|  |  |  |
| --- | --- | --- |
|  | به نام خدا |  |
| **دانشگاه تهران**  **دانشکده‌ مهندسی برق و کامپیوتر**  **شبکه های کامپیوتری**  **پروژه دوم** | | |

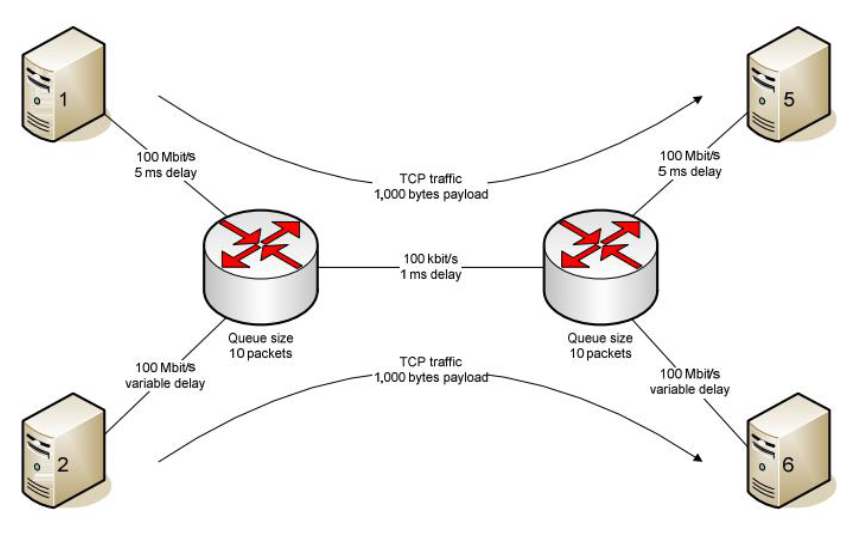
|  |  |  |
| --- | --- | --- |
| سجاد پاکدامن ساوجی | سینا شریفی | نام و نام خانوادگی |
| 810195517 | 810195412 | شماره‌ دانشجویی |
| 7/4/1399 | | تاریخ ارسال گزارش |

# بخش توضیحات پیادهسازی

برای پیادهسازی این پروژه در ابتدا یک فایل tcl به نام main.tcl را تشکیل میدهیم و پس از کامل کردن توپولوژی شبکه، اطلاعات مورد نظرمان را در فایل trace سیو میکنیم و در انتها از کدهای پایتون برای اجرای فایل tcl و رسم نمودار ها استفاده میکنیم.

برای نوشتن اسکریپت tcl به این صورت عمل میکنیم که ابتدا به کمک دستور سرعت هر لینک و سپس به کمک دستور تاخیر هر لینک را تعیین میکنیم(آرگومانهای رندوم در فایل پایتون ساخته شده و به عنوان آرگومان ورودی به فایل tcl داده میشود).

با کامل شدن لینکها میتوانیم به کمک دستور 6 نود تعریف کینم و نودها را به صورت به هم متصل کنیم و در انتها با دستور با سایز بافر نود 3 و 4 را نیز مشخص میکنیم. با انجام این مراحل، تا به اینجای کار توپولوژی شبکه کامل شده است و به صورت زیر درآمده است.



شکل 1- توپولوژی شبکه مورد آزمایش

بعد از آماده شدن شبکه به کمک دستور دو ایجنت tcp میسازیم و آنها را به نود یک و دو اساین میکنیم و سپس نود 5 و 6 را نیز به عنوان مقصد آنها در نظر میگیریم.

برای برقراری جریان داده بین دو مبدا و مقصد، دو ایجنت ftp نیز ساخته و به دو نود یک و دو اساین میکنیم.

با انجام این موارد شبکه آمادهی سیمولیت شدن است، ولی از آنجایی که ما میخواهیم دیتاهای متغیرهای دلخواهمان در فایل trace ذخیره شود، با استفاده از دستور بعضی از متغیرها مثل cwnd\_ , rtt\_, ack\_, ssthresh\_,… را در فایل ذخیره کردیم.

