Table 1 MATLAB® script that accepts the flow rate (*Q*), bottom slope of channel (*J*), Manning's roughness coefficient (*n*), and depth of uniform flow (*h*) as inputs in a dialog box and returns a message box displaying the side slope (*m*) and verified flow rate (Q_c) for the trapezoidal channel depicted in Problem 1

```
%Solution of Problem #1
clear %Clear variables and functions from memory
clc %Clear command window and home the cursor
format short %Scaled fixed point format with 5 digits
disp(datestr(now,0)) %String representation of date without ans

%STEP #1 (Definition of inputs)
a1 = {'Flow rate (L/h)','Bottom slope of channel (m/km)',...
   'Manning's roughness coefficient',...
   'Bottom width of channel (m)','Depth of uniform flow (m)'}; %Title of inputs
a2 = 'Inputs'; %Title of dialog box
a3 = [1 40; 1 40; 1 40; 1 40; 1 40]; %Size of input fields
a4 = inputdlg(a1,a2,a3); %Create input dialog box
tic %Start a stopwatch timer

%STEP #2 (Extraction of inputs from dialog box)
b = char(a4); %Create character array
c = str2num(b); %Convert character array to numeric array
Q = (c(1)./1000)./3600; %Flow rate (m3/s)
J = c(2)./1000; %Bottom slope of channel (m/m)
n = c(3); %Manning's roughness coefficient
B = c(4); %Bottom width of channel (m)
h = c(5); %Depth of uniform flow (m)

%STEP #3 (Calculations)
m = 0; %Initial guess for side slope (see figure)
s = 0.0000001; %Sensitivity value
while 1 %Infinite loop
L = 2.*m.*h + B; %Top width of channel (m)
y = sqrt((h.^2)+((m.*h).^2)); %Length of side slope (m)
A = ((B + L)./2).*h; %Wetted area (m2)
P = 2.*y + B; %Wetted perimeter (m)
R = A./P; %Hydraulic radius (m)
Qc = (1./n).*(R.^(2./3)).*sqrt(J).*A; %Computed flow rate (m3/s)
m = m + s; %Counter for side slope
if (Qc >= Q) %Conditional statement
break %Terminate execution of while loop
end %Termination of if statement
end %Termination of while statement

%STEP #4 (Convert numbers to character representation)
m_s = sprintf('Side slope (m) = %1.4f',m);
Qc_s = sprintf('Verified flow rate = %1.4f L/h',Qc.*3600.*1000);
d = char(m_s,Qc_s); %Create character array
msgbox(d,'Results'); %Create a message box
toc %Read the stopwatch timer
```

Table 2 MATLAB® script that accepts the angle of slope (θ), bottom width of channel (*B*), depth of uniform flow (*h*), and increase in flow rate (*x*) in a dialog box and returns a message box displaying the amount of expansion (*L*) for the trapezoidal channel depicted in Problem 2

```
%Solution of Problem #2
clear %Clear variables and functions from memory
clc %Clear command window and home the cursor
format short %Scaled fixed point format with 5 digits
disp(datestr(now,0)) %String representation of date without ans
```

```
%STEP #1 (Definition of inputs)
a1 = {'Angle of slope (°)','Bottom width of channel (m)',...
    'Depth of uniform flow (m)',...
