CE 3354 Engineering Hydrology
Exercise Set 5

Exercises

1. a 50-acre single-family residential subdivision receives a rainfall intensity of 3 inches per hour for one hour. The average runoff coefficient is 0.50. Using a rational triangular hydrograph [1]

   Determine:

   a) Maximum (peak) discharge rate for the watershed.

   b) A plot of the discharge hydrograph in 6-minute intervals.

   c) The total volume of runoff from the subdivision for the entire storm.

   Solution(s): Entire problem as Jupyter Notebook (ENGR-1330) on following pages.

[1] Essentially apply the Modified Rational Method with t_c equal to the storm duration

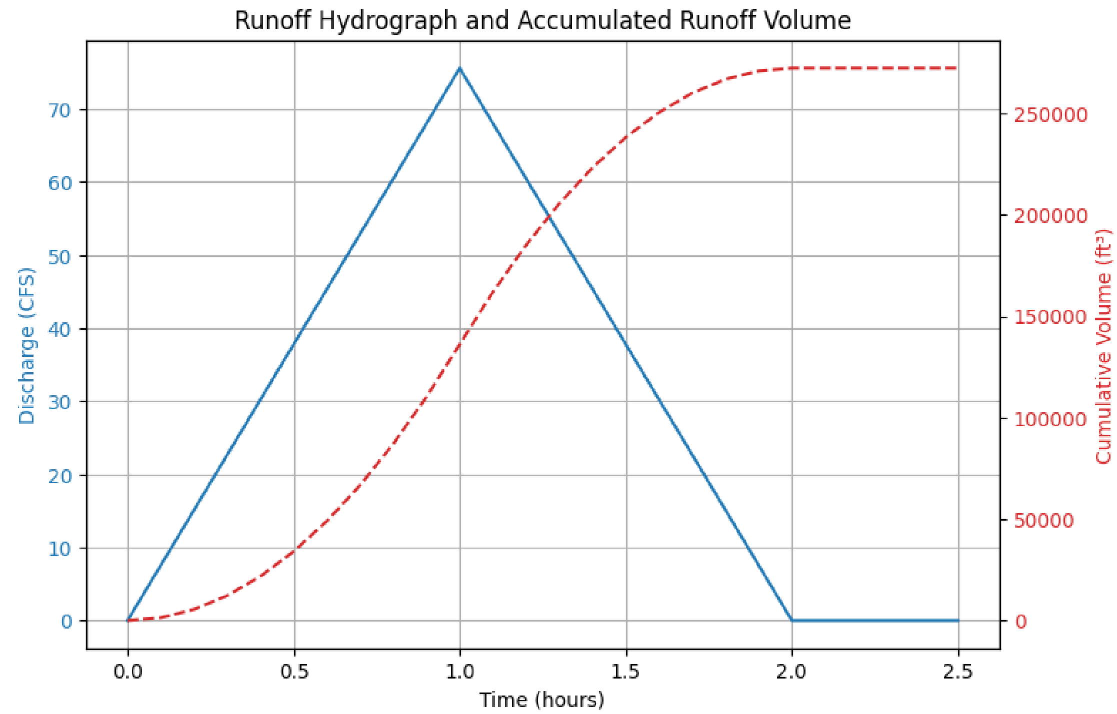
es5-worksheet

August 4, 2025

[64]:
```
# Rational Method Tools
def qpeak(c,i,a,k=1):
    qpeak = k*c*i*a
    return qpeak
#################################
def triangle_hydrograph(qp, time, tc):
    if time < 0:
        triangle_hydrograph = 0.0
    elif 0 <= time <= tc:
        triangle_hydrograph = (qp / tc) * time
    elif tc < time <= 2 * tc:
        triangle_hydrograph = (qp / tc) * time - ( (qp / tc) * (time - tc) ) -␣
↪( (qp / tc) * (time - tc) )
    else:
        triangle_hydrograph = 0.0
    return triangle_hydrograph
############################
# Problem 1
a = 50 # acres given
I = 3 # in/hr given
c = 0.5 # given
tc = 1.0 #hr implied(given)
k = 1.008 #unit conversion 1.008 for US  1/360 for SI
qp=qpeak(c,i,a,k)
print("Rational Method for Peak Discharge")
print(f"Area: {a:.2f} acres")
print(f"Intensity: {i:.2f} inches per hour")
print(f"Runoff Coefficient: {a:.2f} dimensionless")
print(f"Peak Discharge Rate: {qpeak(c, I, a, k):.2f} CFS")
# Recalculate discharge and cumulative runoff
qp = qpeak(c,I,a,k)
time = [] # empty list
discharge = [] #empty list
deltat = 0.1 # 6 minutes in hours
howmanysteps = 26
time.append(0.0)
discharge.append(0.0)
```

```
for i in range(1,howmanysteps):
    time.append(time[i-1]+deltat)
    discharge.append(triangle_hydrograph(qp,time[i],tc))
# Trapezoidal integration to compute cumulative volume
cumulative_volume = [0.0]
for i in range(1, howmanysteps):
    deltaV = 0.5 * (discharge[i] + discharge[i-1]) * deltat * 3600  # ft³
    cumulative_volume.append(cumulative_volume[i-1] + deltaV)
# Plotting both discharge and accumulated volume
fig, ax1 = plt.subplots(figsize=(8, 5))
color = 'tab:blue'
ax1.set_xlabel("Time (hours)")
ax1.set_ylabel("Discharge (CFS)", color=color)
ax1.plot(time, discharge, color=color, label="Discharge (CFS)")
ax1.tick_params(axis='y', labelcolor=color)
ax1.grid(True)
ax2 = ax1.twinx()  # instantiate second axes sharing the same x-axis
color = 'tab:red'
ax2.set_ylabel("Cumulative Volume (ft³)", color=color)
ax2.plot(time, cumulative_volume, color=color, linestyle='--',
 ↪label="Accumulated Volume")
ax2.tick_params(axis='y', labelcolor=color)
fig.tight_layout()
plt.title("Runoff Hydrograph and Accumulated Runoff Volume")
plt.show()
# Final print
print(f"Total Runoff: {cumulative_volume[-1]:.2f} cubic feet")
```

```
Rational Method for Peak Discharge
Area: 50.00 acres
Intensity: 25.00 inches per hour
Runoff Coefficient: 50.00 dimensionless
Peak Discharge Rate: 75.60 CFS
```




```
Total Runoff: 272160.00 cubic feet
```

[ ]:

2. a 50-acre single-family residential subdivision receives a rainfall intensity of 3 inches per hour for one hour. The average runoff coefficient is 0.50. Using the NRCS triangular hydrograph [2]

   Determine:

   a) Maximum (peak) discharge rate for the watershed.

   b) A plot of the discharge hydrograph in 6-minute intervals.

   c) The total volume of runoff from the subdivision for the entire storm.

   Solution(s): Entire problem as Jupyter Notebook (ENGR-1330) on following pages.

