

Lab 3 - Particle Size Analysis

Name:

Group:

Pledge:

"On my honor as a Virginia Tech student, I have neither given nor received unauthorized assistance on this assignment."

Initial: _____

Grade:

Comments:

Goal:

To describe the size of particles in a sample using numbers instead of words.

Why?

Quantifying the size of particles is important for:

- **Plant Design:** Analysis of energy used in grinding
- **Improving Efficiency:** Examining relationships between liberation, particle size, and recovery
- **Process Troubleshooting:** Determining if the feed size to a piece of processing equipment is appropriate, determining if plant classifiers are working as intended
- **Quality Control:** Ensuring the customer is getting what they purchased

How?

We know:

1. If we take a sample and screen it, we can measure the fraction of the sample that passes through the screen, and the fraction that does not pass. Thereby, measuring the fraction of a sample smaller and larger than the holes in the screen.
2. Screens that can be stacked and have openings with a specified size could be manufactured with high precision. A device could also be manufactured that shakes this stack of screens for a specified time period.
3. The equation for a line is $y = mx + b$
4. Particle size distributions exist

Knowing this information, we can use a stack of screens, and a device to shake them, called a Ro-Tap, to collect data. The data will be the fraction of the sample that resides between two screens after shaking, as well as the size of the openings of those screens. After sieving, we can fit a line to the data collected. The parameters $\{m, b\}$ of the line are based on a transformed size distribution.

Procedure

Your laboratory instructor will provide you with a sample of crushed ore from the grinding circuit feed at Freeport's Morenci Mine in Morenci, Arizona. The material is a siliceous chalcopyrite ore containing $\approx 0.4\%$ copper. The minus $\frac{1}{2}$ inch ore was crushed using a laboratory roll crusher and then screened to produce minus 6 mesh material. Using this material, please conduct a sieve analysis and determine the particle size distribution. Your data should be reported on the basis of standard size distribution plots and mathematical fitting expressions.

1. Don all appropriate safety and personal protective equipment
2. Weigh the contents of the bag of sample provided to you by the laboratory instructor. Record the weigh in the space at the bottom of the provided data table (see next page).
3. Obtain the following set of U.S. Standard mesh sieves: 8, 16, 30, 50, 100 and 200 mesh. Weigh each sieve and record the weight in the provided data table. Also weigh the catch pan.
4. Set the six sieves in a stacked nest. The order should be from largest to smallest openings from top to bottom. The catch pan should be placed below the last sieve.
5. Pour your sample onto the top sieve and cover with the top lid. Place the stack in the Ro-Tap and shake the stack for 10 minutes.
6. After shaking, use an analytical balance to find the combined weight of sieve plus sample starting from the top of the stack. Record the combined weight on the provided data table.
7. For each sieve, subtract the empty sieve weight from the weight of sample plus sieve to determine the "individual" sample weight of solid particles retained on each sieve. Do the same for the catch pan. Record the "individual" sample weights in the provided data table.
8. Sum the individual weights and compare the total to the original head sample weight. If the values do not closely agree, then a mistake has been made and the analysis must be repeated. If the values closely agree, then discard the sample, clean the sieves, and reassemble the stack to protect the sieve surfaces for damage during storage.
9. Complete the attached data sheet to determine the individual "percent in class" for each sieve size as well as the cumulative fractions "passed" and "retained" for each sieve size.
10. Use your sieve analysis data to construct Gaudin-Schuhmann and Rosin-Rammler plots on the provided graph paper. Determine and report the unknown empirical fitting parameters for each of these expressions in the provided blanks.

