# ASIGNACION DE RECURSOS: *Una aplicación de la Programación Dinámica*

Sr. Alfonso Sarmiento Vásquez

*El propósito de este artículo no es hacer una explicación detallada sobre la teoría y los métodos de resolución en los cuales se basa la Programación Dinámica, sino más bien, plantear un tipo de problema como es el de asignación de recursos que se puede resolver usando esta técnica y apoyándonos para ello en un programa de computación.*

Este artículo se ha dividido en dos partes. En la primera parte se tocarán los conceptos básicos de la Programación Dinámica (P.D) que son necesarios para entender como trabaja el programa. En la segunda parte se planteará y resolverá un caso de aplicación con ayuda del programa computacional, ya que intentar resolverlo en forma manual requeriría un esfuerzo de cálculo muy amplio.

## I CONCEPTOS BASICOS DE LA PROGRAMACION DINAMICA

La Programación Dinámica es una técnica que sirve para resolver problemas de optimización en los cuales hay que tomar decisiones interrelacionadas. Para ello, el problema original se descompone en una serie de etapas.

Por el Principio de Optimalidad de Bellman (creador de esta técnica), la solución secuencial de los problemas de decisión asociados con cada etapa es equivalente a la solución del problema de decisión original. En otras palabras lo que se hace es dividir el problema total de "n" decisiones en "n" subproblemas de una decisión.

A diferencia de la Programación Lineal no existe en Programación Dinámica un planteamiento estándar del problema que queremos resolver, es decir, la P.D no es un modelo matemático al cual sea posible asociar un algoritmo que pueda programarse de una sola vez para resolver todos los problemas que satisfagan las consideraciones del modelo. Por lo tanto la formulación de los problemas debe ajustarse a cada situación particular.

El programa de computación que se utilizará es el P4, desarrollado por el autor de este artículo y permite resolver ciertos problemas cuya formulación está predefinida.

A continuación se citarán algunas características de los modelos de P.D.

1) El problema puede dividirse en etapas con una decisión de la política requerida en cada etapa.
2) Cada etapa tiene un cierto número de estados asociados a ella. Un estado viene a ser una descripción del sistema en un determinado momento.
   Se puede definir un vector de estado S.

$$S = (s_1, s_2, \ldots, s_n)$$

donde $S_i$ = variable de estado i

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

Etapa n — Etapa n+1

$$S_n \xrightarrow{\text{decisión}} S_{n+1}$$

4) Dado el estado actual, una política óptima para las etapas restantes es independiente de la política adoptada en las etapas previas (Principio de Optimalidad).
   Esto quiere decir que para problemas determinísticos, el estado actual en cualquier etapa tiene que ser un resumen suficiente de todos aquellos aspectos de la condición actual del proceso que sean importantes para decisiones futuras o para el desempeño futuro del proceso.
5) Es posible establecer una relación recursiva mediante la cual se identifica la política óptima para cada estado en cada etapa.

Se verá a continuación un ejemplo muy sencillo cuya finalidad es únicamente aclarar los conceptos arriba mencionados y a familiarizarnos con las variables que maneja el programa P4, más que mostrar la potencia de la P.D.

En este ejemplo se generarán los estados de la misma forma como lo realiza el programa y además se hará una interpretación de la solución óptima.

El ejemplo consiste en un prototipo de problema que se presenta a menudo en el embarque y transporte de carga. En la literatura, este problema recibe el nombre descriptivo de problema tipo mochila (alforja), en inglés Knapsack problem, y se puede considerar como un caso específico del problema de asignación de recursos.

Ejemplo

Un comerciante posee 4 artículos, los cuales debe llevar al mercado para venderlos haciendo un sólo viaje. Para ello posee una camioneta con una capacidad de 13 $m^3$.

A continuación se describen los 4 artículos, indicando para cada uno su volumen ($m^3$) y la utilidad que obtendría el comerciante si los vendiera.

