RW9 Battery

Filippo Boni

22/9/2021

Contents

First analysis of the RW9 battery. In the following sections we will see a description of the dataset and the main concept regarding how the data was collected. Then we will select only the data that we care about for our task and plot some useful graphs. Finally, data will be processed in order to obtain the features to feed the neural network

1 Dataset

1.1 Description of the features

Within the **step** array you will find a struct with the following fields:

Attribute	**Description**
comment	string description of step

Attribute	Description
type	one character identifier of step: 'C' = Charging, 'D' = Discharging, 'R' = Resting (current = 0)
relativeTime	vector of sample time in seconds, referenced to the beginning of the current step
time	vector of sample time in seconds, referenced to the beginning of the experiment
voltage	vector of sample voltage in units of Volts
current	vector of sample current in units of Amps
temperature	vector of sample temperature in units of degrees C
date	date and time at which the current step was started in dd-Mon-yyyy HH:MM:SS format

1.2 The random walk (RW) mode:

1. Selecting a charging or discharging current at random from the set {-4.5A, -3.75A, -3A, -2.25A, -1.5A, -0.75A, 0.75A, 1.5A, 2.25A, 3A, 3.75A, 4.5A}. Negative currents are associated with charging and positive currents indicate discharging.
2. The selected current setpoint is applied until either the battery voltage goes outside the range (3.2V - 4.2V) or 5 minutes has passed.

- These steps are identified with the **comment** field = **"discharge (random walk)"** and **comment** field = **"charge (random walk)"**

3. After each charging or discharging period, there will be a <1s period of rest while a new charging or discharging current setpoint is selected.

- This steps is identified with the **comment** field = **"rest (random walk)"**
- Steps 2 and 3 are repeated 1500 times, then characterization cycles are preformed to benchmark battery state of health.

1.3 Reference charge/discharge

A reference charge and discharge cycle is used to observe the battery capacity after every 1500 RW steps cycles

- Batteries are first charged at 2A (constant current), until they reach 4.2V, at which time the charging switches to a constant voltage mode and continues charging the batteries until the charging current falls below 0.01A.
  - This step is identified with the **comment** field = **"reference charge"**
- Batteries are then rested for a period of time with no current draw
  - This step is identified with the **comment** field = **"rest post reference charge"**
- Batteries are then discharged at 2A until the battery voltage crosses 3.2V
  - This step is identified with the **comment** field = **"reference discharge"**
- Batteries are then rested for a period of time
  - This step is identified with the **comment** field = **"rest post reference discharge"**

Thanks to this reference phase, it's possible to calculate the battery $SOH_t(Q_{nominal}/Q_t * 100\%)$ by simply calculating the present battery capacity ($Q_{present}$) by accumulating total discharge current (I_d) of the battery cell during reference discharge cycle:

$$Q_{present}(t) = \int_0^t I_d(t)dt$$

2 Data Selection, Exploration and Visualization

Here we will select the data useful for our task, namely:

- **"reference charge"** and **"reference discharge"** for the reference step
- **"charge (random walk)"**, **"discharge (random walk)"** and **"rest (random walk)"** for the random walk step

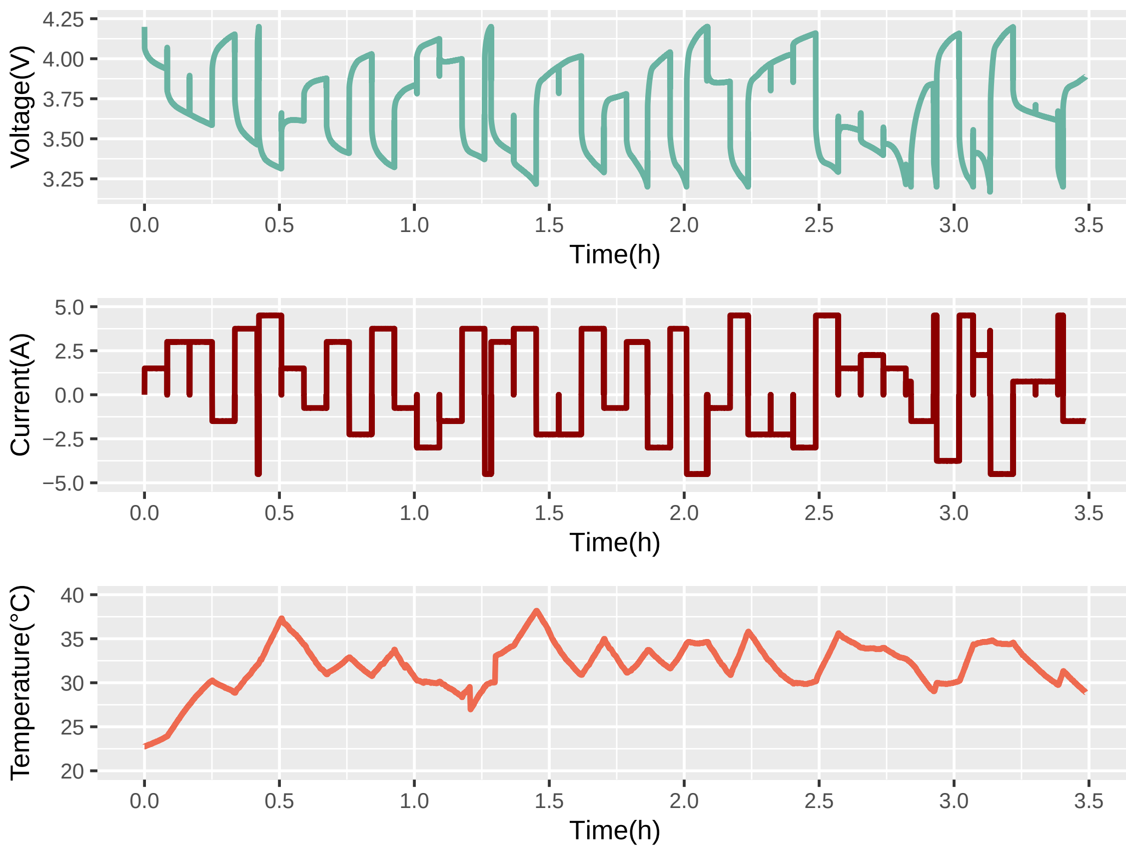
```
#load the data
load("RW9.Rda")
attach(steps)

#select from the "comment" field the experiments of interest
CC <- c("reference charge", "reference discharge", "charge (random walk)", "discharge (random walk)", "r
ReducedSteps <- steps[steps$comment %in% CC,]

#set the levels of the factor column comments to the ones we selected
ReducedSteps$comment <- factor(ReducedSteps$comment,levels = CC)

#remove currents fields with only one current relevation and outside the range -4.5,+4.5 (with comment
#ReducedSteps$comment %in% c("charge (random walk)", "discharge (random walk)") & head(ReducedSteps$cur

#idxes = as.numeric(rownames(refDisSteps[i,]))

#check that we have only the comments we care about
unique(ReducedSteps$comment)
```

```
## [1] reference charge        reference discharge     rest (random walk)
## [4] discharge (random walk) charge (random walk)
## 5 Levels: reference charge reference discharge ... rest (random walk)
```