[2] t_c is set equal to the storm duration

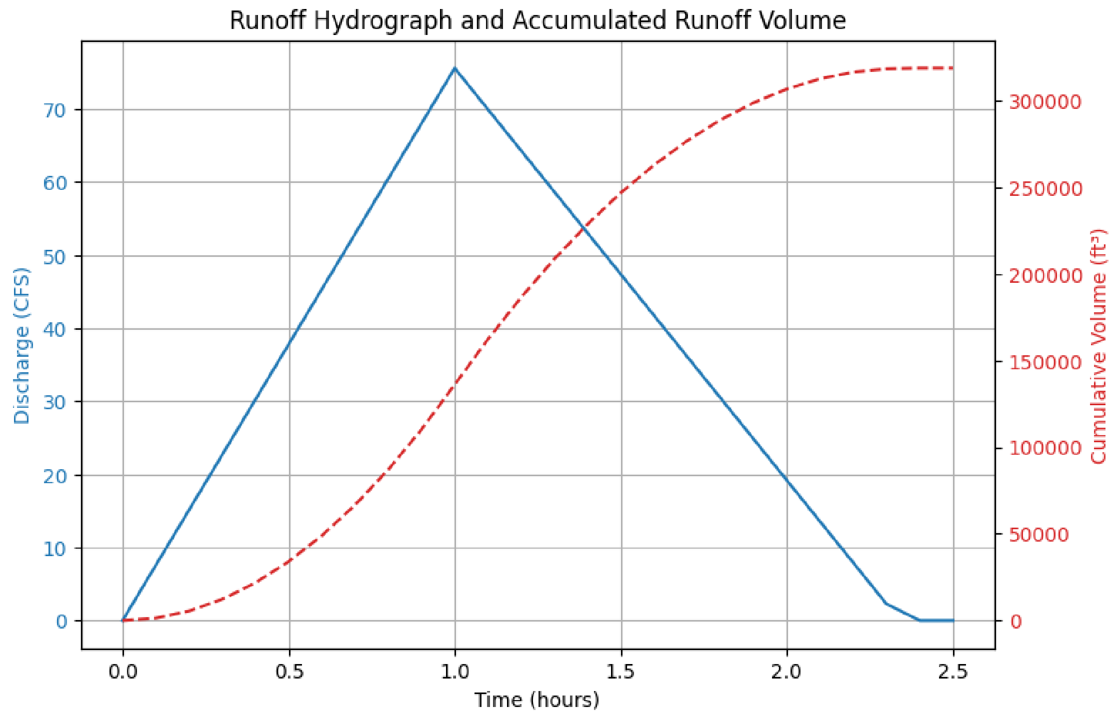
es5-ws1-Copy1

August 4, 2025

[6]:
```
# NRCS Method Tools
import matplotlib.pyplot as plt
#
def qpeak(c,i,a,k=1):
    qpeak = k*c*i*a
    return qpeak
################################
def nrcs_triangle_hydrograph(qp, time, tc):
    if time < 0:
        nrcs_triangle_hydrograph = 0.0
    elif 0 <= time <= tc:
        nrcs_triangle_hydrograph = (qp / tc) * time
    elif tc < time <= 2.34 * tc:
        nrcs_triangle_hydrograph = (qp / tc) * time - ( (qp / tc) * (time - tc)
↪) - ( (qp /1.34*tc) * (time - tc) )
    else:
        nrcs_triangle_hydrograph = 0.0
    return nrcs_triangle_hydrograph
############################
# Problem 2
# NRCS Triangular uses same meanings, but recession limb is 1.34 tc
a = 50 # acres given
I = 3 # in/hr given
c = 0.5 # given
tc = 1.0 #hr implied(given)
k = 1.008 #unit conversion 1.008 for US  1/360 for SI
qp=qpeak(c,I,a,k)
print("NRCS Triangle Method for Peak Discharge (Duration == Tc)")
print(f"Area: {a:.2f} acres")
print(f"Intensity: {I:.2f} inches per hour")
print(f"Runoff Coefficient: {a:.2f} dimensionless")
print(f"Peak Discharge Rate: {qpeak(c, I, a, k):.2f} CFS")
# Recalculate discharge and cumulative runoff
qp = qpeak(c,I,a,k)
time = [] # empty list
discharge = [] #empty list
deltat = 0.1 # 6 minutes in hours
```

```python
howmanysteps = 26
time.append(0.0)
discharge.append(0.0)
for i in range(1,howmanysteps):
    time.append(time[i-1]+deltat)
    discharge.append(nrcs_triangle_hydrograph(qp,time[i],tc))
# Trapezoidal integration to compute cumulative volume
cumulative_volume = [0.0]
for i in range(1, howmanysteps):
    deltaV = 0.5 * (discharge[i] + discharge[i-1]) * deltat * 3600  # ft³
    cumulative_volume.append(cumulative_volume[i-1] + deltaV)
# Plotting both discharge and accumulated volume
fig, ax1 = plt.subplots(figsize=(8, 5))
color = 'tab:blue'
ax1.set_xlabel("Time (hours)")
ax1.set_ylabel("Discharge (CFS)", color=color)
ax1.plot(time, discharge, color=color, label="Discharge (CFS)")
ax1.tick_params(axis='y', labelcolor=color)
ax1.grid(True)
ax2 = ax1.twinx()  # instantiate second axes sharing the same x-axis
color = 'tab:red'
ax2.set_ylabel("Cumulative Volume (ft³)", color=color)
ax2.plot(time, cumulative_volume, color=color, linestyle='--',␣
 ↪label="Accumulated Volume")
ax2.tick_params(axis='y', labelcolor=color)
fig.tight_layout()
plt.title("Runoff Hydrograph and Accumulated Runoff Volume")
plt.show()
# Final print
print(f"Total Runoff: {cumulative_volume[-1]:.2f} cubic feet")
```

```
NRCS Triangle Method for Peak Discharge (Duration == Tc)
Area: 50.00 acres
Intensity: 3.00 inches per hour
Runoff Coefficient: 50.00 dimensionless
Peak Discharge Rate: 75.60 CFS
```




```
Total Runoff: 318670.93 cubic feet
```

[ ]:

3. A watershed is comprised of sandy soil with a 500 foot path to an outlet. The slope on that path is 5-percent. The soil has a high water table limiting the potential watershed storage to 0.5 inches. Using the NRCS Lag Equation method[3]

$$T_c = L^{0.8} \frac{(S_r + 1)^{0.7}}{1140 Y^{0.5}} \tag{1}$$

where:
T_c = time of concentration, hr
L = flow length, ft
S_r = Potential storage (in.); $S_r = \frac{1000}{CN} - 10$
CN = NRCS runoff curve number
Y = average watershed slope, %

Determine:

a) Time of concentration (T_c).

Solution(s): Entire problem as Jupyter Notebook (ENGR-1330) on following pages.

[3] https://directives.nrcs.usda.gov/sites/default/files2/1712930818/31754.pdf

es5-ws3

August 4, 2025

[3]:
```python
# NRCS Method Tools
import matplotlib.pyplot as plt # in case we need to plot

def nrcs_lag(L,Sr,Y):
    term1 = (Sr+1.0)**(0.7)
    term2 = 1140*Y**(0.5)
    term3 = L**(0.8)
    nrcs_lag = term3*term1/term2
    return nrcs_lag

# Problem 3
# NRCS Lag Equation
length = 500 # feet given
slope = 5 # percent given
retention = 0.5 # inches given
tc=nrcs_lag(length,retention,slope)
print("NRCS Lag Equation")
print(f"Length: {length:.2f} feets")
print(f"Slope: {slope:.2f} percent")
print(f"Retention Capacity: {retention:.2f} inches")
print(f"Time of concentration: {tc:.2f} hours")
```

```
NRCS Lag Equation
Length: 500.00 feets
Slope: 5.00 percent
Retention Capacity: 0.50 inches
Time of concentration: 0.08 hours
```

[ ]:
```python

```

4. The runoff hydrograph below was produced by a 100 acre watershed.

Table 1: Somewhere USA Runoff Data

Time (hours)	Runoff (CFS)
0.0	0.0
1.0	70.0
2.0	160.
3.0	110.
4.0	80.0
5.0	60.0
6.0	45.0
7.0	30.0
8.0	20.0
9.0	12.0
10.	5.0
11.	0.0

Determine:

a) Excess precipitation in watershed inches for the hydrograph.

b) A unit hydrograph for the watershed.

c) A plot of the unit hydrograph.

