

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه شاهرود

دانشکده مهندسی برق و کامپیوتر

گروه برق-کنترل

پایان نامه کارشناسی ارشد

کنترل کننده ازدحام تناسبی انتگرالی مشتقی مقاوم شبکه

منصوره پیوندی

استاد راهنما:

جناب آقای دکتر الفی

اسفند ۱۳۹۳

دانشگاه صنعتی شاهرود

دانشکده : برق و کامپیوتر

گروه : کنترل

پایان نامه کارشناسی ارشد آقای / خانم منصوره پیوندی

تحت عنوان:

کنترل کننده ازدحام تناسبی مشتقی انتگرالی مقاوم شبکه

در تاریخ توسط کمیته تخصصی زیر جهت اخذ مدرک کارشناسی ارشد

مورد ارزیابی و با درجه مورد پذیرش قرار گرفت.

امضاء	اساتید مشاور	امضاء	اساتید راهنما
	نام و نام خانوادگی :	دکتر الفی	نام و نام خانوادگی :
	نام و نام خانوادگی :		نام و نام خانوادگی :

امضاء	نماینده تحصیلات تکمیلی	امضاء	اساتید داور
	نام و نام خانوادگی :		نام و نام خانوادگی :
			نام و نام خانوادگی :
			نام و نام خانوادگی :
			نام و نام خانوادگی :

در آستانه بزرگداشت استاد ایم و کرمت را وسیله تقرب به حضور تو قرار داده ایم، پس به ذره ای از علم بی کرانت دانایان
کن.

قطره ی دانش که بنشیدی ز پیش متصل گردان به دریا های خویش

مولانا

بمشکروقدردانی از اساتذگرامی ام

جناب آقای دکتر انصاری

که راهنمایی های ایشان در این مسیر و مشنکر را هم بود.

تعهد نامه

اینجانب منصوره پیوندی دانشجوی دوره کارشناسی ارشد رشته برق کنترل دانشکده برق و کامپیوتر دانشگاه صنعتی شاهرود نویسنده پایان نامه "کنترل کننده ازدحام تناسبی مشتقی انتگرالی مقاوم شبکه" تحت راهنمایی دکتر الفی متعهد می‌شوم تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.

در استفاده از نتایج پژوهش‌های محققان دیگر به مرجع مورد استفاده استناد شده است.

مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.

کلیه حقوق معنوی این اثر متعلق به دانشگاه شاهرود می‌باشد و مقالات مستخرج با نام دانشگاه شاهرود به چاپ خواهد رسید.

حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان‌نامه تاثیرگذار بوده‌اند در مقالات مستخرج از پایان‌نامه رعایت شده است.

در کلیه مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری، ضوابط و اصول اخلاق انسانی رعایت شده است.

تاریخ

امضای دانشجو

مالکیت نتایج و حق نشر

کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه‌های رایانه‌ای، نرم‌افزارها و تجهیزات ساخته شده است) متعلق به دانشگاه شاهرود می‌باشد. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود. استفاده از اطلاعات و نتایج موجود در پایان‌نامه بدون ذکر مرجع مجاز نمی‌باشد.

چکیده

استفاده گسترده از پروتکل کنترل انتقال¹ (TCP) در اینترنت و رشد حجم اطلاعات فرستاده شده از طریق وب در طول سال‌های گذشته علایق محققان را برای مطالعه مشکلاتی چون ازدحام شبکه برانگیخته است. افزایش نرخ گم شدن بسته‌ها و کاهش بازدهی شبکه چالش‌های اساسی را برای کاربران ایجاد نموده است و قابلیت اطمینان شبکه را پایین آورده است. در این مطالعه ابتدا مسأله ازدحام و اهمیت کنترل آن بیان می‌شود و سپس روی راهکارهای پیشنهادی با استفاده از نظریه کنترل بحث می‌شود یعنی کنترل‌کننده با کمک مفاهیم ریاضی طراحی می‌شود و روی مسیریاب‌ها² قرار داده می‌شود. که این امر باعث کاهش احتمال افت بسته می‌شود. برای حصول این مطلوب، پیش‌بینی‌کننده اسمیث به عنوان بهترین ساختار که هم پایداری را تضمین می‌کند و هم با تاخیر به شکل مناسبی برخورد می‌کند روی سیستم پیاده‌سازی شده و عملکرد آن به تنهایی برای کنترل سیستم بررسی می‌شود و سپس کنترل‌کننده‌هایی نظیر داهلین³ و کنترل‌کننده تناسبی انتگرالی مشتقی⁴ و تناسبی انتگرالی مشتقی مقاوم⁵ برای مقایسه جهت رسیدن به اهداف طراحی در این ساختار قرار داده شده است. برای طراحی مقاوم از مفهوم بهره معادل بعلاوه تاخیر زمانی⁶ در ساختار اسمیث استفاده شده است که این دست‌خوش تغییراتی قرار می‌گیرد تا با ورود نامعینی‌های ناشناخته پارامتری، کنترل‌کننده PID موجود در این ساختار بتواند عملکرد قابل قبولی را ارائه دهد که از ساختار کلی آن تحت عنوان کنترل‌کننده PID مقاوم یاد می‌شود. در این مطالعه چون سیستم مربوط به شبکه است برای فهم دقیقتر و مشاهده عینی‌تر یک توپولوژی فرضی با دو گره (مسیریاب) و چندین فرستنده و دریافت‌کننده برای سیستم در نظر گرفته شده و در نرم‌افزار NS⁷ شبیه‌سازی

¹ Transmission control protocol

² Routers

³ Dahlin

⁴ Proportional Integral Derivative (PID)

⁵ Robust PID

⁶ EGPTD : Equivalent Gain Plus Time Delay

⁷ Network Simulator

شده است تا نتایج واقع گرایانه‌تری را در برداشته باشد، نتایج بدست آمده در این مطالعه حاکی از آن است که کنترل‌کننده تناسبی انتگرالی مشتقی مقاوم در شرایط مختلف شبکه عملکرد بهتری را نسبت به سایر کنترل‌کننده‌ها ارائه داده است و توانست در شرایط ازدحام به خوبی طول صف را در مقدار مطلوب نگه دارد و کمترین ظرفیت بافر مسیریاب برای انتقال اطلاعات را مورد استفاده قرار دهد. برای پیاده‌سازی کنترل‌کننده‌ها در این نرم‌افزار از تکنیک‌های تبدیل سیگنال آنالوگ به دیجیتال استفاده شده است، بعد از تهیه کد مربوط به کنترل‌کننده روی مسیریاب مربوط قرار داده شده و نتایج آن بررسی و تحلیل شده است.

کلمات کلیدی

ازدحام، کنترل‌کننده PID مقاوم، مسیریاب، پیش‌بینی‌کننده اسمیث، پروتکل کنترل انتقال

فصل اول

مقدمه ۱

فصل دوم

معرفی ساختار و مدل سیستم ۹

۱-۲- مفهوم شبکه ۱۰

۱-۱-۲- کاربرد شبکه‌های کامپیوتری ۱۰

۲-۱-۲- اجزای اصلی شبکه‌های انتقال داده ۱۰

۳-۱-۲- مسیریاب ۱۱

۴-۱-۲- اینترنت ۱۲

۲-۲- پروتکل ۱۳

۱-۲-۲- تاریخچه مدل چهار لایه‌ای TCP/IP ۱۳

۲-۲-۲- معرفی لایه‌های TCP/IP ۱۴

۱-۲-۲-۲- لایه اول از مدل TCP/IP لایه واسط شبکه ۱۵

۲-۲-۲-۲- لایه دوم از مدل TCP/IP لایه شبکه ۱۶

۳-۲-۲-۲- لایه سوم از مدل TCP/IP لایه انتقال ۱۷

۴-۲-۲-۲- لایه چهارم از مدل TCP/IP لایه کاربرد ۱۸

۳-۲- معرفی لایه اینترنت یا IP ۱۸

۴-۲- ازدحام ۱۹

۱-۴-۲- مفهوم ازدحام ۱۹

۱۹ تاخیر ۱-۱-۴-۲
۱۹ اتلاف بسته ۲-۱-۴-۲
۲۰ مکانیزم های کنترل ازدحام در شبکه ۲-۴-۲
۲۰ مکانیزم انتها به انتها ۱-۲-۴-۲
۲۱ مکانیزم لبه ای ۲-۲-۴-۲
۲۱ مکانیزم میانی ۳-۲-۴-۲
۲۲ دسته بندی دیگر از مکانیزم های کنترل ازدحام ۳-۴-۲
۲۲ الگوریتم های سمت منابع ۱-۳-۴-۲
۲۶ الگوریتم سمت مسیریاب ها ۲-۳-۴-۲
۳۳ ساختار کنترلی ۵-۲
۳۳ مدل دینامیکی TCP/IP ۶-۲
۳۵ خطی سازی مدل دینامیکی ۱-۶-۲

فصل سوم

۳۷ روش های کنترلی
۳۸ پیش بینی کننده اسمیث ۱-۳
۳۹ طراحی پیش بینی کننده اسمیث ۱-۱-۳
۴۰ معرفی پارامتر یولا ۱-۱-۱-۳
۴۴ کنترل کننده داهلین ۲-۳
۴۴ طراحی کنترل کننده داهلین ۱-۲-۳
۴۵ کنترل کننده PID ۳-۳
۵۱ معرفی ۱-۳-۳
۵۱ اهداف طراحی ۲-۳-۳

- ۳-۳-۳- تنظیم پارامترهای کنترل کننده PID ۵۲
- ۳-۴- کنترل کننده PID مقاوم ۴۷
- ۳-۴-۱- مفهوم بهره معادل بعلاوه تاخیر زمانی ۴۷
- ۳-۴-۲- تقریب با EGPTD ۴۹
- ۳-۴-۳- کاربرد EGPTD برای سیستم‌های مرتبه یک ۵۰

فصل چهارم

- شبیه‌سازی و مطالعات عددی در شبیه‌ساز NS ۵۳
- ۴-۱- معرفی NS ۵۴
- ۴-۲- زبان برنامه‌نویسی ۵۴
- ۴-۳- گسسته‌سازی ۵۵
- ۴-۴- روش‌های تقریب ۵۵
- ۴-۵- توپولوژی مورد مطالعه ۵۶
- آزمایش‌ها ۵۶
- ۴-۶- بررسی عملکرد کنترل کننده‌ها به ازای تغییرات پارامترهای شبکه ۶۹
- ۴-۶-۱- میزان انحراف معیار طول صف ۶۹
- ۴-۶-۲- مقدار متوسط طول صف ۷۱

فصل پنجم

- جمع‌بندی و پیشنهادات ۷۳
- پیوست الف ۸۰
- پیوست ب ۸۲
- پیوست پ ۸۶

مراجع ۸۳

فهرست اشکال

- ۱-۲: اجزای اصلی شبکه های انتقال داده..... ۱۳
- ۲-۲: الگوی چهارلایه ای TCP/IP..... ۱۷
- ۳-۲: شماتیک سرانجام بسته های فرستاده شده از مبدا به مقصد..... ۲۲
- ۴-۲: چگونگی کارکرد الگوریتم کنترل RED..... ۳۴
- ۵-۲: مدل پروسه کنترلی TCP..... ۳۶
- ۱-۳: شماتیک فرایند کنترلی با استفاده از تکنیک اسمیث..... ۴۳
- ۲-۳: ساختار کنترلی استاندارد..... ۴۴
- ۳-۳: بلوک دیاگرام ساختار اسمیث..... ۵۲
- ۴-۳: بلوک دیاگرام اصلاح شده ساختار اسمیث با استفاده از تکنیک EGPTD..... ۵۳
- ۱-۴: توپولوژی مورد مطالعه..... ۶۰
- ۲-۴: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم (آزمایش ۱)..... ۶۳
- ۳-۴: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم (آزمایش ۲)..... ۶۵
- ۴-۴: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم (آزمایش ۳)..... ۶۷

- ۴-۵ : توپولوژی برای ترافیک مخلوط از FTP و HTTP ۶۸
- ۴-۶: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم (آزمایش ۴)..... ۷۰
- ۴-۷: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم (آزمایش ۵) ۷۱
- ۴-۸ : انحراف معیار طول صف به ازای الف) تغییرات جریان (بار) ب) تغییرات RTT ج) تغییرات ظرفیت اتصال د) تغییرات پهنای باند اتصال..... ۷۴
- ۴-۹: مقدار متوسط طول صف به ازای الف) تغییرات جریان (بار) ب) تغییرات RTT ج) تغییرات ظرفیت اتصال د) تغییرات پهنای باند اتصال..... ۷۸

فصل اول

مقدمه

امروزه شبکه‌ها به تمام جنبه‌های زندگی ما نفوذ کرده‌اند. امکان برقراری اتصال وب در هر جایی و هر زمانی به یک ضرورت در زندگی روزمره و کسب و کار ما بدل شده است. اینترنت به عنوان بزرگترین شبکه موجود، مؤثرترین، ارزانتین و راحتترین وسیله برای جستجو و تبادل اطلاعات است [۱].

علیرغم اینکه شبکه‌ها سهولت و امکانات زیادی برای بشر فراهم کرده‌اند، ولی کیفیت سرویس دهی و راندمان آنها هنوز فاصله زیادی به نقطه مطلوب و ایده‌آل خود دارد. به عبارتی این شبکه‌ها هنوز از گم شدن بسته‌های اطلاعات، پهنای باند پایین و تأخیرهای طولانی رنج می‌برند [۲]. با توجه به رشد روزافزون استفاده از اینترنت و به خصوص با توجه به اهمیت اینترنت به عنوان ابزاری برای گسترش تجارت جهانی، تلاش‌های زیادی جهت حفظ کیفیت سرویس^۱ (QoS) در اینترنت در حال انجام می‌باشد.

با وجود گسترش فناوری‌های انتقال و افزایش پهنای باند، به مکانیزم‌هایی برای تضمین کیفیت سرویس کاربران نیاز می‌باشد. در حال حاضر اکثر تولیدکنندگان مسیریاب و کلیدهای شبکه اینترنت، در حال بررسی و افزودن مکانیزم‌هایی برای تضمین کیفیت سرویس در محصولات خود می‌باشند [۲]. مفهوم کیفیت سرویس همراه با توسعه شبکه‌های ارتباطی، تغییر زیادی نموده است. در روزهای آغازین ایجاد شبکه‌های کامپیوتری، ارسال بسته‌ها از مبدا به مقصد مهمترین و برجسته‌ترین هدف یک شبکه و دسترسی قابل اطمینان^۲ به شبکه، مساله اصلی در زمینه کیفیت سرویس بود. امروزه با رشد سریع شبکه‌ها، فاکتورهایی چون تاخیر در دریافت^۳، گم شدن و تلفات^۴ و... نیز به مفهوم کیفیت سرویس اضافه شده است که مسئول بسیاری از این مشکلات ازدحام است [۳]. پدیده ازدحام یکی از مشکلات پیش روی طراحان شبکه‌های کامپیوتری است و در سال‌های اخیر باعث کاهش کارایی شبکه‌ها شده است. ازدحام در شبکه زمانی اتفاق می‌افتد که بسته‌ها با سرعتی بالاتر از آنچه بتوان آنها

¹ Quality of Service

² Availability

³ Time Delay

⁴ Packet Loss

را انتقال داد به درگاه ورودی وارد شوند. به دلیل ظرفیت محدود بافر مسیریاب‌ها و پهنای باند، در هنگامی که درخواست سرویس در شبکه افزایش می‌یابد منابع شبکه قادر به پاسخ‌گویی نخواهند بود و اگر روش مناسبی برای تنظیم و مدیریت ترافیک وجود نداشته باشد ممکن است شبکه به کلی ناپایدار شود. در این شرایط طول صف تشکیل شده در مسیریاب‌ها که متشکل از بسته‌هایی است که می‌خواهند از مسیریاب/گلوگاه عبور کنند، آنقدر بزرگ می‌شود که تاخیر رسیدن بسته‌ها به مقصد از حد مجاز بیشتر می‌شود. حتی ممکن است شرایطی پیش آید که برخی بسته‌ها به مقصد نرسند. به عبارتی شبکه دچار افت بسته شده است. پس مسئله پایدارسازی طول صف در یک مقدار از پیش تعیین شده از اهمیت بالایی برخوردار است. یکی از مکانیزم‌های مهم تامین کیفیت سرویس و جلوگیری از وقوع ازدحام در شبکه‌های IP استفاده از مکانیزم‌های مدیریت فعال صف و زمان‌بندی در مسیریاب‌های IP می‌باشد با کمک این مکانیزم‌ها، وقوع ازدحام با کاهش طول صف در مسیریاب‌ها کنترل شده و از کاهش کارایی شبکه جلوگیری می‌شود [۴]. در اکثر شبکه‌های IP از پروتکل TCP استفاده می‌شود. اینترنت به عنوان بزرگترین شبکه موجود از پروتکل فوق به منظور ارتباط دستگاه‌های متفاوت استفاده می‌نماید. امروزه پروتکل TCP پر کاربردترین و مشهورترین پروتکل انتقال است. از این رو مسئله کنترل ازدحام در شبکه‌های مبتنی بر TCP از اهمیت بالایی برخوردار است [۶].

الگوریتم‌های متعددی برای مدیریت فعال صف در حیطه مهندسی ترافیک مربوط شبکه‌های کامپیوتری ارائه شده است. از جمله می‌توان به مکانیزم^۱ RED اشاره کرد که به موجب آن نقاط میانی شبکه قبل از پر شدن صف اقدام به حذف برخی بسته‌ها به صورت تصادفی می‌نمایند [۷]. محققان زیادی در طول تحقیقات خود دریافتند که این روش در برخی موارد به اندازه لازم کارایی مورد نیاز را ندارد و منجر به نوسانی شدن طول صف می‌گردد [۸]. البته مکانیزم‌های دیگری نیز در طول زمان ارائه شدند که بر پایه RED بودند و هرکدام ضعف‌ها و محدودیت‌های خاصی داشتند [۴]. روش‌های

¹ Random Early Detection

جدیدتری نیز برای کنترل صف ارائه شده است از جمله BLUE [۹] و GREEN [۱۰] و REM [۵]. در کنار این روش‌ها در گذشته الگوریتم‌هایی برای کنترل ازدحام سمت منابع، در TCP معرفی شده است. این الگوریتم‌ها همگی بر پایه تنظیم اندازه پنجره ارسال لغزشی طراحی شده‌اند و بر اساس ترافیک شبکه (دریافت خبر رسیدن بسته‌ها و یا تمام شدن مهلت ناشی از نرسیدن آنها) اندازه پنجره ارسالی و اندازه پنجره گیرنده را تنظیم می‌کنند. به عبارت دیگر همگی این روش‌ها بر پایه اندازه پنجره هستند [۶].

مهمترین و ساده‌ترین الگوریتم‌های TCP برای کنترل ازدحام در سمت منابع TCP Reno و TCP Tahoe می‌باشند. تمامی این الگوریتم‌ها بوسیله گم شدن بسته‌ها ترافیک را تشخیص می‌دهند، درحالی‌که الگوریتم‌های در سمت مسیریاب‌ها، بر اساس طول صف ترافیک را می‌سنجند. الگوریتم Tahoe در سال ۱۹۸۸ توسط جاکوبسن^۱ ارائه شد. در این روش ابتدا ارسال در حالت "شروع آهسته"^۲ انجام می‌شود، در این حالت با ارسال تنها چند بسته اتصال برقرار می‌شود، وقتی که پیغام دریافت از گیرنده رسید، فرستنده پنجره ازدحام را دو برابر می‌کند و این روند تا جایی ادامه پیدا می‌کند که اندازه پنجره ازدحام به سطح آستانه برسد. از اینجا به بعد، با دریافت پیغام رسیدن بسته هر بار فقط یک واحد به اندازه پنجره ازدحام افزوده می‌شود، این مرحله را "پرهیز از ازدحام"^۳ می‌نامند. مرحله افزایش یک واحدی پنجره ارسال تا آنجا ادامه می‌یابد که بسته‌ای مفقود شود. در این صورت بلافاصله سطح آستانه نصف مقدار فعلی پنجره ازدحام شده و فرستنده به مرحله شروع آهسته می‌رود.

هر بار مشخص شود که بسته‌ای خاص مفقود شده، فرستنده آن را دوباره ارسال می‌کند و به فاز شروع آهسته می‌رود. وقتی اعلام دریافت آن بسته توسط گیرنده به فرستنده رسید، فرستنده عمل ارسال را از ادامه آن اعلام دریافت، انجام می‌دهد (با اینکه ممکن است برخی بسته‌ها را در این فاصله

¹ Jacobsen

² Slow Start

³ Congestion Avoidance

ارسال کرده باشد) ولی ضعف روش Tahoe این است که به دلیل از دست رفتن متعدد بسته‌ها، فرستنده مدت زیادی را در مرحله شروع آهسته خواهد بود و بهره‌برداری مناسبی از عرض باند شبکه نمی‌شود [۱۱]. الگوریتم Reno نیز که در سال ۱۹۹۰ توسط جاکوبسن معرفی شد. در این الگوریتم به ازای گم‌شدن هر بسته به فاز شروع آهسته نمی‌رویم، بلکه وقتی که سه اعلام وصول پشت سرهم (تکراری) دریافت شد، فرستنده بسته گمشده را دوباره می‌فرستد، پنجره اطلاعات را نصف می‌کند و به فاز "بازیابی سریع"^۱ می‌رود. هنگامی که اولین اعلام دریافت غیرتکراری دریافت شد از فاز بازیابی خارج و به فاز جلوگیری از تراکم می‌رویم. در TCP Reno تنها در حالتی استفاده از بازیابی سریع بهینه است که تنها یک بسته از فرستنده گم شود. هنگامی که از یک پنجره چند بسته گم شود، این الگوریتم تا تمام شدن مهلت آنها صبر کرده و به این ترتیب از عرض باند موجود استفاده کافی نخواهد شد [۱۲]. در کنار اینها الگوریتم دیگری چون TCPW^۲ ارائه شد که یک نوع اصلاح و تعدیل TCP است که جدیداً بسط و گسترش یافته و در سمت منابع اعمال می‌شود [۱۳]. هدف از مطرح کردن این روش دستیابی به سهم بیشتری از عرض باند شبکه می‌باشد. ایده اصلی، تخمین عرض باندی است که به صورت موفقیت‌آمیز توسط منبع مورد استفاده قرار می‌گیرد و این تخمین از طریق جریان Ack^۳ ها صورت می‌گیرد. با این روش اگر تأخیر در صف ماندن وجود نداشته باشد، اندازه پنجره کاهش نمی‌یابد و این یک مزیت نسبت به عملکرد سایر پروتکل‌ها می‌باشد. ضعف‌های روش تخمین عرض باند بیشتر مربوط به حالتی می‌شوند که به دلایلی از قبیل ترافیک‌های تهاجمی و یا فشرده‌سازی بسته‌ها عرض باند موجود بیش از اندازه صحیح تخمین زده شود [۴].

توانایی ذاتی علم کنترل و استفاده از مدل‌های ریاضی در شناسایی سیستم‌ها، همواره موجب به وجود آمدن شناختی دقیق‌تر از آن سیستم‌ها گردیده است به همین دلیل ورود آن به حیطه کنترل ترافیک در شبکه‌های ارتباطی، طبیعی به نظر می‌رسد. از آنجایی که شبکه‌های ارتباطی را می‌توان به

^۱ Fast Recovery

^۲ TCP Westwood

^۳ Acknowledgement

صورت یک سیستم دینامیکی مدل نمود، علم کنترل در این زمینه توانسته بسیار مشکل گشا باشد [۶].

از اواخر دهه ۱۹۹۰ میلادی، ایده به کارگیری مفاهیم نظریه کنترل در حل مسائل کنترل ازدحام مورد توجه قرار گرفته است، که هدف از آنها استفاده از ابزارهای موجود در مهندسی کنترل به منظور تحلیل و طراحی کنترل‌کننده‌های ازدحام مناسب برای شبکه‌ها به عنوان سیستم‌های حلقه بسته است [۱۵]، به طوری که با استفاده از مدل‌های ریاضی مناسب می‌توان رفتار سیستم در حالت‌ها و شرایط مختلف پیش بینی نمود و سیاست‌های لازم را در برخورد با چنین رفتاری، از قبل اتخاذ نمود. شبکه TCP به عنوان یک سیستم غیرخطی پیچیده که دینامیک‌های آن در حال تغییر می‌باشند، همواره مورد توجه محققان بوده است. روش‌های زیادی در جهت شناسایی مدل شبکه‌های کامپیوتری به صورت ریاضی صورت گرفته است. اما شاید بتوان یکی از بهترین مدل‌هایی که در این زمینه ارائه شده است را مدل میسرا^۱ دانست که در سال ۲۰۰۰ میلادی ارائه شده است [۱۶].