TABLA I

| ARTICULO | 1 | 2 | 3 | 4 | i |
|---|---|---|---|---|---|
| VOLUMEN ($m^3$) | 5 | 6 | 7 | 2 | a (i) |
| UTILIDAD | 20 | 30 | 40 | 50 | c (i) |

a(i) = volúmen del artículo
c(i) = utilidad del artículo i

Debido a que la limitación de volumen de la camioneta impide que se puedan cargar los 4 artículos en un sólo viaje. Se quiere saber que artículos debe subir a la camioneta de tal forma que la utilidad que se obtenga por todos ellos sea la mayor.

Para este problema los estados quedan definidos con 2 variables de estado.

Estado: $S = (s_1, s_2)$

donde:

$s_1$ = el artículo para el cual se va a tomar la decisión
$s_1 = \{1,2,3,4\}$
$s_2$ = el volumen disponible ($m^3$) no utilizado
$s_2 = \{0,1,...,13\}$

Antes de decidir que artículos subir a la camioneta, el volumen disponible de ésta es el total, o sea 13$m^3$

Supongamos que ya se hubieran subido a la camioneta los artículos 2 y 3, los cuales juntos ocupan un volumen de 13 $m^3$, entonces ahora el volúmen disponible no utilizado sería de 0 $m^3$, es decir, ya no queda espacio para otro artículo.

Es por ello que el rango para la variable $s_2$ varía de 0 a 13.

Este problema se puede dividir en 4 etapas, una para cada artículo. En cada etapa habrá que tomar una decisión, la cual se representará de la siguiente forma:

$$d = \begin{cases} 0, & \text{no subir el artículo a la camioneta} \\ 1, & \text{subir el artículo a la camioneta} \end{cases}$$

Esto quiere decir, cada vez que se tome una decisión para cada artículo caben sólo 2 posibilidades, subirlo o no a la camioneta. Se ana-lizarán ahora todos los artículos uno a uno y secuencialmente, con el fin de generar los estados en los cuales se basa la P.D para obtener la solución óptima, en este caso la máxima utilidad.

Se debe partir de un estado inicial.

Estado Inicial: $(s_1,s_2) = (1,13)$

Este estado inicial indica que se está en la etapa 1, es decir, se está frente al artículo 1 ($s_1$=1). Además se tiene la camioneta totalmente vacía, o lo que es lo mismo, el volumen disponible no utilizado es el máximo, 13 m³ ($s_2$=13).

En este momento el comerciante debe tomar una decisión. Supongamos que decidió subir el artículo 1 a la camioneta. Tomada la decisión se pasa a la etapa 2, o equivalentemente, se está frente al artículo 2 ($s_1$=2).

Sin embargo, ahora el volumen disponible no utilizado será el volumen máximo menos el volumen que ocupa el artículo 1, es decir,

$$13 - 5 = 8 \text{ m}^3 \ (s_2 = 8)$$

Se puede decir que se ha generado el estado (2,8).

¿Qué hubiera pasado si el comerciante hubiera decidido no subir el artículo 1 a la camioneta ?. Se estaría ahora frente al artículo 2 con la camioneta vacía, o equivalente-mente, con el volumen máximo no utilizado, generando de esta manera el estado (2,13).

Partiendo del estado (2,13) y aplicando el mismo criterio se generan 2 estados más: (3,13) y (3,7).

En la tabla 2 se puede apreciar la generación de estados.

Esta generación de estados puede plantearse matemáticamente me-diante una función para cada variable de estado denominada Función Transición. Para este problema en particular la Función Transición quedaría definida así:

$$sn_1 = s_1 + 1$$
$$sn_2 = s_2 - a\,(s_1) * d$$

donde a ($s_1$) es el volúmen del artículo $s_1$ (Ver tabla 1).

Restricciones

$$1 \leq sn_1 \leq 4$$
$$0 \leq sn_2 \leq 13$$

Los estados pueden representarse también formando una red. (Ver Red)

Si además se ordenan por etapas en orden creciente de la variable $s_2$ se obtiene la tabla 3. El programa P4 tiene una opción para ver esta tabla.

Existe una función denominada la Función de Valor Optimo. Esta función mide el objetivo que se pretende alcanzar, que en este caso es maximizar utilidades.

$f(S) = f(s_1, s_2)$ = el valor de la <u>máxima</u> utilidad que se puede obtener desde el estado inicial hasta el estado $(s_1, s_2)$.

