**UNIVERSITY OF ENGINEERING AND**

**TECHNOLOGY LAHORE**



**Assignment # 4**

**Dynamic Programming solution to Hydrothermal scheduling**

**Course Title: Advanced Power System Operation and Control**

**Course Code: EE 641**

**Submitted to:**

**Dr. Muhammad Asghar Saqib**

**Assistant Professor**

**Submitted by:**

**Muhammad Shamaas**

**ID # 2018-MS-EE-4**

**Date of Submission: 30 July 2020**

**DEPARTMENT OF ELECTRICAL ENGINEERING**

**Problem Statement**

A Hydro plant must be operated in conjunction with a Steam plant to serve a time varying load for a 24-hour period. The day is divided into six individual periods of 4h each. The load demand for the different periods is given below:

|  |  |  |
| --- | --- | --- |
| Period j | Time Interval | (MW) |
| 1 | 00:00-04:00 | 600 |
| 2 | 04:00-08:00 | 1000 |
| 3 | 08:00:12:00 | 900 |
| 4 | 12:00-16:00 | 500 |
| 5 | 16:00-20:00 | 400 |
| 6 | 20:00-24:00 | 300 |

Hydroelectric Plant

The minimum and maximum storage limits for the Hydro plant reservoir are given by:

The storage volume at the start and end of the day must be 10000 acre.ft.

The water use rate of the hydroelectric plant is given by:

Where is the generated hydroelectric power and q is the water discharge rate in acre.ft/h.

There is no spillage and the natural inflow rate is 1000 acre.ft/h.

Steam Plant

The steam plant production cost function is:

The marginal cost function is:

The Dynamic programming algorithm tries to schedule the two power plants optimally to meet the load demand and operational constraints for the six 4-hour time periods. It is also desired to minimize the production cost for the compound generation system.

**First Period**

The results for the first period are shown in the table below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | j=1 |  | |  |
|  | (acre.ft/h) | (MW) | (MW) |  |
| 14,000 | 0 | 0 | 00 | 15,040 |
| 12,000 | 500 | 24 | 57 | 14,523 |
| 10,000 | 1000 | 74 | 52 | 13,453 |
| 8,000 | 1500 | 124 | 47 | 12,392 |
| ,000 | 2000 | 174 | 42 | 11,342 |

The diagram shows the initial trajectories for the dynamic programming algorithm.



**Second Period**

The results for the second period are shown in the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| j=2 | | |  | |  |
|  |  | (acre.ft/h) | (MW) | (MW) |  |
| 18,000 | 14,000 | 0 | 0 | 00 | 39,040 |
|  |  |  |  |  |  |
| 1,000 | 14,000 | 500 | 24 | 97 | 38,484 |
| 1,000 | 12,000 | 0 | 0 | 1000 | 38,523 |
|  |  |  |  |  |  |
| 14,000 | 14,000 | 1000 | 74 | 92 | 37,334 |
| 14,000 |  | 500 | 24 | 97 | 37,967 |
| 14,000 | 10,000 | 0 | 0 | 1000 | 37,453 |
|  |  |  |  |  |  |
| 12,000 | 14,000 | 1500 | 124 | 876 | 36,194 |
| 12,000 | 12,000 | 1000 | 74 | 926 | 39,818 |
| 12,000 | 10,000 | 500 | 24 | 976 | 36,897 |
| 12,000 | 8,000 | 0 | 0 | 1000 | 36,392 |
|  |  |  |  |  |  |
| 10,000 | 14,000 | 2000 | 174 | 826 | 35,064 |
| 10,000 | 12,000 | 1500 | 124 | 876 | 35,677 |
| 10,000 | 10,000 | 1000 | 74 | 926 | 35,747 |
| 10,000 | 8,000 | 500 | 24 | 976 | 35,837 |
| 10,000 | 6,000 | 0 | 0 | 1000 | 35,342 |
|  |  |  |  |  |  |
| 8,000 | 12,000 | 2000 | 174 | 826 | 34,547 |
| 8,000 | 10,000 | 1500 | 124 | 876 | 34,607 |
| 8,000 | 8,000 | 1000 | 74 | 926 | 34,687 |
| 8,000 | 6,000 | 500 | 24 | 976 | 34,787 |
|  |  |  |  |  |  |
| 6,000 | 10,000 | 2000 | 174 | 826 | 33,476 |
| 6,000 | 8,000 | 1500 | 124 | 876 | 33,546 |
| 6,000 | 6,000 | 1000 | 74 | 926 | 33,636 |
|  |  |  |  |  |  |

The diagram below shows the trajectories for the second period of dynamic programming algorithm.



