Reporte de Doctorado

Avances sobre investigación sobre Spread Spectrum

Alfredo Ortega
Instituto Tecnológico de Buenos Aires

Febrero 2008

Resumen

En este artículo se documentan los avances sobre la investigación sobre spread-spectrum como parte del Doctorado en Informática. El foco de la tesis es implementar un esquema criptograficamente seguro a nivel de link (OSI Capa 2) en enlaces de fibra óptica.

Índice

1. Introducción

En el 2007 durantes los cursos del doctorado, se comenzó a explorar la posibilidad de implementar un esquema de spread spectrum sobre fibra óptica. Spread Spectrum es un técnica utilizada exitosamente en muchos protocolos de comunicaciones modernas, tales como Bluetooth, Wifi, etc. Básicamente se expande el espectro de la señal a transmitir utilizando una segunda señal conocida. La modificación puede ser en tiempo, frecuencia, o modulación directa. Esto conlleva una disminución de la eficiencia de la transmisión, pero una señal con espectro expandido tiene muchas otras ventajas, como por ejemplo la coexistencia con otra señal (o ruido) de banda estrecha.

2. Ruido en canal simetrico binario

Para calcular el ruido en un canal simétrico binario, calculamos la probabilidad de no-colision que tendra un usuario determinado, ya que las colisiones seran el ruido del canal (En esta etapa no consideramos otros tipos de ruido que pueda tener el canal físico).
Cantidad de slots por trama: m Cantidad de usuarios: n

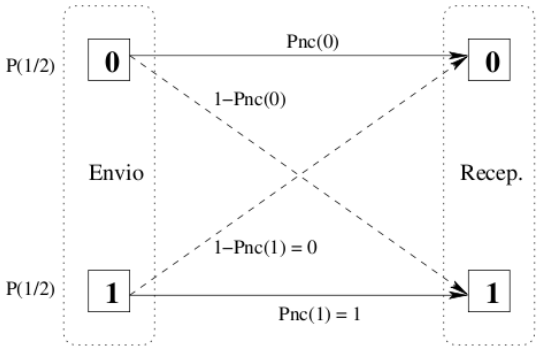

Figura 1: Canal binario: esquema de probabilidad

Probabilidad de no colisión para un usuario en un canal simétrico:

$$P_{nc} = \left(\frac{m-1}{m}\right)^{n-1} \tag{1}$$

Probabilidad de no colisión para un usuario en un canal óptico:

$$P_{nc} = P(1)\cdot P_{nc}(1) + P(0)\cdot P_{nc}(0) \tag{2}$$

$$P_{nc} = \frac{1}{2}\cdot 1 + \frac{1}{2}\cdot \sum_{i=0}^{n-1} C_i^{n-1} \left(\frac{m-1}{m}\right)^i \left(\frac{1}{m}\right)^{n-1-i} \left(\frac{1}{2}\right)^{n-1-i} \tag{3}$$

Donde $\left(\frac{m-1}{m}\right)^i$ es la probabilidad de no colisión de i canales (se suma para todo posible número de canales no colisionando: $1 \leq i \leq n$, que están en otro slot), $\left(\frac{1}{m}\right)^{n-1-i}$ es la probilidad de colisión de los restantes $n-1-i$ (estos están en el mismo slot que el canal actual, el '-1' es para no contar el canal actual), y la colisión se produce cuando los otros canales transmiten 1 cuya probabilidad es $\left(\frac{1}{2}\right)^{n-1-i}$. El factor C_i^{n-1} suma sobre todas las combinaciones posibles de canales no colisionando, que son hechos independientes.
Teniendo en cuenta que $\sum_{i=0}^{n-1} C_i^{n-1} \left(\frac{m-1}{m}\right)^i \left(\frac{1}{2m}\right)^{n-1-i}$ es la potencia $n-1$ de un binomio, reemplanzando tenemos

$$P_{nc} = \frac{1}{2} + \frac{1}{2}\cdot \left(\frac{m-1}{m} + \frac{1}{2m}\right)^{n-1} \tag{4}$$

$$P_{nc} = \frac{1}{2} + \frac{1}{2}\cdot \left(1 - \frac{1}{2m}\right)^{n-1} \tag{5}$$

$$P_{nc} \simeq \frac{1}{2} + \frac{1}{2}\cdot e^{-1/2} \tag{6}$$

Donde la última aproximación vale para $n = m$ y n grande.