Solution(s): Entire problem as Jupyter Notebook (ENGR-1330) on following pages.

es5-ws4

August 4, 2025

[15]:
```python
# Somewhere USA
import matplotlib.pyplot as plt
time =[0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0,11.0]
runoff=[0.0,70.0,160.0,110.0,80.0,60.0,45.0,30.0,20.0,12.0,5.0,0.0]
# Excess precip == runoff (in watershed inches)
area = 100 #acres given
excess = 12.0*(sum(runoff)*3600)/(area*43560)  #excess in inches
print(f"Excess precipitation: {excess:.2f} inches")
```

```
Excess precipitation: 5.87 inches
```

Here is the data in the Excel worksheet - we will have to assume the response is to a single hour of precipitation (or experiment a bit, we will need unit weights greater than zero)

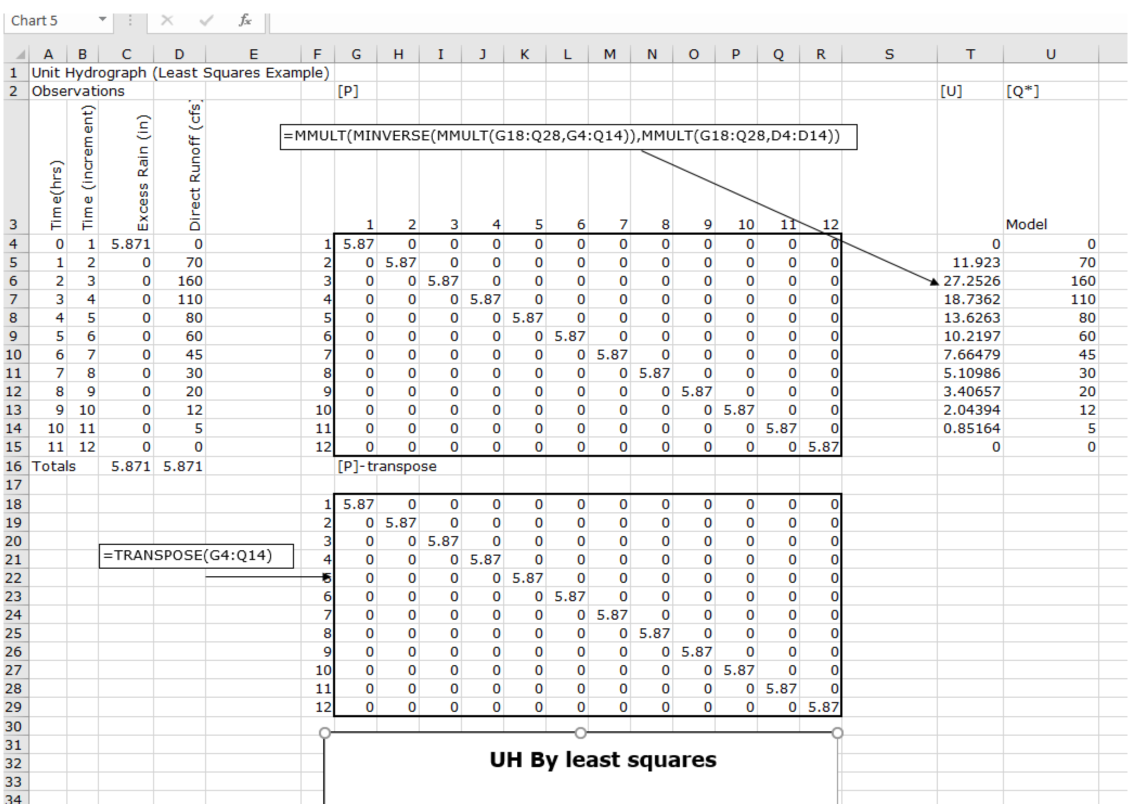


Chart 5

=MMULT(MINVERSE(MMULT(G18:Q28,G4:Q14)),MMULT(G18:Q28,D4:D14))

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Unit Hydrograph (Least Squares Example)																				
2	Observations						[P]													[U]	[Q*]
3	Time(hrs)	Time (Increment)	Excess Rain (in)	Direct Runoff (cfs)			1	2	3	4	5	6	7	8	9	10	11	12			Model
4	0	1	5.871	0		1	5.87	0	0	0	0	0	0	0	0	0	0	0		0	0
5	1	2	0	70		2	0	5.87	0	0	0	0	0	0	0	0	0	0		11.923	70
6	2	3	0	160		3	0	0	5.87	0	0	0	0	0	0	0	0	0		27.2526	160
7	3	4	0	110		4	0	0	0	5.87	0	0	0	0	0	0	0	0		18.7362	110
8	4	5	0	80		5	0	0	0	0	5.87	0	0	0	0	0	0	0		13.6263	80
9	5	6	0	60		6	0	0	0	0	0	5.87	0	0	0	0	0	0		10.2197	60
10	6	7	0	45		7	0	0	0	0	0	0	5.87	0	0	0	0	0		7.66479	45
11	7	8	0	30		8	0	0	0	0	0	0	0	5.87	0	0	0	0		5.10986	30
12	8	9	0	20		9	0	0	0	0	0	0	0	0	5.87	0	0	0		3.40657	20
13	9	10	0	12		10	0	0	0	0	0	0	0	0	0	5.87	0	0		2.04394	12
14	10	11	0	5		11	0	0	0	0	0	0	0	0	0	0	5.87	0		0.85164	5
15	11	12	0	0		12	0	0	0	0	0	0	0	0	0	0	0	5.87		0	0
16	Totals		5.871	5.871			[P]-transpose														
17																					
18						1	5.87	0	0	0	0	0	0	0	0	0	0	0			
19						2	0	5.87	0	0	0	0	0	0	0	0	0	0			
20						3	0	0	5.87	0	0	0	0	0	0	0	0	0			
21			=TRANSPOSE(G4:Q14)			4	0	0	0	5.87	0	0	0	0	0	0	0	0			
22						5	0	0	0	0	5.87	0	0	0	0	0	0	0			
23						6	0	0	0	0	0	5.87	0	0	0	0	0	0			
24						7	0	0	0	0	0	0	5.87	0	0	0	0	0			
25						8	0	0	0	0	0	0	0	5.87	0	0	0	0			
26						9	0	0	0	0	0	0	0	0	5.87	0	0	0			
27						10	0	0	0	0	0	0	0	0	0	5.87	0	0			
28						11	0	0	0	0	0	0	0	0	0	0	5.87	0			
29						12	0	0	0	0	0	0	0	0	0	0	0	5.87			

