

Ime in priimek:

Datum:

Laboratorijska vaja: Izvajanje simulacij v simulatorju COMNET III

1 Namen

Namen vaje je seznaniti učečega z izvajanjem simulacij v simulatorju COMNET III.

2 Izvajanje simulacij z uporabo simulatorja CACI COMNET III

CACI COMNET III je orodje za analizo zmogljivosti telekomunikacijskih omrežij. Za izhodišče potrebujemo podatke, kot so opis omrežja, krmilnih algoritmov in prometne obremenitve. COMNET III nato simulira delovanje opisanega omrežja in nam kot rezultat poda meritve omrežnih zmogljivosti. Za izdelavo in izvajanje simulacije imamo na voljo grafični vmesnik (Windows XP/7 32 bit), preko katerega vnašamo opise omrežij in ostale podatke.

Izvedbo simulacije omrežja razdelimo v tri ločene faze:

- izgradnja modela dejanskega omrežja, opis virov,
- zagon simulacije,
- predstavitev rezultatov.

2.1 Izgradnja modela omrežja

Model omrežja, ki ga želimo simulirati, nima nujno enake zgradbe kot dejansko obstoječe omrežje. Pri njegovi zasnovi je pomembno predvsem to, da v določenem smislu (oz. na nekem segmentu) ohranimo lastnosti in funkcionalnosti dejanskega omrežja, s čemer zagotovimo, da so rezultati simulacije skladni z realnim stanjem (odzivom sistema).

Izgradnjo modela razdelimo v tri faze.

- izgradnja topologije omrežja,
- določitev prometnih tokov in obremenitev,
- obratovanje omrežja.

2.1.1 Izgradnja topologije omrežja

Izgradnja topologije predstavlja prvi korak pri izdelavi modela omrežja. Pojem topologija zajema predstavitev strukture omrežja in njegovih zmogljivosti (angl. *resources*). V simulacijskem orodju jo opišemo s pomočjo treh osnovnih komponent.

Vozlišča (angl. *nodes*)

Vozlišča predstavljajo točke v omrežju, kjer se izvaja procesiranje prometnih tokov. Vozlišča lahko predstavljajo stikala (angl. *switch*), usmerjevalniki (angl. *router*), mostovi (angl. *bridge*), terminalna oprema, itd.

Linije (angl. *links*)

Linije predstavljajo prenosne kanale, preko katerih potujejo prometni tokovi. Lahko so tipa point-to-point (npr. najete linije, ISDN, SDH) ali pa point-to-multipoint (npr. Ethernet, FDDI).

Stične povezave (angl. arcs)

Stične povezave predstavljajo vrata (angl. *port*), ki povezujejo linije z ustreznimi vozlišči. V njih npr. določimo velikosti vhodnega oz. izhodnega predpomnilnika.

2.1.2 Prometni tokovi in dodatna obremenitev omrežja

Prometni izvori in dodatna obremenitev dajo modelu potreben stimulus, ki omogoča izvajanje simulacije. Prometni tokovi so posledica sporočil, ki se prenašajo skozi omrežje med izvornimi in ponornimi vozlišči. Dodatna obremenitev je posledica aktivnosti procesorjev in vodil v posameznih vozliščih.

Definicija in pravilna konfiguracija prometnih izvorov je eden izmed najbolj kritičnih delov izdelave simulacije. Če izvorov ne opišemo pravilno, bodo tudi rezultati simulacije napačni. Pri opisovanju izvorov navadno uporabljamo naključne procese (opisljive s statističnimi porazdelitvami), s katerimi se poskušamo čim bolj približati realnim razmeram. Najbolj znana in preprosta, a pogosto tudi zadovoljivo verodostojna statistična porazdelitev je *eksponentna porazdelitev*. Poleg te se pogosto uporabljamo še druge, npr. *pareto porazdelitev*, *enakomerna porazdelitev*, *normalna porazdelitev*, ... COMNET III pozna tudi nekaj kompleksnejših porazdelitev, npr. *burst porazdelitev*, ki simulira informacijske izbruhe pri tipičnih podatkovnih izvori.

Osnovno enoto v izvoru predstavlja sporočilo (angl. *message*). To je lahko en znak (telnet), elektronska pošta ali pa obsežna datoteka (FTP). Sporočila se v samem izvoru razdelijo v pakete, slednji potujejo po omrežju.

2.1.3 Obratovanje omrežja

Obratovanje omrežja določa način potovanja posameznih sporočil skozi omrežje.