سپس این سیمولشین را برای 1000 ثانیه و ده بار اجرا کرده و نتایج را به کمک 4 فایل پایتون برای هر ثانیه نمایش میدهیم.

به ترتیب، برای رسم تغییرات پنجره ازدحام از متغیر cwnd\_ و برای رسم نرخ goodput از maxseq\_ و با استفاده از سایز هر پکت و پهنای باند استفاده شد.

برای محاسبه goodput باید مقدار داده یکتای انتقال یافته را بر مقدار تئوری داده قابل ارسال تقسیم کنیم. برای این کار متغیر  maxseq- را بر مقدار تئوری پکت های قابل ارسال تقسیم کردیم. روابط مورد نیاز برای محاسبه در زیر آمده است.

البته در این محاسبات مقدار ۴۰ بایت که در handshaking مصرف می‌شود نیز باید در نظر گرفته شود.

روش دیگر برای محاسبه goodput آن است که از متغیر   bytes- در نود گیرنده استفاده کنیم. در این روش باید در زمان های گسسته مقدار این متغیر را در فایلی ذخیره کنیم. در این روش نیز باید مقدار ۴۰ بایت که برای handshaking استفاده می‌شود را در نظر گرفت.

برای محاسبه اندازه پنجره تنها کافی است که متغیر cwnd- را trace کنیم.

برای رسم نرخ از دست رفتن بسته از تگ d در trace file استفاده شد که به معنی dropped است و در نهایت برای رسم نرخ RTT از متغیر rtt\_ استفاده شده است.

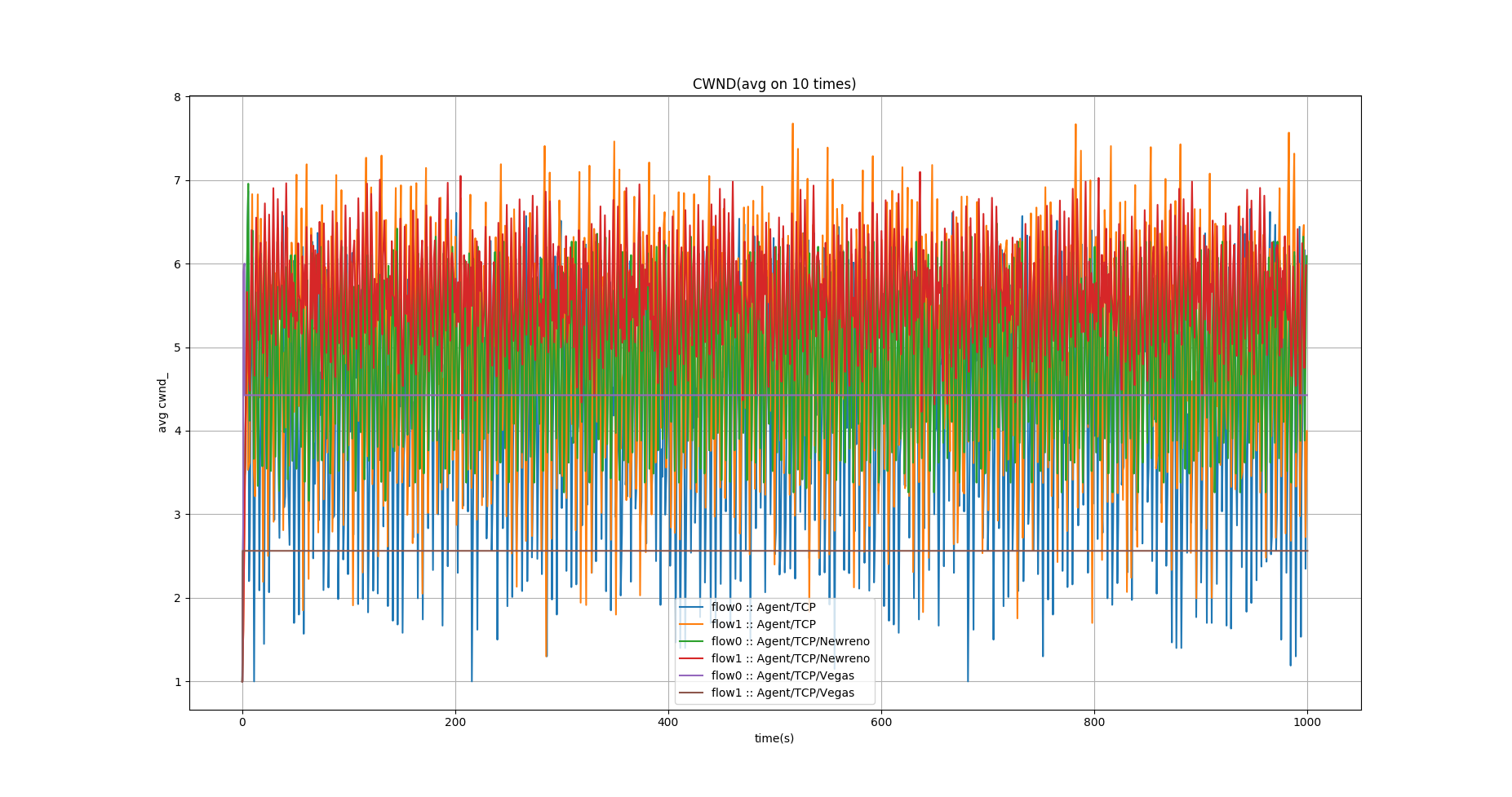
نتایج شبیهسازی و تحلیلهای مربوطه در قسمت بعدی مشخص شدهاند.

