

باسمه تعالی



دانشگاه صنعتی شریف

دانشکده مهندسی برق

شبکه مخابرات داده

تمرین 2

علی محرابیان 96102331

استاد: دکتر پاکروان

بهار 1399



لایه دیتالینک:

1) سرویس های Acknowledged connectionless، Acknowledged connection oriented و Unacknowledged connectionless را برای لایه Network فراهم می کند.

در Unacknowledged connectionless، هدف فقط فرستادن فریم ها است و تلاشی برای بازیابی فریم های ازدست رفته نمی شود. در سیستم های real time و کانال هایی که خطای کمی دارند، استفاده می شود. Ethernet

در Acknowledged connectionless، برای بررسی صحت پیام دریافتی، از acknowledge استفاده می شود. در کانال های نامطمئن مانند کانال بی سیم استفاده می شود. (wi-fi)

در Acknowledged connection oriented، فریم ها شماره گذاری می شوند و لایه پیوند داده، رسیدن یک کپی از هر فریم را به مقصد تضمین می کند.

الگوریتم stop and wait:

1) اگر فریم در مسیر به گیرنده و یا ack در مسیر به فرستنده گم شود، فرستنده مدت زمان طولانی به انتظار می نشیند تا تایید ارسال پیام را از گیرنده دریافت کند. بنابراین از تایمر برای حل این مشکل استفاده می شود.

2) مهم ترین مشکل آن، استفاده نابهیینه از پهنای باند کانال است و performance کمی دارد. این الگوریتم برای فاصله های کمتر، به طور مثال LAN مناسب است و برای ارسال packet های بزرگتر استفاده می شود.

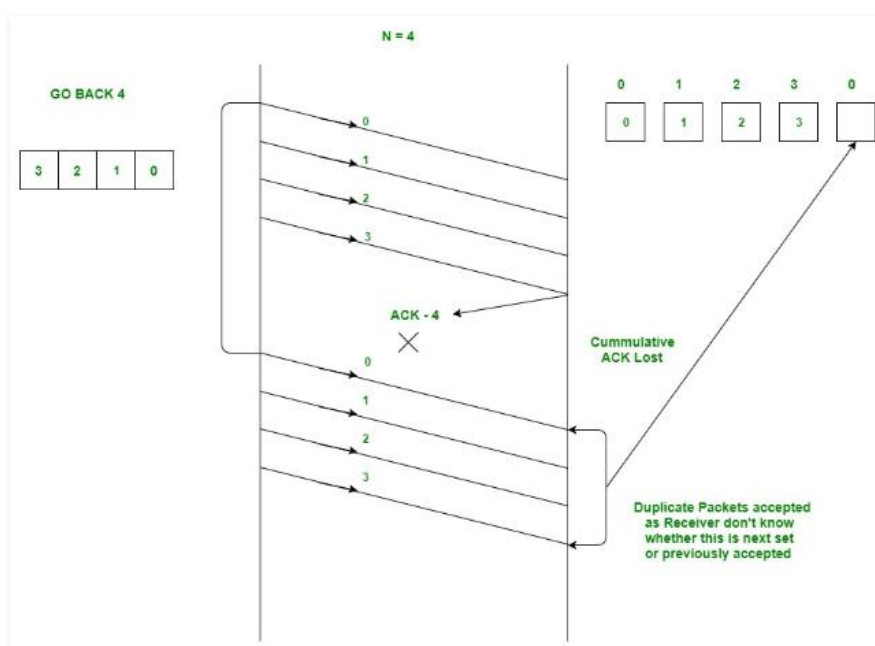


(3) بله، برای ارسال هر دو فریم متوالی هنگام رسیدن ack از فریم قبلی تمایز قائل شویم، نیاز به استفاده از Sequence number داریم. برای تمایز ack های متوالی نیز نیاز به شماره گذاری داریم. برای هر کدام از موارد تنها استفاده از یک بیت کافی است.

