Chapter 12
COMPREHENSIVE MODELS FOR FELDSPARS

Abstract In this chapter we provide R codes for four different models previously developed for feldspars, apatites and other materials exhibiting quantum tunneling phenomena. These are the ground state tunneling (GST), irradiation-GST model (IGST), excited state tunneling model (EST) and thermally-assisted tunneling model (TA-EST). We demonstrate appropriate R functions which can simulate a wide variety of processes in feldspars, for both natural and laboratory irradiated samples. We present specialized codes for analyzing CW-IRSL and TL signals from freshly irradiated samples, as well as for simulating a variety of multiple stage experiments, involving thermal and optical pretreatments of samples in the laboratory. The chapter concludes with several R examples for the TA-EST model, which can be used for low temperature thermochronology studies.

Table 12.1: The various FSF functions used in this chapter. The first column indicates the model for which each function can be used.

Model	THE FSF FUNCTIONS
GST	***AFfortimeT(tim)*** Sets parameter *distr* at the end of the anomalous fading period *tim* (in s)
IGST	***irradfortimeT(tirr)*** Sets parameter *distr* at the end of the irradiation time *timCW* (in s)
EST	***CWfortimeT(timCW)*** Sets parameter *distr* at the end of the IR stimulation time *timCW* (in s)
EST	***CWsignal(timCW)*** Evaluates and and returns the CW-IRSL signal
EST	***stimIRSL()*** Calls *CWfortimeT* and *CWsignal*; sets *distr* and returns the CW-IRSL signal
EST	***heatTo(Tph)*** Sets parameter *distr* at the end of preheating to temperature Tph (in °C)
EST	***heatAt(Tph,tph)*** Sets parameter *distr* at the end of preheating for time *tph* (in s) at a temperature *Tph* (in °C)
EST	***TLsignal(temp)*** Evaluates and and returns the TL signal at temperatures *temp* (in °C)
EST	***stimTL()*** Calls functions *heatTo* and *TLsignal,* sets *distr* and returns the TL signal
TA-EST	***irradandThermalfortimeT(tirr)*** Sets parameter *distr* at the end of irradiation time *tirr* (in s), for a fixed sample temperature (*Tirr*)
TA-EST	***irradattemp(Tirr)*** Sets parameter *distr* at the end of irradiation temperature *Tirr* (in °C), for a fixed irradiation time (*tirr*)

Code 12.1: The nearest neighbor distribution at geological times

```
# The nearest neighbor distribution at geological times
rm(list=ls())
s<-3e15                 # frequency factor
rho<-1e-6               # rho-prime values 0.005-0.02
rc<-0.0                 # for freshly irradiated samples, rc=0
timesAF<-3.154e7*c(0,1e2,1e4,1e6)          # times in seconds
rprimes<-seq(from=rc,to=2.2,by=0.002)    # rprime=0-2.2

##### function to find distribution of distances ###
AFfortimeT<-function(tim){3*(rprimes**2.0)*exp(-(rprimes**3.0))*
    exp(-exp(-(rho**(-1/3))*rprimes)*s*tim)}
######
```

```
distribs<-sapply(timesAF,AFfortimeT)
# Plots
cols=c(NA,NA,NA,"black","red","green","blue")
matplot(rprimes,distribs,xlab="Dimensionless distance r'",
ylab="Nearest neighbor Distribution g(r')",type="l",lwd=4)
legend("topright",bty="n", lty=c(NA,NA,NA,1,2,3,4), lwd=4,
col=cols,legend = c("Elapsed", "time"  ," ","t=0 s",
 expression("10"^"2"*" years"),
expression("10"^"4"*" years"),expression("10"^"6"*" years")))
```

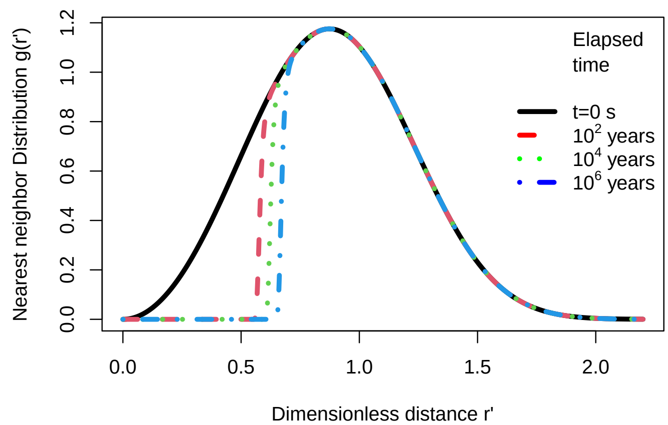

Fig. 12.1: Examples of the nearest neighbor distribution at different times $t = 0, 10^2, 10^4, 10^6$ years. The solid black line represents the *unfaded* nearly symmetric distribution at time $t = 0$. As time increases, the "tunneling front" is the almost vertical line which moves to the right, as more and more electrons are recombining at different distances r'. See also the discussion in Chapter 6.