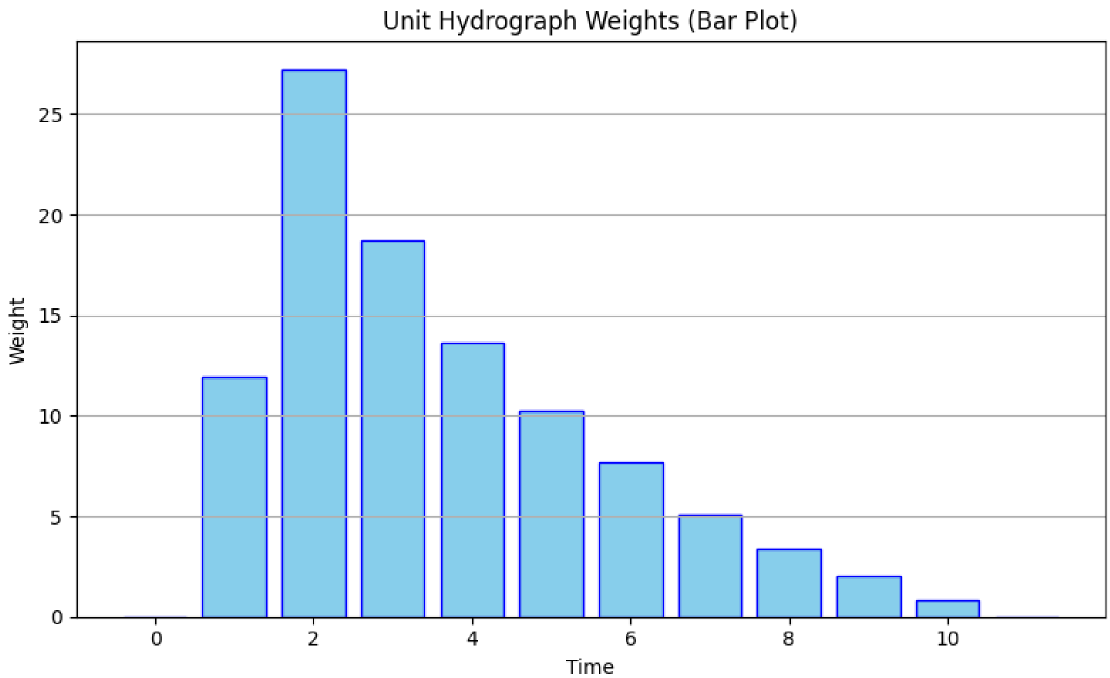
UH By least squares

[16]:
```python
import matplotlib.pyplot as plt
```

```
uh_weights = [0.0, 11.923, 27.2526, 18.7362, 13.6263, 10.2197, 7.66479,
              5.10986, 3.40657, 2.04394, 0.85164, 0.0]
time = list(range(len(uh_weights)))

plt.figure(figsize=(8, 5))
plt.bar(time, uh_weights, color='skyblue', edgecolor='blue')

plt.title("Unit Hydrograph Weights (Bar Plot)")
plt.xlabel("Time")
plt.ylabel("Weight")
plt.grid(True, axis='y')
plt.tight_layout()
plt.show()
```



[17]:
```
# After Modeling using the Excel sheet - rendering results
import matplotlib.pyplot as plt

# === Data Lists ===
excess_rain = [5.871, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
direct_runoff = [0, 70, 160, 110, 80, 60, 45, 30, 20, 12, 5, 0]
model_runoff =   [0, 70, 160, 110, 80, 60, 45, 30, 20, 12, 5, 0]
time = list(range(len(excess_rain)))

# === Create figure and axes ===
fig, ax1 = plt.subplots(figsize=(10, 6))
```

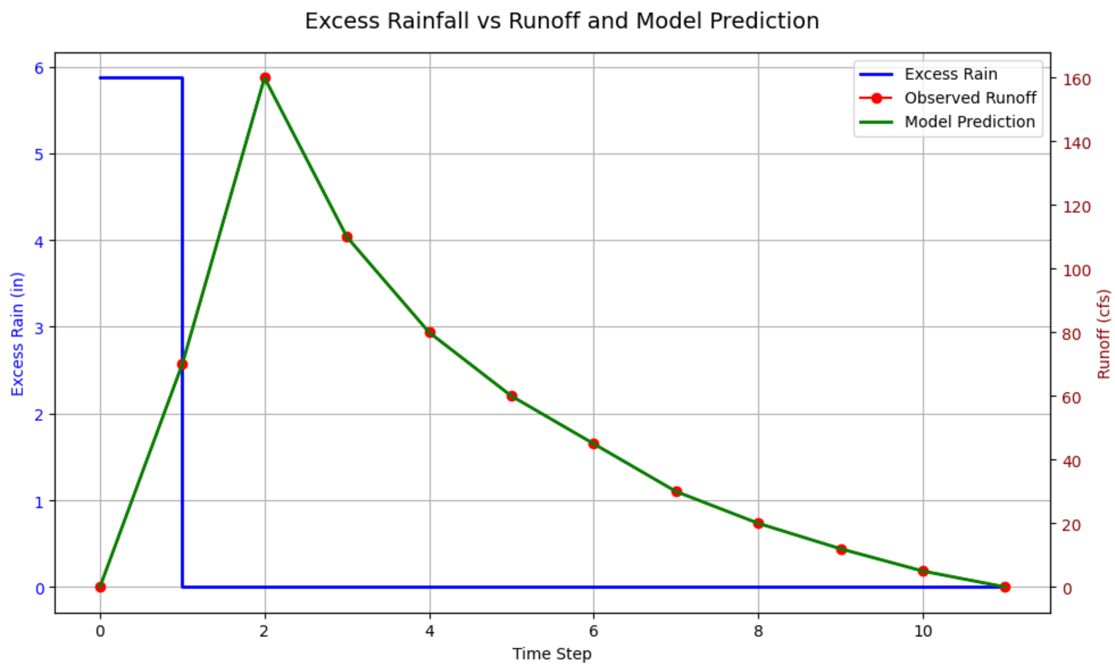
```python
# --- Left Y-axis: Rainfall ---
ax1.set_xlabel("Time Step")
ax1.set_ylabel("Excess Rain (in)", color='blue')
rain_line = ax1.step(time, excess_rain, where='post', color='blue', linewidth=2, label="Excess Rain")[0]
ax1.tick_params(axis='y', labelcolor='blue')

# --- Right Y-axis: Runoff and Model ---
ax2 = ax1.twinx()
ax2.set_ylabel("Runoff (cfs)", color='darkred')
runoff_line, = ax2.plot(time, direct_runoff, 'ro-', label="Observed Runoff", linewidth=1.5)
model_line, = ax2.plot(time, model_runoff, 'g-', label="Model Prediction", linewidth=2)
ax2.tick_params(axis='y', labelcolor='darkred')

# --- Title and Grid ---
fig.suptitle("Excess Rainfall vs Runoff and Model Prediction", fontsize=14)
fig.tight_layout()
ax1.grid(True)

# --- Combined Legend ---
lines = [rain_line, runoff_line, model_line]
labels = [line.get_label() for line in lines]
ax1.legend(lines, labels, loc='upper right')

plt.show()
```



Excess Rainfall vs Runoff and Model Prediction
Excess Rain
Observed Runoff
Model Prediction
Excess Rain (in)
Runoff (cfs)
Time Step
0
1
2
3
4
5
6
20
40
60
80
100
120
140
160
8
10


```
[ ]:
```

5. An agricultural watershed was urbanized over a 20 year interval. A triangular one-hour unit hydrograph was developed for this watershed for an excess rainfall duration of one hour.

   Before urbanization, the average loss rate was 0.30 in/hr.

   Figure 1 is the unit hydrograph that has a peak discharge of 400 cfs/in occurring at 3 hours, and a base time of 9 hours.