    'Increase in flow rate (e.g., 2, 3, etc.)'}; %Title of inputs
a2 = 'Inputs'; %Title of dialog box
a3 = [1 40; 1 40; 1 40; 1 40]; %Size of input fields
a4 = inputdlg(a1,a2,a3); %Create input dialog box
tic %Start a stopwatch timer

%STEP #2 (Extraction of inputs from dialog box)
b = char(a4); %Create character array
c = str2num(b); %Convert character array to numeric array
theta = c(1); %Angle of slope (°)
B = c(2); %Bottom width of channel (m)
h = c(3); %Depth of uniform flow (m)
x = c(4); %Increase in flow rate

%STEP #3 (Calculations)
n = 0.013; %Randomly selected Manning’s roughness coefficient
J = 1./1000; %Randomly selected channel slope (m/m)
syms L %Create a short-cut for construction of symbolic L
L1 = 2.*(h./tand(theta))+B; %Top width of channel before expansion (m)
A1 = ((B+L1)./2).*h; %Wetted area before expansion (m2)
P1 = 2.*(h./(sind(theta)))+B; %Wetted perimeter before expansion (m)
R1 = A1./P1; %Hydraulic radius before expansion (m)
Q1 = (1./n).*(R1.^(2./3)).*sqrt(J).*A1; %Flow rate #1 (m3/s)
A2 = ((B+L+L1+L)./2).*h; %Wetted area after expansion (m2)
P2 = P1+L; %Wetted perimeter after expansion (m)
R2 = A2./P2; %Hydraulic radius after expansion (m)
Q2 = (1./n).*(R2.^(2./3)).*sqrt(J).*A2; %Flow rate #2 (m3/s)
K = x.*Q1-Q2; %Algebraic equation (set equal to zero for symbolic solution)
Y = vpasolve(K,L); %Numerical solution of algebraic equation
W = double(Y); %Symbolic to double object conversion
ind = find(W>0); %Find indices of nonzero elements
L = W(ind); %Amount of expansion (m)

%STEP #4 (Convert numbers to character representation)
L_s = sprintf('Amount of expansion (L) = %1.4f m',L);
F = char(L_s); %Create character array
msgbox(F,'Results'); %Create a message box
toc %Read the stopwatch timer
```

Table 3 MATLAB® script that accepts the flow rate (Q), slope of channel (J), elevation values (e_1, e_2, e_3), width values (w_1, w_2, w_3, w_4), and Manning’s roughness coefficients (n_1, n_2, n_3, n_4) in a dialog box and returns a message box displaying the depth value (h) and verified flow rate (Q_c) for the compound channel depicted in Problem 3

```
%Solution of Problem #3
clear %Clear variables and functions from memory
clc %Clear command window and home the cursor
format short %Scaled fixed point format with 5 digits
disp(datestr(now,0)) %String representation of date without ans

%STEP #1 (Definition of inputs)
a1 = {'Flow rate (L/s)','Slope of channel (m/m)',...
    '1st elevation (m)','2nd elevation (m)','3rd elevation (m)',...
    '1st width (m)','2nd width (m)','3rd width (m)','4th width (m)',...
    '1st Manning’s roughness coefficient','2nd Manning’s roughness coefficient',...
    '3rd Manning’s roughness coefficient','4th Manning’s roughness coefficient'};
```

```
%Title of inputs
a2 = 'Inputs'; %Title of dialog box
a3 = [1 40; 1 40; 1 40; 1 40; 1 40; 1 40; 1 40; 1 40;...
   1 40; 1 40; 1 40; 1 40; 1 40]; %Size of input fields
a4 = inputdlg(a1,a2,a3); %Create input dialog box
tic %Start a stopwatch timer

%STEP #2 (Extraction of inputs from dialog box)
b = char(a4); %Create character array
c = str2num(b); %Convert character array to numeric array
Q = c(1)./1000; %Flow rate (m3/s)
J = c(2); %Slope of channel (m/m)
e1 = c(3); %1st elevation (m)
e2 = c(4); %2nd elevation (m)
e3 = c(5); %3rd elevation (m)
w1 = c(6); %1st width (m)
w2 = c(7); %2nd width (m)
w3 = c(8); %3rd width (m)
w4 = c(9); %4th width (m)
n1 = c(10); %1st Manning’s roughness coefficient
n2 = c(11); %2nd Manning’s roughness coefficient
n3 = c(12); %3rd Manning’s roughness coefficient
n4 = c(13); %4th Manning’s roughness coefficient

%STEP #3 (Calculations)
h = 0; %Initial guess for h value (see figure)
s = 0.0000001; %Sensitivity value (m)
while 1 %Infinite loop
   x = h+1; %Calculation of x depth (see figure);
   y = h-1; %Calculation of y depth (see figure);
   A1 = w1.*h; %Wetted area of the 1st segment (m2)
   P1 = w1+h; %Wetted perimeter of the 1st segment (m)
   R1 = A1./P1; %Hydraulic radius of the 1st segment (m)
   A2 = w2.*(h+e1); %Wetted area of the 2nd segment (m2)
   P2 = e1+w2+e2; %Wetted perimeter of the 2nd segment (m)
   R2 = A2./P2; %Hydraulic radius of the 2nd segment (m)