الگوریتم GO BACK N:

(1) اگر پیام به درستی به مقصد نرسد، به اندازه طول کل پنجره باید دوباره پیام ها را ارسال کنیم که این به معنای به هدر دادن مقدار زیادی از پهنای باند است. بنابراین برای کانال هایی که خطای کمتری دارند، استفاده می شود. از الگوریتم selective repeat برای حل این مشکل استفاده می شود. در سمت گیرنده، پیام هایی که همراه با فریم گم شده یا خراب شده، ارسال می شود، ذخیره می گردند. گیرنده باید توانایی گرفتن پیام ها خارج از ترتیب را داشته باشد. تنها packet هایی که گم شده اند، دوباره ارسال می شوند و از پهنای باند استفاده بهینه ای می شود.

(2) حالت زیر را در نظر بگیریم.

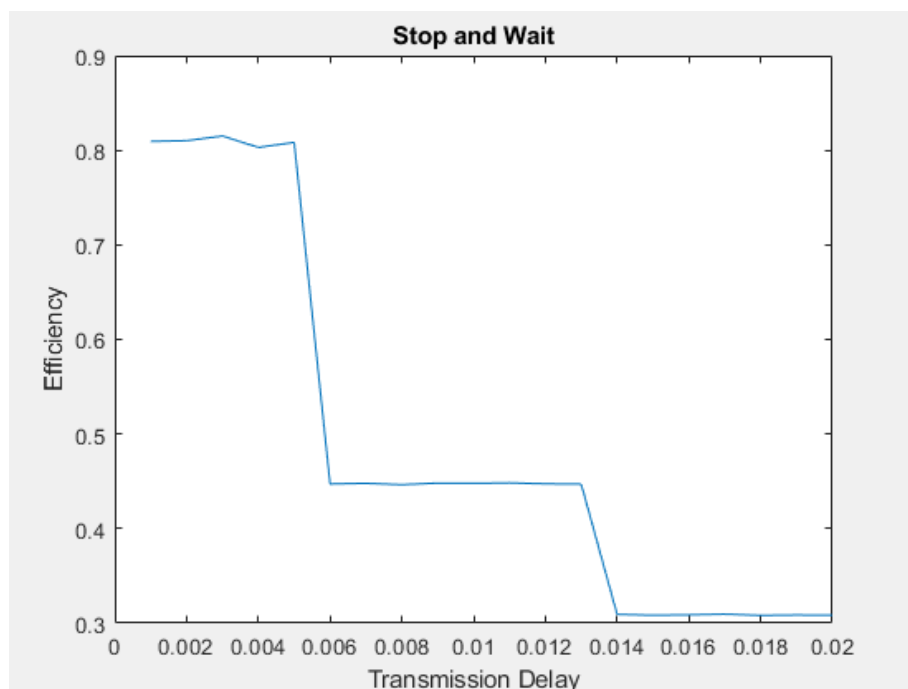




پنجره ای به طول 4 داریم و طول sequence number نیز برابر با 4 است. حال در اولین ارسال، ack آخرین پیام به ما نمی رسد. حال مجموعه پیام های بعدی را می فرستیم. از آنجایی که گیرنده نمی داند که ack آخرین پیام به فرستنده رسیده است، نمی تواند تشخیص بدهد که این مجموعه پیام ها، جدید بوده یا همان پیام های قبلی هستند. پس طول پنجره ما کزیمم 1 واحد کمتر از sequence number است.

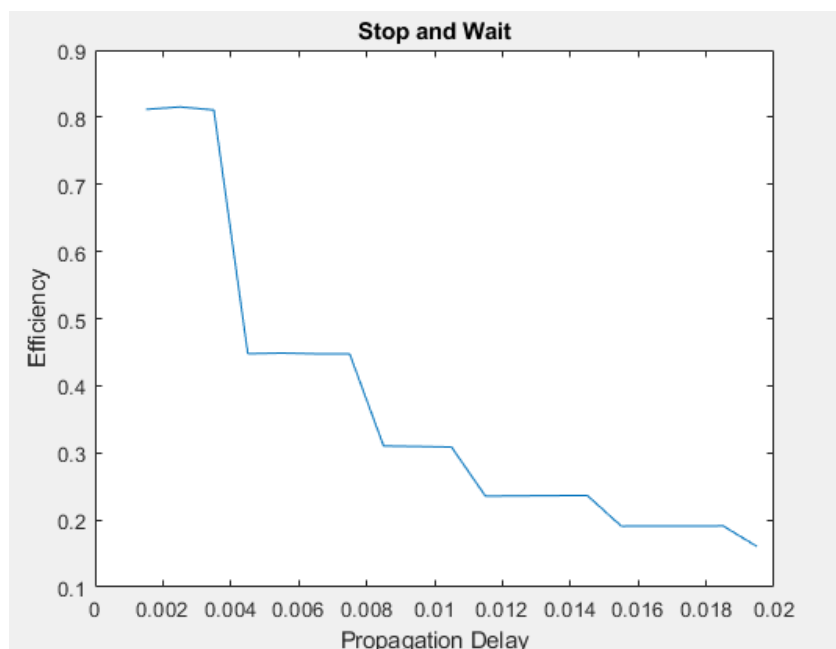
شبیه سازی:

در ابتدا لازم به ذکر است که در اسلایدها، link utilization با استفاده از زمان تاخیرها محاسبه می شد. در این جا، efficiency را به صورت نسبت تعداد فریم ها به تعداد ارسال های مورد نیاز برای فریم ها تعریف می کنیم. در ابتدا تغییرات را برای stop and wait اعمال می کنیم. لازم به ذکر است برای هر سه روش، مقدار مشخصی برای Tout در نظر گرفته ایم.

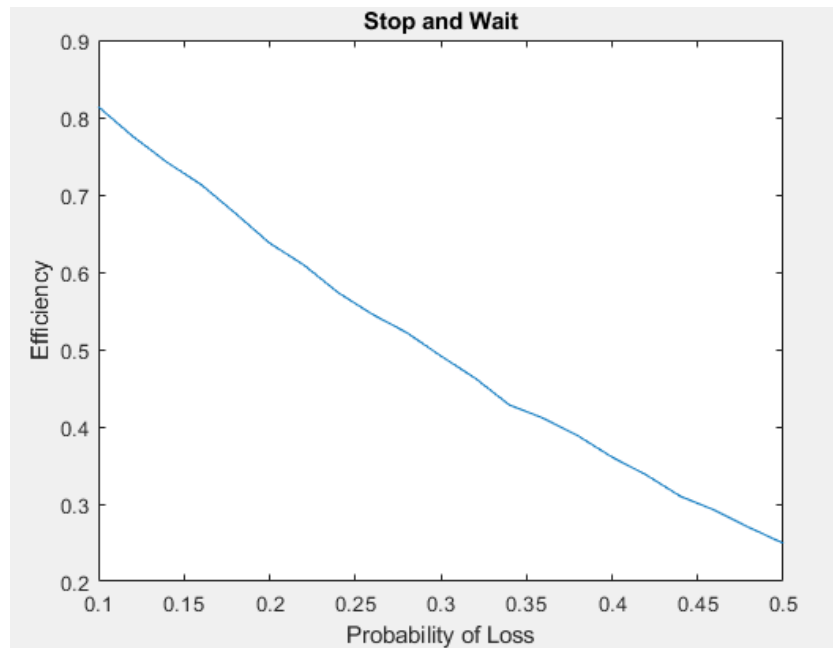




مشاهده می شود که با افزایش T_t ، بهره روند نزولی دارد. برای بازه هایی ثابت بوده و در بعضی جا ها، کاهش ناگهانی دارد. این کاهش به این دلیل است که همواره به شرط انتقال صحیح فریم و ack به گیرنده و فرستنده، اگر $T_t + 2T_p < T_{out}$ باشد، یک بار و در غیر این صورت، $\text{ceil}(\frac{T_t + 2T_p}{T_{out}})$ بار فریم ارسال می شود. دلیل کاهش های ناگهانی نیز این است که در زمان هایی، $T_t + 2T_p = kT_{out}$ می شود و این کاهش رخ می دهد.