Para el caso de *bloom* filters con k filtros[1] la probabilidad de no colisión es:

$$P_{nc}^{k} = P(1) \cdot P_{nc}^{k}(1) + P(0) \cdot P_{nc}^{k}(0) \tag{7}$$
$$\tag{8}$$

Sabiendo que la probabilidad de no colisión para el 0 es:

$$P_{nc}^{k}(0) = 1 - \left(P_{c^k}(0)\right)^k \tag{9}$$

Pero la probabilidad de colisión para el 0 cuando se transmiten k copias es:

$$P_{c^k}(0) = 1 - \left(P_{nc^k}(0)\right) \ , \tag{10}$$

y que además la probabilidad de no colisión para los k slots del bloom filter es

$$\begin{aligned} P(\text{no col.}k) &= P(\text{no col.}1) \cdot P(\text{no col.}2) \cdot P(\text{no col.}3) \cdots P(\text{no col.}k) && (11)\\ &= \left(\frac{m-1}{m}\right) \cdot \left(\frac{m-2}{m-1}\right) \cdot \left(\frac{m-3}{m-2}\right) \cdots \left(\frac{m-k}{m-(k-1)}\right) && (12)\\ &= \frac{m-k}{m} \ . && (13) \end{aligned}$$

Luego la probabilidad de colisión con alguno de las k copias del bit es

$$\begin{aligned} P(\text{col.}k) &= 1 - P(\text{no col.}k) && (14)\\ &= 1 - \frac{m-k}{m} && (15)\\ &= \frac{k}{m} \ . && (16) \end{aligned}$$

Entonces reemplazamos y calculamos:

$$\begin{aligned} P_{c^k}(0) &= 1 - \left(\sum_{i=0}^{n-1} C_i^{n-1} \left(\frac{m-k}{m}\right)^i \left(\frac{k}{2m}\right)^{n-1-i}\right) && (17)\\ &= 1 - \left(1 - \frac{k}{2m}\right)^{n-1} && (18) \end{aligned}$$

[1] Se envían k repeticiones del bit en canales distintos, entonces basta que sólo uno de ellos sea 0 para que recibamos un 0 en un canal óptico.

Reemplazando esta ecuación en 8 obtenemos:

$$P_{nc}^{k} = \frac{1}{2} + \frac{1}{2}\left(1 - \left(1 - \left(1 - \frac{k}{2m}\right)^{n-1}\right)^{k}\right) \tag{19}$$

Sin embargo, este calculo es incorrecto, comparandolo con los datos que da el simulador. La formula entrega valores de error menores con respecto a los reales, como se observa en la figura. Los trazos del mismo color corresponden a el mismo K con azul(K=1), verde (K=2) y rojo(K=4). M=256

3. Entropia

Comenzemos por lo básico:

Segun Shannon, el **contenido de informacion** h(x) de un suceso x dada la posibilidad que suceda P(x) es:

$$h(x) = log_2\left(\frac{1}{P(x)}\right)$$

Y la entropia de un conjunto A, H(A) se define simplemente como el promedio del contenido de información:

$$H(A) = \sum_{xEA_x} P(x) log_2\left(\frac{1}{P(x)}\right)$$

En un canal binario solo dos sucesos existen, uno con probabilidad p, y otro con probabilidad 1-p, por lo tanto para p siendo la probabilidad de error:

$$H(p) = -plog_2(p) - (1-p)log_2(1-p)$$

4. Entropia condicional

Vamos a analizar la entropia de dos conjuntos X de entrada y Y de salida interrelacionados.

