Universidad de Buenos Aires

Facultad de Ciencias Exactas y Naturales

Apunte Redes

Damián Aleman

July 26, 2016

Contents

1 Teoria de la Información (Claude Shannon)

Dos Teoremas Fundacionales:

1. Codificación para una fuente sin ruido
2. Codificación para un canal ruidoso: Describe la máxima eficiencia posible de un método de corrección de errores (codificación) frente a los niveles de ruido y de corrupción de los datos.Es decir, brinda un límite para la transmisión de bits. No dice nada sobre como implementar dicha codificación.

Información: Sea E un suceso que puede prensetarse con probabilidad P(E). Cuando E tiene lugar decimos que hemos recibido $I(E) = log1/P(E)$ unidades de información.

Notar que si $P(1/2), I(E) = 1$ bit. Es decir, un bit es la cantidad de informacíon obtenida al especificar una de dos posibles alternativas igualmente probables.

Si tenemos una fuente emitiendo una secuencia de símbolos pertenecientes a un alfabeto finito y fijo, $S = s_1, s_2, ..., s_q$. Los símbolos emitidos sucesivamente se eligen de acuerdo con una ley fija de probabilidad. En la fuente más sencilla admiteiremos que los símbolos emitidos son estadísticamente independientes. Llamaremos a la fuente de información fuente de memoria nula y puede describirse complemtamente mediante el alfabeto fuente S y las probabilidades con que los símbolos se presentan.

Llamaremos a una fuente de memoria nula, una fuente que emite una secuencia de símbolos pertenecientes a un alfabeto finito y fijo, S = s1,s2,...,sq con una ley fija de probabilidad y donde los símbolos emitidos son estadísticamente independientes. La fuente de memoria nule puede describirse complemtamente mediante el alfabeto fuente S y las probabilidades con que los símbolos se presentan $P(s_1), P(s_2), ..., P(s_q)$

Puede calcularese la información media suministrada por una fuente de información nula en la forma siguiente:
La presencia de un símbolo si correspondiente a una cantidad de información igual a $I(si) = log1/P(si)$ bits.

Entropía H(s) de la fuente de memoria nula:
La probabilidad de que aparezca un símbolo si es precisamente P(si), de modo que la cantidad media de información por símbolo de la fuente es:

$\sum_S P(s_i)I(si)bits = \sum_S P(s_i)log(1/P(s_i))$ bits

Si tenemos una fuente que emite n mensajes s_i, la entropía es $\sum_{i=1}^{n} P(s_i)log(1/P(s_i))$

Interpretaciones de la entropía: El valor medio ponderado de la cantidad de información del conjunto de mensajes posibles. Una medida de la incertidimbre promedio acerca de una variable aleatoria La cantidad de información obtenida al observar la aparición de cada nuevo símbolo.

Propiedades de la entropía: a) La entropía es no negativa y se anula si y sólo si un estado de la variable es igual a 1 y el resto 0. b) La entropía es máxima (mayor incertidumbre del mensaje) cuando todos los valores posibles de la variable s son equiprobables . Si hay n estados equiprobables, entonces $p_i = 1/n$ Luego: $H(S) = -\sum p_i log_2 p_i = -n(1/n)log_2(1/n) == -(log_2(1) - log_2(n)) = log_2(n) = H(S)_max$

Una fuente se puede extender mediante la extensión del alfabeto con una sucesión de los símbolos del alfabeto inicial. Tenemos que $H(S \exp n) = nH(S)$

Modelo de un Sistema de comunicaciones: Canal sometido a ruido, limitado en potencia y en ancho de banda.

Perturbaciones en la transmisión La señal recibida puede diferir de la señal transmitida Analógico - degradación de la calidad de la señal Digital - Errores de bits Causado por: -Atenuación y distorsión de atenuación -Distorsión de retardo -Ruido

Atenuación: La intenisdad de la señal disminuye con la distancia. Depende del medio La intenisdad de la señal recibida: Debe ser suficiente para que se detecte Debe ser suficientemente mayor que el ruido para que se reciba sin error Crece con la frecuencia Ecualización: Amplificar más las frecuencias más altas.

Distorsión de retardo: Solo en medios guidos. La velocidad de propagación en el medio varíá con la frecuencia. Para una señal limitada en banda, la velocidad es mayor cerca de la frecuencia central. Las componentes de frecuencia llegan al receptor en distintios instantes de tiempo, originando desplazamientos de fase entre las distintas frecuencias.

