STU FEI

Počítačové siete 2

Kľúčové vedomosti

Martin Drozda

Podmienky absolvovania

Cvičenia: 30 bodov (min. 15 bodov)

Skúška: 70 bodov, náročnosť skúšky predpokladá min. 10 absolvovaných prednášok

Povinné vedomosti na skúške:

- OSI referenčný model (vymenovať 7 vrstiev)
- Prepočet mW, W na Dbm, Db a naopak
- Čo je to broadcast, multicast a unicast

Správne zodpovedanie potrebné pre úspešné absolvovanie predmetu

Druhy MAC protokolov

Deterministické (scheduled)

– TDMA, Time division multiple access: časový multiplex, vyžaduje presnú synchronizáciu času, pridávanie používateľov vyžaduje nový plán rozdelenia času

– FDMA, Frequency division multiple access: frekvenčný multiplex, rozdielne kanály (frekvencie) pre rozdielnych používateľov

– CDMA, Code division multiple access: kódový multiplex, na dekódovanie je potrebný kód, dekovanie je založené na korelácií

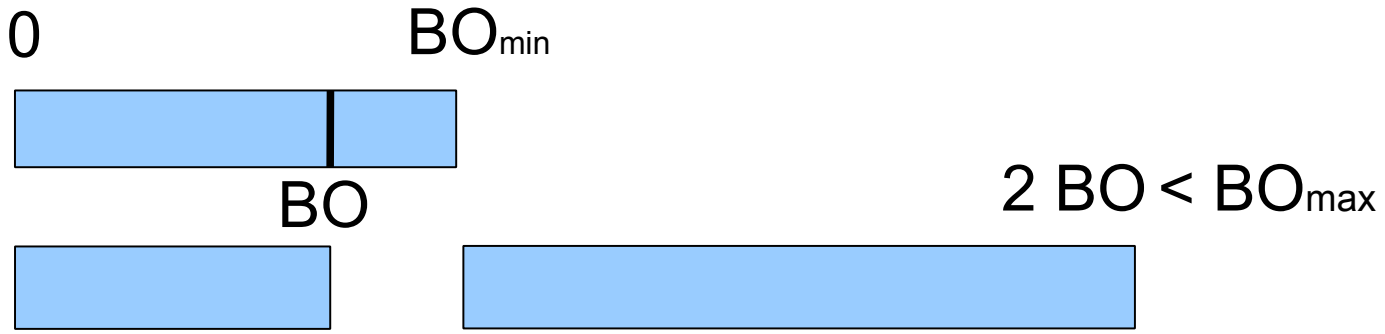
Stochastické (založené na súťaži)

– Súťaž o prístup k médiu: (virtuálna) detekcia nosnej, exponenciálne čakanie.

Exponenciálne čakanie

Oneskorenie nastane, keď je kanál obsadený.

Veľkosť okna súperenia je nastavená na min. veľkosť (BO_{min} ~20 mikrosekúnd). Čakanie je zvolené z okna (uniformne náhodne); ak je kanál stále obsadený veľkosť okna sa zdvojnásobí. Ak je kanál voľný a prebehne poslanie paketu veľkosť okna je znova nastavená na min. veľkosť. Max. veľkosť okna je zvyčajne určená $BO_{max} = \sim 16 BO_{min}$.



Exponenciálne oneskorenie

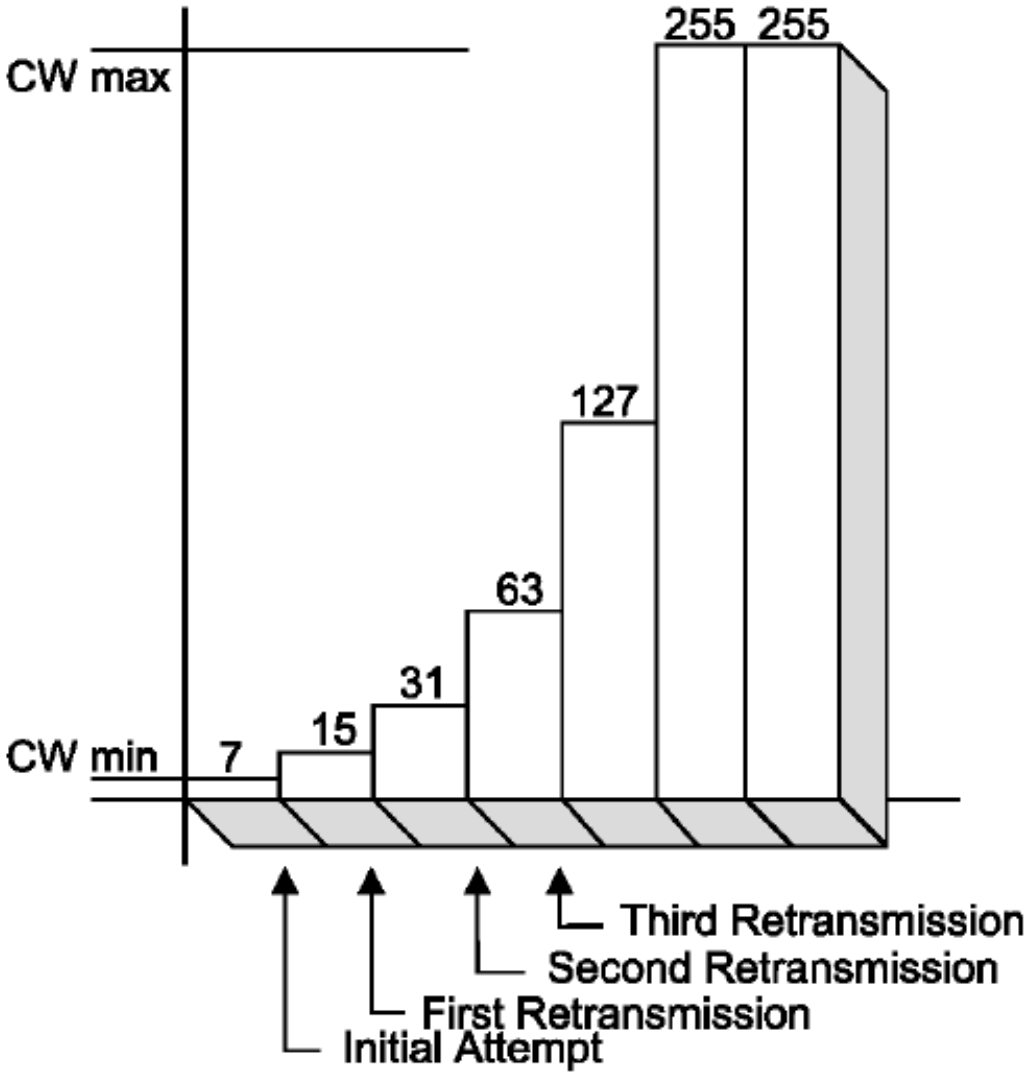
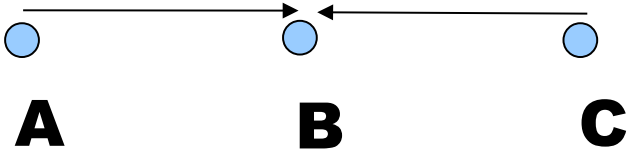


Figure source: IEEE Std 802.11™-2007

Skrytý terminál

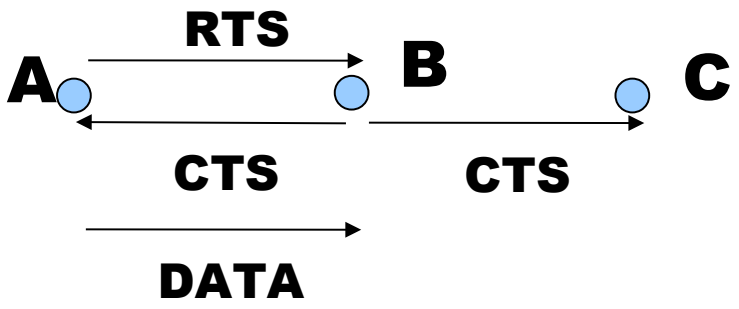
Skrytý terminál je hlavný nedostatok CSMA. A a C počujú B; A nepočuje C a C nepočuje A.



A aj C chcú poslať dáta B, detegujú nosnú a zistia, že kanál je voľný.

MACA: Multiple Access Collision Avoidance

RTS = Ready-to-Send
CTS = Clear-to-Send

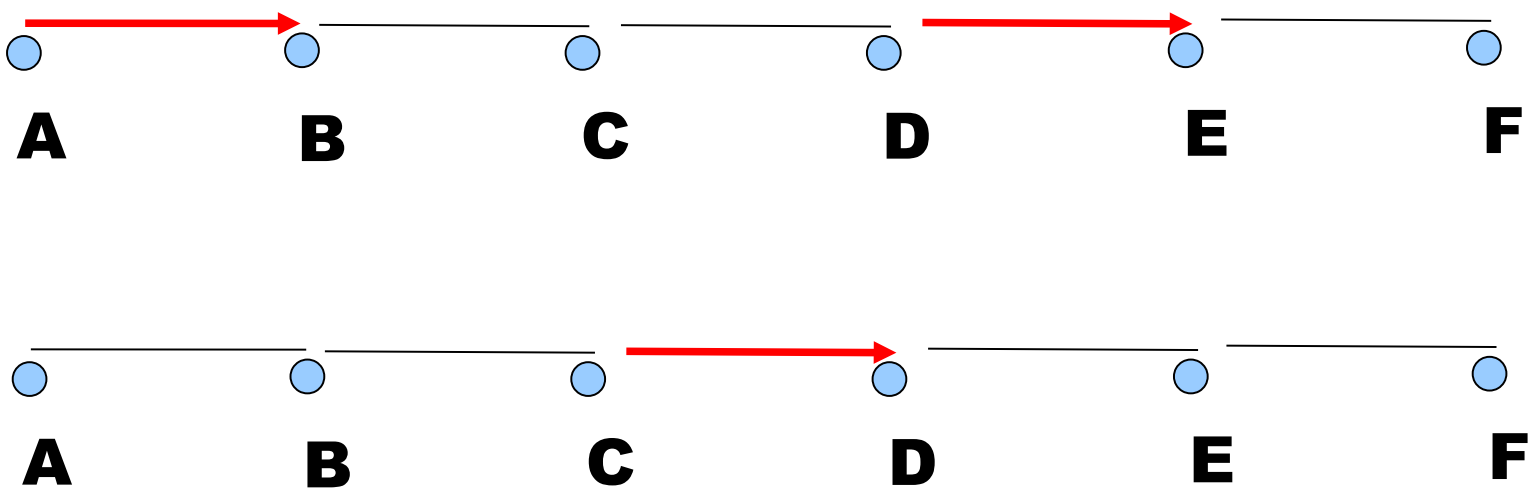


A pošle RTS. Ak je B pripravené prijať dátový paket, tak pošle CTS. A aj C prijmú CTS, C oneskorí posielanie paketov. A pošle paket. Ak A neprijme CTS tak nastane exponenciálne čakanie.