این مدل بر اساس مدل جریان سیال به دست آمده است و یک مدل غیرخطی از شبکه ارائه می‌دهد. این مدل ریاضی موجب به وجود آمدن شاخه‌ای جدید از تحقیقات جهت ارائه روشی برای مدیریت فعال صف گردید. با ارائه این مدل افراد مختلف کوشیدند تا از نظریه کنترل جهت ارائه روشی برای کنترل فعال صف استفاده کنند. ایده اصلی آن بود که از بین بردن تصادفی بسته‌ها در صف قبل از پر شدن صف، از یک قاعده کنترلی پیروی نماید. اولین کنترل‌کننده ارائه شده PI^۲ بود [۱۸] که از مدل خطی شده شبکه برای طراحی کنترل‌کننده استفاده گردید. سادگی روش PI و کارایی آن نسبت به RED موجب پیاده‌سازی تجاری آن شد.

مطالعات و شبیه‌سازی‌ها نشان داد که کنترل‌کننده ارائه شده نسبت به روش RED نتایج بهتری را ارائه می‌دهد. متأسفانه تحقیقات بعدی حاکی از آن بود که کنترل‌کننده PI در برخورد با تغییر پارامترها و همچنین زیاد شدن منابع شبکه به اندازه کافی مقاوم عمل نمی‌کند [۱۹]. از این رو

^۱ Misra

^۲ Proportional-Integral

محققان کوشیدند روش‌های مقاومتری را ارائه نمایند که در برابر تغییرات شبکه به اندازه کافی مقاوم باشد. در نتیجه روش PID مطرح شد. شبیه‌سازی‌ها نشان داد که کنترل‌کننده PID عملکرد بهتری را نسبت به کنترل‌کننده PI و RED ارائه می‌دهد و فرایند با تغییر پارامترها نوسان کمتری را دارد که منجر به تنظیم صف بهتر می‌شود [۲۰]. در روند کاهش ازدحام در شبکه، یک کنترل‌کننده مقاوم که با استفاده از تکنیک کنترل H^∞ طول صف را در نزدیکی نقطه کار نگه می‌دارد، معرفی شد که البته با این کنترل‌کننده نیز با وجود مقاوم بودن احتمال افت بسته وجود دارد [۲۱]. طراحی کنترل‌کننده PID یا به عبارتی تعیین پارامترهای این کنترل‌کننده به روش‌های مختلفی انجام می‌شود. از آنجایی که مقاوم بودن این کنترل‌کننده در شرایط نوسانی شبکه یا وجود نامعینی‌ها بسیار حائز اهمیت است به دنبال یک روش طراحی مقاوم هستیم.

بیشتر پروسه‌های صنعتی به وسیله کنترل‌کننده‌های PID کنترل می‌شوند. محبوبیت کنترل‌کننده‌های PID را می‌توان تا حدی به عملکرد مقاوم آنها در رنج وسیعی از شرایط عملیاتی نسبت داد. اما این واقعیت که کنترل‌کننده فوق برای پروسه‌های با تاخیر زمانی زیاد پاسخگو نیست استفاده از این کنترل‌کننده را برای این نوع سیستم‌ها به چالش می‌کشد [۲۲].

بنابراین به دنبال راه‌حلی برای این مسئله، بر آن شدیم تا با پیاده‌سازی روش‌های کنترلی نظیر کنترل‌کننده PID مقاوم به طریق دیگری با این مسائل در این نوع شبکه‌ها برخورد کنیم و مزایا و معایب آن را با دیگر روش‌های مرسوم در مهندسی کنترل، مورد مقایسه و بررسی قرار دهیم. اما قبل از کنترل صف در این نوع شبکه‌ها باید به طریقی با تاخیر که در مدل سیستم وجود دارد برخورد کنیم. از راه کارهای موجود می‌توان به تقریب‌های متفاوت تاخیر در شبکه اشاره کرد. البته روش مورد استفاده در این تحقیق پیش‌بینی‌کننده اسمیث^۱ (۱۹۵۹) است. روش اسمیث یک رویکرد کلاسیک است که برای طراحی کنترل‌کننده‌های سیستم‌های دارای تأخیر بکار می‌رود به طوری که

¹ Smith Predictor

با استفاده از یک تغییر شکل مناسب در شماتیک کنترلی، تاخیر را از حلقه کنترل خارج می‌کند [۲۳].

هدفی که در این پایان‌نامه دنبال می‌کنیم، پیاده‌سازی ساختارهای کنترلی مختلف از جمله کنترل‌کننده داهلین، PID معمولی و... روی شبکه و مقایسه نتایج آنها با کنترل‌کننده PID مقاوم است.

ساختار پایان‌نامه به شرح ذیل است:

در فصل دوم به بررسی و شناسایی دقیق تر شبکه و پروتکل حاکم بر آن پرداخته می‌شود. همچنین، به اختصار چگونگی عملکرد TCP و محدودیت‌ها و موانع آن برای کنترل ازدحام شرح داده می‌شود. اهداف، حوزه کاری و اهمیت این تحقیق نیز در این فصل ذکر می‌شود.

در فصل سوم به مطالعه ساختارهای کنترلی موجود و روش طراحی هرکدام از آنها پرداخته می‌شود. در فصل چهارم به منظور نمایش واقع‌گرایانه از عملکرد کنترل‌کننده‌ها، ساختارهای کنترلی موردنظر در نرم‌افزار NS روی سیستم پیاده و شبیه‌سازی می‌شود.

در فصل پایانی نتایج بدست آمده از شبیه‌سازی مورد بررسی و تحلیل قرار می‌گیرد. نتایج اجرای روش پیشنهادی با نتایج حاصل از روش‌های دیگر مقایسه می‌گردد و با توجه به آن، نقاط ضعف و قوت روش پیشنهادی بررسی می‌شود. همچنین جمع‌بندی این تحقیق و نظرات نهایی درباره موضوع مورد مطالعه و دستاوردهای آن را در بر خواهد داشت و برخی از توصیه‌ها و کارهای آینده‌ای که می‌تواند در رابطه با این تحقیق مطرح شود نیز در پایان این فصل آورده می‌شود.

فصل دوم

معرفی ساختار و مدل سیستم

۲-۱- مفهوم شبکه

شبکه کامپیوتری، مجموعه‌ای از کامپیوترهای مستقل است که از طریق یک بستر انتقال با یکدیگر به تبادل داده می‌پردازند [۲۴].

۲-۱-۱- کاربرد شبکه‌های کامپیوتری

خدمات شبکه‌های کامپیوتری به قدری فراگیر شده است که تمام شئون زندگی را تحت تأثیر قرار داده است. با توسعه اینترنت یا شبکه‌های ملی، محیط زیست می‌تواند نجات یابد، رفت و آمدهای بهبود یافته به منظور پیگیری کارهای اداری و اقتصادی زندگی حذف می‌شود. به طور عام کاربردهای شبکه‌های کامپیوتری را می‌توان در موارد زیر خلاصه کرد:

- اشتراک منابع
 - حذف محدودیت‌های جغرافیایی در تبادل داده‌ها
 - کاهش هزینه‌ها
 - بالا رفتن قابلیت اعتماد سیستم‌ها
 - افزایش کارایی سیستم
- خدماتی که شبکه‌ها ارائه می‌کنند بسیار وسیع و از شمار خارج هستند ولی عمومی‌ترین نوع کاربرد شبکه‌ها را میتوان به مواردی مثل، کنفرانس از راه دور، تلفن و دورنگار از طریق شبکه، درمان از راه دور، پست الکترونیکی، بانکداری الکترونیکی و... اشاره کرد [۴].

۲-۱-۲- اجزای اصلی شبکه‌های انتقال داده

مطابق با شکل ۲-۱ هر سیستم انتقال داده از چهار قسمت اصلی تشکیل شده است که عبارتند

از:

- فرستنده و گیرنده

- داده‌های ارسالی
- محیط فیزیکی برای تبادل داده‌ها
- پروتکل استفاده شده برای ارسال داده‌ها



شکل ۲-۱: اجزای اصلی شبکه‌های انتقال داده [۲]

در یک شبکه انتقال داده، فرستنده با استفاده از تجهیزات ارسال، اقدام به تبدیل داده‌های ۰ و ۱ خام به سیگنال‌های الکتریکی یا نوری قابل ارسال در محیط فیزیکی می‌نماید. محیط‌های فیزیکی متعددی برای ارسال داده‌ها وجود دارند. در گیرنده نیز با استفاده از تجهیزات مناسب، سیگنال‌های دریافتی به صورت داده‌های اولیه تبدیل می‌شوند. برای مبادله صحیح داده‌ها بین فرستنده و گیرنده، هر دو باید از یک سری قوانین مشترک استفاده نمایند. این قوانین توسط پروتکل‌های ارتباطی مشخص می‌شود. هم فرستنده و هم گیرنده باید از پروتکل ارتباطی یکسانی برای ارسال و دریافت داده‌ها استفاده نمایند، در غیر این صورت تبادل داده‌ها میسر نمی‌باشد [۲۴].

۳-۱-۲- مسیر یاب

مسیریاب، مسیر یک بسته اطلاعاتی ارسالی توسط یک کامپیوتر برای کامپیوتر دیگر را تعیین می‌کند. مسیریاب‌ها کامپیوترهای خاصی می‌باشند که پیام‌های ارسال شده توسط کاربران اینترنت با وجود هزاران مسیر موجود را مسیریابی و در اختیار دریافت کنندگان مربوطه قرار خواهند داد.

مسیریاب دو کار اساسی را در شبکه انجام می‌دهد:

- ایجاد اطمینان در رابطه با عدم ارسال اطلاعات به مکان‌هایی که به آنها نیاز نمی‌باشد.
 - اطمینان از ارسال صحیح اطلاعات به مقصد مورد نظر
- مسیریاب‌ها به منظور انجام عملیات فوق، می‌بایست دو شبکه مجزا را به یکدیگر متصل نمایند. مسیریاب باعث ارسال اطلاعات یک شبکه به شبکه دیگر، حفاظت شبکه‌ها از یکدیگر و پیشگیری از ترافیک می‌گردد. با توجه به اینکه اینترنت از هزاران شبکه کوچکتر تشکیل شده است، استفاده از مسیریاب یک ضرورت است.

۲-۱-۴ - اینترنت

اینترنت مجموعه‌ای از شبکه‌های مستقل و مرتبط با یکدیگر است که ارتباطات همگانی را میسر کرده است. تکنولوژی اینترنت مجموعه‌ای از مکانیزم‌ها و استانداردهای ارتباطی است که در خلال این سال‌ها سعی بر آن داشته تا روشی را عرضه کند که شبکه‌های مختلف بتوانند فارغ از جزییات سخت افزاری و نرم افزاری با یکدیگر مبادله داده داشته باشند. اینترنت در یک رشد هماهنگ با علوم مخابرات و کامپیوتر به ناگاه تبدیل به شبکه‌ای از کل شبکه‌های جهان شد. به گونه‌ای که تخمین زده می‌شود در هر روز بیش از ۱۵۰ میلیون نفر در ۱۱۰ کشور دنیا حداقل یکبار از اینترنت استفاده می‌کنند [۲۴]. با توجه به اینکه رشد اینترنت به دستگاه یا دولت خاصی وابسته نیست، هیچ شاخص معینی برای رشد اینترنت وجود ندارد. یک فرد حقیقی نیز می‌تواند به نوبه خود اینترنت را توسعه بدهد. مثلاً یک هنرمند یا بازرگان می‌تواند با سرمایه‌گذاری و نصب شبکه محلی و ارائه آگاهی و دانش به سهم خود به گسترش اینترنت کمک کند و هیچ محدودیتی برای سرمایه‌گذاری وی وجود ندارد.

سال ۱۹۹۲ یکی از پر فروغ‌ترین سال‌های گسترش اینترنت بود. تا قبل از این تاریخ بیشتر افراد به تناسب نوع نیاز خود از شبکه استفاده می‌کردند. با عرضه محیط تور جهان‌گستر یا وب در این سال

توجه مردم عادی نیز به اینترنت جلب شد و بدین گونه از اینترنت یک رسانه ارتباط جمعی پدید آمد و سیل سرمایه‌ها را بسوی خود کشید.

۲-۲- پروتکل

پروتکل عبارتست از کلیه قراردادهای توافق شده بین دو لایه همتا برای برقراری و پیشبرد یک ارتباط. این قراردادها عبارتند از پایبندی به یک الگوی دقیق و واحد در مورد قالب هر پیام، مفهوم و تعبیر پیام‌ها، شکل و زمانبندی صحیح مبادله پیام‌ها بین دو لایه همتا [۲۴]. عاملی که تمام شبکه‌های مختلف را به صورت موفقیت آمیز به هم پیوند زده است، تبعیت همه آنها از مجموعه پروتکلی است که تحت عنوان TCP/IP در دنیا شناخته می‌شود. این پروتکل برای آن طراحی شد تا کامپیوترهای مختلف با اندازه، نشان تجاری و سیستم‌های عامل متفاوت بتوانند با یکدیگر ارتباط برقرار کنند. به عبارت دیگر TCP/IP مستقل از طراحی هر سخت‌افزار یا نرم‌افزاری طراحی شده است. TCP/IP بهترین پروتکل شبکه‌بندی دنیا نیست. پروتکل‌های بهینه‌تر از آن هم وجود دارند، لیکن فراگیرترین تکنولوژی شبکه‌بندی، در دنیای کامپیوتر محسوب می‌شود. شاید بزرگترین حسن TCP/IP آن باشد که بدون پیچیدگی زیاد، به خوبی کار می‌کند. اینترنت بر اساس TCP/IP بنا شده است. به عبارتی زیر بنای اینترنت ساختار چهار لایه‌ای TCP/IP می‌باشد.

۲-۲-۱- تاریخچه مدل چهار لایه‌ای TCP/IP

TCP/IP زاده جنگ سرد در دهه شصت بود. در اواخر دهه شصت آژانس پروژه‌های پیشرفته تحقیقاتی دولت ایالات متحده آمریکا موسوم به ARPA^۱ با بودجه دولتی تصمیم به پیاده‌سازی یک شبکه WAN^۲ در ۹ دانشگاه و مرکز تحقیقاتی گرفت. این شبکه در ابتدا صرفاً اهداف نظامی را دنبال می‌کرد

^۱ Advanced Research Project Agency

^۲ Wide Area Network

و در عرض دو سال پیاده‌سازی و نصب شد. در این شبکه برای اولین بار روش کلید بسته به کار رفت تا این الگوی کلیدزنی که در آن زمان با شک و تردید زیادی همراه بود تولد خود را جشن بگیرد. موفقیت کلید بسته، مراکز تحقیقاتی مختلف را بر آن داشت تا برای توسعه این نوع از تکنولوژی شبکه شروع به کار مشترک کنند و این یک آغاز نوید بخش برای شبکه‌ای بود که امروز به اینترنت موسوم است. کمیته ARPA که به^۱ ICCB مشهور شد روز به روز شهرت یافت و رشد کرد. این کمیته با همکاری بقیه آژانس‌های تحقیقاتی، کار مشترک تبدیل تکنولوژی ARPA به یک پروتکل استاندارد شبکه به نام TCP/IP را شروع کردند [۲۴].

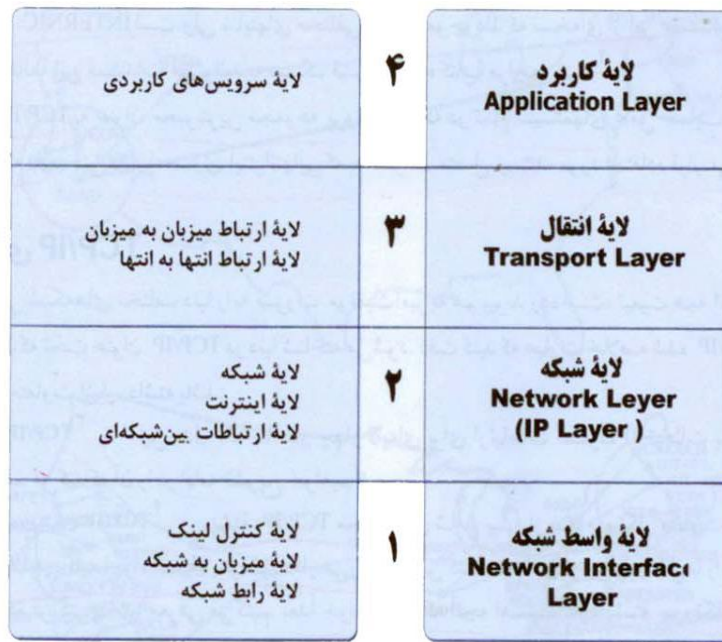
در اوایل دهه هشتاد محیط‌های دانشگاهی نیز از TCP/IP حمایت کردند. دانشگاه برکلی در کالیفرنیا در نسخه یونیکس خود که رایگان بود، پروتکل TCP/IP را پیاده‌سازی و ارائه کرد. رایگان بودن این پروتکل بسیار قدرتمند باعث شد تا دانشجویان و پژوهشگران علوم کامپیوتری به سرعت با TCP/IP آشنا شده و ضمن پیاده‌سازی شبکه‌های مبتنی بر آن از این مدل حمایت نمایند. شاید بزرگترین عامل توسعه و رشد TCP/IP همت دانشگاه برکلی و دانشجویان کامپیوتر در ارائه رایگان TCP/IP بر روی یونیکس بود [۲۲].

امروزه TCP/IP با تمام نواقصی که دارد به عنوان محبوبترین مجموعه پروتکل شبکه در تمام سیستم‌های عامل حمایت می‌شود.

۲-۲-۲ - معرفی لایه‌های TCP/IP

مدل TCP/IP یک الگوی چهار لایه‌ای برای شبکه مطابق به شکل ۲-۲ عرضه کرده است.

^۱ Internet Control and Configuration Board



شکل ۲-۲: الگوی چهارلایه‌ای TCP/IP [۲]

۱-۲-۲-۲- لایه اول از مدل TCP/IP لایه واسط شبکه

در این لایه استانداردهای سخت‌افزار، نرم‌افزارهای راه‌انداز^۱ و پروتکل‌های شبکه تعریف می‌شود. این لایه درگیر با مسائل فیزیکی، الکتریکی و مخابراتی کانال انتقال، نوع کارت شبکه و راه‌اندازهای لازم برای نصب کارت شبکه می‌باشد. در شبکه اینترنت که می‌تواند مجموعه‌ای از عناصر غیرهمگن و نامشابه را به هم پیوند بزند انعطاف لازم در این لایه برای شبکه‌های گوناگون و ماشین‌های میزبان فراهم شده است. یعنی الزام ویژه‌ای در بکارگیری سخت‌افزار ارتباطی خاص، در این لایه وجود ندارد. ایستگاهی که تصمیم دارد به اینترنت متصل شود بایستی با استفاده از پروتکل‌های متعدد و معتبر و نرم‌افزار راه‌انداز مناسب، به نحوی داده‌های خودش را به شبکه تزریق کند. بنابراین، اصرار و اجبار خاصی در استفاده از یک استاندارد خاص در این لایه وجود ندارد. تمام پروتکل‌های LAN و MAN در این لایه قابل استفاده خواهند بود.

¹ Device Driver

یک ماشین میزبان می‌تواند از طریق شبکه محلی^۱، فریم‌های اطلاعاتی را به زیرشبکه^۲ تزریق کند، به این نحو که بسته‌های راه دور را که مقصدشان خارج از شبکه محلی است، به مسیریاب از پیش تعریف شده، هدایت نماید. شبکه‌های محلی از طریق یک یا چند مسیریاب می‌توانند به اینترنت متصل شوند. بنابراین یک بسته اطلاعاتی که از لایه بالاتر جهت ارسال به یک مقصد، به لایه اول در مدل TCP/IP تحویل می‌شود، نهایتاً در قسمت فیلد داده، از فریم شبکه محلی قرار می‌گیرد و مسیر خود را آغاز می‌نماید. پروتکل‌هایی که در لایه اول از مدل TCP/IP تعریف می‌شوند، می‌توانند مبتنی بر ارسال رشته بیت یا مبتنی بر ارسال رشته بایت^۳ باشند [۲۵].

۲-۲-۲-۲ لایه دوم از مدل TCP/IP لایه شبکه

این لایه در ساده‌ترین عبارت وظیفه دارد بسته‌های اطلاعاتی را که از این به بعد آنها را بسته‌های IP می‌نامیم، روی شبکه هدایت کرده و از مبدا تا مقصد به پیش ببرد. در این لایه چندین پروتکل در کنار هم وظیفه مسیریابی و تحویل بسته‌های اطلاعاتی، از مبدا تا مقصد را انجام می‌دهند. کلیدی‌ترین پروتکل در این لایه، پروتکل IP نام دارد. برخی از پروتکل‌های مهم که یک سری وظائف جانبی بر عهده دارند عبارتند از BOOTP، IGMP، ICMP، RIP، RARP، ARP و نظایر آنها، که پروتکل IP را در عملکرد بهتر، مسیریابی صحیح، مدیریت خطاهای احتمالی یا کشف آدرس‌های ناشناخته کمک می‌کنند.

در این لایه یک واحد اطلاعاتی که بایستی تحویل مقصد شود، دیتاگرام^۴ نامیده می‌شود. پروتکل IP می‌تواند یک دیتاگرام را در قالب بسته‌های کوچکتری قطعه قطعه کرده و پس از اضافه کردن اطلاعات لازم برای بازسازی، آنها را روی شبکه ارسال کند.

¹ LAN

² Sub Network

³ Bit oriented & Byte oriented

⁴ Datagram

در این لایه برقراری ارتباط بین مبدا و مقصد به روش بدون اتصال خواهد بود و ارسال یک بسته IP روی شبکه، عبور از مسیر خاصی را تضمین نمی‌کند، یعنی اگر دو بسته متوالی برای یک مقصد یکسان ارسال شود هیچ تضمینی در به ترتیب رسیدن آنها وجود ندارد، چون این دو بسته می‌توانند از مسیرهای متفاوتی به سمت مقصد حرکت نمایند. در ضمن در این لایه پس از آنکه بسته‌ای روی یکی از کانال‌های ارتباطی هدایت شد، از سالم رسیدن یا نرسیدن آن به مقصد هیچ اطلاعی به دست نخواهد آمد، چرا که در این لایه، برای بسته‌های IP، هیچ گونه پیغام دریافت یا عدم دریافت^۱ بین عناصر واقع بر روی مسیر، رد و بدل نمی‌شود، بنابراین سرویسی که در این لایه ارائه می‌شود نامطمئن است. اگر به سرویس‌های مطمئن و یا اتصال‌گرا نیاز باشد، در لایه بالاتر این نیاز تامین خواهد شد. در این لایه مسیرها بایستی از شرایط توپولوژیکی و ترافیکی شبکه اطلاعاتی را کسب نمایند تا مسیریابی به روش پویا انجام شود. همچنین در این لایه باید اطلاعاتی درباره مشکلات یا خطاهای احتمالی در ساختار زیر شبکه بین مسیرها و ماشین‌های میزبان، مبادله شود [۲۵].

۲-۲-۳- لایه سوم از مدل TCP/IP لایه انتقال

این لایه ارتباط ماشین‌های انتهایی (ماشین‌های میزبان) را در شبکه برقرار می‌کند یعنی می‌تواند بر اساس سرویسی که لایه دوم ارائه می‌کند یک ارتباط اتصال‌گرا و مطمئن، برقرار کند. البته در این لایه برای عملیاتی نظیر ارسال صوت و تصویر که سرعت مهمتر از دقت و خطاست، سرویس‌های بدون اتصال سریع و نامطمئن نیز فراهم شده‌است که از طرف دیگر با لایه کاربرد در ارتباط است. داده‌های تحویلی به این لایه توسط برنامه کاربردی و با صدا زدن توابع سیستمی تعریف شده در واسط برنامه‌های کاربردی یا API^۲ می‌باشد، ارسال یا دریافت می‌شوند [۲۵].

^۱ Ack/Nack

^۲ Application Program Interface

۲-۲-۴- لایه چهارم از مدل TCP/IP لایه کاربرد

در این لایه بر اساس خدمات لایه‌های زیرین، سرویس سطح بالایی برای خلق برنامه‌های کاربردی ویژه و پیچیده ارائه می‌شود. این خدمات در قالب، پروتکل‌های استاندارد مانند موارد زیر به کاربر ارائه می‌شود شبیه‌سازی ترمینال یا همان TELNET انتقال فایل^۱، مدیریت پست الکترونیکی، خدمات انتقال صفحات ابرمتنی و ده‌ها پروتکل کاربردی دیگر که ارسال یک واحد اطلاعاتی از لایه چهارم پس از انجام پردازش‌های لازم در لایه‌های زیرین به نحو مناسبی روی زیر شبکه تزریق شده و نهایتاً در ماشین مقصد، تحویل یک برنامه کاربردی خاص خواهد شد [۲۵].