Ruido: Señales adicionales insertadas entre el transmisor y el receptor: Térmico: Debido a la agitación térmica de los electrones Intemodulación: Se produce por falta de linealidad en el canal Diafonía: una senal de uan línea interfiere en otra Impulsivo: Implusos irregulares o picos

Para un cierto nivel de ruido, a mayor velocidad C: menor período de un bit mayor tasa de error (se pueden corromper 2 bits en el tiempo en que antes se corrompía 1 bit)

En principio, si se aumenta el ancho de banda B y la potencia de señal S, aumenta la velocidad binaria C. Pero: Un aumento del ancho de banda B aumenta el ruido Un aumento de potencia de señal S aumenta las no linealidades y el ruido de intermodulación Según Shannon, la velocidad binaria

teórica máxima será para un canal será: C máx (bps) = B (Hz)·log 2 (1 + SNR)

Nyquist: $C(bps) = Vlog_2M = 2Blog_2M = Blog_2M^2$

Luego, no se podrá aumentar M tanto como se quiera: $M \leq \sqrt{(1+SNR)}$

Teorica 2 La condicion necesaria y suficiente para la existencia de un código instantaneo de longitudes l1,l2, .., lq es que $\sum i = 1, qr \exp -li \leq 1$ O cumpliendo la condicón de los prefijos: No exista palabra que sea prefijo de otra palabra de longitud mayor.

Llamamos codificación al establecimiento de una correspondencia entre los símbolos de una fuente y los símbolos del alfabeto de un código. EN la codificación intentaremos lograr una representacón eficiente de la información (eliminando la redundancia).

Un código eficiente asigna las palabras mas cortas a símbolos más probables

Longitud media de un código: $L = \sum p_i l_i$ $Llog(r) \geq H(s)$ donde li e sla longitud de la palabra codificada del mensaje mi r es la cantidad de simbolos del alfabeto del codigo

log r : Cantidad promedio máxima de información de un símbolo del código $h = H(S)/(Llogr)$ Eficiencia del código

Codificador óptimo es aquel que para codificar un mensaje X usa el menor número posible de bits. $H(X) = \sum p(x) log2[1/p(x)]$ EL logaritmo de 1/p(x) representa el número de bits necesario para codificar el mensaje X en un codificador óptimo.

2 Nivel Físico

2.1 Introducción y Fundamentos

Hay dos Modelos fundamentales de arquitecturas de redes. Estos se basan en varias capas que son niveles de abstracción que brindan servicios a las capas superiores.

Se impuso la arquitecturas TCP/IP sobre el modelo OSI: [1]

Diagrama de sistema de comunicaciones: El mensaje se envía por un medio que sufre de ruido. Luego el receptor obtiene un mensaje que contiene variaciones.

Fundamentos de las Señales SOn ondas electromagnetivas que se propagan a través de un medio: En el vacío a la velocidad de la luz (c) En otros medios a una velocidad menor, tomada como factor de c

La onda electomagnetica es un campo electrico y magentico que se propaga por un medio a una velocidad que depende de este. La onda vibra a una frecuencia determinada, con un comportamiento periodico en el eje

[1] http://spectrum.ieee.org/computing/networks/osi-the-internet-that- wasnt

longitudinal de su propagacion. EL periodo se denomida longitud de onda y se define como l=v/f (con c la velocidad y f la frecuencia de oscilacion)

Problemas: La onda, en el caso de chocar con imperfecciones produce reflexiones. Además si el medio tiene perdidas se puede atenuar (generalmente se atenuan propocianalemte a las distancias recorridas)

Longitud de onda: es la distancia espacial entre dos puntos correspondientes a la misma fase en dos ciclos consecutivos.

Ancho de banda: Frecuencia de corte: frecuencia donde se produce una atenuacion de 3dB

Series de Fourier: TOdas las funciones periodiasse pueden expresar como sumas de senos y cosenos.

$f(t) = 1/2a_0 + \sum_{i=1}^{\infty}[a_n cos(n\omega_0 t) + b_n sen(n\omega_0 t)]$

Una onda cuadrada se puede representar como una serie infinira de sonoides armonicamente relacionados.

Existen varios medios de transmisión

- Por guia de onda:
  - Par trenzado de cobre
  - Coaxial
  - Red electrica
  - Fibra optica
- Sin guia de onda (El espectro electromagnetico):
  - Transmision por radio
  - Transmision por microondas
  - Transmision por ondas infrarojas
  - Transmison por laser

Veamos la estructura del sistema telefónico: Objetivo: Transmitir la voz humana en una forma más o menos reconocible Componentes:

- Local loops (pares trenzados, señalización analógica)
- Troncales (fibra óptica o microondas, digital)
- Oficinas de conmutación

Debido a consideraciones económicas, las compañías telefónicas han desarrollado políticas elaboradas para multiplexar varias conversaciones sobre un único troncal físico. TDM (Time Division Multiplexing) Los usuarios toman turnos (en "round robin") obteniendo periódicamente cada uno el ancho de banda completo por un período de tiempo acotado

FDM (Frequency Division Multiplexing) El espectro de frecuencias es subdividido en canales de ancho de banda acotado, que es usado a tiempo completo y exclusivo por cada usuario. Ejemplo de FDM y TDM: la radio AM

Aunque FDM se utiliza todavía sobre cables de cobre o canales de microondas, requiere circuitería analógica no trivial. En contraste TDM puede ser manejado enteramente por electrónica digital, y se ha vuelto de más amplio uso en años recientes. TDM solo puede ser utilizado para datos digitales

En telefonía: Como el "local loop" produce señales analógicas, es necesario realizar una conversión analógico/digital en la "end office", donde todos los "local loops" individuales se combinan sobre los "trunks" (troncales).

