Systems of DE's for Resource Allocation Strategies in Annual Social Insects

#Differential Equation Systems

```
#########################
#### Standard System ####
#########################
SDE_std <- function(t, state, parameters){
  with(as.list(c(state, parameters)),{
          tt <- t/Ttol
          Dw <- Dw4*(Dw1*(tt-1)^2 + Dw2*(1-4*(tt-.5)^2) + Dw3*tt^2)
          Br <- Br4*(Br1*(tt-1)^2 + Br2*(1-4*(tt-.5)^2) + Br3*tt^2)

          dWorkers <- -Dw*Workers+Cw*(1-Br)*Energy
          dReproductives <- Cr*Br*Energy-Dr*Reproductives
          dEnergy <- Kw*Workers+Kr*Reproductives-Energy

          list(c(dWorkers, dReproductives, dEnergy))
  })}
#########################
### Varying Resources ###
#########################
SDE_varK <- function(t, state, parameters){
  with(as.list(c(state, parameters)),{
          tt <- t/Ttol
          Dw <- Dw4*(Dw1*(tt-1)^2 + Dw2*(1-4*(tt-.5)^2) + Dw3*tt^2)
          Br <- Br4*(Br1*(tt-1)^2 + Br2*(1-4*(tt-.5)^2) + Br3*tt^2)
          Kw <- Kf4*(Kf1*(tt-1)^2 + Kf2*(1-4*(tt-.5)^2) + Kf3*tt^2)

          dWorkers <- -Dw*Workers+Cw*(1-Br)*Energy
          dReproductives <- Cr*Br*Energy-Dr*Reproductives
          dEnergy <- Kw*Workers+Kr*Reproductives-Energy
          list(c(dWorkers, dReproductives, dEnergy))
  })}

############################
### Varying Preservation ###
############################
SDE_varE <- function(t, state, parameters){
  with(as.list(c(state, parameters)),{
          tt <- t/Ttol
          Dw <- Dw4*(Dw1*(tt-1)^2 + Dw2*(1-4*(tt-.5)^2) + Dw3*tt^2)
          Br <- Br4*(Br1*(tt-1)^2 + Br2*(1-4*(tt-.5)^2) + Br3*tt^2)
          Pe <- Pe4*(Pe1*(tt-1)^2 + Pe2*(1-4*(tt-.5)^2) + Pe3*tt^2)

          dWorkers <- -Dw*Workers+Cw*(1-Br)*Energy*(1-Pe)
          dReproductives <- Cr*Br*Energy*(1-Pe)-Dr*Reproductives
```

```
          dEnergy <- Kw*Workers+Kr*Reproductives-(1-Pe)*Energy
          list(c(dWorkers, dReproductives, dEnergy))
  })}

##########################
### Bang-Bang Strategy ###
##########################
SDE_bang <- function(t, state, parameters){
  with(as.list(c(state, parameters)),{
          tt <- t/Ttol
          Dw <- Dw4*(Dw1*(tt-1)^2 + Dw2*(1-4*(tt-.5)^2) + Dw3*tt^2)
          if(tt < critval)   {Br <- 0}   else  {Br <- 1}

          dWorkers <- -Dw*Workers+(1-Br)*Cw*Energy
          dReproductives <- Cr*Br*Energy-Dr*Reproductives
          dEnergy <- Kw*Workers+Kr*Reproductives-Energy

          list(c(dWorkers, dReproductives, dEnergy))
  })}
############################
### Bang-Bang w/ Vary K ###
############################
SDE_bangvarK <- function(t, state, parameters){
  with(as.list(c(state, parameters)),{
          tt <- t/Ttol
          Dw <- Dw4*(Dw1*(tt-1)^2 + Dw2*(1-4*(tt-.5)^2) + Dw3*tt^2)
          if(tt < critval)   {Br <- 0}   else  {Br <- 1}

          Kw <- Kf4*(Kf1*(tt-1)^2 + Kf2*(1-4*(tt-.5)^2) + Kf3*tt^2)
          dWorkers <- -Dw*Workers+Cw*(1-Br)*Energy
          dReproductives <- Cr*Br*Energy-Dr*Reproductives
          dEnergy <- Kw*Workers+Kr*Reproductives-Energy

          list(c(dWorkers, dReproductives, dEnergy))
  })}
```