Code 12.2: Anomalous fading at geological and laboratory times

```
# Anomalous fading  over geological and laboratory times
rm(list = ls(all=T))
rho<-1e-6                    # Dimensionless acceptor density
dr<-.01                      #Step in dimensionless distance r'
rprimes<-seq(0,2.2,dr)       #Values of r'=0-2.2 in steps of dr
```

```
s<-3e15
seff<-s*exp(-rprimes*(rho**(-1/3.0)))  # Effective A

###################### Functions  ################

###################### Anomalous fading Functions
#### AFfortimeT
AFfortimeT<-function(timAF){distr<<-distr*exp(-(seff*timAF))}
###################### End of Functions

########################## Simulations ###############
par(mfrow=c(1,2))
#### Example : Anomalous fading
# Long term fading  0-10^4 years in nature
distr<<-3*rprimes^2*exp(-rprimes^3)
timesAF<-3.154e7*c(.1,.2,.5,1,2,5,10,20,50,100,200,500,1000,
2000,5000,1e4)
n<-dr*colSums(sapply(timesAF,AFfortimeT))
plot(timesAF/(3.154e7),100*n,typ="p",lwd=2,pch=1,col="red",
     ylim=c(70,110),xlab=expression("Time [years]"),
     ylab="Remaining charge [%]")
legend("topleft",bty="n",legend=c("(a)"," ","AF in nature",
"Analyt. Eq."),lwd=2,lty=c(NA,NA,NA,1), col=c(NA,NA,NA,"blue"))
lines(timesAF/3.154e7,y=100*exp(-rho*(log(1.8*s*timesAF)**3.0)),
   lwd=2,col="blue")
### Repeat for short term fading 0-10 days in lab
distr<<-3*rprimes^2*exp(-rprimes^3)  # unfaded distribution
timesAF<-3600*24*c(1e-4,1e-3,1e-2,.1,.2,.5,seq(1,10,.5))
n<-dr*colSums(sapply(timesAF,AFfortimeT))
plot(timesAF/(3600*24),100*n,typ="p",lwd=2,pch=1,col="red",
     ylim=c(70,110),xlab=expression("Time [days]"),
     ylab="Remaining charge [%]")
legend("topleft",bty="n",legend=c("(b)"," ","AF in Lab",
 "Remaining charge","0-10 days"))
lines(timesAF/(3600*24),
y=100*exp(-rho*(log(1.8*s*timesAF)**3.0)),lwd=2,col="blue")
```

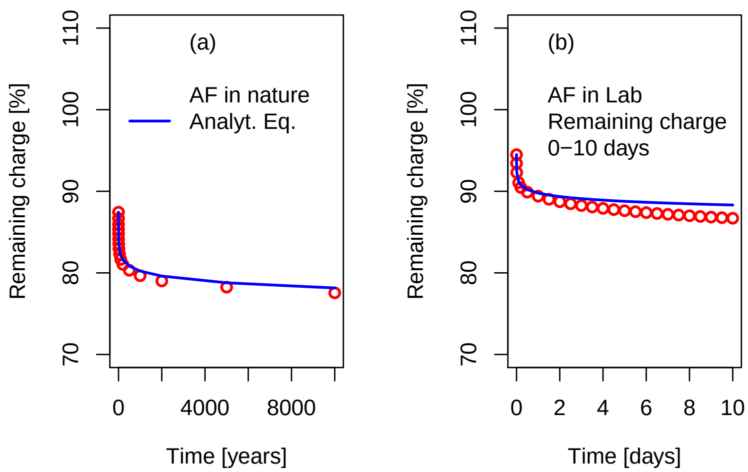

Fig. 12.2: (a) Simulation of long term anomalous fading in nature over a time period of 10^4 years, starting with an unfaded sample. The solid line indicates the approximate analytical Eq.(**??**). (b) Short term AF in the laboratory, over a period of 10 days after the end of irradiation. The parameters in the model are typical for feldspars.