En los medios ópticos se puede incrementar la capacidad transmitiendo diversas longitudes de onda por una única fibra (tipo especial de FDM) Se llama WDM, multiplexacion por division de longitud de onda

2.2 Taxonomía de las redes

Redes de conmutación de circuitos: FDM/TDM

Redes de conmutación de paquetes: Redes de Circuitos Virtuales (VC) Brindan un servicio orientado a conexión (ej. X.25, ATM, etc). Redes de Datagramas (Internet) Brinda un servicio sin conexión Sin embargo el Nivel de Transporte brinda tanto servicios orientados a conexión (TCP) como servicios sin conexión (UDP) Lo que antes era una "conexión física" es ahora una "conexión lógica" (o "sesión")

La ventaja de las redes de conmutación de paquetes es que tienen una división del tiempo, bajo demanda. Es decir se comparten los enlaces encolando los paquetes que compiten por el enlace cuando el mismo no esta diponible.

El Teorema de Muestreo formulado por Nyquist (1924) dice: Si queremos reconstruir una señal de componente frecuencial máxima fm debemos muestrearla según f s ¿ 2 * fm llamada frecuencia de sampling (también de "muestreo", o de "modulación")

COnversion Analogica-Digital: Se muetra al doble de ancho de banda, obteniendo un tren de pulsos de amplitud variable Se cuantifican las muestras aproximándolas mediante un número entero de n bits Aparece el error de cuantificación

Canal PCM (Pulse Code Modulation) Las señales analógicas son digitalizadas por un dispositivo llamado CODEC (COder-DECoder), produciendo símbolos de 8 bits por muestra (en realidad uno es para señalización). El CODEC toma 8000 muestras por segundo (125 μseg/muestra) debido a que el teorema de Nyquist establece que esto es suficiente para capturar toda la información "relevante" de un canal telefónico de 4 KHz de ancho de banda Luego, "Ancho de banda del cada canal de voz" = 64 Kbps. Como consecuencia, virtualmente todos los intervalos de tiempo en el sistema telefónico

son múltiplos de 125 μseg.

Modem convierte de digital a analogico y viceversa COdec concierte de analogico a digital y viceversa

Modulación Proceso de variación de cierta característica de una señal sin mensaje, llamada portadora, de acuerdo con una señal mensaje, llamada moduladora Tipos

Moduladora Analógica/Portadora Analógica: modular por amplitud o por frecuencia

Moduladora Digital/Portadora Analógica: caso conocido la red telefónica Tecnicas: Desplazamiento de Amplitud (ASK): los valores binarios se representan mediante dos amplitudes diferentes de la portadora Desplazamiento de frecuencia (FSK) los valores bianrios se representan mediante dos frecuencias diferentes de la portadora Desplazamiento de Fase(PSK) los valores bianrios se representan mediante dos fases diferentes de la portadora Mixtas: Modulación Multinivel Se consigue una utilización más eficaz del ancho de banda si cada elemento de la señal transmitida representa más de un bit.

Moduladora Analógica/Portadora Digital: Proceso llamado digitalización Dispositivo: codec

Métodos Modulación por impuloss codificados(MIC, o PCM) Modulación Delta: codifica solo las diferencias

Moduladora Digital/Portadora Digital Los datos binarios se transmiten codificando cada bit de datos en cada elemento de señal

NRZ No retorno a cero (NRZ) Consiste en utilizar una tensión negativapara representar un 0 y una positiva para representar un 1 Inconvenientes: para secuencias largas sin cambios se pierde el sincronismo (problemas de "clock recovery")

NRZI No retorno a cero con inversión de unos (NRZI) Los datos se codifican mediante la presencia o ausencia de una transición al principio del intervalo de un 1 Soluciona el problema de muchos 1 consecutivos, pero no el de muchos 0 consecutivos.

Manchester: Se codifica mediante una transición en la mitad del intervalo de duración del bit: de bajo a alto representa un 1 y de alto a bajo un 0 Baud Rate = 2 * Bit Rate

Manchester Diferencial Bifase Diferencial (Manchester Diferencial) La codificación de un 0 se representa por la presencia de una transición al principio del intervalo del bit y un 1 mediante la ausencia de transición.

Alternativa para mayor eficiencia: Reemplazar secuencias de varios bits iguales (que dan lugar a niveles de tensión constante) por otra que proporcione transiciones para que emisor y receptor estén fielmente sincronizados ("preservar el clock") El receptor debe identificar la secuencia reemplazada y sustituirla por la original. Ejemplo: 4B/5B

La Velocidad de Modulación se define como el número de cambios de señal por unidad de tiempo, y se expresa en baudios (símbolos/segundo) La

Velocidad de Transmisión equivale a la velocidad de modulación multiplicado por el número de bits representados por cada símbolo, expresada en bits/segundo: V t = V m . N

El precio a pagar en las modulaciones de orden superior por la mejora en la Velocidad de Transmisión es una mayor Tasa de Errores.

