Supplementary Material
Grain Boundary Resistance in Copper Interconnects from an Atomistic Model to a Neural Network

November 20, 2017

The following R-code corresponds to the neural network described in preprint arXiv:1701.04897. Please feel free to contact the authors if you have any additional questions.

```r
# load libraries
library(ggplot2)
library(grid)
library(gridExtra)
library("neuralnet")

# NEURAL NETWORK
# Read Atomistic Results
training<−read.csv(file="trainingRv7.csv")
# Train
training$R<−log(training$r)
training$angle1<−cos(training$angle1*pi/180)
training$angle2<−cos(training$angle2*pi/180)
training$angle3<−cos(training$angle3*pi/180)
dim(training)

# Test
testing<−read.csv(file="testingRv7.csv")
testing$R<−log(testing$r)
testing$angle1<−cos(testing$angle1*pi/180)
testing$angle2<−cos(testing$angle2*pi/180)
testing$angle3<−cos(testing$angle3*pi/180)
dim(testing)

set.seed(555)
net.sqrt <- neuralnet(r~angle1+angle2+angle3,training, hidden = c(10,6,3), threshold=0.0001, stepmax = 10000000)

#net.sqrt <- neuralnet(r~angle1+angle2+angle3,training, hidden = c(10,6,3), threshold=0.001, stepmax = 10000000)
save(net.sqrt, file="NNResistivity.R")
```
head(net$\sqrt{[[13]]})
print(net$\sqrt{)}

# Plot the neural network
plot(net$\sqrt{)}
deV.copy(jpeg, filename="plot.jpg");
deV.off();

# Test the neural network on some training data
# Generate some squared numbers
testingIN<- testing[,2:4]
testingOUT<- testing[,6]
net.results <- compute(net$\sqrt{), testingIN) # Run them through the neural network

# NN results
print(net$results$net$result)

# Calculating the MSE
\text{cleanoutput}$diff$\sqrt{<- (cleanoutput$\text{SR}$ - cleanoutput$\text{SR}. Expected)^2$
MSE.MLN<- sum(cleanoutput$\text{diff}. \sqrt{)/length(cleanoutput$\text{diff}. \sqrt{)}

Figure 1: Multi-Layer Neural Network for specific resistance on copper interconnect as described in preprint arXiv:1701.04897 obtained by the code described before.