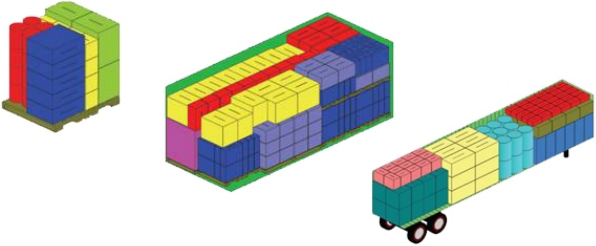
CE 191: Civil and Environmental Engineering Systems Analysis

LEC 15 : DP Examples

Professor Scott Moura
Civil & Environmental Engineering
University of California, Berkeley

Fall 2014

Ex 1: Knapsack Problem



- knapsack has finite volume, K
- can fill knapsack with integer number of items, x_i
- each item has per unit volume of v_i
- each item has per unit value of c_i

Goal: Select number of items x_i to place in knapsack to max total value.

DP Equations

Let $V(y)$ represent the maximal knapsack value if the remaining volume is y.

Consider one unit of item i:

- value added, c_i
- volume remaining, $y - v_i$
- maximal value of knapsack with remaining volume $V(y - v_i)$

Principle of Optimality & Boundary Condition:

$$\begin{aligned} V(y) &= \max_{v_i \leq y} \{c_i + V(y - v_i)\} \\ V(0) &= 0 \end{aligned}$$

From the Midterm

Chris McCandless is traveling into the wilderness. He can bring one knapsack including food and equipment. The knapsack has a finite volume. However, he wishes to maximize the total “value” of goods in the knapsack.

$$\begin{aligned} \max \quad & 2x_1 + x_2 \\ \text{s. to} \quad & 2x_1 + 3x_2 \leq 9 \\ & x_i \geq 0 \in \mathbb{Z} \end{aligned}$$

$$\begin{aligned} V(0) &= 0 \\ V(1) &= \max_{v_i \leq 1} \{c_i + V(1 - v_i)\} = 0 \\ V(2) &= \max_{v_i \leq 2} \{c_i + V(2 - v_i)\} \\ &= 2 + V(2-2) = 2 \\ V(3) &= \max_{v_i \leq 3} \{c_i + V(3 - v_i)\} \\ &= \max\{2 + V(2-2), 1 + V(3-3)\} = 2 \\ V(4) &= \max_{v_i \leq 4} \{c_i + V(4 - v_i)\} \\ &= \max\{2 + V(4-2), 1 + V(4-3)\} = \max\{2+2, 1+0\} = 4 \end{aligned}$$

From the Midterm cont.

$$
\begin{aligned}
V(5) &= \max_{v_i \leq 5} \{c_i + V(5 - v_i)\} \\
&= \max\{2 + V(5-2), 1 + V(5-3)\} = \max\{2+2, 1+2\} = 4 \\
V(6) &= \max_{v_i \leq 6} \{c_i + V(6 - v_i)\} \\
&= \max\{2 + V(6-2), 1 + V(6-3)\} = \max\{2+4, 1+2\} = 6 \\
V(7) &= \max_{v_i \leq 7} \{c_i + V(7 - v_i)\} \\
&= \max\{2 + V(7-2), 1 + V(7-3)\} = \max\{2+4, 1+4\} = 6 \\
V(8) &= \max_{v_i \leq 8} \{c_i + V(8 - v_i)\} \\
&= \max\{2 + V(8-2), 1 + V(8-3)\} = \max\{2+6, 1+4\} = 8 \\
V(9) &= \max_{v_i \leq 9} \{c_i + V(9 - v_i)\} \\
&= \max\{2 + V(9-2), 1 + V(9-3)\} = \max\{2+6, 1+6\} = 8 \\
V(9) &= 8
\end{aligned}
$$

Item 1 - food: 4 units

Item 2 - equipment: 0 units

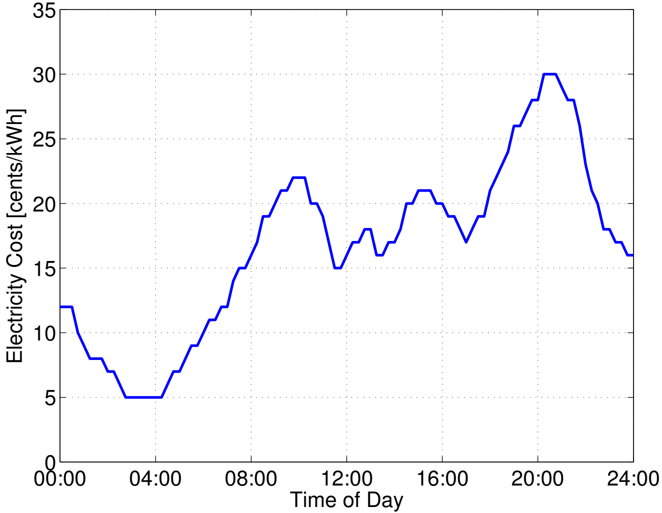
Ex 2: Smart Appliance Scheduling

- appliance (say, dishwasher) has five cycles (each 15 min)

cycle		power
1	prewash	1.5 kW
2	main wash	2.0 kW
3	rinse 1	0.5 kW
4	rinse 2	0.5 kW
5	dry	1.0 kW