3 Nivel de Enlace

Enlaces Punto a Punto Tenemos un "caño" serial (no hay desordenamiento) Pero: sujeto a ruido y fallas Lo que se recibe puede no ser lo que se envió: "error de transmisión" Objetivos: Proveer servicio a la capa superior Confiabilidad. ¿Confiable o no confiable? Control de Errores. ¿Se produjo algún error? ¿Qué hacemos con los errores? Control de Flujo. Más adelante: en Nivel de Transporte.

Estrategia Encapsulamiento o "Framing" Encapsular los bits de Mensaje en Frames agregando información de control

Framing: ¿Cómo se separan los frames en un tren de bits? Largo fijo Largo especificado en el encabezado Delimitadores de frame (con bit-stuffing)

Tipo de Servicio Sin conexion y sin reconocimiento Sin conexion y con reconocimiento Orientado a conexion

Detección y Corrección de errores Redundancia: Definiendo: d la mínima Distancia de Hamming entre todas las codewords de un código. e la cantidad de bits erróneos en una transmisión dada Necesitamos: m bits (datos) + r bits (redundancia) = n bits (codeword) $e + 1 \leq d$ para poder detectar $2e + 1 \leq d$ para poder corregir Para la confiabilidad Surge la nececidad de poder efectuar retransmisiones Implícitas (cuando ocurre un time-out se asume que el dato se perdió) Explícitas (mensajes de control específicos para pedir repetición de envío de datos)

Transmision Confiable: Problema de los dose genrales COnsecuencia: No existe un algoritmo para la confiablididad Enviamos un unico mensaje de reconocimiento (ACK)

Stop And Wait: Cada frame debe ser reconocido por el receptor Problema de las renncarnaciones: Numero de Secuencia Existe un tiempo de bloqueo a la espera de confirmaciones

Para aumentar la eficiencia, es decir estar bloqueado lo menos posible se creo la tecnica de Sliding Window

Idea: LLenar el canal enviando el producto delay x Vtx

Ventana de Emision: $SWS = V_{Tx}RTT/|Frame|$

Enviar segun: $UltimoFrameEnviado \leq UltimoFrameReconocido + SWS$

ACks acumulativos: RWS =1 Acks selectivos: RWS = SWS. Se informa los frames que llegaron incluso si no es el frame esperado.

Y para distinguir reencarnaciones: #frames unívocamente identificables $\geq SWS + RWS$

Delay(retardo total) Delay = Tprop + Ttx + Tencol +Tproc T_{prop} = retardo de propagacion. Depende de la distancia entre los hosts. T_{tx} = Retardo de Transmision = Tamaño Trama / Velocidad de transmisión Significativo para enlaces de baja velocidad (o tramas muy grandes) Tproc: Tiempo requerido en analizar el encabezado y decidir a dónde enviar el paquete T_{encol}: Tiempo en que el paquete espera en un búfer hasta ser transmitido

Protocolos de Acceso Multiple - Medios Compartidos: Vimos que podíamos "compartir" un medio de transmisión guiado o no guiada mediante TDM FDM WDM Otras formas de compartir: Contención estadística "Los sistemas en los cuales varios usuarios comparten un canal común de modo tal que puede dar pie a conflictos se conocen como sistemas de contención"

El problema del acceso a un canal es hay múltiples odos compartiendo un medio, donde la simultaneidad no es posible.

Protocolos de acceso multiple: Objetivo: maximizar el numero de comunicaciones exitosas, en promedio Asegurar average fairness (igualdad de oportunidades en promedio) entre todos los nodos

Se requiere un control descentralizado. CSMA-CD Si está libre, tranmite Si esta ocupado: 1-persistente: espera a que se libere y tranmite p-persistente espera a que se librer y transimite con probabilidad p La logica de recepcion esta establecida en el sensado para detectar colisiones

Es necesario tener un control sobre los envios, para sabir si llegaron sin colisionar Largo minimo de trama: Se envia hasta saber que no hubo colision. MEcanismo de Exponential Back-Off Elegir un slot entre 0 y $2 \exp k - 1$, con k la cantidad de intentos. Esperar slot veces el RTT antes de sensar para retransmitir.

Red de area local: conjunto de estacione que comparten dominio de broadcast Debido a que las Lans pueden ser de varios tipos de tecnologias, las estacioens deben compartit esquemas de direccionamiento.

Para que pueda escalar, los switchs aprenden por que interfaces deben mandarse los mensajes en funcion del trafico de la Lan.