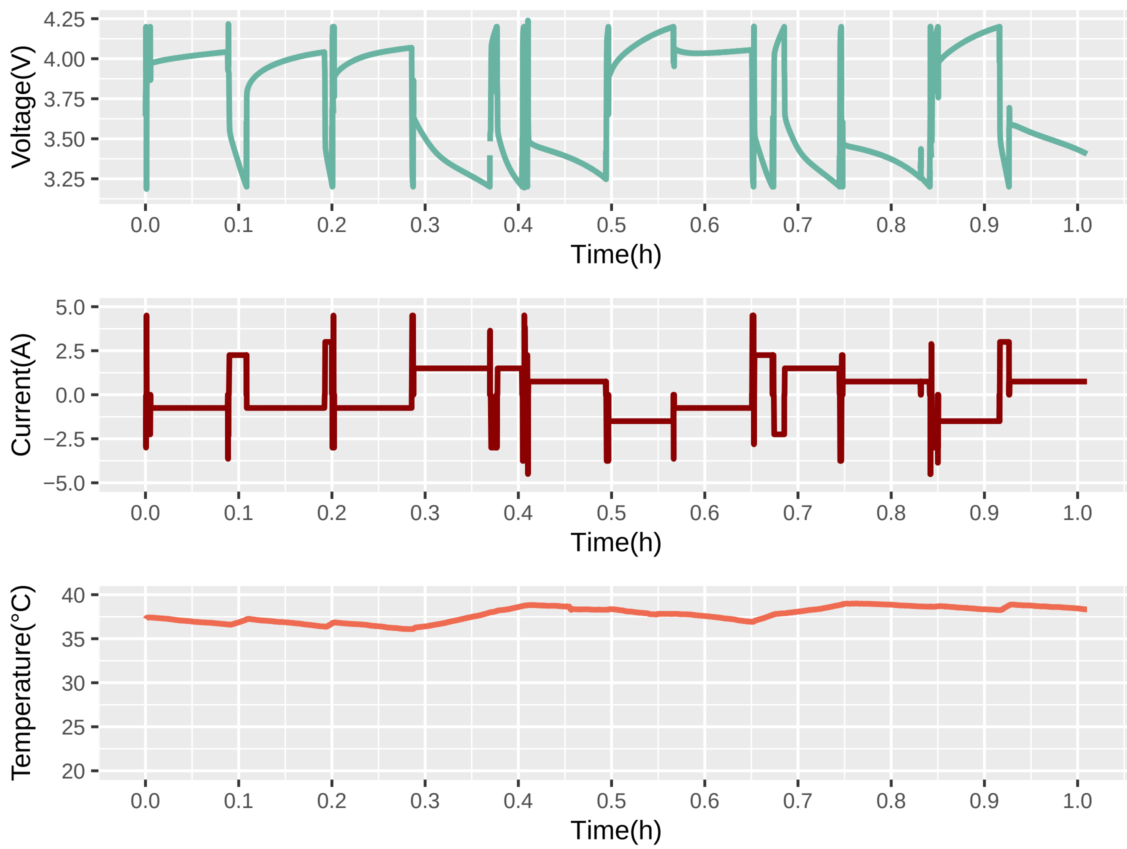
Let's visualize some plots to have an idea of the behavior of the cell during the experiment conducted. Useful to find some relations between the measures.

2.1 V,A and T plots

In the graph below we can see the behavior of Voltage, Temperature and Current in the first 50 RW steps of battery RW9



In the following graphs we see the behaviour of Voltage, current and Temperature in the last 50RW steps for



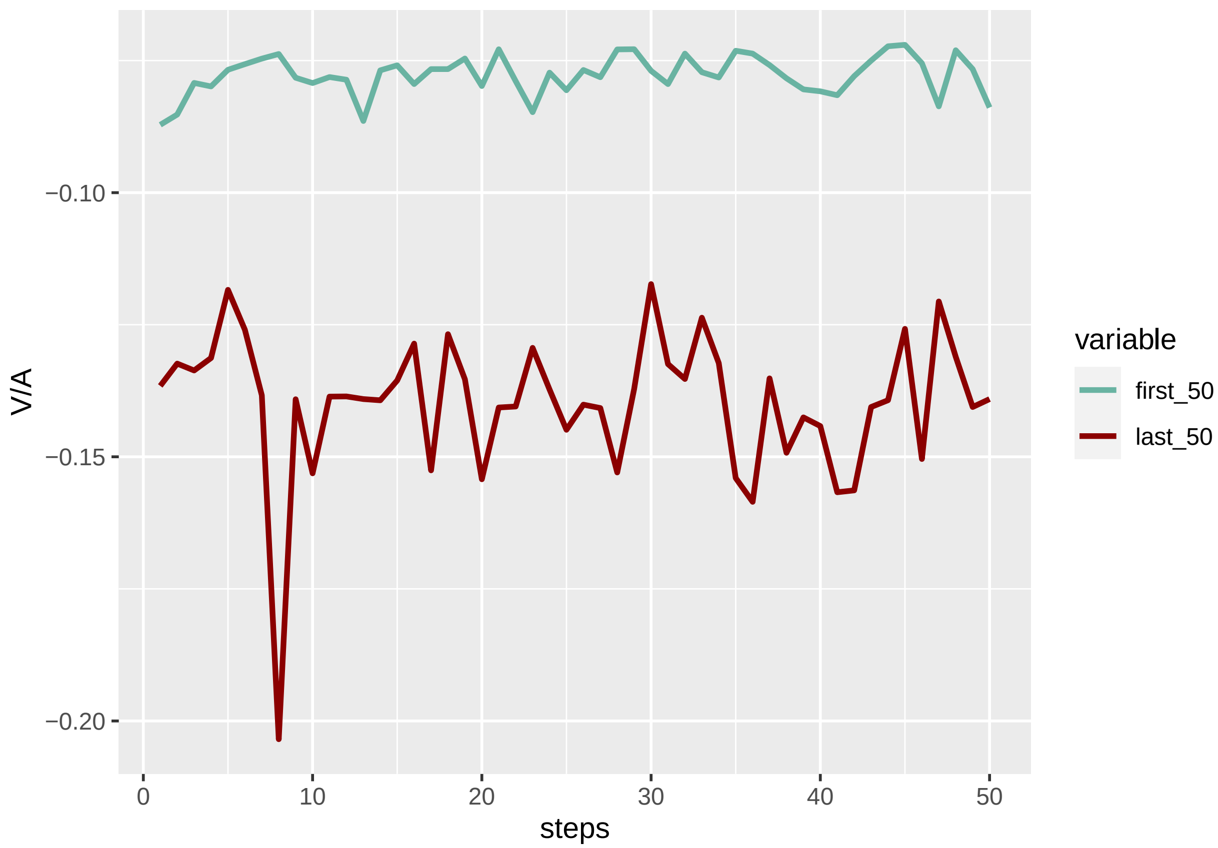
battery RW9

From the plots of the first 50 RW steps is clear how all the three variables are correlated. Take a look at instant 3h: here, as current increase (discharge) , battery voltages starts to decrease and temperature begin to rise gradually. In contrast, at time equal to 2 when current decreases (charge), voltage starts to increase and temperature starts to decrease gradually towards room temperature.