La entropia condicional de X dado $y = b_k$ donde b_k es un valor dado, es la entropia de la distribucion de probabilidad $P(x|y = b_k)$:

$$H(X|y = b_k) = \sum_{x \in A_x} P(x|y = b_k) \log_2\left(\frac{1}{P(x|y = b_k)}\right)$$

La entropia codicional de X dado Y es el promedio, sobre y, de la entropia condicional de X dado y:

$$H(X|Y) = \sum_{xy \in A_x A_y} P(x,y) \log_2 \left(\frac{1}{P(x,y)} \right)$$

5. Información mútua

La información mútua entre X e Y es:

$$I(X;Y) = H(X) - H(X|Y)$$

Mide el promedio de reduccion de la incertidumbre acerca de x que resulta de saber el valor de y, o viceversa: la cantidad promedio de informacion que x revela acerca de y.

6. Capacidad de canal

La capacidad C de un canal discreto sin memoria es :

$$C = \max_{\mathcal{P}_x} I(X;Y) \tag{20}$$

O sea, la máxima información mutua entre los alfabetos X de entrada e Y de salida. Para hallar el maximo podemos derivar $I(X;Y)$ con respecto a la probabilidad P_x. Para un canal binario asimétrico sin memoria con probabilidad de error p, la capacidad máxima C es:

$$C = 1 - H(p) \tag{21}$$

Si expandimos H(p) en 21:

$$c = 1 - \left(p \times \log_2 \left(\frac{1}{p} \right) + (1-p) \cdot \log_2 \left(\frac{1}{1-p} \right) \right)$$

Simplificada:

$$c = 1 + p * \log_2(p) + (1-p) * \log_2(1-p)$$

Sin embargo esta capacidad es menor que la que realmente tenemos en nuestro canal, ya que un Z-channel se adecua mayormente a los medios de transmision opticos.

7. Z-channel

Un canal Z (Z-channel) difiere de un canal binario, ya que las probabilidades de bit-flip son asimetricas. Los Z-channel se usan generalmente para modelar sistemas de transmision opticos.

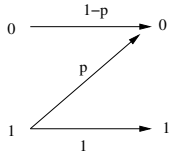

Figura 2: Diagrama: Z-channel

Para un Z-channel, la distribucion de probabilidades de I(X;Y) es diferente, por lo que obtenemos un máximo diferente:

$$C_Z = 1 - \left(\frac{1}{2} * H(p)\right)$$

[2]

Por lo tanto,

$$C_Z = \log_2\left(1 + (1-p)p^{p/(1-p)}\right)$$

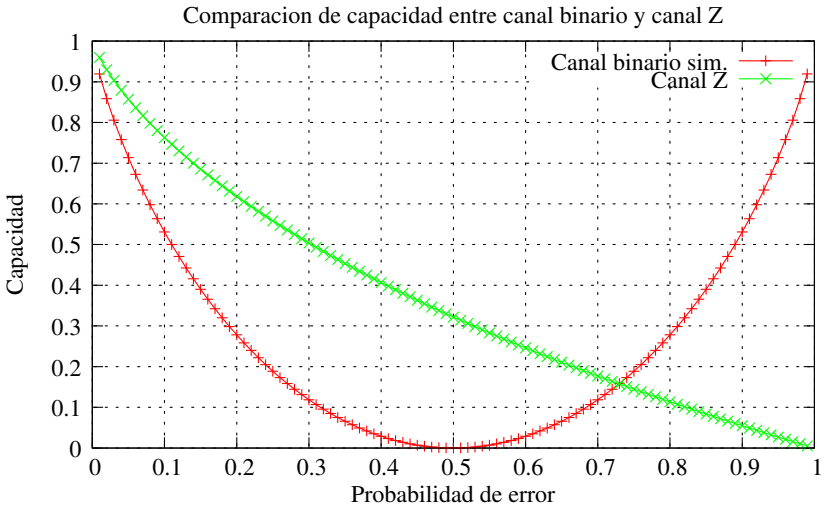

Figura 3: Diagrama: Azul: Capacidad de canal binario Verde: Capacidad de canal Z

8. Parámetro de seguridad

(De Cryptograpy by V. V. IAshchenko)[1]