# سوال ۲ – نتایج شبیهسازی

در ادامه هر 4 متغیر را به ازای 10 مرتبه اجرا برای 1000 ثانیه رسم میکنیم.

برای تحلیل سایز پنجره میتوان گفت که هر 3 الگوریتم، از سایز پنجرهی کوچک شروع میکنند و شروع به افزایش آن میکنند. در دو مورد Tahoe و New Reno آن ها سایز را بالا میبرند تا به packet drop برخورد کنند و در این صورت، سایز را کاهش میدهند به همین دلیل در تمام مدت اجرا نوسان دارند.

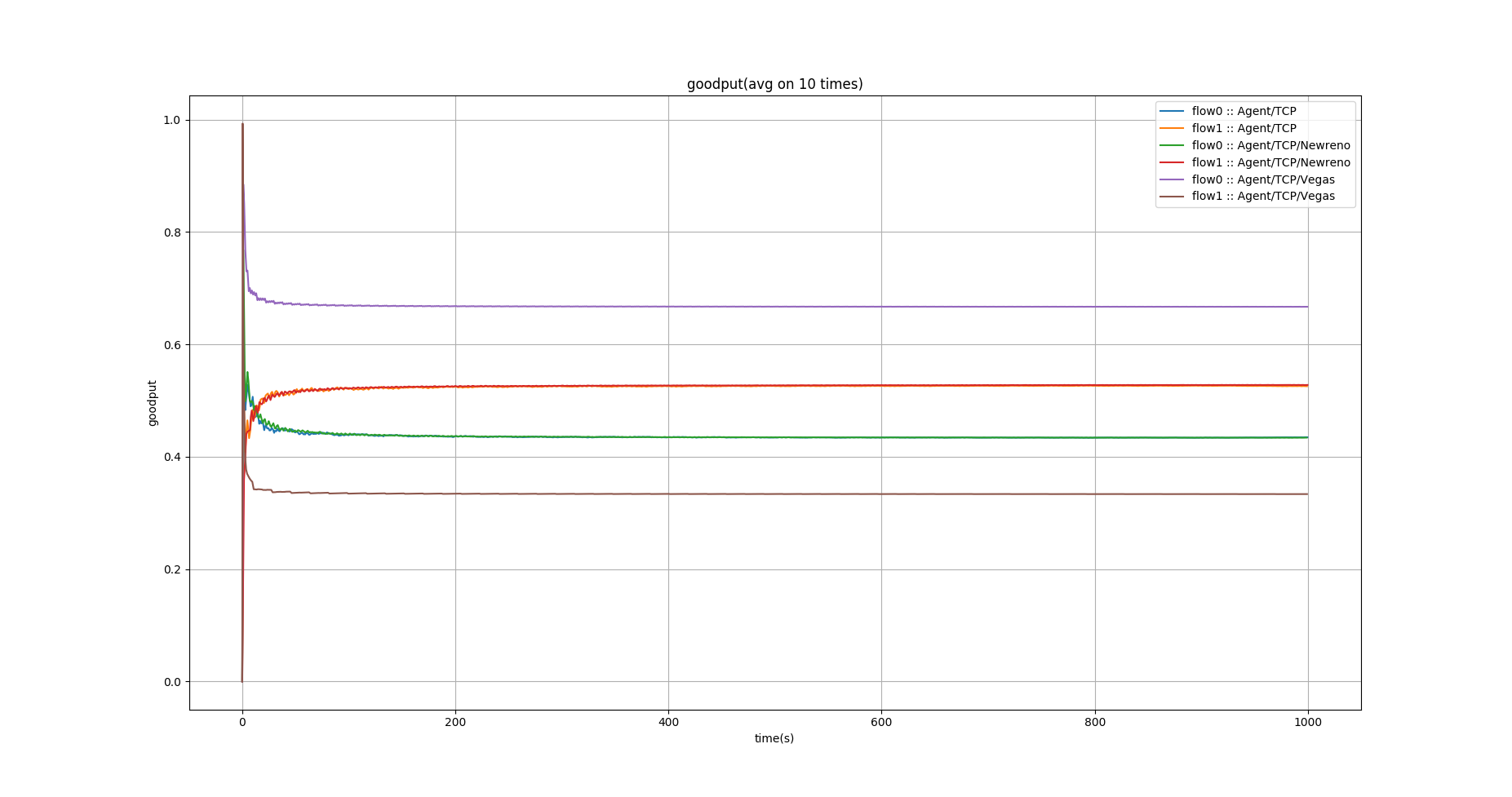
ولی Vegas چون از 2 مقدار threshold استفاده میکند، زمانی که به یک سایز در بین این دو مقدار میرسد، دیگر مقادیر خود را تغییر نمیدهد و به همین دلیل پس از اوایل اجرا، سایز پنجرهی آن ثابت میشود.



شکل 2- تغییرات اندازهی پنجره

برای تحلیل نرخ Goodput میتوان گفت که در دو مورد Tahoe و New Reno آن ها با تغییراتی که در سایز پنجره ایجاد میکنند، تقریبا به مقدار مشابه همگرا میشوند.