Data Records and Calculations

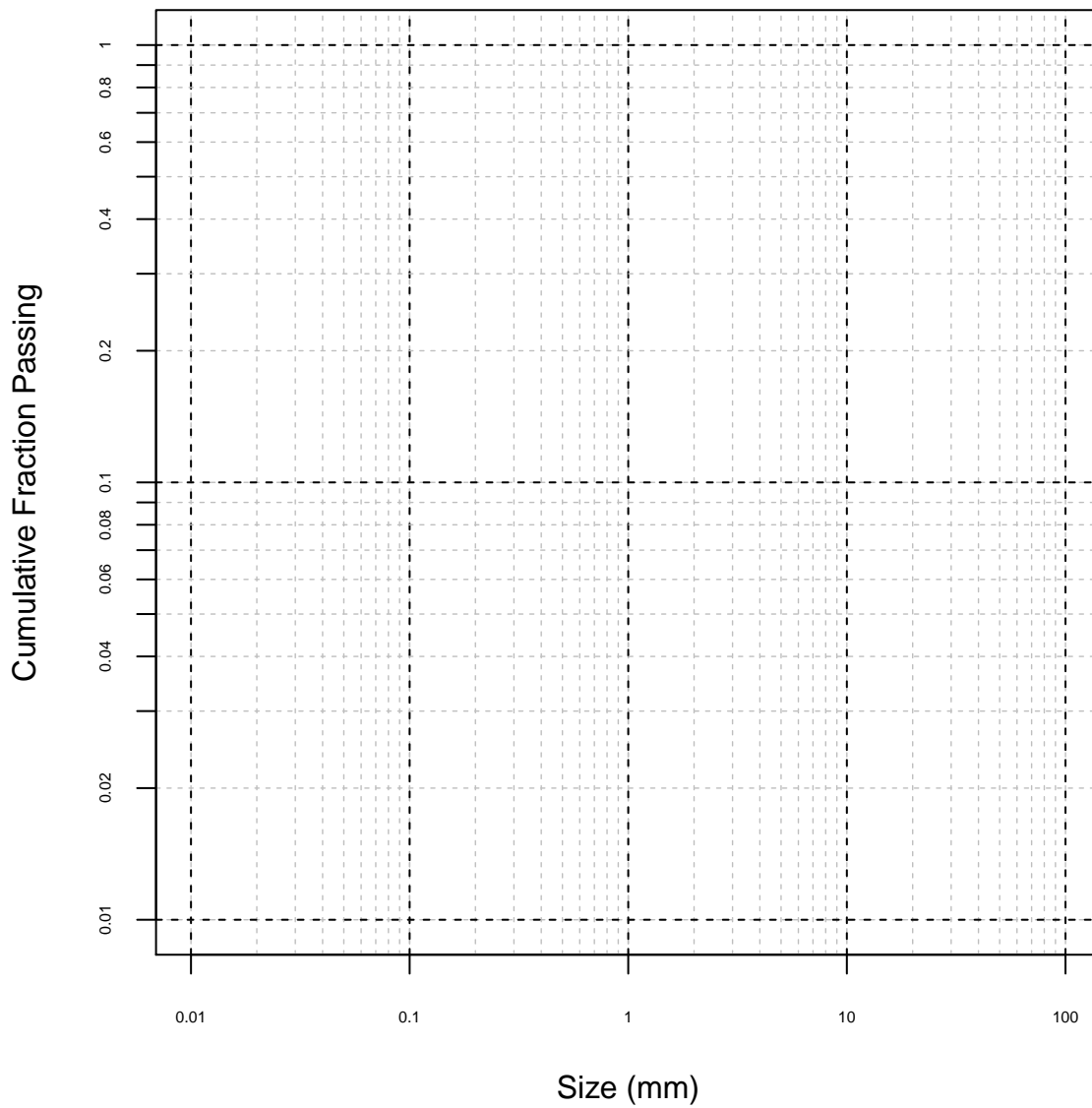
Sieve (Mesh)	Sieve Opening, X(mm)	Avg. Size in Class, $X_{i \times j}$ (mm)	Individual Weights (g)			Percent in Class	Cum. Weights (%)	
			Sieve	Sieve + Sample	Sample		Retained	Passed, Y
6		—	—	—	—	0.00	0.00	100.00
8								
16								
30								
50								
100								
200								
Pan							100.00	0.00
Head Weight =				Totals:				