Okamžitá kapacita MAC vrstvy

Koľko prenosov je možných v tom istom čase na MAC vrstve?

V našom príklade (dole) sú to jeden alebo dva.



Maximálne d2- párovanie

Nech G(V, E) je neorientovaný graf, ktorý reprezentuje bezdrôtovú sieť.

Max. d2-párovanie: nájdi najväčšiu množinu hrán E', kde E' je podmnožina E, takú, že žiadna z dvoch hrán v E' nie je spojená hranou z E.

Zložitosť: NP-úplné; rozhodovací prípad.

Chamtivý algoritmus

1. Opakuj nasledujúce kroky dovtedy, pokým E(G) = {}.

2. Zvoľ hranu e takú, že r(e) = min r(e'); e' patrí E(G).

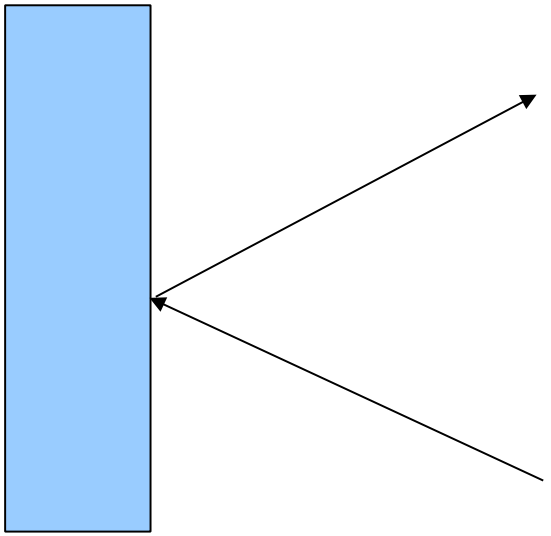
3. Pridaj e do E' a zmaž všetky hrany v E(G) do vzdialenosti 2.

r(e) = r(u) + r(v); u,v sú uzly, r(u) je stupeň uzla u.

O(1) aproximácia.

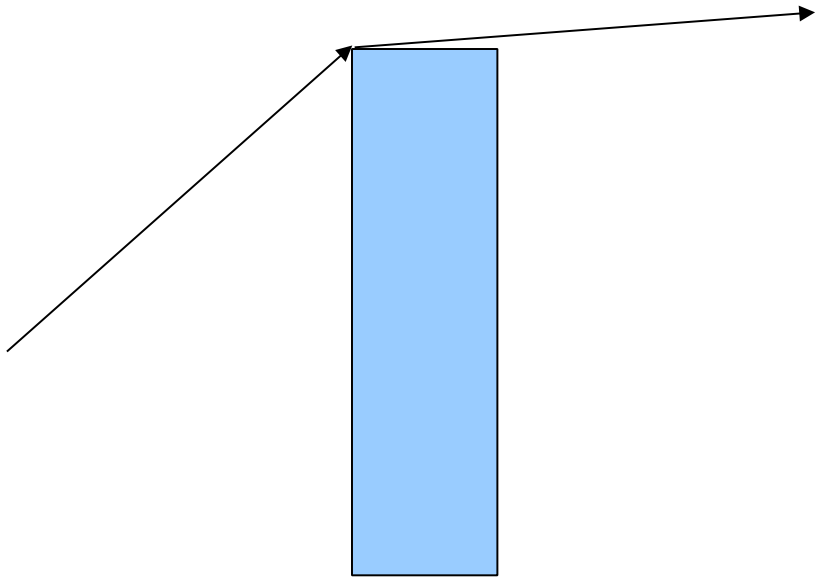
Odraz

Odraz nastane, ak objekt má veľkosť rádovo väčšiu ako vlnová dĺžka.



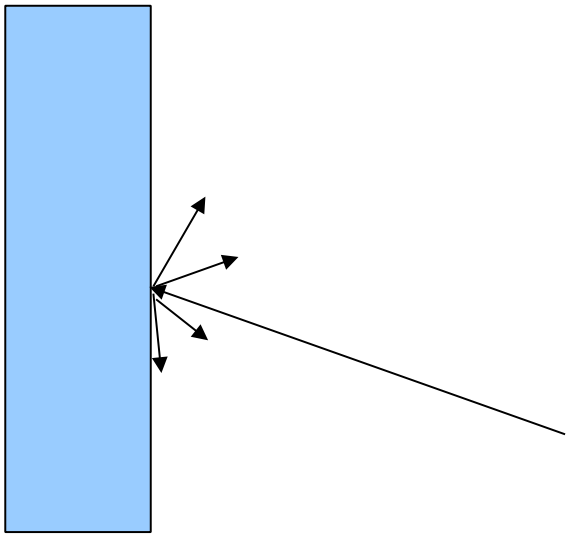
Ohyb (difrakcia)

Ohyb nastane, ak šíreniu signálu prekáža objekt s ostrými hranami. Rádiové vlny zmenia svoj smer, ohnú sa okolo objektu.



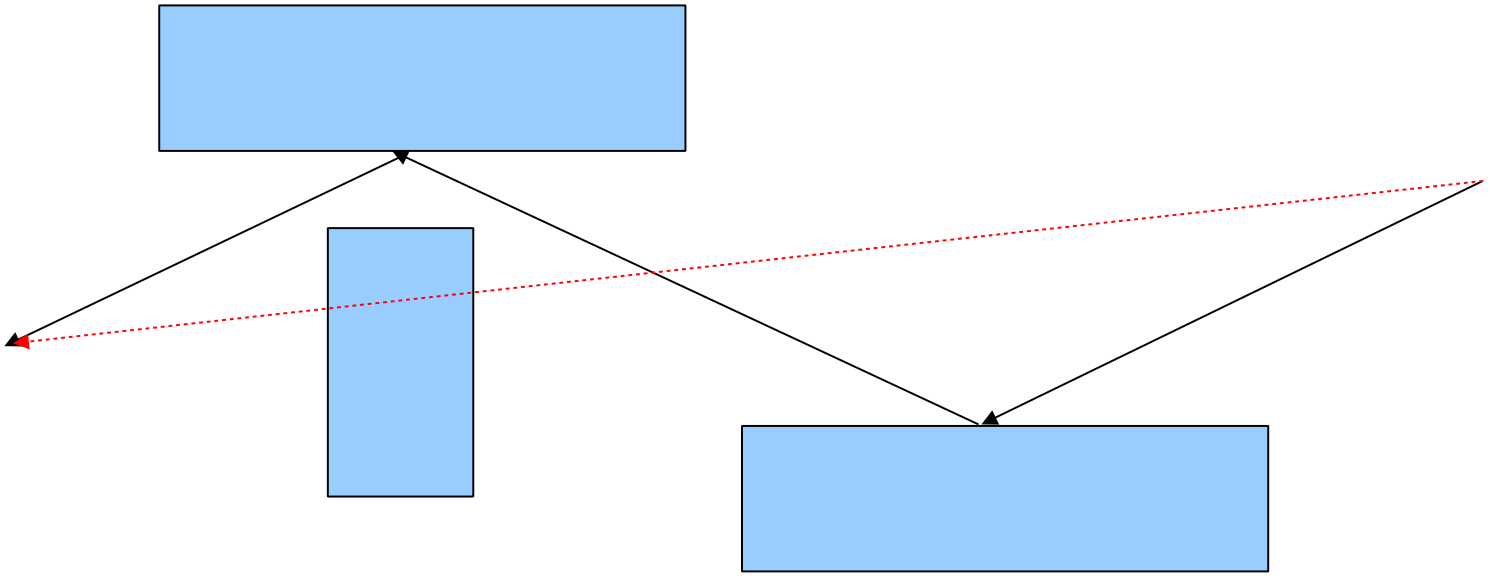
Rozptyl

Rozptyl nastane, ak médium cez, ktoré prechádza rádiová vlna obsahuje objekty porovnateľné alebo menšie ako vlnová dĺžka a počet týchto objektov je veľký. Príklady takýchto objektov sú cestné značenie, stromy s listami, ľudia atď.



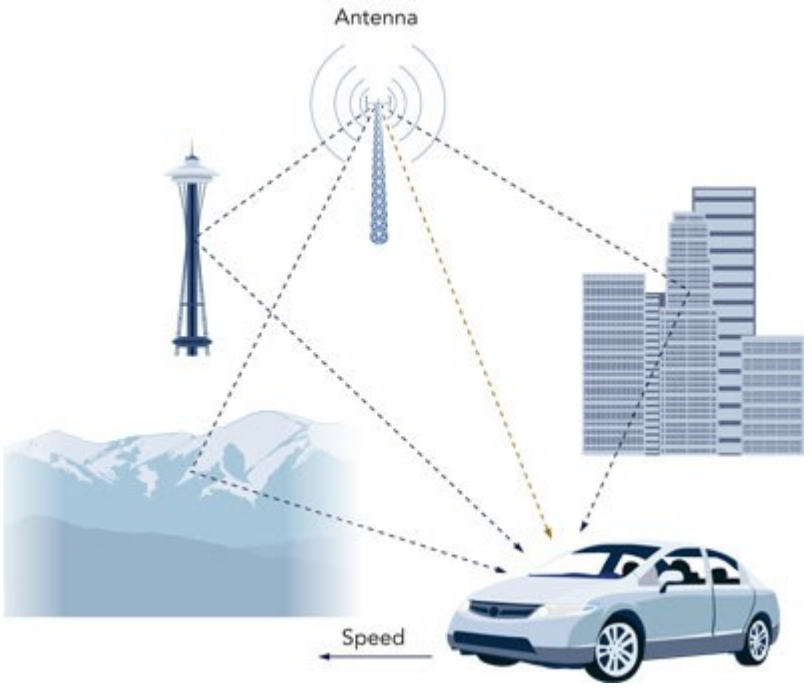
Viac-cestné šírenie (multipath)

V reálnych podmienkach nastáva viac-cestné šírenie signálu, kde sa kombinujú javy ako odraz, ohyb, rozptyl, absorpcia. Tieto javy umožňujú šírenie signálu inač ako len “priamočiaro ” (line of sight).