Para que no hayan ciclos en una red (que generan problemas en especial cuando se hace broadcast) se ejecuta un algoritmos para eliminarlos: el Spanning Tree Protocol. Idea: Cada switch encia paquetes (BDPU) a sus vencinos propagando informacion acerca de la topologia de la Lan de manera periodica.

Para lograr que las Lans escalen(el broadcast no escala), existe el enfoque de LANs virtuales, quienes permiten particionar a una LAN en varias LANs diferentes e interconectarlas.

4 Nivel de Rede -Teorica 4

Un switch de datos es un dispositivo con multiples entradas y multiples salidas. Interconecta enlaces para fomrar redes más grandes.El switch permite construir redes escalables.

Dos grandes paradigmas Orientado a conexión vs Sin conexión

4.1 Circuitos Virtuales

Se requiere una fase para establecer una conexión y otra de finalización de la conexión Los paquetes o celdas que se transmiten después de establecer la conexión utilizan siempre el mismo circuito Analogía: llamada telefónica Cada switch mantiene una tabla vc que tiene:

1. El puerto por el cual llega el paquete.
2. El identificador del circuito virtual (VCI) de entrada
3. El puerto por el cual debe salir el paquete
4. El identificador del circuito virtual (VCI) de salida

Normalmente debe esperarse un RTT completo mientras se establcece una conexión para poder enviar el primer paquete o celda. La solicitud de conexión debe llevarse para poder enviar el primer paquete o celda, pero los demás paquetes sólo tienen un identificador muy pequeño el VCI, haciendo el overhead muy pequeño. Si un switch o un enlace falla, el circuito virtual falla y una nueva conexión debe establcerse. Establcer una conexión de antemano, permite reservar recursos en los switches (espacio en buffers) Tipos de Conexiones

Conexión Permanente (PVC) Establecimiento: Este tipo de conexión la define y la finaliza el administrador de la red: una persona solicita a la red la creación de los registros en las tablas VC. Después de creado el circuito virtual ya se pueden enviar datos. Cierre de conexión: El administrador de la red, una persona, solicita o hace las operaciones que permitan “bajar” el circuito virtual.

Conexión por Solicitud (o conmutado) (SVC) Estaclecimiento:Cuando el nodo A desea enviar datos al nodo B envía un mensaje de solicitud de conexión a la red, luego el switch que la recibe se lo envía al siguiente, hasta llegar al nodo B. Este úlimo, si acepta la conexión, devolverá el identificador de circuito que desea utilizar (4 en el ejemplo anterior) y esta “aceptación” se repite en todos los switches que se encuentran en el camino. Después de construir el circuito virtual se empieza a enviar datos. Cierre de conexión: Cuando el nodo A no desea enviar más datos al nodo B, termina el circuito virtual enviando un mensaje de finalización a la red. El switch que recibe el mensaje borra la línea de la tabla de VC correspondiente a ese circuito y

envía un mensaje de finalización al siguiente switch para que repita la misma acción y así hasta alcanzar al nodo B. Si después de esto el nodo A envía un paquete o celda a la red, este puede ser descartado pues ya no existe el circuito virtual.

4.2 Sin conexión: Datagramas

El nodo puede enviar el paquete cuando quiera, no es necesario el establecimiento de una conexión. Cada paquete se envía independientemente y debe llevar toda la información necesaria para alcanzar su destino Analogía: SIstema postal. Cada switch mantiene una tabla de forwarding. Para cada nodo destino, tiene el puerto de salida que utiliza.

Ventaja: No se debe esperar un RTT para establecenr una conexión. Un nodo puede enviar tan pronto como este listo. El nodo origen no tiene porque saber si la red es capaz de entregar un paquete o frame o si el nodo está listo para recibir los datos. Ya que los paquetes son tratados independientemente, es posible cambiar el camino para evitar los enlaces y los nodos que están fallando. Ya que cada paquete lleva la dirección completa del nodo destino, la información adicional de control (overhead) que lleva es mucho mayor que la utilizada en el modelo orientado a conexión.

Conmutación Source Routing Toda la información sobre la topología de la red que se necesita para conmutar los paquetes es proporcionada por el nodo origen.

4.2.1 IP

- Connectionless (datagram-based)
- Best-effort (unreliable service)
- Paquetes se pueden perder
- Enviar fuera de orden
- Entrega de copias
- No hay un cota para el tiempo de entrega

Campos del header IP

Versión: actualmente 4, comienzan los 6 pero el resto del formato del header no es el mismo en ambas versiones.

Longitud Cabecera: en palabras de 32 bits (mínimo 5, máximo 15)

Longitud total: en bytes, máximo 65535 (incluye la cabecera)

Identificación , DF, MF, Desplaz.