Algoritem usmerjanja (routing)

Algoritem usmerjanja določa pot posameznih paketov skozi simulirano omrežje. Na izbiro imamo več možnosti: Minimum Hop, Minimum Penalty, IGRP, ... Vsi ti protokoli so lahko bodisi statični ali dinamični – odvisno od naše izbire. Seveda imamo tudi možnost lastne definicije poti skozi omrežje – v tem primeru uporabimo statično usmerjanje.

Transportni protokol

Transportni protokol krmili prenos sporočil skozi omrežje. Le-te je potrebno ob vstopu v omrežje razbiti na posamezne pakete in zagotoviti zanesljiv prenos celotnega sporočila. Na voljo imamo večje število že specificiranih protokolov (TCP/IP, SNA, IPX, ...). Po potrebi lahko specificiramo svoje verzije. Protokolom lahko spreminjamo razne parametre, npr. velikost krmilnega okna.

2.2 Zagon simulacije

Pred zagonom simulacije nastavimo določene parametre, kot so npr. trajanje simulacije, število ponovitev, itd. Pred zagonom simulacije moramo obvezno določiti parametre, ki so za nas zanimivi in zaradi katerih simulacijo sploh izvajamo. Tipično so to obremenjenosti prenosnih linij, napolnjenosti vmesnih pomnilnikov, zakasnitve pri prenosu paketov in sporočil, itd.

Po izgradnji modela najprej izvedemo verifikacijo in če jo model uspešno prestane, sledi simulacija. Že med samim izvajanjem simulacije lahko spremljamo določene zanimive parametre, kot je npr. obremenitev linij in stanje vmesnih pomnilnikov.

2.3 Predstavitev rezultatov

Rezultati simulacije predstavljajo cilj vseh naših naporov. Že pred izvajanjem simulacije smo identificirali in sprožili zajem zanimivih informacij. Kot rezultat lahko dobimo dolgoročna povprečja in njihova odstopanja ali pa dejanski časovni potek posameznih parametrov. Prikažemo jih lahko v tekstovni in grafični obliki.

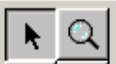


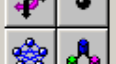

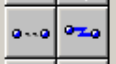







Iz rezultatov razberemo obstoječa in potencialna ozka grla simuliranega omrežja in izvedemo ustrezne korekcije v omrežju. Z ustrezno prerazporeditvijo obstoječe opreme lahko marsikdaj precej izboljšamo razmere v omrežju. Izgrajena simulacija tudi omogoča hitro oceno vpliva vpeljave novih aplikacij v obstoječ sistem.

Simulacijska orodja omogoča tudi testiranje zanesljivosti omrežja in preverjanje odzivov na napake, saj lahko kadarkoli izključimo katerikoli omrežni element, bodisi ročno ali avtomatsko in pomerimo odziv na takšno situacijo v omrežju.

3 Voden zgled izvedbe simulacije v simulatorju CACI COMNET III

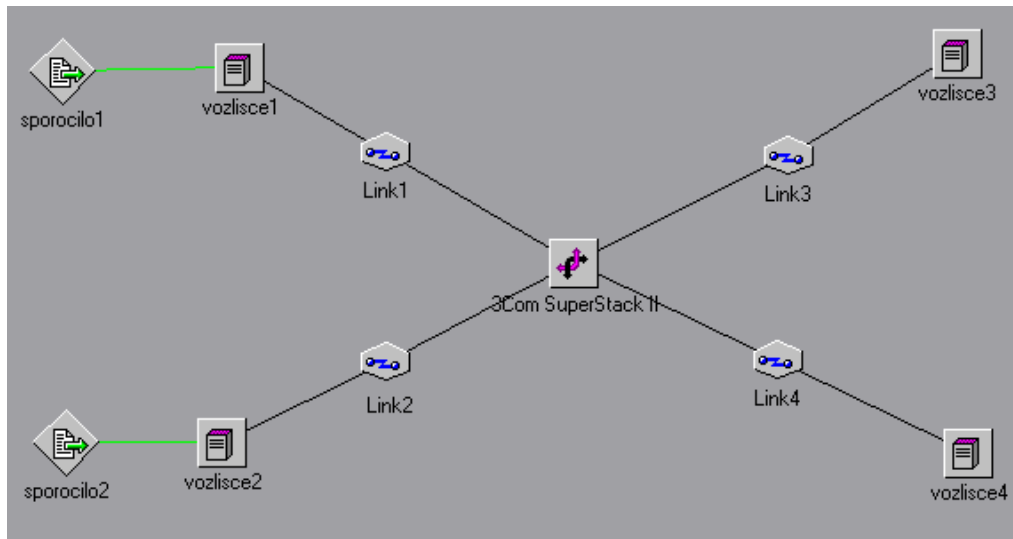
3.1 Osnovni gradniki in izgradnja topologije omrežja

V simulatorju COMNET III imamo na levi strani orodno vrstico z osnovnimi orodji, ki jih vlečemo na delovno površino in si z njimi izgradimo željeno topologijo.

oznaci / izberi		povečaj površino
diagonalna povezava		horizontalna / vertikalna povezava
procesno vozlišče		skupina računalnikov
omrežni element		dostopna točka
prehodno omrežje		podomrežje
oblak		satelitska povezava
WAN povezava ali VC		point-to-point povezava
CSMA/CD povezava		token ring
izvor sporočil		odgovor na sporočilo
aplikacijski izvor		izvor seje sporočil
klicni izvor		izvorni socket
ozadje		oddaljena povezava
ozadje s zemljevidom		tekst v ozadju

Slika : Orodna vrstica (meni)

Postopek izgradnje topologije si pogledjmo na konkretnem preprostem primeru omrežja, podanem na spodnji sliki:



Slika : Primer preprostega omrežja

Omrežje vsebuje dva izvora prometa, ki sta preko povezav in vmesnega stikala povezana na dva ponora.

Izgradnja naj poteka v sledečem zaporedju:

- na delovno površino povlecite štiri procesna vozlišča
- povlecite en omrežni element
- povlecite potrebno število PPP (Point-To-Point) povezav
- povežite med sabo procesna vozlišča preko PPP povezav
- na delovno površino povlecite dva izvora sporočil in jih povežite na procesna vozlišča

Preden bo simulacija nared za zagon, je potrebno nastaviti še nekaj parametrov.

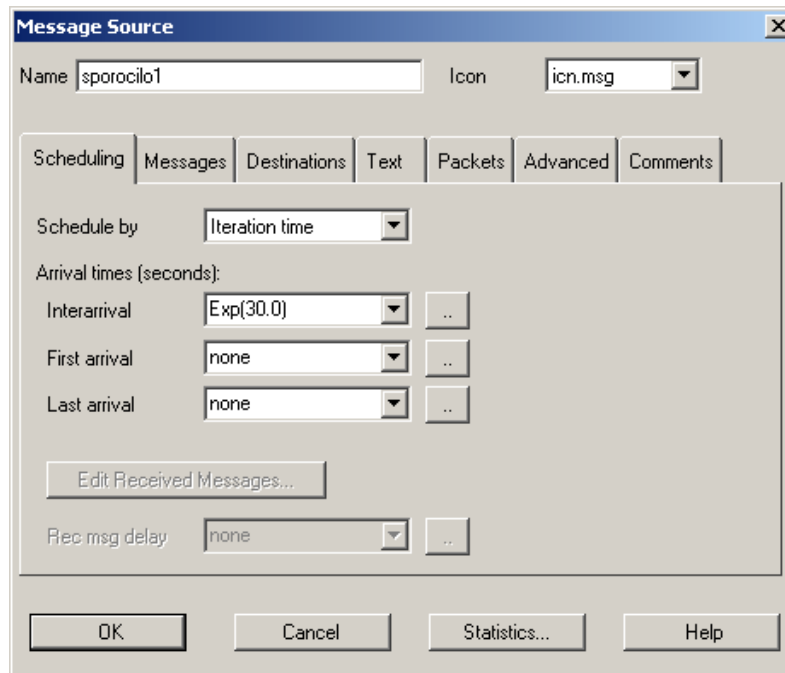
3.2 Nastavitve aplikacijskih izvorov

V prvem primeru si bomo pogledali nastavitve najpreprostejšega izvora, t.j. izvora sporočil. Če dvokliknemo na ikono sporocila1, ki jo imamo na delovni površini, dobimo spodnje okno.

V našem primeru bomo spremenili čas prihoda paketov na eksponentno porazdelitev s povprečno vrednostjo 30 sekund (polje **Interarrival**).

Prenosni protokol, preko katerega se bodo prenašala sporočila, nastavimo na **UDP**; to storimo na zavihku **Packets** v polju **Protocol**.

Med samimi nastavitvami si lahko ogledate tudi ostale možnosti, ki se skrivajo pod jezički. Sem spadajo velikost sporočil, prenosni protokol, cilj sporočila, itd.

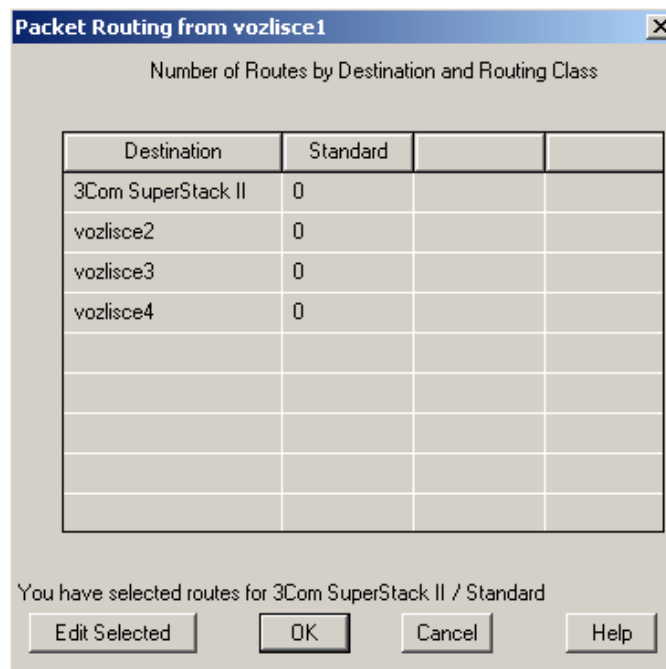


Slika : Nastavitve izvora sporočil

3.3 Nastavitev usmerjevalnega algoritma

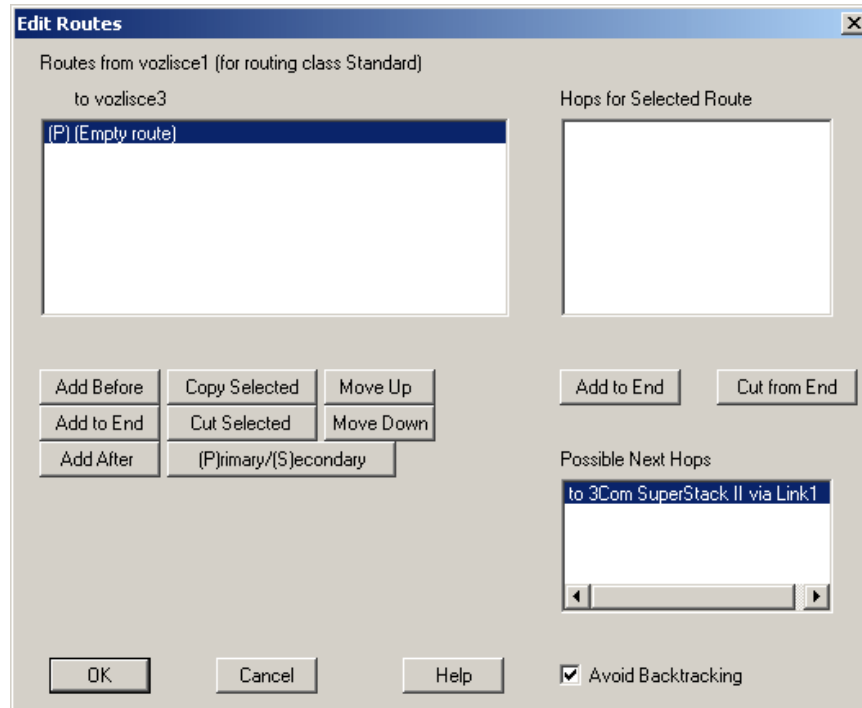
V našem preprostem primeru si bomo ogledali postopek ročnega vnosa statične usmerjevalne tabele. Pod **Define** -> **Backbone Properties** v polju **Packet Routing Protocol** izberemo **User-Defined Routing Tables**.

Ko sedaj dvokliknemo na vozlišce1, se v oknu **Node Properties** pojavi nov zavihek **Routing Tables**. Kliknemo na gumb **Packets** in pojavi se naslednje okno.



Slika : Prazna usmerjevalna tabela

V polju **Destination** vidimo vse možne cilje, ki izhajajo iz vozlišca1. Denimo, da bi radi promet iz vozlišca1 pripeljali na vozlišce3, promet iz vozlišca2 pa na vozlišce4. V tem primeru bomo v zgornjem polju **Destination** označili vozlišce3 in kliknili na gumb **Edit Selected**.

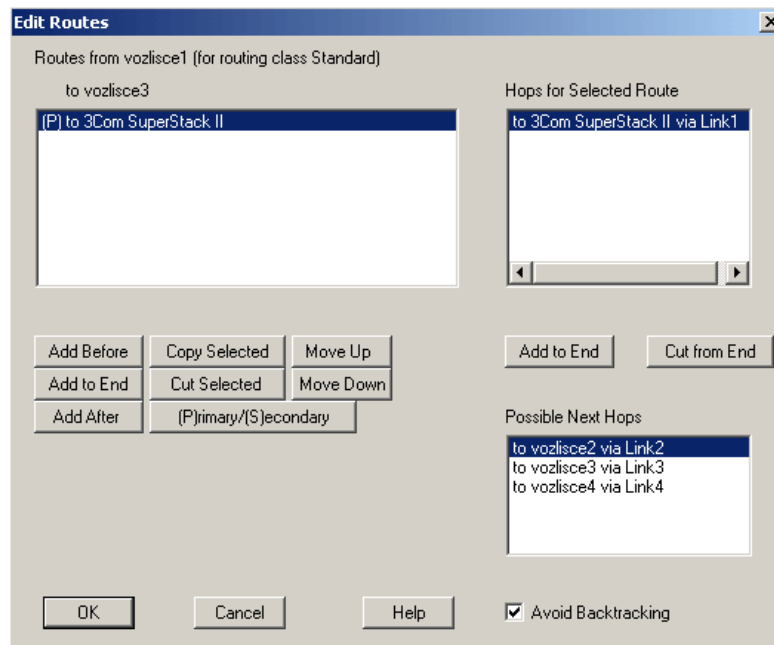


Slika : Izpolnjevanje usmerjevalne tabele (korak 1).

V polju **from vozlišce1 to vozlišce3** vidimo, da imamo prazno tabelo (**Empty Route**). V polju **Possible Next Hops** vidimo, da je možna povezava na 3Com stikalo preko Linka1. Če označimo to možno povezavo in kliknemo na gumb **Add to End**, se nam bo ta informacija prenesla v polje, kjer smo do sedaj imeli prazno tabelo, v polju Possible Next Hops pa se nam bodo pojavile naslednje možne povezave, ki jih vidimo na spodnji sliki.

Podobno zgodbo ponovimo še na vozlišcu2, le da tam za cilj namesto vozlišca3 izberemo vozlišce4. S tem smo določili zgolj usmerjevalne tabele do stikala, ne pa tudi do končnih povezav, zato je potrebno na stikalu definirati povezave do ponorov.

Na stikalu imamo kot možne cilje (destinacije) podana vsa štiri vozlišča, kot ponor izberemo tretje in četrto.



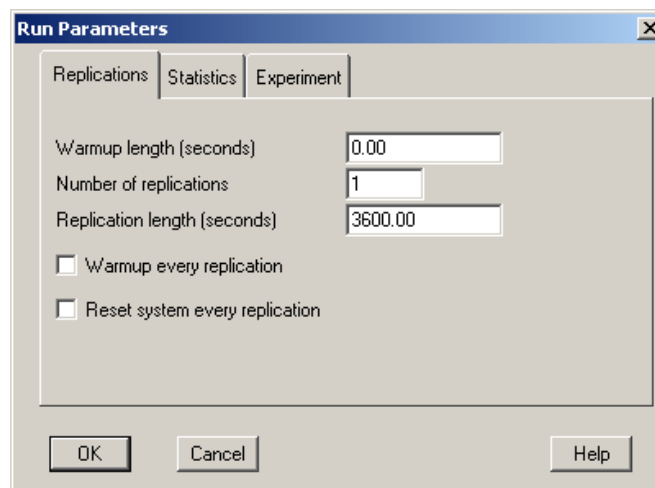
Slika : Izpolnjevanje usmerjevalne tabele (korak 2).

3.4 Destinacija sporočila

Zgradili smo topologijo omrežja, na posameznih vozliščih smo podali usmerjevalne tabele. V nadaljevanju viru (aplikaciji) podajmo informacijo o tem, komu pošilja podatke oz. kje se nahaja njen ponor. To storimo tako, da dvokliknemo na **sporocilo1** in pod jezičkom **Destinations** kliknemo na gumb **Edit Destination List**. V novem oknu kliknemo na gumb **Add** in v našem primeru izberemo ciljno vozlisce3. Podobno za sporocilo2 izberemo ciljno vozlisce4.

3.5 Definicija globalnih parametrov simulacije

Pod **Simulate** -> **Run Parameters** nastavljam globalne parametre simulacije.

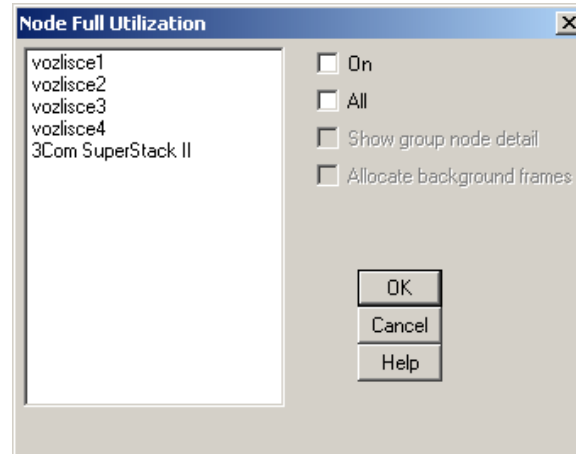


Slika : Globalni parametri

Dolžino simulacije nastavimo v polju **Replication length**.

3.6 Rezultati, ki bi jih radi razbrali iz simulacije

Pred zagonom simulacije je potrebno nastaviti obliko izpisa simulacijskih rezultatov. Na **Report -> Select Reports** se nam odpre okno s podanimi vozlišči, povezavami, izvori prometa itd. Izberimo si **Node -> Node Full Utilization** in kliknemo na gumb **Edit**. Pojavi se naslednje okno.



Slika : Izbira vozlišč

Izberemo si vozlišča, na katerih bi radi spremljali obremenjenost in kliknemo **On**. V tabeli **Select Reports** lahko izberemo še ostale informacije, ki bi nas utegnile zanimati. Omenjeno statistiko bomo dobili v obliki tekstovne datoteke po izteku simulacije, že med samo simulacijo pa lahko spremljamo statistiko na elementih topologije v obliki grafa. V ta namen dvokliknemo na izbrani element topologije (vozlišče, sporočilo, povezava) in v dobljenem oknu izberemo gumb **Statistics**.

Dvokliknemo na **Link1**, izberimo **Statistics**, označimo **Busy Channels from 1st Node**, kliknemo **Edit** ter označimo, da bi radi zajemali statistiko.

3.7 Zagon simulacije

S tem smo zaključili z izgradnjo modela in vnosom parametrov. Simulacijo zaženemo tako, da kliknemo na zeleni semafor. Rdeča barva na semaforju označuje, da je simulacija v teku.

3.8 Pregled statistik

Grafična statistika se nam izrisuje v realnem času, za tekstovni izpis pa je potrebno po koncu simulacije klikniti na **Report -> Browse Reports**, kjer si izberemo izhodno datoteko in jo odpremo. Iz nje z nadaljnjo programsko opremo (npr. Matlab, Excell, skriptni jeziki) izluščimo želene podatke.

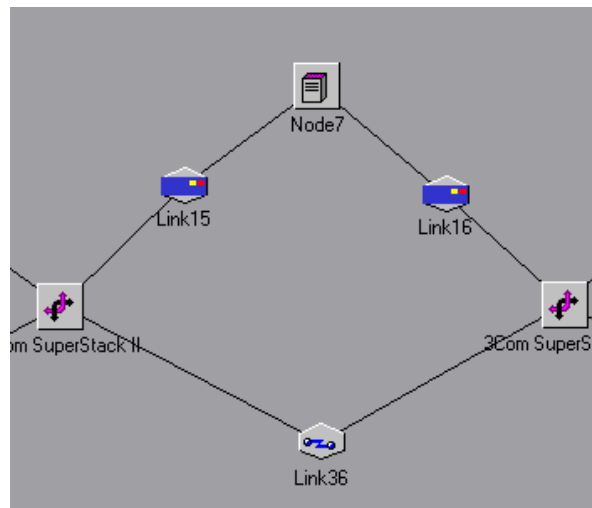
Naloga 1

Po uspešno izvedenem (delujočem) scenariju iz Slika nastavi usmerjevalni algoritem (npr. RIP Minimum Hop) in ponovno zaženi simulacijo.

Spreminjaj dinamiko povezav in vozlišč (rušenje, ponovno vzpostavljanje) in spremljaj izhodno statistiko. Zamenjaj prenosni protokol s protokolom TCP.

Naloga 2

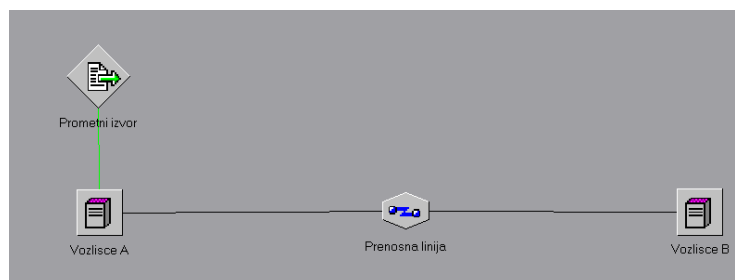
Razširi primer iz Slika tako, da zgradiš nekoliko bolj razvejano vmesno vozlišče z dvema stikali, dodatnim vozliščem in tremi povezavami, kot kaže spodnja slika. Preveri porazdelitev prometa in reakcijo omrežja v primeru izpada posamezne povezave.



Slika : Razvejano vmesno vozlišče

Naloga 3

Zgradite preprost model omrežja z dvema vozlišči, povezanimi preko prenosne linije (point-to-point) s hitrostjo 64 kbit/s. Na enega od vozlišč priključite prometni izvor. Tip vozlišča, na katerega je priključen prometni izvor, naj bo takšen, da vnaša zakasnitev 0,1 ms pri obdelavi 1 kB podatkov. Prometni izvor naj proži iterativni čas z enakomerno verjetnostno porazdelitvijo uniform(1, 10). Velikost sporočil naj se podreja eksponentni verjetnostni porazdelitvi s srednjo vrednostjo 10 kB. Sporočila naj se pošiljajo naključnim sosedom. Paketi naj uporabljajo TCP/IP protokol s časom paketizacije 10 ms. Sproti želimo opazovati obremenitev povezave v obe smeri.

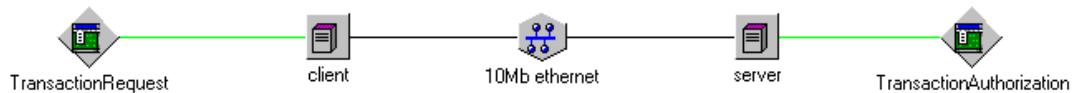


Slika : Dve vozlišči povezani preko prenosne linije

Naloga 4

Komunikacija odjemalec/strežnik poteka preko Ethernet povezave 10 Mb/s. Odjemalec pošlje sporočilo **Authorization Request**, nato pa preide v čakalno stanje (**Wait for server authorization**). Strežnik bo ob dobljenem zahtevku odjemalcu poslal sporočilo **Authorize client request**.

Zgradi topologijo po spodnji sliki:

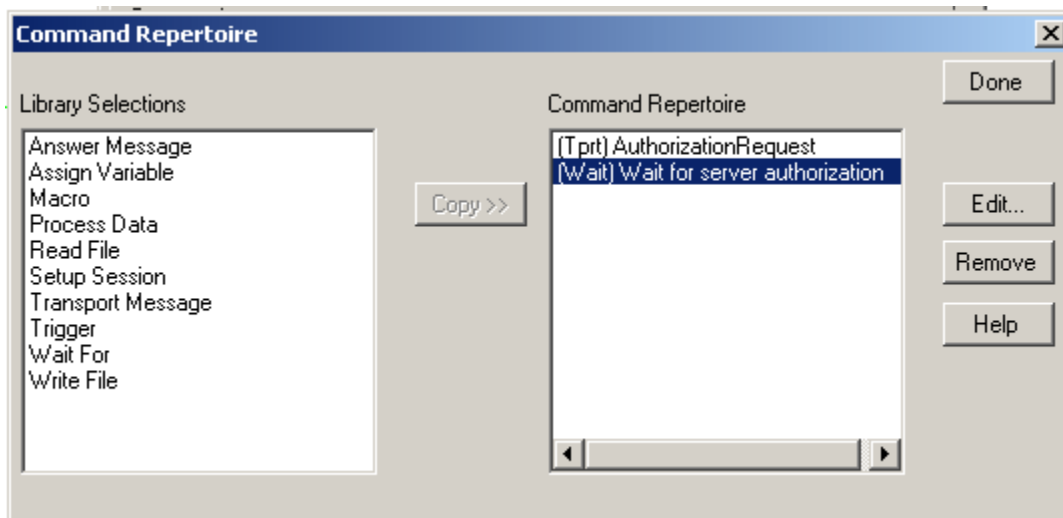


Slika : Topologija Ethernet omrežja

Definiraj vsa tri sporočila, ter zgradite logiko pošiljanja in čakanja na potrditve.