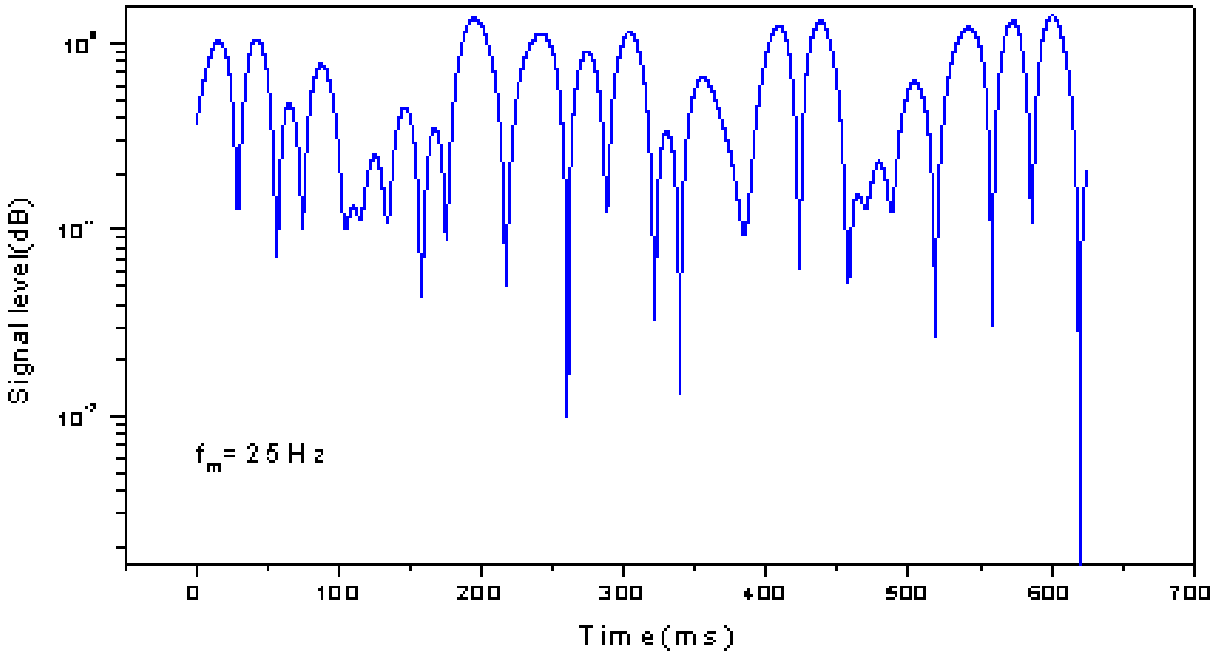
Útlm (fading)

Útlm je dôsledkom odrazu, ohybu, rozptylu a viac-cestného šírenia signálu.



Zdroj obrázku: http://www.tmworld.com/article/509529-RF_fading_simulation.php

Útlm (fading)



Zdroj: M. Djebbouri et al., Estimation of mobile velocity in Rayleigh fading channel. Electronics Letters, 2004.

VBR

Poissonov model:

$$P(N_t = k) = e^{-\lambda t} \frac{(\lambda t)^k}{k!}$$

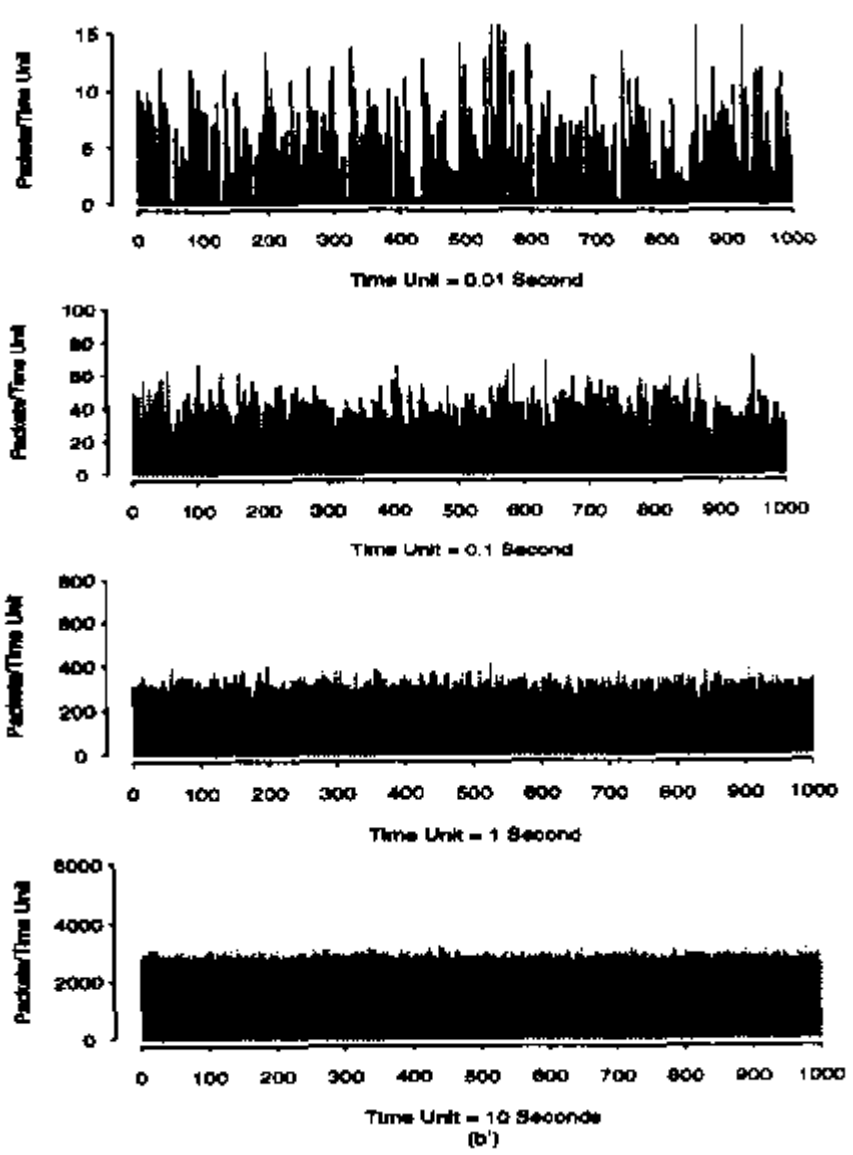
Stredná hodnota = λ, rozptyl = λ, kde λ je parameter.

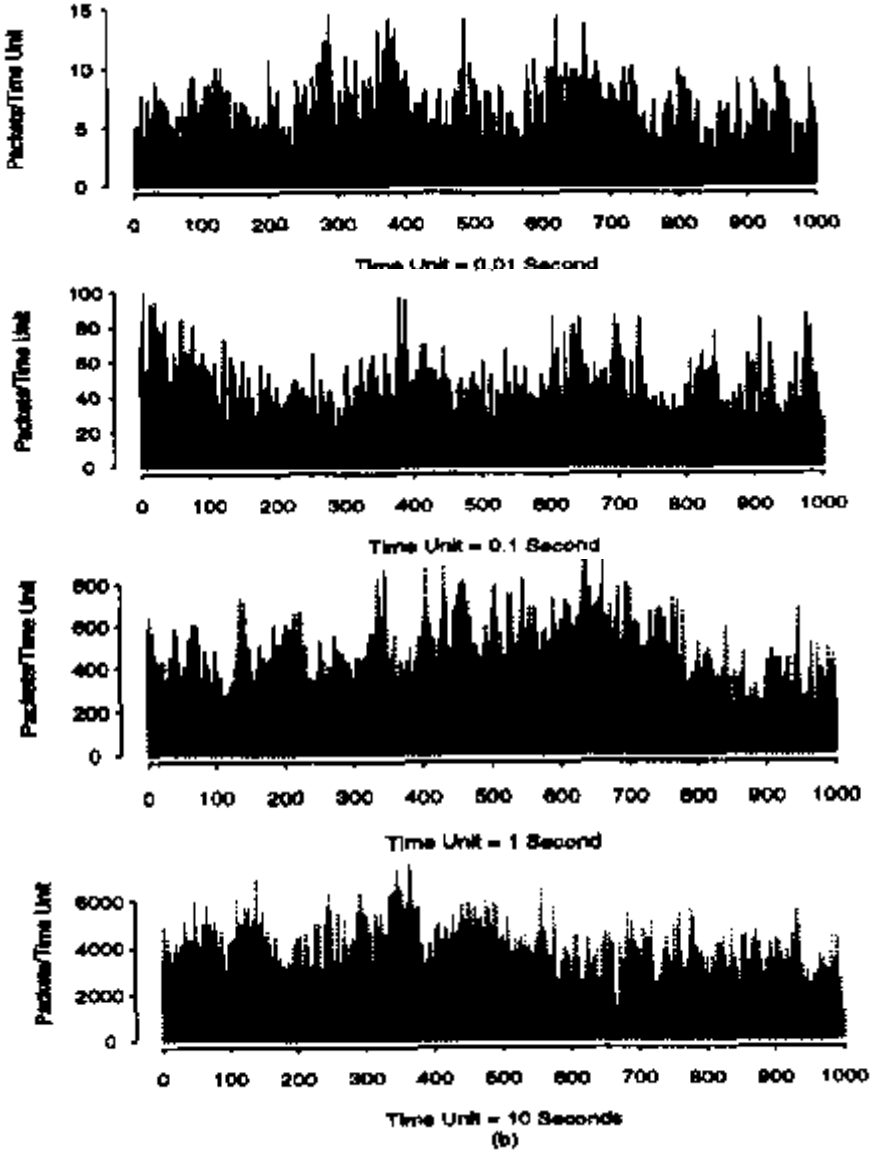
Pre k = 0, 1, 2, 3, ...