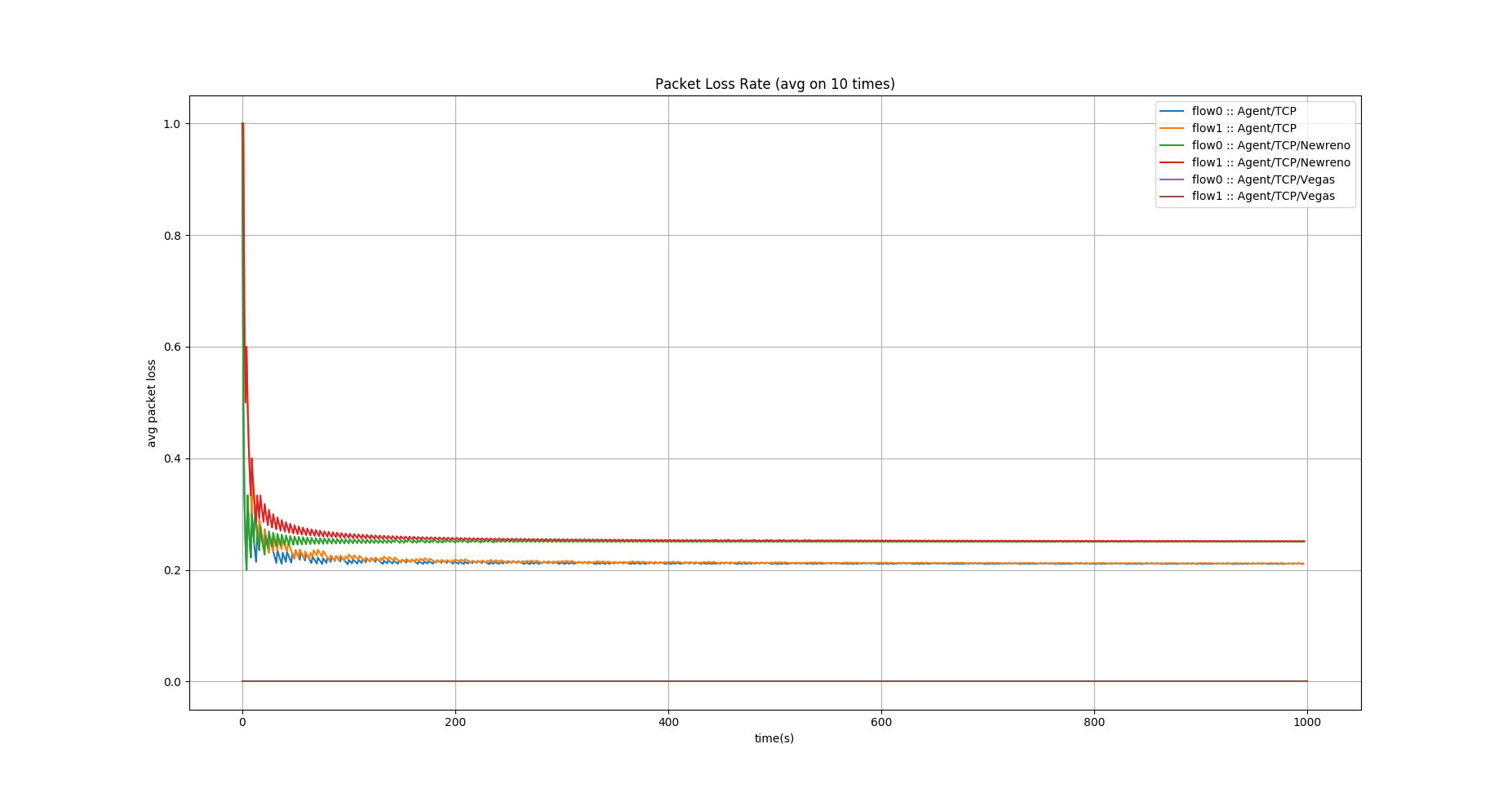
ولی در Vegas چون از 2 مقدار threshold استفاده میکند، لزوما سایز پنجرهها برابر نیست در نتیجه دو flow آن به یک مقدار Goodput همگرا نمیشود.



