UNIVERSIDAD AUTONOMA de ENTRE RIOS

Facultad de Ciencia y Tecnología

Carrera: Licenciatura en Sistemas Informáticos
Cátedra: Investigación Operativa
Tema : Unidad 6 - Programación Dinámica Determinística
a) Guía de Teoría

PROGRAMACION DINAMICA DETERMINISTICA

Introducción

La programación Dinámica (PD) es una técnica matemática que a menudo resulta útil para tomar una sucesión de decisiones interrelacionadas. Proporciona un procedimiento sistemático para determinar la combinación de decisiones que maximice la efectividad global.

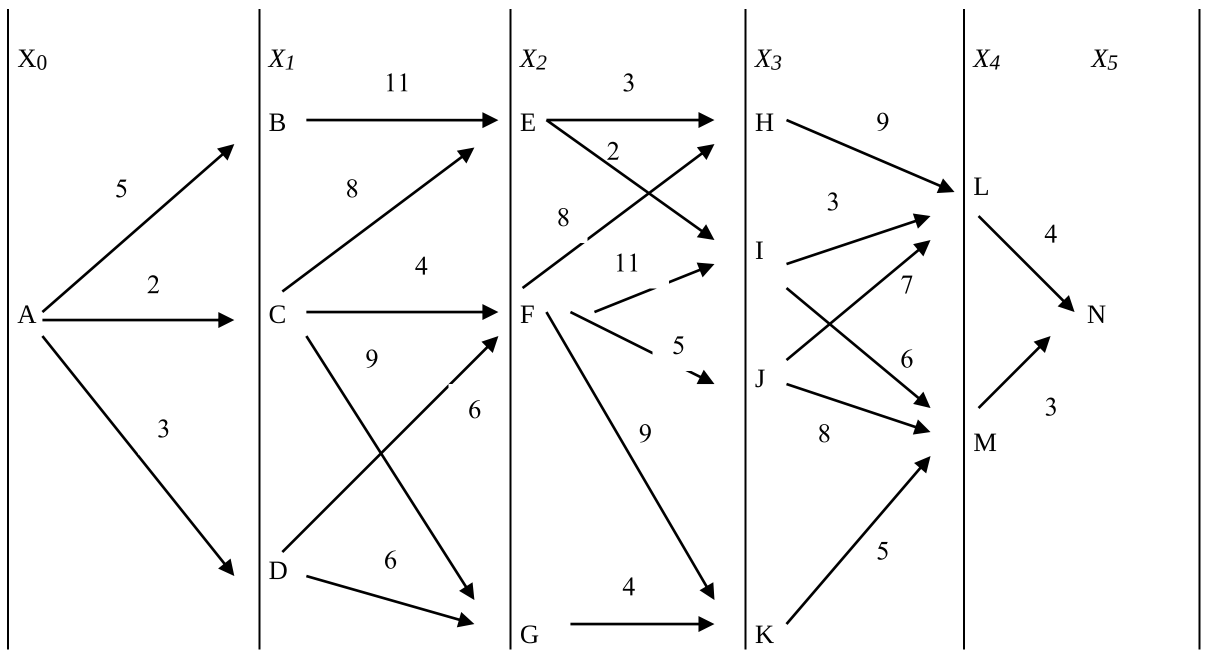
Contrastando con la programación lineal, no existe un planteamiento estándar del problema. Es un tipo general de enfoque para resolver problemas y las ecuaciones particulares usadas deben desarrollarse para cada situación individual.

Por lo tanto se requiere un cierto grado de ingenio y de visión de la estructura general de problemas de PD, a fin de reconocer cuando un problema se puede resolver mediante los procedimientos de ésta programación y como se haría.

Probablemente se puedan desarrollar mejor estas aptitudes por medio de una exposición de ejemplos y de un estudio de las características que son comunes a todas estas situaciones.

Ejemplo de optimización secuencial; construcción de una carretera; método de la red.

Para cada uno de los tramos se ha evaluado el costo de las diversas variantes; nos proponemos encontrar el camino de **costo mínimo** entre A y N.



Variables de decisión	**Posiciones posibles**
X_0	A
X_1	B, C, D
X_2	E, F, G
X_3	H, I, K, J
X_4	L, M
X_5	N

El gráfico podría representar una carretera a construir. Cada vertical es una etapa y cada punto una posible vía de paso de la carretera. Por ej. de B se puede ir solamente a E.