Sample Loss %:

Mass Mean Particle Size (mm):

Gaudin Schumann Plot:

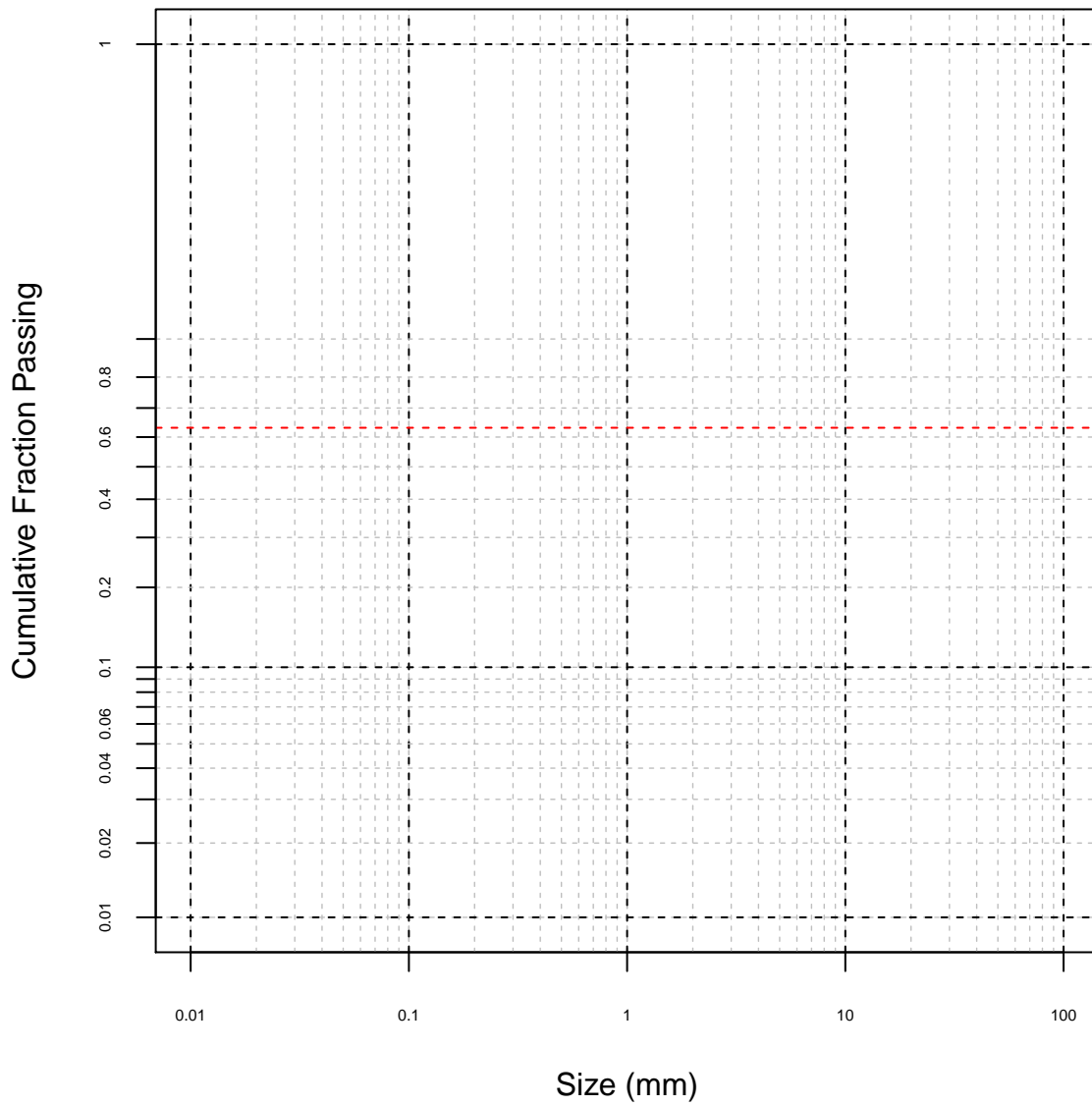
$$Y = \left(\frac{X}{k}\right)^m$$



Solve for k and m here, circle your answers:

Rosin-Rammler Plot

$$Y = 1 - e^{-\left(\frac{x}{R}\right)^m}$$



Solve for R and m here, circle your answers:

Discussion Questions

1. What do the size modulus and distribution modulus represent?
2. Which expression fits your data better: Gaudin-Schuhmann or Rosin-Rammler? Please explain.
3. A manufacturer has determined that the capacity of a 0.5 mm screen is 0.27 tons per hour (TPH) of screen underflow per square foot of screen area. Based on this value, what screen size (in sq. ft) would you specify to size 100 TPH of your feed material?

Conclusions

1. In your own words, what was the objective of this laboratory exercise?
2. What were your major findings?
3. What important fundamental concepts did you learn from the exercise?

Helpful Equations

$$\text{Sample Loss(\%)} \quad 100 \left(\frac{m_{\text{head}} - m_{\text{sample}}}{m_{\text{head}}} \right)$$

$$\text{Average Size in Class}(\ell) \quad X_{i \times j} = \sqrt{X_i \times X_j}$$

except for last size class where: $X_{i \times 0} = e^{-\frac{1}{X_i}}$

$$\text{Mass Mean Diameter}(\ell) \quad \bar{X} = \frac{\sum_{i=1}^n m_{i \times (i+1)} \times X_{i \times (i+1)}}{\sum_{i=1}^n m_{i \times (i+1)}}$$

$$\text{Linear Model} \quad y_i = b_0 + b_1 x_i + e_i$$

$$\text{Best estimate for } b_1 \quad \hat{b}_1 = \frac{\text{Cov}(x, y)}{\text{Var}(x)} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

$$\text{Best estimate for } b_0 \quad \hat{b}_0 = \bar{y} - b_1 \bar{x}$$

$$\text{Gaudin-Schuhmann Equation} \quad Y = \left(\frac{X}{k} \right)^m$$

$$\text{Linearized GS Equation} \quad \ln(Y) = m \ln(X) - m \ln(k)$$

$$\text{GS Distribution Modulus} \quad m = \frac{\sum_{i=1}^n (\ln(X_i) - \overline{\ln(X)}) (\ln(Y_i) - \overline{\ln(Y)})}{\sum_{i=1}^n (\ln(X_i) - \overline{\ln(X)})^2}$$

$$\text{GS Size Modulus} \quad k = \frac{\left(\prod_{i=1}^n X_i \right)^{\frac{1}{n}}}{\left(\prod_{i=1}^n Y_i \right)^{\frac{1}{nm}}}$$

Where: X_i is the size of screen i
 Y_i is the fraction passing screen i

$$\overline{\ln(Y)} = \frac{1}{n} \sum_{i=1}^n \ln(Y_i)$$

$$\overline{\ln(X)} = \frac{1}{n} \sum_{i=1}^n \ln(X_i)$$

$$\text{Rosin-Rammler Equation} \quad Y = 1 - e^{-\left(\frac{X}{R}\right)^m}$$

$$\text{Linearized R-R Eqn} \quad \ln(-\ln(1 - Y)) = m \ln(X) - m \ln(R)$$

$$\text{R-R Distribution Modulus} \quad m = \frac{\sum_{i=1}^n (\ln(X_i) - \overline{\ln(X)}) (\ln(-\ln(1 - Y_i)) - \overline{\ln(-\ln(1 - Y))})}{\sum_{i=1}^n (\ln(X_i) - \overline{\ln(X)})^2}$$

$$\text{R-R Size Modulus} \quad k = \frac{\left(\prod_{i=1}^n X_i \right)^{\frac{1}{n}}}{\left(\prod_{i=1}^n -\ln(1 - Y_i) \right)^{\frac{1}{nm}}}$$