Pravdepodobnosť výskytu udalosti do času t = k.

$P(N_t = 0) = e^{-\lambda t}$ Pravdepodobnosť výskytu prvej udalosti.

Poisson vs realita





Dátové štruktúry pre šírenie signálu

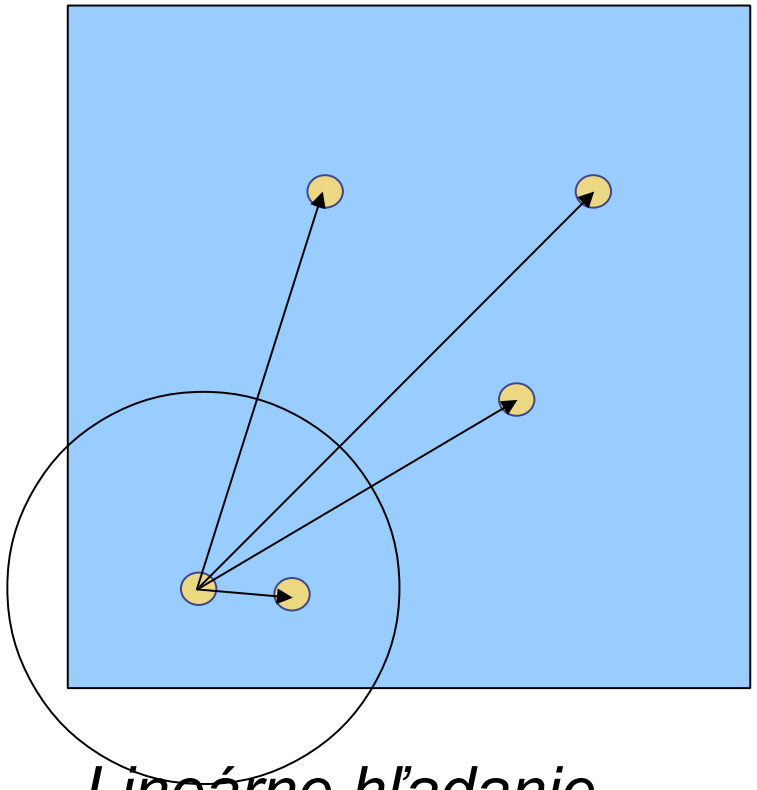
Lineárne hľadanie (O(n)):
Nájdi všetkých susedov a zisti, či sú dosiahnuteľní.

Flat binning:
Simulovaná oblasť je pravidelne rozdelená na štvorce. Hľadaj vo štvorcoch, ktoré sú do určitej vzdialenosti. Vyžaduje vedomosť o príslušenstve k štvorcu.

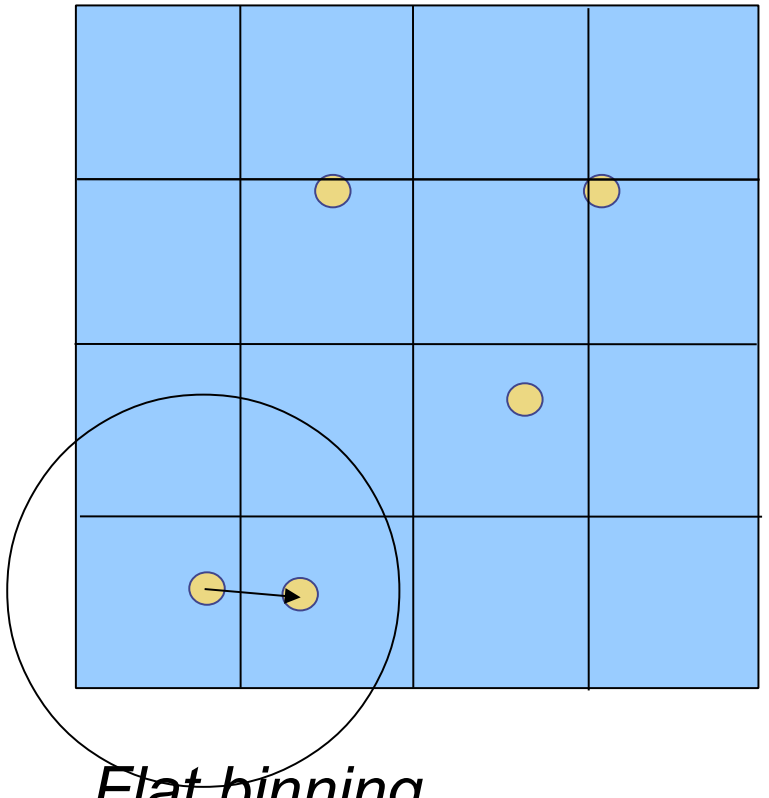
Hierarchické hľadanie:
Rekurzívne rozdeľ simulovanú oblasť pozdĺž osi x a y

Predpoklady: uniformne rozdelené uzly (v jednom štvorci relatívne nízky počet uzlov), akceptovateľná zmena polohy uzlov

Dátové štruktúry pre šírenie signálu

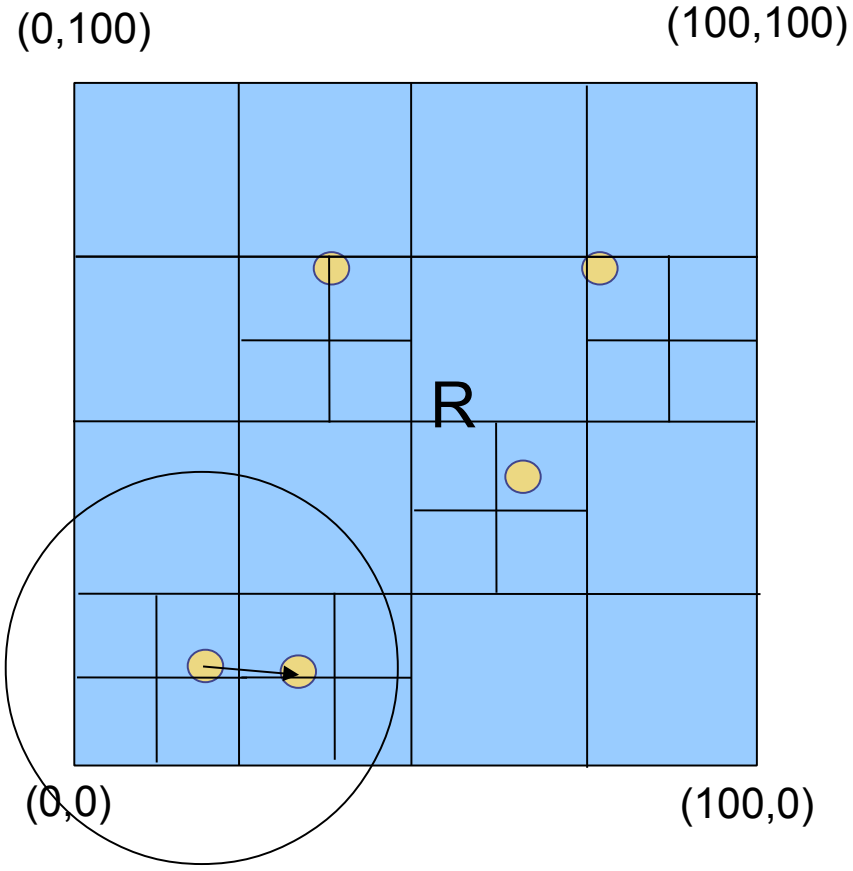


Lineárne hľadanie
Vzdialenosť počítaná pre každý uzol



Flat binning
Vzdialenosť počítaná pre uzly v susedných štvorcoch. Príslušnosť uzla ku štvorcu je potrebná.

Hierarchické hľadanie



Priestorový dekompozičný strom. Deliace body sú fixné koordináty. Štvorce sú listy.

height = log_4(field_size/bin_size)

Príslušnosť ku štvorcu je potrebná.

Začneme v koreni R a delíme priestor rekurzívne. Podobné ako binárne hľadanie, ale v 2D. Je stred štvorca v dosahu signálu?

Smerovanie: link state a distance-vector

Link state: každý uzol udržuje kompletnú vedomosť o sieti tak, že cez broadcast periodicky posiela informáciu o svojich susedoch, kvalite pripojenia k ním. Uzol, ktorý takúto informáciu dostane aktualizuje svoje smerovaciu tabuľku a prepočíta vzdialenosti k všetkým ostatným uzlom.

Distance-vector routing: každý uzol udržuje vedomosť o vzdialenostiach k všetkým ostatným uzlom a o susednom uzle, ktorý leží na najkratšej ceste k cieľovému uzlu. Úplná cesta k cieľovému uzlu nie je známa (okrem prípadu keď je cieľový uzol zároveň aj susedný uzol).

Smerovanie: link state a distance-vector

Link state: v hlavičke paketa je kompletná cesta do cieľového uzla.

Distance-vector routing: v hlavičke paketa je id nasledujúceho uzla, tento nasledujúci uzol rozhodne kam bude paket v ďalšom kroku preposlaný.

Smerovanie: proaktívne a reaktívne

Proaktívne protokoly udržujú cesty k všetkým ostatným uzlom bez ohľadu na to, či nejaká cesta bude v budúcnosti potrebná.

- *DSDV*

Reaktívne protokoly vypočítajú cestu k cieľovému uzla až keď je potrebná.

- AODV, DSR

Hybridné protokoly

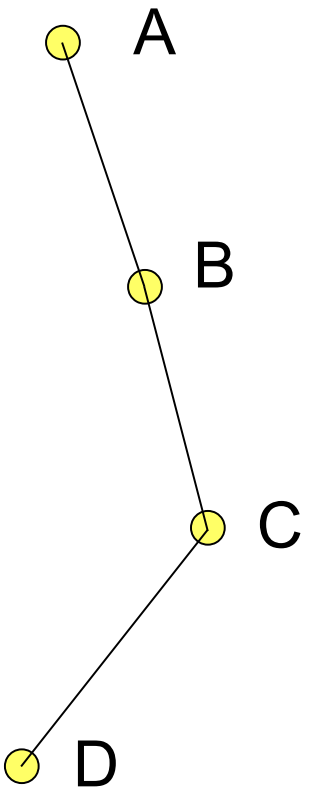
AODV (Ad hoc on-demand distance vector)

Reaktívny protokol.
Distance-vector smerovanie.
Hop-by-hop smerovanie.