- cycle must be run in order, possibly with idle periods in between
- electricity price varies (in 15 min periods)
- find cheapest cycle schedule starting at 17:00 and ending at 24:00

Electricity Price



Formulation

- k indexes 15 min periods; $k = 0$ is 17:00–17:15, $k = 28$ is 24:00–24:15
- $x_k \in \{0, \cdots, 5\}$ is the current cycle; $x_0 = 0$.
- $u_k \in \{0, 1\}$ corresponding to (wait, next cycle).
- state-transition function: $x_{k+1} = f(x_k, u_k) = x_k + u_k$
- cost-per-time-step: $c_k(x_k, u_k) = \frac{1}{4} c_k p_{x_{k+1}} u_k$
  - c_k is the electricity cost in cents/kWh in period k
  - p_i is power of cycle i
- terminal cost: $c_N(x_N) = 0$ for $x_N = 5$; $c_N(x_N) = \infty$ otherwise.

DP Equations

Let $V_k(x_k)$ represent min cost-to-go from time step k to N, given current state x_k.

Principle of Optimality:

$$\begin{aligned} V_k(x_k) &= \min_{u_k \in \{0,1\}} \left\{ \frac{1}{4} c_k \, p_{x_{k+1}} u_k + V_{k+1}(x_{k+1}) \right\} \\ &= \min_{u_k \in \{0,1\}} \left\{ \frac{1}{4} c_k \, p_{x_{k+1}} u_k + V_{k+1}(x_k + u_k) \right\} \\ &= \min \left\{ V_{k+1}(x_k), \;\; \frac{1}{4} c_k \, p_{x_{k+1}} + V_{k+1}(x_k + 1) \right\} \end{aligned}$$

with the boundary condition

$$V_N(5) = 0, \qquad V_N(i) = \infty \;\; \text{for} \;\; i \neq 5$$

Optimal Control Action:

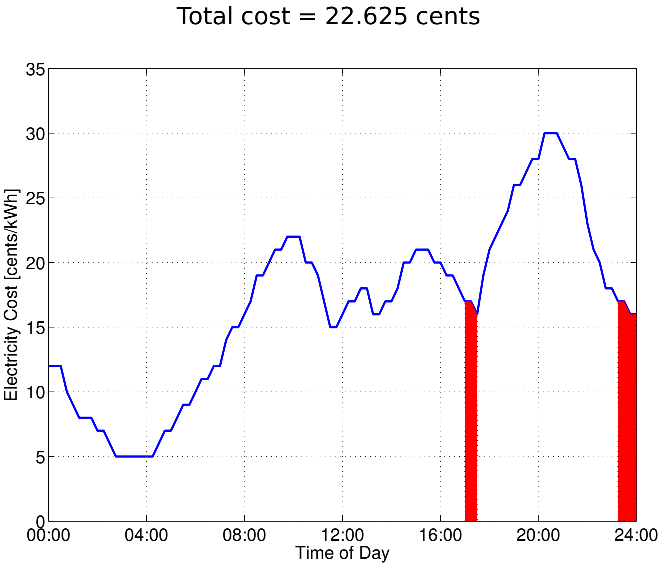
$$u_k^*(x_k) = \arg \min_{u_k \in \{0,1\}} \left\{ \frac{1}{4} c_k \, p_{x_{k+1}} u_k + V_{k+1}(x_{k+1}) \right\}$$

```
%% Problem Data
% Cycle power
p = [0; 1.5; 2.0; 0.5; 0.5; 1.0];

% Electricity Price Data
c = ...
    [12,12,12,10,9,8,8,8,7,7,6,5,5,5,5,5,5,5,6,7,7,8,9,9,10,11,11,...
     12,12,14,15,15,16,17,19,19,20,21,21,22,22,22,20,20,19,17,15,15,16
     17,17,18,18,16,16,17,17,18,20,20,21,21,21,20,20,19,19,18,17,17,..
     16,19,21,22,23,24,26,26,27,28,28,30,30,30,29,28,28,26,23,21,20,18

%% Solve DP Equations
% Time Horizon
N = 28;
% Number of states
nx = 6;

% Preallocate Value Function
V = inf*ones(N,nx);
% Preallocate control policy
u = nan*ones(N,nx);

% Boundary Condition
V(end,end) = 0;
```

