Scilab Textbook Companion for
Metal Cutting Theory And Practice
by D. A. Stephenson And J. S. Agapiou[1]

Created by
Venkata Ajay Kumar G
M Tech
Mechanical Engineering
Jntu Anantapur Aits Rajampet
College Teacher
None
Cross-Checked by
None

July 18, 2019

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website http://scilab.in

Book Description

Title: Metal Cutting Theory And Practice

Author: D. A. Stephenson And J. S. Agapiou

Publisher: Crc Taylor & Francis.

Edition: 3

Year: 2016

ISBN: 978-1-4665-8754

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes

Chapter 2

Metal Cutting Operations

Scilab code Exa 2.1 Calculation of Material removal rate and Power and Torque requ

```
clear; // Remove clear , clc if you want to access the stored variables
clc
// Example 2.1
// Given that
D_one = 50 //Initial diamter in mm
f = 0.4 // Feed for the tool in mm/rev
d = 4 // Depth of cut in mm
v = 70 // Cutting speed in m/min
u_s = 0.065  // Specific cutting energy is obtained from Table 2.1 units kW/cm^3/min
mrr = v*f*d // Material Removal Rate and 1000 mm/m is for the conversion
P = mrr*u_s // The machining power required in the turning operation
D_two = D_one - (2*d)
N = (2*v*1000)/(%pi*(D_one+D_two)) // The Speed (rpm) - 1000 value is for the conversion 1000 mm/m
T = (P*1000*60)/(2*%pi*N)  // Spindle Torque
printf("\n The Material Removal Rate = %.0f cm^3/min, \n The Machining Power = %.2f kW, \n Torque
```

```
required by the Spindle = %.0f N m",mrr,P,T)
// The answers vary due to round off error
```

Scilab code Exa 2.2 Calculation of Machining Time

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.2
// Given that
D_one = 50 //Initial diamter in mm
f = 0.4 // Feed for the tool in mm/rev
d = 4 // Depth of cut in mm
v = 70 // Cutting speed in m/min
u_s = 0.065  // Specific cutting energy is obtained from Table 2.1 units kW/cm^3/min
L = 150 // Axial length of the outer diameter in mm
La = 3 // Approach distance in mm (assumption)
D_two = D_one - (2*d)
N = (2*v*1000)/(%pi*(D_one+D_two)) // The Speed (rpm) - 1000 value is for the conversion 1000 mm/m
tm = (L+La)/(f*N) // Machining time in Minutes
tm_s = ((L+La)/(f*N))*60
printf("\nThe Machining time  = %.3f min, \nThe Machining time in Seconds  = %.1f s",tm,tm_s)
// The answers vary due to round off error
```

Scilab code Exa 2.3 Calculation of uncut chip thickness and Cutting Edge engagemen

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.3
```

```
// Given that
D_one = 50 //Initial diamter in mm
f = 0.4 // Feed for the tool in mm/rev
d = 4 // Depth of cut in mm
v = 70 // Cutting speed in m/min
kappa = 20 // Lead angle of the tool in degrees
s = 0.053  // Specific cutting energy is obtained from Table 2.1 units kW/cm^3/min
a = f*cosd(kappa) // The uncut (nominal) chip thickness in mm
Lm = d*s*(kappa)
printf("\nThe uncut (nominal) chip thickness  = %.3f mm, \n The cutting edge engagement = %.3f mm",a,Lm)
// The answers vary due to round off error
```

Scilab code Exa 2.4 Calculation of Taper on the sidewall of the workpiece

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.4
// Given that
V = 100 // Cutting Velocity in m/min
D = 75 // Diameter of the workpiece in mm
l = 200 // Length of the cut in mm
d = 5 // Depth of cut in mm
ad = 80 // Axial depth of cut in mm
rd = 5 // Radial depth of cut in mm
amax = 0.2 //  Maximum uncut chip thickness in mm/rev
E = 400000 // Modulus of elasticity of the material in N/mm^2
us = 0.08 // Unit specific energy for work material from Table 2.1 in kW/cm^3/min
```

```
nt = 4
N = V*1000/(%pi*D) // Spindle rotational speed in rpm so with 1000 multiplying
vm = acosd(1-((2*d)/D))
ft  = amax / sind(vm) // Feed per tooth in mm
fr = nt*ft*N // Feed rate for the cutter in mm/min
cs = ad*rd // Cross-sectional area of uncut chip in mm^2
mrr = cs*fr  // Material removal rate in mm^3/min , in Power required convert to cm^3/min divide by 1000
P = (mrr/1000)*us // Power required for the operation
F = (P*1000*60)/V  // Cutting force , where 1000 is for converting kW to Nm/s and 60 for s/min
Fr = 0.3*F  // Radial Force , Radial component is 30% of the cutting (tangential) force
//delta = (Fr*)
I = (%pi*(0.8*D)^4)/64
deflection = (Fr*l^3)/(3*E*I)
taper_error = D*(deflection/(2*l))
printf("The taper on the sidewall of the workpiece = %.4f ",taper_error)
// The answer provided in the textbook is wrong
```

Scilab code Exa 2.5 Calculation of Material Removal Rate