Fragmento: campos de fragmentación

Tiempo de vida: contador de saltos hacia atrás (se descarta cuando es cero)

Checksum: de toda la cabecera (no incluye los datos)

Dirección fuente y destino 32 bits

Cada tecnología de red tiene a nivel de enlace un MTU (Maximum Transmission Unit). IP se adapta a la tecnología de red subyacente (MTU !!) Fragmentación ocurre si un router recibe un datagrama que debe reenviar a una red donde su $MTU < |datagrama|$ Los fragmentos reciben la misma cabecera que el datagrama original salvo por los campos 'MF' y 'Desplazamiento del Fragmento'. Los fragmentos de un mismo datagrama se identifican por el campo 'Identificación'. Todos los fragmentos, menos el último, tienen a 1 el bit MF (More Fragments). La unidad básica de fragmentación es 8 bytes. Los datos se reparten en tantos fragmentos como haga falta, todos múltiplos de 8 bytes (salvo quizá el último). Toda red debe aceptar un MTU de al menos 68 bytes (60 de cabecera y 8 de datos). Recomendado 576

4.2.2 IP Forwarding

if NetworkNum del destino = NetworkNum de algunas de mis interfaces **then**
  enviar datagrama al destino por esa interface
else
  if NetworkNum del destino esta en mi forwarding table **then**
    enviar datagrama al NextHop router
  else
    enviar datagrama al default router
  end if
end if

5 Nivel De Transporte

Enlace de Datos versus Transporte:

- Potencialmente conecta muchas maquinas diferentes(establecimiento y termino de conexión explícitos)
- Potencialmente diferentes RTT(requiere mecanismos adaptativos para timeout)
- Potencial,emte largos retardos en la red(requiere estar preparado para el arribo de paquetes muy anitiguos)

- Potencialmente diferentes capacidades en destino
- Potencialemnte diferentes capacidad de red

5.1 Protocolos End to End

- Garantía de entrega de mensajes
- Entrega de mensajes en el mismo orden que son enviados
- Entrega de a lo más una copia de cada mensaje
- Soporte para mensajes arbitrariamente largos mensajes
- Soporte de sincronización
- Permitir al receptor controlar el flujo de datos del transmisor
- Soportar múltiples procesos de nivel de aplicación en cada máquina

5.2 TCP

- **Orientado a conexión:** 3-way handshaje para setup y 2-2 way handshake para la liberación
- **Servicio de flujo de bytes:** App escriben bytes, TCp envía segmentos, app lee bytes
- **Control de flujo:** evita que el transmisor inunde al receptor
- **Control de congestión:** evita que el transmisor sobrecargue la red
- **Full duplex**
- **Es confiable:**
  - Acks
  - Checksums
  - Numeros de secuencia para detectar datos perdidos o desordenados
  - Retransmision de datos perdidos o corruptos despues de un timeout
  - Datos desordenados se podrán reordenar

Cada conexión es identificada por la 4-upla [SrcPort,SrcIPAdress, DstPort,DstAdress].
Ventana deslizante + control de flujo.
Flags: SYN, FIN, RESET, PUSH , URG, ACK.

Checksum: pseudoheader(IP) + TCP header + data.

Un segmento TCP se envía cuando:

1. Segmento full(MSS bytes)
2. No está full, pero hay un timeout
3. Pushed por la aplicación

5.3 Ventana deslizante

Se agrega a la ventana deslizante la advertised window para controlar el flujo.

Tamaño del buffer de envío: MaxSendBuffer Tamaño del buffer de recepción: MaxRcvBuffer

5.3.1 Lado Emisor

LastByteAcked $\leq$ LastByteSent
LastByteSent $\leq$ LAstByteWritten
Se bufferean los bytes entre LstByteAcked y LastByteWritten
LastByteSent - LastByteAcked $\leq$ AdvertisedWindow
EffectiveWindow = AdvertisedWindow - (LastByteSent - LastByteAcked)
LastByteWritten - LastByteAcked $\leq$ MaxSendBuffer
Bloquear Tx si (LastByteWritten - LastByteAcked) + y ¿ MaxSenderBuffer, y bytes que se desean escribir.
Siempre enviar ACK en respuesta a la llegada de segmentos de datos
Tx persiste enviando 1 byte cuando AdvertisedWindow = 0

5.3.2 Lado Receptor

$LastByteRead < NextByteExpected$
NextByteExpected $\leq$ LastByteReceived + 1
Se bufferean los bytes entre LastByteRead y LastByteReceived
LastByteRcvd - LastByteRead $\leq$ MaxRcvBuffer
AdvertisedWindow = MaxRcvBuffer - (LastByteRcvd - NextByteRead)

5.4 UDP

El protocolo UDP no garantiza la confiabilidad, pero brinda la multiplexación de paquetes hacia los distintos procesos de los hosts. Además es rápido ya que no establece una conexión.

5.5 Retransmisión Adaptativa

Dado que hay una red entre los hosts, no existe un RTT fijo.

