

Laboratorio No. 2 P2

Esquemas de detección y corrección

Enlace al repositorio: <https://github.com/bl33h/detectionAndCorrectionSchemes> [a partir del commit realizado 28/07/2024 [No. 11]]

Descripción.

Usando las implementaciones de la parte 1, se busca crear un script o programa que funcione como un sender y receiver mediante el uso de sockets. Se mantiene el uso de distintos lenguajes de programación entre cada script o programa.

Resultados.

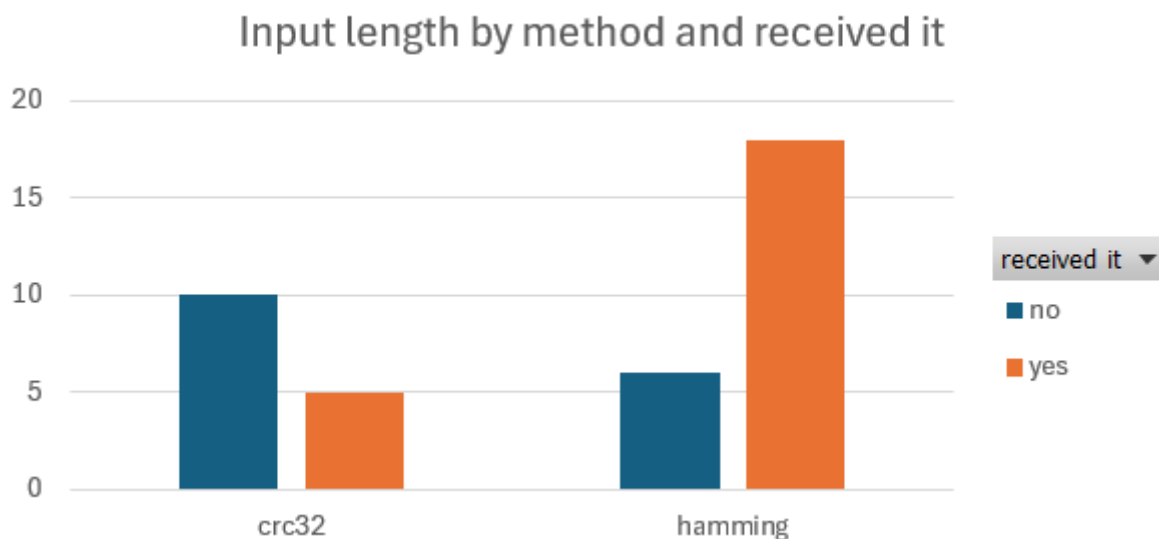


Gráfico No. 1: Longitud de palabras vs se recibieron correctamente por algoritmo

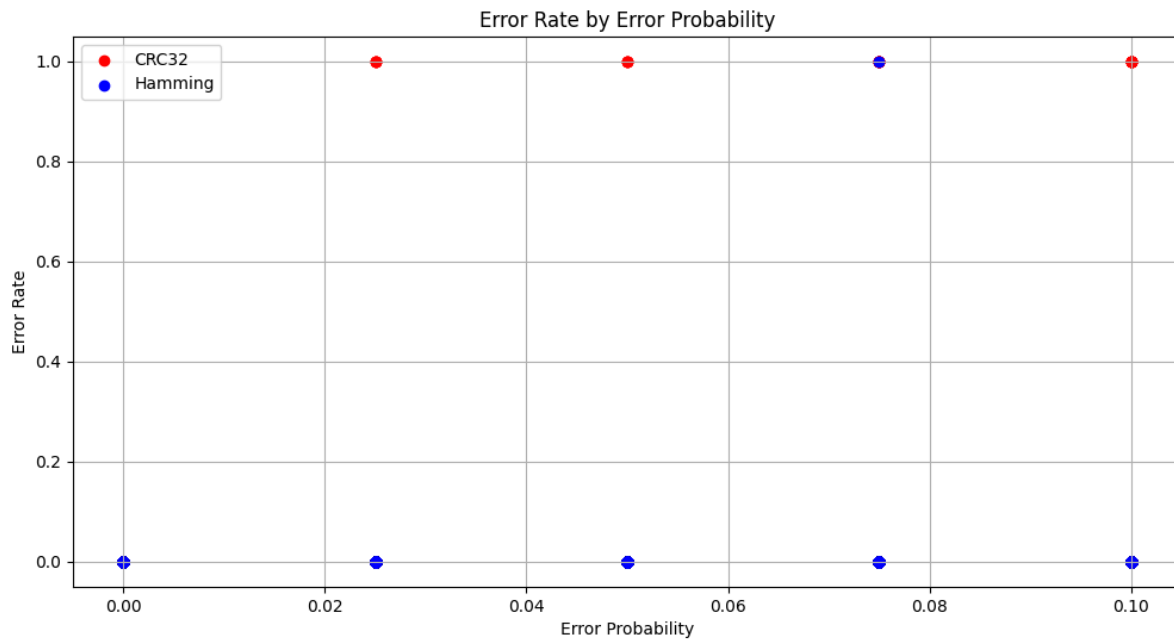


Gráfico No. 2: Tasa de error por algoritmo

Discusión

En general, el algoritmo que mejor funciona es Hamming. Por definición, siempre que el mensaje se entregue completo y/o que la paridad coincida (para errores de más de 1 bit), el algoritmo no fallará, tal como se muestra en el *Gráfico No. 1*. Por otro lado, el algoritmo CRC32 puede fallar con casos muy específicos donde el polinomio no se ajuste en su totalidad al mensaje formado con el checksum. Por ejemplo, en el caso de mandar mensajes codificados en ascii con CRC32, varias combinaciones de 2 letras donde la primera es 'o' fallan. Esto probablemente se deba a que el segundo byte de la letra 'o' (1111) es idéntico al inicio del polinomio que se está usando ($x^{32} + x^{31} + x^{30} + x^{29} \dots + 1$), la probabilidad de error va de la mano como se manifiesta en el *Gráfico No. 2*.

Por esto siempre es mejor emplear Hamming, pues asegura que el mensaje llegue de forma correcta o con algunos fallos detectables con los bits de paridad. El proceso también es menor al no realizar varias operaciones XOR y solo verificar la paridad.

La situación donde CRC32 pueda ser mejor que Hamming es cuando la red es estable, preferiblemente privada, y se busca un ambiente barato.

Comentario.

Esta actividad en general se sintió mucho mejor que las anteriores pues se implementó código. Los algoritmos llegaron a ser un desafío en algunos casos pero fue llevadero.

Conclusiones.

- El algoritmo de Hamming es el más seguro y eficiente en la mayoría de casos.
- El algoritmo de CRC32 es óptimo para retransmisiones factibles con formatos de mensaje establecidos.