```
clear; // Remove clear , clc if you want to access the stored variables
clc
// Example 2.5
// Given that
D = 50 // Diameter of the cutter in mm
IC = 17 // IC size of the round inserts in mm
d = 3 // Depth of cut in mm
```

```
nt = 3 // Number of inserts
V = 150 // Cutting speed in m/min
amax = 0.2 //  Maximum uncut chip thickness in mm/tooth feed
De = D - IC + sqrt((IC^2)-((IC-(2*d))^2)) // The Effective tool diameter
vm = acosd(1-((2*d)/IC))
N = V*1000/(%pi*De) // The spindle speed should be calculated based on the effective diameter in cut in rpm
ft = amax / sind(vm)
fr = nt*ft*N // The feed rate of the cutter in mm/min
b = De // Radial depth of cut in mm
mrr = d*b*fr // Material removal rate in mm^3/min
printf("\nThe Material removal rate  = %.0f mm^3/min",mrr)
// The answers vary due to round off error
// The answer provided in the textbook is wrong for the mrr
```

Scilab code Exa 2.6 Calculation of Material Removal Rate

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.6
// Given that
d = 6 // Depth of cut in mm
D = 25 // Diameter of the tool in mm
a_max = 0.2 // Feed in mm/tooth
V = 150 // Cutting speed in m/min
nt = 2
b = 11 // Radial doc will be half of the effective diameter
```

```
De = 2*sqrt((d*(D-d))) // Effective tool diameter
// Effective tool diameter is not the actual cutter diameter because the axial depth of cut is smaller than the radius of the ballnose
// Since the depth of cut is significantly smaller than the radius of the nose chip thinning occurs
vm = acosd (1-((2*d)/D))
ft = a_max/sind(vm)
N = (V*1000)/(%pi*De) // Spindle speed in rpm and Effective tool diameter need to be considered
// The answers vary due to round off error
fr = nt * ft * N // Feed rate of the cutter
mrr = d*b*fr // Material removal rate in mm^3/min
printf("\n The Material Removal Rate = %.0f mm^3/min ",mrr)
// The answer provided in the textbook is wrong
```

Scilab code Exa 2.7 Calculation of Machining Time

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.7
// Given that
V = 300 // Cutting speed in m/min
D = 200 // Diameter of the face milling cutter in mm
ai = 0.3 // un-cut chip thickness in mm
k = 45 // Lead angle of the cutter in degrees
b = 150 // width of the cut in mm
L = 700 // Length of the cut in mm
nt = 18 // Number of inserts on the cutting tool
N = V*1000/(%pi*D) // Spindle rotational speed in rpm
ft = ai/cosd(k) // Feed per tooth in mm
fr = nt*ft*N // Feed rate of the cutter in mm/min
```

```
Le_max = (D/2)+(D/2) // Maximum length the cutter can travel
Le = (D/2)-((1/2)*(sqrt((D^2)-(b^2)))) // Length of the approach in mm
tm = (L+Le)/fr // Machining time in min
printf("\n The Machining time = %.2f min",tm)
```

Scilab code Exa 2.8 Calculation of Maximum feed rate and Machining time

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.8
// Given that
L = 400 // Lenght of the part in mm
P = 10 // Maximum available cutting power in kW
us = 0.07 // Specific energy forthe workpiece material from Table 2.1 in W.min/cm^3
d = 15 // Radial depth of cut in mm
b = 10 // Width of cut in mm
V = 80 // Cutting speed in m/min
D = 200 // Diameter of the carbide brazed side milling cutter in mm
R = 100 // Radius of the carbide brazed side milling cutter in mm
nt = 16 // Number of inserts on the cutter
n_t = 8 // Number of inserts which are one side of the cutter
a_max = 0.2 // Maximum uncut chip thickness in mm
L_e = 52.7 // Approach distance in mm
mrr = P / us // Material removal rate in cm^3/min
fr = (mrr*1000) / (d*b) // Feed rate in mm/min
N = V*1000/(%pi*D) // Speed of the spindle in rpm
ft = fr / (nt*N) // Feed rate for the cutter in mm/rev/tooth
```

```
V_m = acosd((R-d)/R)
f_t = a_max/sind(V_m) // Maximum feed per tooth allowed in mm/rev/tooth
tm = (L+L_e+10)/(nt*f_t*N)
t_m = (L+L_e+10)/(n_t*f_t*N)
printf("\n Maximum feed per tooth allowed = %.3f mm^/rev/tooth,\n Machining time to groove the part = %.3f min,\n Machining time to groove the part with 8 inserts = %.3f min",f_t,tm,t_m)
// If the cutter has 16 inserts in total, the effective number of teeth is 8 because 8 inserts on one side of the cutter and 8 overlapping inserts on the other side are used to remove the full width of 10 mm
```

Scilab code Exa 2.9 Calculation of Maximum power and Cutting time and Perpendicula