5.5.1 Algoritmo original:

Se mide SampleRTTi por cada para segmento-ack. Calcula el promedio ponderado de RTT.
$EstimatedRTT = \alpha * EstimatedRTT + \beta * SampleRTTi$ donde a +b = 1
Se fija un TimeOut basado en EstimatedRTT: TImeout = 2 * EstiamtedRTT

5.5.2 Algoritmo de Karn-Patridge

No considerarRTT cuando se retransmite Duplicar timeout luego de cada retransmisión

5.5.3 Algoritmo de Jacobson/Karels

$Diff = sampleRTT - EstRTT$
$EstRTT = EstRTT + (d * Diff)$
$Dev = Dev + d(|Diff| - Dev)$

Se considera la varianza cuando fijamos el timeout:
$Timeout = \mu * EstRTT + \phi * Dev$

Los algoritmos son tan buenos/malos como la granularidad del reloj. Es muy importante la precisa estimación del timeout para controlar la congestión. La idea es no retransmitir cuando no es necesario.

Cuando Transmitir: Envío de pocos bytes: retardar el ACK y de las actualizaciones de ventana

5.6 Algoritmo de Nagle

: If el tamaño de la ventana y los datos disponibles en el buffer $\geq$ MSS Enviar un segmento full Else If los datos en vuelo estan sin reconocer bufferear el nuevo dato hasta que llegue el ACk Else enviar todos los datos nuevos ahora

Para que no haya problemas cuando se anuncia AdvertisedWindow = 0, TCP envía periódicamente windows probes con 1 bytes de dato

5.7 Sindrome Silly Window

Se envían datos en bloques y el receptor su app lee muy despacio , por ejemplo un 1 byte a la vez Solución: No enviar aviso de ventana para 1 Byte. Esperar hasta tener una cantidad de buffer considerable

Buffer = MSS que se anuncio en la conexión
O a la mitad de la capacidad del buffer , el valor más pequeño de los dos

6 Performance

6.1 Midiendo la performance

Hay que tener en cuenta, que a pesar de tener un bitrate físico, el medio no tiene una eficiencia total (se asume del %65). Además el bitrate tiene en cuenta a todos los datos, incluyendo los encabezados de IP, TCP.

El modelado de las transmisiones se suele hacer con una cola M/M/1. Sea R el ancho de banda del enlace (en bps), L la longitud del paquete (bits) y a la media de tasa de llagada del paquete se mide la intensidad de tráfico como La/R. Cuando la intensidad es aproximadamente 0 hay un medio de retardo de cola pequeño. A médida que va creciendo la intensidad hay más retardo, en especial cuando la intensidad de tráfico (La/R) es mayor a 1 llega más trabajo del que puede servirse. Este caso tiende a un retardo infinito.

6.2 Bufferbloat

Una médida para que no se pierdan los paquetes es agrandar los tamaños de los buffers. Esto puede llegar a mejorar el throughput de una sesión TCP pero a cosata de un incremento de la latencia (afecta de igual manera en aplicaciones basadas en UDP). Luego el tamaño de los buffers impacta en la performance las aplicaciones. En 1985 John Nagle demuestra que el aumento del tamaño del buffer empeora el fenómeno de la congestión. El término bufferbloat se le llama a la existencia de los buffers excesivamente grande en los sistemas, en particular en los relacionados con las redes y comunicaciones.

Desde 2011 para certificar CM DOCSIS 3.0 es mandatorio soportar control de buffer. Los parámetros de control de buffer limitan la cantidad máxima de datos que pueden ser encolados por cada flujo de servicio. De esta manera se ofrece un mecanismo para balancear la relación throughput y latencia de un servicio.

7 Congestión

La congestión se define como el estado de sobrecarga sosteninda de una red, donde:
La demanda de recursos (enlaces y buffers) se encuentra al límite o excede la capacidad de los mismos.
La consecuencia es perceptible en términos de QoS degradada.

Hay varias formas de encarar la solución:

- Sobredimensionamiento
- Diseño cuidadoso
- Control proactivo
- evitar (control preventivo)
- Decrementar la carga

7.1 Síntomas de Congestión

Pérdida de paquetes los buffers se saturan en routers o switches)

Retardos crecientes por las colas en los buffers

7.2 Consideraciones

Se considera que los nodos tienen una politica de schedulling `FIFO` y una politica de manejo de colas `Drop tail`. Ambas políticas producen una sincronización global cuando los paquetes descartados provienen desde distintas conexiones no sincronizadas entre ellas (lo cual es casi siempre).

7.3 Control de Congestión vs Control de Flujo

La congestión es un efecto global, involucra a todos los hosts y routers compartiendo una subred. Se evita que los transmisores sobrecarguen el interior de la red. En cambio el control de flujo controla el tráfico punto a punto entre un receptor y un transmisor particulares. Se evita que los transmisores sobrecarguen a receptores lentos.