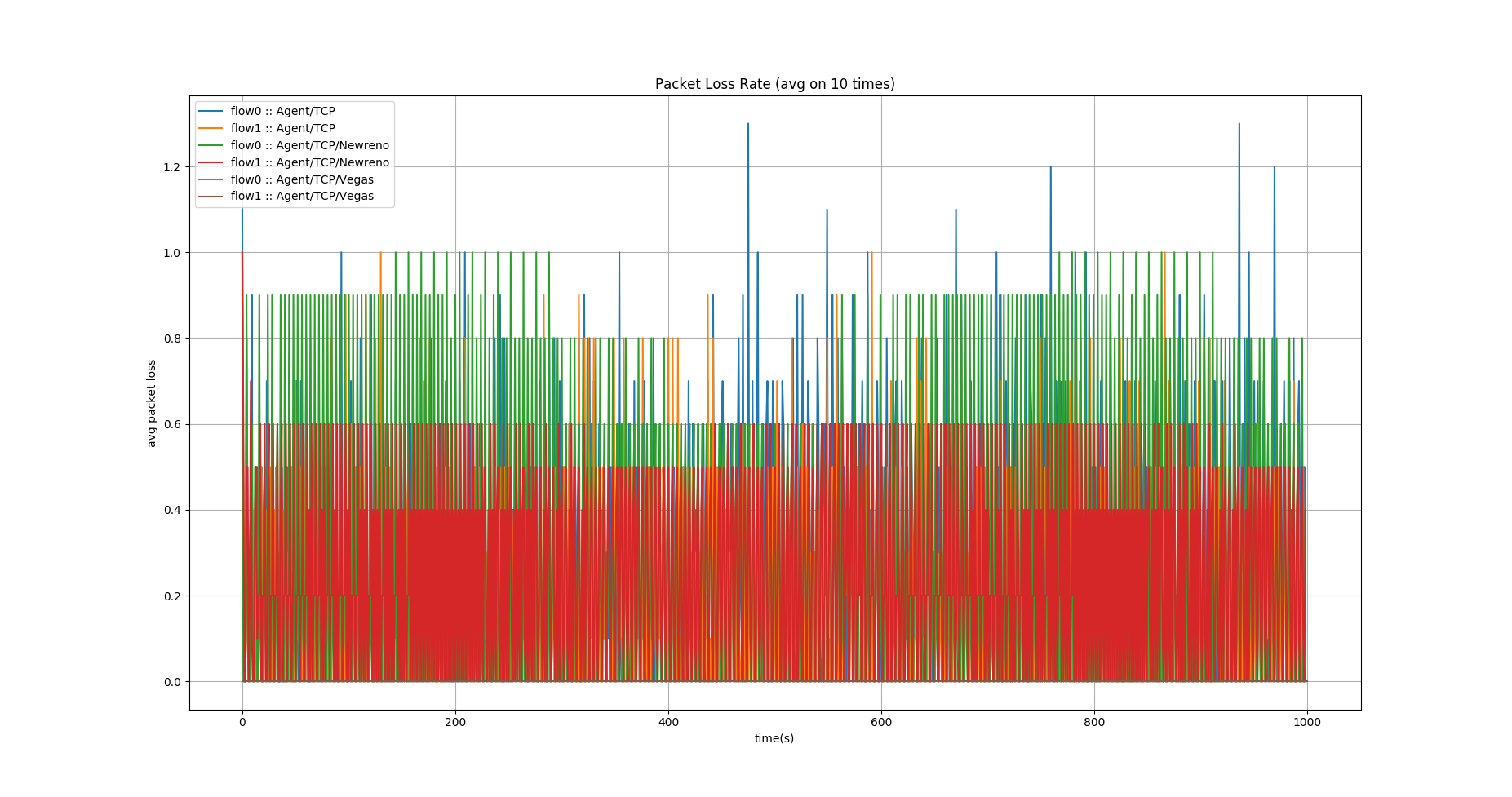
شکل 3- نرخ Goodput

برای تحلیل نرخ بستههای drop شده در واحد زمان میتوان گفت که در دو مورد Tahoe و New Reno آن ها با تغییراتی که در سایز پنجره ایجاد میکنند، سعی میکنند تعداد drop ها را کاهش دهند و طبق نمودار مشخصا موفق میشوند این کار را انجام دهند ولی در Vegas چون از 2 مقدار threshold استفاده میکند، خیلی سریع به سایز پنجره دلخواه میرسد و عملا packet drop در آن رخ نمیدهد.

شکل 4 میانگین نرخ بستههای drop شده در واحد زمان و شکل 5 تعداد بستههای drop شده در واحد زمان را نشان میدهد.