As far as the last 50RW steps, it's evident how the time period is much less since the battery reaches (during the step phase) it's maximum and minimum threshold voltages more often (recall that when max or min voltages are reached, the current step is stopped and a new currend load is chosen and a new step starts). This is probably due to increasing internal resistances which inherently reduces battery storage capacity

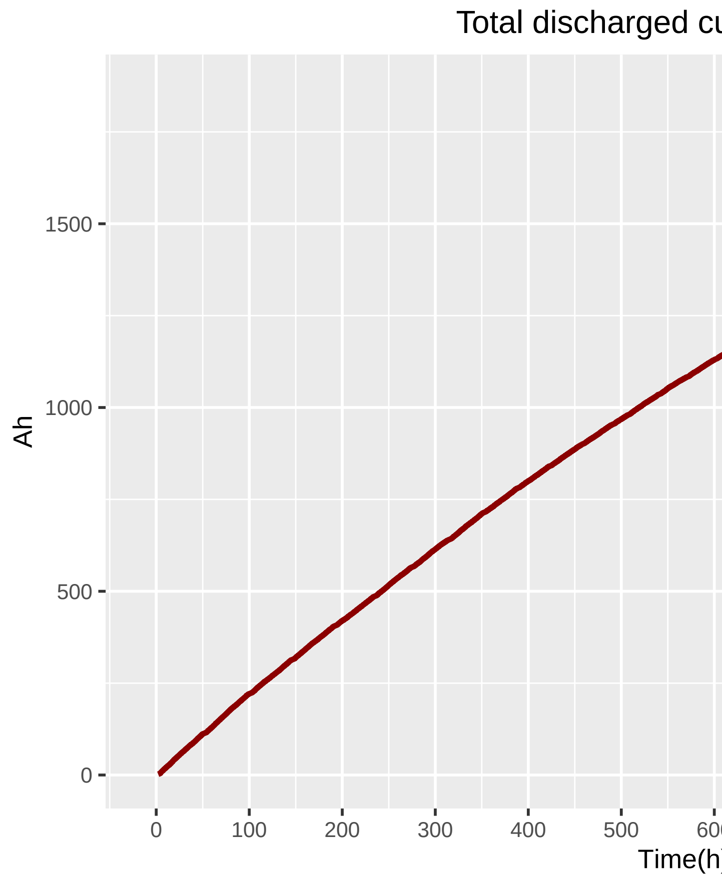
2.2 Study of the ???

For the first 50 RW steps and the lasts, after each "rest (random walk)" the difference of the voltages btw the first voltage acquisition of the new charge/discharge phase and the last voltage acquisition in the previous rest phase is computed and devided by the new current set for the charge/discharge phase . The following are the resulting graphs:



2.3 Total discharge capacity

In the following graph we can observe the cumulative total discharged current from the cell, obtained by selecting only the "reference discharge" and "discharge (random walk)" comments in the *steps* dataset. The



graph cover the experiment on his totality (all the RW9 profile).

Having the total amount of discharged current during RW9 lifetime, we can compute the equivalent cycles that the battery went through

$$equivalent\ cycles = \frac{total\ discharged\ current\ (Ah)}{nominal\ capacity\ (Ah)}$$

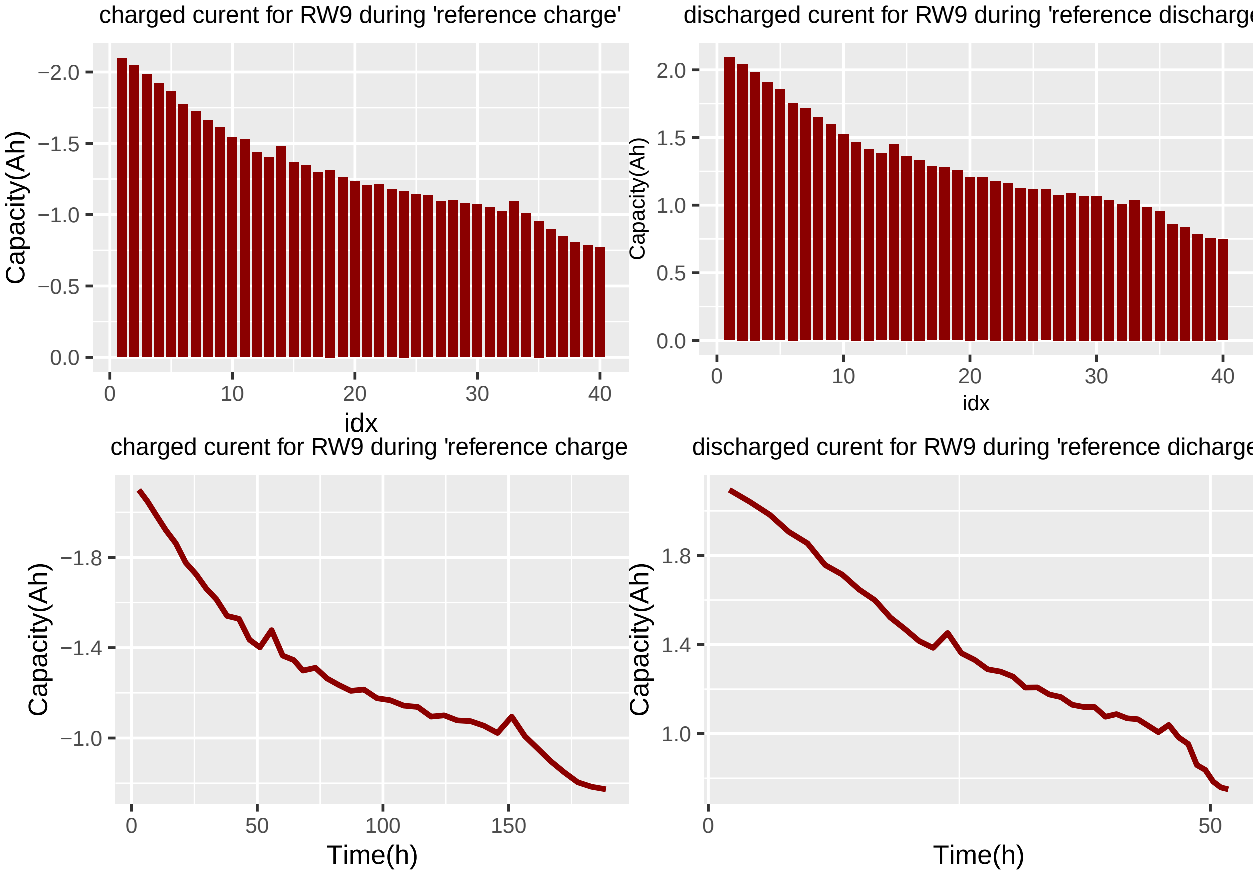
```
## [1] "total discharged current: 1866.971993 Ah"
```

```
## [1] "nominal capacity: 2.10 Ah"
```

```
## [1] "equivalent cycles = 889.034282 "
```