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.9
// Given that
D = 200 // Outside diameter of the flange in mm
Di = 100 // Inside diameter of the flange in mm
N = 400 // Maximum rotational speed in rpm
L = 500 // Length of the tube in mm
ft = 0.25 // Maximum feed rate in mm/rev
d = 8 // Depth of cut in mm
us = 0.025 // Specific Energy from Table 2.1 in W.min/cm^3
E = 50000 // Modulus of elasticity of the material in N/mm^2
mrr = (%pi*D*N*ft*d) // Material removal rate in mm^3/min
P = (mrr/1000)*us  // Maximum power require in kW (
```

```
   the conversion unit of 1000 to formula for mm to
   cm)
tm = (D-Di)/(2*ft*N)
// The perpendicularity error is equivalent to the
   deflection of the tube under the cutting load
F_c = (P*1000)/(%pi*(N/60)*(D/1000)) // Cutting
   force in N; 1000 for kW conversion, 60 for rev/s
   conversion and 1000 for mm to m
I = (%pi*(D^4-Di^4))/64  // Cross sectional moment
   of inertia
deflection = (F_c*(L^3))/(3*E*I)
taper_error = D*(deflection/(2*L))
printf("\n The Maximum power = %.1f kW,\n Cutting
   time = %.1f min,\n The taper or perpendicularity
   error is %.3f mm",P,tm,taper_error)
```

Scilab code Exa 2.10 Calculation of Number of teeth on the broach and length of th

```
clear; // Remove clear, clc if you want to access
   the stored variables
clc
// Example 2.10
// Given that
f_t = 0.1 // Feed per tooth in mm
d = 3 // Deep or Depth of cut in mm
p = 12.5 // Thread pitch in mm
L_e = 80 // Length to be broach in mm
w = 12 // Slot Wide in mm
us = 0.07 // Specific energy of the material from
   Table 2.1 in kW min/cm^3
V = 25 // Cutting speed in m/min
n_t = d/f_t // Number of teeth
L_b = p*n_t // Length of broach
tm = (L_b + L_e)/(V*1000) // Machining time in min,
   1000 is for conversion unit for V
```

```
n_te = ceil(L_e/p) // Number of teeth engage with
    the workpiece
d_e = n_te*f_t // Maximum depth engage of the
    broaching tool in mm
mrr = d_e*w*V*1000  // Material removal rate in mm
    ^3/min
F_c = ((mrr/1000)*us*60)/V // Force in kN
printf("Number of teeth on the Broach = %.0f teeth,\
    n Length of the broch = %.0f mm \n The Machining
    time = %.3f min \n Cutting force = %.1f kN ",n_t,
    L_b,tm,F_c)
```

Scilab code Exa 2.11 Calculation of Machining Time

```
clear; // Remove clear, clc if you want to access
    the stored variables
clc
// Example 2.11
// Given that
f_t = 0.1 // Feed per tooth in mm/rev
n_t = 4 // Number of teeth
V  = 100 // Cutting speed in m/min
D = 25 // Diameter of the tool in mm
p = 3 // Thread pitch in mm
eta = 0.7 // Percentage of thread
D_u = 100 // Minor diameter in mm
L = 30 // Depth in mm
N = (V*1000)/(%pi*D) // Speed of the drill in rev/
    min so conversion unit 1000 multiplied
f_rt = n_t*f_t*N // The Feed rate at the cutting
    edge
D_m =  eta*(1.299*p) + D_u // Major diameter in mm
f_r = f_rt*(D_m - D)/D_m // Centerline Feed rate in
    mm/min
D_x = D_m - D // Travel distance for single orbit
```

```
L_x = sqrt((p^2)+((%pi)^2)*((D_x)^2)) // Total length of travel of the tool in mm
n_orbit = L /(3*p)
tm = n_orbit*(L_x/f_r) // Time for machining in min
// This computed time is for cutting and does not including the approach and exit distances for each orbit
printf("\n The Feed rate at the cutting edge = %.0f mm/min \n The total travel length = %.0f mm \n The Machining time = %.1f min",f_rt,L_x, tm)
// The answers vary due to round off error
```

Scilab code Exa 2.12 Calculation of Material Removal Rate and Cutting Time and Tor

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.12
// Given that
D = 10 // Diameter of the drill in mm
f = 0.3 // Feed of the drill in mm/rev
V = 200 // Cutting speed in m/min
L = 30 // Length of hole to be drilled in mm
row = 60 // Drill point angle in degrees, in question it was 120, divided by 2 is row
delta_L = 3 // Assume the approach and over travel distance for the drill as 3mm
us = 0.008 // For magnesium alloys specific cutting energy from Table 2.1 0.008 W.min/cm^3
N = (V*1000)/(%pi*D) // Speed of the drill in rev/min so conversion unit 1000 multiplied
fr = N*f // Feed rate
mrr = ((%pi*(D^2))/4)*fr // Material removal rate in mm^3/min, need in cm^3/min divide by 1000
tm = (L+(D/tand(row))+delta_L)/fr // Cutting time in
```

```
min, if multiplied by 60 cutting time will be in seconds
P = mrr*us // Power required to drill the hole in N.m/s
T = P*60 / (2*%pi*N) // Torque on the drill in N.m, Muliplied by 60 for conversion to s/min
printf("\n The Material Removal Rate = %.0f mm^3/min \n The Cutting time = %.2f min  \n The Torque on the drill = %.1f N m",mrr,tm,T)
```