Cada punto representa un ***proceso de decisión***. Estas decisiones dependen de lo que se está buscando.

Una ***política*** sería un camino cualquiera de A N (Ej. ADFJLN).

Una ***subpolítica*** es una porción continua de camino (DFK), (CEHL).

Debemos llegar a que una política determinada sea la de mínimo costo. Por lo tanto será la ***política óptima.***

El costo total de la carretera es:

$$F(X_0; X_1; X_2; X_3; X_4; X_5) = V_1(X_0; X_1) + ...+ V_5(X_4; X_5).$$

Nótese primero que el procedimiento poco inteligente de seleccionar el recorrido más barato ofrecido en cada etapa sucesiva no necesariamente conduce a una decisión óptima global. Siguiendo esta estrategia, la carretera elegida sería ACFJLN, la que da un costo igual a (2+4+5+7+4 = 22) que no es el óptimo, como veremos más adelante.

Otro enfoque para resolver este problema es hacerlo por tanteos con todas las alternativas que se nos presentan. Sin embargo el número de éstas es bastante grande (24) para este sencillo problema.

Por fortuna la PD suministra una solución con mucho menos esfuerzo que la enumeración exhaustiva (los ahorros de cálculo serían enormes para versiones más grandes de este problema). La Programación Dinámica parte de una pequeña porción del problema y encuentra la solución óptima para este problema más pequeño. Entonces gradualmente agranda el problema, hallando la solución óptima en curso a partir de la anterior, hasta que se resuelve por completo el problema original.

Busquemos inicialmente el **costo mínimo del tramo I (T_I)** para c/u de los puntos terminales (B, C, D) y llamemos al mismo $f_I(X_1)$.

$f_I(B) = V_1(A;B) = 5$ costo para llegar a B, desde A

$f_I(C) = V_1(A;C) = 2$ costo para llegar a C, desde A

$f_I(D) = V_1(A;D) = 3$ costo para llegar a D, desde A

Llamemos $\mathbf{f_{I,II}(X_1;\ X_2)}$ **al costo mínimo para ambos tramos a la vez** y hallémoslo para los diferentes valores de la variable de decisión $\mathbf{X_2}$

$\mathbf{f_{I,II}(X_1;\ X_2 = E) = mín\ [f_I(X_1) + V_2(X_1;\ X_2)]}$ = [5+11; 2+8; 3+M] = $\mathbf{10\ (X_1 = C)}$

$\mathbf{X_1}$ = B, C, D $\qquad$ $\mathbf{X_1}$=B $\quad$ $\mathbf{X_1}$=C $\quad$ $\mathbf{X_1}$=D

$\mathbf{f_{I,II}(X_1;\ X_2 = F)}$ **=**

$\mathbf{f_{I,II}(X_1;\ X_2 = G)}$ **=**

Llamemos $\mathbf{f_{I,II,III}(X_1; X_2; X_3)}$ **al costo mínimo de los tramos I, II y III globalmente** y vemos que nos independizamos del tramo I, pero no se pierde su incidencia porque ya se consideró en el tramo II.

$\mathbf{f_{I,II,III}(X_1; X_2; X_3 = H) = \text{mín}\ [f_{I,II}(X_2) + V_3(X_2; X_3)]}$ = [10+3; 6+8] = **13 (X_2=E)**

X_2 = E, F $\quad$ X_2=E $\quad$ X_2=F

$\mathbf{f_{I,II,III}(X_1; X_2; X_3 = I)}$ **12 (X_2=E)**

$\mathbf{f_{I,II,III}(X_1; X_2; X_3 = J) =}$ **11 (X_2=F)**

$\mathbf{f_{I,II,III}(X_1; X_2; X_3 = K) =}$ **13 (X_2=G)**

Luego, el **tramo I,II,III,IV**.

$\mathbf{f_{I,II,III,IV}(X_1; X_2;X_3;X_4=L)=\text{mín}\ [f_{I,II,III}(X_3)+V_4(X_3; X_4)]}$ = [13+9; 12+3;11+7] = **15 (X_3=I)**

$\mathbf{f_{I,II,III,IV}(X_1;X_2;X_3;X_4=M)=}$ = **18 (X_3=I,K)**

Y finalmente, el **tramo I,II,III,IV,V**.