۲-۳- معرفی لایه اینترنت یا IP

جوهره اینترنت به گونه‌ای شکل گرفته است که مجموعه‌ای از شبکه‌های خودکار^۲ را به همدیگر وصل می‌نماید. هیچ‌گونه ساختار حقیقی و ثابتی نمی‌توان برای اینترنت تصور کرد. این نکته را بایستی یادآور شویم که در شبکه اینترنت تعدادی از خطوط ارتباطی با پهنای باند (نرخ ارسال) بسیار بالا و مسیریاب‌های بسیار سریع و هوشمند برای پیکره شبکه جهانی اینترنت یک ستون فقرات تشکیل داده است. شبکه‌های منطقه‌ای و محلی پیرامون این ستون فقرات شکل گرفته و ترافیک داده آنها به نحوی از این ستون فقرات خواهد گذشت. ستون فقرات در شبکه اینترنت که با سرمایه‌گذاری عظیمی در آمریکا، اروپا و قسمت‌هایی از اقیانوسیه و آسیا ایجاد شده است، حجم بسیار وسیعی از بسته‌های اطلاعاتی را در هر ثانیه حمل می‌کنند و اکثر شبکه‌های محلی یا ارائه‌دهندگان سرویس که ISP^۳ نامیده می‌شوند، به نحوی با یکی از گره‌های این ستون فقرات در ارتباط هستند. به گونه‌ای که در بخش قبلی اشاره شد قراردادی که حمل و تردد بسته‌های اطلاعاتی و همچنین مسیریابی صحیح آنها را از مبدا به مقصد، مدیریت و سازماندهی می‌نماید پروتکل IP نام دارد. در

¹ File Transfer Protocol

² Autonomous

³ Internet Service Provider

حقیقت پروتکل IP که روی تمام ماشین‌های شبکه اینترنت وجود دارد بسته‌های اطلاعاتی را از مبدا تا مقصد هدایت می‌نماید [۲۵].

۲-۴- ازدحام

۲-۴-۱- مفهوم ازدحام

یکی از مسائلی که بعد از پیدایش اینترنت مطرح شد، ازدحام در شبکه بود. ازدحام هنگامی رخ می‌دهد که اجزای شبکه بطور همزمان و مداوم درخواست استفاده از منابع محدود شبکه نظیر پهنای باند اتصالات، بافر مسیریاب‌ها و کلیدها را داشته باشند و میزان منابع در دسترس نیز کمتر از مجموع درخواست‌ها باشد. بنابراین ترافیک بار از ظرفیت شبکه بیشتر شده و ازدحام اتفاق می‌افتد. به عبارت ساده‌تر، مواقعی که حجم ارسال و دریافت داده آن چنان زیاد می‌شود، که بار شبکه از منابع موجود در برخی از بخش‌های سیستم بیشتر شده و شبکه از عهده این بار بر نمی‌آید، اصطلاحاً پدیده ازدحام رخ می‌دهد. عدم کنترل ازدحام در شبکه، منجر به افت شدید شاخص‌های کارایی مانند گذردهی و بهره‌وری اتصالات شبکه، افزایش طول صف، تعداد بسته‌های گم شده و تاخیر می‌گردد. بر این اساس کنترل ازدحام در اینترنت همواره از موضوعات تحقیقاتی بسیار داغ می‌باشد [۲۵].

۲-۴-۱-۱- تاخیر^۱

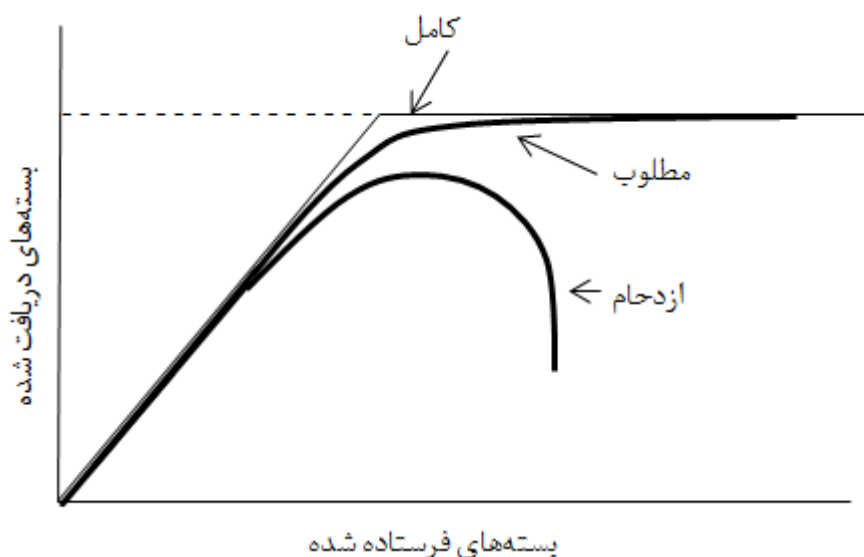
تاخیر میانگین کل زمانی است که طول می‌کشد تا یک بسته پس از تولید در مبدا، تحویل گیرنده نهائی آن در مقصد شود.

۲-۴-۱-۲- اتلاف بسته^۲

میانگین از بین رفتن بسته‌های متعلق به یک جریان واحد که به دلایل متعدد در زیر ساخت شبکه اتفاق می‌افتد به اتلاف بسته شهرت دارد [۲۵].

¹ Delay

² Packet Loss



شکل ۲-۳: شماتیک سرانجام بسته‌های فرستاده شده از مبدا به مقصد

۲-۴-۲- مکانیزم‌های کنترل ازدحام در شبکه

برای مدیریت مسئله‌ای مانند ازدحام، ابتدا باید تا حد امکان از وقوع آن جلوگیری و اجتناب نمود. ولی فارغ از پیچیدگی‌هایی که در عمل برای این کار وجود دارد، بسیاری از روش‌های اجتناب از ازدحام موجب کاهش بازدهی سیستم می‌شوند. بدین خاطر روش‌هایی کنترلی ارائه شده‌اند، که عدم وقوع ازدحام را تضمین نمی‌کنند، بلکه پس از وقوع، سعی در کنترل آن را دارند تا حدی که به بازدهی سیستم آسیبی چندانی وارد نشود [۲۵].

کیفیت سرویس در شبکه‌ها از طریق سه مکانیزم پیاده‌سازی می‌شود. این مکانیزم‌ها عبارتند از:

- مکانیزم انتها به انتها
- مکانیزم لبه‌ای
- مکانیزم میانی

۲-۴-۲-۱- مکانیزم انتها به انتها

این مکانیزم‌ها در هر دو انتهای یک ارتباط عمل می‌کنند و قادرند رفتار یک ارتباط مشخص را کنترل یا تطبیق نمایند. بعلاوه این مکانیزم‌ها قادرند کیفیت یک ارتباط را کنترل کنند مثالی از

این مکانیزم‌ها، مکانیزم‌های کنترل ازدحام TCP هستند. این مکانیزم‌ها نرخ ترافیک جریان را در واکنش به دور ریخته شدن بسته‌ها کاهش می‌دهند [۲].

۲-۴-۲-۲ مکانیزم لبه‌ای

این مکانیزم‌ها بر روی واسط کاربر- شبکه عمل کرده و به دو دسته نظارت^۱ و شکل‌دهی^۲ تقسیم می‌شوند. این دو دسته مشابه هم هستند با این تفاوت که مکانیزم‌های شکل‌دهی، در سمت شبکه و مکانیزم‌های نظارت، در سمت کاربر عمل می‌کنند.

مکانیزم‌های شکل‌دهی، نرخ تولید ترافیک کاربر را با نرخ‌ی که قبلاً بین کاربر و شبکه توافق شده بود منطبق می‌کنند. اگر منبعی، ترافیکی با نرخ‌ی بیش از آنچه در قراردادش مشخص شده ارسال نماید، شکل دهنده، ترافیک وارده را ذخیره کرده و آن را بگونه‌ای به شبکه می‌فرستد که از مقادیر مشخص شده در قرارداد ترافیکی پیروی کند. در این مکانیزم، بسته‌های متخلف، دور ریخته نمی‌شوند. مکانیزم‌های نظارت، که در سمت کاربر اجرا می‌شوند چک می‌کنند آیا ترافیکی که تولید شده توسط کاربر با قرارداد آن با شبکه، تطابق دارد یا نه. در صورت عدم تطابق، مکانیزم به نحوی عمل می‌کند که ترافیک را با قرارداد منطبق سازد. در صورتی که منبع ترافیک از نرخ ترافیک توافق شده تجاوز کند، دو راهکار توسط این مکانیزم‌ها اجرا می‌شود.

- دور ریختن بسته، که در آن بسته‌های متخلف دور ریخته می‌شوند.
- علامت زدن بسته، که در آن بسته‌های متخلف توسط شبکه علامت زده می‌شوند تا در صورت

بروز ازدحام، اولین بسته‌هایی باشند که دور ریخته می‌شوند [۲].

۲-۴-۲-۳ مکانیزم میانی

این مکانیزم‌ها در نودهای میانی شبکه، مانند مسیریاب‌ها و کلیدها اجرا می‌شوند و به دسته‌های زیرطبقه‌بندی می‌گردند:

¹ Policing
² Shaping

- بافرکردن : وقتی که ترافیک ورودی به یک مسیریاب، بیشتر از ظرفیت اتصال خروجی باشد، بسته‌ها به طور موقت، در محلی به نام بافر ذخیره می‌گردند.
- مدیریت صف : مکانیزم‌های مدیریت صف، مکانیزم‌هایی هستند که در صورت سرریز بافر، بسته‌ای را که باید دور ریخته شود انتخاب می‌کنند. هر صف، یک مکانیزم مدیریت صف مربوط به خود دارد.
- زمانبندی: زمانبندی، بسته بعدی را از میان بسته‌هایی که در بافرها قرار دارند انتخاب و به درگاه خروجی می‌فرستد. زمانبندی فقط باید بسته‌ای که بیشترین زمان انتظار را دارد انتخاب کند [۲].

۲-۴-۳ - دسته بندی دیگر از مکانیزم‌های کنترل ازدحام

در کنترل ازدحام شبکه اینترنت، مطابق با مکانی که الگوریتم کنترل ازدحام روی شبکه اجرا می‌شود، به دو دسته تقسیم می‌شوند:

- الگوریتم‌های سمت منابع^۱
- الگوریتم‌های سمت مسیریاب‌ها^۲

۲-۴-۳-۱ - الگوریتم‌های سمت منابع

الگوریتم‌های کنترل ازدحام که در پروتکل TCP استفاده می‌شود به طور گسترده شامل الگوریتم‌های سمت منابع است که تحت عنوان کنترل ازدحام TCP شناخته شده‌اند. تمامی این الگوریتم‌ها در سمت منابع به وسیله گم شدن بسته‌ها متوجه ترافیک می‌شوند.

به عبارتی، این پروتکل بر مبنای پنجره ازدحام عمل نموده و با افزایش و کاهش این پنجره حجم داده ارسالی خود را کنترل می‌کند. در این پروتکل فرستنده پس از دریافت اعلام وصول^۳ یا Ack هر بسته ارسالی به افزایش یا کاهش و یا ثابت نگاه داشتن حجم ارسال داده می‌پردازد و در صورت عدم دریافت

¹ Source Algorithm

² Link Algorithm

³ Acknowledgement

Ack در یک بازه زمانی معین و یا دریافت سه Ack تکراری، این امر را نشانه‌ی گم‌شدن بسته و در نتیجه وجود ازدحام در شبکه تلقی نموده و در مواجهه با این وضعیت TCP با کاهش اندازه پنجره ازدحام خود، حجم داده‌های ارسالی خود را کاهش می‌دهد. البته در شروع یک ارتباط در شبکه، سایز پنجره مناسب تعیین می‌شود. گیرنده بر اساس سایز بافر خود می‌تواند سایز پنجره را تعیین کند. اگر میزان داده‌های ارسالی از فرستنده در حد سایز پنجره باشد، مشکلی پیش نمی‌آید. در غیر این صورت است که در سمت گیرنده سرریز^۱ اتفاق می‌افتد [۲۵].

کنترل ازدحام در شبکه TCP از بالا رفتن ظرفیت شبکه جلوگیری می‌کند. در واقع به فرستنده اجازه می‌دهد نرخ ارسال داده در شبکه را به منظور جلوگیری از ازدحام تنظیم نماید. مکانیزم‌های کنترل ازدحام که توسط TCP حمایت می‌شوند عمدتاً شامل ۴ مرحله اصلی می‌باشند که در ادامه هر کدام به اختصار توضیح داده می‌شود.

- شروع آهسته^۲
- پرهیز از ازدحام^۳
- ارسال مجدد سریع^۴
- بازیابی سریع^۵

شروع آهسته ۱-۱-۳-۴-۲

در شروع یک ارتباط، ارسال داده‌ها به سمت گیرنده به اندازه حداکثر ظرفیت سمت گیرنده انجام نمی‌شود. بلکه فرستنده تعدادی بسته در شروع، ارسال می‌کند و منتظر دریافت Ack بسته‌های ارسالی می‌شود و سپس بتدریج نرخ ارسال را افزایش می‌دهد. این مکانیزم به فرستنده TCP این اجازه را می‌دهد که وضعیت موجود شبکه اعم از پهنای باند در دسترس را شناسایی کند. در شروع یک

^۱ Overflow

^۲ Slow start

^۳ Congestion avoidance

^۴ Fast Retransmission

^۵ Fast Recovery

ارتباط سایز این پنجره یک سگمنت است. فرستنده TCP با ارسال یک سگمنت وارد فاز شروع آهسته شده و منتظر دریافت Ack می‌شود. بعد از دریافت Ack فرستنده سایز پنجره ارسال را از یک به دو افزایش می‌دهد و دو سگمنت فرستاده می‌شود. زمانی که Ack این دو سگمنت از سمت گیرنده دریافت شد، فرستنده سایز پنجره ارسال را از ۲ به ۴ افزایش می‌دهد و ۴ سگمنت فرستاده می‌شود. رشد نمایی سایز پنجره ارسال ادامه می‌یابد تا زمانی که به حداکثر سایز پنجره سمت گیرنده برسد یا بسته‌ها بدلیل ازدحام در شبکه اتلاف شوند.

فرستنده TCP به دو طریق می‌تواند اتلاف بسته‌ها در شبکه را تشخیص دهد.

- دریافت Ack تکراری.

- سر ریز شدن زمان سنج.

عدم وجود یک سگمنت در بین سگمنت‌های ارسالی در یک پنجره باعث ایجاد Ack تکراری در مقصد می‌شود. فرستنده TCP انتظار دریافت Ack از سمت مقصد به نشانه دریافت موفق بایت‌های جدید در رشته داده‌ها را دارد. هر زمانی که فرستنده یک سگمنت را ارسال می‌کند یک زمان سنج شروع بکار می‌کند و منتظر دریافت Ack می‌شود. اگر زمان سنج قبل از اینکه Ack داده‌های ارسالی دریافت گردد، سرریز شود فرستنده TCP فرض را بر این می‌گذارد که سگمنت گم شده یا آسیب دیده است. در نتیجه مجدداً آن را ارسال می‌کند [۲۶].

۲-۴-۱-۳-۲ - اجتناب از ازدحام

روشی که در قرارداد TCP برای اجتناب از وقوع ازدحام به کار گرفته می‌شود مبتنی بر پنجره است و بر پایه دو تکنیک ابتدایی افزایش تدریجی و کاهش ناگهانی استوار است [۲۵].

در سمت فرستنده اندازه پنجره ازدحام نگهداری می‌شود، که تعیین کننده حداکثر تعداد بسته‌هایی است که یک فرستنده می‌تواند بفرستد بدون آنکه تصدیق آنها را دریافت کرده باشد. هنگامی که مبدأ یک جریان در اثر دور انداخته شدن یکی از بسته‌هایش یا تأخیر زیاد، از وقوع ازدحام اطلاع یافت، نرخ ارسال خود را به نصف تقلیل می‌دهد (اندازه پنجره ازدحام به نصف کاهش می‌یابد)، به این ترتیب در

هنگام وقوع ازدحام نرخ ارسال به طور نمایی کاهش می‌یابد (کاهش ناگهانی). اما هنگام آغاز جریان بر روی یک اتصال تازه، یا پس از رفع ازدحام قبلی، فرستنده با دریافت هر تصدیق یک واحد اندازه پنجره خود را افزایش می‌دهد و به این ترتیب نرخ ارسال به طور خطی افزایش می‌یابد (افزایش تدریجی) [۲۹].

۲-۴-۳-۱-۳- ارسال مجدد سریع

همانطور که قبلاً بیان شد، TCP با دریافت Ack تکراری متوجه اتلاف بسته‌ها می‌شود. از طرفی دریافت Ack تکراری می‌تواند به مفهوم دریافت بسته‌ها بر خلاف ترتیب اصلی نیز باشد. پس بجای نشان عکس‌العمل سریع نسبت به دریافت Ack تکراری با ارسال مجدد بسته گم شده، فرستنده TCP منتظر می‌شود تا ۳ بار Ack تکراری دریافت کند. این روش کارایی TCP را به ۳ طریق افزایش می‌دهد:

- اگر بسته‌ها به ترتیب دریافت نشده باشند از ارسال مجدد آنها و در نتیجه پر کردن ظرفیت شبکه جلوگیری می‌کند.
- بهره‌وری بیشتر و بازدهی بالا خطوط ارتباطی را موجب می‌شود.
- در صورت اتلاف بسته‌ها، TCP قبل از سرریز زمان سنج اقدام به ارسال مجدد بسته می‌کند و منتظر سرریز شدن زمان سنج نمی‌شود [۲۶].

۲-۴-۳-۱-۴- بازیابی سریع

زمانیکه مبدا TCP، Ack تکراری دریافت می‌کند، داده‌ها به سمت مقصد در جریان هستند زیرا در مقصد اگر سگمنت‌ها به ترتیب دریافت نشوند Ack تکراری تولید می‌شود. در این مورد فرستنده TCP به طور ناگهانی نرخ ارسال داده را با رفتن به فاز شروع آهسته کاهش نمی‌دهد. در عوض بعد از ارسال مجدد سگمنت گم شده (در پاسخ به ۳ Ack تکراری) مبدا TCP میزان پنجره ارسال را به نصف کاهش می‌دهد و وارد فاز اجتناب از ازدحام می‌شود. این عملیات باعث بازدهی بهتر TCP می‌شود.

شود. بازیابی سریع از خالی شدن جریان TCP بین مبدا و مقصد بعد از اتلاف یک بسته و ارسال مجدد آن جلوگیری می‌کند [۲۶].

۲-۴-۳-۱- الگوریتم سمت مسیریاب‌ها

در این میان الگوریتم‌های سمت مسیریاب‌ها، همان طور که از نامش پیداست، روی تجهیزات شبکه مانند مسیریاب‌ها و کلیدها عمل می‌کنند که نقش آنها تشخیص وقوع ازدحام در شبکه و فیدبک گرفتن از این اطلاعات است. به طور کلی الگوریتم‌های سمت مسیریاب‌ها یک استراتژی کنترلی برای تشخیص ازدحام بر پایه ارتباطات شبکه هستند که تحت عنوان کنترل ازدحام IP نیز شناخته می‌شوند. در حال حاضر تحقیقات روی الگوریتم مسیریاب‌ها، بر روی جوانب الگوریتم‌های مدیریت فعال صف متمرکز شده است [۲۶].

الگوریتم‌های در سمت مسیریاب‌ها بر اساس طول صف ترافیک را می‌سنجند. ارسال این بازخوردها به مبادی جریان‌ها، به صورت صریح یا ضمنی و به صورت گم شدن بسته‌ها، تاخیر و یا نشان‌گذاری بسته‌ها صورت می‌گیرد. الگوریتم‌های مدیریت فعال صف باید میزان ازدحام موجود در مسیریاب را اندازه‌گیری کرده و به مبادی جریان ارسال نماید. نحوه محاسبه میزان ازدحام و چگونگی ارسال آن به مبادی جریان از مختصات یک مکانیزم AQM می‌باشد [۲۹]. در فرایند طراحی مکانیزم AQM لازم است، که اهداف زیر مدنظر قرار گیرند:

- کاهش تعداد بسته‌های دور انداخته شده از مسیریاب: اگر تعداد بسته‌های گم شده در شبکه افزایش یابد، ضمن کاهش گذردهی شبکه نوعاً باعث ارسال مجدد بسته‌ها شده و منابع شبکه را تلف می‌کند.
- افزایش میزان بهره‌وری اتصالات: اگر بهره‌وری یک اتصال پایین باشد، بدین معنی است که الگوریتم حاکم بر شبکه، منابع شبکه را تلف کرده و ظرفیت بهره‌برداری از کل منابع موجود را ندارد.

- کاهش متوسط طول صف در مسیریاب‌های شبکه: کم بودن طول صف باعث کاهش تأخیر بسته‌های شبکه شده و احتمال گم‌شدن بسته‌ها را کاهش می‌دهد.
- یکسان‌سازی در تخصیص منابع شبکه: یکسان‌سازی بدین معنی است که پهنای باند اتصال گلوگاه به صورت مساوی بین جریان‌های گذرنده از آن تقسیم شود.

اساسی‌ترین هدف هر الگوریتم کنترل ازدحام، ممانعت از بروز ازدحام و به خصوص پدیده اضمحلال ناشی از ازدحام^۱ می‌باشد [۳]. بنابراین مهمترین معیارهای مطرح در طراحی مکانیزم‌های کنترل ازدحام، "کاهش نرخ گم شدن بسته‌ها"، "افزایش گذردهی"، "افزایش بهره‌وری اتصالات" و "کاهش طول صف مسیریاب" می‌باشد، که در TCP پیاده‌سازی شده و به صورت پویا نرخ ارسال را متناسب با شدت ازدحام مسیر تنظیم می‌کند [۶].

بر روی یک پیوند^۲ هنگامی ازدحام رخ می‌دهد که ترافیک رسیده، از بیشینه ظرفیت آن تجاوز کند. هنگامی که در یک مسیریاب ازدحام رخ دهد، طول صف‌ها افزایش می‌یابد و پردازش آنها طولانی‌تر می‌شود. در چنین حالتی ممکن است مسیریاب به منظور مقابله با ازدحام اقدام به دور انداختن بسته از درون صف کند. به هر حال اثر وقوع ازدحام در نقاط انتهایی به صورت افزایش تأخیر ظاهر می‌شود و از آنجایی که اکثر قراردادهای حمل در چنین شرایطی از مهلت زمانی و ارسال مجدد استفاده می‌کنند، واکنش آنها به وقوع ازدحام، ارسال مجدد بسته‌ها خواهد بود. ارسال مجدد به جای سبک‌تر کردن ترافیک، که در هنگام وقوع ازدحام ضروری است، آنرا افزایش می‌دهد و در نتیجه باعث بدتر شدن اوضاع می‌گردد. به این ترتیب این چرخه ازدحام و افزایش ترافیک ادامه می‌یابد تا جایی که شبکه به صورت غیرقابل مصرف در می‌آید که به آن اضمحلال ناشی از ازدحام گفته می‌شود. با مسأله ازدحام یا از طریق روش‌ها و الگوریتم‌هایی برای اجتناب از ازدحام در فرستنده مانند روش مورد

¹ Congestion Collapse

² Link

استفاده در TCP و یا با به کار بردن تکنیک‌هایی به منظور مدیریت و اختصاص ظرفیت اتصال در مسیریاب‌ها، یعنی روش‌های مدیریت صف مقابله می‌شود [۲۶].

تاکنون روش‌های متفاوتی برای مقابله با مسأله ازدحام به کار گرفته شده است. تمرکز یک مجموعه از این راهکارها بر روی کشف وقوع ازدحام در شبکه به وسیله نقاط انتهایی (در لایه کاربرد یا در لایه انتقال) و تنظیم نرخ انتقال به منظور ایجاد تعادل بین بار تولید شده و ظرفیت موجود است. از طرف دیگر پیشنهادهایی هم مبنی بر استفاده از روش‌های کنترل ورودی^۱ و اختصاص منابع به منظور جلوگیری از وقوع ازدحام یا کاهش آن داده شده است. بهترین مکان برای تشخیص و کنترل ازدحام مسیریاب است [۲۸]. در این قسمت الگوریتم‌های ساده و پویای مدیریت صف که به منظور مقابله با ازدحام در مسیریاب‌ها ارائه شده‌اند را مرور می‌کنیم.

۲-۴-۳-۱- مدیریت ایستای صف

در روش‌های مدیریت ایستای صف هیچ‌گونه پیش‌بینی برای وقوع ازدحام در آینده نمی‌شود. بافرها دارای طول ثابت و ظرفیت محدودی هستند و تا هنگامی که پر نشده‌اند همه بسته‌های ورودی پذیرفته می‌شوند و به محض پر شدن، همه ورودی‌ها دور انداخته می‌شوند تا وقتی که ظرفیت خالی ایجاد شود. الگوریتم‌های مختلفی در این حوزه ارائه شده‌اند که به نحوی سعی در عادلانه‌تر کردن مدیریت صف و یا سعی در کاهش پیچیدگی و افزایش کارایی دارند.

الگوریتم FIFO^۲ ابتدایی‌ترین الگوریتمی است که به منظور مدیریت صف در مسیریاب‌ها بکار می‌رود. در این روش بسته‌های رسیده به انتهای صف اضافه می‌شوند و هنگامی که ظرفیتی برای ارسال وجود داشته باشد، بسته‌ها از ابتدای صف استخراج می‌شوند. هنگامی که صف پر می‌شود، بسته‌هایی که می‌رسند دور انداخته می‌شوند. یکی از مهم‌ترین اشکالاتی که به روش صف‌بندی FIFO وارد می‌شود این است که این روش بین بسته‌هایی که از مبادی مختلف رسیده تمایزی قایل نمی‌شود، در نتیجه

^۱ Admission Control

^۲ First-in, First-out

ممکن است برخی از آن مبادی سهم زیادی از پهنای باند یک مسیریاب را اشغال کنند. در الگوریتم^۱ FQ سعی شده این اشکال رفع شود.