Code 12.3: Feldspar irradiation in nature

```
## Distribution of distances for feldspar irradiation in nature
rm(list = ls(all=T))
rho<-2e-6                    # Dimensionless acceptor density
Do<-538                      #Do in Gy
yr<-365*24*3600              #year is seconds
Ddot<-3/(1e3*yr)             #Low natural dose rate = 3 Gy/kA
dr<-.04                      #Step in dimensionless distance r'
rprimes<-seq(0.01,2.2,dr) #Values of r'=0-2.2 in steps of dr
s<-2e15                      # Frequency factors in s^-1
seff<-s*exp(-rprimes*(rho**(-1/3.0)))  # Effective s
tau<-1/seff
###################### Irradiation Function
#### irradfortimeT
irradfortimeT<-function(tirr){distr<<-unFaded*(Ddot*tau/
(Do+Ddot*tau))*  (1-exp(-(Do+Ddot*tau)/(Do*tau)*tirr))}
# function calculates new distribution at end of irradiation
```

```
###################### End of Functions
unFaded<-3*rprimes^2*exp(-rprimes^3)  # Unfaded sample
irrTimes<-c(6.67e4,1.67e5,1e6)*yr
distribs<-sapply(irrTimes,irradfortimeT)
########## plot distibutions
matplot(rprimes,distribs,xlab="Dimensionless Distance r'",
typ="o",lty="solid", ylab="Distribution of r'",pch=c(2,3,4),
lwd=2,col=c("blue","green","red"))
lines(rprimes,unFaded,col="black",pch=1,typ="p",lwd=2)
legend("topright",bty="n",legend=c(
  expression("Natural Irradiation",
"Unfaded","tirr=6.7x10"^4*" y", "tirr=1.7x10"^5*" y",
"Field saturation")),  pch=c(NA,1,2,3,4),
col=c(NA,"black","blue","green","red"),lwd=2)
```

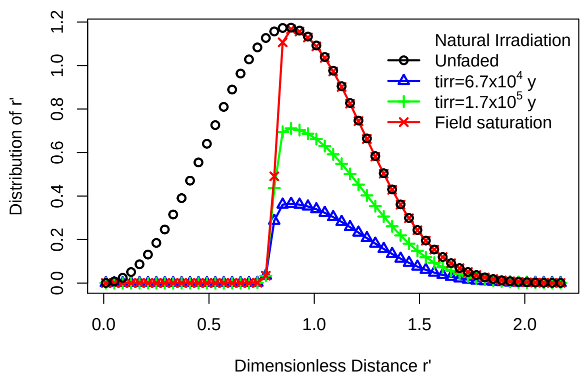

Fig. 12.3: Simulation of irradiation process in nature. As the irradiation time increases, the asymmetric distribution of distances r' approaches the field saturation distribution ($\times$ symbols). The symmetric curve indicates the initial distribution of distances, for the unfaded sample (o symbols). Compare with the results of Li and Li [23].

Code 12.4: Feldspar irradiation in nature- Dose response

```
## Feldspar irradiation in nature  - Dose response
rm(list = ls(all=T))
rho<-2e-6                    # Dimensionless acceptor density
```

```
s<-3e15                  # Frequency factors in s^-1
Do<-538                    #Do in Gy
yr<-365*24*3600            #year is seconds
Ddot<-2.85/(1e3*yr)        #Low natural dose rate = 2.85 Gy/Ka
dr<-.05                    #Step in dimensionless distance r'
rprimes<-seq(0.01,2.2,dr) #Values of r'=0-2.2 in steps of dr
seff<-s*exp(-rprimes*(rho**(-1/3.0)))  # Effective s
tau<-1/seff
###################### Irradiation Function
#### irradfortimeT
irradfortimeT<-function(tirr){distr<<-unFaded*(Ddot*tau/
  (Do+Ddot*tau))*  (1-exp(-(Do+Ddot*tau)/(Do*tau)*tirr))}
# function calculates new distribution at end of irradiation
###################### End of Function ##########
par(mfrow=c(1,2))
unFaded<-3*rprimes^2*exp(-rprimes^3)  # Unfaded sample
irrTimes<-10^seq(2,6,by=.2)*yr
distribs<-sapply(irrTimes,irradfortimeT)
########## plot distibutions
matplot(rprimes,distribs,typ="l",ylim=c(0,1.8),lty="solid",
xlab="Dimensionless Distance r'",  ylab="Distribution of r'",
lwd=2)
legend("topleft",bty="n",legend=c(expression("(a)",
"Irradiation in Nature","Distributions of r'",
"tirr=100-10"^"6"*"y")))
plot(irrTimes/yr,colSums(distribs)*dr,typ="p",lwd=2,
xlab="Time [y]",ylab="Trap filling ratio n(t)/N",ylim=c(0,1.1))
legend("topleft",bty="n",legend=c("(b)"," ",
"Trap filling n(t)/N",  "Analyt. Eq."),lwd=2,
lty=c(NA,NA,NA,1),pch=c(NA,NA,1), col=c(NA,NA,NA,"red"))
lines(irrTimes/yr,(1-exp(-Ddot*irrTimes/Do))*exp(-rho*
(log(Do*s/Ddot)**3.0)),lwd=2,col="red")
```