2.4 Reference tests: charged currents and discharged currents

Integral over the time of the charging and discharging currents, for each reference charge/discharge

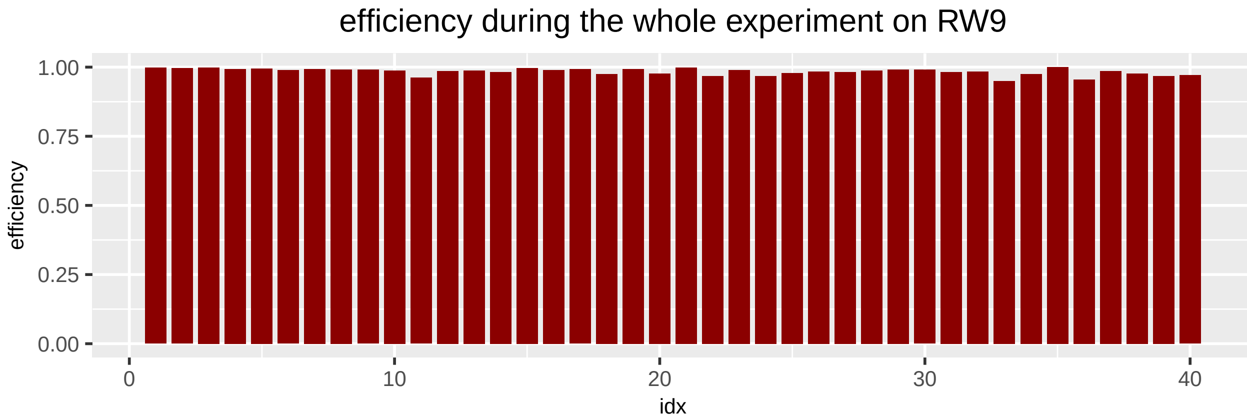


From the above graphs, we can observe the capacity degradation for the RW9 cell during the experiment.

2.5 Efficiency

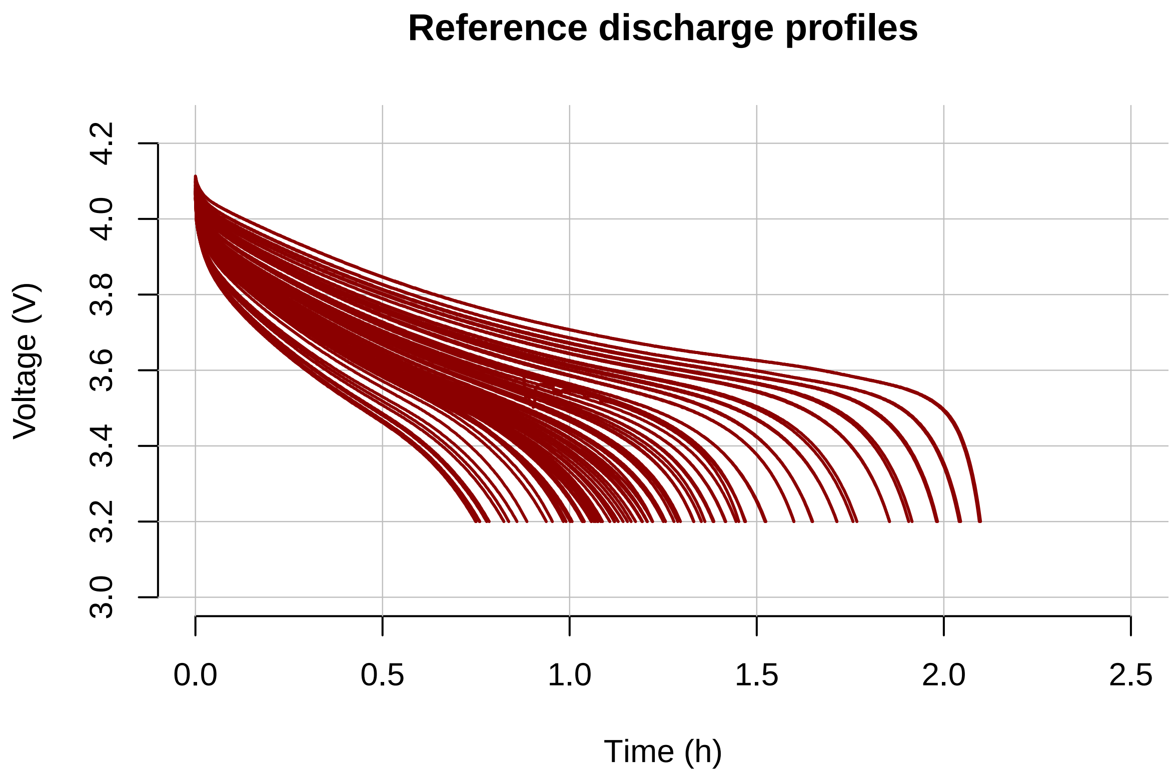
We are now observing the efficiency of the battery RW9 among all the profile (made by 80 charge and discharge reference test), computed as

$$efficiency_t = \frac{Ah_{discharge,t}}{Ah_{charge,t}}$$



2.6 Reference discharge profile

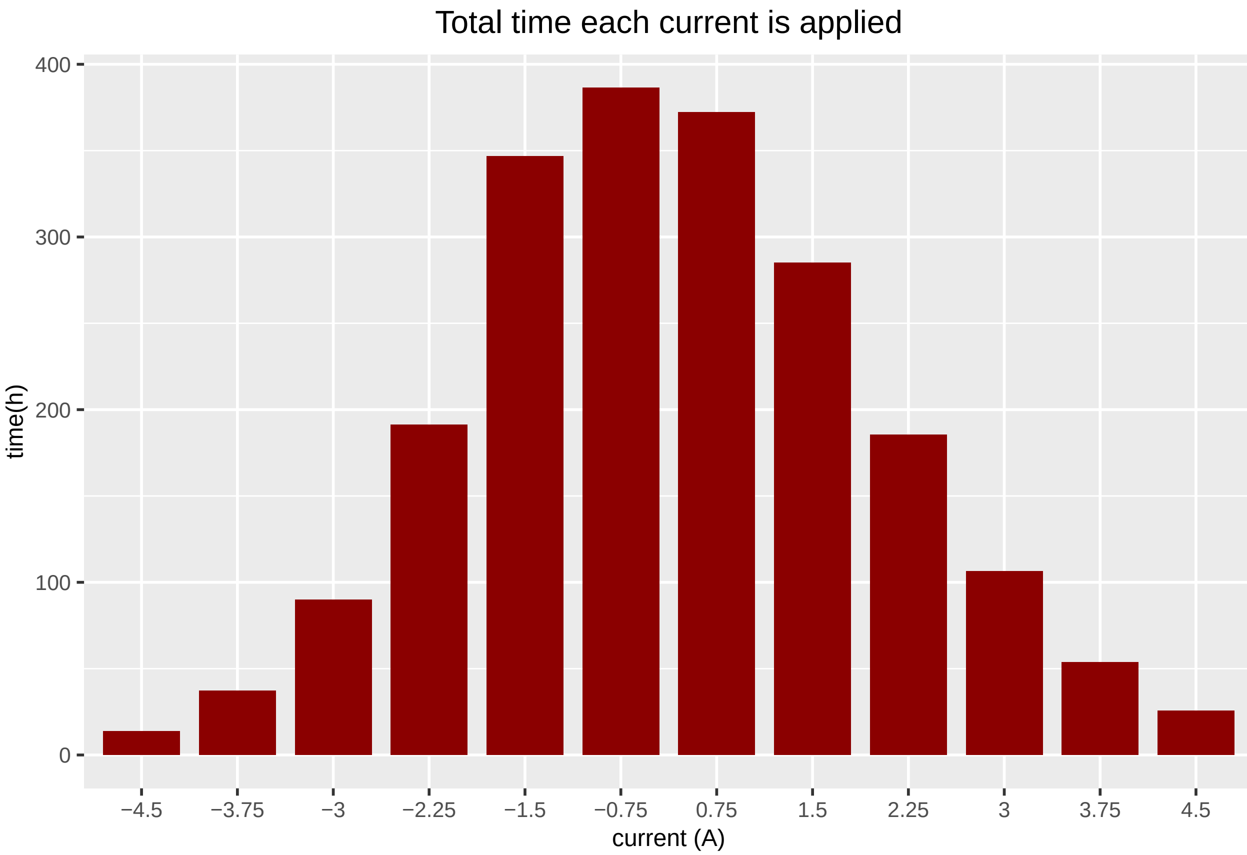
Constant load profile are run after every 1500 RW step to have calculate the capacity of the battery. Let's plot the voltages for each reference discharge in order to observe how the decay rate of the battery changes during it's lifetime.