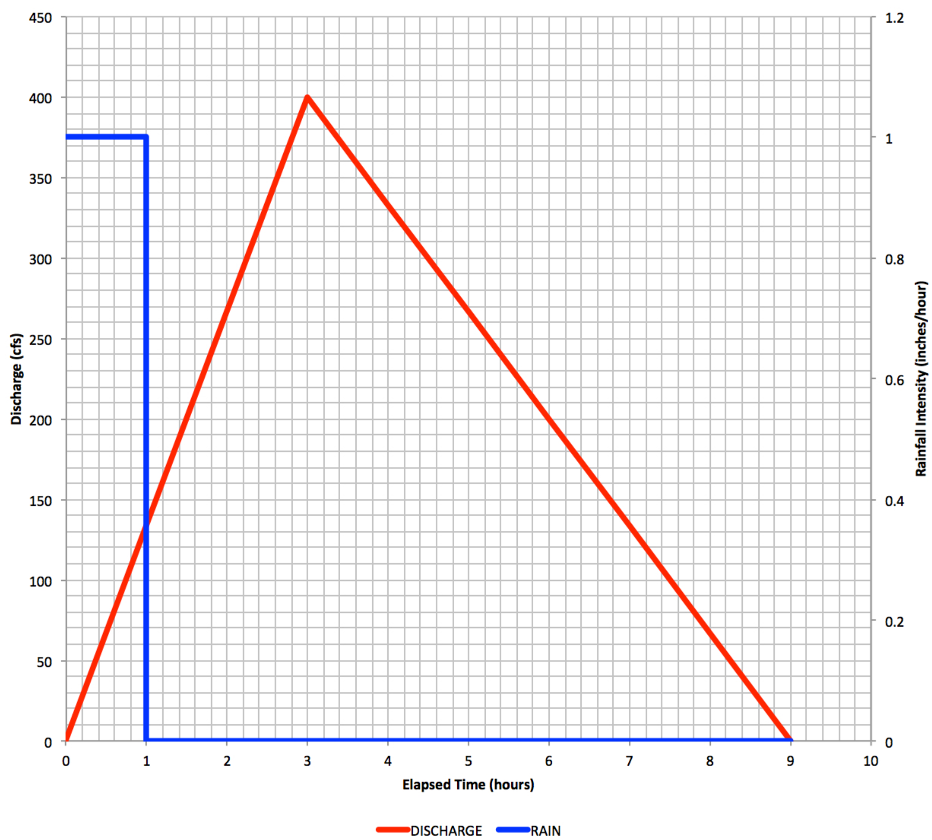


Figure 1: Pre-Urbanization unit hydrograph for excess rainfall of 1 in/hr for 1 hour.

After urbanization the loss rate was reduced to 0.15 in/hr and the peak discharge of the unit hydrograph increased to 600 cfs/in occurring at 1 hour, and the base time reduced to 6 hours. Figure 2 is the unit hydrograph with a peak discharge of 600 cfs occurring at 1 hours, and a time base of 6 hours.

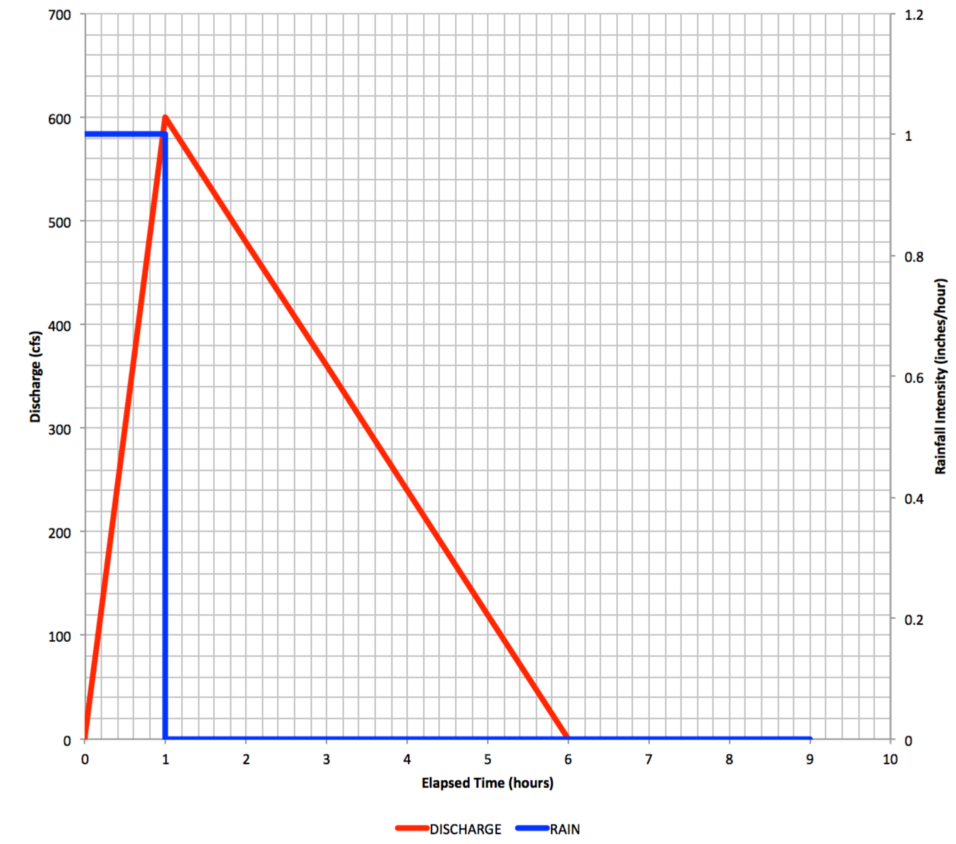


Figure 2: Post-Urbanization unit hydrograph for excess rainfall of 1 in/hr for 1 hour.

For a two hour storm in which 1 inch of rain fell in the first hour and 0.5 inch in the second hour, determine the direct runoff hydrographs before and after urbanization.[4]

[4]This exercise is the same as problem 7.5.7, pg. 238 in Chow, Maidment, Mays

Solution

a) Using Figure 1 as a template, the two increments of rainfall are directly plotted onto the template, then loss is applied to each increment. The resulting unitgraphs from each increment are plotted (in magenta/purple) and then ordinate-by-ordinate addition is used to construct the composite direct runoff hydrograph, as depicted in Figure 3

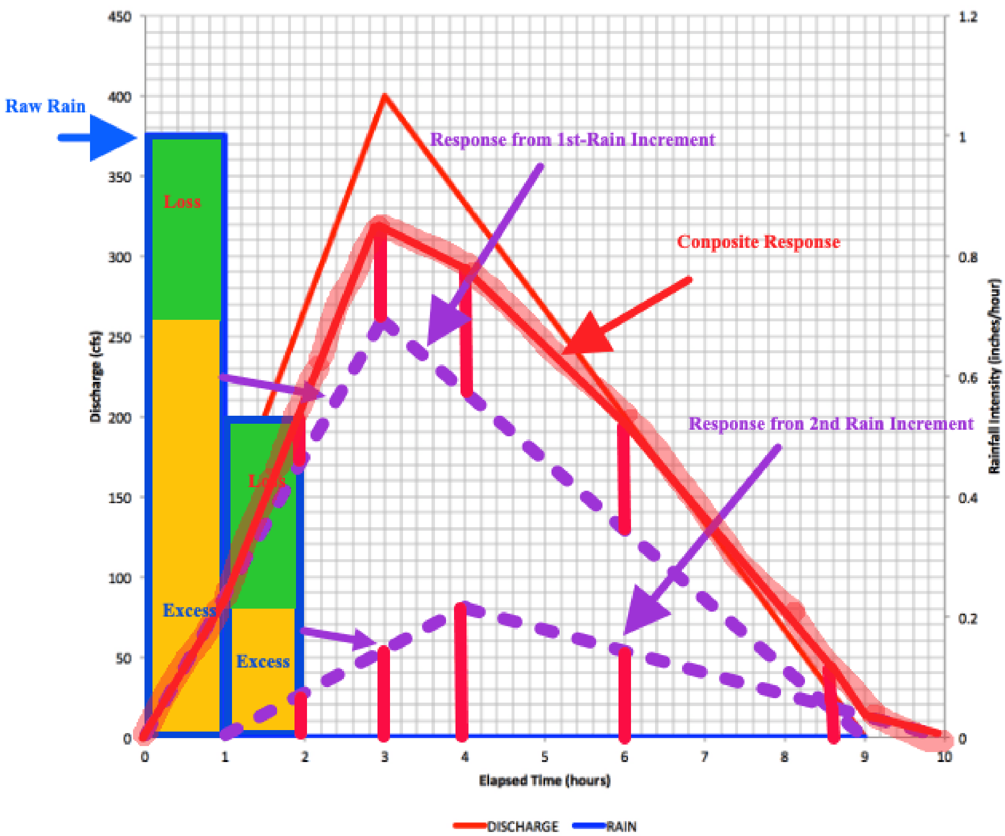


Figure 3: Pre-Urbanization unit hydrograph for excess rainfall of 1 in/hr for 1 hour.