$\mathbf{f_{I,II,III,IV,V}(X_1;X_2;X_3;X_4;X_5=N)=\text{mín}\ [f_{I,II,III,IV}(X_4)+V_5(X_4; X_5)]}$=[15+4;18+3]=**19 ($X_4$=L)**

X_4=L,M $\quad$ X_4=L $\quad$ X_4=M

El camino de costo mínimo es A C E I L N: 19 unidades monetarias.

Las ecuaciones así planteadas se denominan **recursivas** y son las que manejan o gobiernan el sistema que estamos considerando.

Programación Dinámica Determinística. Proceso de decisión de n etapas. Política óptima.

La programación Dinámica es un método de optimización de los sistemas o de su representación matemática sobre la cual se opera por fases, secuencias o etapas. Las fases pueden ser espaciales o temporales. Un ejemplo de secuencia temporal es dar agua a una zona de riego durante 4 estaciones; un ejemplo de secuencia espacial es dar agua a 4 subzonas de riego durante una estación.

Un proceso de decisión de n etapas es el que puede separarse en un cierto número de pasos secuenciales, o etapas, las cuales pueden ser completadas en una o más formas. Las opciones para completar las etapas se denominan decisiones. Una política es una secuencia de decisiones, una para cada etapa del proceso.

La condición del proceso en una etapa dada se denomina estado en esa etapa; cada decisión produce una transición del estado actual a un estado asociado con la siguiente etapa.

Un proceso de decisión de n etapas es finito si hay solamente un número finito de etapas en el proceso y un número finito de estados asociados a c/etapa.

Muchos procesos de decisión de n etapas tienen rendimientos (costos o beneficios) asociados a cada decisión, y estos rendimientos pueden variar tanto con la etapa como con el estado del proceso.

<u>Programación Dinámica Determinística</u>. Un proceso de decisión de n etapas es determinístico si el resultado de cada decisión inmediata se conoce exactamente y queda completamente determinado por el estado y la decisión de la política en la etapa actual.

<u>Programación Dinámica Probabilística</u> difiere de la <u>Programación Dinámica Determinística</u> en que el estado en la etapa siguiente no queda completamente determinado por el estado y la decisión de la política en la etapa actual. En lugar de ello, existe una distribución de probabilidad para lo que será el estado siguiente.

El objetivo de analizar tales procesos es determinar una política óptima que de como resultado el mejor rendimiento total.

Si tenemos un sistema que puede cambiar de estado en cada etapa, por una decisión, siendo k el número de fases (variando de 1 a N, número finito numerable), se llamara política a una cierta sucesión de decisiones desde k=1 hasta k=n y llamaremos subpolítica a una serie de decisiones ligadas que forman parte de una política.

Principio de Bellman (Principio de óptimalidad).

Una política óptima tiene la propiedad de que, independientemente de las decisiones precedentes tomadas para llegar a un estado particular en una etapa particular, las decisiones que queden por tomar, deben constituir una política óptima, para abandonar ese estado.

Características generales de la Programación Dinámica.

a) Los componentes de la función objetivo, en PD son función de una variable de decisión y no proporcional como en PL.

Lineal $Z = c_1 X_1 + c_2 X_2$

Dinámica $Z = c_1 f(X_1)$

Puede aplicarse a problemas no lineales y no es necesario que la función sea continua. solo se necesita que estén definidas en algunos puntos.

b) El problema puede dividirse en etapas, con una decisión de la política requerida en cada etapa.

c) Cada etapa tiene un cierto número de estados asociados a ella.

d) El efecto de la decisión de una política en cada etapa es transformar el estado actual en un estado asociado con la etapa siguiente.

e) Dado el estado actual, una política óptima para las etapas restantes es independiente de la política adoptada en las etapas previas. Para los problemas de PD en general, el conocimiento del estado actual del sistema comunica toda la información acerca de su comportamiento previo, necesaria para determinar la política óptima de allí en adelante (propiedad de las cadenas de Markov). A veces se menciona esta propiedad como el principio de optimidad.

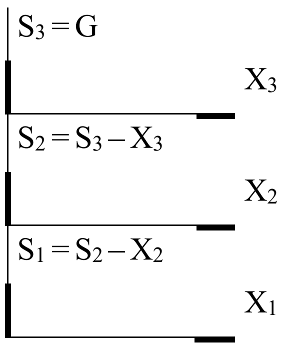
f) Se dispone de una relación recursiva que identifica la política óptima para cada estado en la etapa n, dada la política óptima para cada estado en la etapa (n+1 o n-1).

g) Usando esta relación recursiva, el procedimiento de resolución se mueve hacia atrás o hacia adelante, etapa por etapa ---hallando en cada ocasión la política óptima para cada estado de esa etapa--- hasta que se encuentra la política óptima cuando se parte de una etapa inicial.

