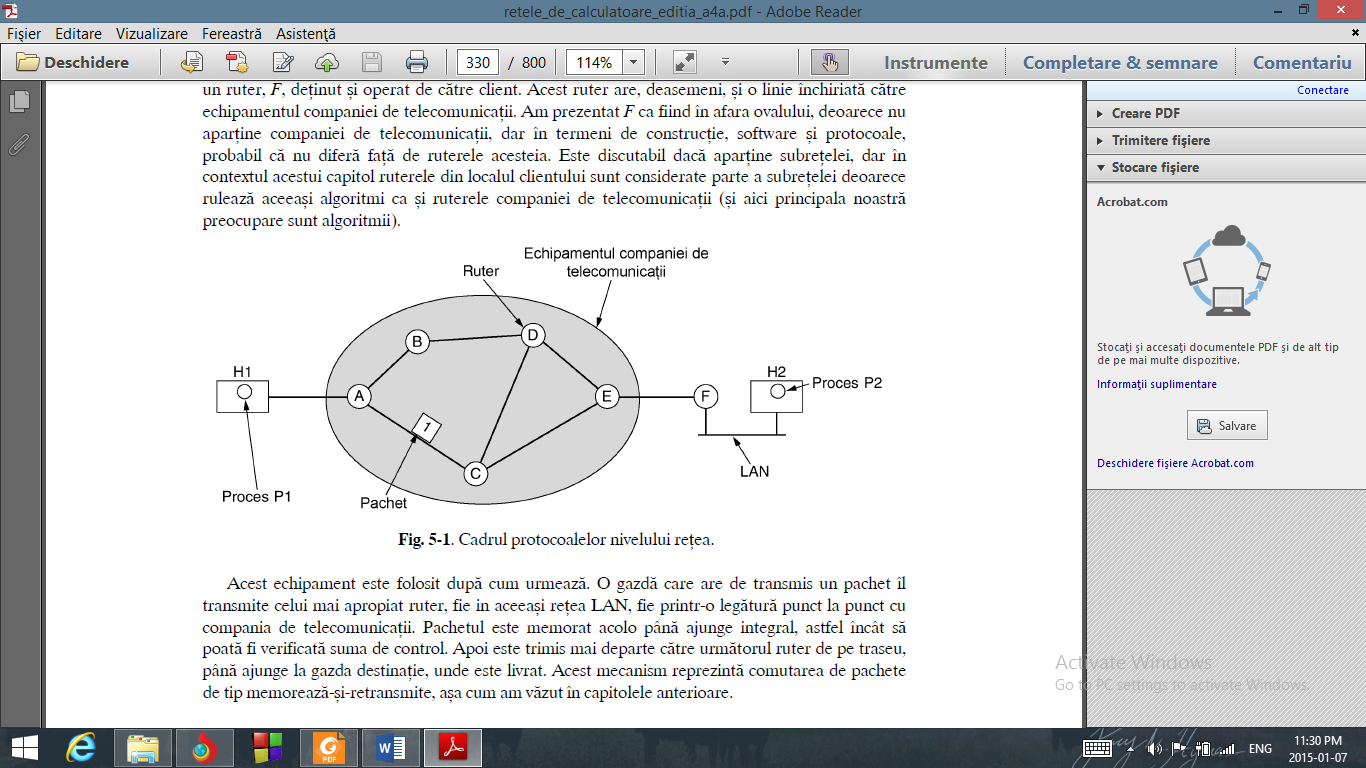
3. Nivelul retea

3.a. CERINŢELE DE PROIECTARE ALE NIVELULUI REŢEA

# Comutare de pachete de tip Memorează-şi-Retransmite (Store-and-Forward)

Componentele majore ale sistemului sunt echipamentul companiei de telecomunicaţii (rutere  
conectate prin linii de transmisie), prezentat în interiorul ovalului umbrit, şi echipamentul  
clientului, prezentat în afara ovalului. Gazda H1 este conectată direct la unul dintre ruterele  
companiei de telecomunicaţii, A, printr-o linie închiriată. În contrast, H2 este într-o reţea LAN cu un ruter, F, deţinut şi operat de către client. Acest ruter are, deasemeni, şi o linie închiriată către echipamentul companiei de telecomunicaţii.