b) Using Figure 2 as a template, the two increments of rainfall are directly plotted onto the template, then loss is applied to each increment. The resulting unitgraphs from each increment are plotted (in magenta/purple) and thenordinate-by-ordinate addition is used to construct the composite direct runoff hydrograph, as depicted in Figure 4

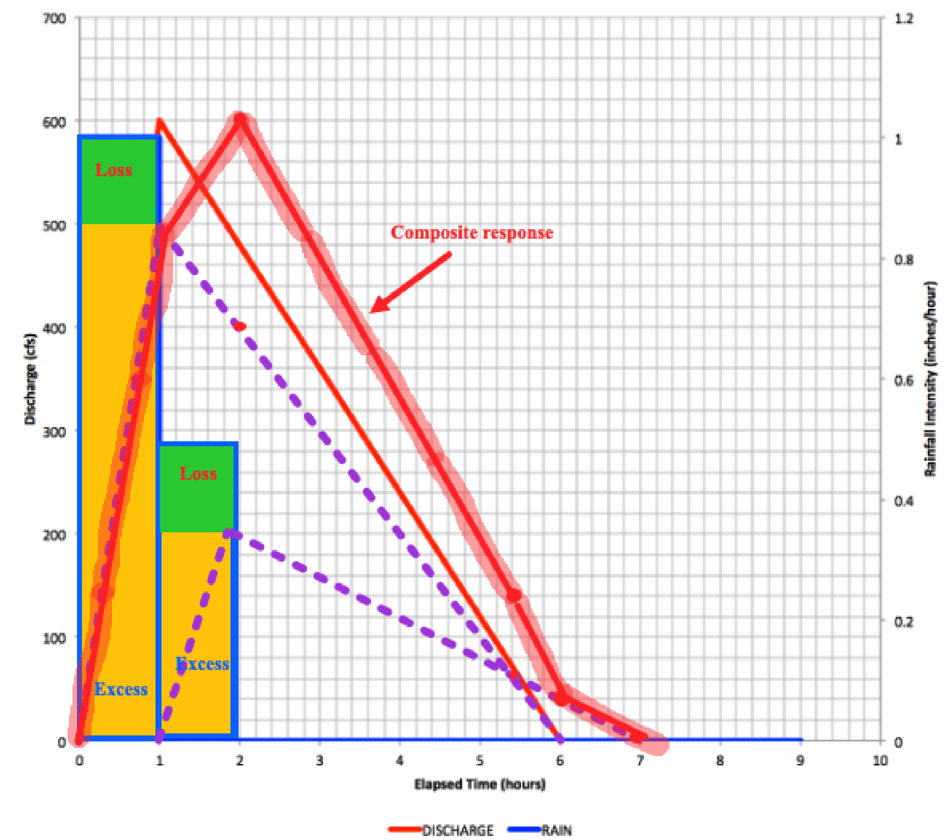
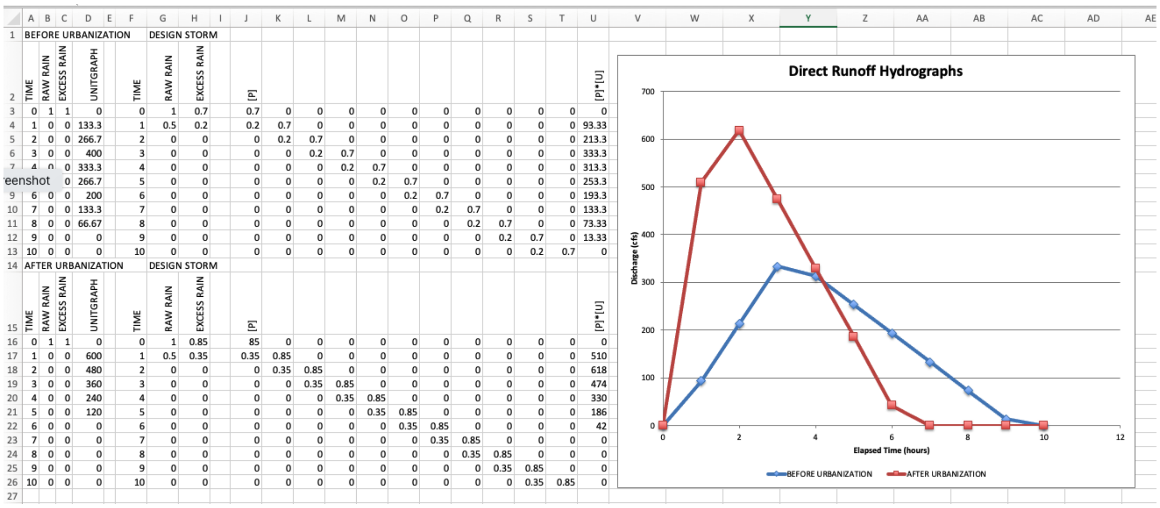


Figure 4: Post-Urbanization unit hydrograph for excess rainfall of 1 in/hr for 1 hour.

c) Alternatively (and more usefully) the responses can be incorporated into a spreadsheet as depicted in Figure 5.



	A	B	C	D	F	G	H	J	K	L	M	N	O	P	Q	R	S	T	U
1	BEFORE URBANIZATION					DESIGN STORM													
2	TIME	RAW RAIN	EXCESS RAIN	UNITGRAPH	TIME	RAW RAIN	EXCESS RAIN	[P]											[P]*[U]
3	0	1	1	0	0	1	0.7	0.7	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	133.3	1	0.5	0.2	0.2	0.7	0	0	0	0	0	0	0	0	0	93.33
5	2	0	0	266.7	2	0	0	0	0.2	0.7	0	0	0	0	0	0	0	0	213.3
6	3	0	0	400	3	0	0	0	0	0.2	0.7	0	0	0	0	0	0	0	333.3
7	4	0	0	333.3	4	0	0	0	0	0	0.2	0.7	0	0	0	0	0	0	313.3
eenshot			0	266.7	5	0	0	0	0	0	0	0.2	0.7	0	0	0	0	0	253.3
9	6	0	0	200	6	0	0	0	0	0	0	0	0.2	0.7	0	0	0	0	193.3
10	7	0	0	133.3	7	0	0	0	0	0	0	0	0	0.2	0.7	0	0	0	133.3
11	8	0	0	66.67	8	0	0	0	0	0	0	0	0	0	0.2	0.7	0	0	73.33
12	9	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0.2	0.7	0	13.33
13	10	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0.2	0.7	0
14	AFTER URBANIZATION					DESIGN STORM													
15	TIME	RAW RAIN	EXCESS RAIN	UNITGRAPH	TIME	RAW RAIN	EXCESS RAIN	[P]											[P]*[U]
16	0	1	1	0	0	1	0.85	85	0	0	0	0	0	0	0	0	0	0	0
17	1	0	0	600	1	0.5	0.35	0.35	0.85	0	0	0	0	0	0	0	0	0	510
18	2	0	0	480	2	0	0	0	0.35	0.85	0	0	0	0	0	0	0	0	618
19	3	0	0	360	3	0	0	0	0	0.35	0.85	0	0	0	0	0	0	0	474
20	4	0	0	240	4	0	0	0	0	0	0.35	0.85	0	0	0	0	0	0	330
21	5	0	0	120	5	0	0	0	0	0	0	0.35	0.85	0	0	0	0	0	186
22	6	0	0	0	6	0	0	0	0	0	0	0	0.35	0.85	0	0	0	0	42
23	7	0	0	0	7	0	0	0	0	0	0	0	0	0.35	0.85	0	0	0	0
24	8	0	0	0	8	0	0	0	0	0	0	0	0	0	0.35	0.85	0	0	0
25	9	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0.35	0.85	0	0
26	10	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0.35	0.85	0
27																			

Figure 5: Spreadsheet solution

The working spreadsheet is located at `http://54.243.252.9/ce-3354-webroot/2-Exercises/ES-6/ES6-SourceCode/ES6-Solution.xlsx`. Figure 5 is captures from Tab Sheet "P1."