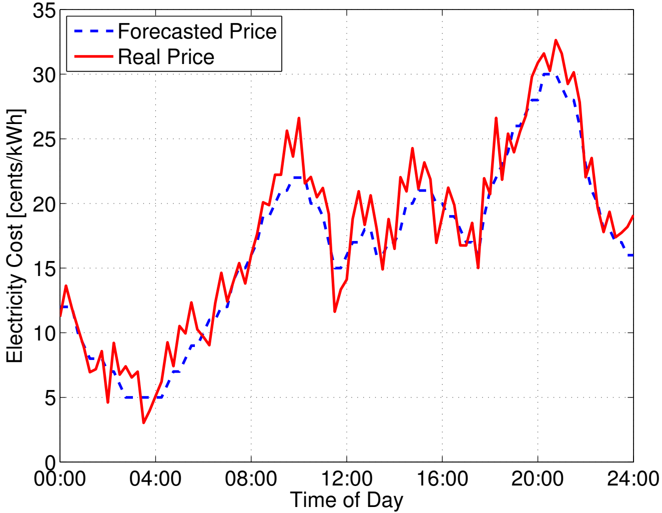
Matlab Implementation - 2

```
% Iterate through time backwards
for k = (N-1):-1:1;

    % Iterate through states
    for i = 1:nx

        % If you're in last state, can only wait
        if(i == nx)
            V(k,i) = V(k+1,i);

        % Otherwise, solve Principle of Optimality
        else
                                   %Choose  u=0 ; u=1
            [V(k,i),idx] = min([V(k+1,i); 0.25*c(69+k)*p(i+1) + ...
                V(k+1,i+1)]);

            % Save minimizing control action
            u(k,i) = idx-1;
        end
    end
end
```

Optimal Schedule



Ex 3: Smart Appliance Scheduling w/ Random Cost



Random Cost

$$\begin{aligned} \text{True cost} &= \text{forecasted cost} + \text{random perturbation} \\ &= c_k + w_k \end{aligned}$$

- Variable w_k is random
- Example probability distributions: uniform, normal, log-normal, Poisson, chi-squared, gamma, Pareto, non-parametric
- Suppose the most basic statistic is known: **the expected value**.
- Let $\overline{w}_k = E[w_k]$
- Random cost-per-time-step: $c_k(x_k, u_k, w_k) = \frac{1}{4}\left(c_k + w_k\right) p_{x_{k+1}} u_k$

Stochastic Optimization

$$
\begin{aligned}
\min \quad & J = E\left[\sum_{k=0}^{N-1} c(x_k, u_k, w_k) + c_N(x_N)\right] \\
\text{s. to} \quad & x_{k+1} = x_k + u_k \\
& x_0 = 0 \\
& u_k \in \{0, 1\}
\end{aligned}
$$

Stochastic Dynamic Programming (SDP)

Let $V_k(x_k)$ represent expected min cost-to-go from time step k to N, given current state x_k.

Principle of Optimality:

$$\begin{aligned}
V_k(x_k) &= \min_{u_k} E\left\{c(x_k, u_k, w_k) + V_{k+1}(x_{k+1})\right\} \\
&= \min_{u_k \in \{0,1\}} \left\{ E\left[\frac{1}{4}(c_k + w_k)\, p_{x_{k+1}} u_k\right] + V_{k+1}(x_{k+1}) \right\} \\
&= \min_{u_k \in \{0,1\}} \left\{ \frac{1}{4}(c_k + \overline{w}_k)\, p_{x_{k+1}} u_k + V_{k+1}(x_k + u_k) \right\} \\
&= \min \left\{ V_{k+1}(x_k),\ \ \frac{1}{4}(c_k + \overline{w}_k)\, p_{x_{k+1}} + V_{k+1}(x_k + 1) \right\}
\end{aligned}$$

with the boundary condition

$$V_N(5) = 0, \qquad V_N(i) = \infty \ \text{ for } \ i \neq 5$$

Optimal Control Action:

$$u_k^*(x_k) = \arg \min_{u_k \in \{0,1\}} E\left\{ \frac{1}{4}(c_k + w_k)\, p_{x_{k+1}} u_k + V_{k+1}(x_{k+1}) \right\}$$

SDP Summary

- Incorporate **uncertainty** as **random variables**
- Need to know **some statistics** about the random variables
- Minimize **expected** cost

Additional Reading

Revelle § 6.G, 13.A, 13.B

Denardo, Eric V. Dynamic programming: models and applications. DoverPublications.com, 1982.

Bertsekas, Dimitri P. Dynamic programming and optimal control. Vol. 1. No. 2. Belmont: Athena Scientific, 1995.