

Sveučilište u Zagrebu  
Fakultet elektrotehnike i računarstva  
Ak.g. 2021./22.  
Komunikacijski protokoli

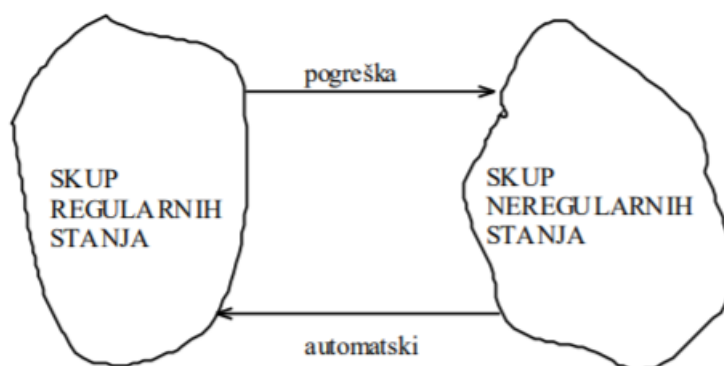
Ignac Lovrek  
Modeli telekomunikacijskih procesa - teorija i primjena Petrijevih mreža  
Školska knjiga, Zagreb, 1997.

#### 4. ANALIZA I SINTEZA KOMUNIKACIJSKIH PROTOKOLA

##### 4.1. METODE RAZVOJA KOMUNIKACIJSKIH PROTOKOLA

Komunikacijski postupak kojim se ostvaruje izmjena podataka, uklanjanje posljedica kvarova i smetnji te koordinacija u sustavu komunicirajućih procesa naziva se komunikacijski protokol. Djelovanje smetnji te pojava pogrešaka i kvarova su realnost tako da je osnovna pretpostavka pri razvoju protokola "sačuvati upravljanje", a time i integritet podataka i operativnost sustava.

Promatrajući stanja komunicirajućih procesa, takav se zahtjev može formulirati kao povratak u regularno stanje u konačnom vremenu (s konačnim brojem operacija) iz neregularnog stanja u koje je sustav prešao pod djelovanjem pogreške (sl.4.1).



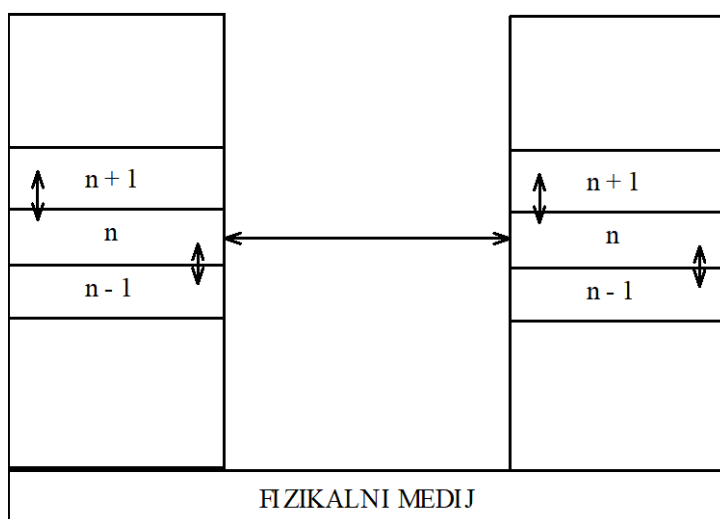
*Slika 4.1. Uklanjanje pogreške u sustavu komunicirajućih procesa*

Uobičajeno predočavanje komunikacije između udaljenih procesa osniva se na primjeni slojevitih modela. U slojevitom se modelu ukupne operacije podijele na više slojeva za svaki komunicirajući entitet, odnosno stranu u komunikaciji. Svaki sloj sadrži skup funkcija kojima, na osnovi postavljenog zahtjeva, posluhuje viši sloj, a sam postavlja zahtjeve nižem sloju. Između dva susjedna sloja na istoj strani se provodi vertikalna komunikacija. Horizontalna komunikacija unutar istog sloja na različitim stranama je virtualna komunikacija, osim u najnižem sloju (sl.4.2).

Općenito, posrijedi je sedam slojeva, i to:

1. fizikalni sloj,
2. sloj podatkovne veze,
3. mrežni sloj,
4. transportni sloj,
5. sloj sesije,
6. sloj prikaza,
7. sloj primjene,

od kojih je fizikalni sloj najniži, a sloj primjene najviši.



*Slika 4.2. Slojeviti model komunikacije*

U telekomunikacijskoj mreži komunicirati mogu korisnici međusobno, zatim korisnik i funkcijski entitet mreže, korisnik i funkcijski entitet u mreži ili izvan nje, funkcijski entiteti u mreži te funkcijski entitet mreže i funkcijski entitet smješten izvan nje. Model povezivanja i izmjene korisničke i upravljačke informacije u digitalnoj mreži integriranih usluga naziva se referentnim modelom protokola.

Svrha komunikacije jest ostvarivanje telekomunikacijskih usluga - prijenosnih usluga, teleusluga i dodatnih usluga. Zato referentni model protokola koji se zasniva na slojevitom prikazu ima posebna obilježja koja omogućuju predočavanje različitih usluga u komunikaciji glasom, slikom, podacima i s više oblika korisničke informacije te njima pridružene upravljačke informacije.

Razlikovanjem informacijskih tokova uvode se dvije ravnine:

- upravljačka ili C-ravnina (*control*) i
- korisnička ili U-ravnina (*user*).

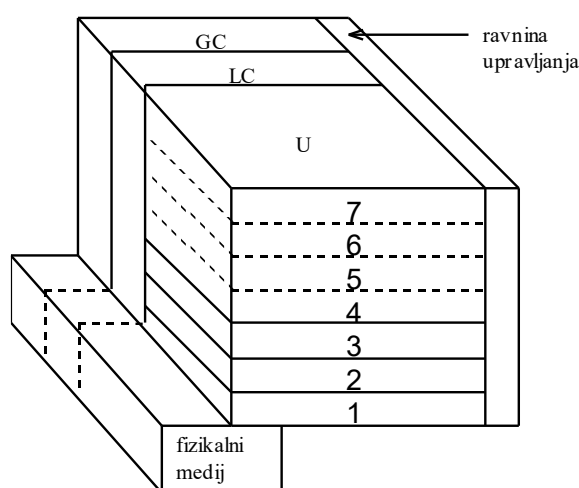
Osnovna zadaća korisničke ravnine jest prijenos informacija između korisnika, a za upravljačku ravninu to je upravljanje vezom, upravljanje uporabom veze i pridjeljivanje dodatnih usluga. Takav prikaz protokola dopušta rad s upravljačkim tokom izdvojenim iz korisničkog toka (*out-of-band-signalling*) ili pomiješanim s korisničkim tokom (*in-band signalling*).

Kad je riječ o upravljačkoj informaciji, važno je razlikovati njezino lokalno i globalno značenje. Izvorišni funkcijski entitet se pri ostvarivanju telekomunikacijske usluge povezuje s bliskim (pridruženim) entitetima i udaljenim entitetima koji su spojeni s bliskim entitetima. Upravljačka informacija koja se odnosi na izvorišni i bliski entitet ima lokalno značenje, a ako je riječ o udaljenom entitetu globalno značenje. Usluga ima globalno, a upravljanje resursima lokalno značenje u upravljačkoj ravnini. Dodatna usluga može imati globalno ili lokalno značenje. Globalna informacija može se prenositi transparentno, obrađivati ne zadirući u sadržaj ili mijenjajući sadržaj.

Pojmovi ravnine i njezina značenja proširuju pojam sloja i temeljna su obilježja protokolnog bloka kojim se opisuju različiti dijelovi mreže. Generički protokolni blok (sl 4.3) sadrži:

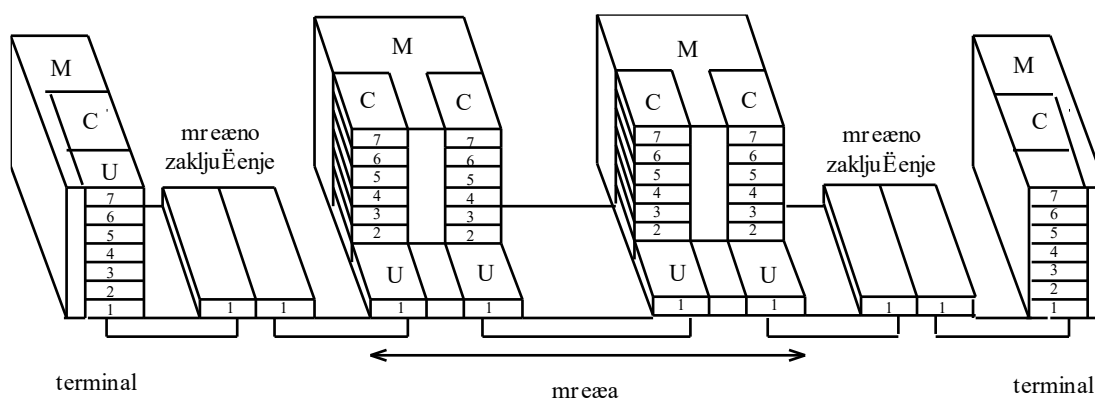
- lokalnu upravljačku ravninu LC
- globalnu upravljačku ravninu GC
- korisničku ravninu U,

svaku podijeljenu u slojeve. Neki slojevi mogu biti prazni, neupotrijebljeni, odnosno bez zadaće u pojedinoj primjeni. Isto se tako u pojedinoj primjeni može pojaviti samo jedna ili dvije ravnine, a ne sve tri. Suradnja ravnina u jednome protokolnom bloku ostvaruje se funkcijom upravljanja ravnina. Protokolni blokovi međusobno se povezuju fizikalnim medijem.



Slika 4.3. Protokolni blok

Primjer komunikacije između dva terminala priključena na čvorove mreže predložen je slikom 4.4. Odabran je prikaz komunikacije u digitalnoj mreži integriranih usluga - izmjena korisničke informacije komutacijom kanala po B kanalu između dva terminala. Uvedene su tri vrste protokolnih blokova koji opisuju terminal, mrežno zaključenje i čvor u mreži koji se razlikuju prema ravninama i slojevima. Terminalni protokolni blok sadrži sve tri ravnine i svih sedam slojeva čime je omogućena teleusluga komunikacije B kanalom. Protokolni blok mrežnog zaključanja reduciran je na prvi sloj što odgovara funkcijama za prijenos informacije. Mrežni protokolni blok ima također korisničku ravninu samo u prvom sloju (transparentnost s obzirom na korisničku informaciju!), a ostale ravnine u punom sastavu, do sloja primjene.

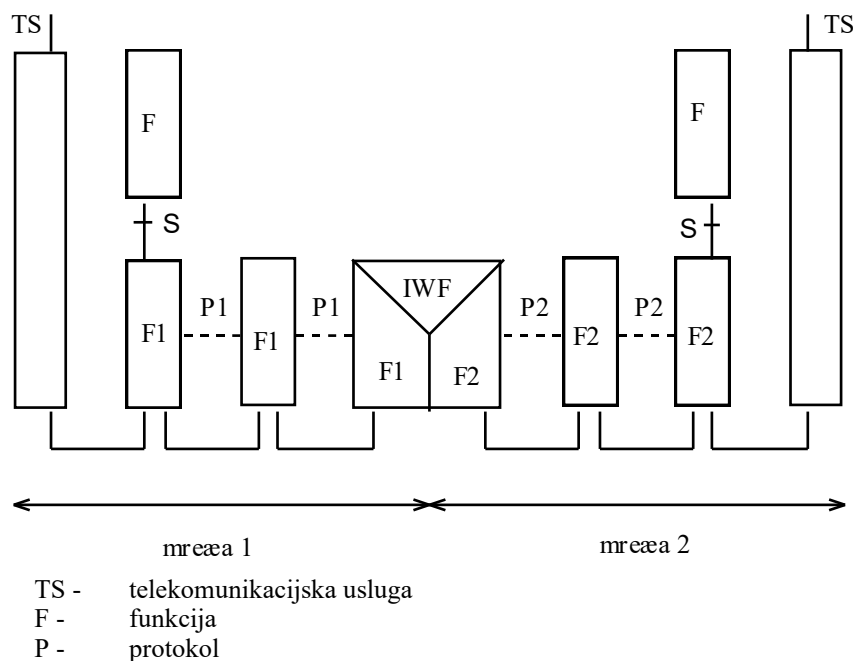


Slika 4.4. Informacijski tok između dva terminala

Međudjelovanje raznovrsnih protokolnih blokova (različitih protokola, različitih mreža) ostvaruje se funkcijama međudjelovanja (IWF - *interworking*). Načelo međudjelovanja dviju mreža pri ostvarivanju telekomunikacijske usluge predloženo je slikom 4.5. Moguće su ove situacije:

- usluga S koja se izvodi jednaka je zahtijevanoj usluzi TS, jer je obje mreže mogu ostvariti; prilagodna funkcija F nije potrebna
- usluga S koja se izvodi različita je od zahtijevane usluge TS, jer su se obje mreže tako dogovorile; prilagodna funkcija F nije potrebna
- uslugu S ostvaruje svaka mreža drugim sredstvima, tako da je potrebna prilagodna funkcija F.

Usluga S ostvaruje se funkcijama F1 i F2 te protokolima P1 i P2, a IWF izvodi preslikavanje F1 - F2.



*Slika 4.5. Međudjelovanje dviju mreža*

Komunikacijski protokoli su redovito složeni i sadrže velik skup pravila za izmjenu informacijskih jedinica u sustavu koji je distribuiran. Koncept protokolnog bloka i slojevita struktura pomažu da se ukupni problem raščlani i tako složenost učini prihvatljivom.

Složenost postupaka razvoja protokola može se smanjiti i preciznim definiranjem razvojnih faza i s njima povezanog izbora formalnog modela. Pri razvoju protokola provodi se:

- specifikacija (definiranje i formalizacija opisa protokola)
- verifikacija (provjera specificiranog protokola)
- implementacija (izvedba protokola na temelju provjerene specifikacije), te
- ispitivanje podudarnosti (provjera izvedenog u odnosu prema specificiranom protokolu).

Ključni problem je kako formalno specificirati protokol i time omogućiti njegovu provjeru i izvedbu. Pri tome treba imati na umu dvije osnovne funkcije protokola - upravljanje komunikacijom i prijenos informacijskih jedinica. Pri specifikaciji te se funkcije manifestiraju kao definicija upravljačkih i proceduralnih aspekata komunikacije koje je teško obuhvatiti istim formalnim modelom. Tipične upravljačke funkcije su uspostavljanje i prekidanje komunikacije, akcije pri pojavi pogrešaka i kvarova i tome slično. Procedurnim se funkcijama rješavaju označavanje i odbrojavanje informacijskih jedinica, broj informacijskih jedinica koje odašilje predajna strana iako nije primila potvrdu od prijamne strane, ponavljanje pri neispravnom prijenosu i sl.

Sa stajališta specifikacije, formalizacija protokola razlikuje se prema osnovnom modelu. Najviše postupaka upotrebljava:

- konačni automat i izvedene modele
- Petrijevu mrežu i izvedene modele
- grafičke modele
- visoke programske jezike
- formalne jezike i gramatike, ili
- kombinirane modele.

Uzme li se u obzir i provjera (verifikacija) protokola, veliku važnost imaju dvije skupine modela - oni koji se oslanjaju na koncept sustava uvjeta i događaja i oni koji su jezično orijentirani. Pritom su automat i Petrijeva mreža povoljniji za upravljačke, a visoki programski jezik za procedurene aspekte. U 2. poglavlju ove knjige razrađena su osnovna načela modeliranja komunikacijskih protokola konačnim automatom

Provjerom protokola potrebno je ustanoviti:

- nepotpunost definicije
- neregularno upravljanje komunikacijom:
  - nepredviđena interakcija komunicirajućih procesa,
  - dijelovi procesa koji se nikad ne izvode,
  - pogrešna inicijalizacija i sinkronizacija procesa,
  - blokiranje komunikacije.
- neregularni prijenos informacijskih jedinica:
  - gubitak informacijske jedinice,
  - višestruki prijam informacijske jedinice,
  - pogrešan prijam,
  - promjena redoslijeda informacijskih jedinica.
- neregularno upravljanje resursima:
  - prekoračenje kapaciteta kanala,
  - nemogućnost prijama,
  - pogrešna dodjela.
- reakciju na kvarove i druge izvanredne situacije.

U nastavku će biti predloženo modeliranje komunikacijskih protokola koje sadrži specifikaciju i verifikaciju utemeljenu na Petrijevoj mreži kao formalnom modelu. Specifikacija ima obilježja sinteze komunikacijskog protokola, a verifikacija njegove analize.

## 4.2. MODELIRANJE KOMUNIKACIJSKIH PROTOKOLA PETRIJEVOM MREŽOM

### 4.2.1. Model osnovnoga komunikacijskog protokola

Svaki se proces modelira Petrijevom mrežom, a međusobno se povezuju zajedničkim mjestima tako da se kanal također opisuje Petrijevom mrežom. Zajedničko mjesto označava postuvjet za prijelaz koji odašilje informacijsku jedinicu, a preduvjet za prijelaz kojim se ista informacijska jedinica prima.

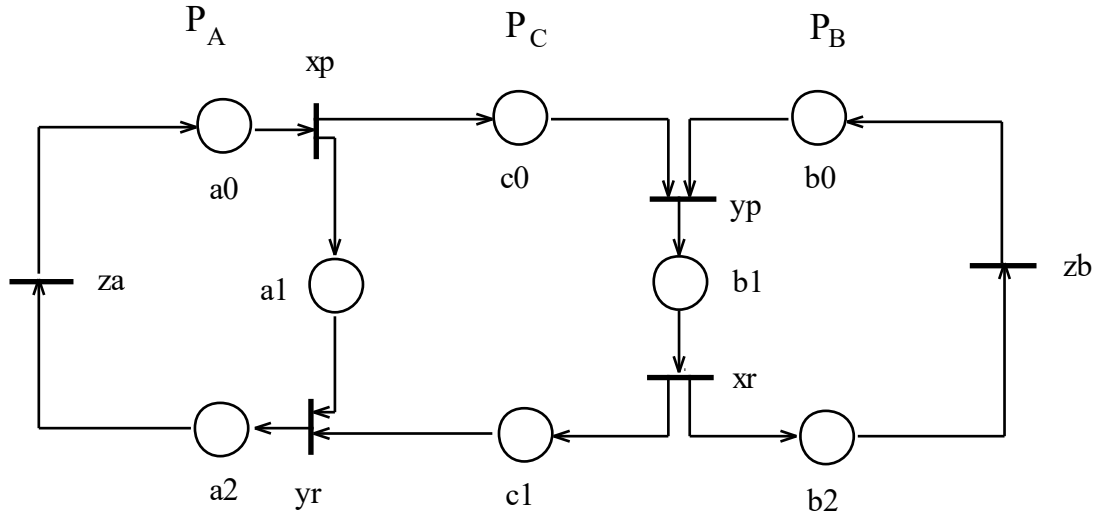
U modelu osnovnoga komunikacijskog protokola dva procesa izmjenjuju poruke i potvrde. Svakom se procesu  $P_A$ ,  $P_B$ , kao i kanalu  $C$  pridjeljuje odgovarajuća Petrijeva mreža  $P_A$ ,  $P_B$  i  $P_C$  (sl.4.6) s mjestima koja opisuju uvjete:

- $a_0$  pripravnost za predaju poruke
- $a_1$  čekanje potvrde
- $a_2$  primljena potvrda
- $b_0$  pripravnost za prijam poruke
- $b_1$  primljena poruka
- $b_2$  predana potvrda
- $c_0$  poruka na kanalu
- $c_1$  potvrda na kanalu

i prijelazima koji opisuju događaje:

- $x_p$  predaja poruke
- $y_r$  prijam potvrde
- $z_a$  unutrašnji prijelaz
- $y_p$  prijam poruke
- $x_r$  predaja potvrde
- $z_b$  unutrašnji prijelaz.

Odabrani je model ekvivalentan onom upotrijebljenom za objašnjenje automat-modela u primjeru 2.2.



Slika 4.6. Izvorna Petrijeva mreža za model protokola

Očito je da je sustav komunicirajućih procesa opisan jednom mrežom koja je nastala povezivanjem pojedinačnih Petrijevih mreža. Komunikacijski kanal je uveden izravno i povezuje komunicirajuće procese. Pritom su utvrđeni uvjeti koji opisuju kanal. Ovaj model je složeniji od automat-modela, ali su njegovi elementi jednostavniji od onih upotrijebljenih u automatu. Stanja nisu pretpostavka za analizu, već će se do njih doći analizom.

Struktura Petrijeve mreže opisuje ovisnost uvjeta i događaja i njihov međusobni utjecaj. To se može nazvati statičkim modelom komunikacijskog protokola. Dinamička komponenta uvodi se izvedbom Petrijeve mreže uz zadane početne uvjete. U tom primjeru to je označavanje:

$$\mu_0 = (\mu_{a0}, \mu_{a1}, \mu_{a2}, \mu_{b0}, \mu_{b1}, \mu_{b2}, \mu_{c0}, \mu_{c1}) = (1, 0, 0, 1, 0, 0, 0, 0)$$

odnosno ispunjeni uvjeti  $a_0$  i  $b_0$ .

U tom stanju može se izvesti samo prijelaz  $x_p$ . Ulazno mjesto gubi jednu oznaku ( $a_0$ ), a izlazna mjesta dobivaju po jednu oznaku ( $a_1, c_0$ ), te se uspostavlja novo stanje

$$\mu_1 = (0, 1, 0, 1, 0, 0, 1, 0)$$

u kojemu su označena mjesta ( $a_1, c_0, b_0$ ).

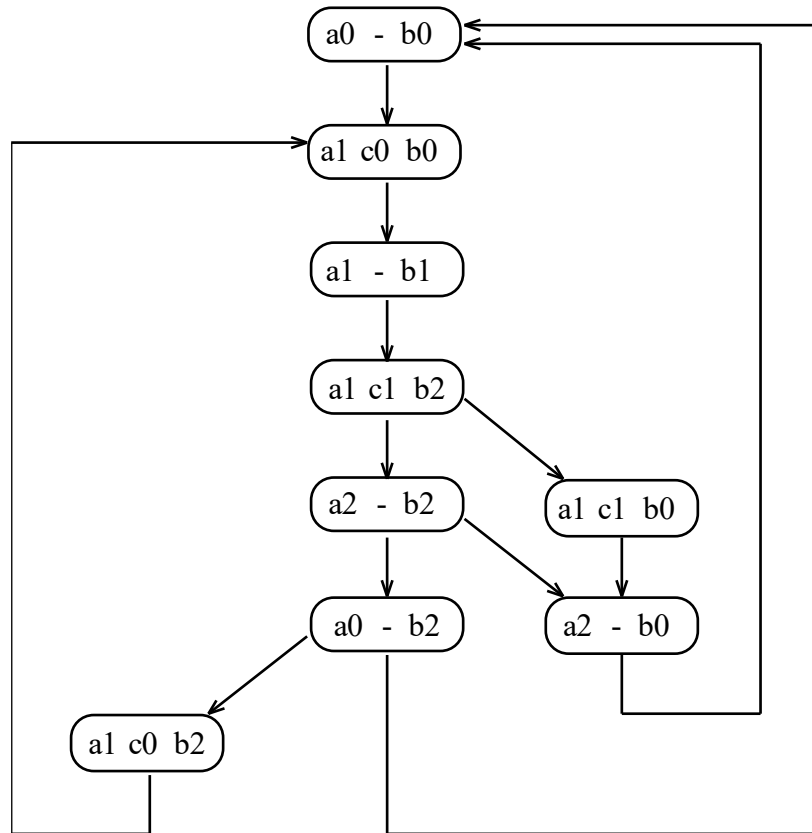
Potpuni graf stanja koji je predložen slikom 4.7 ekvivalentan je grafu stanja za isti komunikacijski protokol modeliran automatom i analiziran postupkom perturbacije stanja. Razlika u složenosti stanja koje sada ima tri komponente, uz pojedinačna stanja automata i stanje kanala, posljedica je izravno uvedenog modela kanala.

Ova je mreža sigurna, aktivna i perzistentna, a nije konzervacijska.



U strukturnom smislu mreža je restriksijska, jer nema višestrukog povezivanja mjesta i prijelaza, ni vlastitih petlji. Svako mjesto ima samo jedan ulazni i samo jedan izlazni prijelaz što odgovara pojmu označenoga grafa.

Prijelazi  $y_r$  i  $z_b$ ,  $x_p$  i  $z_b$  te  $z_a$  i  $z_b$  su paralelni, a konfliktnih prijelaza nema.

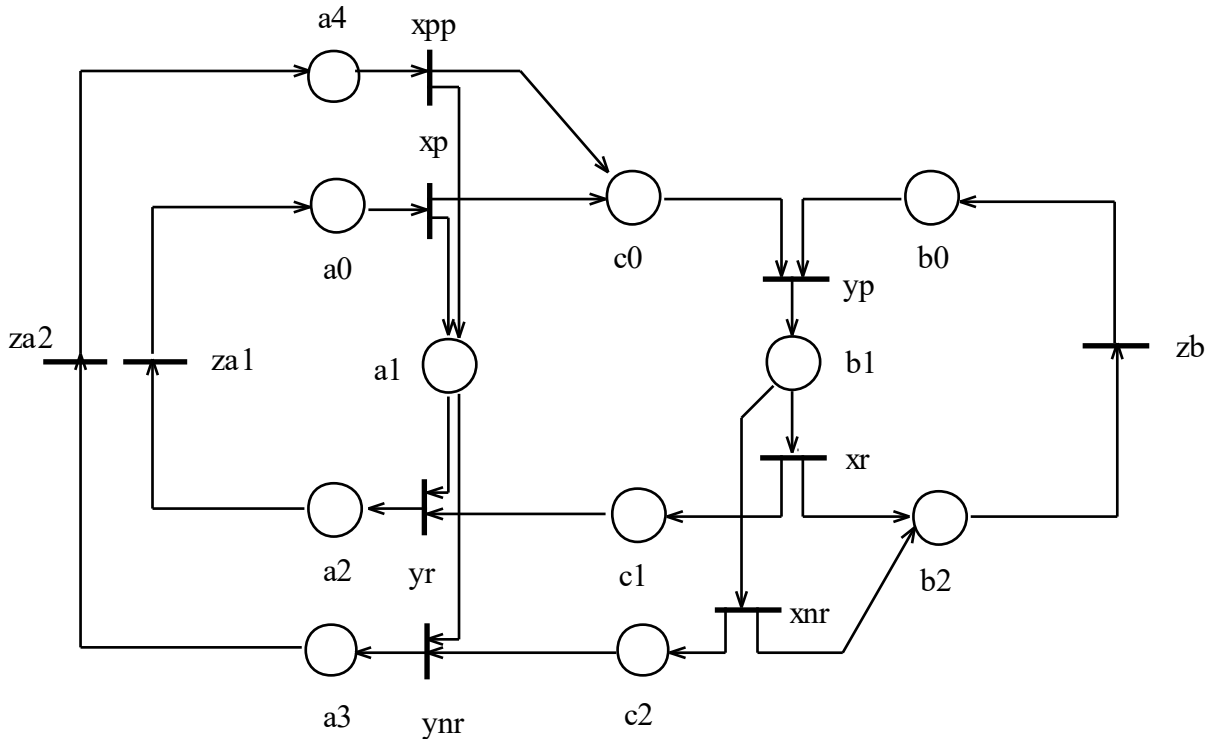


*Slika 4.7. Graf stanja izvorne Petrijeve mreže za model protokola*

Opisani komunikacijski protokol ne sadrži reakciju na negativnu potvrdu pri neuspješnoj komunikaciji. Uvođenjem pozitivne ( $r$ ) i negativne potvrde ( $nr$ ) dobiva se mreža prema slici 4.8. Nova mjesta i prijelazi su:

$a_3$	negativna potvrda prijama
$a_4$	poruka pripravna za ponovni prijenos (retransmisiju)
$c_2$	negativna potvrda na kanalu
$x_{nr}$	predaja negativne potvrde
$y_{nr}$	prijam negativne potvrde
$x_{pp}$	retransmisija poruke.

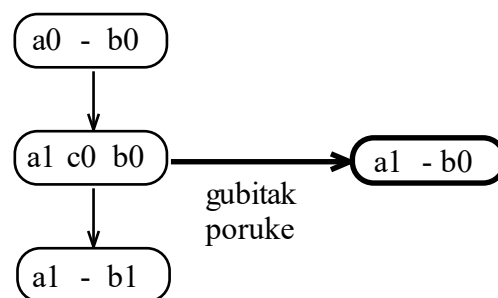
Mreža je složenija nego za osnovni model protokola, a složeniji je i graf stanja. Mreža je aktivna i sigurna, a nije konzervacijska i perzistentna. Strukturno je restriksijska, te odgovara pojmu mreže slobodnog izbora. Prijelazi  $x_r$  i  $x_{nr}$ , te  $y_r$  i  $y_{nr}$  su konfliktne.



Slika 4.8. Model protokola s pozitivnom i negativnom potvrdom

Prijašnja analiza potvrdila je ispravnost komunikacije samo ako nema nikakve pogreške. Međutim, komunikacijske protokole valja verificirati potpuno, što znači da pri sintezi valja ispitati i moguće neregularnosti. Model utemeljen na Petrijevoj mreži omogućuje simulaciju pogreške što će se pokazati na primjeru gubitka poruke.

Neka se u stanju  $(a_1, c_0, b_0)$  izgubi poruka što se simulira prijelazom u novo stanje  $(a_1, b_0)$  kao što je prikazano na dijelu grafa stanja (sl.4.9).



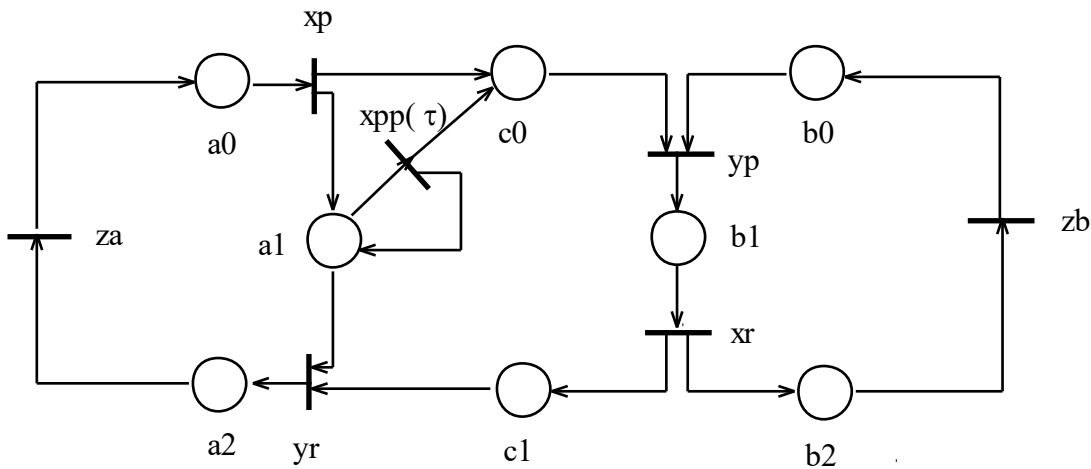
Slika 4.9. Prijelaz u neregularno stanje

U tom stanju ne može se izvesti nijedan prijelaz i riječ je o stanju blokiranja. Drugim riječima djelovanjem pogreške iz skupa regularnih stanja prešlo se u neregularno stanje koje se ne može napustiti logikom opisanog protokola. Potrebni su novi mehanizmi za upravljanje komunikacijom i koordinaciju

procesa. Teorijski i praktično provodi se vremenska kontrola komunikacije, što u ovom pristupu uvodi vremensku Petrijevu mrežu.

#### 4.2.2. Primjena vremenske Petrijeve mreže

U stanju blokiranja ( $a_1, b_0$ ) proces  $P_A$  neprekidno provjerava je li stigla potvrda, a proces  $P_B$  je li stigla poruka, odnosno čekaju jedan na drugoga da nešto učini. Uobičajeno je rješenje da predajni proces koji očekuje potvrdu provodi vremensku kontrolu nakon koje ponavlja poruku, pretpostavljajući da nije stigla do prijemnog procesa. Drugim riječima osnovni model treba proširiti vremenskim prijelazom koji će odašiljati poruku nakon isteka vremenske kontrole (sl.4.10) i vratiti procese u regularno stanje (sl.4.11).



Slika 4.10. Vremenska Petrijeva mreža za model protokola

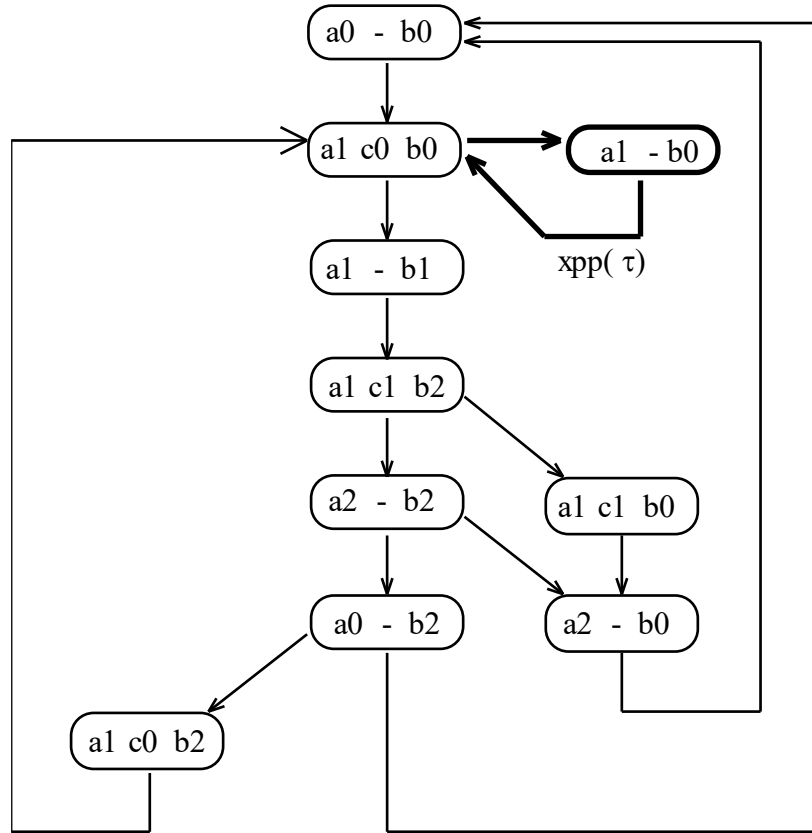
Može se dokazati da uvedeni vremenski prijelaz  $x_{pp}(\tau)$  rješava gubitak poruke i potvrde u bilo kojem stanju i ostvaruje povratak u regularno stanje.

Postavlja se pitanje kako odrediti trajanje vremenske kontrole. Definiranje vremenskih ograničenja je složeno zbog najmanje dva razloga. Prvi je određivanje relativnih odnosa graničnih vremena realizacije prijelaza i zadržavanja u nekom stanju. Drugi je određivanje ponašanja sustava ako se neko od uvedenih vremenskih ograničenja ne ispuni. Najkraće trajanje vremenskog uvjeta mora biti veće od najduljeg trajanja prijenosa:

$$\tau_{min} > T_{c0max}(a_1, c_0, b_0) = T_{c0max},$$

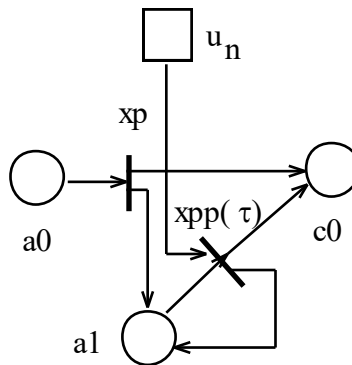
odnosno zadržavanja sustava u stanju ( $a_1, c_0, b_0$ ) zbog ispunjenosti uvjeta  $c_0$ .

Tada se poruka neće ponoviti prije nego što je mogla stići do predajnog procesa.



Slika 4.11. Povratak u regularno stanje

Ponavljanje broja pokušaja ispravnog i potvrđenog prijensa poruke potrebno je ograničiti. Za to se može primijeniti na primjer proširena Petrijeva mreža kojom se uvodi "vanjski" uvjet  $u_n$  sa značenjem "izvedi prijelaz  $n$  puta" (sl.4.12).

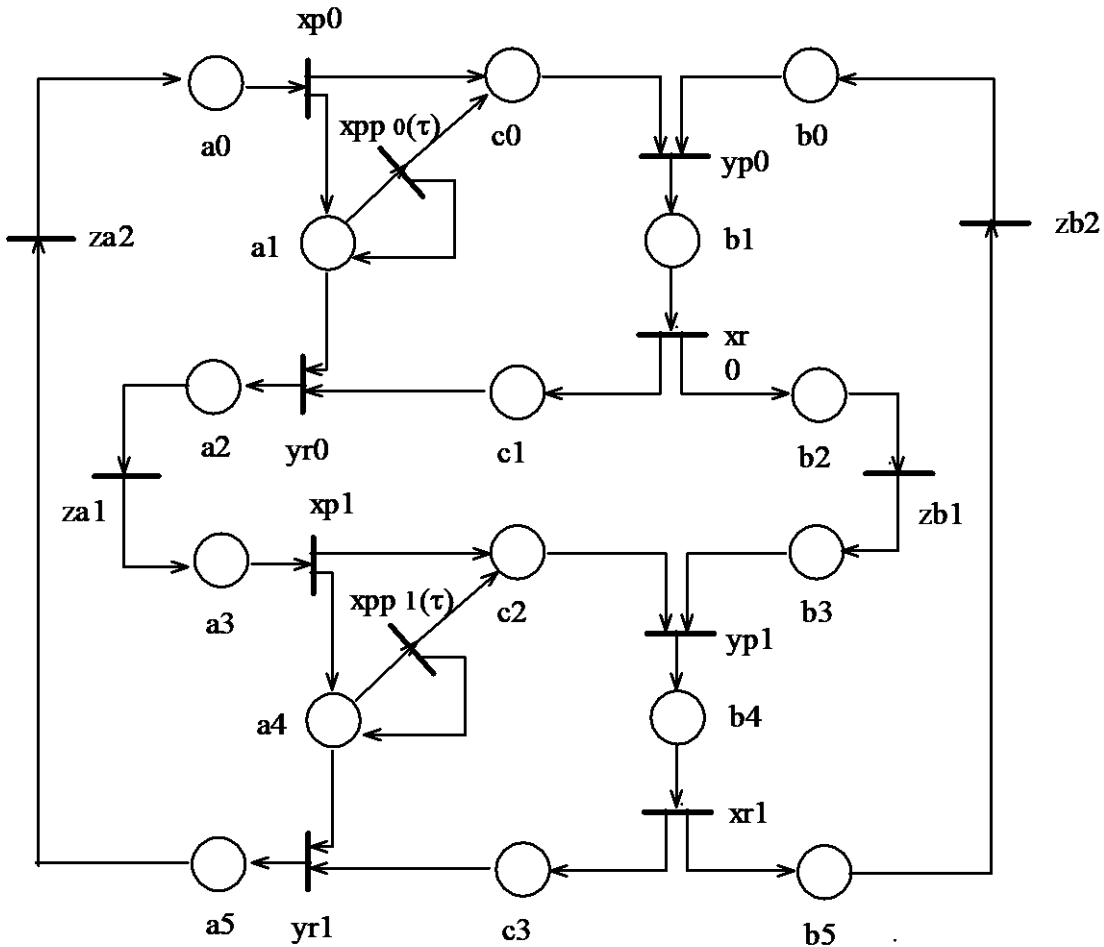


Slika 4.12 Vanjski uvjet za model protokola

Pokazat će se također kako procedurni aspekti komunikacije utječu na složenost modela. Tipična situacija nastane kad je potrebno razlikovati informacijske jedinice. Složenost mreže ovisi o broju informacijskih jedinica i njihovoj obradi.

Zbog toga se često primjenjuje podjela protokola na više razina kojima se rješavaju pojedini problemi.

Protokol za prijenos numeriranih poruka označenih s 0 i 1 naizmjenično, tzv. protokol s alternirajućim bitom poslužiti će kao primjer (sl.4.13).



Slika 4.13. Protokol s alternirajućim bitom

Analizira se gubitak poruka  $r_0$  i  $r_1$  u stanjima kanala koja imaju ispunjene uvjete  $c_0$  i  $c_2$ . Stanja blokiranja su  $(a_1, b_0)$  i  $(a_4, b_3)$ . Povratak u regularno stanje ostvaren je prijelazima  $x_{pp0}$  i  $x_{pp1}$  s vremenskim uvjetima:

$$\tau_{pp0min} > Ta1max$$

$$\tau_{pp1min} > Ta4max$$

i ispunjeni su uvjeti za povratak u regularno stanje nakon gubitka poruke.

Za gubitak u  $C_0$  vrijedi:

$$T_{yrmin}(a_1, c_1, b_2) < T_{zb1max}(a_1, c_1, b_2)$$

$$T_{za1min}(a_2, b_2) < T_{zb1max}(a_2, b_2)$$

$$T_{xp1min}(a_0, b_2) < T_{zb1max}(a_0, b_2),$$

a u izraze za  $c_2$  valja uvrstiti ekvivalentne prijelaze i uvjete. Mreža se vraća u regularno stanje ako se vremenski prijelazi ne izvedu prije nego što se utvrdi stvarni gubitak poruke. Da bi se izbjegla izvedba tih prijelaza potrebno je uvesti dva nova, također vremenska prijelaza  $y_{pp1}$  i  $y_{pp0}$  između mjesta  $c_0$  i  $b_3$ , odnosno  $c_2$  i  $b_0$  za koja vrijedi:

$$\tau_{ypp1max} < \tau_{xp0min} + \tau_{yp0min} + \tau_{za2min}$$

$$\tau_{ypp0max} < \tau_{xp1min} + \tau_{yp1min} + \tau_{za1min}.$$

Alternativa izravnom uvođenju vremenskih uvjeta i radu s vremenskom Petrijevom mrežom jest obrada vremenskih uvjeta na drugoj razini koja se u mrežu preslikava kao vanjski uvjet za izvedbu prijelaza. Tada se dodjeljuju vanjski uvjeti prijelazu  $t_j$  sa značenjem:

$P_{t_j}$  - predikat dopuštenja za izvedbu prijelaza

$F_{t_j}$  - funkcija akcije pokrenute izvedbom prijelaza.

Takav se prijelaz izvodi ako su ispunjeni uvjeti a predikat dopuštenja je istinit. Izvedbom prijelaza generira se novo stanje i pokreće funkcija akcije.

Takav pristup može poslužiti i za povezivanje upravljačkog i procedurnog segmenta protokola. Upravljački se modelira Petrijevom mrežom, a procedurni nadovezuje interakcijom preko predikata dopuštenja i funkcija akcije.