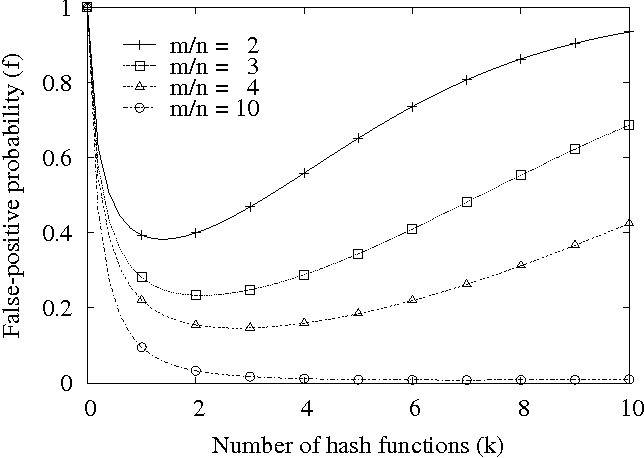
Bloom Filters

Fonseca Zárate Israel 183708

Los Bloom filters se tratan de una estructura de datos probabilística cuyo objetivo es ser rápida y compacta, esto a costa de perder un cierto porcentaje en cuanto a su certeza al definir si un elemento está insertado, es decir, puede arrojar falsos positivos. El objetivo es hacer de los Bloom filters lo más eficientes posibles, para ello se debe jugar con 3 variables.

m que corresponde al tamaño del arreglo para insertar los datos.

k el número de funciones de hash, será el número de veces que sea insertado un elemento en el arreglo.

p porcentaje o tolerancia permitida de falsos positivos.

Por un lado, a mayor m el algoritmo pierde eficacia pues se hace más pesado y lento, sin embargo, se obtiene mayor certidumbre. Inversamente, a menor m el algoritmo es más rápido y compacto, pero es menos acertado. Por otro lado, en el caso de k se debe hallar un tamaño óptimo, pues, no solo impacta en la velocidad de ejecución, también un número bajo o alto aumentan las probabilidades de colisión.

Existen muchas investigaciones respecto al tamaño adecuado y número de funciones que debe tener un Bloom filter; sin embargo, la más aceptada resulta ser un par de fórmulas que permiten obtener m y k, a partir de un número de datos esperados (n) y un porcentaje de falsos positivos, siendo:

m= -(n\*ln(p))/(ln(2)^2)

k= (m/n)\*ln(2)

Por otro lado, las funciones de hash deben ser lo más rápidas, por lo que lo más eficiente son funciones no-criptográficas, lo que los lleva a ser vulnerables a ataques. El utilizado fue MurmurHash 2.0, algoritmo que consiste en 2 operaciones básicas: multiplicar y rotar.