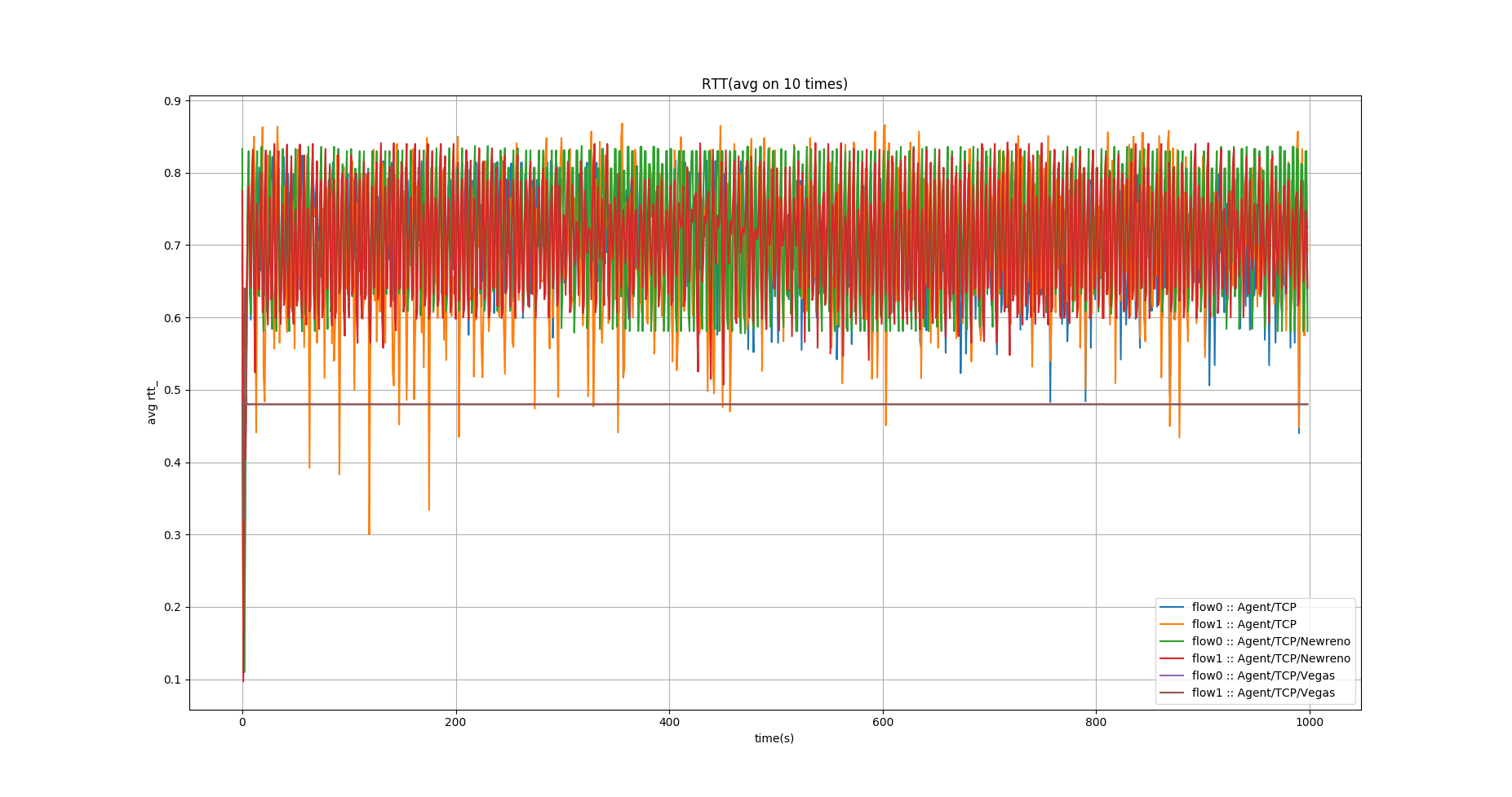
شکل 4- میانگین نرخ بستههای drop شده در واحد زمان



شکل5 - تعداد بستههای drop شده در واحد زمان

برای تحلیل نرخ RTT میتوان گفت که در دو مورد Tahoe و New Reno چون سایز پنجره آنها هیچگاه ثابت نمیشود، rtt آنها همیشه نوسان دارد.

ولی در Vegas چون سایز پنجره سریعا ثابت میشود، مقادیر rtt آن هم مطابق نمودار 6، سریعا ثابت میشود.



شکل 6 – نرخ rtt