**Third Period**

The results for the third period are shown in the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| j=3 | | |  | |  |
|  |  | (acre.ft/h) | (MW) | (MW) |  |
| 18,000 | 18,000 | 1000 | 74 | 826 | 59,064 |
| 18,000 | 16,000 | 500 | 24 | 876 | 59,638 |
| 18,000 | 14,000 | 0 | 0 | 900 | 59,034 |
|  |  |  |  |  |  |
| 1,000 | 18,000 | 1500 | 124 | 776 | 57,944 |
| 1,000 | 16,000 | 1000 | 74 | 826 | 58,508 |
| 16,000 | 14,000 | 500 | 24 | 876 | 58,488 |
| 16,000 | 12,000 | 0 | 0 | 900 | 57,894 |
|  |  |  |  |  |  |
| 14,000 | 18,000 | 2000 | 174 | 726 | 56,833 |
| 14,000 | 16,000 | 1500 | 124 | 776 | 57,388 |
| 14,000 | 14,000 | 1000 | 74 | 826 | 57,358 |
| 14,000 | 12,000 | 500 | 24 | 876 | 57,348 |
| 14,000 | 10,000 | 0 | 0 | 900 | 56,764 |
|  |  |  |  |  |  |
| 12,000 | 16,000 | 2000 | 174 | 726 | 56,278 |
| 12,000 | 14,000 | 1500 | 124 | 776 | 56,238 |
| 12,000 | 12,000 | 1000 | 74 | 826 | 56,218 |
| 12,000 | 10,000 | 500 | 24 | 876 | 56,218 |
| 12,000 | 8,000 | 0 | 0 | 900 | 56,247 |
|  |  |  |  |  |  |
| 10,000 | 14,000 | 2000 | 174 | 726 | 55,128 |
| 10,000 | 12,000 | 1500 | 124 | 776 | 55,098 |
| 10,000 | 10,000 | 1000 | 74 | 826 | 55,088 |
| 10,000 | 8,000 | 500 | 24 | 876 | 55,700 |
| 10,000 | 6,000 | 0 | 0 | 900 | 55,176 |
|  |  |  |  |  |  |
| 8,000 | 12,000 | 2000 | 174 | 726 | 53,987 |
| 8,000 | 10,000 | 1500 | 124 | 776 | 53,967 |
| 8,000 | 8,000 | 1000 | 74 | 826 | 54,570 |
| 8,000 | 6,000 | 500 | 24 | 876 | 54,630 |
|  |  |  |  |  |  |
| 6,000 | 10,000 | 2000 | 174 | 726 | 52,857 |
| 6,000 | 8,000 | 1500 | 124 | 776 | 53,450 |
| 6,000 | 6,000 | 1000 | 74 | 826 | 53,500 |
|  |  |  |  |  |  |

The diagram below shows the trajectories for the third period of dynamic programming algorithm.