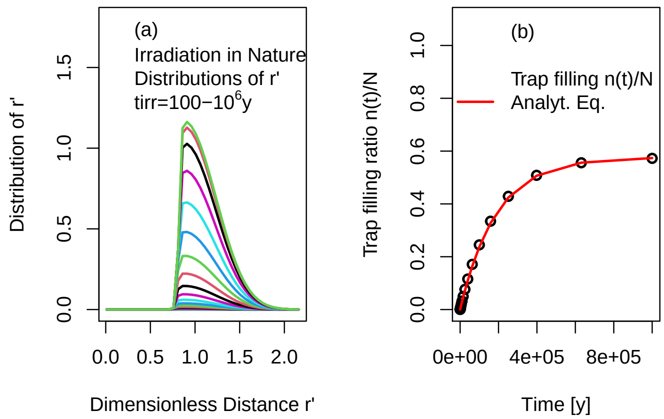

Fig. 12.4: Simulation of irradiation process in nature with a slow dose rate of 2.85 Gy/ka. As the irradiation time increases, the asymmetric distribution of distances r' approaches the field saturation distribution. The symmetric curve indicates the initial distribution of distances, for the unfaded sample (o symbols).

F

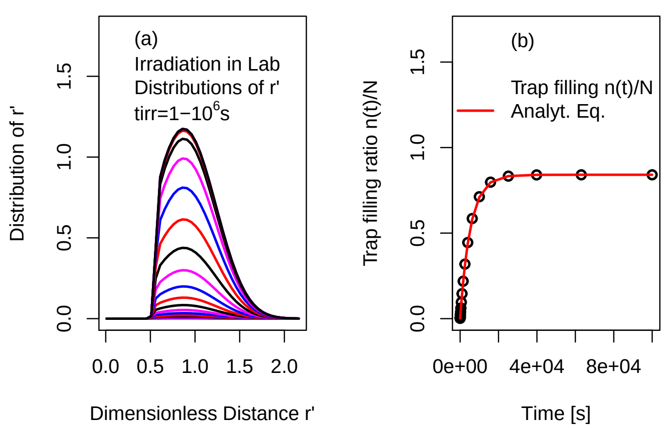

Fig. 12.5: Simulation of irradiation process in the *laboratory* with a dose rate of 0.1 Gy/s and for irradiation times $tirr = 1 - 10^6$ s. As the irradiation time increases, both the asymmetric distribution of distances r' in (a), and the trap filling ratio $n(t)/N$ in (b) approach saturation.

Code 12.5: Simulation of CW-IRSL signal from freshly irradiated feldspars

```
## CW-IRSL function for freshly irradiated feldspars (EST)
rm(list = ls(all=T))
source("Pagonis2020FSF.R")# Loads the functions
rho<-.013                  # Dimensionless acceptor density
dr<-.05                    # Step in dimensionless distance r'
rprimes<-seq(0,2.2,dr)     # Values of r'=0-2.2 in steps of dr'
A<-3                       # A=stun*sigma*I/B
Aeff<-A*exp(-rprimes*(rho**(-1/3.0)))  # Effective A
########################## Simulations ###############
par(mfrow=c(1,3))
#### Example : IRSL for unfaded sample
distr<<-3*rprimes^2*exp(-rprimes^3)  # unfaded distribution
distr1<-distr
timesCW<-seq(1,100)
IRsignal1<-stimIRSL()
distr2<-distr
########## End of Simulations, next Plot the results #######

plot(rprimes,distr1,ylim=c(0,1.9),typ="o",pch=1,col="black",
     ylab="Distribution of r'",xlab="Dimensionless distance r'")
lines(rprimes,distr2,ylim=c(0,1.9),typ="o",pch=2,col="blue",
     ylab="Distribution of r'",xlab="Dimensionless distance r'")
legend("topleft",bty="n",legend=c("(a)"," ",
" Distribution of r'", " before and after IR"))
matplot(timesCW,t(sapply(timesCW,CWsignal)),typ="l",lty="solid",
ylim=c(0,.004), lwd=2, xlab=expression("Times [s]"),
ylab="CW-IRSL [a.u.]")
legend("topleft",bty="n",legend=c("(b)","    ","CW-IRSL curves",
   "for each r'"))
plot(timesCW,IRsignal1,typ="p",lwd=2,pch=1,col="red",
ylim=c(0,.17),xlab=expression("Times [s]"),
ylab="CW-IRSL [a.u.]")
legend("topleft",bty="n",legend=c("(c)"," ","CW-IRSL signal",
  "Unfaded sample"," ","Sum of CW-IRSL","curves"))
lines(timesCW,3*rho*A*1.8*exp(-rho*(log(1+1.8*A*timesCW))**3.0)*
```

```
(log(1 +1.8* A*timesCW) ** 2.0)/(1 + 1.8*A* timesCW),lwd=2,
col="blue")
```