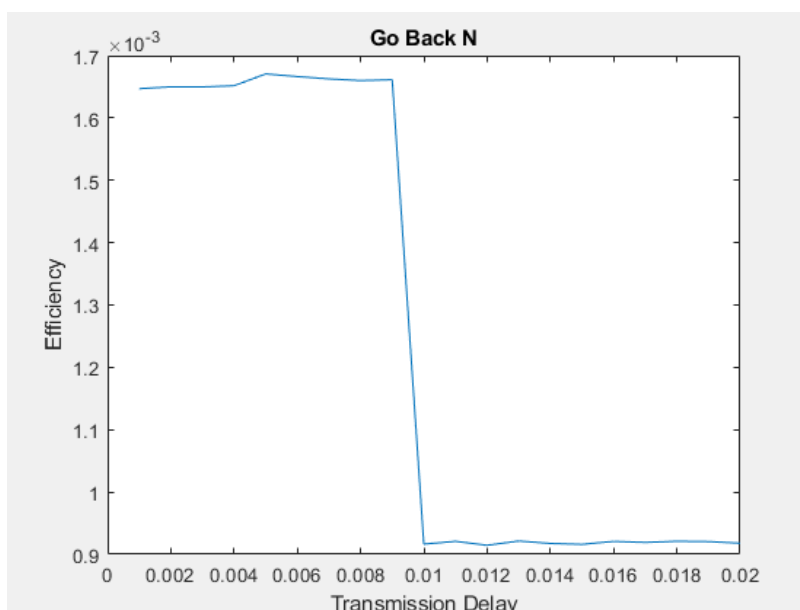


در این مورد نیز مشابه قبل، روند نزولی را شاهد هستیم. به دلیل ضریب 2 در رابطه که گفتیم، در این حالت کاهش های بیشتری را شاهد هستیم.

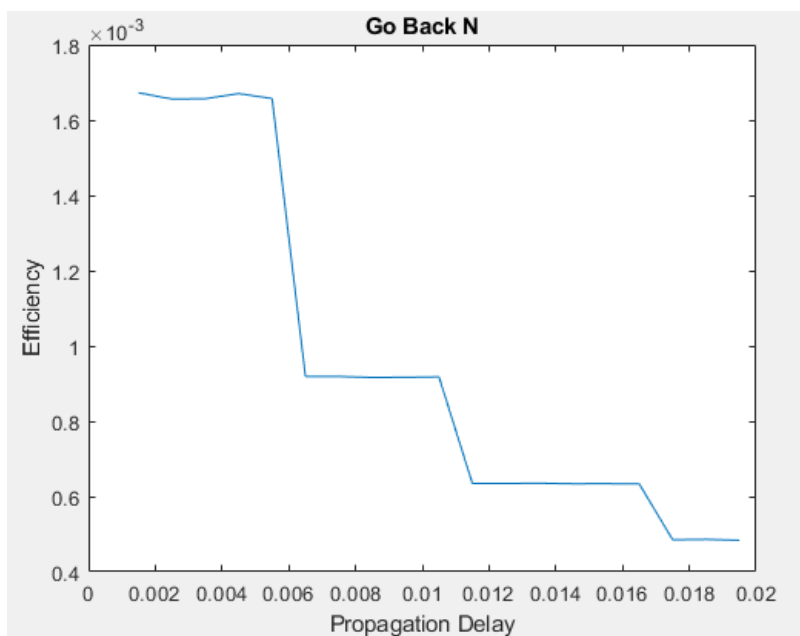


با افزایش احتمال خطای کانال، بهره ما رفته به رفته کاهش می یابد.

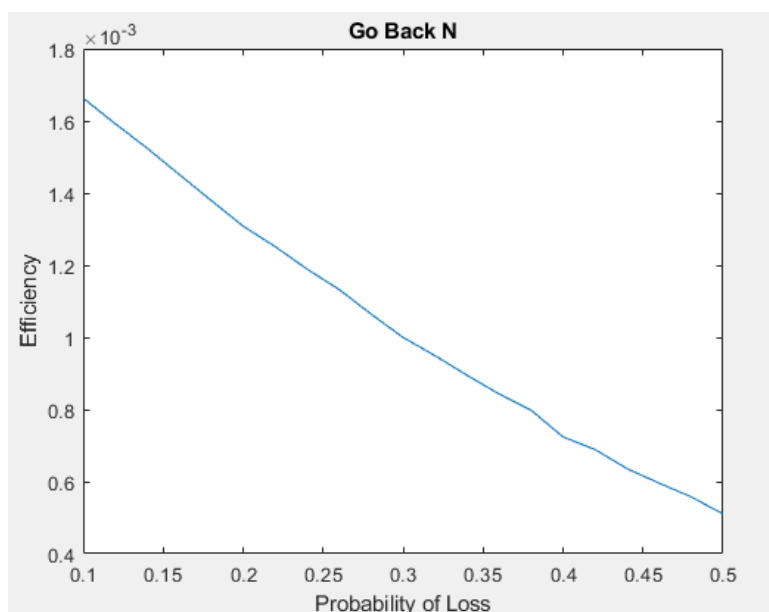
حال تغییرات را برای go back N بررسی می کنیم. لازم به ذکر است با توجه به تعریفی که از efficiency ارائه دادیم، چون برای ارسال هر فریم گم شده، به اندازه کل طول پنجره دوباره پیام ارسال می شود، بنابراین بهره کاهش پیدا خواهد کرد. به طور دقیق تر، اگر $T_t + 2T_p < T_{out}$ باشد، یک بار و در غیر این صورت، $ceil\left(\frac{T_t + 2T_p}{T_{out}}\right) * w$ بار فریم ارسال می شود.