به این ترتیب که برای هر جریان عبورکننده از مسیریاب یک صف جداگانه در نظر گرفته می‌شود. سپس به این صف‌ها به طریق چرخشی سرویس داده می‌شود. به این ترتیب هیچ جریانی نمی‌تواند بیش از سهم خود از پهنای باند استفاده کند. در شرایط وقوع ازدحام نیز فقط بسته‌های مربوط به جریان‌هایی دور انداخته می‌شوند که در واقع بوجود آورنده ازدحام بوده‌اند، و مانع اثر سوء آنها بر روی سایر جریان‌ها می‌شود [۲۶].

اگرچه الگوریتم FQ مزایای خوبی برای حل مسأله کنترل ازدحام دارد، اما پیچیدگی پیاده‌سازی آن مانع از بکارگیری آن در شبکه‌های با سرعت بالا می‌شود.

۲-۴-۳-۲- مدیریت فعال صف^۲

مدیریت پویای صف به معنی مدیریت صف به منظور اجتناب از پر شدن و حبس^۳ آن است. اکثر روش‌های مدیریت پویای صف طی مراحل و با استفاده از رفتار اخیر صف، ازدحام را کشف می‌کنند. این روش‌ها، به منظور اعلام وقوع ازدحام به فرستنده و کمک به کوچک نگه‌داشتن متوسط طول صف، بسته‌ها را سریعاً و پیش از این که صف سرریز کند، دور می‌ریزند. روش‌های مدیریت پویای صف مزایای زیر را برای فرستنده و جریان‌های آنها فراهم می‌کند:

- باعث کاهش تعداد بسته‌های دور انداخته شده در مسیریاب می‌شود:

محدود کردن تعداد دور انداختن‌ها مسأله مهمی است که به روش‌های گوناگونی مورد توجه قرار گرفته است. روش‌های مدیریت پویای صف همان‌گونه که متوسط اندازه صف را کوچک نگه می‌دارند، می‌توانند به ترافیک‌های منقطع^۴ هم، بدون اینکه بسته‌هایشان را دور بریزند، اجازه قرار گرفتن در صف را بدهند. ترافیک منقطع به ترافیک‌هایی گفته می‌شود که در آنها فاصله بین دفعات ارسال

^۱ Fair Queuing

^۲ Active Queue Managment

^۳ Lock-out

^۴ Bursty Traffic

بسته توسط فرستنده زیاد است، مثلاً هنگامی که فاصله فرستنده از گیرنده طولانی است، پس از فرستادن یک بسته فرستنده باید مدت زیادی منتظر بماند تا تصدیق آن را دریافت کند و سپس بسته‌های بعدی را ارسال کند [۲۹].

در بیشتر تکنیک‌های AQM یک جریان بخصوص، نسبت به حالت عادی، با تعداد کمتری دور انداخته شدن بسته مواجه می‌شود.

- تأخیر صف بندی کمتری ایجاد می‌شود:

متوسط طول صف در روش‌های AQM نسبت به روش FIFO به طور قابل ملاحظه‌ای کوچکتر است. در نتیجه، تأخیر ناشی از قراردادن بسته‌ها در صف کوچکتر خواهد بود و این مسأله برای برنامه‌های تعاملی مهم است.

- اجتناب از حبس

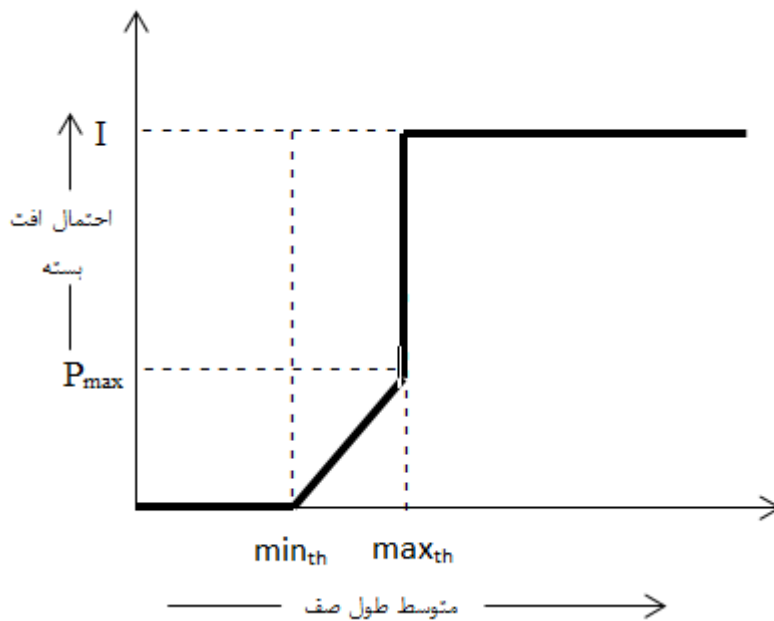
هنگام استفاده از روش‌های مدیریت پویای صف، چون صف‌ها به طور معمول پر نیستند، بیشتر بسته‌های رسیده در صف قرار داده می‌شوند. بنابراین احتمال اینکه مکرراً همه یا اکثر بسته‌های یک جریان خاص دور انداخته شوند کم است. در شرایط وقوع ازدحام، به جای اینکه تعداد کمی از جریان‌ها با دور انداخته شدن تعداد زیادی از بسته‌هایشان مواجه شوند، جریان‌های زیادی با دور انداخته شدن تعداد کمی از بسته‌هایشان مواجه خواهند شد، که در نتیجه باعث خواهد شد که تمام جریان‌های مرتبط بار خود را تنظیم کنند.

در میان روش‌های مدیریت پویای صف، الگوریتم RED از همه مشهورتر است. الگوریتم‌هایی که پس از RED ارائه شده‌اند عمدتاً ضعف‌های RED را جبران نموده‌اند. تمامی آنها بر پایه یک سری روش‌های آماری استوار بوده و هیچ کدام مدل ترافیک را بطور مستقیم در نظر نگرفته و در نتیجه جواب بهینه را ارائه نداده و در برخی موارد به ناپایداری شبکه نیز منجر می‌شوند.

مکانیزم کنترل ترافیک RED به این صورت است که متوسط طول صف را برای هر صف خروجی محاسبه می‌کند و در هنگام وقوع ازدحام، بطور تصادفی یک اتصال را برای خیردهی ازدحام انتخاب

می‌کند، به گونه‌ای که احتمال انتخاب شدن اتصالی که سهم بیشتری از پهنای باند را اشغال کرده، بیشتر باشد. هدف اصلی روش RED فراهم کردن مکانیزمی برای اجتناب از ازدحام به وسیله کنترل متوسط طول صف است.

شکل ۲-۴ نموداری از چگونگی عملکرد RED را به شکل بسیار ساده نشان می‌دهد. در الگوریتم RED با رسیدن هر بسته، متوسط طول صف محاسبه و با دو آستانه کمینه و بیشینه مقایسه خواهد شد. هنگامی که متوسط طول صف از یک آستانه کمینه کوچکتر باشد، هیچ بسته‌ای علامت نمی‌خورد. هنگامی که متوسط طول صف از یک آستانه بیشینه بزرگتر باشد، تمام بسته‌های ورودی علامت می‌خورند. هنگامی که متوسط طول صف بین آستانه‌های کمینه و بیشینه باشد، هر بسته دریافتی با احتمال P علامت می‌خورد.



شکل ۲-۴: چگونگی کارکرد الگوریتم کنترلی RED

در [۲۹] نشان داده شده است که بکار بردن RED در برابر ترافیک‌هایی که به ازدحام واکنش مناسب نشان نمی‌دهند یا آنهایی که نسبت به دور انداخته شدن بسته‌هایشان شکننده و بسیار حساس هستند، چندان مناسب نیست. نشان داده می‌شود وقتی ترکیبی از این نوع ترافیک‌ها از یک اتصال استفاده کنند، ممکن است الگوریتم RED باعث تسهیم ناعادلانه پهنای باند شود.

از اهدافی که در AQM دنبال می‌شود کاهش تاخیر در انتقال داده‌ها و افزایش بهره‌برداری از پهنای باند موجود است و این در حالی است که این دو هدف باهم در تضاد هستند. کاهش تاخیر موجب کاهش بهره‌برداری از پهنای باند و افزایش بهره‌برداری از پهنای باند موجب افزایش تاخیر می‌شود، لذا باید بین آنها مصالحه‌ای برقرار کرد.

یکی از عواملی که بر روی این دو هدف تاثیر دارد انتخاب مقدار مرجع برای طول بافر صف، Q_{ref} است. که اگر بزرگ انتخاب شود بهره‌برداری از پهنای باند بیشتر و تاخیر نیز بیشتر خواهد بود و برعکس. حال اگر فرض کنیم Q_{ref} برای یک عملکرد بهینه انتخاب شده است، عملکرد کنترلی سیستم در پاسخ به این مرجع می‌تواند به عنوان معیار کارایی عملکرد الگوریتم AQM قرار بگیرد. از این رو هرچه پاسخ سیستم (اندازه بافر گلوگاه) به مقدار مرجع Q_{ref} سریعتر، با فراجاهش کمتر، خطای حالت پایایی کمتر و نوسانات کمتر همگرا شود، عملکرد کنترلی مطلوبتر است.

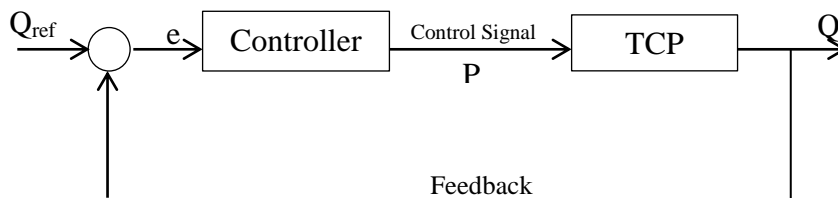
با توجه به مطالب گذشته، ایده AQM قرار دادن قوانین منظمی در مسیرباف برای مدیریت کردن صف است، البته این روش به تنظیم پارامترهای دینامیکی شبکه حساس است [۳۰]. بعد از این بر پایه کار روی [۳۱] برای دستیابی به پاسخ بهتر و عملکرد کنترلی توسط محاسبه احتمال افت بسته با استفاده از طول صف آنی به جای متوسط طول صف مانند RED، یک کنترلر PI طراحی شد [۳۰].

نتایج شبیه‌سازی‌ها نشان داد که عملکرد کنترل کننده PI در تنظیم طول صف در مدل فضای حالت برای رسیدن به یک مقدار مطلوب در سطح مختلف ازدحام نسبت به RED عملکرد بهتری دارد. اگرچه در مواردی که بافر کوچک باشد یا مسیرباف متحمل بار سنگینی باشد، دوره گذار از این حالت قبل از اینکه صف به مقدار هدف همگرا شود، به اندازه قابل ملاحظه‌ای زیاد است. در نتیجه کنترل کننده PI و مدل‌های اصلاح شده آن برای مدیریت فعال صف و کنترل ازدحام در شبکه‌های پیچیده مناسب نیستند. علاوه بر این در هر حال کنترل کننده فوق برای یک منطقه عملکردی خاص طراحی شده است و این در حالی است که پارامترهای شبکه دائما در حال تغییر می‌باشند.

همان طور که می‌دانیم کنترل‌کننده‌های PI روی اصلاح خطاهای گذشته مانور می‌دهند و برای خطاهای آینده پیش‌بینی‌ای ندارند این نقطه ضعف، راه را برای ورود یک کنترل‌کننده مشتقی باز می‌کند.

۵-۲- ساختار کنترلی

پروسه TCP با شماتیک AQM، مطابق با استراتژی افت بسته، به عنوان یک سیستم حلقه بسته کنترلی به شکل ۵-۲ است.



شکل ۵-۲: مدل پروسه کنترلی TCP

در شکل ۵-۲، Q طول صف و Q_{ref} اندازه صف مرجع است. خروجی کنترل‌کننده P احتمال افت بسته است که تصمیم می‌گیرد بسته ورودی را بپذیرد یا بیاندازد. هدف کنترل نگهداری سایز Q در سطح هدف یعنی Q_{ref} می‌باشد.

۶-۲- مدل دینامیکی TCP/IP

برای کنترل به روش‌های ریاضی و کلاسیک در ابتدای امر به مدل دینامیکی سیستم نیازمندیم. در مطالعه‌ی در دست انجام از یک مدل دینامیکی از رفتار TCP بر اساس آنالیز معادله تفاضلی اتفاقی^۱ و جریان سیال^۲ استفاده شده است. نتایج شبیه‌سازی‌های انجام گرفته در [۱۶] دقت مدل را نشان داده است. این مدل مقادیر میانگین متغیرهای کلیدی شبکه را در نظر گرفته و بر اساس مفروضات مستدل، مدل زیر را ارائه داده است [۳۱].

¹ Stochastic Differential Equation

² Fluid-Flow Model

$$\dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))} p(t-R(t)) \quad (1-2)$$

$$\dot{q} = \frac{W(t)}{R(t)} N(t) - C(t) \quad (2-2)$$

$q(t)$ طول صف در مسیریاب (بر حسب بسته)، $W(t)$ سایز پنجره TCP مدنظر (بر حسب بسته) و $C(t)$ ظرفیت اتصال (بر حسب (pkt/sec)) و $R(t)$ زمان رفت و برگشت هر بسته^۱ (بر حسب ثانیه) می باشد که از دو بخش تاخیر صف ($q(t)/C(t)$) و تاخیر انتشار (T_p) تشکیل شده است. یعنی:

$$R(t) = \frac{q(t)}{C(t)} + T_p \quad (3-2)$$

$\dot{W}(t)$ و $\dot{q}(t)$ به ترتیب انحراف زمان $W(t)$ و $q(t)$ و $N(t)$ ضریب بار TCP را نشان می دهد. طول صف (q) و سایز پنجره $W(t)$ مقادیری مثبت و محدود هستند، یعنی $q \in [0, q_{max}]$ و $W \in [0, W_{max}]$ که q_{max} و W_{max} به ترتیب ظرفیت بافر و بیشینه سایز پنجره را نشان می دهند. همچنین احتمال افت بسته نیز مقداری در بازه $[0, 1]$ را می پذیرد.

مدل فوق، مدل دقیقی از TCP و رفتار صف را بیان می کند [۳۲]، که از دو بخش افزایش و کاهش تشکیل شده است، بخش افزایش نشانگر آن است که به ازای تغییرات زمانی به اندازه یک رفت و برگشت زمانی مقدار پنجره ازدحام به اندازه یک واحد افزایش می یابد. این امر در مرحله جلوگیری از ازدحام پروتکل TCP اتفاق می افتد. در بدست آوردن این مدل فرض می شود که جریان TCP به مرحله جلوگیری از ازدحام رسیده و فقط تغییرات پنجره ازدحام در این مرحله در معادله ظاهر شده است. بخش کاهش بیان می دارد که در صورت بروز تلفات مقدار پنجره ازدحام بایستی نصف گردد. در مجموع، دینامیک تغییرات پنجره به صورت الگوریتم افزایش جمعی و کاهش ضربی^۲ (AIMD) مدل شده است.

^۱ RTT-Round Trip Time

^۲ Additive Increase Multiplicative Decrease

بخش دوم مدل AQM مربوط به نقاط میانی می باشد که براساس دینامیک تغییرات طول صف نوشته شده است.

۲-۶-۱- خطی سازی مدل دینامیکی

ساده ترین راه تحلیل یک سیستم دینامیکی خطی سازی آن می باشد. معادلات (۱-۲) و (۲-۲) معادلات غیرخطی تاخیردار با دو متغیر حالت W و q می باشند. برای خطی سازی فرض می کنیم:

$$N(t) = N; R(t) = R; C(t) = C \quad (۴-۲)$$

که N ، R و C مقادیری ثابت هستند.

برای یافتن نقطه تعادل داریم:

$$\dot{W}(t) = 0 \quad (۵-۲)$$

$$\dot{q}(t) = 0 \quad (۶-۲)$$

با حل این معادلات نقطه تعادل (W_0, q_0, p_0) بدست می آید.

$$W_0^2 p_0 = 2 \quad (۷-۲)$$

$$W_0 = \frac{R_0 C}{N}; R_0 = \frac{q_0}{C} + T_p \quad (۸-۲)$$

معادلات دیفرانسیل فوق در نقطه تعادل منجر به معادلات زیر می شود.

$$\delta \dot{W}(t) = -\frac{2N}{R^2 C} \delta W(t) - \frac{RC^2}{2N^2} \delta p(t-R) \quad (۹-۲)$$

$$\delta \dot{q}(t) = \frac{N}{R} \delta W(t) - \frac{1}{R} \delta q(t) \quad (۱۰-۲)$$

که در آن $\delta W = W - W_0$, $\delta q = q - q_0$, $\delta p = p - p_0$ می باشد.

همه جزئیات خطی سازی در مرجع [۳۰] آمده است. باید توجه داشته باشیم که برای خطی سازی، از طبیعت متغیر با زمان بودن تاخیر زمانی رفت و برگشت بسته، در ترم $t-R(t)$ صرف نظر می شود و با فرض $W \square 1$ به صورت $t-R$ تخمین زده می شود. به عبارت دیگر وابستگی زمان رفت و برگشت بسته به طول صف در پارامترهای دینامیکی را در نظر نمی گیریم.

با انتقال معادلات دیفرانسیل (۲-۹) و (۲-۱۰) به حوزه لاپلاس دینامیک‌های خطی شده می‌تواند به عنوان یک تابع انتقال مرتبه ۲ با تاخیر زمانی مدل شوند:

$$G(s) = \frac{Ke^{-Rs}}{(T_1s+1)(T_2s+1)} \quad (۱۱-۲)$$

که در آن $K = (RC)^3 / 4N^2$, $T_1 = R^2C / 2N$, $T_2 = R$

همان طور که می‌دانیم متغیرهای طول صف در مسیریاب گلوگاه اتصال قرار گرفته‌اند. ورودی کنترل‌کننده خطای بین سائز صف مرجع و سائز صف آنی می‌باشد $(q_{ref} - q)$. در حالی که خروجی کنترل‌کننده، احتمال افت بسته است (P) . احتمال افت بسته تصمیم می‌گیرد که بسته ورودی را قبول کند یا دور بیاندازد تا طول صف در نزدیکی مقدار مرجع بماند [۳۲].

قابل مشاهده است که تابع تبدیل ورودی-خروجی سیستم محاسبه شد. حال با دستیابی به این مدل می‌توان کنترل‌کننده‌های مناسب را برای رفع نواقص این سیستم‌ها طراحی، محاسبه و سپس روی سیستم اعمال کرد. این عملیات در فصول بعد انجام شده است. با توجه به مقادیر پارامترها از مرجع [۴۱] سیستم دارای تابع تبدیل (۲-۱۳) می‌باشد.

$$\begin{aligned} T_2 = R &= 0.246 \\ T_1 &= 1.8911 \\ K &= 54517.58799 \end{aligned} \quad (۱۲-۲)$$

$$G = \frac{Ke^{-Rs}}{(T_1s+1)(T_2s+1)} = \frac{117189.04943e^{-0.246s}}{(s+0.53)(s+4.065)} \quad (۱۳-۲)$$

فصل سوم

روش‌های کنترلی

الگوریتم‌های موجود در حوزه مدیریت فعال صف عموماً بر اساس وضعیت فعلی صف و طول آن میزان ازدحام شبکه را سنجیده و نسبت به ارسال بازخوردهای لازم به مبادی جریان اقدام می‌کنند، تا در گام بعدی نرخ ارسال مبادی جریان بر اساس آن تنظیم گردد. این در حالی است که وضعیت ازدحام شبکه علاوه بر طول صف گلوگاه، به حجم ترافیک آینده نیز وابسته است. بنابراین اگر بتوانیم به شکلی نرخ ترافیک آینده را تخمین بزنیم، می‌توانیم وضعیت ازدحام شبکه را به شکل دقیق‌تر برآورد کرده و بازخوردهای دقیق‌تر و مفیدتری را به مبادی جریان ارسال نمائیم. برخی کنترل‌کننده‌های کلاسیک مبتنی بر مدل ریاضی این قابلیت را برای ما فراهم آورده‌اند.

در ادامه با توجه به مدل دینامیکی که در فصل قبل از سیستم موردنظر ارائه شد، چند روش کنترلی موجود در مهندسی کنترل که برای سیستم‌های تاخیردار مناسب‌ترند برای کنترل این فرایند پیشنهاد می‌شود. سپس محاسبات هر روش را به طور جداگانه انجام داده و روی سیستم پیاده می‌کنیم تا در نهایت امر با استفاده از نتایج حاصل از شبیه‌سازی بهترین کنترل‌کننده مورد مطالعه ارزیابی و انتخاب گردد.

۳-۱- پیش‌بینی‌کننده اسمیث

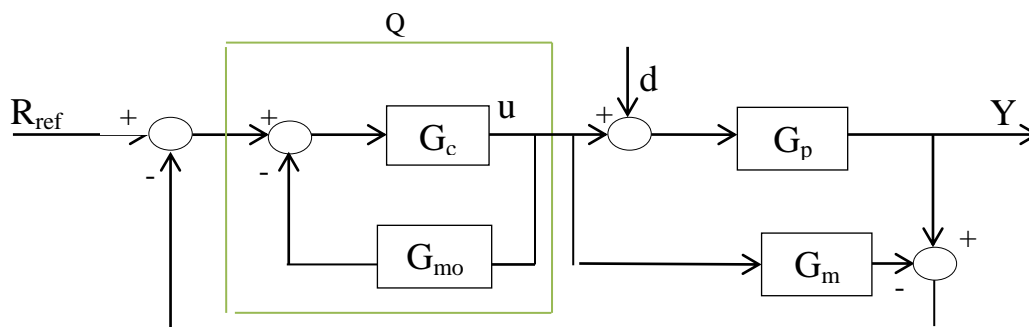
در مقایسه با فرایندهای بدون تاخیر، وجود تاخیر زمانی تحلیل طراحی سیستم کنترل را پیچیده و دستیابی به کنترل رضایت بخش را دشوارتر می‌کند. در حدود سال ۱۹۵۳ کوهن^۱ مشکل طراحی کنترل‌کننده برای سیستم‌های تاخیردار را که وابسته به میزان تاخیر و ثابت زمانی سیستم است مطرح کردند [۱]. زمانی که ثابت زمانی سیستم در مقایسه با تاخیر زمانی کوچک باشد باید بهره حلقه پایین باشد تا باعث ناپایداری نشود [۳۳]. برای حل این مشکل، اسمیث برای تخمین تاخیر زمانی سیستم‌های تک ورودی تک خروجی یک تکنیک ساده و قدرتمند پیشنهاد داد که امروزه پیش‌بینی‌کننده اسمیث نامیده می‌شود که از جذابیت‌های آن می‌توان به کاربردش برای پروسه‌های با

¹ Cohen

تابع تبدیل کسری سیستم‌های نامینیمم فاز اشاره کرد. اگرچه این روش به عدم تطابق مدل حساس بوده و قابلیت رد کردن اغتشاش آن کمی ضعیف است. چگونه فایق آمدن بر این نواقص، موضوع بسیاری از مطالعات می باشد [۳۳-۳۴].

۳-۱-۱- طراحی پیش بینی کننده اسمیث

ساختار کنترلی پیش‌بینی‌کننده اسمیث در شکل ۳-۱ نشان داده شده است. G_c تابع انتقال جبران‌ساز است که کنترل‌کننده اولیه نامیده می‌شود. G_p سیستم واقعی یا تابع انتقال اکیدا سره پایدار سیستم و G_m مدل نامی و G_{mo} مدل نامی بدون تاخیر زمانی می‌باشد. در روش اسمیث یک حلقه داخلی وجود دارد که کنترل‌کننده اصلی در آن تعبیه شده است و اثرات اغتشاش و خطای مدل کردن نیز توسط یک حلقه خارجی تصحیح می‌شود. حلقه کوچک، خروجی دارای تاخیر را برای رسیدن به مقدار پیش‌بینی شده توسط کنترل‌کننده اولیه، حذف می‌کند [۳۵].



شکل ۳-۱: شماتیک فرایند کنترلی با استفاده از تکنیک اسمیث

سیستم تحت بررسی، شبکه جهانی یا اینترنت می‌باشد که در فصل قبل مدل دینامیکی آن بیان شد. این سیستم دارای تابع تبدیل زیر است که البته از تغییرات تاخیر در این مدل صرف نظر شده است و از آنجایی که قطب غالب آن 0.53 می‌باشد و تقریباً 10 برابر از دیگری بزرگتر است می‌توانیم از تقریب مرتبه اول آن استفاده کنیم:

$$G_m = \frac{Ke^{-0.246s}}{(T_1s+1)} = \frac{28828.79445e^{-0.246s}}{(s+0.53)} \quad (1-3)$$

$$G_{mo} = \frac{K_m}{(T_{1m}s+1)} = \frac{28828.79445}{(s+0.53)} \quad (2-3)$$

در ادامه پایداری ساختار اسمیث بیان می‌شود.