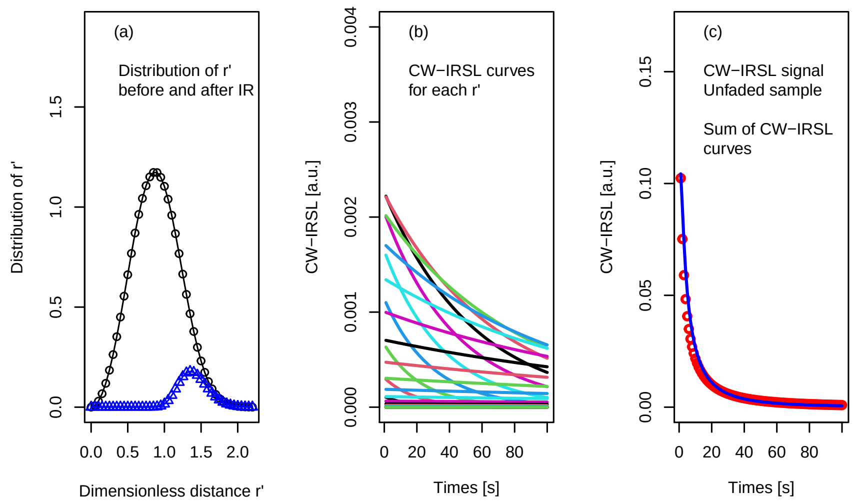

Fig. 12.6: Simulation of CW-IRSL experiment in the EST model, for freshly irradiated samples. (a) The distributions of trapped charge $n(r', t)$ at the beginning and at the end of the CW-IRSL experiment ; (b) The CW-IRSL curves evaluated for each distance r'; (c) The sum of the curves shown in (b), yields the total CW-IRSL signal. The solid line in (c) is the analytical KP-CW equation developed by Kitis and Pagonis [13]. The parameters in the model are typical for feldspars.

Code 12.6: Simulation of TL signal from freshly irradiated feldspars

```
## Simple TL function for freshly irradiated feldspars
rm(list = ls(all=T))
source("Pagonis2020FSF.R")# Loads the FSF functions
rho<-.013                   # Dimensionless acceptor density
s<-3.5e12                   # Frequency factors in s^-1
E<-1.45                     # Energy in eV
Tph<-300                    #Preheat temperature (C)
tph<-10                     #Preheat time (s)
dr<-.1                     #Step in dimensionless distance r'
rprimes<-seq(0.01,2.2,dr)   #Values of r'=0-2.2 in steps of dr
kb<-8.617e-5                #Boltzmann constant
```

```
beta<-1
Tpreheat<-273+320         # Preheat Temperature
seff<-s*exp(-rprimes*(rho**(-1/3.0)))  # Effective s
########################## Simulations ###############
par(mfrow=c(1,3))
# #### Example #1: TL for unfaded sample
distr<<-3*rprimes^2*exp(-rprimes^3)
distr1<-distr
temps<-273+seq(1:400)     # Temperatures for TL
manyTL<-t(sapply(temps,TLsignal))
TL1<-stimTL()             # Evaluate TL signal
distr2<-distr

########## End of Simulations, next Plot the results #######
plot(rprimes,distr1,ylim=c(0,1.9),typ="o",pch=1,col="black",
     ylab="Distribution of r'",xlab="Dimensionless distance r'")
lines(rprimes,distr2,typ="o",pch=2,col="red",
     ylab="Distribution of r'",xlab="Dimensionless distance r'")
legend("topleft",bty="n",legend=c("(a)"," ",
"   Distribution of r'",  "   before and after", "TL"))
matplot(temps-273,manyTL,typ="l",lty="solid",
xlim=c(175,450),ylim=c(0,.03),lwd=2,
xlab=expression("Temperature ["^"o"*"C]"), ylab="TL [a.u.]")
legend("topleft",bty="n",legend=c("(b)","    ","TL curves ",
  "for each r'"))
plot(temps-273,TL1,typ="l", lwd=2,pch=1,col="black",
xlim=c(175,450),ylim=c(0,.018),
xlab=expression("Temperature ["^"o"*"C]"),ylab="TL [a.u.]")
legend("topleft",bty="n",legend=c("(c)"," ","Sum of all",
   "TL curves "))
```