Využíva:
- *RREQ: route request*
- *RREP: route reply*
- *RERR: route reply*

Hľadanie cesty:
- *Uzol potrebuje poslať paket a cesta k cieľovému uzlu nie je známa, Potom cez broadcast pošle RREQ, každý RREQ má sekvenčné číslo.*
- *Len prvý prijatý RREQ s daným sekvenčným číslom je preposlaný cez broadcast.*
- *(odpoveď) Uzol, ktorý pozná cestu k cieľovému uzlu alebo samotný cieľový uzol odpovedajú s RREP na prijatý RREQ. RREP je poslaný cez unicast.*

Problém počítania donekonečna



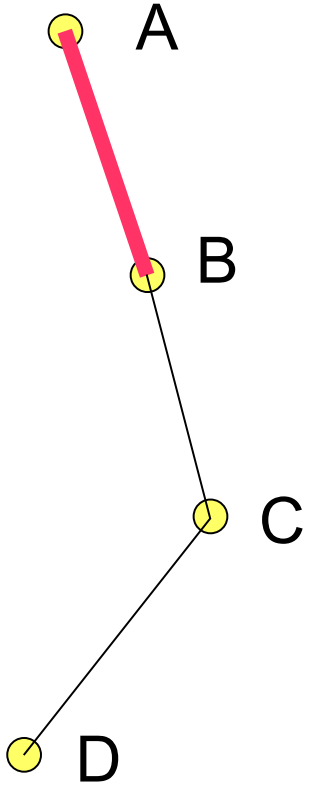
Smerovacia tabuľka A:
B: 1 C: 2 D: 3

Smerovacia tabuľka B:
A: 1 C: 1 D:2

A:2 B: 1 D:1

A: 3 B:2 C:1

Problém počítania donekonečna



Linka A-B prestane existovať

B inkrementuje vzdialenosť k A, pretože vidí, že C má platnú cestu k A

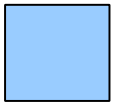
Potom C inkrementuje svoju vzdialenosť k A, pretože B inkrementovalo svoju vzdialenosť k A

Z tohto dôvodu sú potrebné sekvenčné čísla

Životný cyklus dátového paketa

- Dáta sú generované na aplikačnej vrstve (aplikáciou)
- Transportný protokol otvorí spojenie, dáta sú vložené do UDP/TCP paketu s hlavičkami
- Paket je ďalej vložený do IP paketu s hlavičkou
- Cesty do cieľového uzla sú najdené pomocou smerovacieho protokolu
- Paket je vložený do MAC paketa s hlavičkou
- Paket je prenesený fyzickou vrtstvou

TCP: slow start/congestion avoidance



1 segment je poslaný

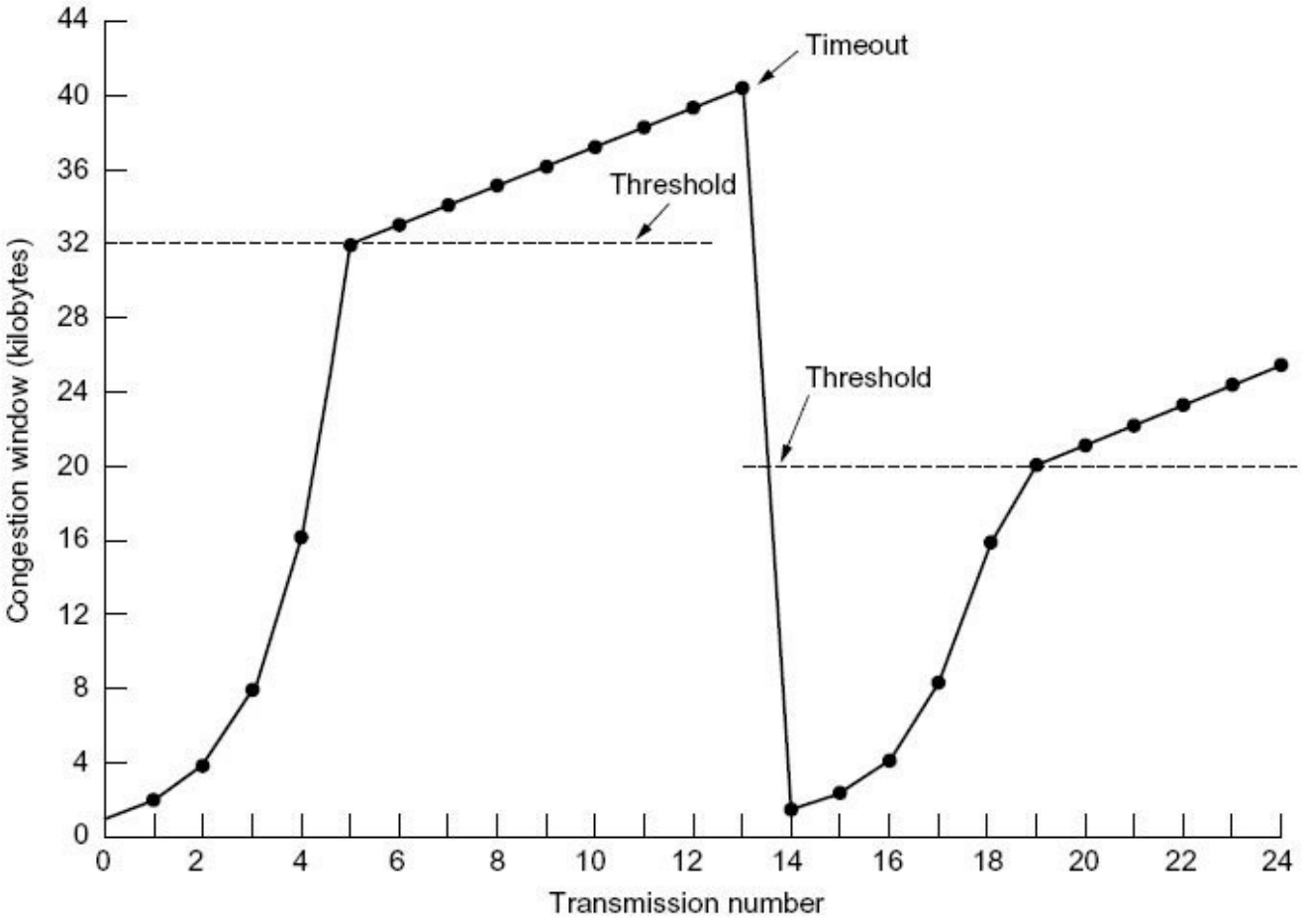
1 segment je potvrdený, veľkosť cwnd je inkrementovaná

1 segment je potvrdený, 1 nie je zatiaľ potvrdený

2 segmenty sú potvrdené

Exponenciálny pomalý štart je implementovaný na báze potvrdenia každého segmentu

TCP: slow start/congestion avoidance



TCP: fast retransmit

Duplicate ACK: ak je prijatý segment mimo správnu sekvenciu

Ak sú prijaté 3 alebo viaceré ACK, potom je segment opätovne poslaný

Fast retransmit = nečaká sa na time-out

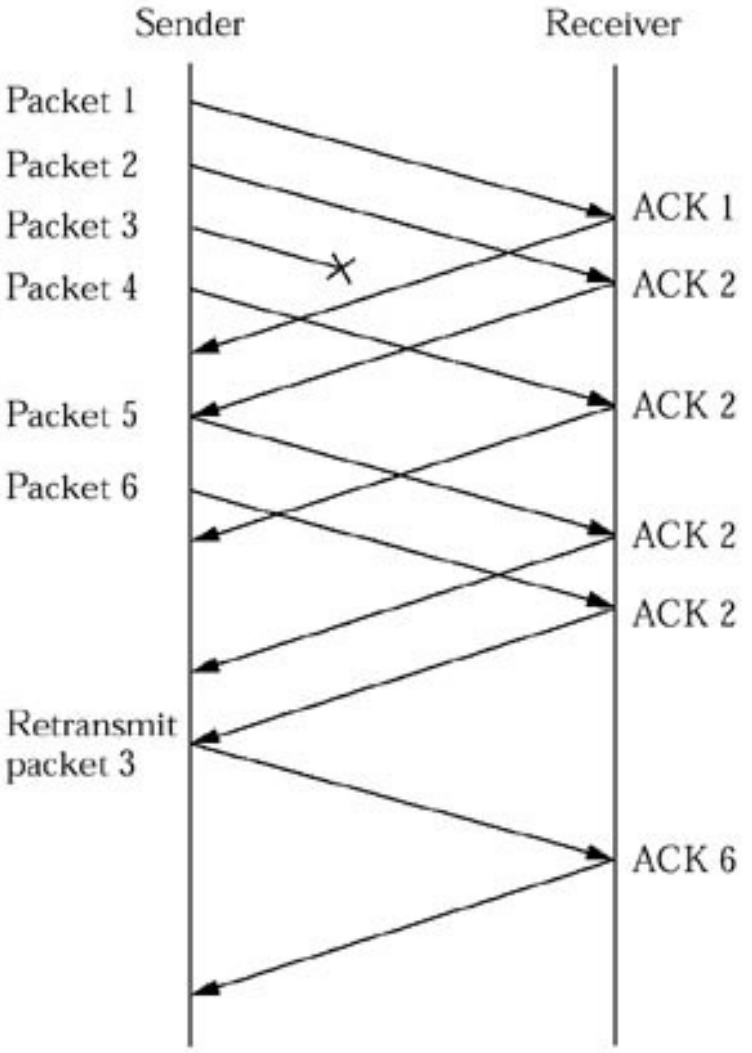


Figure source: http://www.soi.wide.ad.jp/class/99007/slides/23/19.html

Monitorovanie paketovej fronty

Každý uzol má paketovú frontu, t.j. buffer pre prijaté pakety

Veľkosť paketovej fronty je obmedzená a môže sa teda naplniť

Možnosti uvoľnenia paketovej fronty:

- tail drop; zmazanie posledných *n* prijatých paketov
- random drop; náhodné zmazanie prijatých paketov
- Random early detection (RED)