6. A storm on April 16, 1977, on the Shoal Creek watershed at Northwest Park in Austin, Texas, resulted in the rainfall-runoff values in Figure 6.

   Use the linear regression method to determine the half-hour unit hydrograph for the watershed. The watershed drainage area is 7.03 mi^2. Assume that a uniform loss rate (constant loss model) is valid.[5]

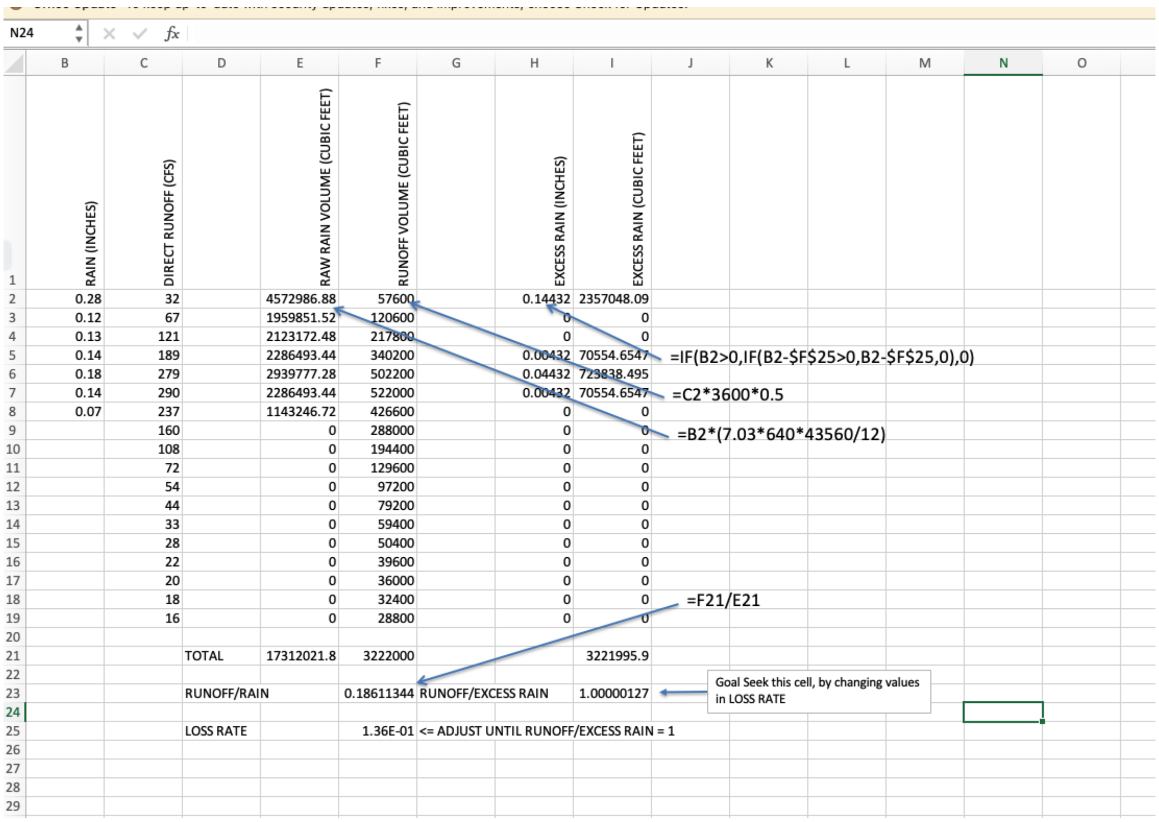
TIME (HRS)	RAIN (IN)	DIRECT RUNOFF (CFS)
0.5	0.28	32.0
1.0	0.12	67.0
1.5	0.13	121.0
2.0	0.14	189.0
2.5	0.18	279.0
3.0	0.14	290.0
3.5	0.07	237.0
4.0		160.0
4.5		108.0
5.0		72.0
5.5		54.0
6.0		44.0
6.5		33.0
7.0		28.0
7.5		22.0
8.0		20.0
8.5		18.0
9.0		16.0

Figure 6: Observed storm rainfall incremental depths and observed direct runoff hydrograph

[5]This exercise is a hybrid of problems 7.6.2 and 7.6.5, pg 239 in Chow, Maidment, and Mays.

Solution

a) Using data from Figure 6 and the supplied watershed area, the rainfall is converted into watershed input volume in same units as watershed runoff volume. A constant loss is applied to the rainfall increments until the ratio of output to input is unity. as a template, the two increments of rainfall are directly plotted onto the template, then loss is applied to each increment. Predict-and-correct or Goal Seek is sufficient as depicted in Figure 7



	B	C	D	E	F	G	H	I
1	RAIN (INCHES)	DIRECT RUNOFF (CFS)		RAW RAIN VOLUME (CUBIC FEET)	RUNOFF VOLUME (CUBIC FEET)		EXCESS RAIN (INCHES)	EXCESS RAIN (CUBIC FEET)
2	0.28	32		4572986.88	57600		0.14432	2357048.09
3	0.12	67		1959851.52	120600		0	0
4	0.13	121		2123172.48	217800		0	0
5	0.14	189		2286493.44	340200		0.00432	70554.6547
6	0.18	279		2939777.28	502200		0.04432	723838.495
7	0.14	290		2286493.44	522000		0.00432	70554.6547
8	0.07	237		1143246.72	426600		0	0
9		160		0	288000		0	0
10		108		0	194400		0	0
11		72		0	129600		0	0
12		54		0	97200		0	0
13		44		0	79200		0	0
14		33		0	59400		0	0
15		28		0	50400		0	0
16		22		0	39600		0	0
17		20		0	36000		0	0
18		18		0	32400		0	0
19		16		0	28800		0	0
20								
21			TOTAL	17312021.8	3222000			3221995.9
22								
23			RUNOFF/RAIN		0.18611344	RUNOFF/EXCESS RAIN		1.00000127
24								
25			LOSS RATE		1.36E-01	<= ADJUST UNTIL RUNOFF/EXCESS RAIN = 1		

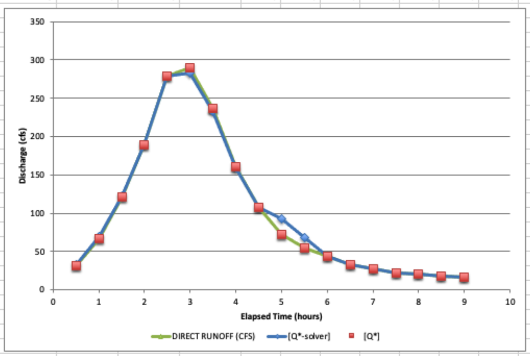
Figure 7: Volume Balance for Shoal Creek data to infer constant loss rate

b) Use the Matrix-Vector representation and ordinary-least-squares to construct a unit hydrograph for the watershed as depicted in Figure 8.[6]

N37

=MMULT(MINVERSE(MMULT(TRANSPOSE(E2:V19),E2:V19)),MMULT(TRANSPOSE(E2:V19),C2:C19))