LOGICA DE UN PROCESO DE OPTIMIZACION SECUENCIAL DE DECISION CON UNA VARIABLE DE ESTADO

Se puede identificar a **G** como un recurso que hay que distribuir óptimamente entre n usuarios, de quienes se obtendrán beneficios netos $\mathbf{v_i(X_i)}$. Supongamos de inicio, 3 usuarios (1, 2 y 3).

Un esquema de la asignación que se esta desarrollando sería:



Modelo Matemático de optimización de Programación Dinámica:

Por tratarse de un modelo de optimización, tendremos que distinguir variables de decisión, función objetivo y restricciones. En particular en los problemas de PD, se plantea para cada etapa de decisión, una ecuación recursiva con límites para las variables intervinientes (decisión y de estado).

Variables de decisión: X_i [ua], unidades de asignación al usuario i, $\forall$ i = 1, 2, 3.

Función Objetivo: Maximizar los beneficios [um] por X_i unidades de asignación

Máx Z (um) = máx $\{b_1(X_1) + b_2(X_2) + b_3(X_3)\}$, donde

$b_i(X_i)$: beneficio [um] para el usuario i, dada una asignación X_i [ua]

Restricciones: a) Disponibilidad de recurso para todos los usuarios.

$X_1 + X_2 + X_3 <= G$ (ua), donde G es la disponibilidad total

b) No negatividad $X_i >= 0 \ \forall$ i = 1, 2, 3

Variables de estado: S_i, asignación acumulada de recurso[ua], hasta el usuario i $\forall$ i = 1, 2, 3

Ecuaciones recursivas: Indican el valor de la función objetivo hasta la etapa i. Son una discretización de la función objetivo para cada etapa. Obsérvese que para cada ecuación recursiva, están restringidas cada una de las variables.

$$f_1(S_1) = \underset{0 <= X_1 <= S_1}{\text{máx}} \{b_1(X_1)\}, \qquad 0 <= S_1 <= G$$

$$f_2(S_2) = \underset{0 <= X_2 <= S_2}{\text{máx}} \{b_2(X_2) + f_1(S_2 - X_2)\}, \qquad 0 <= S_2 <= G$$

$$f_3(S_3) = \underset{0 <= X_3 <= S_3}{\text{máx}} \{b_3(X_3) + f_2(S_3 - X_3)\}, \qquad 0 <= S_3 <= G$$

El procedimiento de resolución empieza suponiendo que solamente falta una asignación del recurso (al usuario 1) y que existe una cantidad de recurso disponible con valor **$0 <= S_1 <= G$.**

En la primer ecuación definimos **$f_1(S_1)$** como el mejor uso posible del recurso disponible por parte del usuario 1.

$$\mathbf{f_1(S_1) = máx\ [v_1(X_1)] \qquad 0 <= X_1 <= S_1}$$

Si ahora hubiera 2 etapas de decisión en el proceso, la optimización conjunta (máximo beneficio total) sería:

$$\mathbf{f_2(S_2) = máx\ [v_2(X_2) + f_1(S_1)] \qquad 0 <= X_2 <= S_2}$$

Pero como las decisiones deben tomarse en la misma etapa 2 (con la misma variable **X_2**), y como **$f_1(S_1) = f_1(S_2\text{-}X_2)$** queda:

$$\mathbf{f_2(S_2) = máx\ [v_2(X_2) + f_1(S_2-X_2)] \qquad 0 <= X_2 <= S_2}$$

De esta forma, incorporando etapas, podemos plantear las ecuaciones recursivas para todas las etapas del problema y así hallar el beneficio total máximo.

Resolución de un ejemplo

Asignación a una zona de riego durante 4 meses, conociendo las funciones de beneficio.

Dado un volumen de agua en un embalse se debe satisfacer la demanda de una zona de riego en un período de 4 meses. Si se conocen las funciones de beneficio por volúmenes regados para cada mes y el volumen total disponible es menor que la suma de volúmenes individuales en cada mes y es igual a 50 hm^3.