Where: $\overline{\ln(-\ln(1 - Y))} = \frac{1}{n} \sum_{i=1}^n \ln(-\ln(1 - Y_i))$

Copy-and-Paste™ Calculation of Fitting Constants in R

It is possible to complete this lab in excel or by doing calculations by hand. However, after the installation of R and R studio and adjustment of two lines of code to incorporate your data, Copy-and-Paste™ technology allows for calculations and plotting more accurately and efficiently than previously mentioned methods.

Given the following data set:

Screen Size (mm)	Fraction Passing
0.01	0.081
0.09	0.182
0.84	0.396
7.65	0.702
70.00	0.949

The data collected showed that for mesh sizes (mm): $X = \{0.01, 0.09, 0.84, 7.65, 70\}$ we saw a fraction passing of: $Y = \{0.081, 0.182, 0.396, 0.702, 0.949\}$. Define vectors X and Y as is done below. Note that the order of the data in X and Y is consistent when creating these vectors. The third entry in X and the third entry in Y are on the same row in the table above. You will also need to substitute your name in for the `user.name` variable inside the quotation marks.

```
X <- c(0.01, 0.09, 0.84, 7.65, 70)
Y <- c(0.081, 0.182, 0.396, 0.702, 0.949)

user.name <- "Your Name Here"
```

To do this, first open up a new R script file (Ctrl+Shift+N), then Copy-and-Paste™ the code above from the pdf on Canvas. Substitute your data, ensuring each entry is separated by a comma, then highlight it and hit Ctrl+Enter to run the code. Make sure you use fraction passing, a number between 0 and 1, not percent. To finish the analysis, Copy-and-Paste™ the rest of the code below.

After taking the natural log of the Gaudin Schumann equation, we get the linear model:

$$\ln(Y) = m \ln(X) - m \ln(k)$$

We can use the equations for the fitting constants in a linear model to find m and k . First we take the natural log of X and Y :

```
ln.X <- log(X)
gs.ln.Y <- log(Y)
```

We know from the formula sheet that $m = \frac{s_{\ln(X), \ln(Y)}}{s_{\ln(X)}^2}$. We calculate m as such:

```
m <- cov(ln.X, gs.ln.Y)/var(ln.X)
m
```

```
## [1] 0.2831552
```

Then, we can calculate k using the linear regression formula for the intercept of a line: $-m \ln(k) = \overline{\ln(Y)} - m \overline{\ln(X)}$ implies $k = \exp(-\frac{1}{m} \overline{\ln(Y)} + \overline{\ln(X)})$:

```
lnX.bar <- mean(ln.X)
gs.lnY.bar <- mean(gs.ln.Y)
k <- exp(-(1/m)*gs.lnY.bar + lnX.bar)
k
```



```
## [1] 42.05382
```

Hint for Discussion Q1: When you substitute k for X in $\left(\frac{X}{k}\right)^m$, what fraction passing do you get? (You get the same result from your data set)

```
(k/k)^m
```

```
## [1] 1
```

You can Copy-and-Paste™ the following code to produce a plot for your

For the Rosin-Rammler fit, the only difference in the calculations is that we use $\ln(-\ln(1-Y))$ instead of $\ln(Y)$, since the linearized form is $\ln(-\ln(1-Y)) = m \ln(X) - m \ln(R)$. Transform the initial data in this way, and run the code below. Note what variable names have been changed.

```
rr.ln.Y <- log(-log(1-Y))
rr.lnY.bar <- mean(rr.ln.Y)
rr.m <- cov(ln.X, rr.ln.Y)/var(ln.X)
rr.m
```

```
## [1] 0.4027139
```

```
R <- exp(-(1/rr.m)*rr.lnY.bar + lnX.bar)
R
```

```
## [1] 4.698044
```

