concordance=TRUE

1

# Deterministic compartmental model for erythropoeisis

### Dr Suman Kumar

### March 28, 2015

# Contents

# 1 Model specification

## 1.1 Introduction

Erythropoeisis is the process by which Red Blood Cells (RBCs) are formed from the mother cells (Stem Cells) inside the bone marrow. RBCs contain Hemoglobin, oxygen carrier and are destroyed in the spleen and other reticulo-endothelial cells after a pre-determined time period due to shortening of their cell membranes.

The model aims to describe the process of erythropoeisis in terms of differential equations. After validating this model, we can combine various systems of erythropoeisis to understand dynamics of competing erythropoesis as seen in post bone marrow transplant settings.

The model is deterministic one with delayed differential equations (DDE). It is a compartmental model with following compartments:

1. Stem cells/BFUs (Burst Forming Units) **(S)**

2. CFUs (Colony Forming Units) **(C)**

3. Precursors **(P)**

4. Mature RBCs **(R)**

5. Hemoglobin **(H)**

6. There are two additional state variables:

   (a) New entrants into P compartment **(newP)**
   (b) New entrants into R compartment **(newR)**

## 1.2 S compartment

S compartment cells are very small in numbers. They are autonomous, with long term chances of self replication, albeit at a slow rate. They die with $1^{st}$ order kinetics. They continuously supply cells for next compartment (C).

The process can be explained by following equation.

$$S'(t) = (ss - sc - sd) \cdot S(t) \tag{1}$$

where,

**ss:** rate of $S \rightarrow S$

**sc:** rate of $S \rightarrow C$

**sd:** rate of $S \rightarrow Death$

## 1.3 C compartment

C compartment cells are more in numbers. They can again self replicate, can form next lineage of cells and can die. They follow $1^{st}$ order kinetics for all the three situations. The death of C cells is in negative feedback loop with hemoglobin concentration (which lead to oxygenation).

The whole process can be expressed as follows:

$$cd(t) = h \cdot H(t) \tag{2}$$

$$C'(t) = sc \cdot S(t) + [cc - cp - cd(t)] \cdot C(t) \tag{3}$$

where,

3

**h:** determinant of $C \rightarrow Death$

**cc:** rate of $C \rightarrow C$

**cp:** rate of $C \rightarrow P$

**cd:** rate of $C \rightarrow Death$

### 1.3.1 Modification of cd(t)

On running the simulations, it is found that the feedback mechanism is not functioning properly. There is too much of initial fluctuations after which the values stabilises to normal levels.

The conceivable problem is in equation 2, in which the inhibition is only related to Hb. It is not considering the fact that the value cannot be more than the remaining C compartment cells. There has to be some sort of ceiling in death of C compartment cells.

It should also be a feasible hypothesis that the upper value of cd should be constrained by some value of Hb.

In real life situation, Hb regulates C cells death by regulating erythropoeitin (Epo). Epo helps in reducing C compartment cell deaths. At Hb = 0, there would be a maximum value of Epo secretion ($E_{max}$), which will not completely prevent deaths, it will only minimise deaths ($cd_{min}$ ¿ 0). As Hb increases, oxygenation improves, leading to decreasing levels of Epo. After a certain value of Hb ($Hb_{thresh}$), blood viscosity increases, leading to decrease in oxygenation, thereby preventing any further decrease in Epo (I donot know if it leads to increase in Epo levels). Another hypothesis would be that, as the Hb reaches a threshold ($Hb_{thresh}$), Epo production stops ($E_{min} \rightarrow 0$). After that $Hb_{thresh}$, the cd will not increase further, thereby reaching a sort of ceiling.

Equation 2 can be modified as follows:

$$cd'(t) = h \cdot H(t) \cdot [H_{thresh} - H(t)] \tag{4}$$

where,

$\mathbf{H_{thresh}}$ : Threshold Hb value above which there is no further increase in cd

## 1.4 P compartment

P compartment is a bit more complicated. Cells will proliferate and *after a delay of time*, they will differentiate into mature R compartment cells. The cells die with $1^{st}$ order kinetics. This part of model will need DDE. The process can be expressed as follows:

$$newP'(t) = cp \cdot C(t) - newP(t) \tag{5}$$

$$pr(t) = \frac{newP(t - \tau_p) \cdot e^{\tau_p \cdot pp}}{\int_0^{\tau_p} newP(t - u) \cdot e^{u \cdot pp} \cdot du} \tag{6}$$

4

$$P'(t) = newP(t) + [pp - pd - pr(t)] \cdot P(t) \qquad (7)$$

where,

$\tau_{\mathbf{p}}$: lag period after which $P \to R$. $newP(t) = 0$, for $t \leq \tau_p$