Para el estado inicial (1,13) se tiene que $f(1,13) = 0$, esto quiere decir que, inicialmente mientras no se tome decisión alguna las utilidades permanecen nulas.

Sea a (S, d) el beneficio que se obtiene desde el estado $S_{d-1}$ hasta el estado S al tomar la decisión d..

$$S_{n-1} \xrightarrow{a\,(S,\,d)} S$$

Para este ejemplo

$a(S,d) = c(s_i)^* d$, siendo $c(s_1)$ el volúmen del artículo $s_1$.

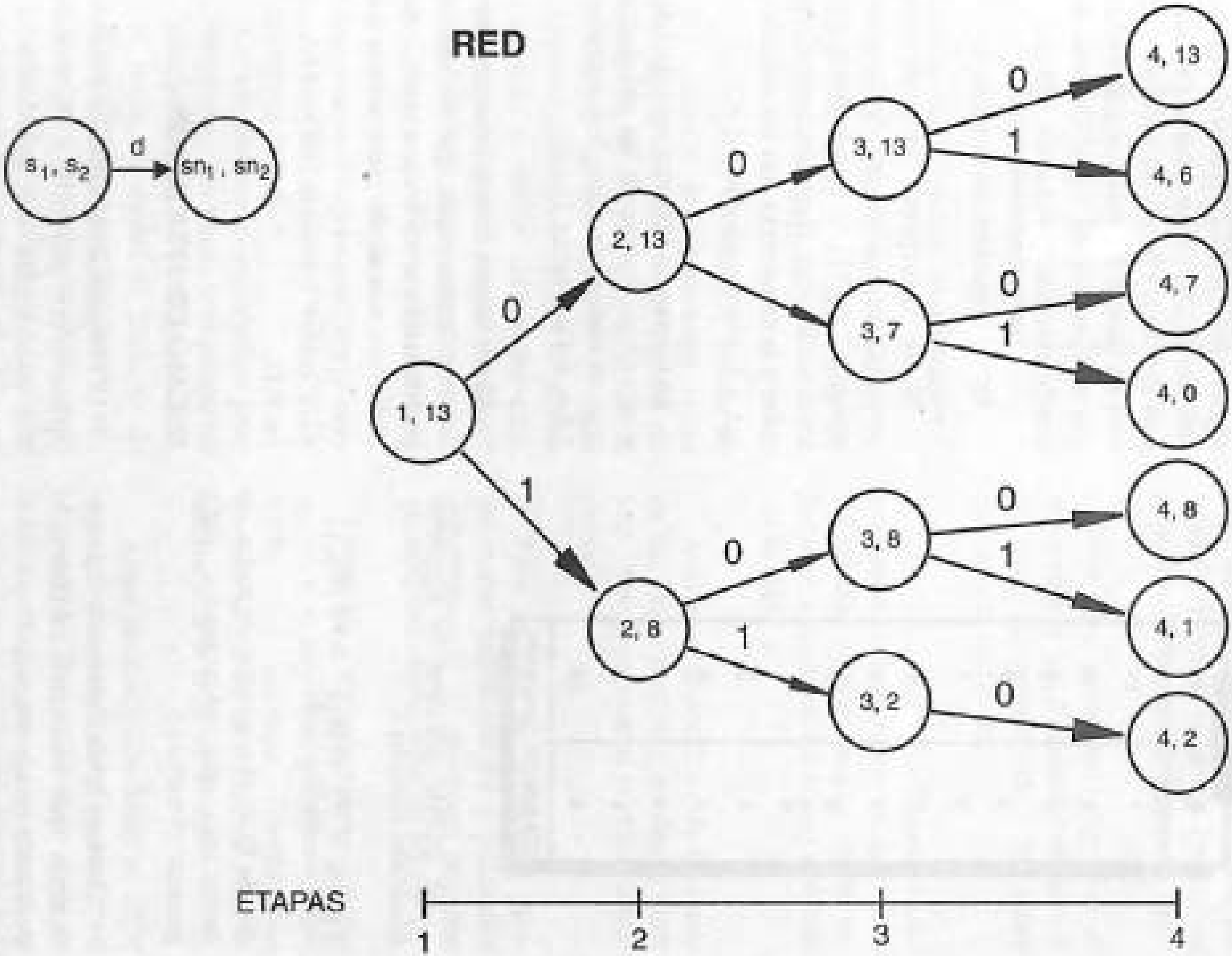
Esto quiere decir que si d=0 no se obtiene ninguna utilidad, sin embargo si d = 1 la utilidad obtenida depende de la etapa $s_1$ y es $c(s_1)$.