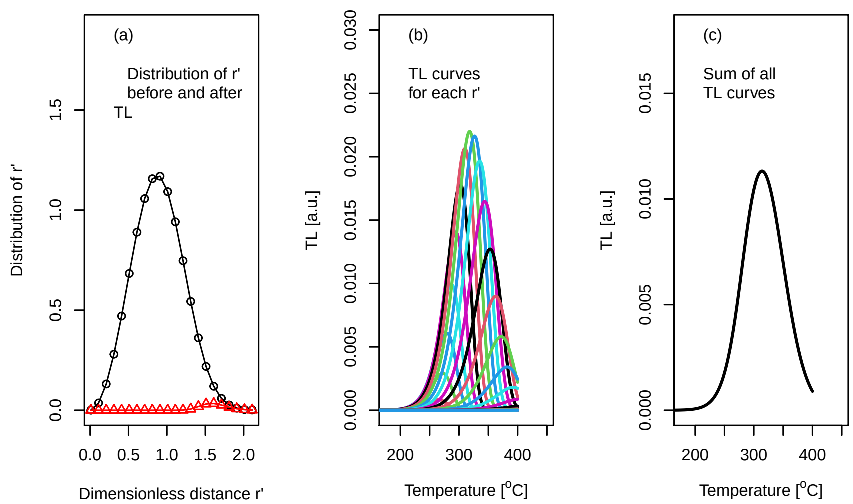

Fig. 12.7: Simulation of TL glow curve for freshly irradiated samples by heating up to 380°C, just below the high temperature end of the TL glow curve. (a) The distributions of distances r' before and after the heating the sample are shown as circles and triangles, respectively; (b) The partial first order TL glow curves; (c) The sum of the glow curves from (b).

Code 12.7: TL from thermally/optically pretreated feldspar samples

```
## Examples of TL for thermally and optically treated samples
rm(list = ls(all=T))
source("Pagonis2020FSF.R")# Loads the FSF functions
rho<-.013                    # Dimensionless acceptor density
Po<-s<-3.5e12                # Frequency factors in s^-1
E<-1.45                      # Energy in eV
Tph<-300                     #Preheat temperature (C)
tph<-10                      #Preheat time (s)
dr<-.05                      #Step in dimensionless distance r'
rprimes<-seq(0.01,2.2,dr)  #Values of r'=0-2.2 in steps of dr
kb<-8.617e-5                 #Boltzmann constant
beta<-1
seff<-s*exp(-rprimes*(rho**(-1/3.0)))  # Effective s
```

```
A<-5
Aeff<-A*exp(-rprimes*(rho**(-1/3.0)))  # Effective A
######################### Simulations ###############
par(mfrow=c(1,2))
   ## Example #1: TL of unfaded sample
distr<<-3*rprimes^2*exp(-rprimes^3)  # unfaded distribution
distr1<-distr              # Store distr for plotting later
temps<-273+seq(1:500)      # Temperatures for TL
TL1<-stimTL()              # Evaluate TL signal
   ## Example #2: Heat to temperature Tpreheat, then measure TL
distr<<-3*rprimes^2*exp(-rprimes^3)  # unfaded distribution
Tpreheat<-273+320          # Preheat Temperature
heatTo(Tpreheat)           # Heat to 320 degC
distr2<-distr              # Store distr for plotting
TL2<-stimTL()              # Store TL
## Example #3: Heat for 30 s at Tpreheat=320C, then measure TL
distr<<-3*rprimes^2*exp(-rprimes^3)
heatTo(Tpreheat)           # Heat to 320 degC
heatAt(Tpreheat,30)        # Heat for 30 s at 320C
distr3<-distr
TL3<-stimTL()
   ## Example #4: CW-IRSL excitation for 10 s, then measure TL
distr<<-3*rprimes^2*exp(-rprimes^3)  # unfaded distribution
timesCW<-seq(1,10)
invisible(sapply(timesCW,CWfortimeT) ) # IR for 10 s
distr4<-distr
TL4<-stimTL()
########## End of Simulations, next Plot the results #######
plot(rprimes,distr1,ylim=c(0,1.9),typ="l",pch=1,col="black",
lwd=2,ylab="Distribution of r'",
xlab="Dimensionless distance r'")
legend("topleft",bty="n",legend=c("(a)","Various Examples",
 "of r' Distributions"))
text1 <- data.frame(x=c(.5,.8,1.,1.25),
 y=c(1,.7,.55,.3), labels=c("1","2","3","4"))
text(text1)
lines(rprimes,distr2,typ="l",col="red",lwd=2)
lines(rprimes,distr3,typ="l",col="blue",lwd=2)
lines(rprimes,distr4,typ="l",col="magenta",lwd=2)
plot(temps-273,TL1,xlim=c(200,450),ylim=c(0,.017),typ="l",
     col="black",lwd=2,
xlab=expression("Temperature ["^"1"*"C]"), ylab="TL [a.u.]")
lines(temps-273,TL2,typ="l",col="red",lwd=2)
lines(temps-273,TL3,typ="l",col="blue",lwd=2)
```

```
lines(temps-273,TL4,typ="l",col="magenta",lwd=2)
legend("topleft",bty="n",legend=c("(b)"," ",
 "Corresponding TL"))
text1 <- data.frame(x=c(270,310,330,350),
y=c(0.009,.006,.004,.002), labels=c("1","2","3","4"))
text(text1)
```