Tail drop / random drop môžu spôsobiť resynchronizáciu TCP, t.j. veľkosť okna zahltenia môže byť zmenená pre mnoho uzlov

Random early detection (RED)

- Ak je veľkosť fronty menej ako min, žiadne pakety nie sú zmazané
- Ak je veľkosť fronty viacej ako max, všetky nové prijaté pakety sú zmazané
- Ak je min ≤ veľkosť<max, nový prijatý paket je zmazaný s pravdepodobnosťou pa

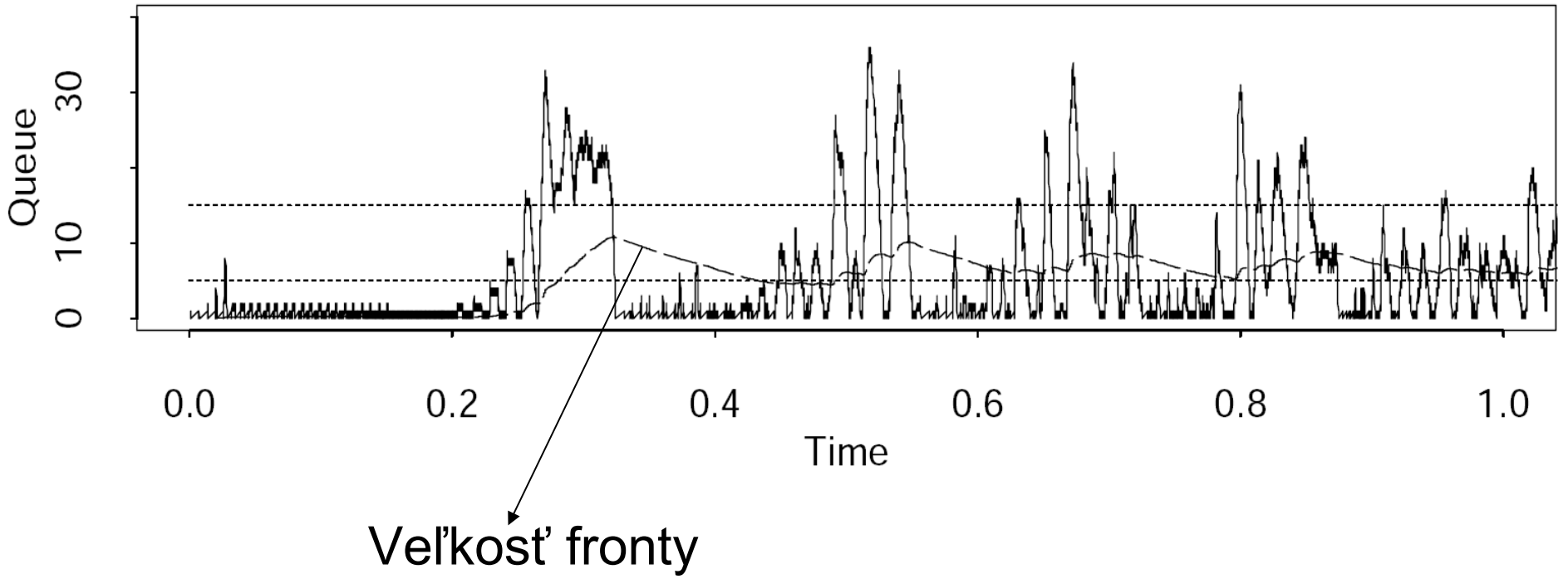
pb = maxp (avg – min)/(max – min); pb = <0, maxp>

pa = pb / (1 – count.pb); count = count je počet paketov od posledného zmazaného paketa

Vlastnosti:

- Garantuje max. veľkosť fronty
- Menej re-synchronizácie

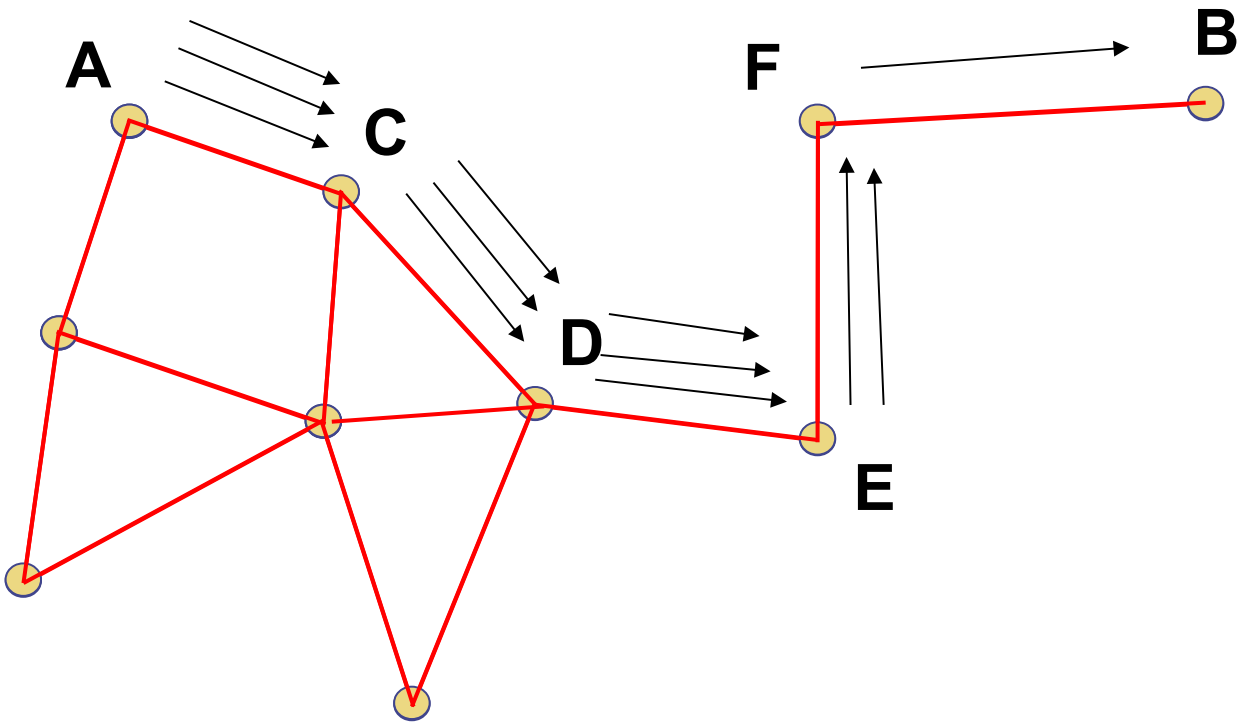
Random early detection (RED)



Source: S. Floyd, V.Jacobson. „Random early detection gateways for congestion avoidance“, IEEE/ACM Transaction on Networking, 1993.

