Appendices

Christine San Antonio

May 17, 2017

#### Appendix A - Supplemental Methods - Experiment Procedure

**Laboratory Setup**

To perform this experiment, I used the pH-stat carbon dioxide dosing system at UMass Boston, which consists of 24 10-gallon aquarium tanks each equipped with airstones for bubbling ambient air and carbon dioxide, separate solenoids that regulate the CO2 dosing, and individual tank heaters that maintain temperature. The system operates using a hobbyist aquarist Digital Aquatics Reef Keeper Elite system with separate SL1 units for each tank that measure and log pH and temperature data continuously; slight variations in pCO2 levels within one tank trigger the solenoids to switch on and dose until the programmed pH setpoint is reached (Wilcox-Freeburg et al 2013). As this was a joint project with the New England Aquarium, the juvenile American lobsters were reared to stages VII-X in the aquarium's lobster hatchery and then transferred to the aquarium system at UMass Boston. Lobsters were gently wrapped "burrito style" with seawater soaked paper towels and layered within small conatainers kept within a cooler. The lobsters were allowed a three-day acclimation period upon being placed in the system tanks and prior to any experimental conditions being applied. Carapace length measurements were recorded (average of three measurements) for each lobster before being placed in the tanks.

Initial aquarium water consisted of 75% artificial seawater (Instant Ocean sea salts mixed with DI water) and 25% filtered natural Boston Harbor seawater supplied by the New England Aquarium to help establish a biofilter, to help minimize stress as the lobsters acclimated to the system, and to help ensure the ubiquitous bacteria was present and became established within the system. Instant ocean was mixed in a 1000 L tank with a submersible pump to achieve a consistent salinity of 32.0 ppt; water was then transferred to an intermediate 600 L reservoir connected to each tank through a PVC piping system. The intermediate reservoir allowed for greater control over salinity and temperature as well as minimizing the time needed for water changes, reducing stress on the lobsters. Active bioballs were also kept in the intermediate reservoir to help maintain a biofilter throughout the experiment.

To ensure the system read consistently accurate pH values, the seawater millivolts were measured from each tank daily and compared to the system pH readout. pH electrodes that were > 0.03 off target were calibrated using synthetic seawater buffers, TRIS and Aminopyridine, made fresh prior to the experiment (Dickson et al 2007). Seawater alkalinity from each tank was also acquired daily using an open-cell titrator, which was calibrated before every use. Temperature and salinity from each tank were also measured daily using a YSI probe; the temperature values were compared to the values being recorded by the system, which was calibrate as needed. At the completion of the experiment, mean pH, alkalinity, temperature, and salinity values for each tank were input into the analytical program, ORNL/CDIAC-105 CO2SYS (<http://cdiac.ornl.gov/oceans/co2rprt.html>) to calculate the corresponding pCO2, DIC, and saturation state of aragonite for each tank (Pierrot et al 2006).

#### Appendix B - Other Model Output

All models fit and converged with very similar postchecks (all within range of sigma).

##4. Build Models - treatment, temp, pH, temp + pH  
set.seed(601)  
  
#4A. Model 1: Predictors = treatment indexed + growth  
esd\_mod\_1 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ bT[treat\_idx] + bG\*growth,  
   
 #Priors  
 bT[treat\_idx] ~ dnorm(0,10),  
 bG ~ dnorm(0,10),  
 phi ~ dcauchy(0,2)  
)  
  
#4B. Model 2: Predictors = temperature + growth  
esd\_mod\_2 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ b\_temp[temp\_idx] + bG\*growth,  
   
 #Priors  
 b\_temp[temp\_idx] ~ dnorm(0,10),  
 bG ~ dnorm(0,10),  
 phi ~ dcauchy(0,2)  
)  
  
#4C. Model 3: Predictors = pH + growth  
esd\_mod\_3 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ b\_ph[ph\_idx] + bG\*growth,  
   
 #Priors  
 b\_ph[ph\_idx] ~ dnorm(0,10),  
 bG ~ dnorm(0,10),  
 phi ~ dcauchy(0,2)  
)  
  
#4D. Model 4: Predictors = pH + temp + growth  
esd\_mod\_4 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ b\_temp[temp\_idx] + b\_ph[ph\_idx] + bG\*growth,  
   
 #Priors  
 b\_temp[temp\_idx] ~ dnorm(0,10),  
 b\_ph[ph\_idx] ~ dnorm(0,10),  
 bG ~ dnorm(0,10),  
 phi ~ dcauchy(0,2)  
)

set.seed(601)  
  
#4E. Treatment predictor - growth random effect  
esd\_mod\_5 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ a\_growth\*growth + bT[treat\_idx],  
 a\_growth ~ dnorm(0, sigma\_growth),  
   
 #Priors  
 phi ~ dcauchy(0,2),  
 sigma\_growth ~ dcauchy (0,2),  
 bT[treat\_idx] ~ dnorm(0,10)  
)  
  