La tabla siguiente indica los beneficios (um) de las asignaciones para 0, 10, 20, 30, 40 y 50 hm^3, en los diferentes meses considerados.

hm^3	0	10	20	30	40	50
b_1	0	2	2	4	8	10
b_2	0	3	4	7	7	8
b_3	0	1	3	5	9	7
b_4	0	4	4	4	5	7

Encontrar la asignación que optimice el objetivo del problema.

Modelo Matemático:

Variables de decisión: X_i, asignación de agua [hm^3], en el mes i $\forall$ i = 1, 2, 3, 4.

Función Objetivo: Maximizar los beneficios [um] por volúmenes regados X_i, en los meses i:

Máx Z (um) = máx $\{b_1(X_1) + b_2(X_2) + b_3(X_3) + b_4(X_4)\}$, donde

$b_i(X_i)$: beneficio [um] en el mes i, dada una asignación X_i [hm^3]

Restricciones: a) Volumen disponible mayor ó igual que la suma de volúmenes óptimos de cada mes.

$X_1 + X_2 + X_3 + X_4 <= V$ (hm^3), donde V es la disponibilidad total

b) No negatividad $X_i >= 0 \ \forall$ i = 1, 2, 3, 4

Variables de estado: S_i, asignación de agua acumulada [hm^3], hasta el mes i $\forall$ i = 1, 2, 3, 4.

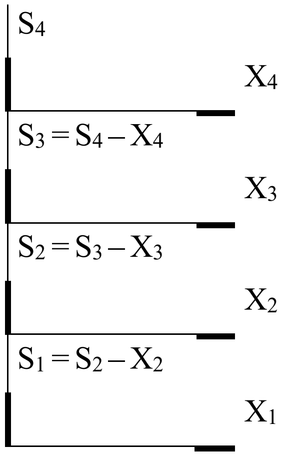
Ecuaciones recursivas: Indican el valor de la función objetivo hasta la etapa i. Son una discretización de la función objetivo para cada etapa.

$f_1(S_1) = \text{máx } \{b_1(X_1)\}$, $0 <= S_1 <= V$
$0 <= X_1 <= S_1$

$f_2(S_2) = \text{máx } \{b_2(X_2) + f_1(S_2 - X_2)\}$, $0 <= S_2 <= V$
$0 <= X_2 <= S_2$

$f_3(S_3) = \text{máx } \{b_3(X_3) + f_2(S_3 - X_3)\}$, $0 <= S_3 <= V$
$0 <= X_3 <= S_3$

$f_4(S_4) = \text{máx } \{b_4(X_4) + f_3(S_4 - X_4)\}$, $0 <= S_4 <= V$
$0 <= X_4 <= S_4$



Resolución: como el problema tiene 4 (n = 4) etapas, y para todo problema de programación dinámica sólo es necesario n – 1 tablas de resolución (independientemente de su forma), realizaremos 3 tablas; en la primera valuaremos las 2 primeras ecuaciones recursivas.

$S_2 = S_1 + X_2$	X_2	$b_2(X_2)$	S_1	$f_1(S_1)$	$f_2(S_2) = \text{máx } \{b_2(X_2) + f_1(S_2 - X_2)\}$
0 *	0 *	0	0 *	0	0 *
10	0	0	10	2	2
	10	3	0	0	3 *

$S_3 = S_2 + X_3$	X_3	$b_3(X_3)$	S_2	$f_2(S_2)$	$f_3(S_3) = \text{máx } \{b_3(X_3) + f_2(S_3 - X_3)\}$
0	0	0	0	0	0 *
10					
20					
30					
40 */**					
50					

$S_4 = S_3 + X_4$	X_4	$b_4(X_4)$	S_3	$f_3(S_3)$	$f_4(S_4) = \text{máx } \{b_4(X_4) + f_3(S_4 - X_4)\}$
50 */**	0	0	50	12	12

Solución óptima * **$Z = 13 \times 10^3$ um** **Solución óptima **** **$Z = 13 \times 10^3$ um**

$S_1 = 0$	$X_1 = 0$	$S_1 = 10$	$X_1 = 10$
$S_2 = 0$	$X_2 = 0$	$S_2 = 40$	$X_2 = 30$
$S_3 = 40$	$X_3 = 40$	$S_3 = 40$	$X_3 = 0$
$S_4 = 50$	$X_4 = 10$	$S_4 = 50$	$X_4 = 10$