends on direction of r
> rt8.16 <- (1:length(rmu8.16))[rmu8.16>=cb8.16][1]
> rt16.8 <- (1:length(rmu16.8))[rmu16.8<=cb8.16][1]
> rt8.16
> rt16.8
> ###EWS stats
> stats8.16 <- dget(file='../results/stats816.rdata')
> stats16.8 <- dget(file='../results/stats168.rdata')
> ###
> stats8.16 <- na.omit(do.call(rbind,stats8.16))
> stats16.8 <- na.omit(do.call(rbind,stats16.8))
> ###
> stats8.16 <- stats8.16[1:min(c(nrow(stats8.16),nrow(stats16.8))),]
> stats16.8 <- stats16.8[1:min(c(nrow(stats8.16),nrow(stats16.8))),]
> ###
> par(mfrow=c(2,ncol(stats8.16)/2),
+     mai=c(0.25,0.01,0.25,0.01))
> for (i in 1:ncol(stats8.16)){
+   plot(density(stats8.16[,i]),
+        main=colnames(stats8.16)[i],
```

```
+        xlim=c(min(c(stats8.16[,i],stats16.8[,i])),
+          max(c(stats8.16[,i],stats16.8[,i]))),
+        xaxt='n',yaxt='n',bty='n')
+   lines(density(stats16.8[,i]),lty=2)
+ }
> ###
> par(mfrow=c(2,ncol(stats16.8)/2))
> for (i in 1:ncol(stats8.16)){
+   plot(stats16.8[,i]~stats8.16[,i],
+        xlab=paste(colnames(stats8.16)[i],'8.16'),
+        ylab=paste(colnames(stats16.8)[i],'16.8'))
+   abline(lm(stats16.8[,i]~stats8.16[,i]))
+ }
> ###
> par(mfrow=c(4,ncol(stats8.16)/2),
+     mai=c(0,0,0,0))
> for (i in 1:ncol(stats8.16)){
+   plot(stats8.16[1:(nrow(stats8.16)-1),i]~stats8.16[2:(nrow(stats8.16)),i],
+        type='l',xaxt='n',yaxt='n',bty='n')
+ }
> for (i in 1:ncol(stats16.8)){
+   plot(stats16.8[1:(nrow(stats16.8)-1),i]~stats16.8[2:(nrow(stats16.8)),i],
+        type='l',col='darkgrey',xaxt='n',yaxt='n',bty='n')
+ }
> ###Ensemble N~stats
> pairs(data.frame(Ni=(1:nrow(stats8.16)),stats8.16),pch=19,cex=0.10,col='black')
```

```
> pairs(data.frame(Ni=(1:nrow(stats16.8)),stats16.8),cex=0.10,col='black')
>

> library(earlywarnings)
> set.seed(1)
> drmc8.16 <- disrupt.mc(sd=0.01,ri=ri,rf=rf,dump=TRUE)
> set.seed(1)
> drmc16.8 <- disrupt.mc(sd=0.01,ri=rf,rf=ri,dump=TRUE)
> ews8.16 <- generic_ews(drmc8.16$N)
> ews16.8 <- generic_ews(drmc16.8$N)
> stats8.16 <- cor(ews8.16,method='ken')
> stats16.8 <- cor(ews8.16,method='ken')
>
>

>   ##Re-doing noise in r shifting up and down across 8-16
> source('../src/unstable_states.R')
> cb8.16 <- 2.57 #choatic boundary, 8-16 cycles
> ri <- (cb8.16)-(0.02/2)
> rf <- (cb8.16)+(0.02/2)
> set.seed(1)
> drmc8.16 <- disrupt.mc(sd=0.01,ri=ri,rf=rf,dump=TRUE)
> set.seed(1)
> drmc16.8 <- disrupt.mc(sd=0.01,ri=rf,rf=ri,dump=TRUE)
> par(mfrow=c(3,2))
> plot(drmc8.16$r,xlab='time',ylab='r')
> abline(h=cb8.16,col=2)
```

```
> plot(drmc16.8$r,xlab='time',ylab='r')
> abline(h=cb8.16,col=2)
> plot(drmc16.8$N,xlab='time',ylab='N')
> plot(drmc8.16$N,xlab='time',ylab='N')
> plot(drmc16.8$N~drmc16.8$r,xlab='r',ylab='N',type='l')
> plot(drmc8.16$N~drmc8.16$r,xlab='r',ylab='N',type='l')
> ###Phase space
> par(mfrow=c(1,2))
>                                             #plus n time steps
> for (n in 25:50){
+   plot(drmc16.8$N[(1:(length(drmc16.8$N)-n))],drmc16.8$N[((n+1):(length(drmc16.8$
+        xlab='N',ylab='N+1',type='l')
+   plot(drmc8.16$N[(1:(length(drmc8.16$N)-n))],drmc8.16$N[((n+1):(length(drmc8.16$
+        xlab='N',ylab='N+1',type='l')
+ }
> #EWS
> library(earlywarnings)
> ews8.16 <- generic_ews(drmc8.16$N)
> ews16.8 <- generic_ews(drmc16.8$N)
>
```

Summary to date (going back in time):

- At the 8-16 cycle threshold, error in r leads to early, sudden shifts
- Slow ramping can also be seen visually
- Sudden jumps across cycle boundaries can be seen visually