Another Hint for Discussion Q1: When you substitute R for X in $1 - e^{-\left(\frac{X}{R}\right)^m}$, what fraction passing does the equation give? (You also get the same result for your data set)

```
1-exp(-(R/R)^rr.m)
```

```
## [1] 0.6321206
```

To generate plots, first save your R script. Then copy the following code into the script:

```
setwd(dirname(rstudioapi::getActiveDocumentContext())$path))

size.vec <- seq(from = 0.01, to = 0.1, by = 0.01)
size.use <- numeric()
for(i in 1:4){
  size.use <- c(size.use,size.vec*10^(i-1))
}
size.use <- unique(size.use)
size.majors <- 0.01*10^seq(from = 0, to = 4, by = 1)
size.majors.gs <- log(size.majors)
frac.pass <- seq(from = 0.01, to = 0.1, by = 0.01)
frac.pass <- c(frac.pass,frac.pass*10)
frac.pass.majors <- c(0.01,0.1,1)
frac.pass.majors.gs <- log(frac.pass.majors)
frac.pass <- unique(frac.pass)
frac.pass <- frac.pass[-10]
gs.transf.size <- log(size.use)
gs.transf.frac <- log(frac.pass)
```

```

main.label <- paste("G-S Plot By: ", user.name)
user.first <- unlist(strsplit(user.name, split = " ")) [1]
user.last <- unlist(strsplit(user.name, split = " ")) [2]

pdf(paste(user.last, user.first, "G-S_Plot.pdf", sep = "_"))
par(mar = c(4,4,4,1))
plot(1, ylim = range(gs.transf.frac), xlim = range(gs.transf.size),
     main = main.label, xlab = "Size (mm)", ylab = "Cumulative Fraction Passing",
     xaxt = "n", yaxt = "n", cex = 0.8)

abline(v = gs.transf.size, lty = 2, col = "grey", lwd = 0.7)
abline(v = size.majors.gs, lty = 2)
abline(h = gs.transf.frac, lty = 2, col = "grey", lwd = 0.7)
abline(h = frac.pass.majors.gs, lty = 2)
axis(1, at = size.majors.gs, labels = as.character(size.majors), cex.axis = 0.5)
axis(2, at = gs.transf.frac, labels = FALSE, cex.axis = 0.6)
axis(2, at = gs.transf.frac[c(1, seq(from = 2, to = 10, by = 2),
                               seq(from = 11, to = 19, by = 2))], tick = FALSE,
     labels = frac.pass[c(1, seq(from = 2, to = 10, by = 2),
                               seq(from = 11, to = 19, by = 2))], cex.axis = 0.5)
points(ln.X, gs.ln.Y, pch = 19, col = "red")
lines(gs.transf.size, m*gs.transf.size - m*log(k))
abline(v = log(k), col = "red", lty = 2)
text(x = log(k), y = log(0.01),
     labels = paste("k = ", round(k, digits = 3), "mm", sep = ""), adj = c(0,1))
dev.off()

size.majors.rr <- log(size.majors)
frac.pass.majors.rr <- log(-log(1-frac.pass.majors))
frac.pass <- unique(frac.pass)
frac.pass <- frac.pass[-10]
rr.transf.size <- log(size.use)
rr.transf.frac <- log(-log(1-frac.pass))
main.label <- paste("R-R Plot By: ", user.name)

pdf(paste(user.last, user.first, "R-R_Plot.pdf", sep = "_"))
par(mar = c(4,4,4,1))
plot(NA, ylim = range(rr.transf.frac), xlim = range(rr.transf.size),
     xlab = "Size (mm)", main = main.label,
     ylab = "Cumulative Fraction Passing", xaxt = "n", yaxt = "n", cex = 0.8)
abline(v = rr.transf.size, lty = 2, col = "grey", lwd = 0.7)
abline(v = size.majors.rr, lty = 2)
abline(h = gs.transf.frac, lty = 2, col = "grey", lwd = 0.7)
abline(h = frac.pass.majors.rr, lty = 2)
abline(h = rr.transf.frac[length(rr.transf.frac)], lty = 2)
abline(h = log(-log(1-0.6321)), lty = 2, col = "red")
axis(1, at = size.majors.rr, labels = as.character(size.majors), cex.axis = 0.5)
axis(2, at = rr.transf.frac, labels = FALSE, cex.axis = 0.6)
axis(2, at = rr.transf.frac[c(1, seq(from = 2, to = 10, by = 2),
                               seq(from = 11, to = 19, by = 2))],
     tick = FALSE, labels = frac.pass[c(1, seq(from = 2, to = 10, by = 2),
                               seq(from = 11, to = 19, by = 2))],
     cex.axis = 0.5)