TABLA II

| $s_1$ | $s_2$ | d | $sn_1$ | $sn_2$ |
|---|---|---|---|---|
| 1 | 13 | 0 | 2 | 13 |
| | | 1 | 2 | 8 |
| 2 | 13 | 0 | 3 | 13 |
| | | 1 | 3 | 7 |
| 2 | 8 | 0 | 3 | 8 |
| | | 1 | 3 | 2 |
| 3 | 13 | 0 | 4 | 13 |
| | | 1 | 4 | 6 |
| 3 | 7 | 0 | 4 | 7 |
| | | 1 | 4 | 0 |
| 3 | 8 | 0 | 4 | 8 |
| | | 1 | 4 | 1 |
| 3 | 2 | 0 | 4 | 2 |
| | | 1 | (*) - | - |

(*) como la variable $s_2$ es negativa no se genera el estado

## RED



s1, s2
d
sn1, sn2
1, 13
0
1
2, 13
2, 8
0
0
1
3, 13
3, 7
3, 8
3, 2
0
1
0
1
0
1
0
4, 13
4, 6
4, 7
4, 0
4, 8
4, 1
4, 2
ETAPAS
1
2
3
4

TABLA III

| $s_1$ | $s_2$ |
|---|---|
| 1 | 13 |
| 2 | 8 |
| 2 | 13 |
| 3 | 2 |
| 3 | 7 |
| 3 | 8 |
| 3 | 13 |
| 4 | 0 |
| 4 | 1 |
| 4 | 2 |
| 4 | 6 |
| 4 | 7 |
| 4 | 8 |
| 4 | 13 |
| (14 estados generados) | |

Se puede plantear la siguiente ecuación recursiva.

$$f(S) = \max_{d \in D} \; c(s_1) * d + f(S_{n-1})$$

donde D es el conjunto de todas las decisiones que se tomaron para generar el estado S.

Con esta ecuación recursiva garantizamos que el beneficio obtenido para cada estado en cada etapa sea el máximo posible.

Como se indicó al comienzo del artículo, no se va a desarrollar el algoritmo de resolución (tal vez en un artículo posterior se podrá explicar con detalle). Por ahora sólo se dirá que en base a los estados generados y a la Función de Valor Optimo se puede calcular la solución óptima.

Esta se muestra en la tabla 4.

Si se interpreta la solución centrándose únicamente en las variables "$s_1$" y "d", se puede decir que los artículos que el comerciante debe subir a la camioneta son los artículos 1,2 y 4, pues para ellos d = 1.

La Ganancia Máxima será de 100, y se calcula sumando las utilidades que se obtendrían con los artículos 1,2 y 4. (Ver tabla 1)

El problema anterior fue un ejercicio bastante simple, que incluso se podría haber resuelto por tanteo, sin embargo, nos sirvió para aclarar los conceptos básicos que son necesarios para poder formular los problemas en P.D.

## II CASO DE APLICACION

La Programación Dinámica puede aplicarse con provecho a algunos de los problemas de selección de inversiones, fundamentalmente a los que se refieren a inversiones de implantación o de expansión en los

| TABLA IV | | |
|---|---|---|
| $s_1$ | $s_2$ | d |
| 1 | 13 | 1 |
| 2 | 8 | 1 |
| 3 | 2 | 0 |
| 4 | 2 | 1 |

cuales existe opción de alternativas.

A continuación se presentará un caso orientado a la selección de inversiones de expansión.

La empresa DINAP S.A. dueña de una cadena de almacenes tiene prevista la ampliación de capital de alrededor de 10 millones de soles. El Directorio ha determinado que ese dinero se destine a extender la cadena con uno o varios almacenes más, dejando que sea el comité ejecutivo quien decida si es mejor abrir un almacén o repartir los medios disponibles entre varios de menor superficie.

Para ello se han realizado los siguientes estudios:

- Un estudio socio-económico para determinar las zonas más propicias al negocio, con base en la población, la renta, el gasto, el poderío de otros almacenes ya existentes, etc.
- Un estudio de mercado en las cinco zonas más favorables sobre hábitos y lugares de compra.

En base a ambos estudios se ha podido llegar a una presunta potencialidad de compra en cada zona, así como a estimar ventas y gastos en función del tamaño a implementar.

Por otra parte se sabe que el inmueble y las instalaciones tienen un costo aproximado de 1 000 soles por metro cuadrado.

Con todo ello se ha llegado a una previsión de beneficios en cada una de las cinco zonas, según sea la superficie a dedicar, tal como aparece en la tabla 5.

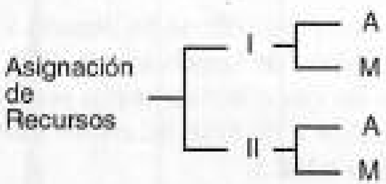
Averiguar la política más rentable de las cinco posibles, es decir, abrir uno, dos, tres, cuatro o cinco almacenes.

Para cargar el programa simplemente tecleamos P4 y pulsamos <ENTER>. (Si se tiene un monitor monócromo tecleamos P4 MONO)

Una vez cargado el programa aparece el logotipo, pulsamos <ENTER> nuevamente. Ahora nos encontramos en el editor el cual es similar a una hoja de cálculo.

Antes de llenar los datos se debe seleccionar el tipo de algoritmo con el cual se va a trabajar; esto es importante porque así el programa optimiza la utilización de la memoria. Por ejemplo hay ciertos algoritmos que sólo utilizan una columna para llenar sus datos, pudiendo así asignársele más filas.

El tipo de algoritmo para resolver nuestro problema es el de Asignación de Recursos. El programa tiene formulados hasta 4 algoritmos de Asignación de Recursos, los cuales se pueden clasificar de la siguiente forma:




Este cuadro indica que dependiendo del tipo de problema se puede usar una o dos tablas para ingresar los datos ( I o II ) y dependiendo de la Función de Valor Optimo, ésta puede ser aditiva o multiplicativa (A o M).

Para este caso sólo se tiene una tabla (INVERSION vs BENEFICIOS); siendo la Función de Valor Optimo aditiva (A), ya que el beneficio total es la suma de los beneficios obtenidos en cada una de las zonas.

Por lo tanto seleccionamos el algoritmo 6 posicionando el cursor en el lugar correspondiente y pulsando <ENTER>. (Ver figura 1).

Para este caso particular las variables de estado $s_1$, $s_2$, y d quedarían definidas de la siguiente manera:

$s_1$ = la zona para la que se va a decidir cuanto asignar.

Como la variable $s_1$ debe ser numérica vamos a considerar las zonas V, W, X, Y, Z como las zonas 1, 2, 3, 4, 5.



TABLA V

| SUPERFICIE (m$^2$) | INVERSION (mill. soles) | BENEFICIOS PARA CADA ZONA (mill. soles) | | | | |
|---|---|---|---|---|---|---|
| | | V | W | X | Y | Z |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1000 | 1 | 14 | 12 | 7 | 3 | 5 |
| 2000 | 2 | 18 | 19 | 12 | 8 | 7 |
| 3000 | 3 | 22 | 27 | 20 | 11 | 9 |
| 4000 | 4 | 29 | 32 | 25 | 19 | 13 |
| 5000 | 5 | 35 | 37 | 29 | 21 | 19 |
| 6000 | 6 | 38 | 40 | 33 | 24 | 23 |
| 7000 | 7 | 46 | 42 | 37 | 28 | 29 |
| 8000 | 8 | 49 | 44 | 42 | 34 | 32 |
| 9000 | 9 | 51 | 45 | 44 | 38 | 39 |
| 10000 | 10 | 53 | 46 | 45 | 40 | 43 |

FIGURA 1

1. Alforja
2. Reemplazo de Equipos
3. Asig. de Recursos II (M)
4. Carga de Bultos
5. Asig. de Recursos II (A)
6. Asig. de Recursos I (A)
7. Asig. de Recursos I (M)
8. Control de Inventarios

$s_1$ = { 1,2,3,4,5 }
$s_2$ = la cantidad de millones de soles no asignados.
$s_2$ = { 0,1,...,10 }
d = cantidad de millones de soles a asignar a la zona $s_1$
d = { 0,...,$s_2$ }
Estado Inicial ($s_1$, $s_2$) = (1,10)

Función de Valor Optimo

f(s) = el máximo beneficio que se puede obtener desde la zona 1 hasta la zona $s_1$, partiendo con $s_2$ millones de soles

Función Transición

$$sn_1 = s_1 + 1$$
$$sn_2 = s_2 - d$$

Restricciones

$$1 \leq sn_1 \leq 5$$
$$0 \leq sn_2 \leq 10$$

Ecuación Recursiva

$$f(S) = \max_{d \in D} c(s_1, d) + f(S_{n-1})$$

donde, c ($s_1$,d) es el beneficio de la zona $s_1$ al habérsele asignado "d" millones de soles.

Para poder llenar los datos de la tabla 5 hay que tener en cuenta dos detalles:

Primero, estos deben ser llenados según la forma del indicador fila/ columna (se encuentra en la parte superior izquierda).

Para este algoritmo el indicador es de la forma "**d / n**" , por lo tanto las filas corresponden a las decisiones y las columnas a los niveles o etapas (en este caso las zonas).

Segundo, hay que tener en cuenta cual es la mínima decisión que se puede hacer. Para este problema la menor decisión que se puede hacer es cero, es decir se puede no asignar ningún millón de soles a alguna de las zonas. Sin embargo vemos en el editor que la numeración de las filas empieza en el número 1 y no en el cero como debería ser. Para lograr esto simplemente pulsar la tecla F5.

Una vez ingresados los datos (Ver figura 2) se llama al menú de opciones pulsando la tecla slash "/". Aquí se selecciona la opción **Generar** (Ver figura 3), apareciendo una ventana con 3 alternativas. Se escoge la alternativa **Rango**, apareciendo una segunda ventana. En esta ventana hay 3 posiciones. En la primera se tiene una letra "**M**" que indica que la Función de Valor Optimo será maximizada, si se quisiera minimizarla se pulsa la barra espaciadora cambiando la "**M**" por "**m**".

Como en el problema se busca maximizar beneficios se deja la opción tal como está, es decir en "**M**".

Para pasar a la siguiente posición se pulsa la tecla <TAB>.

Aquí se llena el número de niveles o etapas que se quieren leer del editor de datos, en este caso se teclea 5 (no pulsar <ENTER> todavía).

Ahora se pasa a la siguiente posición pulsando nuevamente la tecla <TAB>. Aquí se llena el número de decisiones que se quieren leer del editor, en este caso se teclea 10. Si se hubiera cometido algún error al llenar las posiciones anteriores se puede volver a ellas pulsando <SHIFT TAB>. Si no hay errores se pulsa ahora sí <ENTER>.

Se ve que la segunda ventana desaparece quedando la primera, aquí se selecciona la opción **Ir**.

En ese momento aparece una ventana que indica la cantidad de estados que se van generendo y además los niveles que se van recorriendo (Ver figura 4).

Para este caso se han generado 45 estados y se han recorrido los 5 niveles (5/5 significa 5 niveles recorridos de los 5 posibles).

La respuesta o ruta solución es mostrada en función de las 3 variables $s_1$, $s_2$ y d. Sin embargo si nos centramos en las variables "$s_1$" y "d" y las interpretamos podemos dar una respuesta administrativa.

| ZONA | INVERSION (mil. soles) | SUPERFICIE ($m^2$) |
|---|---|---|
| V | 1 | 1000 |
| W | 5 | 5000 |
| X | 3 | 3000 |
| Y | 0 | 0 |
| Z | 1 | 1000 |

Si se asigna el capital de 10 millones de soles para abrir 4 almacenes conforme al cuadro anterior, se obtendrá una utilidad máxima de 76 millones de soles.

Finalmente se puede decir que para resolver un problema por P.D es importante saber identificar las etapas, las variables de estado, la variable de decisión, así como saber plantear la Función Transición y la Función de Valor Optimo. Por lo tanto se requiere de cierto grado de ingenio y de visión de la estructura general de los problemas de P.D, a fin de reconocer cuando un problema se puede resolver mediante los procedimientos de esta programación y como se haría.