We can observe that the voltage decay rate increases as the battery ages which leads to a decrease in battery capacity.

2.7 Time for each segment of currents

We will now study how much time each segment of current is applied on the cell during all the RW steps.



2.8 distribution of the input currents

Lets verify that the currents chosen for the discharge and charge phases are actually randomly picked

```
data <- data.frame(x,curr_counts)
data$x <- factor(data$x,levels = c("-4.5","-3.75","-3","-2.25","-1.5","-0.75","0.75","1.5","2.25","3","3

#TODO
```

So the randomization of current selection is verified.

2.9 80% of SOH

All the tests are performed by NASA considering the whole cell lifetime (30%-50% SOH) but we know that battery manufacturers consider the EOL of a cell when it reaches the 80% of its SOH since its performances starts to decrease significantly at faster rate below 80%. Let's prepare the basis for a future experiment considering only the portion of the dataset where SOH >= 80%. \ The nominal capacity of the cells in these experiments is: **2.1 Ah** \ The EOL, i.e the 80% of it's capacity is: **1.68 Ah**

```
#create the new dataset where we keep only Capacities >= 80% (1.68 Ah)

#select only the "reference discharge" steps
refDisSteps <- steps[steps$comment %in% "reference discharge",]
```

```
#check where the Ah goes below 1.68 and get the idx of the row
for(i in 1:length(refCharSteps$comment)){
  Ah <- mapply(trapz,refDisSteps$relativeTime[i],refDisSteps$current[i])/3600

  if(Ah < 1.68){
    eol_idx = as.numeric(rownames(refDisSteps[i,]))
    Ldata <- ReducedSteps[1:eol_idx,]
    cat("The number of reference discharge with SOH >= 80% is ", i/2 + 0.5)
    cat("\n")
    cat("The index of EOL is ", eol_idx)
    break
  }
}
```

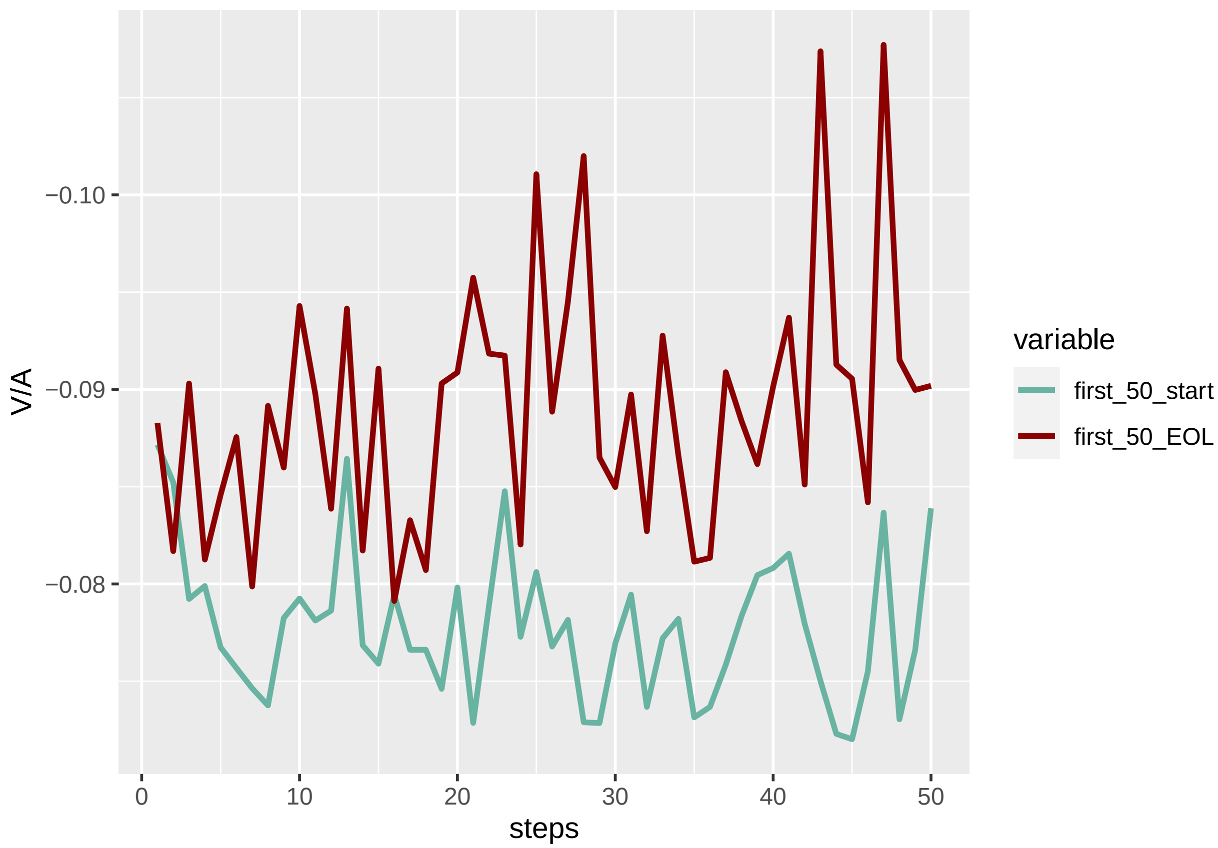
```
## The number of reference discharge with SOH >= 80% is  8
## The index of EOL is  21188
```