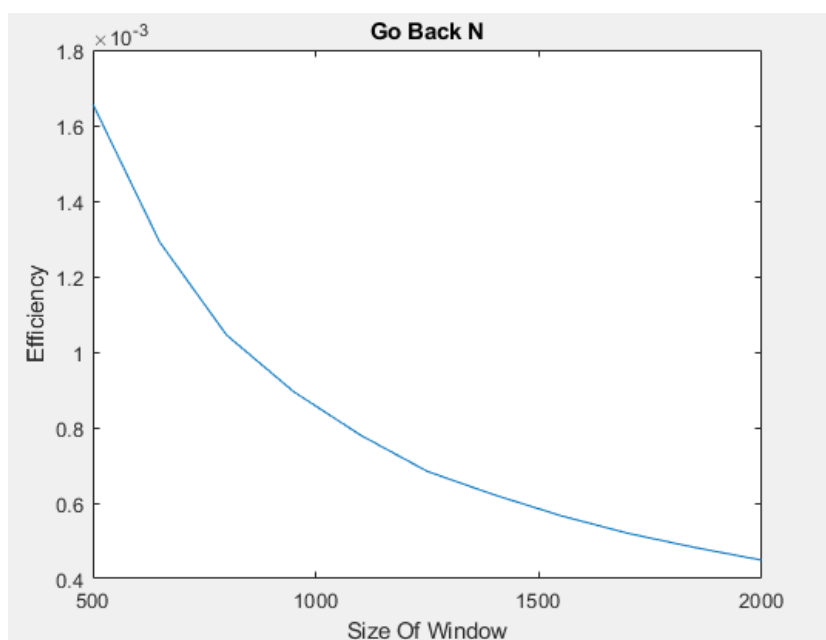
همانند قبل، با افزایش Transmission delay، بهره ما کاهش می یابد.



به دلیل ضریب 2، کاهش های ناگهانی بیشتری را شاهد هستیم.



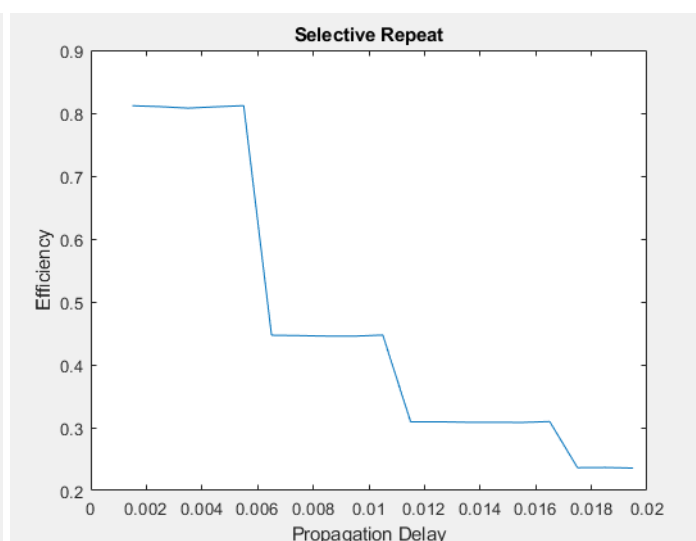
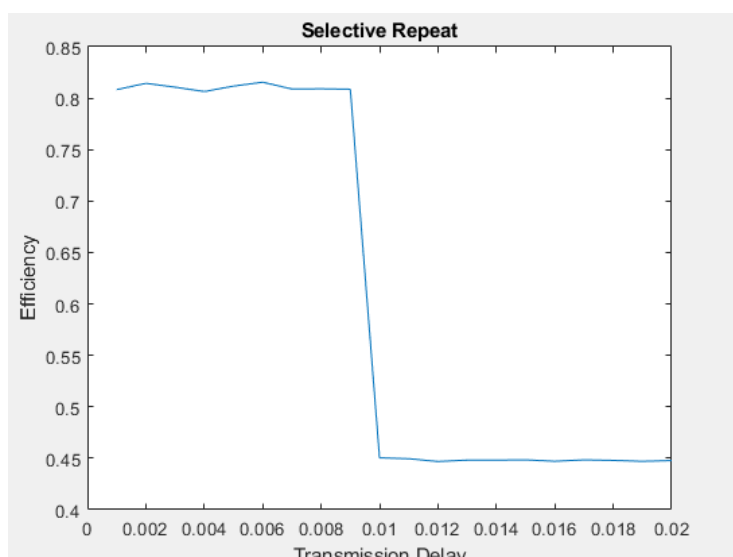
با افزایش احتمال خطای کانال، بهره کاهش می یابد.