Scilab code Exa 2.13 Calculation of Material Removal Rate and Machining Time

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 2.13
// Given that
D = 150 // Diameter of the wheel in mm
V = 2000 // Cutting speedin m/min
V_work = 2200 // Velcoity of the work in mm/min
doc_total = 0.2 // Depth to be ground
d = 0.01 // Grinding depth in mm/pass
Lp = 350 // Length of the workpiece in mm
b = 75 // Width of the workpiece in mm
La = 40
mrr = b*d*V_work
tmp = (Lp+D+La)/mrr // Time to trvel across the part length
tm = ((doc_total/d)*tmp)+tmp
 printf("\nThe Material Removal Rate = %.2f mm^3/min,\n The Machining time =  %.2f min ",mrr,tm)
// The answers vary due to round off error
```

Scilab code Exa 2.14 Calculation of Power required to Grind and the Machining Time

```
clear; // Remove clear , clc if you want to access
   the stored variables
clc
// Example 2.14
// Given that
Dw = 75 // Diameter of the workpiece in mm
d = 0.015 // Grinding depth in mm/pass
fr = 1500 // Traverse feed rate in mm/min
us = 0.08 // Specific cutting energy in kW min/cm^3
doc_total = 0.2 // Depth to be ground or Total depth
    of cut in mm
Lp = 200 // Length of the bar to be performed in mm
Wt = 25 // Wheel Thickness in mm
La = 8 // Overall travel distance from both ends of
   the workpiece
mrr = %pi*Dw*d*fr/1000 // Material removal rate in
   cm^3/min divide with 1000
// The specific energy for grinding should be much
   larger than that give in Table 2.1 based on
   turngin test. Therefore the value of 0.08 kW min/
   cm^3 for cast iron from Table 2.1 should be
   multiplied by five times for grinding operations
   for a rough estimaate.
P = mrr*(5*us) // Estimated power required
tmp = (Lp+Wt+La)/fr // Time to travel across the
   part length in min/pass
tm = (doc_total*tmp/d)+(2*tmp) // The machining time
    in minutes
tmp = (Lp+Wt+La)/fr // Time travel across the part
   length (to perform one pass) in min/pass
printf("\nThe Power required to grind = %.2f kW,\n
   The Machining time =  %.2f min, \n Time travel
   across the part length (to perform one pass) = %
   .3f min/pass",P,tm,tmp)
// The answers vary due to round off error
```

Chapter 3

Machine Tools

Scilab code Exa 3.1 Calculation of Stiffness of a Strut

```
clear; // Remove clear, clc if you want to access
   the stored variables
clc
// Example 3.1
// Given that
k_s = 140 // Stiffness of the ball screw in N/um
k_g = 25 // Stiffness of the Gimbal joint in N/um
k_t = 1/((1/k_s)+(2*(1/k_g)))
printf("The Stiffness of the strut = %.1f N/um",k_t)
```

Chapter 4

Cutting Tools

Scilab code Exa 4.1 Calculation of Material Removal Rate for both the drill and th

```
clear; // Remove clear, clc if you want to access
    the stored variables
clc
// Example 4.1
// Given that
Vd = 100 // Cutting speed for uncoated carbide drill
     in m/min
Dm = 10 // Diameter of drill from figure 4.4
D = 9 // Diameter of drill for a 10mm is about 9mm
Vt = 30 // Cutting speed of HSS tap, Assumption
f = 0.15 // Feed for drilling cast iron assumed in
    mm/flute
L = 27 // The bottom depth of the blind hole assumed
     in mm
row = 120 // Drill point angle in degrees
p = 1 // Pitch for the M10 x 1mm tap is 1 mm
n_t = 2 // Number of Inserts
Lt = 23
Le = 2  // Assumed since a hole is made in mm
N_d = Vd*1000/(%pi*D) // Spindle speed for drilling
    in rpm ; 1000 is for the conversion unit
```