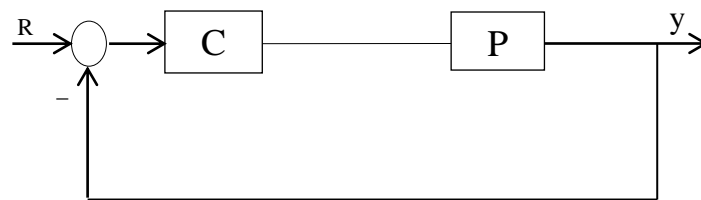
۳-۱-۱-۱- معرفی پارامتر یولا^۱

فرض کنید C مجموعه تمام کنترل‌کننده‌های سرهای است که سیستم حلقه بسته را پایدار داخلی می‌کند. بنابراین داریم:

$$C = \left\{ \frac{Q}{1-pQ}, Q \in S, P \in S \right\} \quad (3-3)$$

که در آن Q پارامتر یولا، P مدل سیستم پایدار اکیدا سره و S مجموعه تمام توابع گویای حقیقی پایدار و سره می‌باشد.

با مراجعه به ساختار کنترلی استاندارد (شکل ۲-۳) در بحث پایداری داخلی داریم:



شکل ۲-۳: ساختار کنترلی استاندارد

$$\frac{1}{1+PC} \begin{bmatrix} 1 & -P & -1 \\ C & 1 & -C \\ PC & P & 1 \end{bmatrix} \xrightarrow{C = \frac{Q}{1-PQ}} \begin{bmatrix} 1-PQ & -P(1-PQ) & -(1-PQ) \\ Q & 1-PQ & -Q \\ PQ & P(1-PQ) & 1-PQ \end{bmatrix} \quad (4-3)$$

¹ Yolla

از آنجا که $P, Q, 1 \in S$ می‌باشند پس PQ نیز متعلق به S است. پس تمامی ۹ تابع تبدیل متعلق به S است، بنابراین پایداری سیستم حلقه بسته تضمین می‌شود.

حال با استفاده از قضیه پارامیترازیسون یولا و با توجه به بلوک دیاگرام ساختار اسمیث داریم :

$$G_c = \frac{Q}{1 - G_{mo}Q} \rightarrow Q = \frac{G_c}{1 + G_{mo}G_c} \quad (5-3)$$

با فرض اینکه $G_p = G_m$ تابع تبدیل خروجی به اغتشاش (d) برابر است با

$$H = 1 - G_m Q \quad (6-3)$$

هدف اصلی هر پروسه کنترلی فیدبکی این است که اختلاف بین مقدار مطلوب با مقدار خروجی در کمترین میزان ممکن نگه داشته شود یا اثر منفی اغتشاش در کمترین میزان ممکن نگه داشته شود. در اینجا هدف عملکردی مینیمم کردن $\|WH\|_2$ می‌باشد که W تابع وزنی است.

یکی از راهکارهای مبارزه با تاخیر $e^{-\theta s}$ استفاده از تقریب آن می‌باشد. با تقریب مرتبه اول، تابع تبدیل سیستم مورد نظر به شکل (۷-۳) می‌شود.

$$G \approx K \frac{(1 - \frac{\theta}{2}s)}{(T_1 s + 1)(1 + \frac{\theta}{2}s)} \quad (7-3)$$

این مدل برای استنتاج کردن کنترل‌کننده مورد استفاده قرار گرفته است. خطایی که با تقریب به وجود می‌آید به نامعینی اضافه می‌شود.

در فرایند کنترلی، کنترل‌کننده برای ورودی پله واحد طراحی می‌شود. از این رو می‌توانیم $W = \frac{1}{s}$ قرار دهیم.

توجه کنید که WH یک قطب روی محور موهومی یعنی $s = 0$ دارد. برای اینکه نرم ۲ این عبارت محدود باشد باید یک قید به فرایند طراحی اضافه شود.

$$\lim_{s \rightarrow 0} H = 0 \quad (8-3)$$

به عبارت دیگر، برای خنثی کردن اثر قطب W باید ضمانت کنیم که H یک صفر در $s=0$ دارد. بنابراین داریم:

$$Q(0) = \frac{1}{G_m(0)} = \frac{1}{K} \quad (9-3)$$

در نتیجه، مجموعه تمام Qs هایی که قید فوق را لحاظ می کنند برابر است با:

$$Q = \frac{1}{K} + sQ_1 \quad (10-3)$$

در این صورت:

$$\|WH\|_2^2 = \left\| \frac{\frac{\theta T_1 s + (\theta + T_1)}{2}}{(T_1 s + 1)(\frac{\theta}{2} s + 1)} - \frac{K(1 - \frac{\theta}{2} s)}{(T_1 s + 1)(\frac{\theta}{2} s + 1)} Q_1(s) \right\|_2^2 \quad (11-3)$$

با تعریف Φ به عنوان خانواده‌ای از توابع پایدار اکیدا سره، Φ_0 زیر مجموعه‌ای از Φ ، Φ_0^\perp توابع تبدیل اکیدا سره تحلیلی که بخش حقیقی آنها غیر مثبت است ($\text{Re } s \leq 0$)، بنابراین $\Phi_0^\perp + \Phi$ شامل تمام توابع تبدیل اکیدا سره بدون قطب روی محور $j\omega$ می‌باشد. علاوه بر این هر تابع F در Φ می‌تواند به شکل زیر بیان شود [34].

$$F = F_1 F_2, \quad F_1 \in \Phi_0, F_2 \in \Phi_0^\perp \quad (12-3)$$

بنابراین، داریم:

$$\|F_1 + F_2\|_2^2 = \|F_1\|_2^2 + \|F_2\|_2^2 + 2 \text{Re} \left[\frac{1}{2\pi j} \oint F_1(-s) F_2(s) ds \right] = \quad (13-3)$$

$$\|F_1\|_2^2 + \|F_2\|_2^2$$

$$\|WH\|_2^2 = \left\| \frac{1 - \frac{\theta}{2}}{1 + \frac{\theta}{2}} \left[\frac{\frac{T_1 \theta}{2} s + (\theta + T_1)}{(T_1 s + 1)(1 - \frac{\theta}{2} s)} - \frac{K}{T_1 s + 1} Q_1 \right] \right\|_2^2 \quad (14-3)$$

با مقادیر مفروض داریم:

$$\|WH\|_2^2 = \left\| \left[\frac{\theta}{1 - \frac{\theta}{2}s} - \frac{T_1}{(T_1s+1)} - \frac{KQ_1}{(T_1s+1)} \right] \right\|_2^2 \quad (15-3)$$

$$\|WH\|_2^2 = \left\| \frac{\theta}{1 - \frac{\theta}{2}s} \right\|_2^2 + \left\| \frac{T_1}{(T_1s+1)} - \frac{KQ_1}{(T_1s+1)} \right\|_2^2 \quad (16-3)$$

حالا کاملا واضح است که Q_1 ناسره بهینه برابر است با

$$Q_{im} = \frac{T_1}{K} \quad (17-3)$$

با محاسبات ابتدایی داریم:

$$Q_{im} = \frac{T_1s+1}{K} \quad (18-3)$$

از آنجا که Q_{im} باید در فرکانس بالا یک Q سره باشد، با انتخاب فیلتر پایین گذر به فرم

$$J = \frac{1}{\lambda s + 1}, \lambda > 0$$

داریم:

$$Q = Q_{im}J = \frac{T_1s+1}{K(\lambda s+1)} \quad (19-3)$$

با جایگزینی (۱۹-۳) در (۵-۳) داریم:

$$G_c = \frac{Q}{1 - G_{mo}Q} = \frac{1}{K} \frac{T_1s+1}{\lambda s} \quad (20-3)$$

۳-۲- کنترل کننده داهلین

برای کنترل پروسه‌های با تاخیر زمانی کوچک، کنترل کننده داهلین یکی از روش‌های کاربردی است [۳۳]. مزیت اصلی آن، سادگی طراحی سیستم است. کنترل کننده داهلین یک تکنیک کنترلی ساده و موثر می‌باشد که به طور گسترده در پروسه‌های کنترلی تاخیردار مورد استفاده قرار می‌گیرد. این کنترل کننده دارای پارامترهای قابل تنظیم است که می‌تواند برای تنظیم پاسخ سیستم مورد کنترل، به شکل همزمان مورد استفاده قرار بگیرد. که این برای کاربردهای عملی، راحت و مناسب است [۳۴].

۳-۲-۱- طراحی کنترل کننده داهلین

ساختار کنترلی شکل ۳-۱ را با فرض اینکه مدل نامی با سیستم واقعی مطابقت ندارد در نظر بگیرید.

$$G_p \neq G_m \quad (۲۱-۳)$$

با قرار دادن تقریب قسمت تاخیردار از مرتبه اول در ساختار اسمیث داریم:

$$G_m = \frac{28828.79445(1 - \frac{\theta}{2}s)}{(s+0.53)(1 + \frac{\theta}{2}s)} = \frac{28828.79445(1 - 0.123s)}{(s+0.53)(1 + 0.123s)} \quad (۲۲-۳)$$

با در نظر گرفتن رابطه (۳-۲۰) به عنوان کنترل کننده، در روش داهلین، وقتی بین مدل نامی و

سیستم واقعی تطابق وجود ندارد مقدار λ تقریباً 0.6 برابر θ در نظر گرفته می‌شود بنابراین داریم:

$$G_c = \frac{1}{K} \frac{T_I s + 1}{\lambda s} = \frac{1}{28828.79445} \frac{1.8911s + 1}{0.1476s} \quad (۲۳-۳)$$

که یک کنترل کننده PI می‌باشد.

۳-۳- کنترل کننده PID

عموما الگوریتم کنترل PID برای کنترل بیشتر پروسه‌های صنعتی کاربرد دارد و به عنوان پایه‌ای برای استراتژی‌های دیگر به شمار می‌رود. کاربرد این الگوریتم کنترلی در رنج وسیعی از فرایندها، آن را به عنوان یک انتخاب خوب برای کنترل دینامیک TCP کاندید می‌کند [۴۱].

۳-۳-۱- معرفی

همان‌طور که در فصل قبل بیان شد هر الگوریتم مدیریت پویای صف نقاط ضعف و قوتی دارند. با نظر به طراحی بر اساس تئوری‌های کنترل، پر واضح است که کنترل کننده تناسبی بسیار سریع و پاسخگو است اما وجود یک خطای حالت ماندگار اجازه نمی‌دهد صف روی مقدار مرجع پایدار شود. از طرف دیگر، طراحی تناسبی-انتگرالی از خطای حالت ماندگار جلوگیری می‌کند (به دلیل وجود بخش انتگرالی)، اما سرعت پاسخ‌گویی را کاهش می‌دهد. با استفاده از تئوری کنترل به خوبی می‌توان دریافت که بخش مشتقی در طراحی یک کنترل کننده PID به کاهش فراجش و زمان نشست، کمک شایانی می‌کند. با حضور بخش مشتقی، با افزایش اثر انتگرالی سیستم سریعتر و فراجش نسبتاً کمتر می‌شود [۴۱].

۳-۳-۲- اهداف طراحی

قبل از تنظیم کنترل کننده PID، اهداف که در ادامه ذکر شده به عنوان اساس کار مطرح می‌شود:

- پاسخگویی سریع یکی از مهمترین اهداف در طراحی است.
 - فراجش کاهش یابد.
 - طراحی منتخب باید در طیف وسیعی از شرایط، از ناپایداری جلوگیری کند.
- با توجه به اینکه اجرای این الگوریتم در سیستم TCP باید در رنج وسیعی از شرایط کاری مسیریاب، مورد استفاده قرار بگیرد، تضمین عملکرد قابل اطمینان شبکه با وجود این کنترل کننده، ضروری است.

۳-۳-۳ - تنظیم پارامترهای کنترل کننده PID

در این قسمت به دنبال یافتن بهترین تنظیم پارامتر، برای کنترل کننده PID معمولی هستیم. یک کنترل کننده PID دارای سه بخش تناسبی، انتگرالی و مشتقی است:

$$G_c(s) = K_p + \frac{K_I}{s} + K_D s \quad (۲۴-۳)$$

به طوری که K_p بهره تناسبی، K_I ثابت انتگرالی و K_D ثابت مشتقی می‌باشد. تنظیم پارامترهای PID به مفهوم یافتن مقادیر K_p ، K_I و K_D ای است که سیستم را پایدار کند. کنترل کننده PID یکی از مهمترین ساختارهای فیدبکی در کاربردهای مختلف می‌باشد. از معروفترین روش‌های تنظیم پارامترها، زیگلر و نیکولز^۱ است [۴۱]. این روش می‌تواند هم برای تنظیم حلقه باز و هم حلقه بسته به کار رود. اما تنظیم به روش دستی هم وجود دارد که بر اساس سلسه مراتب تغییر در اندازه‌ی پارامترها می‌باشد به طوری که در یک زمان یک پارامتر را (اول K_I و سپس K_p و در آخر K_D) افزایش می‌دهند تا سیستم حلقه بسته ناپایدار شود و سپس دامنه آن را کاهش می‌دهند تا به مرز پایداری برسد. در پایان فرایند بسته به اینکه به چه عملکردی از سیستم نیازمندیم، مقادیر پارامترها انتخاب و تنظیم می‌شوند. با استفاده از روش دستی با توجه به مرجع [۴۱] مقادیر پارامترها در بهترین حالت برای چنین سیستمی برابر است با:

$$K_D = 5 \times 10^{-6}, K_p = 4 \times 10^{-5}, K_I = 4 \times 10^{-5} \quad (۲۵-۳)$$

¹ Ziegler and Nichols

۳-۴ - کنترل کننده PID مقاوم

در اکثر موارد نامعینی سیستم‌ها محدود به وجود اغتشاش نیست، بلکه خطای مدل‌سازی غیر قابل اجتناب نیز اتفاق می‌افتد که کنترل این نوع سیستم‌های دارای تاخیر، با روش کنترل کننده تناسبی مشتقی انتگرالی مقاوم^۱ پاسخ بهتری را نتیجه می‌دهد.

اگرچه پیش‌بینی کننده اسمیث روشی اصلاح مناسبی را در عملکرد حلقه بسته پروسه‌های با تاخیر زمانی‌های بزرگ ارائه می‌دهد ولی کاربرد آن در صنعت محدود به چند مسئله است. یکی از مسائل اساسی حساسیت آن به خطای مدل‌سازی است. در مواجهه با عدم تطابق غیرقابل اجتناب بین مدل و سیستم واقعی، عملکردش بسیارضعیف است و منجر به ایجاد ناپایداری می‌شود [۳۹]. به همین دلیل، اخیراً تحقیقات زیادی روی تنظیم مقاوم پیش‌بینی کننده اسمیث در حال انجام است. در این بخش تنظیم کنترل کننده PID مقاوم برای پیش‌بینی کننده اسمیث با وجود نامعینی ارائه شده است.

در متد ارائه شده، با استفاده از مفهوم " بهره معادل بعلاوه تاخیر زمانی"^۲ اثر همزمان نامعینی در تمام پارامترهای مدل می‌توانند داخل شوند. به ویژه برای تنظیم مقاوم برای سیستم‌های مرتبه اول با تاخیر زمانی^۳ و سیستم‌های مرتبه دوم با تاخیر زمانی^۴ مورد استفاده قرار می‌گیرد و مهم‌تر از آن، این است که تقریب بر اساس EGPTD، ساختار تابع انتقال کلی سیستم را عوض نمی‌کند [۴۰].

۳-۴-۱ - مفهوم بهره معادل بعلاوه تاخیر زمانی [۴۰]

در این قسمت به معرفی EGPTD به منظور تحت کنترل گرفتن اثرات نامعینی‌های همزمان در پارامترهای مدل می‌پردازیم.

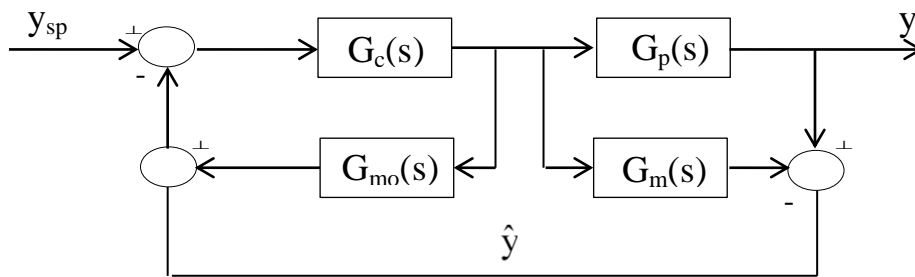
^۱Robust PID Controller

^۲EGPTD : Equivalent Gain Plus Time Delay

^۳FOPTD : First Order Plus Time Delay

^۴SOPTD : Second Order Plus Time Delay

ساختار اسمیث را مطابق شکل ۳-۶ در نظر بگیرید.



شکل ۳-۳: بلوک دیاگرام ساختار اسمیث

تابع تبدیل حلقه بسته سیستم برابر است با:

$$\frac{y(s)}{r(s)} = \frac{G_c(s)G_p(s)}{1 + G_c(s)[G_{mo}(s) + G_p(s) - G_m(s)]} \quad (۲۶-۳)$$

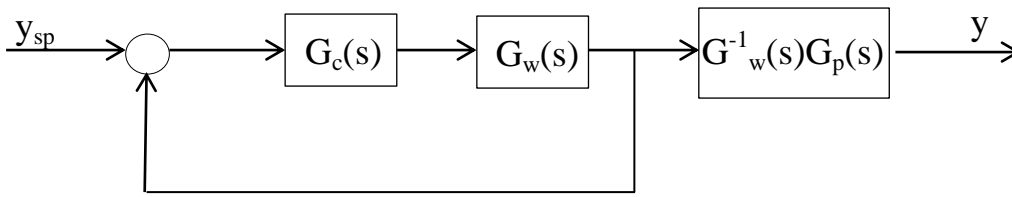
ایده اصلی ساده کردن عبارت پیچیده $G_m^*(s) + G_p(s) - G_m(s)$ در معادله مشخصه است به گونه‌ای که یک قسمت آن بدون عامل تاخیر و قسمت دیگر یک بهره معادل بعلاوه تاخیر زمانی باشد. یعنی:

$$G_m^*(s) + G_p(s) - G_m(s) \approx G_m^*(s) K_{eq} e^{-\theta_{eq}s} = G_w(s) \quad (۲۷-۳)$$

که K_{eq} و θ_{eq} به ترتیب بهره معادل و تاخیر معادل را نشان می‌دهند. دلایل متعددی برای تطبیق EGPTD برای تخمین وجود دارد: (۱) انطباق خوبی با تابع انتقال متناظر در رنج فرکانس میانی و پایین برقرار می‌کند که در فرایند کنترل بسیار حائز اهمیت است. (۲) در فرکانس‌های بالا، حداقل یک تقریب محافظه کارانه می‌دهد که مقاوم بودن را تامین کند.

تخمین تابع انتقال $G_w(s)$ ، برای بدترین مورد در نظر گرفته می‌شود تا پایداری مقاوم برای همه اعضای خانواده فرآیندهای ممکن تضمین شود. با استفاده از $G_w(s)$ ، تابع تبدیل حلقه بسته می‌تواند به صورت (۳۱-۳) در آید.

$$\frac{y(s)}{r(s)} = \frac{G_c(s)G_w(s)}{1 + G_c(s)G_w(s)} G_w^{-1}(s)G_p(s) \quad (۲۸-۳)$$



شکل ۳-۴: بلوک دیاگرام اصلاح شده ساختار اسمیث با استفاده از تکنیک EGPTD

از ویژگی‌های معرفی $G_w(s)$ ، تبدیل مدل پیچیده با عدم تطابق به مدل ساده مناسب است. بنابراین کنترل‌کننده را نیز می‌توان به سادگی تنظیم کرد. همان طور که در شکل ۳-۷ نشان داده شده است $G_p(s)$ به خارج از حلقه می‌رود به طوری که هیچ اثری روی پایداری حلقه بسته نمی‌گذارد.

۳-۴-۲- تقریب با EGPTD [۴۱]

بهره معادل بعلاوه تاخیر زمانی یعنی $K_{eq} e^{-\theta_{eq}s}$ مطابق با یک تابع تبدیل مانند $G(s)$ به راحتی از بسط سری مک لورن قابل حصول است. ابتدا بسط مک لورن $G(s)$ را در نظر می‌گیریم:

$$G(s) = G(0) + \left. \frac{dG(s)}{ds} \right|_{s=0} s + \frac{1}{2} \left. \frac{d^2G(s)}{ds^2} \right|_{s=0} s^2 + \dots \quad (۲۹-۳)$$

توسعه EGPTD، $K_{eq} e^{-\theta_{eq}s}$ را در یک سری مک لورن کاندید می‌کند.

$$K_{eq} e^{-\theta_{eq}s} = K_{eq} - K_{eq} \theta_{eq} s + \frac{1}{2} K_{eq} \theta_{eq}^2 s^2 + \dots \quad (۳۰-۳)$$

K_{eq} و θ_{eq} برای رنج فرکانسی تا جای ممکن مهم، برای تخمین $G(s)$ باید تعیین شوند. با در نظر گرفتن شرایط کنترل فرایند، اولین ترم از (۳۲-۳) بسیار مهم‌تر از ترم بعدی آن است. K_{eq} و θ_{eq} به ترتیب با مقایسه اولین و دومین ترم از (۳۲-۳) و (۳۳-۳) بدست خواهند آمد. بنابراین

$$K_{eq} = G(0) \quad (۳۱-۳)$$

$$\theta_{eq} = - \left. \frac{1}{K_{eq}} \frac{dG(s)}{ds} \right|_{s=0} \quad (۳۲)$$

به عنوان روش دیگر، مقایسه ترم سوم از (۳-۳۲) و (۳-۳۳) نتیجه زیر را برای تاخیر زمانی معادل در پی دارد.

$$\theta_{eq} = \sqrt{\frac{1}{K_{eq}} \left. \frac{d^2 G(s)}{ds^2} \right|_{s=0}} \quad (۳-۳۳)$$

روش محاسبه تاخیر زمانی معادل با در نظر گرفتن یک مصالحه بین عملکرد و میزان مقاوم بودن انتخاب می‌شود. EGPTD بر اساس پایین‌ترین بخش، حاصل عملکرد بهتر ولی میزان مقاوم بودن کمتر در رنج فرکانسی بالا، می‌باشد [۴۱].

۳-۴-۳ کاربرد EGPTD برای سیستم‌های مرتبه یک

برای اطمینان از پایداری، EGPTD بر اساس بدترین حالت ممکن انتخاب می‌شود تا با بزرگترین گین معادل و تاخیر زمانی متناظر شود.

با تقسیم دو طرف معادله (۳-۳۰) بر G_m^* خواهیم داشت:

$$1 + \frac{G_p^*(s)}{G_m^*(s)} e^{-\theta_p s} - e^{-\theta_m s} \approx K_{eq} e^{-\theta_{eq} s} \quad (۳-۳۴)$$

کنید مدل و سیستم مرتبه اول است.

$$G_m(s) = \frac{K_m e^{-\theta_m s}}{\tau_m s + 1} \quad (۳-۳۵)$$

$$G_p(s) = \frac{K_p e^{-\theta_p s}}{\tau_p s + 1} \quad (۳-۳۶)$$

سپس می‌توانیم بدست آوریم:

$$1 + \frac{K_p}{K_m} \frac{\tau_m + 1}{\tau_p + 1} e^{-\theta_p s} - e^{-\theta_m s} \approx K_{eq} e^{-\theta_{eq} s} \quad (۳-۳۷)$$

بنابراین گین و تاخیر زمانی معادل با استفاده از (۳۴-۳) و (۳۵-۳) برابر است با:

$$-۳) K_{eq} = \frac{K_p}{K_m} \quad (۳۸-۳)$$

$$\theta_{eq} = (\tau_p - \tau_m) + \theta_p - \frac{K_m}{K_p} \theta_m \quad (۳۹)$$

وجود نامعینی در پارامترهای حقیقی مدل ارائه شده از سیستم، امری طبیعی است. با توجه به بازه

نامعینی پارامترها که در فصل بعد به آن می‌پردازیم، داریم:

$$\left. \begin{array}{l} \left\{ \begin{array}{l} 5 \leq N \leq 400 \\ 5 \leq R \leq 600 \\ 5 \leq C \leq 600 \end{array} \right. \\ \left\{ \begin{array}{l} K_p = \frac{(RC)^3}{4N^2} \\ T_i = \frac{R^2 C}{2N} \end{array} \right. \end{array} \right\} \begin{array}{l} 1.5 * 10^{-4} < K_p < 72900 \\ 0.05 < \theta_p < 0.6 \\ 1.5 * 10^{-4} < T_{ip} < 2.7 \end{array} \quad (۴۰-۳)$$

از آنجایی که بدترین مورد باید بزرگترین گین و تاخیر زمانی معادل را داشته باشد، EGPTD برای

بدترین مورد باید برابر باشد با:

$$K_{eq} = \frac{K_m + \delta_K}{K_m} \quad (۴۱-۳)$$

$$\theta_{eq} = \frac{\delta_K}{K_m + \delta_K} \theta_m + \delta_\theta + \delta_\tau \quad (۴۲-۳)$$

یک بار که EGPTD برای بدترین حالت محاسبه شود، کنترل‌کننده $G_c(s)$ با هدف پایداری و

ردگیری، بر اساس عبارت متناظر $G_m^*(s) K_{eq} e^{-\theta_{eq}s}$ محاسبه می‌شود. هرگونه روش مرسوم تنظیم

کنترل‌کننده PID، مانند زیگلر نیکولز و ... برای تنظیم کنترل‌کننده، قابل استفاده است [۴۱]. در این

تحقیق از کنترل‌کننده PID مرجع [۳۴] استفاده شده است.