O gazdă care are de transmis un pachet îl transmite celui mai apropiat ruter, fie in aceeaşi reţea LAN, fie printr-o legătură punct la punct cu compania de telecomunicaţii. Pachetul este memorat acolo până ajunge integral, astfel încât să poată fi verificată suma de control. Apoi este trimis mai departe către următorul ruter de pe traseu, până ajunge la gazda destinaţie, unde este livrat. Acest mecanism reprezint comutarea de pachete de tip memorează-şi-retransmit.

# Servicii furnizate nivelului transport

Serviciile nivelului reţea au fost proiectate având în vedere următoarele scopuri:

1. Serviciile trebuie să fie independente de tehnologia ruterului.
2. Nivelul transport trebuie să fie independent de numărul, tipul şi topologia ruterelor existente
3. Adresele de reţea disponibile la nivelul transport trebuie să folosească o schemă de numerotare uniformă, chiar în cadrul reţelelor LAN şi WAN.

Problema centrală a discuţiei este dacă nivelul reţea trebuie să furnizeze servicii orientate pe conexiune sau servicii neorientate pe conexiune.

O tabără (reprezentată de comunitatea Internet) afirmă că scopul ruterului este de a transfera pachete şi nimic mai mult deoarece subreţeaua este inerent nesigură, indiferent cum ar fi proiectată.. Acest punct de vedere duce rapid la concluzia că serviciul reţea trebuie să fie neorientat pe conexiune, cu două primitive SEND PACKET şi RECEIVE PACKET şi cu foarte puţin în plus.

Cealaltă tabără (reprezentată de companiile de telefoane) afirmă că subreţeaua trebuie să asigure un serviciu orientat pe conexiune sigur.

# Implementarea serviciului neorientat pe conexiune

Sunt posibile două organizări diferite, în funcţie de tipul serviciului oferit. Dacă este oferit un serviciu neorientat pe conexiune, atunci pachetele sunt trimise in subreţea individual şi dirijate independent de celelalte.

In acest context, pachetele sunt numite frecvent datagrame (datagrams) (prin analogie cu telegramele), iar subreţeaua este numită subreţea datagramă (datagram subnet). Dacă este folosit serviciul orientat conexiune, atunci, înainte de a trimite pachete de date, trebuie stabilită o cale de la ruterul sursă la ruterul destinaţie

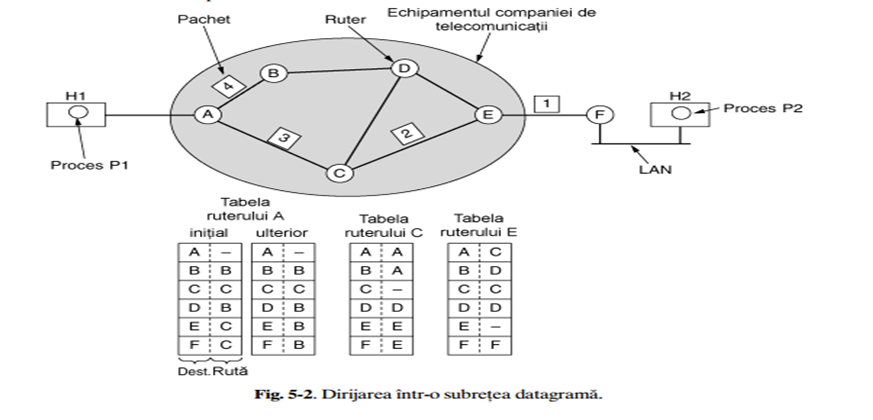
Această conexiune este numită VC (virtual circuit, circuit virtual), prin analogie cu circuitele fizice care se stabilesc în sistemul telefonic, iar subreţeaua este numită subreţea cu circuite virtuale (virtual-circuit subnet).

Să presupunem că procesul P1 din fig. 5-2 are un mesaj lung pentru procesul P2. El transmite mesajul nivelului transport, cu instrucţiunile de livrare către procesul P2 aflat pe calculatorul gazdă H2. Codul nivelului transport rulează pe calculatorul gazdă H1, de obicei în cadrul sistemului de operare.

