

Problemas, Modelos y Recursos

La complejidad es el estudio matemático de la computación y se dice que las matemáticas se encargan de estudiar los números y el espacio, sin embargo las matemáticas van más allá que esos conceptos.

Las matemáticas se relacionan con las actividades más primitivas del ser humano, abarcando el contabilizar, medir el tiempo y estudiar el movimiento usando números, intervalos, formas y distancias. El concepto de matemáticas fue dado por Saunders Mac Lane (1909-2008), matemático estadounidense, cofundador de la teoría de categorías.

Todas las actividades mencionadas anteriormente se formalizan en forma de teoremas, conceptos, axiomas, definiciones y pruebas.

Así que la complejidad, al ser un estudio matemático, abarca todo lo mencionado anteriormente y así estudia todos los recursos que se requieren en la computación para resolver problemas.

Es importante mencionar que tanto la teoría y abstracción de conceptos matemáticos se puede ver limitada por dos factores importantes; los recursos de un equipo de cómputo, ya que se requiere memoria y tiempo; y de nuestros conocimientos como seres humanos.

La complejidad juega un papel muy importante en este sistema o problema, ya que se espera que a partir de modelos formales para algunos problemas que son simples se pueda desarrollar un análisis que además sea práctico.

Problemas

El problema que se quiere tratar, de entrada, es el **problema de decisión** limitado a un lenguaje formal, el cual tiene como alfabeto una secuencia de bits o caracteres en un alfabeto finito. En particular, el problema de decisión para lenguajes formales consiste en saber si una cadena forma parte de un lenguaje o no. De aquí el paso para pensar en un conjunto de problemas como un conjunto de lenguajes formales. En particular, el estudio de la complejidad se limita a aquellos lenguajes formales cuya solución puede darse bajo cierta cantidad de recursos como límite.

Asintota

Para poder resolver un ejemplar del conjunto de los problemas de decisión se considera que los recursos usados para encontrar una salida o resultado van a depender de la entrada, es decir, si la entrada usa muchos recursos

entonces el encontrar una solución tomará muchos recursos. Estos recursos se definen como una función $f(n)$, donde n es el tamaño de la entrada y la función $f(n)$, corresponde a la mayor cantidad de recursos utilizados para resolver un problema de decisión.

La medida para estas funciones se conoce como notación asintótica, la cual se basa en las operaciones de relación básicas. En complejidad, sin embargo, es usual usar otro tipo de medida para la función $f(n)$.

- Función constante, representada por $O(1)$.
- Función logarítmica, representada por $O(\log n)$.
- Función polinomial, representada por $O(n^{\alpha})$
- Función exponencial, representada por 2^n .

Modelos computacionales

Los modelos computacionales considerados en la lección, son máquinas basadas en la máquina de Turing y circuitos. Una máquina se puede ver como una computadora, la cual cuenta con la capacidad de leer entradas representadas por cadenas, realizar operaciones y regresar una salida. Por otra parte un circuito es una abstracción electrónica de los chips que permiten el paralelismo.

Para que una computadora pueda realizar operaciones necesita de un programa y un estado interno. Las operaciones hechas por un programa consisten en escritura y lectura en memoria, lo cual afecta la memoria y el estado interno de la computadora. Se dice que la cantidad de instrucciones depende del tamaño de la entrada de un programa. Usualmente los programadores devuelven una salida dependiendo del estado de aceptación o el estado de rechazo. Es así, si se obtiene o no un programa con una entrada específica a un programa, esto reduce el problema a un problema de decisión.

El modelo de los circuitos, a diferencia de las máquinas, se compone de puertos y alambres y se basa en un sistema de valores booleanos. Los puertos que puede tener un circuito depende directamente del tamaño de la entrada de puertos y un conjunto de operaciones AND, OR y NOT, además de que un circuito cumple con la propiedad de ser acíclico.

Medida de recursos

La medida de recursos de una computadora se divide en dos partes: medición de tiempo; medición de memoria. Por otro lado, la medida para los circuitos son: tamaño y profundidad.

El tiempo de ejecución de un algoritmo en una máquina es la cantidad de operaciones o cálculos que se tienen que hacer para dar un resultado o para terminar la ejecución del algoritmo.

La clase de tiempo que nos interesa es la clase P, que se refiere a aquellos problemas cuya solución toma tiempo polinomial para responder, además de la clase EXP, que son aquellos problemas que toman tiempo exponencial para dar una respuesta.

Por otro lado, el tiempo usado por un algoritmo en una computadora corresponde a la cantidad de bits usados en memoria y de aquí se desprenden tres clases de tiempo:

- DSPACE(1): Son aquellos programas decidibles en tiempo constante sin importar la entrada que pase al algoritmo. Esta clase se puede ver como un automata finito determinista.
- DSPACE(log n): Son aquellos problemas decidibles en espacio logarítmico.
- DSPACE($n^{O(1)}$): Son aquellos problemas decidibles cuya solución toma espacio polinomial. Se dice que cualquier algoritmo que no forme parte de esta clase no puede ser decidable en la práctica, ya que se necesitan más recursos en la memoria.

Para medir el rendimiento de los circuitos, como se mencionó anteriormente, se considera el tamaño y la profundidad.

- Tamaño: el tamaño de un circuito es la cantidad de puertos que contiene el circuito y a su vez, se divide en dos partes.
 - PSIZE: es la clase de lenguajes decidibles para familias de circuitos, limitados por algún polinomio.
 - SIZE: es la clase de lenguajes decidibles por familias de circuitos de n entradas es $O(f(n))$.
- Profundidad: es la longitud de la ruta más larga desde un puerto de entrada hasta el puerto de salida. La profundidad corresponde al tiempo necesario para evaluar un circuito en paralelo.
 - Clase AC⁰: profundidad constante, tamaño polinomial. Es la clase de lenguajes decidibles por familia de circuitos con profundidad constante, tamaño polinomial.
 - Clase AC¹: profundidad logarítmica, tamaño polinomial. Es la clase de lenguajes decidibles por familia de circuitos, con profundidad logarítmica, tamaño polinomial.