**Fourth Period**

The results for the fourth period are shown in the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| j=4 | | |  | |  |
|  |  | (acre.ft/h) | (MW) | (MW) |  |
| 18,000 | 18,000 | 1000 | 74 | 426 | 70,376 |
| 18,000 | 16,000 | 500 | 24 | 476 | 70,286 |
| 18,000 | 14,000 | 0 | 0 | 500 | 69,664 |
|  |  |  |  |  |  |
| 1,000 | 18,000 | 1500 | 124 | 376 | 69,336 |
| 1,000 | 16,000 | 1000 | 74 | 426 | 69,236 |
| 16,000 | 14,000 | 500 | 24 | 476 | 69,156 |
| 16,000 | 12,000 | 0 | 0 | 500 | 69,118 |
|  |  |  |  |  |  |
| 14,000 | 18,000 | 2000 | 174 | 326 | 68,306 |
| 14,000 | 16,000 | 1500 | 124 | 376 | 68,196 |
| 14,000 | 14,000 | 1000 | 74 | 426 | 68,106 |
| 14,000 | 12,000 | 500 | 24 | 476 | 68,610 |
| 14,000 | 10,000 | 0 | 0 | 500 | 67,988 |
|  |  |  |  |  |  |
| 12,000 | 16,000 | 2000 | 174 | 326 | 67,166 |
| 12,000 | 14,000 | 1500 | 124 | 376 | 67,066 |
| 12,000 | 12,000 | 1000 | 74 | 426 | 67,560 |
| 12,000 | 10,000 | 500 | 24 | 476 | 67,480 |
| 12,000 | 8,000 | 0 | 0 | 500 | 66,867 |
|  |  |  |  |  |  |
| 10,000 | 14,000 | 2000 | 174 | 326 | 66,036 |
| 10,000 | 12,000 | 1500 | 124 | 376 | 66,520 |
| 10,000 | 10,000 | 1000 | 74 | 426 | 66,430 |
| 10,000 | 8,000 | 500 | 24 | 476 | 66,360 |
| 10,000 | 6,000 | 0 | 0 | 500 | 65,757 |
|  |  |  |  |  |  |
| 8,000 | 12,000 | 2000 | 174 | 326 | 65,489 |
| 8,000 | 10,000 | 1500 | 124 | 376 | 65,389 |
| 8,000 | 8,000 | 1000 | 74 | 426 | 65,309 |
| 8,000 | 6,000 | 500 | 24 | 476 | 65,249 |
|  |  |  |  |  |  |
| 6,000 | 10,000 | 2000 | 174 | 326 | 64,359 |
| 6,000 | 8,000 | 1500 | 124 | 376 | 64,269 |
| 6,000 | 6,000 | 1000 | 74 | 426 | 64,199 |
|  |  |  |  |  |  |

The diagram below shows the trajectories for the fourth period of dynamic programming algorithm.



**Fifth Period**

The results for the fifth period are shown in the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| j=5 | | |  | |  |
|  |  | (acre.ft/h) | (MW) | (MW) |  |
| 18,000 | 18,000 | 1000 | 74 | 326 | 78,936 |
| 18,000 | 16,000 | 500 | 24 | 376 | 79,420 |
| 18,000 | 14,000 | 0 | 0 | 400 | 78,788 |
|  |  |  |  |  |  |
| 1,000 | 18,000 | 1500 | 124 | 276 | 77,915 |
| 1,000 | 16,000 | 1000 | 74 | 326 | 78,389 |
| 16,000 | 14,000 | 500 | 24 | 376 | 78,289 |
| 16,000 | 12,000 | 0 | 0 | 400 | 77,667 |
|  |  |  |  |  |  |
| 14,000 | 18,000 | 2000 | 174 | 226 | 76,905 |
| 14,000 | 16,000 | 1500 | 124 | 276 | 77,369 |
| 14,000 | 14,000 | 1000 | 74 | 326 | 77,259 |
| 14,000 | 12,000 | 500 | 24 | 376 | 77,169 |
| 14,000 | 10,000 | 0 | 0 | 400 | 76,557 |
|  |  |  |  |  |  |
| 12,000 | 16,000 | 2000 | 174 | 226 | 76,359 |
| 12,000 | 14,000 | 1500 | 124 | 276 | 76,239 |
| 12,000 | 12,000 | 1000 | 74 | 326 | 76,139 |
| 12,000 | 10,000 | 500 | 24 | 376 | 76,059 |
| 12,000 | 8,000 | 0 | 0 | 400 | 76,049 |
|  |  |  |  |  |  |
| 10,000 | 14,000 | 2000 | 174 | 226 | 75,229 |
| 10,000 | 12,000 | 1500 | 124 | 276 | 75,119 |
| 10,000 | 10,000 | 1000 | 74 | 326 | 75,029 |
| 10,000 | 8,000 | 500 | 24 | 376 | 75,551 |
| 10,000 | 6,000 | 0 | 0 | 400 | 74,999 |
|  |  |  |  |  |  |
| 8,000 | 12,000 | 2000 | 174 | 226 | 74,109 |
| 8,000 | 10,000 | 1500 | 124 | 276 | 74,009 |
| 8,000 | 8,000 | 1000 | 74 | 326 | 74,521 |
| 8,000 | 6,000 | 500 | 24 | 376 | 74,501 |
|  |  |  |  |  |  |
| 6,000 | 10,000 | 2000 | 174 | 226 | 72,998 |
| 6,000 | 8,000 | 1500 | 124 | 276 | 73,501 |
| 6,000 | 6,000 | 1000 | 74 | 326 | 73,471 |
|  |  |  |  |  |  |