#Parameters

```
env <- c( 0.4 , 0.5 , 0.5 , 0.8 )
col <- c( 0.1 , 0.1 , 0.9 , 0.4 )
Kf0 <- c( 1.0 , 0.75, 0.15, 1.7)
Pe0 <- c( 0.0 , 0.5 , 0.0 , 1.0 )

parameters <- c(Dw1 = env[1], Dw2 = env[2], Dw3 = env[3], Dw4 = env[4],
                Br1 = col[1], Br2 = col[2], Br3 = col[3], Br4 = col[4],
                Kf1 = Kf0[1], Kf2 = Kf0[2], Kf3 = Kf0[3], Kf4 = Kf0[4],
                Pe1 = Pe0[1], Pe2 = Pe0[2], Pe3 = Pe0[3], Pe4 = Pe0[4],
                Cw = 1/3, Cr = 1/4, Kr = 0, Kw = 2, Dr = .01, Ttol = 100, critval=0.9)

times <- seq(0, 100, by = .1)

state <- c(Workers = 0, Reproductives = 1, Energy = 1)
```

#Standard System

```
SDE_std_sol <- ode(y = state, times = times, func = SDE_std, parms = parameters)

SDE_std_sol_df <- melt(as.data.frame(SDE_std_sol), id.vars = "time")

plot1a <- ggplot(SDE_std_sol_df, mapping = aes(x=time,y=value, color=variable)) + geom_line() +
  scale_colour_manual(values=c("blue", "red", "green")) + ylab("Value") + xlab(" ") +
  ggtitle("Standard System")

############################
#### Parameter Plot 1 ####
############################

p <- as.list(parameters); t <- seq(0, 1, length.out=1001)
Dw <- p$Dw4*(p$Dw1*(t-1)^2 + p$Dw2*(1-4*(t-.5)^2) + p$Dw3*t^2)
Br <- p$Br4*(p$Br1*(t-1)^2 + p$Br2*(1-4*(t-.5)^2) + p$Br3*t^2)
plot1df <- data.frame(time = rep(t,2),value = c(Dw,Br),
                      type=c(rep("Mortality",length(t)),rep("Birth Rate",length(t))))

plot1b <- ggplot(data = plot1df, mapping = aes(x=time,y=value, linetype=type)) + geom_line() +
  ylim(c(0,1)) + xlab("") + ylab(" ") +
  ggtitle("Parameter Values")
```

#Varying Resource Availability

```
SDE_varK_sol <- ode(y = state, times = times, func = SDE_varK, parms = parameters)

SDE_varK_sol_df <- melt(as.data.frame(SDE_varK_sol), id.vars = "time")

plot2a <- ggplot(SDE_varK_sol_df, mapping = aes(x=time,y=value, color=variable)) + geom_line() +
  scale_colour_manual(values=c("blue", "red", "green")) + ylab("Value") + xlab(" ") +
  ggtitle("Varying Resource Availability")

############################
#### Parameter Plot 2 ####
############################
p <- as.list(parameters)
Dw <- p$Dw4*(p$Dw1*(t-1)^2 + p$Dw2*(1-4*(t-.5)^2) + p$Dw3*t^2)
Br <- p$Br4*(p$Br1*(t-1)^2 + p$Br2*(1-4*(t-.5)^2) + p$Br3*t^2)
Kw <- p$Kf4*(p$Kf1*(t-1)^2 + p$Kf2*(1-4*(t-.5)^2) + p$Kf3*t^2)

plot2df <- data.frame(time = rep(t,3),value = c(Dw,Br,Kw),
                  type=c(rep("Forager Mortality",length(t)),
                         rep("Reproductive Allocation",length(t)),
                         rep("Resource Availability",length(t))))

plot2b <- ggplot(data = plot2df, mapping = aes(x=time,y=value, linetype=type)) + geom_line() + #ylim(c(
  xlab("") + ylab(" ") +
  ggtitle("Parameter Values")
```

