# Supplementary Materials 1 to: Wojczulanis-Jakubas K (2021) "Being the winner is being the loser when playing a parental tug-of-war - a new framework on stability of biparental care" (to be updated with journal details)

# This is a short script being a core of the application:

# https://kasiawojczulanis-jakubas.shinyapps.io/biparental\_care\_gametheory/

# You can use to explore the inside content of the application,

# and/or to caluculate "exact" values for the cost of parental care and probability of breeding success

# To run the script, just copy all and paste in a new R-script file, then

# set up the values in the lines 12 (index) and 13 (scenario\_procent) manually (respecting indicated range)

# and run simply execute the whole code - observe plots and values in the console

index <- 15 # manually entered (range 2-20)

scenario\_procent <- 20 # manually entered (range from: 10-90)

if(!"ggplot2" %in% installed.packages()[,"Package"]) install.packages("ggplot2")

library(ggplot2)

scenario <- 100/scenario\_procent

# Figure one -----

 # Data

 mx <- 10

 # basic sets

 female\_good <- seq(0,mx,1)

 male\_good <- rev(female\_good)

 female\_poor <- seq(0,mx+index,1)

 male\_poor <- rev(female\_poor)

 df\_good <- data.frame(female\_good, male\_good)

 df\_poor <- data.frame(female\_poor, male\_poor)

 # game set

 suboptimal\_range1 <- median(female\_poor) - median(female\_good)

 female\_env\_val <- max(female\_good)+suboptimal\_range1/scenario

 female\_env <- seq(0,female\_env\_val,1)

 male\_env <- rev(female\_env)

 df\_env <- data.frame(female\_env, male\_env)

# Plot

x11()

ggplot() +

 # rectangles

 annotate("rect", xmin = 0, xmax = median(female\_poor), ymin = 0, ymax = median(male\_poor), fill = "grey", alpha = 0.8) +

 # basic lines

 geom\_line(data = df\_good, aes(x = female\_good, y = male\_good), col = "lightsteelblue3", size = 2) + # max good

 geom\_line(data = df\_poor, aes(x = female\_poor, y = male\_poor), col = "lightsalmon3", size = 2) + # max bad

 # dashed lines

 annotate("segment", x = median(female\_poor), xend = median(female\_poor), y = 0, yend = median(male\_poor), linetype = "dashed", size = 0.5) +

 annotate("segment", x = 0, xend = median(female\_poor), y = median(male\_poor), yend = median(male\_poor), linetype = "dashed", size = 0.5) +

 # points

 annotate("point", x = median(female\_poor), y = median(male\_poor), size = 3) +

 annotate("point", x = median(female\_poor), y = 0, size = 3) +

 annotate("point", x = 0, y = median(male\_poor), size = 3) +

 geom\_line(data = df\_env, aes(x = female\_env, y = male\_env), linetype = "dashed", col = "black", size = 0.8) + # env

 theme\_classic() +

 scale\_x\_continuous(expand = c(0,0), limits = c(0,30), name = "Female effort") +

 scale\_y\_continuous(expand = c(0,0), limits = c(0,30), name = "Male effort") +

 theme(axis.ticks = element\_blank(),

 axis.text = element\_blank(),

 axis.title = element\_text(size = 12))

# Figure two -----

 capacity <- seq(1,100,1)

 cost <- capacity^index

 success <- -capacity^index

 range01 <- function(x){(x-min(x))/(max(x)-min(x))}

 norm\_cost <- range01(cost)

 norm\_succ <- range01(success)

 df <- data.frame(capacity, cost, success, norm\_cost, norm\_succ)

 n <- length(cost)-1

 gr\_out <- numeric(n)

 for(i in 1:n){

 gr\_out[i] <- (max(cost)-cost[i])/max(cost)

 }

 vl <- min(which(gr\_out<0.995))

 # values for plot2

 female\_effort\_env <- max(female\_env) #100% of deserted female effort

 female\_effort\_maxopt <- median(female\_poor)

 # if TRUE

 if(female\_effort\_env<female\_effort\_maxopt) {

 percent\_effort <- female\_effort\_env/female\_effort\_maxopt

 capacity\_value <- ceiling(percent\_effort\*vl)

 values\_df <- df[df$capacity == capacity\_value,]

 values\_df$success\_mod <- max(cost)+values\_df$success

 }

 # if FALSE

 if(female\_effort\_env>=female\_effort\_maxopt) {

 subop\_are\_onset <- max(female\_poor) - median(female\_poor)

 female\_excess <- female\_effort\_env-female\_effort\_maxopt

 female\_excess\_percent <- female\_excess/subop\_are\_onset

 female\_exces\_cap <- ceiling(vl + max(capacity)\*female\_excess\_percent)

 values\_df <- df[df$capacity == female\_exces\_cap, ]

 values\_df$success\_mod <- max(cost)+values\_df$success

 }

# Plot 2

x11()

ggplot() +

 scale\_x\_continuous(expand = c(0,0), name = "Parental capacity") +

 scale\_y\_continuous(name = "Cost of parental care (solid) \n Probability of breeding success") +

 theme\_bw() +

 geom\_rect(data = df, aes(xmin = 0 , xmax = vl,

 ymin = min(cost), ymax = max(cost)), fill = "lightsteelblue3") +

 geom\_rect(data = df, aes(xmin = vl , xmax = max(capacity),

 ymin = min(cost), ymax = max(cost)), fill = "lightsalmon3") +

 geom\_line(data = df, aes(x = capacity, y = cost), size = 1.1) +

 geom\_line(data = df, aes(x = capacity, y = max(cost) + success),

 linetype = "dashed", size = 1.1) +

 annotate(geom = "text", x = 1, y = min(cost)-(max(cost)/30), label="breeders",

 color = "black", hjust = 0) +

 geom\_hline(aes(yintercept = min(cost)-(max(cost)/10)), size = 1.1, color = "grey") +

 annotate(geom = "text", x = 1, y = min(cost)-(max(cost)/14), label="nonbreeders",

 color = "darkgrey", hjust = 0) +

 theme(panel.grid = element\_blank(),

 axis.ticks = element\_blank(),

 axis.text = element\_blank(),

 axis.title = element\_text(size = 12)) +

 geom\_point(data = values\_df, aes(x = capacity, y = cost), shape = 19, color = "black", size = 4) +

 geom\_point(data = values\_df, aes(x = capacity, y = success\_mod), shape = 18, color = "black", size = 4)

# Specific values for a deserted parent:

# cost of parental care

parcost <- round(values\_df$norm\_cost,4)

# probability of breeding success

breedsucc <- round(values\_df$norm\_succ,4)

cat(paste("In species with evolutionary tolerance for environmental variability set up on value of ", index,

" and the environmental scenario set up on value of ", scenario,

" for a parent that has been deserted by the partner and so extending its parental effort,",

"the cost of parental care is: ", parcost,

" and the probability of breeding success is: ", breedsucc, sep = ""))