##Delta-V full capacity vs EOL battery

We will now study the difference of the voltages (like we did before) but now comparing the full capacity cell (the first 50 steps) with the cell at his EOL (first 50 steps also here)

```
#Grafico primi 50 step/primi 50 step sulla variazione di corrente (Delta V sulla corrente misurata per
#Quindi prendi dopo ogni rest il valore della corrente (I) e fai il rapporto con la variazione di volta

CC <- c("discharge (random walk)", "charge (random walk)", "rest (random walk)")

# Identify step indexes that match the comment codes
inds <- 1:dim(steps)[1] # set vector of step indexes
indCC <- inds[steps$comment %in% CC]

# identify discontinuities in the indCC array
Breaks <- c(0,which(diff(indCC) != 1),length(indCC))

# separate continuous sequences of indexes into a list
refInds<- lapply(seq(length(Breaks)-1),
                function(i) indCC[(Breaks[i]+1):Breaks[i+1]])

# identify index sequence for first 50 steps
inds <- head(refInds[[1]],n=100) # set n = 100 to get fifty current steps and 50 rest steps

#set the first element of the two vectors
curs_2 <- steps$current[[inds[1]]][1]
voltagesDiff_2 <- steps$voltage[[inds[1]]][length(steps$voltage[[inds[1]]])]

for(i in 2:length(inds)){
  if(steps$comment[[inds[i]]]!="rest (random walk)"){
    curs_2 <- c(curs_2,steps$current[[inds[i]]][1])
    voltagesDiff_2 <- c(voltagesDiff_2,steps$voltage[[inds[i]]][1]-tail(steps$voltage[[inds[i]-1]],n=1)
  }
}
dV_i_2 = voltagesDiff_2[2:length(voltagesDiff_2)]/curs_2[2:length(curs_2)]
idx = c(1:50)
```

```
#----------------------------------------------------------------------------------------------------

#now select the first 50 steps of the EOL dataset (where capacity is less than 80%) and repeat
EOLdata <- ReducedSteps[eol_idx+1:length(ReducedSteps$comment),]

inds <- 1:dim(EOLdata)[1] # set vector of step indexes
indCC <- inds[EOLdata$comment %in% CC]

# identify discontinuities in the indCC array
Breaks <- c(0,which(diff(indCC) != 1),length(indCC))

# separate continuous sequences of indexes into a list
refInds_eol<- lapply(seq(length(Breaks)-1),
                function(i) indCC[(Breaks[i]+1):Breaks[i+1]])

# identify index sequence for first 50 steps
inds <- head(refInds_eol[[1]],n=100) # set n = 100 to get fifty current steps and 50 rest steps

#set the first element of the two vectors
currs <- EOLdata$current[[inds[1]]][1]
voltagesDiff <- EOLdata$voltage[[inds[1]]][length(EOLdata$voltage[[inds[1]]])]

for(i in 2:length(inds)){
  if(EOLdata$comment[[inds[i]]]!="rest (random walk)"){
    currs <- c(currs,EOLdata$current[[inds[i]]][1])
    voltagesDiff <- c(voltagesDiff,EOLdata$voltage[[inds[i]]][1]-tail(EOLdata$voltage[[inds[i]-1]],n=1)
  }
}
dV_f = voltagesDiff[2:length(voltagesDiff)]/currs[2:length(currs)]

#create a dataframe with both the measures and the common indexes
data <- data.frame(
  first_50_start = dV_i_2,
  first_50_EOL = dV_f,
  steps = idx
)

#convert the dataset into long format
data_long <- melt(data,id = "steps")

#plottiamo il rapporto delta(V)/I in relazione con i 50 step iniziali e finali
d <-ggplot(data = data_long,
       aes(x=steps,y = value,colour = variable)) +
      geom_line(size=1)+
      scale_y_reverse()+
      ylab("V/A")
d+scale_color_manual(values = c("#69b3a2","red4"))
```

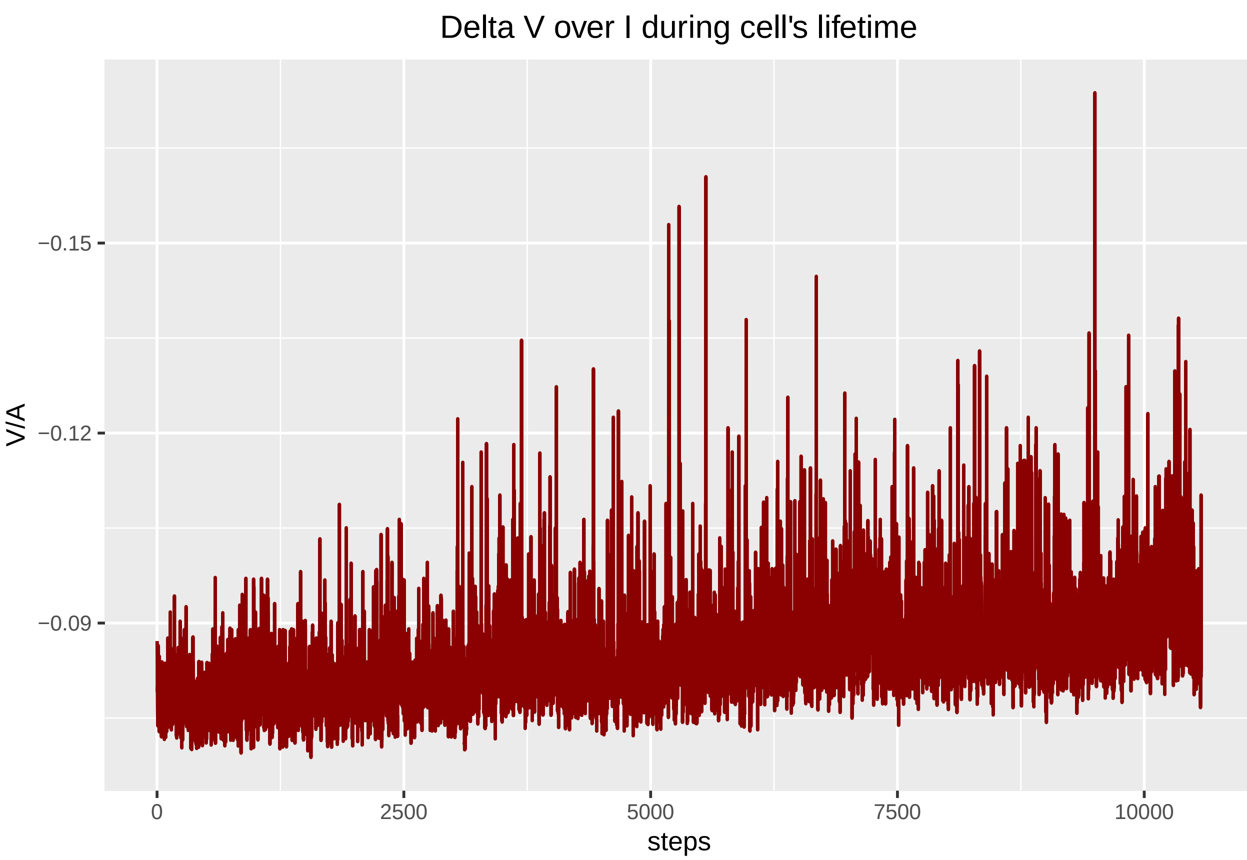


##Delta-V during all useful life (till 80%)