فصل چهارم

شبیه‌سازی و مطالعات عددی در شبیه‌سازی NS

۱-۴ - معرفی NS :

با توجه به مطالب فصل قبل، جهت رسیدن به اهداف طراحی از کنترل‌کننده‌های داهلین و PID معمولی و PID مقاوم در ساختار اسمیث استفاده می‌کنیم که به ترتیب در (۳-۲۳) و (۳-۲۶) و (۳-۲۹) آورده شده‌اند. براساس پارامترهای مورد استفاده در شبیه‌سازی، کنترل‌کننده‌های بدست آمده عبارتند از :

$$(۲-۴) G_{c_Dahlin} = \frac{1}{28828.7944} \frac{1.8911s + 1}{0.1476s} \quad (۱-۴)$$

$$(۳-۴) G_{c_PID} = 4 \times 10^{-5} + \frac{4 \times 10^{-5}}{s} + 5 \times 10^{-6} s$$

$$G_{c_RobustPID} = 1.13167 \times 10^{-4} \left(1 + \frac{1}{1.8911s} + 0.123s \right) \frac{1}{0.0987s + 1}$$

شبیه‌سازی واقع‌گرایانه آزمایش‌های انجام شده، از شبیه‌ساز NS استفاده شده است. NS یک شبیه‌ساز توسعه یافته است که به طور مداوم به‌روز می‌شود [۴۱]. در این مطالعه ورژن ۲/۳۵ مورد استفاده قرار گرفته است. این نرم‌افزار روی سیستم عامل لینوکس^۱ قابل نصب است. تمامی مراحل که برای شبیه‌سازی لازم است در پیوست الف و ب به تفسیر شرح داده شده است.

۲-۴ - زبان برنامه‌نویسی

NS با زبان برنامه‌نویسی C++ نوشته شده است و تمام الگوریتم‌های پیاده‌سازی شده در C++ از طرف OTCL^۲ فراخوانی می‌شود. این زبان برای توضیح توپولوژی شبکه، نوع ارسال اطلاعات، جزئیات مسیریابی و هر گونه اطلاعات مربوط به شبیه‌سازی (زمان‌سنج و پروتکل) می‌باشد.

^۱ linux

^۲ Object-oriented TCL

۳-۴ - گسسته‌سازی

برای پیاده‌سازی کنترل‌کننده در شبیه‌ساز NS باید تابع انتقال کنترل‌کننده از حالت پیوسته به گسسته تبدیل شود.

برای بردن سیستم به حالت گسسته باید فرکانس نمونه‌برداری (f_s) در نظر گرفت. توصیه می‌شود که فرکانس نمونه‌برداری ۱۰ یا ۲۰ برابر پهنای باند انتخاب شود [۴۱]. در این تحقیق پهنای باند ۱۵ هرتز در نظر گرفته شده است بنابراین فرکانس نمونه‌برداری ۱۵۰ انتخاب می‌شود. گسسته‌سازی روی تابع تبدیل کنترل‌کننده انجام می‌گیرد. باید در نظر داشت $\delta q = q - q_{ref}$ به عنوان ورودی و $\delta p = p - p_{ref}$ به عنوان خروجی کنترل‌کننده است و p و q به ترتیب طول صف و احتمال افت بست هستند که P_{ref} می‌تواند روی صفر تنظیم شود (چرا که هدف کنترل‌کننده دستیابی به طول صف پایدار است). بنابراین قرار می‌دهیم:

$$\frac{p}{\delta q} = G_c(s) \quad (۱-۵)$$

۴-۴ - روش‌های تقریب

چندین روش برای بردن یک تبدیل لاپلاس به حوزه Z وجود دارد که از جمله می‌توان به روش " تفاوت محدود"^۱، " اختلاف وارونه"^۲ و "توستین"^۳ اشاره کرد [۴۱]. البته در این مطالعه برای تمام کنترل‌کننده‌ها از روش توستین استفاده شده است. در روش توستین $s = \frac{2}{T_s} \frac{z-1}{z+1}$ در نظر گرفته می‌شود. روابط برای گسسته‌کردن کنترل‌کننده داهلین به عنوان مثال، در پیوست پ آورده شده است.

¹ Finite Difference

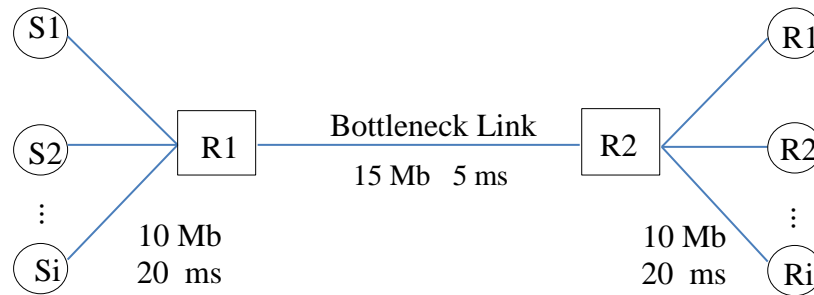
² Backward Difference

³ Tustin

۴-۵- توپولوژی مورد مطالعه

توپولوژی پیاده‌سازی شده در این شبیه‌سازی معروف به توپولوژی دمبل^۱ می‌باشد که در شکل ۴-۱

نشان داده شده است [۳۲].



شکل ۴-۱: توپولوژی مورد مطالعه [۳۲]

برای آنالیز شبیه‌سازی انجام شده در این مطالعه، پنج آزمایش نمونه طراحی شده است. طیف گسترده‌ای از شرایط شبکه، به منظور انجام یک مقایسه خوب بین کنترل‌کننده‌های پیشنهاد شده در این پایان‌نامه نشان داده شده است. در ادامه، شرح توپولوژی و سیستم‌ها ارائه می‌شود.

در تمام آزمایش‌ها، پروتکل پیاده‌سازی شده، TCP می‌باشد. در ادامه فصل، تنظیمات برای هر آزمایش بیان شده و نتایج گرافیکی آنها با هم مقایسه می‌شود.

آزمایش‌ها

در تمامی آزمایش‌ها، حد بالای ظرفیت صف، ۲۰۰ بسته در نظر گرفته شده است و مقدار مرجع طول صف که کنترل‌کننده برای آن طراحی شده ۵۰ بسته می‌باشد. آزمایش اول در شرایط نامی انجام شده است. در آزمایش‌های بعدی رفتار الگوریتم، تحت تغییرات پارامترهای اصلی شبکه یا به عبارتی نامعینی‌های احتمالی شبکه، مطالعه شده است. در تمامی آزمایشات از تابع تبدیل بیان شده در (۲-۱) استفاده شده که در آن از پارامتر N (تعداد جریان) در تفسیر دینامیک سیستم استفاده می‌کند.

¹ Dumb-bell

این پارامتر باید گویای تعداد جریان‌هایی باشد که از مسیریاب عبور می‌کند، اما همه جریان‌ها یک مشخصه را ندارند. در چهار آزمایش اول، جریان‌ها تنها شامل بسته‌های FTP بودند. در ادامه بسته‌های HTTP نیز به جریان‌ها اضافه می‌شوند. ترافیک HTTP شامل دانلودهای کوچک از سرور است. از نقطه نظر مسیریاب، این نوع از ترافیک باعث رفتار متغیر صف می‌شود و بعضی الگوریتم‌های AQM صریحا از آن اجتناب می‌کنند. الگوریتم‌های نظریه کنترل، تعداد بسته‌های موجود در صف را بر اساس تنظیمات مقدار متوسط بر پایه اندازه‌گیری آماری و نه اندازه آنها، در نظر می‌گیرند. این سیاست به تمامی بسته‌ها اجازه می‌دهد به همین طریق پردازش شوند و از اینرو ارسال‌های HTTP هم عرض جریان‌های FTP می‌تواند جریان یابد.

قبل از انجام آزمایش‌ها مفاهیمی چون HTTP¹ و FTP² باید توضیح داده شوند.

در اینترنت همانند سایر شبکه‌های کامپیوتری از پروتکل‌های متعدد و با اهداف مختلف استفاده می‌گردد. هر پروتکل از یک ساختار خاص برای ارسال و دریافت اطلاعات (بسته‌های اطلاعاتی) استفاده می‌نماید و ترافیک مختص به خود را در شبکه ایجاد می‌نماید. HTTP اصطلاحا به پروتکلی گفته می‌شود که برای ایجاد ارتباط، دریافت و ارسال داده‌ها بین سرویس دهنده³ و سرویس گیرنده⁴ استفاده می‌شود. این پروتکل از پروتکل TCP/IP برای بستن پلی میان سرویس دهنده و سرویس گیرنده استفاده می‌کند. طریقه ارتباط آنها با استفاده از پروتکل HTTP به این ترتیب است که داده‌ها، از طریق بسته‌های اطلاعاتی، بین سرویس گیرنده و سرویس دهنده رد و بدل می‌شود. به این ترتیب که برای ارسال داده‌ای به سمت مقصد، در ابتدا داده، به بخش‌های کوچکتری شکسته می‌شود و سپس هر کدام به سمت مقصد و با ترتیب مشخص ارسال می‌شود.

¹ Hyper Text Transfer Protocol

² File Transfer Protocol

³ Server

⁴ Client

FTP نیز یک پروتکل یا قرارداد برای انتقال فایل در شبکه جهانی اینترنت است که مانند HTTP محتوای وب را منتقل می‌کند. با FTP می‌توان فایل‌های موردنظر را از یک رایانه به رایانه دیگر، در یک شبکه بر پایه TCP، منتقل کرد.

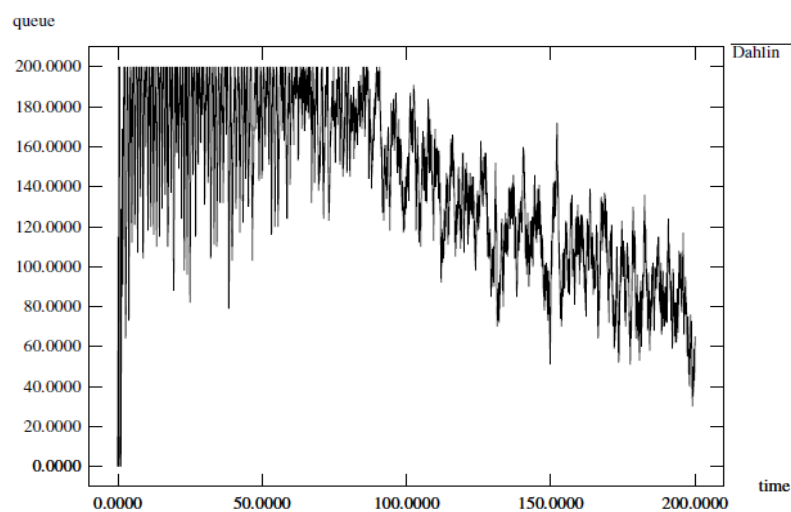
آزمایش ۱:

اولین آزمایش در شرایط نامی (شرایطی که سیستم حول آن خطی شده است) انجام می‌شود [۱۶]. مقادیر پارامترهای این آزمایش عبارتند از:

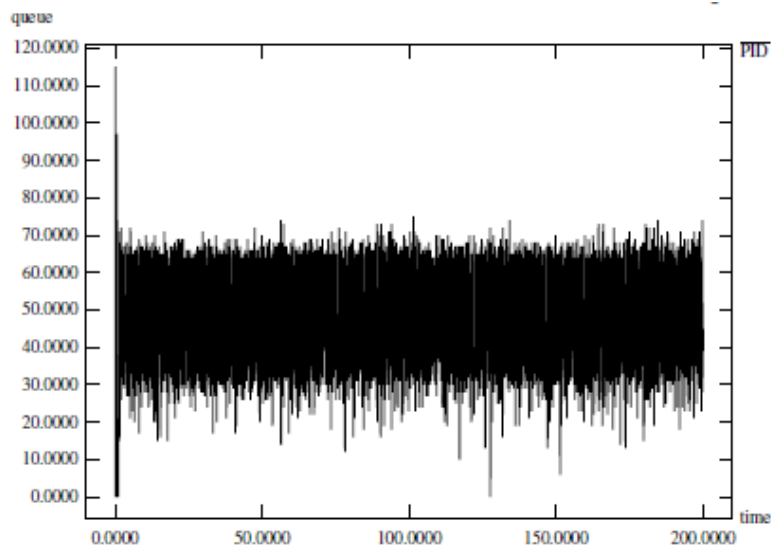
تعداد جریان (N) : FTP : ۶۰

زمان رفت و برگشت بسته (RTT) : ۲۴۶ ms

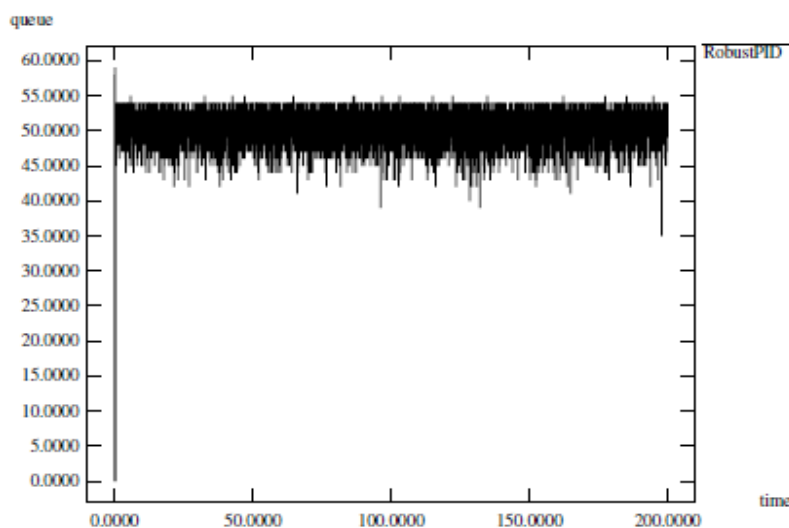
ظرفیت اتصال (C) : ۳۷۵۰ pkt/sec (۱۵ Mb/s)



(الف)



(ب)



(ج)

شکل ۴-۲: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم

نتایج نشان می‌دهد کنترل کننده PID مقاوم در مقایسه با کنترل کننده‌های PID معمولی و داهلین، می‌تواند طول صف را در مقدار مطلوب یعنی ۵۰ بسته حفظ کند و ضمن دارا بودن پاسخ گذرای سریعتر، کمترین میزان نوسان در تغییرات طول صف، در بازه زمانی شبیه‌سازی را شامل شود. در شکل ۴-۲ مشهود است که با وجود تعداد جریان FTP ۶۰، کنترل کننده داهلین در زمان گذرا توانایی جلوگیری از سرریز بافر را ندارد و این پدیده تا ثانیه ۸۰ به طول می‌انجامد. بعد از حدود ۲۰۰ ثانیه، شروع به همگرا شدن به مقدار مرجع طول صف می‌نماید و از آنجا به بعد کنترل کننده داهلین، پاسخ

لختی به شرایط دینامیک شبکه می‌دهد. کنترل‌کننده PID هم اگرچه سعی می‌کند طول صف را حدود ۲۰۰ بسته حفظ کند ولی در طول شبیه‌سازی به یک مقدار معین همگرا نمی‌شود یا به عبارتی دامنه نوسانات بسیار بالاتر از حالتی است که از روش مقاوم استفاده شده است.

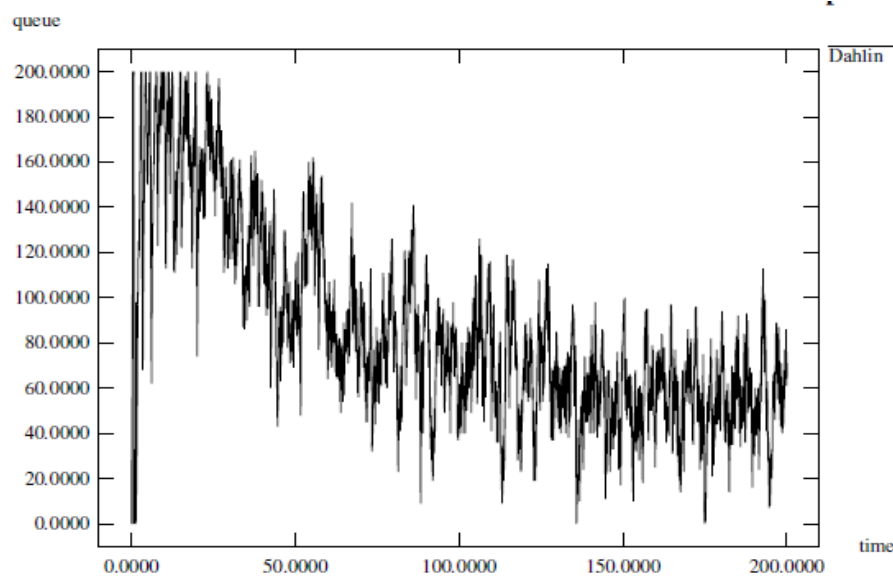
آزمایش ۲ :

این آزمایش نیز دارای ساختار و توپولوژی آزمایش ۱ است با این تفاوت که تعداد جریان‌ها کاهش یافته است. به منظور اجتناب از طول صف نامطلوب، رفتار سیستم را تحت این شرایط نیز، باید مد نظر داشت. لیست مقادیر پارامترهای آزمایش ۳ به شرح ذیل است.

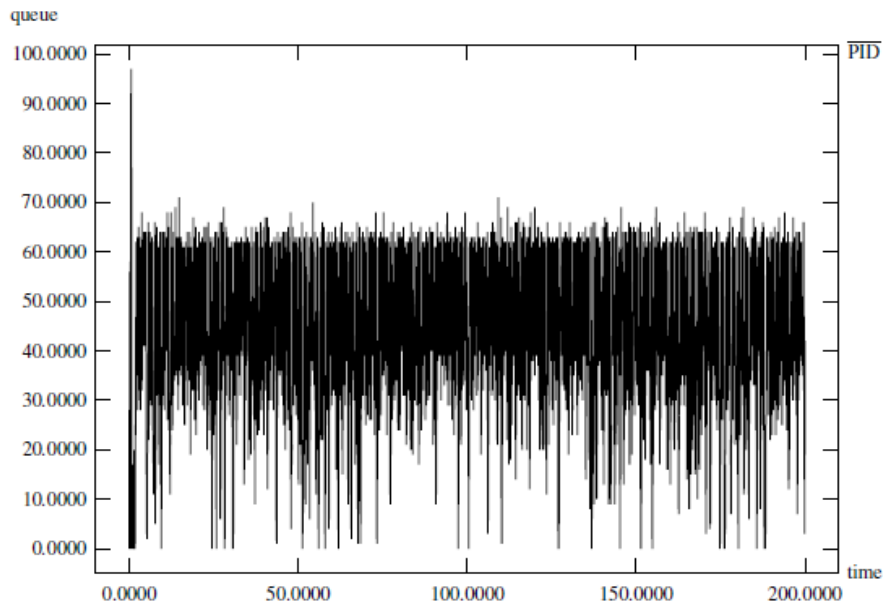
تعداد جریان‌ها (N) : ۳۰ FTP

زمان رفت و برگشت بسته (RTT) : ۲۴۶ ms

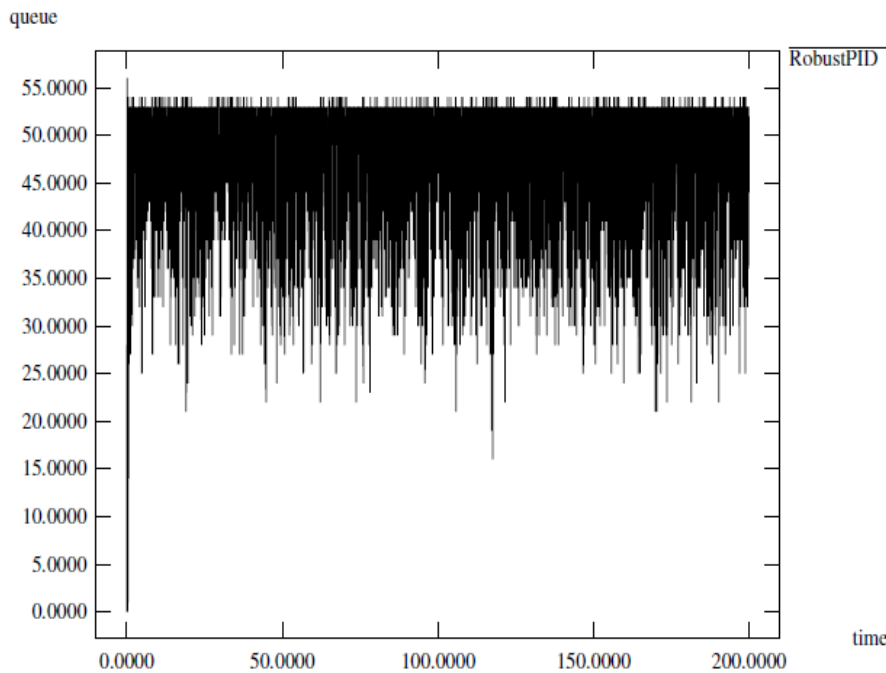
ظرفیت اتصال (C) : ۳۷۵۰ pkt/sec



(الف)



(ب)



(ج)

شکل ۴-۳: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم

نتایج آزمایش ۲ نشان می‌دهد در حضور تعداد کمی جریان FTP، کنترل کننده PID معمولی و مقاوم و داهلین (به استثنای زمان گذرا) رفتاری قابل قبول دارند به این معنی که وقتی تعداد جریان کم است همه کنترل کننده‌های فوق توانسته‌اند متوسط طول صف را در مقدار مطلوب نگه دارند و از

حداقل ظرفیت اتصال برای ارسال بسته‌ها استفاده کرده‌اند. در زمان گذرا همه کنترل‌کننده‌ها راندمان ضعیفی دارند که در این میان داهلین نامناسب‌ترین پاسخ و PID مقاوم، بهترین پاسخ گذرا را دارند.

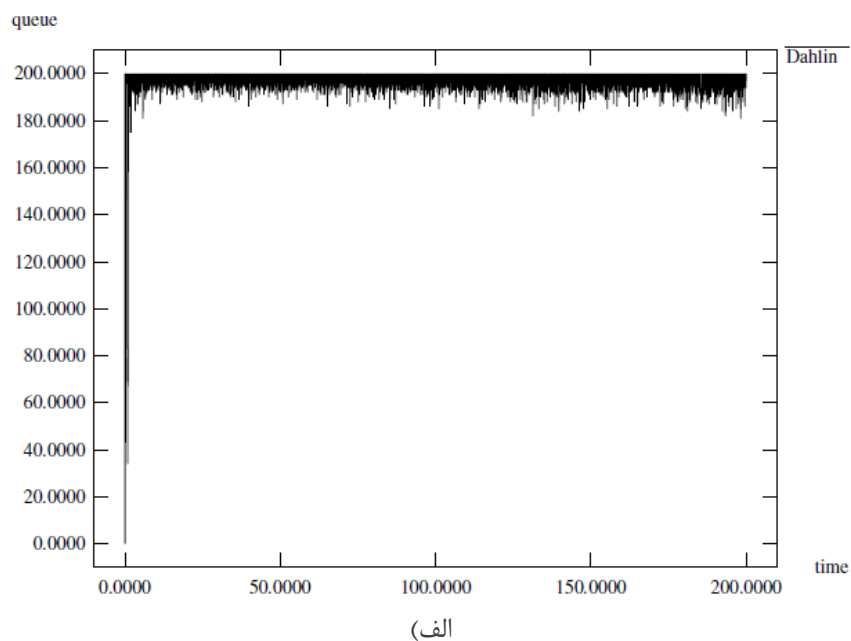
آزمایش ۳ :

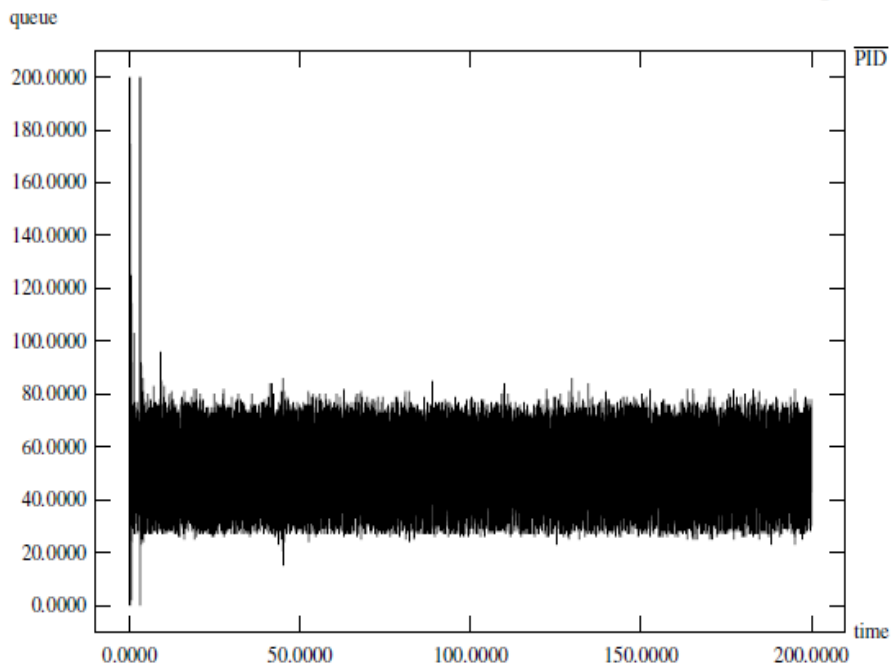
بررسی و تایید رفتار کنترل‌کننده زمانی که تعداد جریان‌ها رو به رشد است، بسیار حائز اهمیت است، یعنی در شرایطی که کنترل ازدحام باید بسیار موثرتر روی تمام اتصالات به طور هم زمان، مدیریت شود. لیست مقادیر پارامترهای این آزمایش به شرح ذیل است.