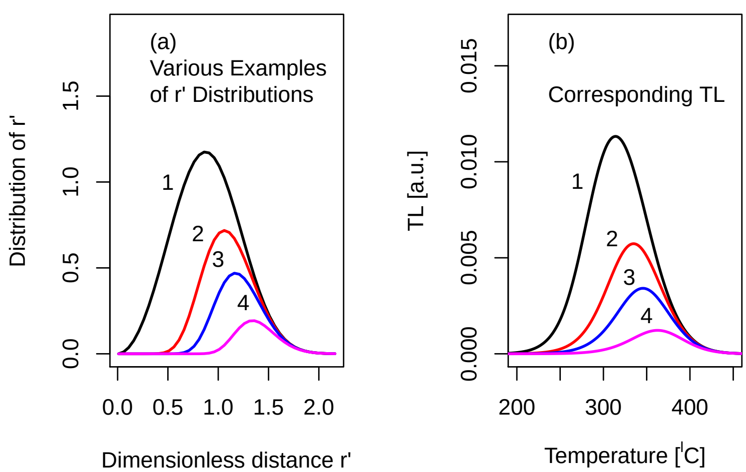

Fig. 12.8: Several examples of the simulation functions for thermally and optically treated samples. The parameters in the model are typical for feldspars. (a) The distribution of distances and (b) The corresponding TL glow curves, for the following processes: (1) TL for unfaded sample; (2) Heat to a preheat temperature T=320°C, then measure TL; (3) Heat for 30 s at T=320°C, then measure TL; (4) CW-IRSL excitation for 10 s, then measure TL.

Code 12.8: Dose response of feldspar in the TA-EST model of Brown et al.

```
#Dose response of feldspars, irradiation in nature (TA-EST)
rm(list = ls(all=T))
rho<-3e-3                   # Dimensionless acceptor density
Po<-s<-2e15                 # Frequency factors in s^-1
E<-1.3                      # Energy in eV
```

```
Do<-1600                     #Do in Gy
yr<-365*24*3600              #year is seconds
Ddot<-2.85/(1e3*yr)          #Low natural dose rate = 1 Gy/kA
dr<-.05                      #Step in dimensionless distance r'
rprimes<-seq(0.01,2.2,dr) #Values of r'=0-2.2 in steps of dr
kb<-8.617e-5                 #Boltzmann constant
seff<-s*exp(-rprimes*(rho**(-1/3.0)))  # Effective s
Tirr<-273-4
Peff<-(1/(1/s+1/seff))*exp(-E/(kb*Tirr))

########## Irradiation functions  ##########
#### irradfortimeT
irradandThermalfortimeT<-function(tirr){distr<<-distrUnfaded*
  (Ddot*rprimes/(Do*Peff+Ddot))*(1-exp(-(Ddot/Do+Peff)*tirr))}
###################### End of Functions ##########
par(mfrow=c(1,2))
distrUnfaded<-3*rprimes^2*exp(-rprimes^3)  # Unfaded sample
irrTimes<-10^seq(2,6,by=.2)*yr
distribs<-sapply(irrTimes,irradandThermalfortimeT)

########## plot distibutions
matplot(rprimes,distribs,typ="l",ylim=c(0,.45),lty="solid",
xlab="Dimensionless Distance r'",
ylab="Distribution of r'",lwd=2)
legend("topleft",bty="n",legend=c(expression("(a)",
"Irradiation in nature",
"Distributions of r'","tirr=10"^3*"-10"^6*"y")))
plot(irrTimes/yr,colSums(distribs)*dr,typ="o",lwd=2,
 xlab="Time [y]",ylab="Trap filling ratio n(t)/N",
 ylim=c(0,.27))
legend("topleft",bty="n",legend=c(" ","(b)",
  "Trap filling n(t)/N"))
```