	TIME (HRS)	EXCESS RAIN (INCHES)	DIRECT RUNOFF (CFS)	[P]																		[U]	[Q*]
2	0.5	0.1443364	32	0.144	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	221.7	32
3	1	0	67	0	0.144	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	464.2	67
4	1.5	0	121	0	0	0.144	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	838.3	121
5	2	0.0043364	189	0.004	0	0	0.144	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1302.8	189
6	2.5	0.0443364	279	0.044	0.004	0	0	0.144	0	0	0	0	0	0	0	0	0	0	0	0	0	1850.9	279
7	3	0.0043364	290	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	0	0	0	0	0	0	0	1834.8	290
8	3.5	0	237	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	0	0	0	0	0	0	1331.4	237
9	4	0	160	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	0	0	0	0	0	627.5	160
10	4.5	0	108	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	0	0	0	0	85.4	108
11	5	0	72	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	0	0	0	-160.4	72
12	5.5	0	54	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	0	0	-108.8	54
13	6	0	44	0	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	0	69.5	44
14	6.5	0	33	0	0	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	0	188.4	33
15	7	0	28	0	0	0	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	0	244.0	28
16	7.5	0	22	0	0	0	0	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	0	188.6	22
17	8	0	20	0	0	0	0	0	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	0	114.8	20
18	8.5	0	18	0	0	0	0	0	0	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	0	57.4	18
19	9	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0.004	0.044	0.004	0	0	0.144	24.6	16



Negative UH weights -- try Solver

	[U*]	[Q*-solv	[SE]
	224.3	32.38	0.146
Solver:	479.7	69.24	5.018
1) Compute SE from ([P][U*]-[Q_obs])^2	850.1	122.7	2.915
2) Compute SSE by sum(above)	1305	189.3	0.075
3) Use solver to min SSE, by changing [U*]	1845	278.4	0.37
3.A) Force NON-Negative [U*]	1783	283.2	45.67
	1294	232.1	23.58
	623.9	159.6	0.172
	88.63	108	3E-12
	0	92.65	426.5
	0	67.79	190.2
	71.66	44	2E-09
	182.7	33	1E-08
	191.3	28	9E-10
	150.3	22	6E-11
	111.1	20	5E-09
	60.7	18	6E-09
	42.08	16	3E-09
		[SSE] =>	694.6

Figure 8: Spreadsheet solution

The working spreadsheet is located at `http://54.243.252.9/ce-3354-webroot/2-Exercises/ES-6/ES6-SourceCode/ES6-Solution.xlsx`. Figure 5 is captured from Tab Sheet "P2-UH-CL-LOSS-FIT"

[6] This watershed will yield two negative ordinates; a SOLVER result is also shown. Recall negative ordinates are not physically relevant, and require addressing. They may simply be artifacts, or (more likely) sample aliasing.

7. Table 2 is a 15-minute unit hydrograph for Somewhere Else USA. Table 3 is a precipitation input time-series for the watershed

Table 2: Somewhere Else USA Unit Hydrograph Tabulation

Time (hours)	Runoff (CFS)
0.00	0
0.25	70
0.50	182
0.75	137
1.00	68
1.25	33
1.50	16
1.75	9
2.00	5
2.25	2
2.50	1
2.75	0.0

Table 3: Somewhere Else USA Excess Rain Input

Time (hours)	Rainfall Excess (inches)
0.00	0
0.25	0.50
0.50	1.25
0.75	0.75

Determine:

a) The design (direct runoff hydrograph) for the excess rainfall input time series.

b) A plot of the design hydrograph.

Solution(s): Entire problem as Jupyter Notebook (ENGR-1330) on following pages.

es5-ws7

August 4, 2025

Simply perform the matrix multiplication ($Q = PU$).

The screen capture below uses an Excel worksheet. Extends the data range with zero post-padding.

O8 | f_x 0

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1					P (matrix)																						
2	Time(hours)	UH_weight	Excess Rain		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	Q=PU
3	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0.25	70	0.5		0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0.5	182	1.25		1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35
6	0.75	137	0.75		0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	178.5
7	1	68	0		0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	348.5
8	1.25	33	0		0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	341.75
9	1.5	16	0		0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	204.25
10	1.75	9	0		0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100.25
11	2	5	0		0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	49.25
12	2.25	2	0		0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	25.75
13	2.5	1	0		0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	0	14
14	2.75	0	0		0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	0	6.75
15	3	0	0		0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0	2.75
16	3.25	0	0		0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0	0.75
17	3.5	0	0		0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0	0
18	3.75	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0	0
19	4	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0	0
20	4.25	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0	0
21	4.5	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0	0
22	4.75	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0	0
23	5	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0	0
24	5.25	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.75	1.25	0.5	0	0
25																											

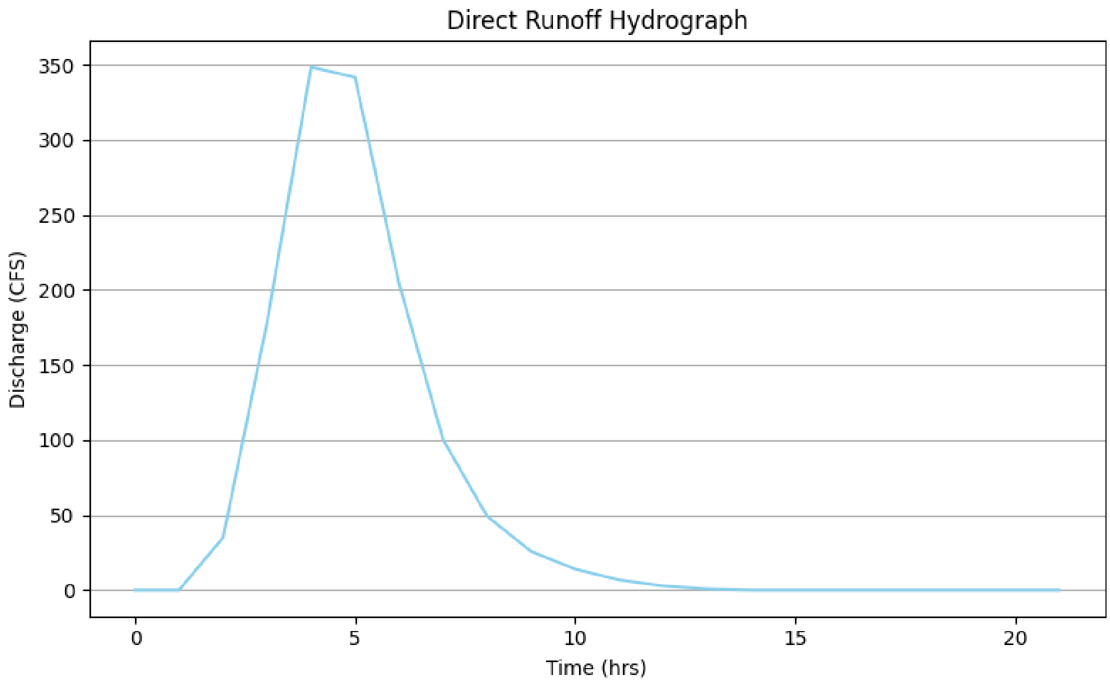
Then plot the results

[9]:
```python
import matplotlib.pyplot as plt

discharge = [0,0,35,178.5,348.5,341.75,204.25,100.25,49.25,25.75,14,6.75,2.75,0.
↪75,0,0,0,0,0,0,0,0]
time = list(range(len(discharge)))

plt.figure(figsize=(8, 5))
plt.plot(time, discharge, color='skyblue')

plt.title("Direct Runoff Hydrograph")
plt.xlabel("Time (hrs)")
plt.ylabel("Discharge (CFS)")
plt.grid(True, axis='y')
plt.tight_layout()
plt.show()
```



Direct Runoff Hydrograph
Discharge (CFS)
0
50
100
150
200
250
300
350
Time (hrs)
0
5
10
15
20

[ ]: