



Analisis y diseño de algoritmos

Unidad II
Practica 04:
Programación dinamica
Fernando Ruiz Correa 3M2

Docente:
M. en C. Erika Sanchez-Fermat

Unidad Profesional Interdedisciplinaria de Ingenieria Campus
Zacatecas
Instituto Politecnico Nacional
Diciembre 2023

1. Introducción

Al terminar la Segunda Guerra Mundial, se empezó a considerar que había un gran número de actividades que podían ser clasificadas como problemas de decisión divisibles en subproblemas en diversos campos como ingeniería, economía, industrial y militar. Pronto se vió que estos problemas matemáticos que iban surgiendo sobrepasaban los límites convencionales del análisis y que se necesitaban nuevos métodos para una resolución exitosa de estos.

Las técnicas clásicas de cálculo eran a veces útiles en estas nuevas áreas, pero estaban claramente limitadas en alcance y versatilidad, y en ocasiones eran ineficientes en lo que se refiere a proporcionar respuestas numéricas. La aceptación de estos hechos llevó a la creación de una serie de nuevos métodos y teorías matemáticas. Entre estos se halla la técnica de la programación dinámica, basada en el principio de optimalidad, con un gran potencial para el campo de las computadoras digitales, que en aquel momento se encontraba en plena expansión. Esta fue desarrollada por el matemático Richard E. Bellman en 1953, y posteriormente en su libro [BeDr].

El principio de optimalidad de Bellman dice que, dadas las decisiones óptimas que se toman en cada una de las etapas, se pueden ir dividiendo en pequeños problemas, estas no dependerán de las decisiones óptimas que se hayan tomado en las etapas anteriores y que la solución global del problema se obtiene a partir de las soluciones de los subproblemas.

Basandose en esta idea se puede ir resolviendo de manera recursiva cada uno de los subproblemas de manera independiente, los cuales serán más sencillos que el problema principal, para al final combinar todas las decisiones y así hallar la solución.

2. Desarrollo

2.1. ¿Qué es?

La programación dinámica es una técnica de resolución de problemas en informática y matemáticas que se utiliza para optimizar la eficiencia de algoritmos mediante la descomposición de un problema en subproblemas más pequeños y resolviéndolos de manera recursiva. La técnica de programación dinámica es adecuada para resolver problemas cuya solución debe resolverse de manera recursiva, como el caso de los problemas de divide y vencerás, en la que los subproblemas se resuelven de manera más sencilla hasta un punto determinado, sin necesidad de repetir innecesariamente pasos, mediante la memorización de cada subproblema en una tabla.

2.2. ¿Cómo funciona?

La programación dinámica funciona de manera que se divide un problema en pequeños subproblemas y que se resuelven de una manera recursiva, como se había mencionado ante-

riormente, sin embargo existen propiedades fundamentales que deben cumplirse para que la programación dinámica funcione de manera adecuada.

- Solapamiento de subproblemas:
El problema original, debe poder dividirse en problemas mas pequeño, de tal manera que se resuelvan de manera independiente, estos problemas deben compartir soluciones forzosamente, debido a que este problema se resolvera muchas veces.
- Subestructura Optima
La solución general debe obtenerse forzosamente combinando las soluciones de los subproblemas.

La resolución de este tipo de problemas se puede obtener de la siguiente manera:

- Dividir el problema en sub problemas.
- Resolver cada subproblema.
- Almacenar soluciones en una memoria o tabla.
- Combinar todas las soluciones obtenidas.

2.3. Características esenciales

- El problema se puede dividir en etapas que requieren una política de decisión en cada una.
- Cada etapa tiene cierto número de estados asociados a ella.
- El efecto de la política de decisión en cada etapa es transformar el estado actual en un estado asociado con la siguiente etapa.
- El procedimiento de solución esta diseñado para encontrar una política óptima para el problema completo.
- Dado un estado actual, una política subsecuente óptima para las etapas restantes es independiente de la política adoptada en las etapas anteriores (principio de optimalidad).

- El procedimiento de solución se inicia al encontrar la política optima para la ultima etapa.
- Se dispone de una relación recursiva que identifica la política optima par la etapa n dada la política optima para la etapa $(n+1)$

2.4. Ventajas y desventajas

Ventajas

- Una de las principales ventajas de usar programación dinámica es que acelera el procesamiento, ya que se usan referencias que fueron previamente calculadas. Como es una técnica de programación recursiva, reduce las líneas de código del programa.

Desventajas

- Se necesita mucha memoria para almacenar el resultado calculado de cada subproblema, sin poder garantizar que el valor almacenado se utilizará o no.
- Muchas veces, el valor de salida se queda almacenado sin nunca ser utilizado en los siguientes subproblemas durante la ejecución. Esto conlleva a una utilización innecesaria de la memoria.
- En la programación dinámica las funciones se llaman recursivamente. Esto hace que la memoria de pila se mantenga en constante aumento.

2.5. Recursividad

Existen dos formas de plantear la fórmula de recursividad en los problemas de programación dinámica:

- Recursividad de Retroceso: el problema se resuelva partiendo de la última etapa hacia la primera.
- Recursividad de Avance: el problema se resuelve partiendo de la primera etapa hacia la última.

Cada etapa se define dependiendo el problema que se quiere solucionar, se puede decir en terminos de calculo de complejidad, ambas son iguales, pues no afecta el orden de las etapas.

2.6. Ejemplo de aplicación

- *problema de la mochila* El problema de la mochila (Knaspack problem) es un problema clásico. Este problema es considerado NP (Non Probabilistic Problem) ya que existe una combinación exponencial de instancias que, en su totalidad, no pueden ser resueltas. Existen variantes relacionadas con este problema: problema con cantidad de productos limitada, problema con cantidad de productos ilimitada, elección múltiple, elección de un producto de diferentes categorías, como un problema relacionado con el peso de los productos, como un problema relacionado con el monto económico, entre otros.

El problema de la mochila (KP) puede ser definido con un conjunto de n artículos donde cada artículo es identificado por n_x , con un valor entero p_x , y un peso w_x . El problema consiste en elegir un subconjunto de n artículos maximizando el beneficio obtenido considerando el peso total de los artículos seleccionados, sin exceder la capacidad c de la mochila.

3. Conclusiones

- La programación dinámica es un enfoque de la programación que permite resolver problemas de una manera mas sencilla, aplicando la técnica de divide y vencerás. Sin embargo, es necesario conocer los problemas a fondo, pues si no se conocen a fondo, es muy difícil identificar los subproblemas y los caso base.
- La programación dinámica no siempre se puede utilizar, y es necesario aclarar que no siempre es la mejor alternativa, es necesario saber si el problema se puede dividir en pequeños problemas, de lo contrario no se puede aplicar este enfoque de programación.

4. Referencias

- Corvo, H. S. (2020, 14 marzo). Programación dinámica: características, ejemplo, ventajas, desventajas. Lifeder. <https://www.lifeder.com/programacion-dinamica/>
- Del Estado De Hidalgo, U. A. (s. f.). Boletín Científico:: UAEH. <https://www.uaeh.edu.mx/scige/bolet>