The diagram below shows the trajectories for the fifth period of dynamic programming algorithm.

****

**Sixth Period**

The results for the sixth period are shown in the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| j=6 | | |  | |  |
|  |  | (acre.ft/h) | (MW) | (MW) |  |
| 10,000 | 10,000 | 1000 | 74 | 226 | 82,241 |
| 10,000 | 8,000 | 500 | 24 | 276 | 82,260 |
| 10,000 | 6,000 | 0 | 0 | 300 | 81,738 |
|  |  |  |  |  |  |

The diagram below shows the trajectories for the sixth period of dynamic programming algorithm.



**Final Result**

The minimum cost trajectory is shown in the figure below.



The minimum production cost for the hydrothermal power plant is $ 81,738.

**MATLAB Code**

clc;clear all;

Pload=[0 600 1000 900 500 400 300];

Vs(1,1)=10000;

Vsmin=[10 6 6 6 6 6 10]\*1000;

Vsmax=[10 18 18 18 18 18 10]\*1000;

clength=[1 0 0 0 0 0 0];

TCactual=ones(3000,7)\*inf;

TCactual(1,1)=0;

for cindex=2:7

for rindex=1:clength(cindex-1)

for q=0:500:2260

Vsnew=Vs(rindex,cindex-1)+4\*1000-4\*q;

PHnew=((q-260)/10);

if (q==0)

PHnew=0;

end

PSnew=(Pload(cindex)-PHnew);

if ((Vsnew>=Vsmin(cindex))&&(Vsnew<=Vsmax(cindex))&&(mod(Vsnew,2000)==0)&&(PSnew>=200)&&(PSnew<=1200))

clength(cindex)=clength(cindex)+1;

Vs(clength(cindex),cindex)=Vsnew;

Vsold(clength(cindex),cindex)=Vs(rindex,cindex-1);

PH(clength(cindex),cindex)=((q-260)/10);

if (q==0)

PH(clength(cindex),cindex)=0;

end

PS(clength(cindex),cindex)=(Pload(cindex)-PH(clength(cindex),cindex));

TCactual(clength(cindex),cindex)=TCactual(rindex,cindex-1)+4\*(700+4.8\*PS(clength(cindex),cindex)+(PS(clength(cindex),cindex)\*PS(clength(cindex),cindex))/2000);

rindold(clength(cindex),cindex)=rindex;

end

end

end

end

TC=ones(7,18,18)\*inf;

for row=1:length(TCactual)

for col=2:7

if (TCactual(row,col)~=inf)

TC(col,Vsold(row,col)/1000,Vs(row,col)/1000)=min(TC(col,Vsold(row,col)/1000,Vs(row,col)/1000),TCactual(row,col));

end

end

end

hold on;

for r=1:length(Vs)

for c=2:7

line(([c-1 c]-1)\*4,[Vsold(r,c) Vs(r,c)]);

end

end

hold on

xlabel('Time (Hour)')

ylabel('Storage Volume (acre.ft)')

axis([0 24 4000 20000])

find=min(TCactual(:,7));

for row=1:length(TCactual)

if (TCactual(row,7)==find)

find=rindold(row,7);

minTCpath(1,7)=Vs(row,7);

end

end

for col=6:-1:1

minTCpath(1,col)=Vs(find,col);

find=rindold(find,col);

end

figure;

plot(0:4:24,minTCpath)

hold on

title('Trajectory for minimum production cost')

xlabel('Time (Hour)')

ylabel('Storage Volume (acre.ft)')

axis([0 24 4000 20000])