   A3 = w3.*x; %Wetted area of the 3rd segment (m2)
   P3 = w3+e3; %Wetted perimeter of the 3rd segment (m)
   R3 = A3./P3; %Hydraulic radius of the 3rd segment (m)
   A4 = w4.*y; %Wetted area of the 4th segment (m2)
   P4 = w4+y; %Wetted perimeter of the 4th segment (m)
   R4 = A4./P4; %Hydraulic radius of the 4th segment (m)
   Q1 = (1./n1).*(R1.^(2./3)).*sqrt(J).*A1; %Flow rate #1 (m3/s)
   Q2 = (1./n2).*(R2.^(2./3)).*sqrt(J).*A2; %Flow rate #2 (m3/s)
   Q3 = (1./n3).*(R3.^(2./3)).*sqrt(J).*A3; %Flow rate #3 (m3/s)
   Q4 = (1./n4).*(R4.^(2./3)).*sqrt(J).*A4; %Flow rate #4 (m3/s)
   Qc = Q1+Q2+Q3+Q4; %Total flow rate (m3/s)
   h = h + s; %Counter for h value
   if (Qc >= Q) %Conditional statement
      break %Terminate execution of while loop
   end %Termination of if statement
end %Termination of while statement

%STEP #4 (Convert numbers to character representation)
h_s = sprintf('h value = %1.4f m',h);
Qc_s = sprintf('Verified flow rate = %1.4f L/s',Qc.*1000);
K = char(h_s,Qc_s); %Create character array
msgbox(K,'Results'); %Create a message box
toc %Read the stopwatch timer
```

Table 4 MATLAB® script that accepts the flow rate difference (dQ), initial wastewater level below half-filled level (h_1), slope of channel (J), Manning's roughness coefficient (n), wastewater increment (h_w) after storm event in a dialog box and returns a message box displaying the diameter value (D) for the circular combined sewer depicted in Problem 4

```
%Solution of Problem #4
clear %Clear variables and functions from memory
clc %Clear command window and home the cursor
format short %Scaled fixed point format with 5 digits
disp(datestr(now,0)) %String representation of date without ans

%STEP #1 (Definition of inputs)
a1 = {'Flow rate difference (L/s)','Initial WWL below half-filled level (cm)',...
    'Slope of channel (per thousand)','Manning's roughness coefficient',...
    'Wastewater increment after storm event (cm)'}; %Title of inputs
a2 = 'Inputs'; %Title of dialog box
a3 = [1 40; 1 40; 1 40; 1 40; 1 40]; %Size of input fields
a4 = inputdlg(a1,a2,a3); %Create input dialog box
tic %Start a stopwatch timer

%STEP #2 (Extraction of inputs from dialog box)
b = char(a4); %Create character array
c = str2num(b); %Convert character array to numeric array
dQ = c(1)./1000; %Flow rate difference (m3/s);
h1 = c(2)./100; %Initial WWL below half-filled level (m)
J = c(3)./1000; %Slope of channel
n = c(4); %Manning's roughness coefficient
hw = c(5)./100; %Wastewater increment after storm event (m)

%STEP #3 (Calculations)
syms D %Create a short-cut for construction of symbolic D
R = D./2; %Radius of channel (m)
theta = acosd(h1./R); %Half of the wastewater surface angle #1 (°)
A1 = (1./2).*R.*R.*sind(2.*theta); %Area of triangle #1 (m2)
A2 = (pi.*(R.^2).*(2.*theta))./360; %Area of entire circle slice #1 (m2)
A3 = A2-A1; %Wetted area #1 (m2)
L1 = (2.*pi.*R.*(2.*theta))./360; %Length of circle arc #1 (m)
Q1 = (1./n).*((A3/L1).^(2./3)).*sqrt(J).*A3; %Flow rate in dry condition (m3/s)
h2 = hw-h1; %WWL above half-filled level (m)
beta = acosd(h2./R); %Half of the wastewater surface angle #2 (°)
A4 = (1./2).*R.*R.*sind(2.*beta); %Area of triangle #2 (m2)
A5 = (pi.*(R.^2).*(2.*beta))./360; %Area of entire circle slice #2 (m2)
A6 = A5-A4; %Area of empty part (m2)
A7 = pi.*(D.^2)./4; %Area of entire circle (m2)
A8 = A7-A6; %Wetted area #2 (m2)
L2 = (2.*pi.*R.*(2.*beta))./360; %Length of circle arc #2 (m)
L3 = 2.*pi.*R; %Perimeter of entire circle (m)
L4 = L3-L2; %Length of circle arc #3 (m)
Q2 = (1./n).*((A8/L4).^(2./3)).*sqrt(J).*A8; %Flow rate in storm condition (m3/s)
X = dQ-(Q2-Q1); %Algebraic equation (set equal to zero for symbolic solution)
Y = vpasolve(X,D); %Numerical solution of algebraic equation
W = double(Y); %Symbolic to double object conversion
ind = find(W>0); %Find indices of nonzero elements
D = W(ind); %Diameter of channel (m)

%STEP #4 (Convert numbers to character representation)
D_s = sprintf('Diameter of channel (D) = %1.4f m',D);
F = char(D_s); %Create character array
msgbox(F,'Results'); %Create a message box
toc %Read the stopwatch timer
```