Here we analyze the same derived measure as before (*delta-V over delta-I*) but for all the steps till the battery reaches his EOL

```
#select from Ldata (lifetime data) only the rows for the walk
walk_life <- Ldata[Ldata$comment %in% c("charge (random walk)", "discharge (random walk)", "rest (rando

#set the first element of the two vectors
currs <- walk_life$current[[1]][1]
voltagesDiff <- walk_life$voltage[[1]][length(walk_life$voltage[[1]])]

for(i in 2:dim(walk_life)[1]){
  if(walk_life$comment[[i]]!="rest (random walk)"){
    currs <- c(currs,walk_life$current[[i]][1])
    voltagesDiff <- c(voltagesDiff,walk_life$voltage[[i]][1]-tail(walk_life$voltage[[i-1]],n=1)) #get t
  }
}
dV = voltagesDiff[2:length(voltagesDiff)]/currs[2:length(currs)]
idx = c(1:(length(voltagesDiff)-1))

#---------------------------------print-------------------------------------

#create a dataframe with the measures and the common indexes
data <- data.frame(
  measure = dV,
  steps = idx
```

```r
)

d <- ggplot(data,aes(x=steps,y=measure))+
        geom_line(color = "red4",size = 0.6)+
        theme(plot.margin = unit(c(0.3,0,0,0),"cm"),axis.text = element_text(size = 8),
              axis.title = element_text(size = 10),plot.title = element_text(hjust = 0.5,size = 12))+
        ylab("V/A")+
        scale_y_reverse() +
        ggtitle("Delta V over I during cell's lifetime")

d
```

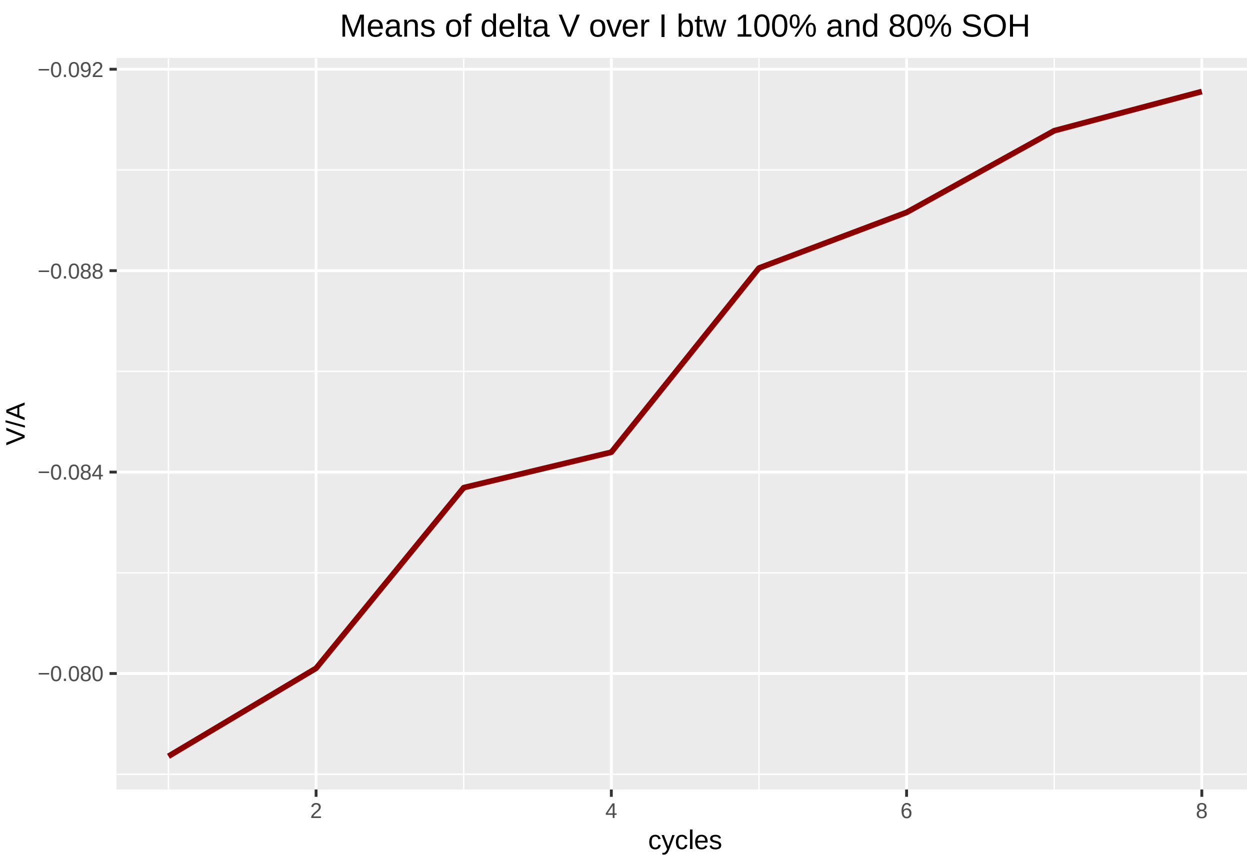


##Delta-V during all useful life (till 80%) but taking means for each cycle

Here we analyze the same derived measure as before (*delta-V over delta-I*) but for all the cycles means till the battery reaches his EOL

```r
#initialize to zero the vectors that will contain the measurements for all the cycle
deltas <- 0

#initialize the curr and voltage vectors
currs<-0
voltagesDiff <- steps$voltage[[refInds[[1]][1]]][length(steps$voltage[[refInds[[1]][1]]])]

#set the first element of the two vectors (using refInds that is took from Ldata)
for(i in 2:length(refInds[[1]])){
```

```r
  if(steps$comment[[refInds[[1]][i]]]!="rest (random walk)"){
    currs <- c(currs,steps$current[[refInds[[1]][i]]][1])
    voltagesDiff <- c(voltagesDiff,steps$voltage[[refInds[[1]][i]]][1]-tail(steps$voltage[[refInds[[1]]
  }
}

deltas <- c(deltas,mean(voltagesDiff[2:length(voltagesDiff)]/currs[2:length(currs)]))

for (j in 2:8){
  currs <- 0
  voltagesDiff <- 0
  for(i in 2:length(refInds[[j]])){
  if(steps$comment[[refInds[[j]][i]]]!="rest (random walk)"){
    currs <- c(currs,steps$current[[refInds[[j]][i]]][1])
    voltagesDiff <- c(voltagesDiff,steps$voltage[[refInds[[j]][i]]][1]-tail(steps$voltage[[refInds[[j]]
  }
}

deltas <- c(deltas,mean(voltagesDiff[2:length(voltagesDiff)]/currs[2:length(currs)]))
}

#---------------------------------print--------------------------------------

#create a dataframe with the measures and the common indexes
data <- data.frame(
  measure = deltas[2:length(deltas)],
  cycles = c(1:(length(deltas)-1))
)

d <- ggplot(data,aes(x=cycles,y=measure))+
        geom_line(color = "red4",size = 1)+
        theme(plot.margin = unit(c(0.3,0,0,0),"cm"),axis.text = element_text(size = 8),
              axis.title = element_text(size = 10),plot.title = element_text(hjust = 0.5,size = 12))+
        ylab("V/A")+
        scale_y_reverse() +
        ggtitle("Means of delta V over I btw 100% and 80% SOH")

d
```