Para Harvey Wagner formular un modelo en términos de la relación de recurrencia de la P.D es en parte un arte.

Sólo a través del estudio de un cierto número de ejemplos representativos de distintos tipos será posible acostumbrarse a visualizar un problema en términos de una técnica de solución recurrente en vez de un modelo matemático.


Bibliografía

Daellenbach, Hans. Introducción a las Técnicas de Investigación de Operaciones

Hiller, Frederich. Introducción a la Investigación de Operaciones

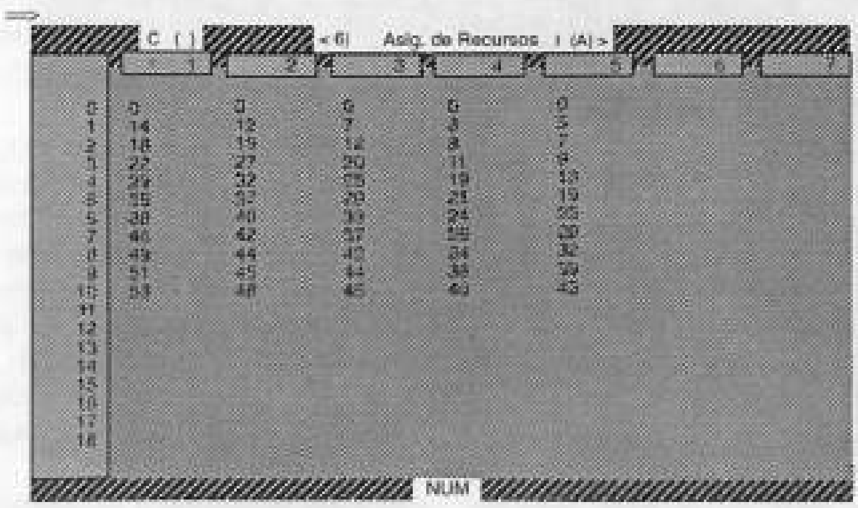
Muro Saénz, Javier. Práctica de la Investigación Operativa Empresarial

Prawda, Juan. Métodos y Modelos de Investigación de Operaciones. Vol.I.

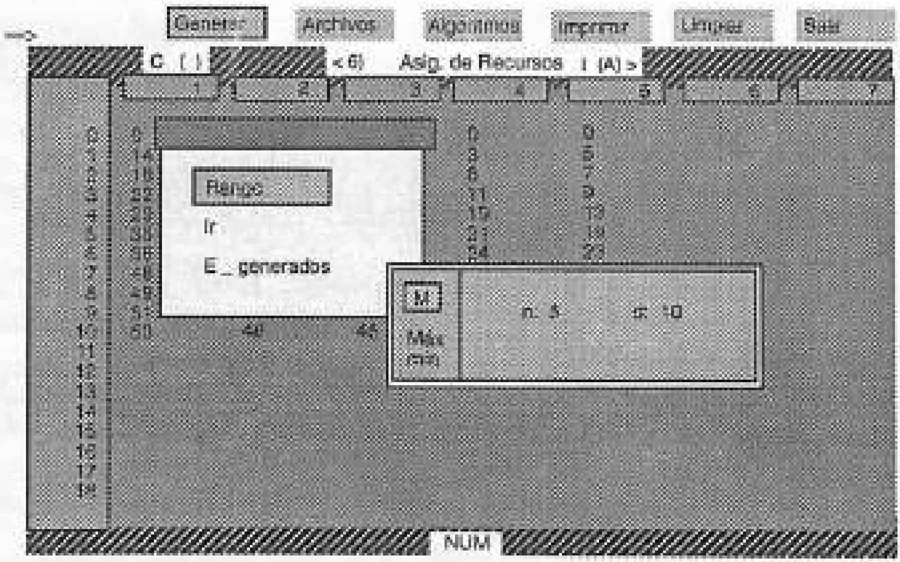
Varella, Jaime. Investigación de Operaciones

Wagner M., Harvey. Principles of Operations Research


FIGURA 2



## FIGURA 3




## FIGURA 4