7.4 Métricas

El crecimiento de alguna o varias de estas métricas indican congestión:

- Porcentaje de paquetes descartados por falta de espacio en buffer
- Longitud media de una cola (buffer)
- Cantidad de paquetes que generan timeout y son RTX
- average packet delay
- standard deviation of packet delay

7.5 Causas

- Inundo con tráfico destinado a una misma línea de salida (la cola se llena, tail drop)
- Procesadores lentos o problemas con software de routeo
- Cuellos de botella en algunas partes del sistema (difetentes velocidades)
- El efecto de congestión tiende a realimentarse y empeorar

7.6 Control de Congestión

Es el esfuerzo hecho por los nodos de la red para prevenir o responder a sobrecargas de la red que conducen a pérdidas no controladas de paquetes. Se pueden preasignar recursos para evitar la congestión. Otra opción es la de liberar recursos y controlar la congestión sólo si ocurre (cuando ocurre).

La idea es que sea utilizada eficientemente y al mismo tiempo en forma equitativa.
Un buen indicador de la eficiencia es la $potencia = throughput/delay$. El control se puede implementar en los extremos de la red como en los routers dentro de ella.

7.7 Performance de la red en función de la carga

A medida que la carga de la red aumenta, el throughput se incrementa linealmente. Sin embargo, a médida que la carga alcanza la capacidad de la red los buffers en los routers comienzan a llenarse. Esto causa el incremento del tiempo de respuesta y disminuye el throughput. Una vez que los buffers de los routers comienzan a sobrecargarse ocurre la pérdida de paquetes. Bajo cargas extremas, el tiempo de respuesta tiende a infinito y el throughput tiende a cero. A esto se le llama el punto de colapso de congestión. Cuando empieza a caer rápidamente el throughput se le llama cliff.

7.8 Teoría de Control

Los algoritmos de control de congestión se pueden clasificar en lazo abierto y lazo cerrado. A su vez los de lazo cerrado se pueden clasificar de a cuerdo a como realizan la realimentación: implícita (TCP) o explícita (ECN). EL control de lazo abierto se ve principalmente en redes de conmutación de circuitos.

El control de lazo cerrado con feedback explícito se da por ejemplo cuando se envía un mensaje con el bit ECN (Explicit Congestion Notification) en 1. El lazo cerrado con feedback implícito se ve por ejemplo en

TCP cuando se intuye que hay congestión a partir de la llegada de los acks duplicados y los timeouts.

7.9 Random Early Detection

Algoritmo de notificación implícita de inminencia de congestión. Simplemente descarta el paquete (luego en TCP habrá timeouts). Podría hacerse explícita manrcando el paquete (como en ECN). El descarte es aleatorio temprano, es decir: en lugar de esperar a que se llene la cola, descarta cada paquete de entrada con alguna probabilidad de descarte cada vez que la cola exceda algun nivel de descarte.

7.10 Flow Random Early Detection

Red tiene problemas de imparcialidad ya que no es justo con las conexiones de baja velocidad.
Cuando se alcanza el umbral máximo, RED descarta los paquetes aleatoriamente. Puede darse que el paquete descartado pertenezca a un flujo que esté utulizando menos recursos de los que le correspondería en igualdad de conciones. Cuando la logitud media de la cola está en un punto fijo dentro de los dos umbrales, todos los paquetes entranstes se descartan con la misma probabilidad. FRED soluciona estos problemas de imparcialidad menteniendo umbrales y ocupaciones del buffer para cada flujo activo. Luego es muy costoso en cuanto al alto operación en los routers. FRED debe identificar cada flujo que tenga paquetes en el buffer y actualizar la información de cada paquete.

7.11 Traffic Shaping

Cuando el tráfico es rafagoso impacta en el nivel de congestión. Traffic shaping es un método de lazo abierto que trata de guiar la congestión, forzando a los paquetes a transmitirse a una velocidad máß predecible. Intenta mantener el tráfico constante regulando la tasa media de transmisión de los datos.

7.11.1 Leaky Bucket

Para entender este algoritmo se puede hacer una analogía con un balde que tiene un agujero. Cuando el balde tiene agua (datos a enviar) este transmitirá a una velovidad constate. En cambio tendrá una velocidad nula cuando el balde este vacío. El mecansimo de leaky buckey no es más que un sstema single server queing con tiempo de servicio constante y cola finita. Los paquetes llegan en cualquier instante, pero a los hosts se les permite poner solo un paquete por tick en la red. Cuando los paquetes son de diferentes tamaños, es mejor usar un número fijo de bytes por tick. En el

caso de que la cola este llena, los paquetes qie llegan son descartados (tail drop).

7.11.2 Token Bucket

Token bucket permite picos de tráfico cuando le llega una ráfaga grande de paquetes. Cada balde genera una cantidad de tokens por una determinada cantidad de tiempo. Para tranmitir se necesita consumir un token. Si no hay, se espera. En definitiva Token bucket permite un BurstSize y Average Rate.

7.11.3 Control de Congestión en TCP

Additive increase y multiplicative decrease por RTT, de acuerdo a si hay ACK o no.
Fórmula de Mathis: $BW = \frac{MSS*C}{RTT\sqrt{p}}$