#Varying Energy Preservation

```
SDE_varE_sol <- ode(y = state, times = times, func = SDE_varE, parms = parameters)

SDE_varE_sol_df <- melt(as.data.frame(SDE_varE_sol), id.vars = "time")
```

```
plot3a <- ggplot(SDE_varE_sol_df, mapping = aes(x=time,y=value, color=variable)) + geom_line() +
  scale_colour_manual(values=c("blue", "red", "green")) + ylab("Value") + xlab(" ") +
  ggtitle("Varying Energy Preservation")

##########################
#### Parameter Plot 3 ####
##########################
p <- as.list(parameters)
Dw <- p$Dw4*(p$Dw1*(t-1)^2 + p$Dw2*(1-4*(t-.5)^2) + p$Dw3*t^2)
Br <- p$Br4*(p$Br1*(t-1)^2 + p$Br2*(1-4*(t-.5)^2) + p$Br3*t^2)
Pe <- p$Pe4*(p$Pe1*(t-1)^2 + p$Pe2*(1-4*(t-.5)^2) + p$Pe3*t^2)

plot3df <- data.frame(time = rep(t,3),value = c(Dw,Br,Pe),
                      type=c(rep("Forager Mortality",length(t)),
                             rep("Reproductive Allocation",length(t)),
                             rep("Energy Preservation",length(t))))

plot3b <- ggplot(data = plot3df, mapping = aes(x=time,y=value, linetype=type)) +
  geom_line() + xlab("") + ylab(" ") + ylim(c(0,1)) +
  ggtitle("Parameter Values")
```

#Parameter Variation - Varying Resource Availability

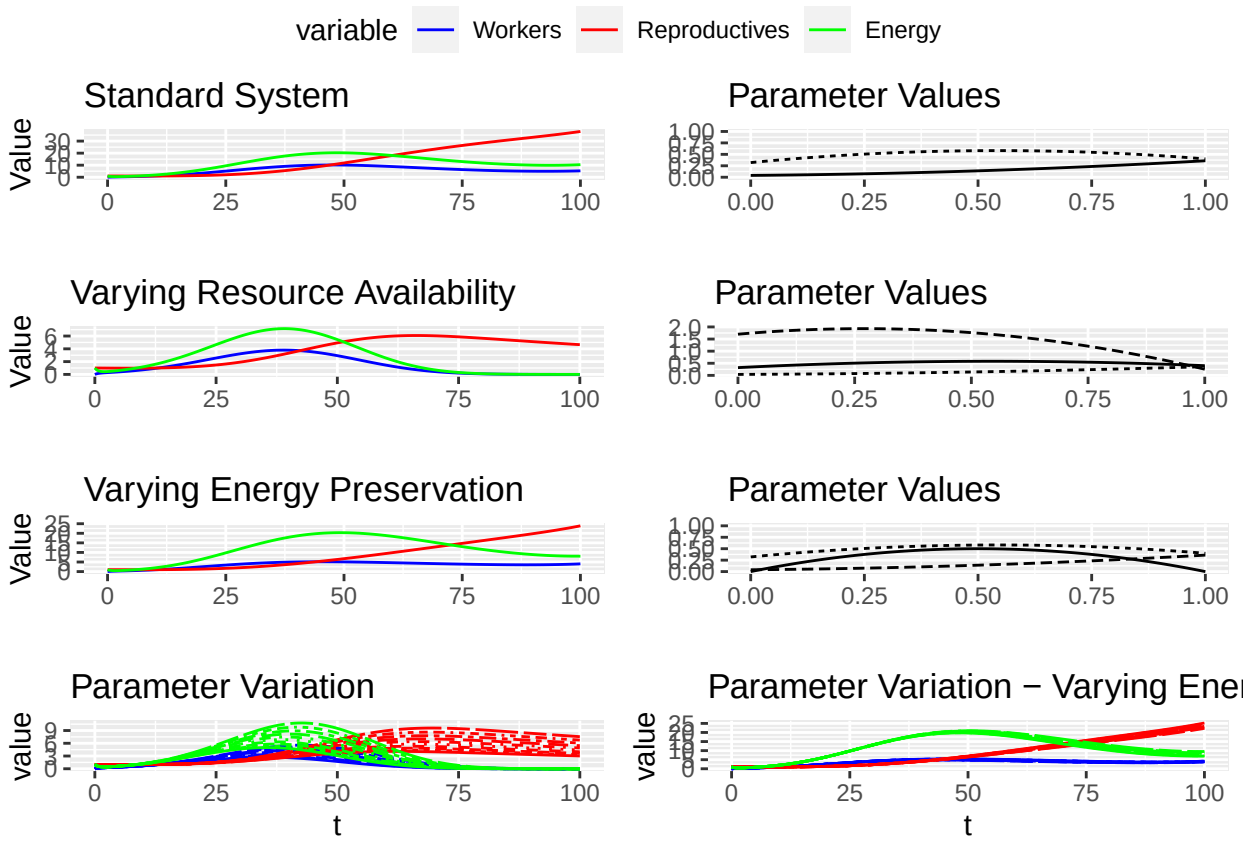
```
n    <- 10
vary <- "Kf2"
range<- 0.1

param.vals <- seq(from = parameters[vary]-range/2,to = parameters[vary]+range/2,length.out = n)
SDE_sol_list <- list(); p <- list(); paramplot_df <- data.frame();
temp <- ode(y = state, times = times, func = SDE_varK, parms = parameters)
dfDEsol3 <- melt(as.data.frame(temp), id.vars = "time")[,c(1,2)]

for (i in 1:n)
{
  parameters[vary] <- param.vals[i]
  paramplot_df <- rbind(paramplot_df,data.frame(as.list(parameters)))
  SDE_tempsol <- ode(y = state, times = times, func = SDE_varK, parms = parameters)
  SDE_sol_list[[i]] <- melt(as.data.frame(SDE_tempsol), id.vars = "time")
  dfDEsol3 <- cbind(dfDEsol3,SDE_sol_list[[i]][,3])
}

names(dfDEsol3) <- c("t","var",as.character(1:n))
forplot <- melt(dfDEsol3,id.vars = c("t","var"))

#for(j in 1:n)
#{p[[j]] <- ggplot(SDE_sol_list[[j]]) +
#  geom_line(mapping = aes(x=time,y=value,color=variable))}

plot4 <- ggplot(forplot) + geom_line(aes(x=t,y=value,color=var,linetype = variable)) +
  scale_colour_manual(values=c("blue", "red", "green")) + ggtitle("Parameter Variation")
```

#Parameter Variation - Varying Energy Provisioning

```
n    <- 10
vary <- "Pe3"
```

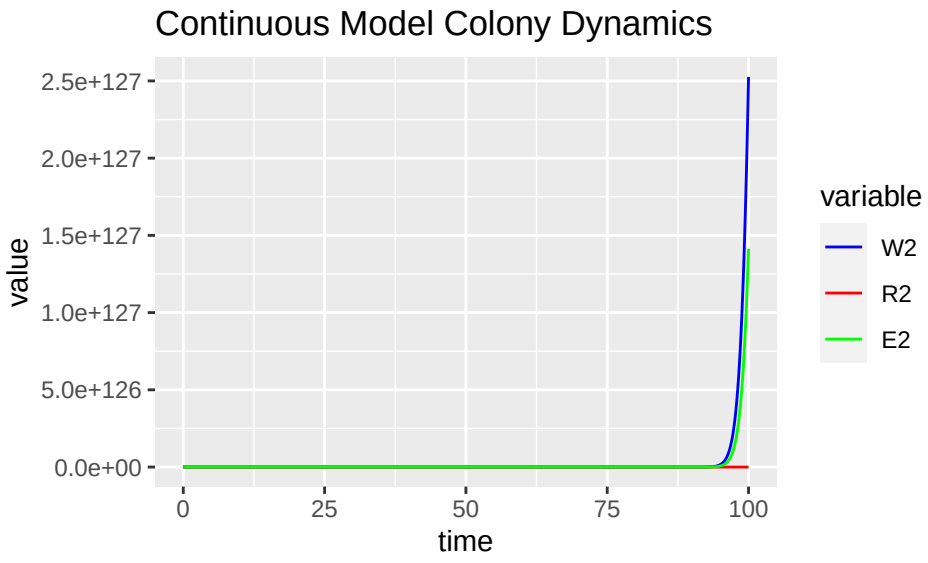
```
range<- 0.4

param.vals <- seq(from = parameters[vary]-range/2,to = parameters[vary]+range/2,length.out = n)
SDE_sol_list <- list(); p <- list();
temp <- ode(y = state, times = times, func = SDE_varE, parms = parameters)
dfDEsol3 <- melt(as.data.frame(temp), id.vars = "time")[,c(1,2)]

for (i in 1:n)
{
  parameters[vary] <- param.vals[i]
  SDE_tempsol <- ode(y = state, times = times, func = SDE_varE, parms = parameters)
  SDE_sol_list[[i]] <- melt(as.data.frame(SDE_tempsol), id.vars = "time")
  dfDEsol3 <- cbind(dfDEsol3,SDE_sol_list[[i]][,3])
}

names(dfDEsol3) <- c("t","var",as.character(1:n))
forplot <- melt(dfDEsol3,id.vars = c("t","var"))

#for(j in 1:n)
#{p[[j]] <- ggplot(SDE_sol_list[[j]]) +
#  geom_line(mapping = aes(x=time,y=value,color=variable))}

plot5 <- ggplot(forplot) + geom_line(aes(x=t,y=value,color=var,linetype = variable)) +
  scale_colour_manual(values=c("blue", "red", "green")) + ggtitle("Parameter Variation - Varying Energy
```

```
ggarrange(plot1a,plot1b,plot2a,plot2b,plot3a,plot3b,plot4,plot5,ncol=2,nrow=4,common.legend=TRUE)
```



#Non-dimensionalization

```
# env <- c( 0.1 , 0.1 , 1, 0.1)
# col <- c( 0.1 , 0.5 , 0.5, .1)
# Kf0 <- c( 0.01 , 0.01 , 0.1, 0.001)
#
# state <- c(W2 = 0,R2 = .1, E2 = .1)
# parameters <- c(Dw1 = env[1], Dw2 = env[2], Dw3 = env[3], Dw4 = env[4],
#                 Br1 = col[1], Br2 = col[2], Br3 = col[3], Br4 = col[4],
#                 Kf1 = Kf0[1], Kf2 = Kf0[2], Kf3 = Kf0[3], Kf4 = Kf0[4],
#                 Cw = 3.5, Cr = 5, Pe = 0, Dr = .01, Kr = 0, Ttol = 100,
#                 critval = .6)
#
times <- seq(0, 100, by = .1)

state_nondim <- c(W2 = 0,R2 = .1, E2 = .1)
#times_nondim <- times*parameters["Dr"]
#parameters["Ttol"] <- parameters["Ttol"]*parameters["Dr"]

SDE_nondim <- function(t, state, parameters){
  with(as.list(c(state, parameters)),{
          tt <- t/Ttol
          Dw <- Dw4*(Dw1*(tt-1)^2 + Dw2*(1-4*(tt-.5)^2) + Dw3*tt^2)
          Br <- Br4*(Br1*(tt-1)^2 + Br2*(1-4*(tt-.5)^2) + Br3*tt^2)
          Kw <- Kf4*(Kf1*(tt-1)^2 + Kf2*(1-4*(tt-.5)^2) + Kf3*tt^2)

          w <- ((1-Br)*Kw)/((Cw)*(Dw^2))
          r <- (Br*Kr)/((Cr)*(Dw^2))

          dW2 <- w*E2-W2
          dR2 <- r*E2-R2
          dE2 <- W2+R2-E2

          list(c(dW2, dR2, dE2))
})}

SDE_nondim_sol <- ode(y = state_nondim, times = times, func = SDE_nondim, parms = parameters)

SDE_nondim_sol_df <- melt(as.data.frame(SDE_nondim_sol), id.vars = "time")

ggplot(SDE_nondim_sol_df, mapping = aes(x=time,y=value, color=variable)) + geom_line() +
  scale_colour_manual(values=c("blue", "red", "green")) +
  ggtitle("Continuous Model Colony Dynamics")
```


Continuous Model Colony Dynamics
2.5e+127
2.0e+127
1.5e+127
1.0e+127
5.0e+126
0.0e+00
value
0
25
50
75
100
time
variable
W2
R2
E2