#4F.Temperature predictor - growth random effect  
esd\_mod\_6 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ a\_growth\*growth + b\_temp[temp\_idx],  
 a\_growth ~ dnorm(0, sigma\_growth),  
   
 #Priors  
 phi ~ dcauchy(0,2),  
 sigma\_growth ~ dcauchy (0,2),  
 b\_temp[temp\_idx] ~ dnorm(0,10)  
)  
  
#4G. pH predictor - growth random effect  
esd\_mod\_7 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ a\_growth\*growth + b\_ph[ph\_idx],  
 a\_growth ~ dnorm(0, sigma\_growth),  
   
 #Priors  
 phi ~ dcauchy(0,2),  
 sigma\_growth ~ dcauchy (0,2),  
 b\_ph[ph\_idx] ~ dnorm(0,10)  
)  
  
#4H. Temp and pH predictor - growth random effect  
esd\_mod\_8 <- alist(  
 #likelihood - dbeta2  
 esd\_ratio ~ dbeta2(mu, phi),  
   
 #DGP - logit link   
 logit(mu) ~ a\_growth\*growth + b\_temp[temp\_idx] + b\_ph[ph\_idx],  
 a\_growth ~ dnorm(0, sigma\_growth),  
   
 #Priors  
 phi ~ dcauchy(0,2),  
 sigma\_growth ~ dcauchy (0,2),  
 b\_temp[temp\_idx] ~ dnorm(0,50),  
 b\_ph[ph\_idx] ~ dnorm(0,10)  
)

##5. Fit models and Evaluate  
  
#5A.  
set.seed(601)  
  
esd\_fit\_1 <- map2stan(esd\_mod\_1, data = esd,  
 iter=3000, chains = 3)

set.seed(601)  
  
#5B.  
esd\_fit\_2 <- map2stan(esd\_mod\_2, data = esd,  
 iter=3000, chains = 3)

set.seed(601)  
  
#5C.  
esd\_fit\_3 <- map2stan(esd\_mod\_3, data = esd,  
 iter=3000, chains = 3)

set.seed(601)  
  
#5D.  
esd\_fit\_4 <- map2stan(esd\_mod\_4, data = esd,  
 iter=3000, chains = 3)

set.seed(601)  
  
#5E.  
esd\_fit\_5 <- map2stan(esd\_mod\_5, data = esd,  
 iter=3000, chains = 3)

set.seed(601)  
  
#5F.  
esd\_fit\_6 <- map2stan(esd\_mod\_6, data = esd,  
 iter=3000, chains = 3)

set.seed(601)  
  
#5G.  
esd\_fit\_7 <- map2stan(esd\_mod\_7, data = esd,  
 iter=3000, chains = 3)

set.seed(601)  
  
#5H.  
esd\_fit\_8 <- map2stan(esd\_mod\_8, data = esd,  
 iter=3000, chains = 3)

##6. Evaluating all fits - Postchecks, plot convergence, precis output, WAIC  
  
postcheck(esd\_fit\_1, par(mfrow = c(2,3)))  
postcheck(esd\_fit\_2, par(mfrow = c(2,3)))  
postcheck(esd\_fit\_3, par(mfrow = c(2,3)))  
postcheck(esd\_fit\_4, par(mfrow = c(2,3)))  
  
postcheck(esd\_fit\_5, par(mfrow = c(3,2)))  
postcheck(esd\_fit\_6, par(mfrow = c(2,3)))  
postcheck(esd\_fit\_7, par(mfrow = c(2,3)))  
postcheck(esd\_fit\_8, par(mfrow = c(2,3)))  
  
plot(esd\_fit\_1)  
plot(esd\_fit\_2)  
plot(esd\_fit\_3)  
plot(esd\_fit\_4)  
  
plot(esd\_fit\_5)  
plot(esd\_fit\_6)  
plot(esd\_fit\_7)  
plot(esd\_fit\_8)  
  
precis(esd\_fit\_1, depth = 2)  
precis(esd\_fit\_2, depth = 2)  
precis(esd\_fit\_3, depth = 2)  
precis(esd\_fit\_4, depth = 2)  
  
precis(esd\_fit\_5, depth = 2)  
precis(esd\_fit\_6, depth = 2)  
precis(esd\_fit\_7, depth = 2)  
precis(esd\_fit\_8, depth = 2)  
  
coeftab(esd\_fit\_1, esd\_fit\_2, esd\_fit\_3, esd\_fit\_4, esd\_fit\_5, esd\_fit\_6, esd\_fit\_7, esd\_fit\_8)  
compare(esd\_fit\_1, esd\_fit\_2, esd\_fit\_3, esd\_fit\_4, esd\_fit\_5, esd\_fit\_6, esd\_fit\_7, esd\_fit\_8)  
  
plot(compare(esd\_fit\_1, esd\_fit\_2, esd\_fit\_3, esd\_fit\_4, esd\_fit\_5, esd\_fit\_6, esd\_fit\_7, esd\_fit\_8))