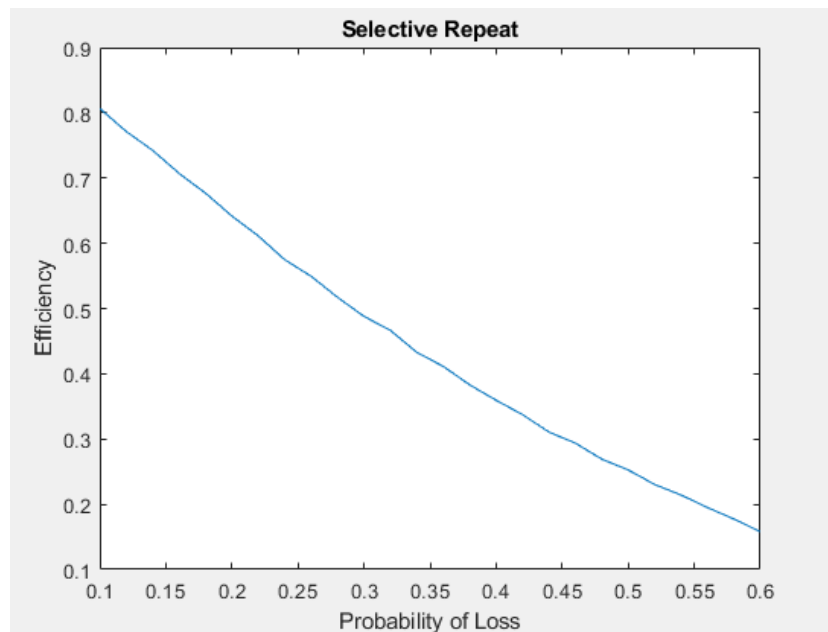
مشاهده می شود که با افزایش طول پنجره، بهره ما کاهش می یابد که انتظار آن را داشتیم.



برای حالت selective repeat، دیگر نیازی به ارسال کل پنجره برای packet گم شده نداریم. بنابراین بهره ما افزایش پیدا خواهد. حتی با توجه به ارسال پنجره ای پیام ها، در زمان نیز نسبت به stop and wait صرفه جویی خواهیم کرد. با توجه به تعریف efficiency، برای پیدا کردن کل فریم های ارسالی مورد نیاز، روش کار شبیه Stop and wait است.



چون T_{out} در sliding window protocol نسبت به stop and wait بزرگتر در نظر می گیریم، شاهد کاهش های ناگهانی کمتری هستیم.



با افزایش خطای کانال، بهره کاهش می یابد.

همان طور که ذکر شد، برای ارسال فریم های گم شده شیبه stop and wait عمل می کند، بنابراین با تغییر طول پنجره، efficiency ما با توجه به تعریف، تغییری نمی کند. بنابراین، stop and wait، هم از لحاظ زمانی طولانی است و هم از پهنای باند بهینه استفاده نمی کند. در go back N، با توجه به ارسال تعداد بیشتر فریم در پنجره، در زمان صرفه جویی می شود ولی در صورت وجود خطا، پهنای باند زیادی به صورت نابهینه استفاده می شود. در selective repeat، هم در زمان صرفه جویی می شود و هم پهنای باند ما به صورت بهینه استفاده می شود.