**pr:** rate of $P \to R$

**pp:** rate of $P \to P$

**pd:** rate of $P \to Death$

## 1.5   R compartment and H

R compartment cells do not proliferate, but only *die after a lag time*. They carry hemoglobin r per cell. As depicted in equations 2 and 3, total hemoglobin (H) negatively regulates the death of C compartment cells. The process can be depicted as follows:

$$newR'(t) = pr(t) \cdot P(t) - newR(t) \qquad (8)$$

$$R'(t) = newR(t) - newR(t - \tau_r) \qquad (9)$$

$$H(t) = rh \cdot R(t) \qquad (10)$$

where,

$\tau_{\mathbf{r}}$: lag period after which $R \to Death$. $newR(t) = 0$, for $t \leq \tau_r$

**pr:** rate of $P \to R$. Vide equation 6

**rh:** conversion factor $R \to H$

# 2   R code

## 2.1   Parameters

```
param.s <- c(sc = 0.01,
             sd = 0.01,
             ss = 0.02)
param.c <- c(cp = 0.1,
             cc = 0.05,
             h = 0.1)
param.p <- c(pp = 0.1,
             pd = 0.05,
             tau.p = 5)
param.r <- c(tau.r = 100,
             rh = 1e-3)
```

```
tf <- 500
PARAMETERS <- c(param.s,param.c,param.p,param.r,tf)
```

## 2.2   State variables

```
yini <- c(S = 1e5,
       C = 0,
       P = 0,
       R = 0,
       newP = 0,
       newR = 0)
```

## 2.3   Integrator function

```
foo <- function(t, y, param,...) {
  with(as.list(c(y, param)), {
  dS <- (ss - sc - sd) * S
  cd <- rh * h * R
  dC <- (sc * S) + (cc - cp - cd) * C
  dnewP <- (cp * C) - newP
  tlagP <- t - as.integer(seq(tau.p,to=0))
  lagnewP <- rep(0,length.out=length(tlagP))
  for (i in 1:length(tlagP)) {
    if (tlagP[i] <= 0) {
      lagnewP[i] <- 0
    } else {
      lagnewP[i] <- lagvalue(tlagP[i],5)
    }
  }
  sum.int <- sum(lagnewP*exp(pp*(tau.p:0)))
  sum.int <- ifelse(sum.int == 0, 1, sum.int)
  pr <- (lagnewP[1]*exp(pp*tau.p))/sum.int
  dP <- newP + (pp - pd - pr) * P
  tlagR <- t - as.integer(tau.r)
  dnewR <- (pr * P) - newR
  if (tlagR > 0) {
    lagnewR <- lagvalue(tlagR,6)
  } else {
    lagnewR <- 0
  }
  dR <- newR - lagnewR
  return(list(c(dS, dC, dP, dR, dnewP, dnewR), Hb = rh*R))
  })
}
```

## 2.4 Optional event data

```
eventdat <- data.frame(var = c('S','S'), ...
time = c(200,300),value = c(0.5,2), method = c("mult","mult"))
```

## 2.5 DDE solver

```
TIME <- seq(from  = 0, to = tf, by = 0.1)
sol <- dede(y = yini, times = TIME, ...
func = foo, parms = PARAMETERS,events=list(data=eventdat))
```

## 2.6 Plottings

```
plot(sol)
```

## 2.7 Diagnostics

```
diagnostics(sol)
```

# 3 Simulation

## 3.1 Setting up initial parameters and state variables

S compartment parameters are set, so that to maintain a steady outflow into C compartment and stable S population. C compartment death is a dynamic process which increases with increasing H. C proliferation rate is about 5 times of that of S compartment. In contrast to S compartment (in which self proliferation is 2 times that of differentiation), C compartment is more into differentiation into P rather than self proliferation. P compartment has much higher proliferation rate, with a lag time of 5 days before it converts into R compartment.

The usual normal values of few of the parameters are as follows:

- RBC = $5 \cdot 10^6 / \mu L$

- Hb = 15 g/dL = $15 \cdot 10^{-4}$ g/$\mu$L

- MCH (Hb present in one RBC) = 30 pg = $30 \cdot 10^{-12}$ g

- We can calculate, 1 RBC is equivalent to $\mathbf{3 \cdot 10^{-10}}$ g of Hb.

```r
param.s <- c(sc = 0.01,      # S to C
             sd = 0.01,      # Death
             ss = 0.02)      # S to S
param.c <- c(cp = 0.1,       # C to P
             cc = 0.05,      # C to C
             h = 0.1)        # Controls Deaths in C comp
param.p <- c(pp = 0.1,       # P to P
             pd = 0.05,      # Death
             tau.p = 5)   # time after which P differentiates into R
param.r <- c(tau.r = 100,    # time after which R dies
             rh = 1e-3,
             hmax = 8)       # Factor for converting R to H
tf <- 1000
PARAMETERS <- c(param.s,param.c,param.p,param.r,tf)



#----STATE VARIABLES----------

yini <- c(S = 1e5,   # Stem cell/BFU compartment
          C = 0,    # CFU compartment
          P = 0,    # Precursor compartment
          R = 0,    # RBC compartment
          newP = 0,
          newR = 0)
          #cd = 0)
#------WORK HORSE FUNCTION-----------


foo <- function(t, y, param,...) {
  with(as.list(c(y, param)), {

    if (t >= 500) tau.r <- 60
    dS <- (ss - sc - sd) * S
    #dcd <- (rh * h * R * (hmax - (rh*R))) - cd    # Determinant of death in C compartment.
    cd <- rh * h * R
    dC <- (sc * S) + (cc - cp - cd) * C
    dnewP <- (cp * C) - newP  # new P entrant at time t
    #if (t > 0) browser()
    tlagP <- t - as.integer(seq(tau.p,to=0))
    lagnewP <- rep(0,length.out=length(tlagP))
    for (i in 1:length(tlagP)) {
      if (tlagP[i] <= 0) {
        lagnewP[i] <- 0
      } else {
        lagnewP[i] <- lagvalue(tlagP[i],5)
      }
```

```
    }
    sum.int <- sum(lagnewP*exp(pp*(tau.p:0)))
    sum.int <- ifelse(sum.int == 0, 1, sum.int)
    pr <- (lagnewP[1]*exp(pp*tau.p))/sum.int
    dP <- newP + (pp - pd - pr) * P
    tlagR <- t - as.integer(tau.r)
    dnewR <- (pr * P) - newR
    #if (lagvalue(tlagR,4) > 0) browser()
    if (tlagR > 0) {
      lagnewR <- lagvalue(tlagR,6)
    } else {
      lagnewR <- 0
    }
    #if (lagR > 0) browser()
    dR <- newR - lagnewR

    return(list(c(dS, dC, dP, dR, dnewP, dnewR), Hb = rh*R))  #, dcd
  })
}


#-------EVENT DATA----------------
#eventdat <- data.frame(var = c('S','S'), time = c(200,300),value = c(0.5,2), method = c("m



#-------DIFFERENTIAL EQUATION--------------

TIME <- seq(from  = 0, to = tf, by = 1)
sol <- dede(y = yini, times = TIME, func = foo, parms = PARAMETERS) #,events=list(data=even

plot(sol)
```
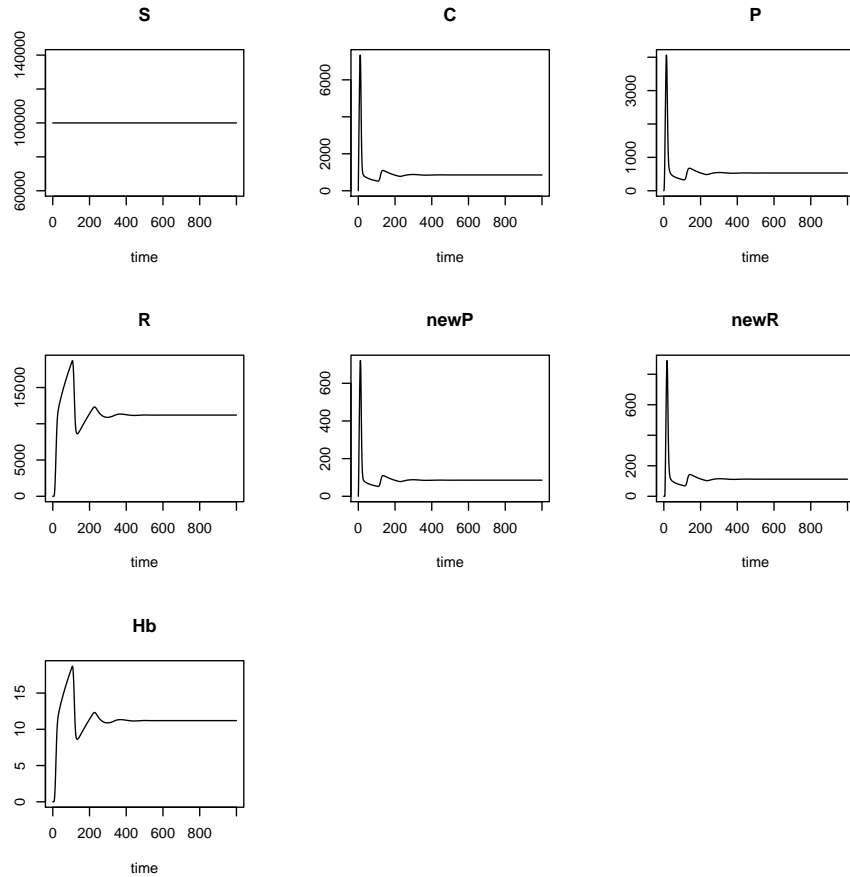
# 4 Acknowledgement

My sincerest thanks and gratitude to the following:

The above work is completed by me over last 1 year based on my understanding on erythropoeisis as acquired by reading textbooks and articles over years. This work has not been substantiated by any experiment.

The user of this model is responsible for the consequences of running and using the simulation.

The user is free to use, modify, distribute, further develop the model and publish the results coming out of the simulation after informing the author and citing the following web address.