تعداد جریان‌ها (N) : ۴۰۰ FTP

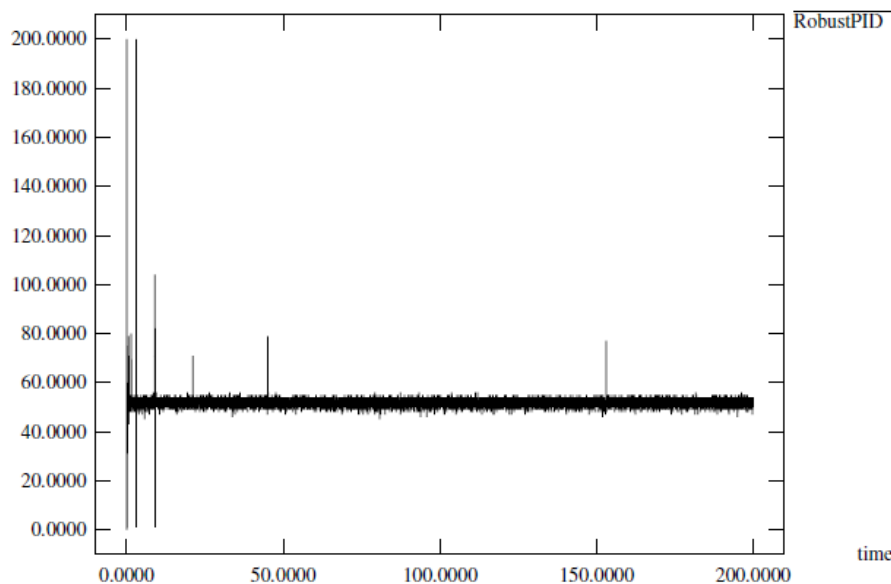
زمان رفت و برگشت بسته (RTT) : ۲۴۶ ms

ظرفیت اتصال (C) : ۳۷۵۰ pkt/sec





(ب)



(ج)

شکل ۴-۴: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم

بهترین کنترل کننده را زمانی می توان انتخاب کرد که ازدحام در آستانه وقوع است. همان طور که در شکل ۴-۴ مشهود است کنترل کننده ای که به بهترین شکل در این شرایط کمترین میزان نوسان را داشته و متوسط طول صف را در مقدار مطلوب نگه می دارد، کنترل کننده PID مقاوم است. با وجود کنترل کننده PID معمولی، ظرفیت بیشتری از لینک مورد استفاده قرار گرفته و دامنه نوسانات گویای

راندمان کمتر آن نسبت به حالت مقاوم است. کنترل‌کننده داهلین نیز نتوانسته کنترلی روی طول صف داشته باشد و در تمام مدت سرریز اتفاق افتاده است و این نشان از ناتوانی این کنترل‌کننده در شرایط بحرانی شبکه است.

آزمایش ۴ :

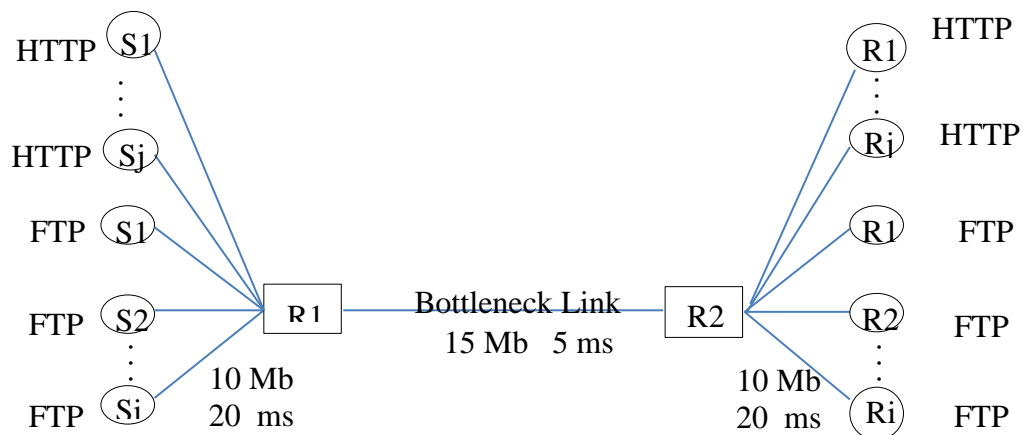
در این آزمایش مخلوطی از جریان بسته‌های FTP و HTTP از طریق مسیریاب، به منظور تجزیه و تحلیل رفتار الگوریتم‌های کنترل ازدحام در مواجهه با تعداد زیادی جریان بسته، با مشخصه‌های مختلف، فرستاده می‌شود. برای جدا نگه‌داشتن اتصالات FTP و HTTP گره منابع جدید برای HTTP در نظر گرفته شده است. در مقصد بر حسب نیاز در کنار کش، سرور HTTP نیز اضافه شده است (شکل ۴-۵). تنظیمات این آزمایش به شرح زیر است:

تعداد جریان FTP: ۶۰

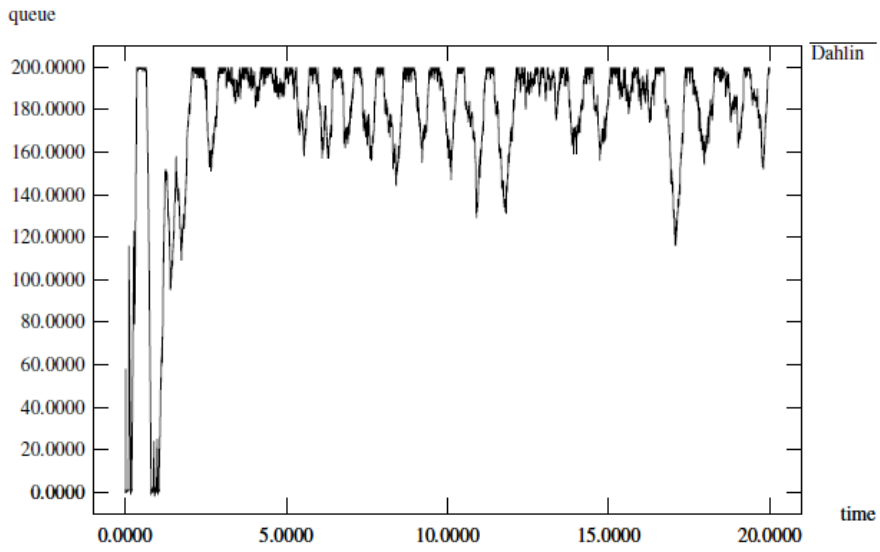
تعداد جریان HTTP: ۱۸۰

زمان رفت و برگشت بسته ثابت و برابر ۲۴۶ ms در نظر گرفته شده است.

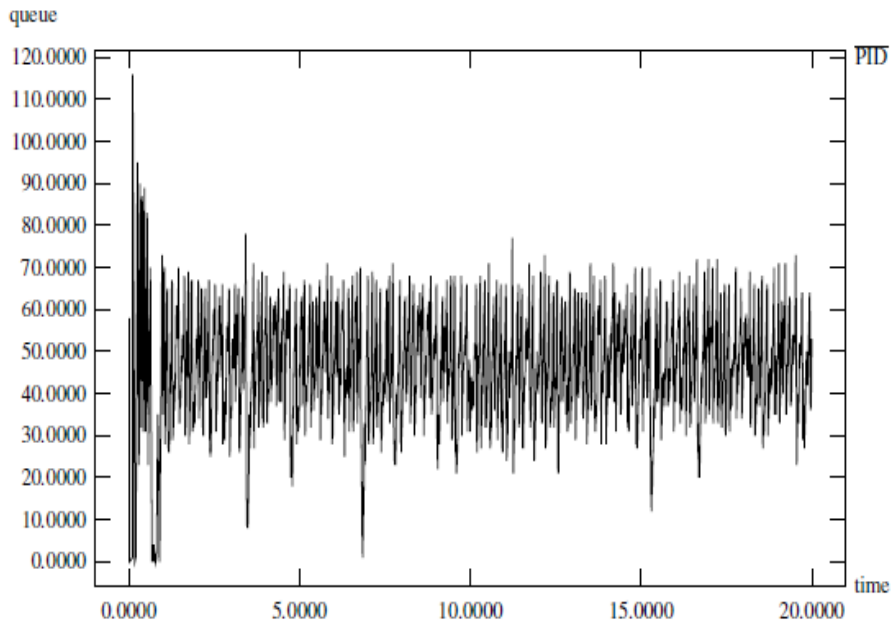
ظرفیت اتصال (C) : ۳۷۵۰ pkt/sec



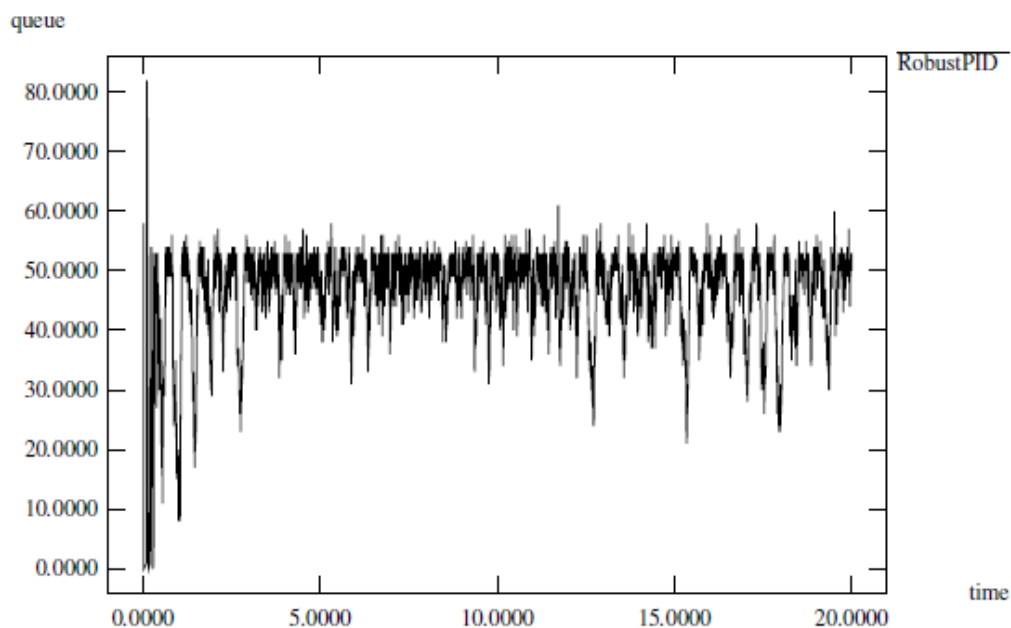
شکل ۴-۵: توپولوژی برای ترافیک مخلوط از FTP و HTTP



(الف)



(ب)



(ج)

شکل ۴-۶: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم

شکل ۴-۶ سیر تکاملی صف، تحت جریان‌های FTP و HTTP روی مسیریاب را نشان می‌دهد. در مقایسه با آزمایش وجود جریان FTP به عنوان تنها بار روی مسیریاب، معرفی بسته‌های HTTP روی عملکرد کنترل کننده تاثیر می‌گذارد. همان‌طور که می‌بینیم کنترل کننده PID مقاوم در مقایسه با کنترل کننده PID معمولی و داهلین بهترین پاسخ را داشته و توانسته به خوبی بسته‌ها را مدیریت کند و کمترین میزان نوسان در طول صف، در بازه زمانی شبیه‌سازی مشاهده می‌شود. در زمان همان‌طور که مشهود است کنترل کننده داهلین از بالاترین ظرفیت اتصال استفاده می‌کند و این نشان از افت بسته یا همان سرریز صف دارد.

آزمایش ۵:

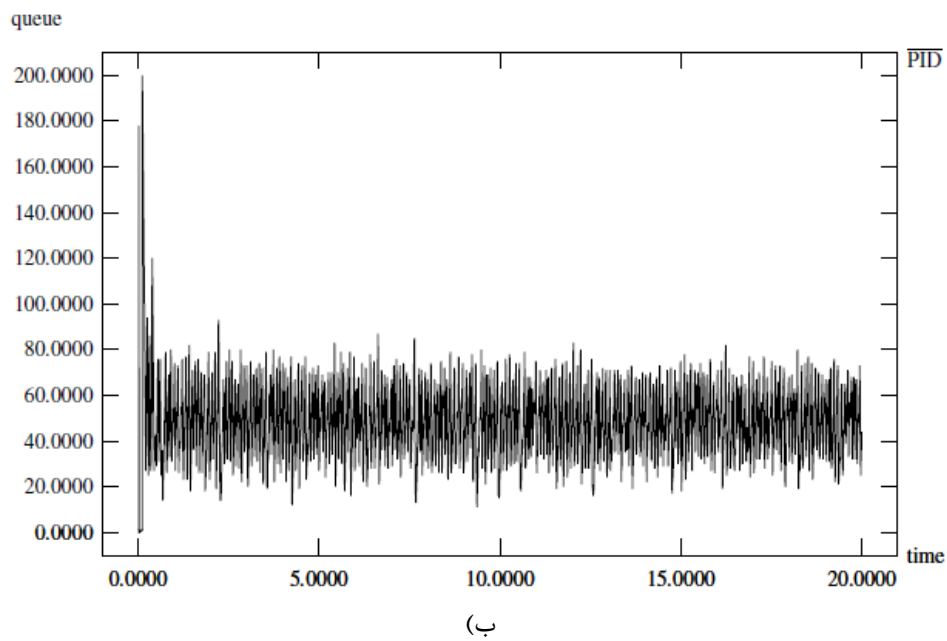
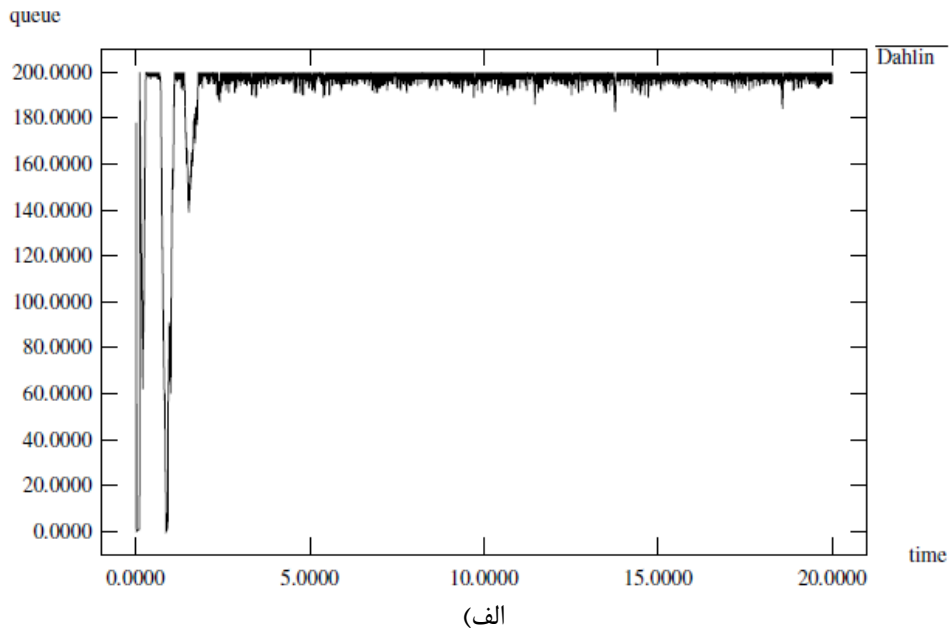
ساختار آزمایش ۴ بعلاوه افزایش تعداد جریان‌های FTP و HTTP برای این آزمایش هم در نظر گرفته شده است.

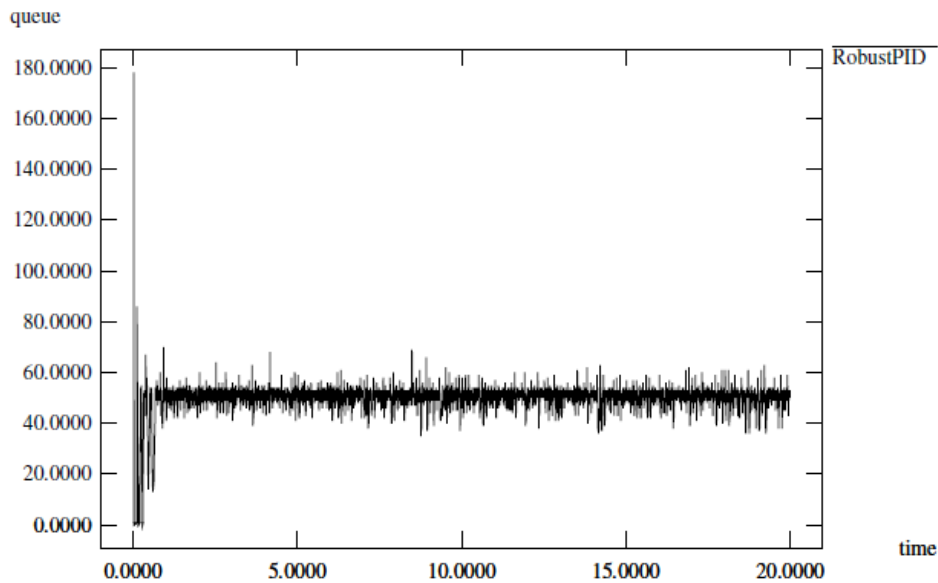
تعداد جریان FTP: ۱۸۰

تعداد جریان HTTP: ۳۶۰ (۳۶۰ مشتری + ۱ سرور)

زمان رفت و برگشت بسته ثابت و برابر ۲۴۶ ms در نظر گرفته شده است.

ظرفیت اتصال (C): ۳۷۵۰ pkt/sec





(ج)

شکل ۴-۷: تغییرات طول صف (بر حسب ثانیه) با وجود الف) کنترل کننده داهلین ب) کنترل کننده PID معمولی ج) کنترل کننده PID مقاوم

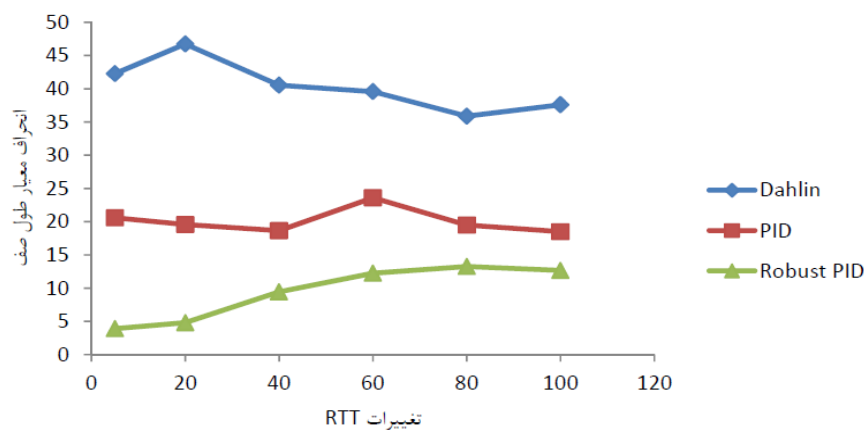
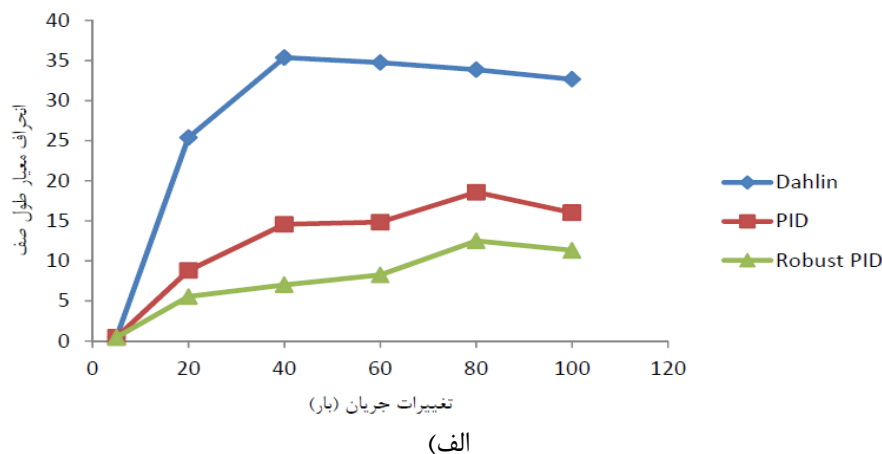
جلوگیری از سرریز صف، حتی در زمان نشست، سودمند است. زیرا علاوه بر اینکه به مسیریاب این امکان را می‌دهد تا طول صف را پایدار نگه دارد، برای نگهداری حاشیه‌ها، برای ترافیک‌های بی‌موقع نیز سودمند است. در شکل ۴-۷ توزیع طول صف با کنترل کننده داهلین نشان می‌دهد صف کاملاً پر شده است و هدف جلوگیری از سرریز صف، محقق نشده است. اما کنترل کننده PID معمولی نه تنها از سرریز صف جلوگیری کرده، بلکه تعداد بسته‌های بازگشت داده شده در زمان گذرا را کاهش داده است. کنترل کننده PID مقاوم نیز علاوه بر عملکرد مطلوب کنترل کننده PID معمولی، پایدارترین طول صف را داراست و به عبارتی کمترین میزان ظرفیت اتصال را برای ارسال بسته‌ها اشغال می‌کند که این نشان از مدیریت خوب کنترل کننده PID مقاوم دارد.

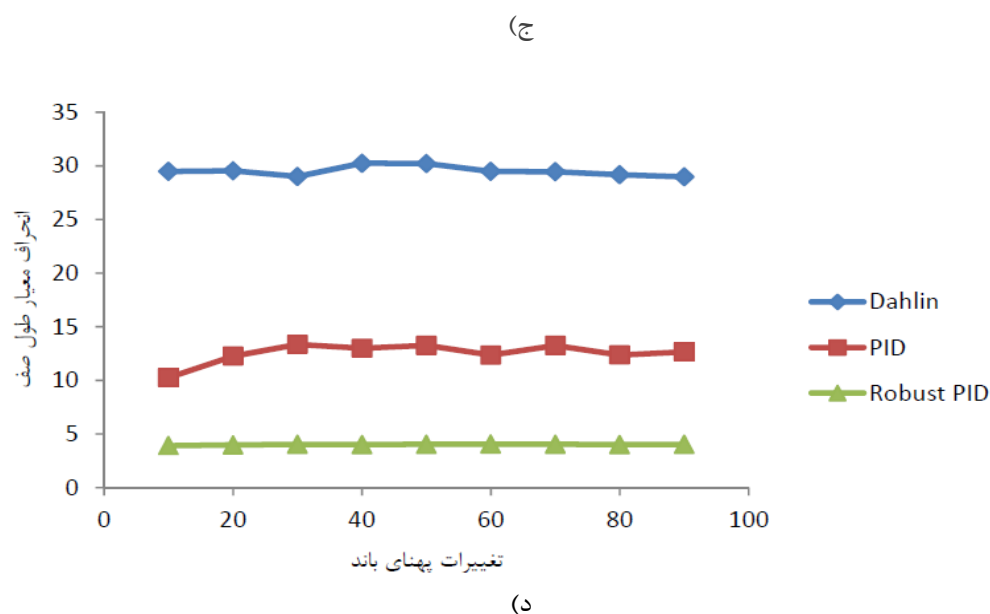
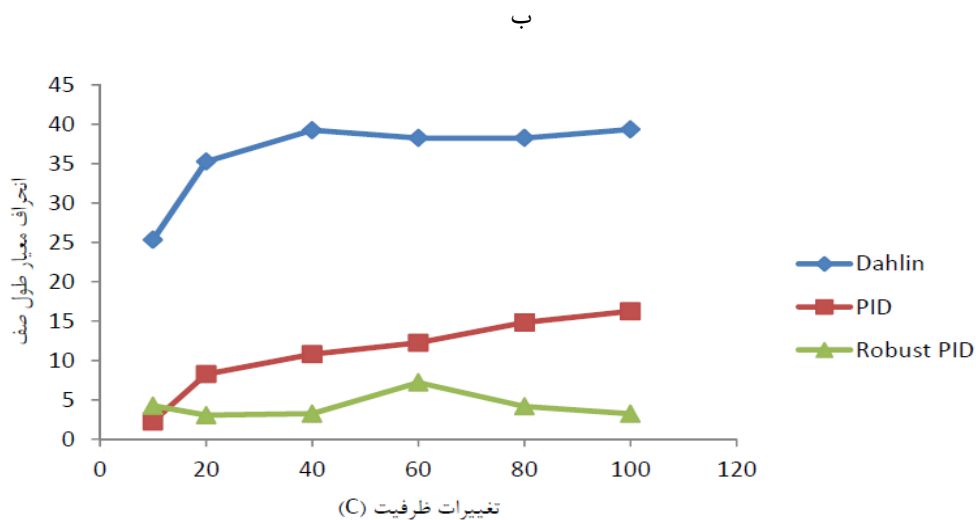
۴-۶- بررسی عملکرد کنترل‌کننده‌ها به ازای تغییرات پارامترهای

شبکه

۴-۶-۱- میزان انحراف معیار طول صف

در این آزمایش به دنبال بررسی رفتار کنترل‌کننده‌ها با استفاده از مفهوم انحراف معیار هستیم. همان‌طور که می‌دانیم انحراف معیار، میزان انحراف دست‌یافته‌ها، از مقدار استاندارد تعیین شده، است. در تمام مراحل ارزیابی مقدار مرجع (استاندارد) ۵۰ تعریف شده است. این آزمایش در شرایط مختلف، از جمله تغییرات بار (جریان)، تغییرات زمان رفت و برگشت بسته (RTT)، تغییرات ظرفیت اتصال (C) و تغییرات پهنای باند انجام شده است.