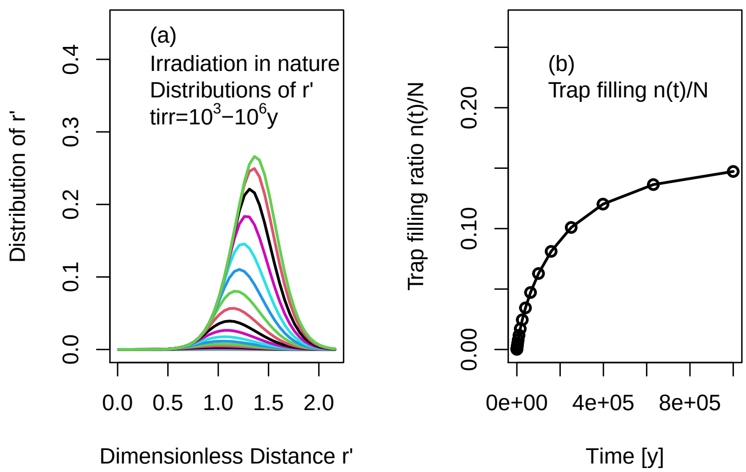

Fig. 12.9: Simulation of irradiations in nature in the TA-EST model, for a fixed burial temperature −4°C. (a) The distributions of the distance parameter r' at various irradiation times. Compare the shape of these distributions with Fig.12.4. (b) The corresponding dose response, shown as the trap filling ratio $n(t)/N$.

Code 12.9: Irradiations at various steady-state temperatures (TA-EST model)

```
## Multiple feldspar irradiations at steady-state temperatures

rm(list = ls(all=T))
rho<-1e-2                      # Dimensionless acceptor density
Po<-s<-2e15                    # Frequency factors in s^-1
E<-1.3                         # Energy in eV
Do<-1600                       #Do in Gy
yr<-365*24*3600                #year is seconds
Ddot<-2.85/(1e3*yr)            #Low natural dose rate = 1 Gy/kA
dr<-.05                        #Step in dimensionless distance r'
rprimes<-seq(0.01,2.2,dr) #Values of r'=0-2.2 in steps of dr
kb<-8.617e-5                   #Boltzmann constant
seff<-s*exp(-rprimes*(rho**(-1/3.0)))  # Effective s
```

```
# Peff<-(1/(1/s+1/seff))*exp(-E/(kb*Tirr))   Effective P

########## Irradiation functions  ##########
#### irradatsometemp
irradatsometemp<-function(Tirr){
  Peff<-(1/(1/s+1/seff))*exp(-E/(kb*Tirr))
  distr<<-distrUnfaded*
  (Ddot*rprimes/(Do*Peff+Ddot))*(1-exp(-(Ddot/Do+Peff)*tirr))}
###################### End of Functions ##########
tirr<-1e3*yr
Tirrs<-273+c(-4,0,4,8)      # Burial temperatures
distrUnfaded<-3*rprimes^2*exp(-rprimes^3)  # Unfaded sample
distribs<-sapply(Tirrs,irradatsometemp)

########## plot distibutions
cols=c("black","red","blue","magenta")
pchs=c(1,2,3,4)
matplot(rprimes,distribs,typ="o",pch=pchs,col=cols,
ylim=c(0,.003),lty="solid",
xlab="Dimensionless Distance r'", ylab="Distribution of r'",
lwd=2)
legend("topleft",bty="n",legend=c("Irradiation in nature",
"Various Burial Temperatures"," ","Distributions of distances"))
legend("topright",bty="n",legend=c(expression("-4 "^o*"C",
  "  0  "^o*"C"," +4  "^o*"C"," +8  "^o*"C")),pch=pchs,col=cols)
```

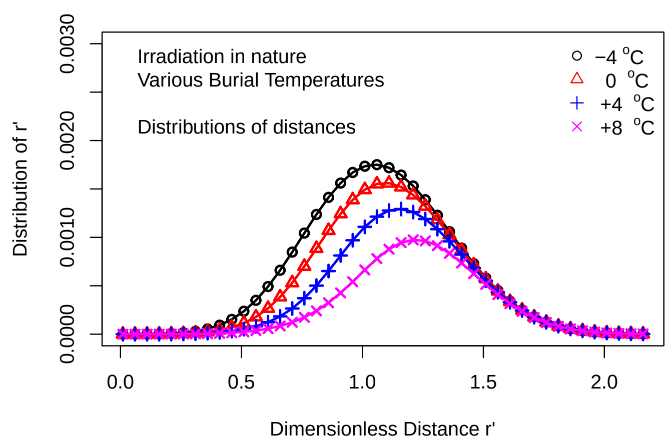

Fig. 12.10: Multiple irradiations in nature using the TA-EST model, for a variable burial temperature $Tirr$= -4, 0, 4, 8 °C, and for a fixed irradiation time $tirr = 10^3$ y.