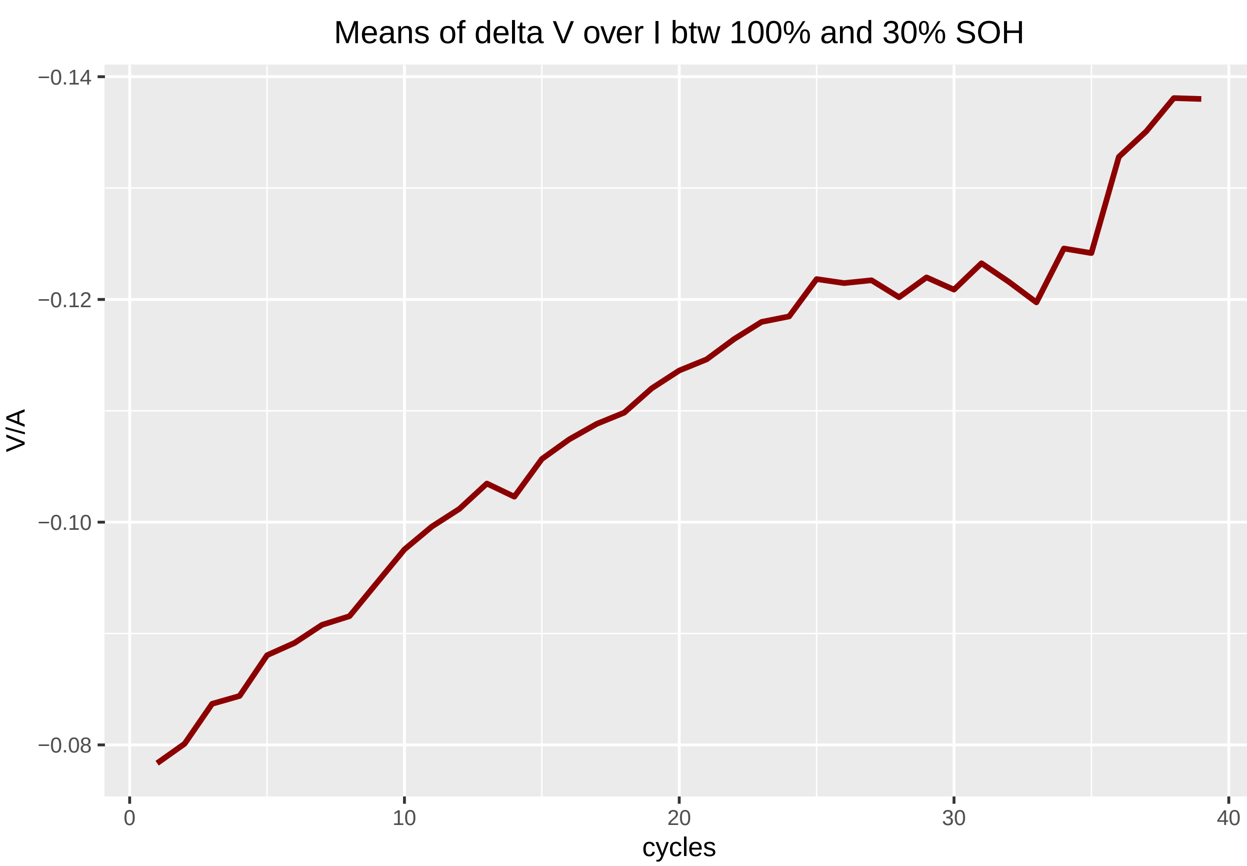


##Delta-V during all the experiment (100% to 30% SOH) taking means for each cycle

Here we analyze the same derived measure as before (*delta-V over delta-I*) but for all the cycles means till the battery reaches his EOL

```
#initialize to zero the vectors that will contain the measurements for all the cycle
deltas <- 0

#initialize the curr and voltage vectors
currs<-0
voltagesDiff <- steps$voltage[[refInds[[1]][1]]][length(steps$voltage[[refInds[[1]][1]]])]

#set the first element of the two vectors (using refInds that is took from Ldata)
for(i in 2:length(refInds[[1]])){
  if(steps$comment[[refInds[[1]][i]]]!="rest (random walk)"){
    currs <- c(currs,steps$current[[refInds[[1]][i]]][1])
    voltagesDiff <- c(voltagesDiff,steps$voltage[[refInds[[1]][i]]][1]-tail(steps$voltage[[refInds[[1]]
  }
}

deltas <- c(deltas,mean(voltagesDiff[2:length(voltagesDiff)]/currs[2:length(currs)]))

for (j in 2:length(refInds)){
  currs <- 0
  voltagesDiff <- 0
  for(i in 2:length(refInds[[j]])){
  if(steps$comment[[refInds[[j]][i]]]!="rest (random walk)"){
```

```
    currs <- c(currs,steps$current[[refInds[[j]][i]]][1])
    voltagesDiff <- c(voltagesDiff, steps$voltage[[refInds[[j]][i]]][1]-tail(steps$voltage[[refInds[[j]]
  }
}

deltas <- c(deltas,mean(voltagesDiff[2:length(voltagesDiff)]/currs[2:length(currs)]))
}

#---------------------------------print---------------------------------------

#create a dataframe with the measures and the common indexes
data <- data.frame(
  measure = deltas[2:length(deltas)],
  cycles = c(1:(length(deltas)-1))
)

d <- ggplot(data,aes(x=cycles,y=measure))+
        geom_line(color = "red4",size = 1)+
        theme(plot.margin = unit(c(0.3,0,0,0),"cm"),axis.text = element_text(size = 8),
              axis.title = element_text(size = 10),plot.title = element_text(hjust = 0.5,size = 12))+
        ylab("V/A")+
        scale_y_reverse() +
        ggtitle("Means of delta V over I btw 100% and 30% SOH")

d
```



Resistenza iniziale:

```
cat("initial resistance = ", deltas[2]) #starting from 2 becuase we built the deltas vector as zero-sta
```

```
## initial resistance =  -0.07835632
```

Considerando fine vita tra 1.6-2 volte la resistenza iniziale, possiamo considerare come fine vita nel range:

```
rng <- c(deltas[2]*1.6,deltas[2]*2)
cat(rng)
```

```
## -0.1253701 -0.1567126
```

2.10 Final Obstervations

With the graphs about the internal resistance, we tried to study the EOL of the cell by the resistance point of view, knowing that the EOL is when the cell reaches a range btw [1.6,2] times the initial resistance (found on *Review on state-of-health of lithium-ion batteries: Characterizations,estimations and applications (Journal of Cleaner Production)*) . With the last graph, we can see that the cell reaches his **EOL_resistance** exactly at the end of the experiments (cycle 40) so probably this is the criterion NASA conducted the experiments.

As far as the **EOL_capacity**, we see that is reached just at the 8 cycle, so it is not a good indicator of the EOL of the cell in this experiment