```

```
points(ln.X, rr.ln.Y, pch = 19, col = "blue")
lines(rr.transf.size, rr.m*rr.transf.size - rr.m*log(R))
abline(v = log(R), col = "red", lty = 2)
text(x = log(R), y = log(0.01),
      labels = paste("R = ", round(R, digits = 3), "mm", sep = ""), adj = c(0,1))
dev.off()
```

Mesh Sizes

Standard	U.S.	Tyler	(mm)	Standard	U.S.	Tyler	(mm)
125 mm	5"	–	125	3.35 mm	No. 6	6 M	3.35
106 mm	4.24"	–	106	2.80 mm	No. 7	7 M	2.8
100 mm	4"	–	100	2.36 mm	No. 8	8 M	2.36
90 mm	3-1/2"	–	90	2.00 mm	No. 10	9 M	2
75 mm	3"	–	75	1.70 mm	No. 12	10 M	1.7
63 mm	2-1/2"	–	63	1.40 mm	No. 14	12 M	1.4
53 mm	2.12"	–	53	1.18 mm	No. 16	14 M	1.18
50 mm	2"	–	50	1.00 mm	No. 18	16 M	1
45 mm	1-3/4"	–	45	850 um	No. 20	20 M	0.85
37.5 mm	1-1/2"	–	37.5	710 um	No. 25	24 M	0.71
31.5 mm	1-1/4"	–	31.5	600 um	No. 30	28 M	0.6
26.5 mm	1.06"	1.050"	26.5	500 um	No. 35	32 M	0.5
25.0 mm	1"	–	25	425 um	No. 40	35 M	0.425
22.4 mm	7/8"	0.883"	22.4	355 um	No. 45	42 M	0.335
19.0 mm	3/4"	0.742"	19	300 um	No. 50	48 M	0.3
16.0 mm	5/8"	0.624"	16	250 um	No. 60	60 M	0.25
13.2 mm	0.530"	0.525"	13.2	212 um	No. 70	65 M	0.212
12.5 mm	1/2"	–	12.5	180 um	No. 80	80 M	0.18
11.2 mm	7/16"	0.441"	11.2	150 um	No. 100	100 M	0.15
9.5 mm	3/8"	0.371"	9.5	125 um	No. 120	115 M	0.125
8.0 mm	5/16"	2-1/2 M	8	106 um	No. 140	150 M	0.106
6.7 mm	0.265"	3 M	6.7	90 um	No. 170	170 M	0.09
6.3 mm	1/4"	–	6.3	75 um	No. 200	200 M	0.075
5.6 mm	No. 3-1/2	3-1/2 M	5.6	63 um	No. 230	250 M	0.063
4.75 mm	No. 4	4	4.75	53 um	No. 270	270 M	0.053
4.00 mm	No. 5	5	4	45 um	No. 325	325 M	0.045
(cont.)	(cont.)	(cont.)	(cont.)	38 um	No. 400	400 M	0.038