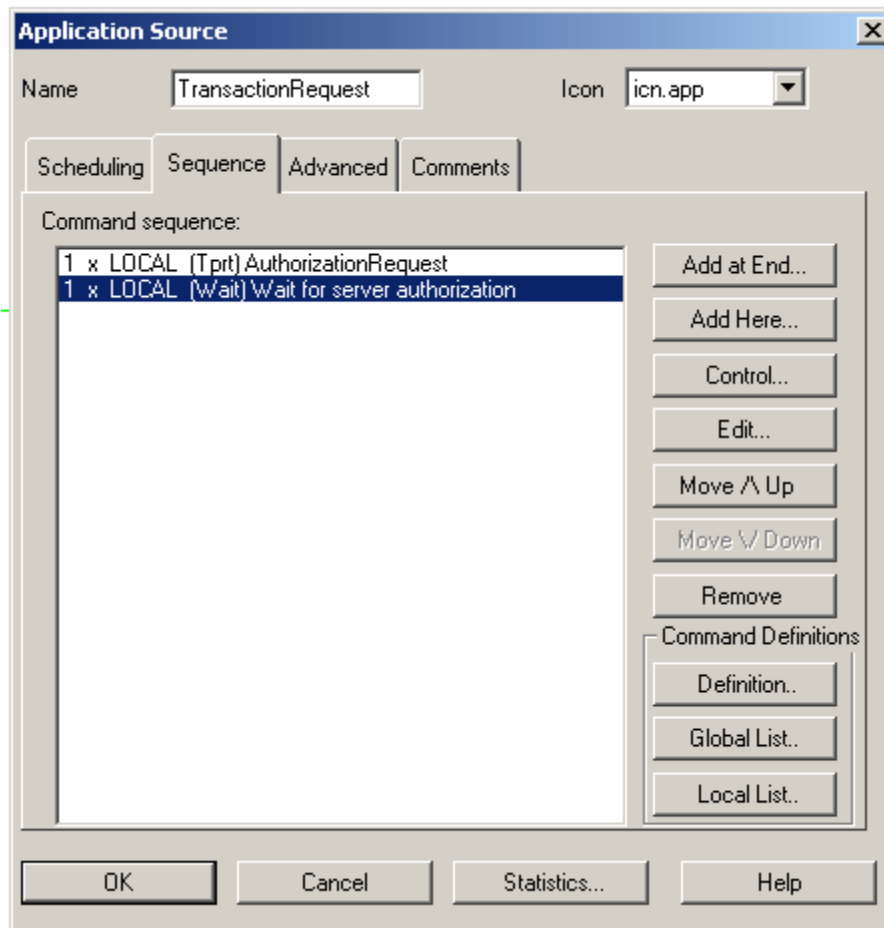
V aplikaciji, ki jo bomo poimenovali **TransactionRequest** definiraj dve sporočili. V zavihku **Sequence** klikni na gumb **Local List** in izberi:

- **Transport Message**, ki smo mu dodelili ime **Authorization Request**
- **Wait For**, ki smo mu dodelili ime **Wait for server authorization**



Slika : Logika pošiljanja in čakanja potrditve

Nato s gumbom **Add to End** dodaj sporočili aplikaciji na odjemalcu.



Enako zaporedje ponovi še na strežniku, s tem da je tu potrebno definirati sporočilo **Authorize Client Request** in ga ravno tako dodati aplikaciji. Na strežniku izberi Scheduled by **Received Message**, da bo le-ta dejansko čakal na sporočilo **Authorization Request**.

Naloga 5

Oglej si primer (naloga5), ki simulira tri skupine bankomatov, ki izvajajo avtentikacijo na centralni enoti preko zakupljenih vodov (WAN povezav). Spreminjajte parametre in opazujte reakcije v omrežju.

Naloga 6

Oglej si primer (naloga6), ki simulira skupino računalnikov s nameščenim Office paketom, ki preko lokalnega omrežja pošilja razne zahteve na tri strežnike. Spreminjajte parametre in opazujte reakcije v omrežju.

Naloga 7

Oglej si primer (naloga7), ki operaterja, ki na lokacijah uporabnikom ponuja storitve VoD in IPTV. Primerjajte situacijo (obremenitve na povezavah v omrežju), ko so vsebine shranjene centralno oz. porazdeljeno (Mirror).