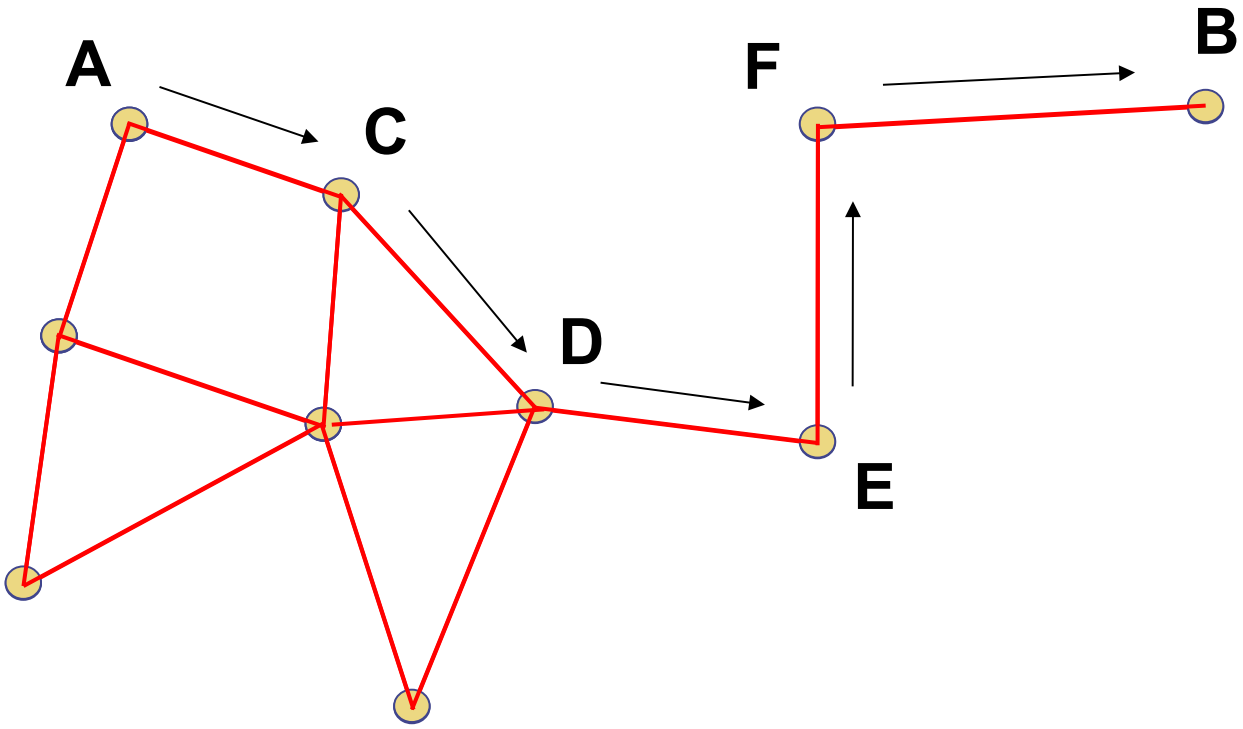
Unicast

Je potrebné poslať identickú informáciu uzlom E, F a B

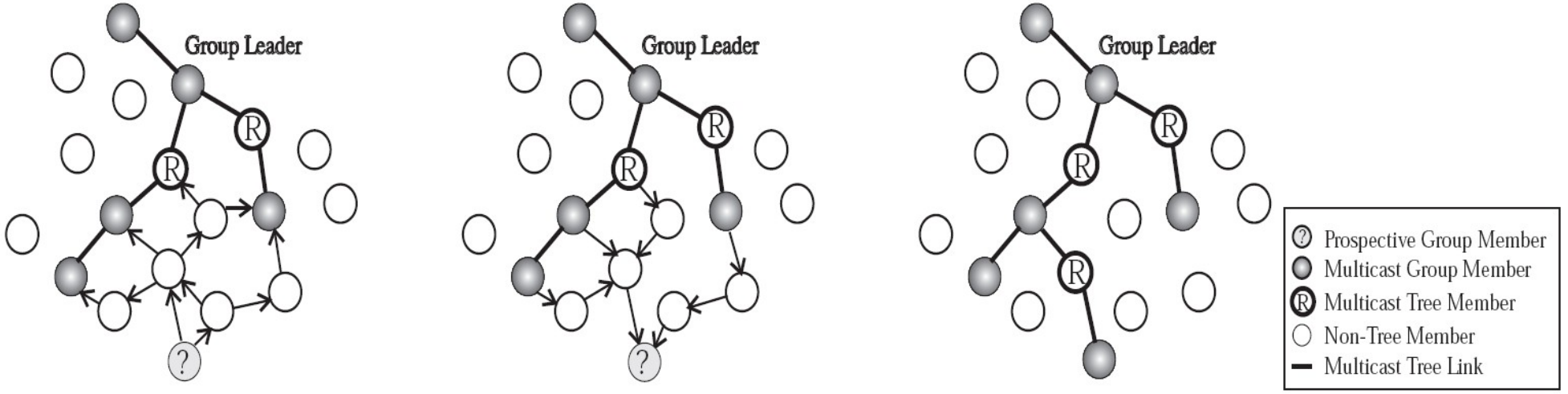


Multicast

Multicast = jeden používateľ posiela identickú informáciu podmnožine používateľov



MAODV: pristúpenie



(a) RREQ Message Propagation (b) RREPs sent back to source (c) Multicast Tree Branch Addition

Multicast strom obsahuje aj uzly, ktoré nie sú členom multicast skupiny.

Obrázok zdroj:Elizabeth M. Royer and Charles E. Perkins. Multicast Operation of the Ad-hoc On-Demand Distance Vector Routing Protocol. Proceedings of MobiCom '99, Seattle, WA, August 1999, pp. 207-218.

Útoky na ad hoc siete

Prečo: pre získanie prístupu k službám siete, z dôvodu šetrenia vlastných zdrojov, pripadne z dôvodu poškodenia siete.

Ako: útok jediným uzlom alebo skupinou uzlov.

Detekcia: pomocou uzlov, ktoré nie sú ovládané útočníkom.

Útok môže byť prevedený pomocou laptopov, uzly samotnej siete môžu byť v porovnaní výrazne menej výkonné a s obmedzenými zdrojmi (napr. napájané batériou).

Detekcia

- Signatúra: potrebná distribúcia, čo v prípade ad hoc sietí môže byť energicky náročné.

- Anomálie: systém sa učí, čo je normálne a signalizuje odchýlku od normálneho správania.

Reputačné systémy: výpočet reputácie uzla na základe jeho správania.

Anomaly (Merriam-Webster): something different, abnormal, peculiar, or not easily classified.

Súvislosť siete

Algorithm CONNECT

Input: (1) Multihop wireless network M = (N, L) (2) Least-power function λ

Output: Power levels p for each node that induces a connected graph

begin

1. sort node pairs in non-decreasing order of mutual distance
2. initialize $|N|$ clusters, one per node
3. **for each** (u,v) in sorted order **do**
4. **if** cluster(u) $\neq$ cluster(v)
5. $p(u) = p(v) = distance(u, v)$
6. merge cluster(u) with cluster(v)
7. **if** number of clusters is 1 **then end**
8. perNodeMinimalize(M, λ, p, 1)

end

Definition II.4: The *least-power* function $\lambda(d)$ gives the minimum power needed to communicate a distance of d.

Algorithm from: Ramanathan, R.; Rosales-Hain, R.; , "Topology control of multihop wireless networks using transmit power adjustment," INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE , vol.2, no., pp.404-413 vol.2, 2000

Súvislosť siete

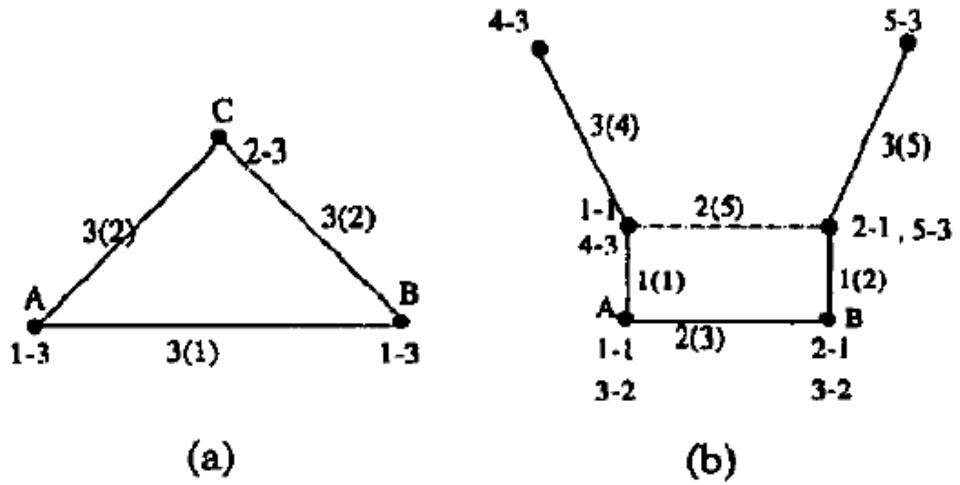


Fig. 1. Illustrating side effect edges. Side effect edges are shown with dashed lines. Legend for nodes is $s-p$, where s is the step number, and p is the power assigned during that step. Legend for edges is $d(s)$, where d is the distance between corresponding nodes, and s is the step during which this edge was formed. Figure (a) is per-node minimal, but in figure (b), the powers of A and B can be reduced back to 1 and still keep the graph connected.

Súvislosť siete

Procedure perNodeMinimalize(M, λ, p, k)
begin
1. let S = sorted node pair list
2. **for** each node u do
3. T = { $(n_1, n_2) \in$ S : $u = n_1$ or $u = n_2$ }
4. sort T in non-increasing order of distance
5. discard from T all (x, y) such that $\lambda(d(x, y)) > p(u)$
6. **for** $(x, y) \in$ T using binary search **do**
7. **if** graph with $p(u) = \lambda(d(x, y))$ is not k-connected, **stop**
8. **else** $p(u) = \lambda(d(x, y))$

end

Interakcie protokolov: riešenia

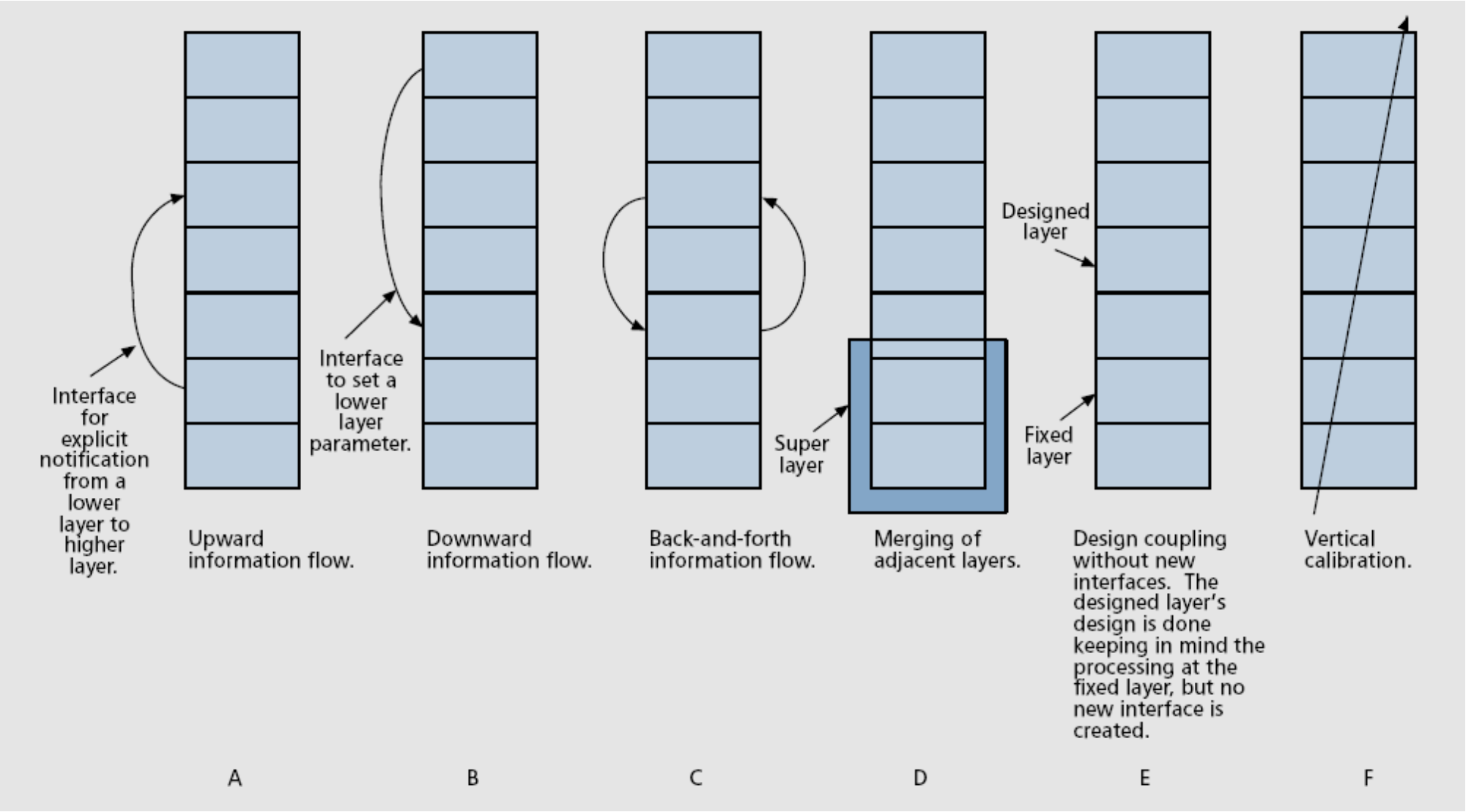


Figure source: Srivastava, V.; Motani, M.; , "Cross-layer design: a survey and the road ahead," Communications Magazine, IEEE , vol.43, no.12, pp. 112- 119, Dec. 2005