Primero, para el análisis de complejidad de un sistema criptográfico se suele utilizar un parametro variable que mide el tamaño del problema y representa a la vez los requerimientos del algoritmo criptográfico tanto como la probabilidad de un adversario de romper la seguridad en el sistema, Este es el llamado parametro de seguridad, por ejemplo esto puede ser el largo de clave. Este parámetro puede tomar valores arbitrariamente grandes. Segundo, la definición de seguridad depende de la tarea que el adversario trata de realizar y de la informacion acerca del esquema criptográfico disponible. Tercero, se debe especificar el valor en el cual la cantidad de calculos necesarios por el atacante se presumen irrealizables, esto sera funcion del parametro de seguridad. Segun la thesis de Edmond, el algoritmo del atacante se considera eficiente si esta limitado en tiempo polinomial sobra la longitud de la entrada (el parametro de seguridad en este caso) de otra manera se considera infeasible. Notese que el algoritmo criptografico mismo debe ser eficiente.

Finalmente se debe fijar un limite para la probabilidad negligible. El contrato criptografico estandard dice trata a la probabilidad como negligible si no excede $\frac{1}{p(n)}$ para un polinomio p y el parametro de seguridad n.

Aceptadas esas cuatro definiciones, para probar que un algoritmo criptografico es seguro basta con probar que la no-existencia de un algoritmo polinomial que realize la tarea del adversario.

De todas maneras, el estado actual de la teoria de complejidad no permite justificar un limite inferior de un problema como super-polinomial ($P = NP$?) por lo que todas las pruebas en el mundo de la seguridad estan basadas en asumciones no probadas. Por lo tanto, la investigacion se concentra usualmente en buscar las condiciones suficientes mas debiles (o necesarios y suficientes) para la existencia de un esquema seguro. Las asumciones son usualmente generales (Basadas en la teoria de complejidad) o basadas en intratabilidad de problemas en la teoria de numeros, etc.

8.1. algoritmo1

Se estudiara el parámetro de seguridad para el algoritmo 1:

Este algoritmo consta de r clientes cada uno con una serie de n posiciones pre-seleccionados. La cantidad posible de combinaciones de n posiciones es de ${}_nP_r = \frac{n!}{(n-r)!}$. Definimos que el algoritmo se rompe cuando un atacante puede inferir la serie de posiciones n para un cliente.

Existe un algorimto de selección de n que suponemos no posee ninguna debilidad, o sea, elije r conjuntos de n posiciones tal que teniendo una, no

se pueda inferir otra. Un algoritmo sencillo que cumple estas características es una simple búsqueda exaustiva. No es óptima en tiempo pero cumple con las características de seguridad requeridas.

Suponemos:

- El algoritmo de selección no tiene debilidades.
- El atacante no posee control de los datos a transmitir, y estos son totalmente randomizados.

Dadas estas suposiciones (Equivalen a un ataque con plain-text desconocido en jerga criptográfica) para inferir el conjunto de posiciones n de un cliente, el atacante debera probar exaustivamente todo el conjunto de ${}_nP_r$ sobre una trama capturada, y ese sera el parámetro de seguridad, hasta que demostremos lo contrario.

9. mediciones

A continuación expondremos algunos graficos resultados de las simulaciones. La figura **??** muestra la capacidad teórica máxima de un canal binario frente a otro canal Z, con respecto aumentan los clientes en un frame de 256 clientes.

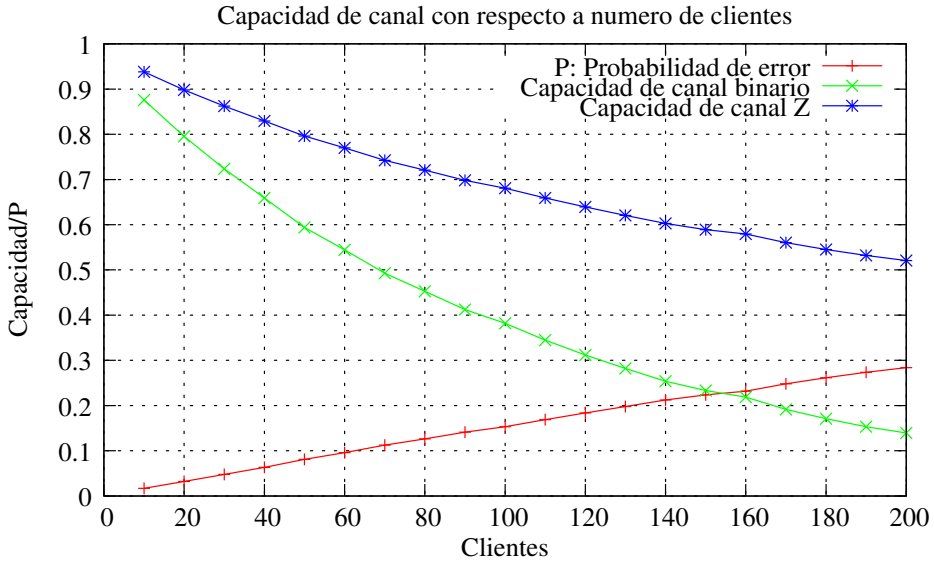

Figura 4: Comparacion de capacidad: Binario (Verde) Z-channel (Rojo)

Sobre este sistema, se corrieron varias simulaciones, y la figura **??** muestra dos lineas, que representan el error de algoritmos LDPC+Reed-Solomon y Reed-Solomon puro. El punto en el que las curvas se elevan de cero, marca la capacidad maxima del algoritmo, que puede contrastarse con la capacidad maxima teórica.

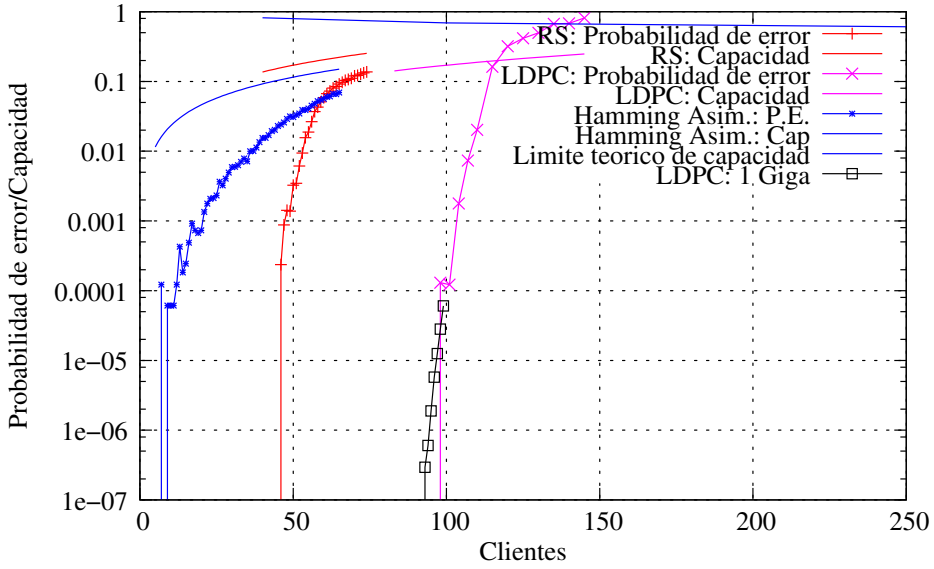

Figura 5: Simulaciones: LDPC y Reed-Solomon contra capacidad teórica

Referencias

[1] Charles Louis Xavier Joseph de la Vallée Poussin, 1879. A strong form of the prime number theorem, 19th century.

[2] Al-Bassam Tallini, 1984. On the capacity and codes for the Z-channel.