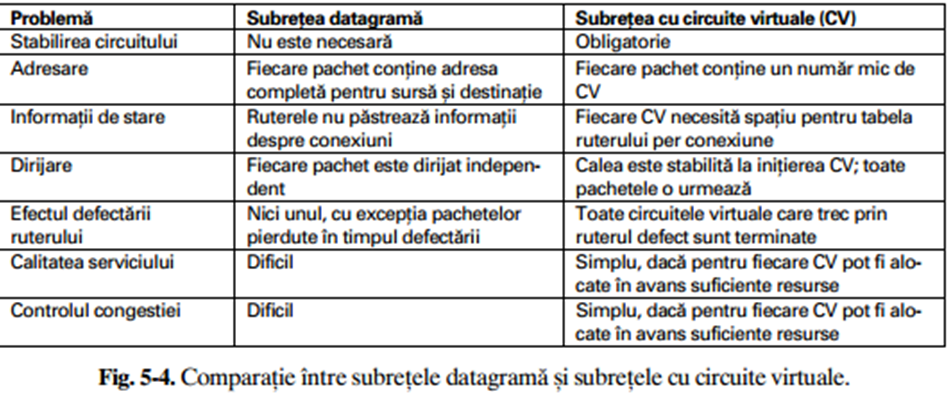
Acesta inserează la începutul mesajului un antet corespunzător nivelului transport şi transferă rezultatul nivelului reţea, probabil o altă procedură din cadrul sistemului de operare. Să presupunem că mesajul este de patru ori mai lung decât dimensiunea maximă a unui pachet, aşa că nivelul reţea trebuie să îl spargă în patru pachete, 1, 2, 3, şi 4 şi să le trimită pe fiecare în parte ruterului A, folosind un protocol punct-la-punct, de exemplu, PPP. Cum au ajuns la A, pachetele 1, 2 şi 3 au fost memorate pentru scurt timp (pentru verificarea  
sumei de control). Apoi fiecare a fost trimis mai departe către C conform tabelei lui A. Pachetul 1 a fost apoi trimis mai departe către E şi apoi către F. Când a ajuns la F, a fost încapsulat într-un cadru al nivelului legătură de date şi trimis către calculatorul gazdă H2 prin reţeaua LAN.

# Implementarea serviciilor orientate pe conexiune

Pentru serviciile orientate conexiune, avem nevoie de o subreţea cu circuite virtuale. Ideea care se stă la baza circuitelor virtuale este evitarea alegerii unei noi căi (rute) pentru fiecare pachet trimis. În schimb, atunci când se stabileşte o conexiune, se alege o cale între maşina sursă şi maşina destinaţie, ca parte componentă a iniţializării conexiunii şi aceasta este memorată în tabelele ruterelor.



# Comparaţie între subreţele cu circuite virtuale şi subreţele datagram



În interiorul subreţelei există situaţii în care trebuie să se aleagă între facilităţi antagoniste specifice fie circuitelor virtuale, fie datagramelor. Un astfel de compromis este acela între spaţiul de memorie al ruterului şi lăţimea de bandă.

Alt compromis este cel între timpul necesar stabilirii circuitului şi timpul de analiză a adresei. Folosirea circuitelor virtuale presupune existenţa unei faze iniţiale de stabilire a căii, care cere timp şi consumă resurse.

O altă problemă este cea a dimensiunii spaţiului necesar pentru tabela din memoria ruterului. O subreţea datagramă necesită o intrare pentru fiecare destinaţie posibilă, în timp ce o reţea cu  
circuite virtuale necesită o intrare pentru fiecare circuit virtual.

3.b. Algoritmi de dirijare

Algoritmul de dirijare (routing algorithm) este acea parte a software-ului nivelului reţea care  
răspunde de alegerea liniei de ieşire pe care trebuie trimis un pachet recepţionat

# Calea cea mai scurta

Modelul topologic al unei retele este un graf in care nodurile corespund comutatoarelor de pachete, iar muchiile corespund liniilor de comunicatie.

Asociind fiecarei muchii o lungime, se poate calcula calea cea mai scurta intre oricare doua noduri, deci cea mai indicata pentru dirijarea pachetelor intre nodurile respective (algoritmul lui Dijkstra).

Lungimea poate avea diverse semnificatii. Daca toate liniile au lungimea unu, gasim caile cu numar minime de noduri intermediare.

Lungimea poate fi distanta geografica intre noduri, costul comunicatiei, intirzierea medie masurata etc

* Algoritmul lui Dijkstra gaseste caile cele mai scurte de la o sursa la toate celelalte noduri.
* El trebuie sa dispuna de informatii topologice generale asupra retelei: listele nodurilor si legaturilor,costurile asociate legaturilor.
* Prin natura sa el este centralizat
* Algoritmul este iteratiasi calculeaza la fiecare iteratie ë cea mai scurta cale de la sursa la un nod al retelei.

# Dirijarea centralizata

Varianta centralizata a algoritmului drumurilor minime (Floyd) utilizeaza un tablou A al distantelor minime, A[i][j] fiind distanta minima de la nodul i la nodul j. Initial,

Calculul drumurilor minime se face iterativ.

La iteratia k, A[i][j] va avea ca valoare cea mai buna distanta intre i si j, pe cai care nu contin noduri numerotate peste k (exceptind i si j).

Deficientele acestei metode sint determinate de:

* vulnerabilitatea retelei,dependenta de functionarea centrului de control (se recurge la dublarea lui);
* supraincarcarea traficului prin transmiterea rapoarelor si a tabelelor de dirijare;
* utilizarea in noduri,in anumite perioade, a unor tabele necorelate, datorita receptiei la momente de timp distincte a noilor tabele.

# Dirijarea izolata

Pachetul receptionat de nod este plasat in coada cea mai scurta. (O varianta ia in consideratie lungimea cozilor anumitor linii, selectate conforme cailor celor mai scurte)

Desi nepractic, algoritmul este folosit in aplicatii militare (datorita robustetii sale) sau in comparatii de performanta cu alte tehnici (deoarece are un timp de intirziere minim).

# Dirijarea distribuita

Varianta modificata a algoritmului lui Dijkstra care calculeaza drumurile minime de la toate nodurile catre o anumita destinatie.

Conduce in mod natural la o varianta descentralizata

Algoritmul este convergent,asigurind gasirea drumurilor minime intr-un numar finit de pasi

Poate fi utilizat doar pentru datagrame.

# Dirijarea ierarhica

Se utilizeaza pentru retele de mari dimensiuni la care tabelele de dirijare ar fi voluminoase.

Comutatoarele sint grupate in regiuni,fiecare comutator cunoscind detaliat caile din regiunea proprie,dar necunoscind structura interna a altor regiuni.

Doua regiuni sint legate prin conectarea unui anumit nod din prima regiune cu un anumit nod din a doua regiune.

Tabela de dirijare se poate reduce,ea avind cite o intrare pentru fiecare nod din regiunea proprie si cite o intrare pentru fiecare din celelalte regiuni.

# Inundarea

Un alt algoritm static este inundarea (flooding), în care fiecare pachet recepţionat este trimis mai  
departe pe fiecare linie de ieşire, cu excepţia celei pe care a sosit.

Inundarea generează un mare număr de pachete duplicate, de fapt un număr infinit dacă nu se iau unele măsuri pentru a limita acest proces.

3.c. ALGORITMI PENTRU CONTROLUL CONGESTIEI

Atunci cand mai multe pachete sunt prezente intr-o subretea performatele se degradeaza. Aceasta situatie se numeste congestie. Congestia poate fi produsă de mai mulţi factori. Dacă dintr-o dată încep să sosească şiruri de pachete pe trei sau patru linii de intrare şi toate necesită aceeaşi linie de ieşire, atunci se va forma o coadă. Dacă nu există suficientă memorie pentru a le păstra pe toate, unele se vor pierde. Şi procesoarele lente pot cauza congestia.

# Solutii:

1. **Prealocarea zonelor tampon**
   * Este aplicabila circuitelor virtuale si consta in rezervarea uneia sau mai multor zone tampon in fiecare nod intermediar, la deschiderea circuitului. In lipsa de spatiu, se alege o alta cale sau se rejecteaza cererea de stabilire a circuitului.
2. **Distrugerea pachetelor**
   * Daca nu exista spatiul necesar memorarii, pachetul receptionat de un nod este ignorat. Deoarece astfel se pot ignora pachete de confirmare, care ar duce la eliberarea spatiului ocupat de pachetele confirmate, se mentine cel putin un tampon de receptie pentru fiecare linie, permitindu-se inspectarea pachetelor primite. De asemenea, se poate limita numarul zonelor tampon de transmisie ale fiecarei linii.
3. **Pachete de permisiune.**
   * Se initializeaza reteaua cu pachete de permisiune (in numar fix). Cind un nod vrea sa transmita, el captureaza un pachet de permisiune si trimite in locul lui pachete de date. Receptorul regenereaza pachetul de permisiune. Se garanteaza astfel ca numarul maxim de pachete nu depaseste numarul de pachete de permisiune, fara a se asigura distribuirea lor conform necesitatilor nodurilor. In plus, pierderea pachetelor de permisiune conduce la scaderea capacitatii retelei.
4. **Pachete de soc.** 
   * Sunt transmise de comutatoare surselor de date pentru a micsora rata de generare a pachetelor.

# Evitarea blocarii definitive.

Blocarea reprezinta o situatie limita a unei retele congestionate, cind lipsa de spatiu impiedica transmiterea vreunui pachet.

O solutie de evitare a blocarii definitive este utilizarea in fiecare nod a m+1 zone tampon, m fiind lungimea maxima a cailor retelei. Un pachet sosit de la calculatorul gazda local este acceptat in zona 0. In urmatorul nod trece in 1, apoi in 2 s.a.m.d. Zona "m" a unui nod poate fi goala, poate contine un pachet pentru gazda locala, care este livrat, sau are un pachet pentru un nod distant, care este distrus. In toate cazurile zona "m" se elibereaza, putind avansa un pachet din zona "m-1",apoi "m-2" etc.

Alta varianta pastreaza pentru fiecare pachet o informatie de vechime. La comunicarea dintre doua noduri A si B putem intilni situatiile urmatoare (presupunem ca A are de transmis lui B un pachet mai vechi decit B catre A):

* B are un tampon liber si poate primi cel mai vechi pachet al lui A catre B;
* B nu are un tampon liber, dar are un pachet pentru A si poate primi, prin schimb, cel mai vechi pachet al lui A catre B;
* B nu are nici un tampon liber si nici un pachet catre A; in acest caz, B este fortat sa transmita lui A un pachet la alegere si sa primeasca cel mai vechi pachet al lui A catre B

3.d. Protocolul ip

O datagramă IP constă dintr-o parte de antet şi o parte de text. Antetul are o parte fixă de 20 de octeţi şi o parte opţională cu lungime variabilă. El este transmis în ordinea **big endian** (cel mai semnificativ primul): de la stânga la dreapta, începând cu bitul cel mai semnificativ al câmpului Versiune. (Procesorul SPARC este de tip bigendian; Pentium este de tip **little endian** - cel mai puţin semnificativ primul). Pe maşinile de tip little endian, este necesară o conversie prin program atât la transmisie cât şi la recepţie.

* Utilizat de sisteme autonome in vederea interconectarii
* Serviciu de transmitere de pachete (host­to­host)
* Translatare dintre diferite protocoale legatura de date
* Ofera servicii neorientate‐conexiune, nesigure: datagrame
* Fiecare datagrama este independenta de celelalte
* Nu se garanteaza transmiterea corecta a datagramelor (pierdere, multiplicare,...)
* Foloseste doar adresele logice ale gazdelor
* Adresele IP nu sunt identice cu cele ale nivelului MAC (e.g., adresele hardware ale placilor de retea) pentru ca IP trebuie sa suporte diferite implementari hardware (retele eterogene)