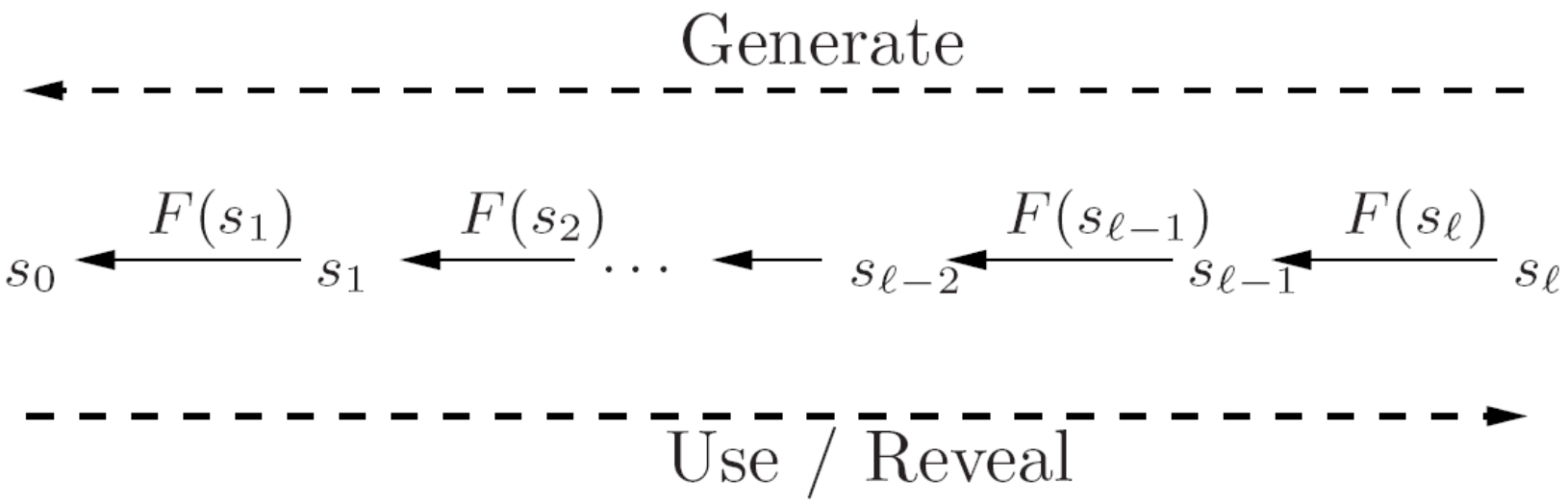
Neistota pri posielaní správ

Time	Magnitude	Distribution
Send and Receive	0 – 100 ms	nondeterministic, depends on the processor load
Access	10 – 500 ms	nondeterministic, depends on the channel contention
Transmission / Reception	10 – 20 ms	deterministic, depends on message length
Propagation	< 1µs for distances up to 300 meters	deterministic, depends on the distance between sender and receiver
Interrupt Handling	< 5µs in most cases, but can be as high as 30µs	nondeterministic, depends on interrupts being disabled
Encoding plus Decoding	100 – 200µs, < 2µs variance	deterministic, depends on radio chipset and settings
Byte Alignment	0 – 400µs	deterministic, can be calculated

Timed efficient stream loss-tolerant authentication (TESLA)

Jednosmerné hašovanie:

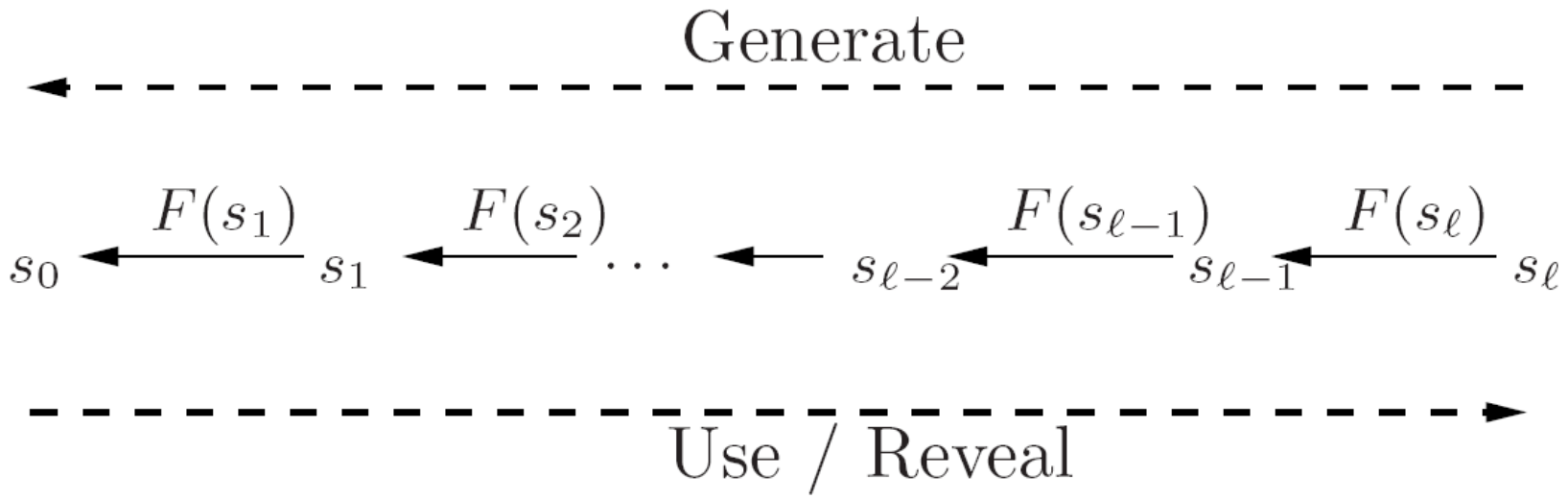
- Na generovanie jednosmernej reťazi hašov použijeme jednosmernú hašovaciu funkciu, napr. gen. náhodných čísel
- Hašovacia funkcia je použitá L-krát
- Kľúče s_i sú použité pre autentifikáciu správ (MAC = Message authetification code)



Obrázok zdroj: Perrig, A. and Canetti, R. and Tygar, JD and Song, D. The TESLA Broadcast Authentication Protocol, RSA CryptoBytes, 2002.

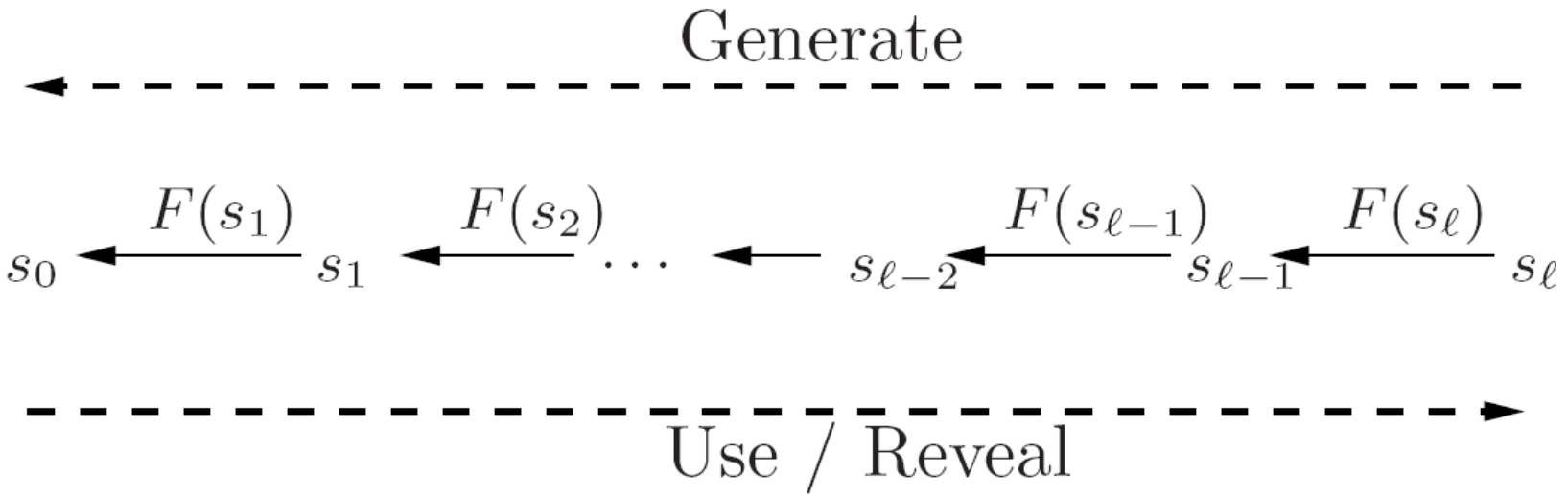
Timed efficient stream loss-tolerant authentication (TESLA)

- Kľúče sú použité a zverejnené v opačnom poradí ako sú generované
- Kľúče sú zverejnené s časovým oneskorením
- Uzol pred overením správy overí, či kľúč už bol zverejnený, ak áno, správa je zmazaná



Timed efficient stream loss-tolerant authentication (TESLA)

- Ak sa kľúč stratí, je možné ho generovať z neskoršie zverejnených kľúčov
- TESLA vyžaduje približnú synchronizáciu času, odosielateľ zverejní čas zverejnenia kľúča (oneskorenie po jeho použití)



Približná synchronizácia času

Synchronizácia odosielateľa S a prijímateľa R

- R pošle S: Nonce
- S pošle R: {ts, Nonce} podpísaný privátnym kľúčom S
- R overí pomocou verejného kľúča S, či je správa podpísaná uzlom S

Horné ohraničenie času odosieľatela: $t - t_R + t_S$

Nonce je náhodný bit-string s dĺžkou h

t = čas teraz; t_S = čas odosielateľa; t_R = čas prijímateľa

Ďalšia možnosť: „microcomputer-compensated crystal oscillator“.
Presnosť v sekundách v trvaní niekoľko mesiacov.

Ďakujem

Otázky