```
// The answer provided in the textbook is wrong for N_d
N_t = Vt*1000 / (%pi*Dm) // Spindle speed for tapping in rpm ;  1000 is for the conversion unit
f_r = n_t*f*N_d // Feedrate for drilling in mm/min
f_rt = p*N_t // Feedrate for tapping in mm/min
t_md = (L+(D/tand(row/2))+2)/f_r  // Cutting time for drilling with 120degrees point drill in min
t_mt = (Lt+Le)/f_rt  // Cutting time for tapping in min
mrr_d = ((%pi*(D^2))/4)*f_r // Material removal rate  for the drill in mm^3/min
// The answer provided in the textbook is wrong for mrr
lambda = atand(p/(%pi*Dm))
mrr_t = ((p/4)+((Dm-D)/tand(row/2)))*((Dm-D)/4)*((p*N_t)/sind(lambda)) // Material removal rate for the tapping in mm^3/min
// The answer provided in the textbook is wrong for mrr
printf("\n The Material removal rate for drilling = %.0f mm^3/min, \n The Material removal rate for tapping = %.0f mm^3/min", mrr_d,mrr_t)
```

Scilab code Exa 4.2 Calculation of Change in Machining Time for Drilling and Tappi

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 4.2
// Given that
Vd = 30 // Cutting speed for uncoated HSS drill in m/min
D = 9 // Diameter of drill for a 10mm is about 9mm
n_t = 2
```

```
L = 27 // The bottom depth of the blind hole assumed in mm
f = 0.15 // Feed for drilling cast iron assumed in mm/flute
row = 120 // Drill point angle in degrees
N_d = Vd*1000/(%pi*D) // Spindle speed for drilling in rpm ; 1000 is for the conversion unit
// The answer provided in the textbook is wrong for N_d
f_r = n_t*f*N_d // Feedrate for drilling in mm/min
// The answer provided in the textbook is wrong for f_r
t_md = (L+(D/tand(row/2))+2)/f_r  // Cutting time for drilling with 120degrees point drill in min
// t_T = 1.78 + 1.57 + delta_t  // The total machining time to drill and tap the hole is the sum of the drilling and tapping times plus several other tool travel times i.e., retract the tool from the bottom of hole, approach time.
// t_T2 = 6 + 1.57 + delta_t
// t_Tx = t_T2 - t_T = 7.57 + delta_t - 3.35 - delta_t
t_Tx = 7.57 - 3.35
printf("\n The change in machining time for drilling and tapping = %.2f s",t_Tx)
```

Scilab code Exa 4.4 Calculation of the Deflection at the tool point for bar A and

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 4.4
// Given that
E_steel = 206700 // Youngs modulus of steel in MPa
E_heavymetal = 330000 // Youngs modulus of heavy
```

```
metal in MPa
D_i = 55 // Initial diameter of the bore in mm
D_o = 59 // Final diameter of the bore in mm
D_B = 50 // Diameter of the Bar B in mm
D_abar = 35 // Diameter of the bore in mm
L = 250 // Length of the bar in mm
V = 100 // Cutting speed of the bar in m/min
fr = 0.2 // Feed in mm/rev
us = 0.06 // Specific energy from Table 2.1
D_avg = (D_i + D_o)/2 // Diameter average in mm
N = (V*1000)/(%pi*D_avg) //
R = (D_i + D_o)/4
mrr = ((%pi*((D_o^2) - (D_i^2)))/4)*N*fr
F_r = (mrr*us)/(4*%pi*R*N/(60*1000)) // conversion unit 60 for min to seconds and 1000 for m to mm
I_A = ((%pi*(D_abar^4))/64)  // The answer for the I_A was wrong in the textbook
// The answers vary due to round off error
I_B = ((%pi*(D_B^4))/64)
deflection_A = (F_r*(L^3))/(3*E_steel*I_A)
// The deflection of 0.246 mm is very large and the hole diameter will be smaller by 0.492 mm and will generate the scratch marks during tool retraction. Therefore, the deflection should be reduced by reducing the radial force, which is proportional to the area of cut. Hence, the doc or feed must be changed in order to reduce the force. Since a large reduction on the deflection is required, both doc and feed will be reduced by 50%. However, the reduction of the DOC by 50% will require two passes to remove the 2 mm full depth from the bore. The reduction of the feed from 0.2 to 0.1 mm /rev is acceptable but the machining time will be doubled. The reduction of the cutting conditions results in lower productivity
F_r2 = F_r/4
deflection_2 = deflection_A/4
```

```
deflection_heavymetal = deflection_A *(E_steel/
    E_heavymetal)
deflection_heavymetal2 = deflection_heavymetal/4 //
    Using the 50% reduced feed and DOC, the new
    deflection
deflection_B = F_r*(((L^3)/(3*E_steel*I_A))+((L^3)
    /(3*E_steel*I_B))+(((L*L)/(E_steel*I_B))*(L+L)))
printf("The deflection at the cutting tool point for
     bar A = %.3f mm, \n The deflection at the
    cutting tool point changing the boring bar
    material from steel to heavy metal = %.3f mm,\n
    The deflection at the cutting tool point for bar
    B = %.3f mm",deflection_A,deflection_heavymetal2,
    deflection_B)
// The answers vary due to round off error
```

Chapter 5

Toolholders and Workholders

Scilab code Exa 5.3 Calculation of total balancing grade

```
clear; // Remove clear , clc if you want to access
    the stored variables
clc
// Example 5.3
// Given that
W_spindle = 20 // Mass of the spindle in kg
W_holder = 1.27 // Mass of the toolholder in kg
W_tool = 0.35 // Mass of the tool in kg
G = 1.17 // Quality grade as defined in ANSI/S2.19
rpm_spindle = 29250 // Maximum operating speed (r/
    min) from the graph in Fig. 5.82
rpm_holder = 11697 // Maximum operating speed (r/min
    ) from the graph in Fig. 5.82
rpm_tool = 4644 // Maximum operating speed (r/min)
    from the graph in Fig. 5.82
U_spindle = (9549*G*W_spindle)/rpm_spindle
U_holder =  (9549*G*W_holder)/rpm_holder
U_tool =  (9549*G*W_tool)/rpm_tool
U_total = U_spindle + U_holder + U_tool
printf("The total balancing grade of an assembled \n
     spinlde-toolholder-tool system = %.3f g mm",
```

```
U_total)
```

Scilab code Exa 5.4 Calculation of Unbalance force reduction

```
clear; // Remove clear, clc if you want to access
    the stored variables
clc
// Example 5.4
// Given that
U = 30 // Unbalance in mm
N = 15000 // Speed of the tool holder in rpm
F_r = (U*((2*%pi*N)/60)^2 )*10e-7  // Unbalance
    force reduction in N
printf("The Unbalance force reduction = %.0f N", F_r
    )
// The radial force for 30 g mm unbalance at 15000
    rpm is 74N and for 75 g mm unbalance the force
    becomes 185 N. Hence the force reduction is
    significant if a pre-balanced tool holder is used
```

Scilab code Exa 5.5 Calculation of allowable unbalance for a balance quality grade

```
clear; // Remove clear, clc if you want to access
    the stored variables
clc
// Example 5.5
// Given that
G = 2.5 // Quality grade as defined in ANSI/S2.19
W = 3.63 // Estimated total tool weight in kg
N = 10000 // Speed of the tool holder in rpm
U = (9549*G*W)/N
printf("The allowable Unbalance for the boring bar =
     %.2f g mm", U)
```

```
// The answers vary due to round off error
```

Scilab code Exa 5.6 Determining the effect of toolholder clamping length

```
clear; // Remove clear, clc if you want to access
   the stored variables
clc
// Example 5.6
// Given that
F_1 = 80 // Cutting force on the boring bar in N
L_1 = 7 // Extended length of the boring bar 7XD
L_3 = 2 // Tool shank length A bar
L_3B = 4 // Tool shank length B bar
F_3 = (F_1*L_1)/L_3  // Force acting on the back end
    of the boring bar A in N
F_2 = F_1+F_3  // Force acting at the front of the
   chuck in bar A in N
F_3B = (F_1*L_1)/L_3B  // Force acting on the back
   end of the boring bar B in N
F_2B = F_1+F_3B  // Force acting at the front of the
    chuck in bar B in N
printf("The Force at the bar A = %.0f N, \n The
   Force at the bar B = %0.f N",F_2,F_2B)
printf("\n The reaction force acting at the free end
    of the chuck is %.0f N \n is smaller than %.0f N
    at bar A, \n This will reduce the deformation of
    the chuck at the free end",F_2B,F_2 )
```

Scilab code Exa 5.8 Calculation of deflection at the tool point

```
clear; // Remove clear, clc if you want to access
   the stored variables
clc
```

```
// Example 5.8
// Given that
k_s = 1.375e9 // Spring constant linear for CAT40 from Table 5.7 in n/m
k_h = 3e8  // Spindle stiffness in N/m (linear)
k_theta = 8.108e6 //  Spindle stiffness in N/m (rotational)
D1 = 25 // Diameter of the tool-holder spindle in mm from figure 5.106
D2 = 44 // Major Diameter of the tool-holder spindle in mm from figure 5.106
f_r = 0.15   // Feed in mm/rev
Fr = 185 // Radial cutting force in N
V = 100 // Cutting speed in m/min
L1 = 170 //Length of the tool-holder in mm from figure 5.106
L2 = 40 //Length of the tool-holder chuck in mm from figure 5.106
L = L1+L2  // Total length of the tool-holder and chuck in mm from figure 5.106
La = 20 // Distance between the spindle nose and the rotational spring in mm
E1 = 206700 // Youngs modulus of steel in MPa
E2 = 206700 // Youngs modulus of steel in MPa
I1 = (%pi*D1^4)/64 // The answers vary due to round off error
I2 = (%pi*D2^4)/64
k = 1/((1/k_s)+(1/k_h))
deflection_1 = (Fr/k)*1000  // Linear deflection in N, 1000 is the conversion unit for m to mm
deflection_2 = f_r*(((L1^3)/(2*E1*I1))+((L2^3)/(2*E2*I2))+(((L1*L2)*(L1+L2))/(E2*I2)))*1000  //The deflection of the bar structure with two cross sections in mm, 1000 is the conversion unit for m to mm
// The answers vary due to round off error
theta = (Fr*(L+La))/(k_theta*1000) // Angle in radians
```

```
deflection_r = (L+La)*tan(theta) // Since the theta
    value is in radians, not inserting tand here.
deflection_total = deflection_1+deflection_2+
    deflection_r
// The answer provided in the textbook is wrong
printf("The total deflection = %.3f mm",
    deflection_total)
```

Chapter 6

Mechanics of Cutting

Scilab code Exa 6.1 `Calculation of mean angle of friction and mean shear stress an`

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 6.1
// Given that
b = 2.5 // Thickness of tube in mm
a = 0.3 // feed in mm/rev
V = 200 // cutting speed in m/min
ac = 0.7 // chip thickness in mm
lf = 0.5 // tool-chip contact length in mm
Fc = 900 // tangential cutting force in N
Fz = 600 // Axial (feed) force in N
alpha = 0 // Rake angle of tool is Zero
tsz = 0.02 // thickness of shear zone assumed as 0.02 mm
// Sample Problem 6.1 on page no. 436
printf("\n  Problem 6.1  \n")
beta = atand(Fz/Fc)
rc = a/ac // Cutting ratio
phi = atand((rc*cosd(alpha)/(1-(rc*sind(alpha)))))
    // Mean shear angle between the direction of
```

```
cutting speed and the shear plane
tow_s = ((((Fc*cosd(phi))-(Fz*sind(phi)))*sind(phi))/(a*b)) // Mean shear stress of the workpiece
tow_f = ((sqrt((Fc^2)+(Fz^2))*sind(beta))/(b*lf)) // Mean frictional stress at the tool-chip interface
gamma = cotd(phi)+(tand(phi-alpha))
delta_t = (tsz*60)/(V*1000*sind(phi))
gamma_dot = gamma/delta_t // Strain rate in the chip formation if thickness of shear zone assumed as 0.02 mm
FE = (((sqrt((Fc^2)+(Fz^2))*sind(beta))*rc)/Fc)*100 // Fricition energy
printf("\n Mean angle of friction on the tool face = %.1f ,\n Mean shear stress produced in cutting the workpiece = %.0f N/mm^2,\n Mean frictional stress at the tool-chip interface = %.0f N/mm^2 \n Shear Strain = %.2f ,\n Strain rate in the chip formation = %.0f s^-1, \n Friction energy accounts for %.0f Percentage of the total energy ", beta,tow_s,tow_f,gamma,gamma_dot, FE)
// The answers vary due to round off error
```

Scilab code Exa 6.2 Calculation of Tangential force and feed force

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 6.2
// Given that
alpha = -8 // Rake angle of tool -8 from the Problem 6.1
b = 2.5 // Thickness of tube in mm
a = 0.3 // feed in mm/rev
V = 200 // cutting speed in m/min
```

```
ac = 0.85 // chip thickness in mm
lf = 0.5 // tool-chip contact length in mm
Fc = 900 // tangential cutting force in N
Fz = 600 // Axial (feed) force in N
tsz = 0.02 // thickness of shear zone assumed as 0.02 mm
rc = a/ac // Cutting ratio
phi = atand ((rc*cosd(alpha))/(1-(rc*sind(alpha))))
// The shear angle is reduced from 23.2 to 21.8 degrees as expected since the rake angle was reduced.
tow_s = ((((Fc*cosd(phi))-(Fz*sind(phi)))*sind(phi))/(a*b)) // Mean shear stress of the workpiece
F_s = (tow_s*a*b)/sind(phi)
beta_a = atand(Fz/Fc)+alpha // Beta angle calculated from Eq 6.30 using Figure 6.27
R = F_s/(cosd(phi+beta_a-alpha))  // Resultant force in N
F_c = R*cosd(beta_a-alpha) // Cutting force in N
F_z = R*sind(beta_a-alpha) // Feed force in N
printf("The Tangential (cutting) force = %.0f N, \n The Feed force = %.0f N",F_c,F_z)
// The answer provided in the textbook is wrong for mrr
```

Chapter 9

Tool Wear and Tool life

Scilab code Exa 9.2 Calculation of parameters for the extended Taylor tool life eq

```
clear; // Remove clear , clc if you want to access the stored variables
clc
// Example 9.2
// Given that
V_one = 70 // Cutting speed
V_two = 70 // Cutting speed
V_three = 140
f_1 = 0.15
f_3 = 0.15
f_2 = 0.25
T_1 = 120 // Mean tool life in min
T_2 = 105 // Mean tool life in min
T_3 = 24 // Mean tool life in min
n = log(V_three/V_one)/log(T_1/T_3)  // (V_three/V_one = (T_1/T_3)^n)
a = n*(log(T_2/T_1)/log(f_1/f_2))
K_t = V_one*(T_1^n)*(f_1^a)
printf("The parameters for extended tool life = %.2f & %.2f, \n The tool life equation = %.0f",n,a,K_t)
```

```
// The answers vary due to round off error
```

Chapter 10

Surface Finish Integrity and Flatness

Scilab code Exa 10.1 Calculation of Maximum feed

```
clear; // Remove clear, clc if you want to access
   the stored variables
clc
// Example 10.1
// Given that
R_a = 0.003 // Surface finish in micro meter
K_reone = 10 // Minor cutting edge angle (ECEA) in
   degrees
K_re = 20 // Lead angleof the tool (SCEA) in degrees
alpha  = K_re+K_reone
f = 4*R_a*(cotd(90 - alpha)+ cotd(K_reone))  // Feed
f_max = f/1.4  // Maximum feed
printf("The Maximum feed = %.3f mm",f_max)
```

Scilab code Exa 10.2 Calculation of Corner Radius

```
clear; // Remove clear, clc if you want to access the stored variables
clc
// Example 10.2
// Given that
R_a = 0.001 // Surface finish in micro meter
f = 0.1 // Feed in mm/rev
r_n = (0.0321*f^2)/R_a  // Average Geometric roughness from Eq.10.10
printf("The Average Geometric roughness = %.3f mm", r_n)
```

Chapter 13

Machining Economics and Optimization

Scilab code Exa 13.2 Calculation of Production cost for a rough turning

```
clear; // Remove clear , clc if you want to access
   the stored variables
clc
// Example 13.2
// Given that
L = 200 // Length of the bar in mm
L_e = 2 // Approach length in mm assumed
L_t = 80
f_rapid = 6000 // Feed in mm/min from the Table 13.1
C_o = 1 // it was given 60$/h from the Table 13.1 ,
   when convert to min
nt = 3 // Number of cutting edges per insert
cte = 9 // Tool cost in dollars per insert
f = 0.3 // Feed rate in mm/rev from the Table 13.1
N = 700 // Spindle rpm from the Table 13.1
D = 70 // Diameter of the bar in mm
t_h = 20 // Part load/unload time in seconds
t_l = 1 // Tool load/unload time in Minutes
t_cs = 8 // Tool interchange time between operations
```

```
    in seconds
n = 0.25 // Tool life exponent from the Table 13.1
ct = 500 // Tool life constant from the Table 13.1
t_m = (L+L_e)/(f*N) // Machining time in min
V = %pi*D*N/1000  // Cutting speed in m/min
T = ct^4*(V^-4) // Tool life in min VT^n = Ct
t_x = (L+L_t)/f_rapid
C_te = cte/nt // Per cutting edge in dollars
C_u = C_o*t_m + ((t_m/T)*((C_o*t_l)+C_te))+(C_o*((
    t_cs/60)+(t_h/60)+t_x))
printf("The Production cost = $ %.2f",C_u)
```

Scilab code Exa 13.3 Calculation of Minimum Production Cost

```
clear; // Remove clear, clc if you want to access
    the stored variables
clc
// Example 13.3
// Given that
L = 200 // Length of the bar in mm
L_e = 2 // Approach length in mm assumed
L_t = 80
f_rapid = 6000 // Feed in mm/min from the Table 13.1
C_o = 1 // it was given 60$/h from the Table 13.1,
    when convert to min
nt = 3 // Number of cutting edges per insert
cte = 9 // Tool cost in dollars per insert
f = 0.3 // Feed rate in mm/rev from the Table 13.1
fh = 0.5 // Maximum allowable feed rate in mm/rev
    given in example 13.3
a = 0.5 // Exponent
D = 70 // Diameter of the bar in mm
t_h = 20 // Part load/unload time in seconds
t_l = 1 // Tool load/unload time in Minutes
t_cs = 8 // Tool interchange time between operations
```

```
    in seconds
n = 0.25 // Tool life exponent from the Table 13.1
Kt = 500 // Tool life constant from the Table 13.1
t_x = (L+L_t)/f_rapid
C_te = cte/nt // Per cutting edge in dollars
V_opc = Kt/((fh^a)*((((1-n)/n)*(t_l+(C_te/C_o)))^n))
N = (V_opc*1000)/(%pi*D)
t_m = (L+L_e)/(fh*N) // Machining time in min
T = Kt^4*(V_opc^-4)*(fh^-2) // Tool life in min VT^n
    = Ct
// The answer in the textbook is wrong
C_u = C_o*t_m + ((t_m/T)*((C_o*t_l)+C_te))+(C_o*((
   t_cs/60)+(t_h/60)+t_x))
printf("The Production cost = $ %.2f, \n The
   production cost is reduced from $1.51 to $ %.3f
   by using an optimum cutting speed",C_u,C_u)
```

Chapter 16

Accuracy and Error Compensation of CNC Machining Systems

Scilab code Exa 16.1 Calculation of magnitude of the Abbe error

```
clear; // Remove clear, clc if you want to access
   the stored variables
clc
// Example 16.1
// Given that
deflection_y = 0.01 // Curvature of the x-axis in
   the y-direction in mm
X = 1000
P = 600  // Offset of the part feature to be
   machined in mm
theta_x = (8*deflection_y)/X
delta = P*sind(theta_x/2)
printf("The Magnitude of the Abbe error = %.6f mm",
   delta)
// The answers vary due to round off error
```