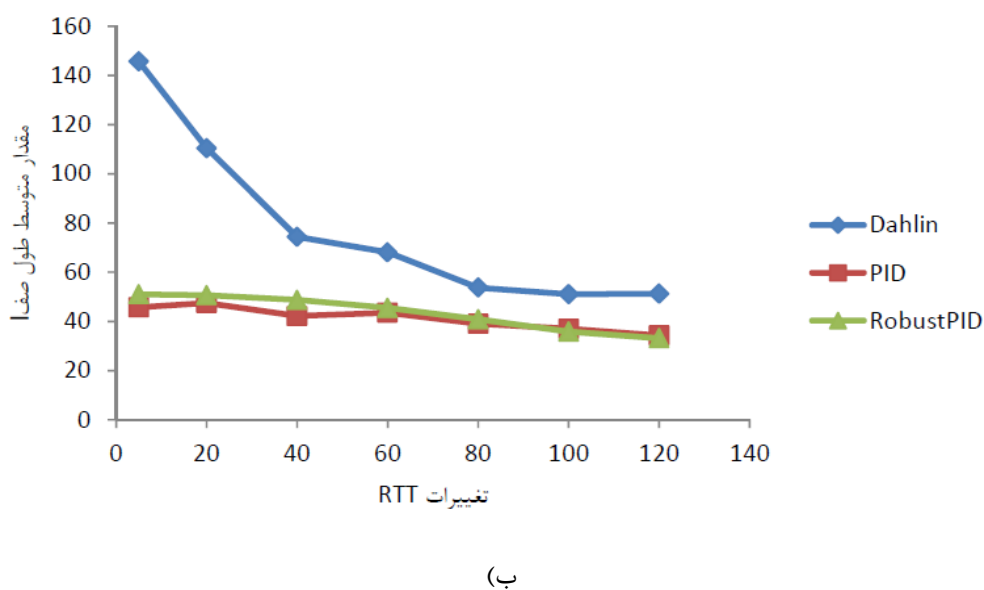
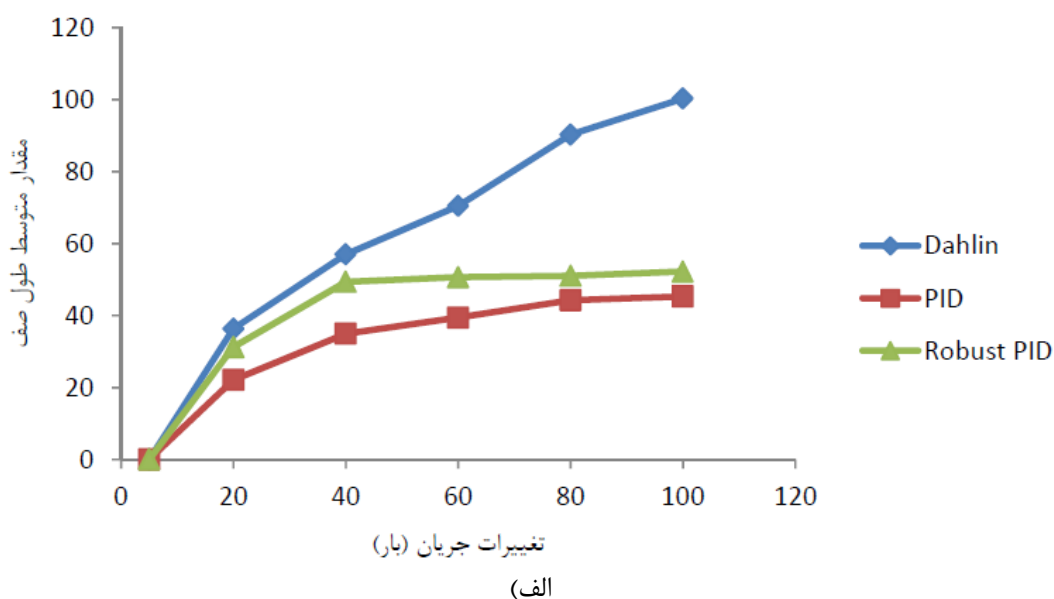


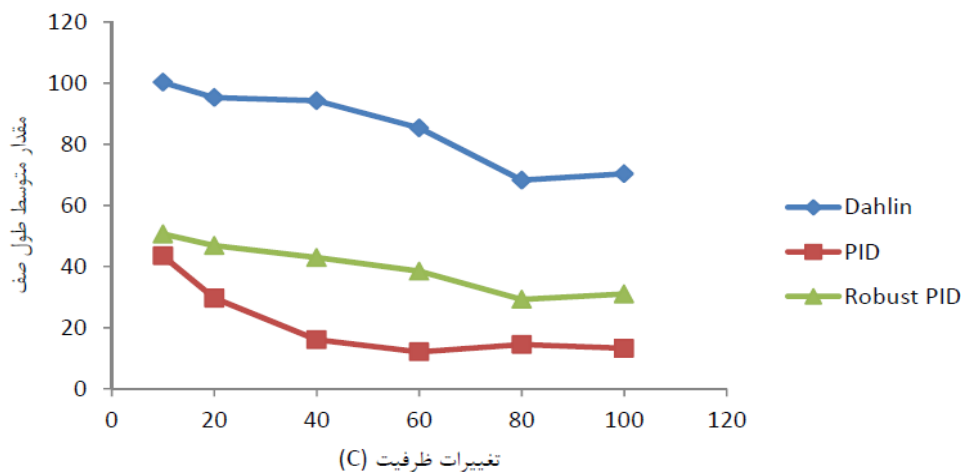
شکل ۴-۸: انحراف معیار طول صف به ازای الف) تغییرات جریان (بار) ب) تغییرات RTT ج) تغییرات ظرفیت اتصال د) تغییرات پهنای باند اتصال

از شکل‌های ۴-۸ مشهود است که کمترین میزان انحراف از مقدار استاندارد در تمام شرایط، منسوب به کنترل‌کننده PID مقاوم است. یعنی کنترل‌کننده PID مقاوم بیشترین ثبات را در برخورد با انواع نامعینی‌ها دارد. کنترل‌کننده PID رفتاری قابل قبول، نه ایده‌آل، از خود نشان می‌دهد و در این میان کنترل‌کننده داهلین انحراف شدیدی از مقدار استاندارد دارد که البته آزمایش‌های قبلی گویای دلیل این نتایج هستند.

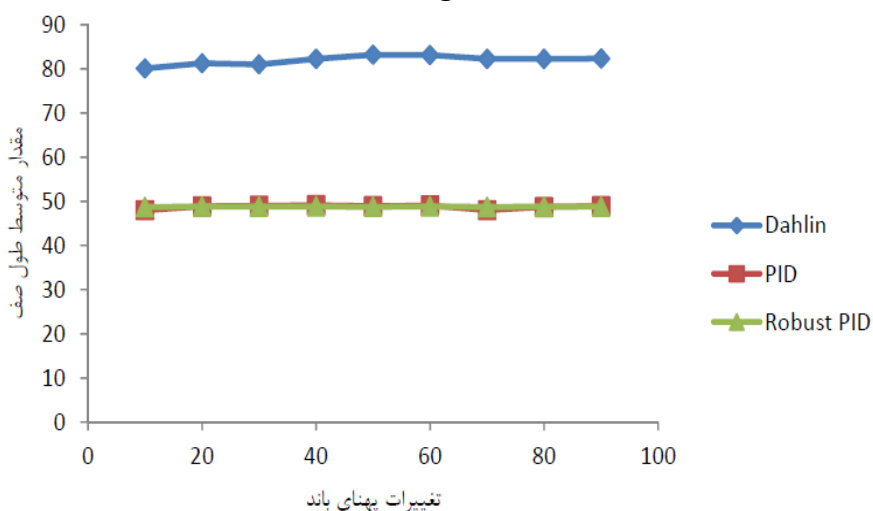
۴-۶-۲- مقدار متوسط طول صف

مقدار متوسط طول صف و پایدار نگه داشتن این مقدار حول مقدار مرجع، از اهداف اصلی کنترل کننده است که البته آزمایش قبلی به طور عمقی مبین این مسئله بود. در اینجا نیز به طور ضمنی، این معیار در بوته آزمایش قرار گرفته و تغییرات پارامترهای مختلف و اثرات آن روی مقدار متوسط طول صف بررسی شده است.





(ج)



(د)

شکل ۴-۹: مقدار متوسط طول صف به ازای الف) تغییرات جریان (بار) ب) تغییرات RTT ج) تغییرات ظرفیت اتصال د) تغییرات پهنای باند اتصال

تغییرات پارامترهای اساسی شبکه از جمله پهنای باند، ظرفیت اتصال و بار روی شبکه، کمترین تاثیر را روی عملکرد کنترل کننده PID مقاوم داشته، به طوری که شکل ۴-۹ نشان می دهد مقدار متوسط طول صف، تقریباً در مقدار مرجع، یعنی ۵۰ بسته، حفظ شده است که این همان هدف نهایی و استفاده از حداقل ظرفیت اتصال است. همان طور که نتایج نشان می دهد کنترل کننده PID نیز عملکرد خوبی را در مقابل این تغییرات از خود نشان می دهد که البته در مقایسه با حالت مقاوم آن، از نظر کیفیت عملکرد، در رده پایین تری قرار می گیرد.

فصل پنجم

جمع بندی و پیشنهادات

در این مطالعه، ابتدا به چالش مربوط به نواقص شبکه‌های ارتباطی پرداخته شد و سپس کنترل‌کننده مقاوم PID، به عنوان یک راه‌حل مناسب، برای مقابله با ازدحام در این شبکه‌ها پیشنهاد و طراحی شد. بدین منظور از کنترل‌کننده PID مقاوم در ساختار کنترلی اسمیث استفاده شد. پیش‌بینی‌کننده اسمیث یک تکنیک ساده و قدرتمند است که برای پروسه‌های با تاخیر زمانی استفاده می‌شود. تاخیر در شبکه‌های ارتباطی پدیده‌ای غیرقابل اجتناب است که به دلیل انتقال حجم انبوهی از اطلاعات اتفاق می‌افتد. پس از برخورد با تاخیر هدف بعدی دنبال شده، کاهش ازدحام یا کاهش احتمال افت بسته در شبکه انتقال است. از این‌رو کنترل‌کننده مقاوم طراحی شده، روی مسیریاب شبکه پیاده شده است و عملکرد آنها در مواجهه با اکثر شرایط احتمالی که ممکن است در شبکه رخ دهد، از قبیل تغییر در تعداد جریان‌های اعمالی به اتصال، تغییر در ظرفیت اتصال و یا پهنای باند اتصال و ... که در مقدار نامی نیستند، مورد ارزیابی قرار گرفته است. برای مقایسه رفتار این کنترل‌کننده با سایر کنترل‌کننده‌های طراحی شده بر پایه مدل ریاضی، کنترل‌کننده‌هایی نظیر PID و داهلین نیز طراحی و عملکرد آنها با کنترل‌کننده PID مقاوم مورد بررسی قرار گرفته است.

شبیه‌سازی‌ها نشان داد که کنترل‌کننده داهلین نسبت به سایر کنترل‌کننده‌ها کمترین راندمان را داراست. از این جهت که کوچکترین تغییری در شبکه اعم از تغییر پهنای باند با کمترین میزان ممکن، روی عملکرد کنترل‌کننده تاثیر منفی می‌گذارد. کنترل‌کننده PID نیز علی‌رغم کاربرد فراوان آن، در شرایطی همچون زیاد شدن بار یا تغییر پهنای باند و ظرفیت اتصال، عملکرد مطلوبی را نسبت به PID مقاوم ارائه نداده است. اما در این میان کنترل‌کننده PID مقاوم در مقایسه با کنترل‌کننده‌های دیگر توانست در اکثر شرایط شبکه، استفاده از ظرفیت اتصال را به حداقل برساند. در آزمایش‌های ۱ تا ۳ در شرایطی که تنها بار اعمالی به شبکه از نوع FTP بود و همچنین آزمایش‌های ۴ و ۵ که بارهای HTTP نیز بر شبکه تحمیل شد، با وجود کنترل‌کننده مقاوم، کمترین میزان نوسان در طول صف مشاهده شد و این از قابلیت‌های قابل پیش‌بینی، کنترل‌کننده PID مقاوم بود. چرا که مقاوم بودن به معنی پایدار نگه‌داشتن پاسخ در مقدار مطلوب با وجود نامعینی‌هایی که در شبکه وجود دارد، می‌باشد.

نتایج نشان می‌دهد چه در شرایط نامی و چه در بدترین حالت، تکنیک اصلی ارائه شده در این مطالعه، به سرعت خود را با شرایط تطبیق می‌دهد و بهترین پاسخ و مطلوب ترین پاسخ را در مقایسه با سایر کنترل کننده‌های ارائه شده در این تحقیق، دارد.

برای ادامه کار، روش هایی چون کنترل کننده‌های تطبیقی، فازی و عصبی را می توان بر اساس مدل خطی شده، برای کنترل ازدحام روی این توپولوژی، طراحی و پیاده‌سازی کرد. همچنین تغییر توپولوژی شبکه و پیاده‌سازی کنترل کننده‌های ارائه شده در این مطالعه بر روی آن، می‌تواند نتایج جالب توجه‌ای را برای ارزیابی عملکرد کنترل کننده منتخب یعنی PID مقاوم، در مقایسه با سایر کنترل کننده‌ها به همراه داشته باشد.

پیوست الف

در این پیوست یک راهنمای مرحله به مرحله از پایه شبیه‌ساز NS، برای انجام آزمایش‌ها توضیح داده شده است. بعد از نصب NS، تمام کتابخانه‌های مربوط به آن نیز باید به میزکار اضافه شود تا شبیه‌ساز بتواند ساختار کار را فراهم کند. کتابخانه‌ها شامل بیشتر ابزار کار هستند. تنها مسئله‌ای که احتیاج به اصلاح دارد الگوریتم صف است که موضوع اصلی این تحقیق می‌باشد. الگوریتم‌های صف شامل دو فایل .cc و .h می‌باشند که به زبان C++ نوشته می‌شوند. که اولی تمام توابع را توضیح می‌دهد، در حالی که دومی استفاده از متغیرها و توابع در فایل متناظر را اعلام می‌کند. هر دو فایل باید در پوشه "ns-allinone-2.35/ns-2.35/queue" قرار داده شوند.

اجرای الگوریتم‌های صف نیاز به پارامترهایی دارد که می‌توان با اصلاح کردن فایل .cc. (و نه تغییر آن) به آن دست یافت. در پوشه "/ns-allinone-2.35/ns-2.27/tcl/lib" فایلی تحت عنوان "ns-default.tcl" وجود دارد که برای هر گونه اجرای، تعدادی پارامتر مربوط به کنترل‌کننده باید در آن تنظیم شود. این اطلاعات مطابق با لیست پارامترهایی که برای هر کنترل‌کننده گزارش شده است، به شکل زیر به فایل اضافه می‌شود.

```
Queue/DLN set bytes_ false
Queue/DLN set queue_in_bytes_ false
Queue/DLN set a_ 50
Queue/DLN set b_ 35
Queue/DLN set w_ 160
Queue/DLN set qref_ 50
Queue/DLN set mean_pktsize_ 500
Queue/DLN set setbit_ false
Queue/DLN set prob_ 0
Queue/DLN set curq_ 0
```

قبل از کامپایل کردن برنامه، کنترل‌کننده صف باید به کتابخانه توابع شبیه‌ساز NS، اضافه شود. این عملیات با نوشتن کد ذیل در فایل "ns-default.tcl" در "ns-allinone-2.35/ns-2.35/tcl/lib" امکان پذیر است.

```
Simulator instproc simplex-link { n1 n2 bw delay qtype args } {
....
```

```

....
if {[string first "RED" $qtype] != -1 ||
    [string first "PI" $qtype] != -1 ||
    [string first "PID" $qtype] != -1 ||
    [string first "HINF" $qtype] != -1 ||
    [string first "DLN" $qtype] != -1 ||
    [string first "Vq" $qtype] != -1 ||
    [string first "REM" $qtype] != -1 ||
    [string first "GK" $qtype] != -1 ||
    [string first "RIO" $qtype] != -1 ||
    [string first "XCP" $qtype] != -1} {
    $q link [$link_($sid:$did) set link_]
}

```

در مرحله بعد با اضافه کردن خطوط زیر به فایل “makefile” در پوشه “ns-allinone-2.35/ns-2.35” الگوریتم‌های جدید به برنامه شناسانده می‌شوند.

```

OBJ_CC = \
....
....
queue/red-pd.o queue/pi.o queue/pid.o queue/dln.o queue/Hinf.o queue/vq.o
queue/rem.o queue/gk.o

```

بعد از هر گونه اصلاح یا تغییر الگوریتم و یا پارامترهای کنترل کننده، برنامه باید با تغییرات جدید کامپایل شود که با اجرای دستور “make” در ترمینال لینوکس این عمل انجام خواهد شد.

بعد از تنظیم الگوریتم صف، با نوشتن یک فایل *.tcl توپولوژی شبکه باید طراحی شود. در پیوست ب، کد نوشته شده مربوط به توپولوژی شبکه که در فصل ۴ اشاره شده، گزارش داده شده است.

پیوست ب

کد مربوط به فایل tcl شبیه‌سازی

```
set ns [new Simulator]

set trFile [open out.tr w]
$ns trace-all $trFile
set namFile [open out.nam w]
$ns namtrace-all $namFile
set qsize [open queuesize.tr w]
set qbw [open queuebw.tr w]
set qlost [open queuelost.tr w]

#number of FTP flows (it varies according to the experiments)
set r 60

# number of HTTP connections (expressed as a multiple of FTP connections)
set s [expr $r*3]

# number of UDP flows for Experiment 7 - 10% of the FTP flows
set u [expr $r/10]

# The topology (nodes and links) is then designed
# define core nodes (routers)
set nc0 [$ns node]
set nc1 [$ns node]

#define source and destination nodes for FTP connections
set i 1
while {$i <= $r} {
    set src($i) [$ns node]
    set des($i) [$ns node]
    incr i
}

#define router properties: capacity, latency time, QUEUING SCHEME and queue
maximal length
$ns duplex-link $nc0 $nc1 15Mb 10ms DLN
$ns queue-limit $nc0 $nc1 200

#define links between end-point nodes and core nodes, with bandwidth and propagation
delay
set i 1
while {$i <= $r} {
    $ns duplex-link $src($i) $nc0 10Mb 5ms DropTail
```

```

    $ns duplex-link $des($i) $nc1 10Mb 5ms DropTail
    incr i
  }

#random generator
set rng [new RNG]
$rng seed 0

#parameters for random variable in link delay

set RVdly [new RandomVariable/Uniform]

$RVdly set min_ 130
$RVdly set max_ 220
$RVdly use-rng $rng

for {set i 1} {$i <= $r} {incr i} {

set dly($i) [expr [$RVdly value]]
}

#After having specified physical and network-layer elements, the FTP transport-level
connections must be created:
set i 1
while {$i <= $r} {
set tcp($i) [$ns create-connection TCP/Newreno $src($i) TCPSink $des($i) 0]
set ftp($i) [$tcp($i) attach-source FTP]
$ftp($i) set packetSize 500
incr i
}

#The simulations are 200 seconds long and the data are collected in the whole period of
simulation. A tracing variable is added in the router queue in order to get the results on
its behavior.
for {set i 1} {$i <= $r} {incr i} {
    $ns at 0.0 "$ftp($i) start"
    }
# $ns at 0.0 "start-http"
#for {set j 1} {$j <= $u} {incr j} {
#    $ns at 0.0 "$cbr($j) start"
#    }

#Tracing the queue
set pidq [[ $ns link $nc0 $nc1] queue]
set tchan_ [open all_3.q w]
$pidq trace curq_
$pidq attach $tchan_
$ns at 200 "finish"

```

```

# Define 'finish' procedure (include post-simulation processes)
proc finish {} {
    global tchan_
    set awkCode {
        {
            if ($1 == "Q" && NF>2) {
                print $2, $3 >> "temp_3.q";
                set end $2
            }
            else if ($1 == "a" && NF>2)
                print $2, $3 >> "temp_3.a";
        }
    }
    set f [open temp_3.queue w]
    puts $f "Device: Postscript"
    if { [info exists tchan_] } {
        close $tchan_
    }
    exec rm -f temp_3.q temp.a
    exec touch temp_3.a temp.q
    exec awk $awkCode all_3.q
    puts $f "\"Robpid1
    exec cat temp_3.q >@ $f
    close $f
    exec xgraph -bb -tk -bg white -x time -y queue temp_3.queue -geometry
800x600 &
    exit 0
}

global ns trFile namFile
$ns flush-trace
close $namFile
close $trFile
close $qsize
close $qbw
close $qlost
}

}

# QUEUE MONITORING
set qfile [$ns monitor-queue $nc0 $nc1 [open queue.tr w] 0.05]
[$ns link $nc0 $nc1] queue-sample-timeout;
# The following procedure records queue size , bandwidth and loss rate
proc record {} {
    global ns qfile qsize qbw qlost nc0 nc1
    set time 0.05
    set now [$ns now]
    # print the current queue size in $qsize , the current used

```

```
# bandwidth in $qbw , and the loss rate in $qloss
$logfile instvar parrivals_ pdepartures_ bdrops_ bdepartures_ pdrops_
puts $qsize "$now [expr $parrivals_-$pdepartures_-$pdrops_]"
puts $qbw "$now [expr $bdepartures_*8/1024/ $time]"
set bdepartures_ 0
puts $qlost "$now [expr $pdrops_/$time]"
$ns at [expr $now+$time] "record"
}
$ns at 0.0 "record"

#The simulation object is then executed.
$ns run
```

پیوست پ

گسسته سازی کنترل کننده داهلین

کنترل کننده داهلین دارای تابع تبدیل (پ-۱) می باشد.

$$\frac{p}{\delta q} = K_1 \frac{K_2 s + 1}{\lambda s} \quad (\text{پ-۱})$$

برای گسسته سازی با روش توستین باید قرار دهیم:

$$s = \frac{2}{T_s} \frac{z-1}{z+1} \quad (\text{پ-۲})$$

بنابراین داریم:

$$\frac{p}{\delta q} = K_1 \frac{K_2 \left(\frac{2}{T_s} \frac{z-1}{z+1} \right)}{\lambda \left(\frac{2}{T_s} \frac{z-1}{z+1} \right)} \quad (\text{پ-۳})$$

$$2\lambda(p_k - p_{k-1}) = 2K_1 K_2 (q_k - q_{k-1}) + K_1 T_s (q_k + q_{k-1} - 2q_{ref}) \quad (\text{پ-۴})$$

$$p_k = p_{k-1} + a(q_k - q_{k-1}) + b(q_k + q_{k-1} - 2q_{ref})$$

$$a = \frac{K_1 K_2}{\lambda} \quad (\text{پ-۵})$$

$$b = \frac{K_1 T_s}{2\lambda}$$

کد متناظر با رابطه فوق به صورت زیر است.

$$p = p_{old} + a(q - q_{old}) + b(q - q_{old} - 2q_{ref})$$

$$p_{old} = p \quad (\text{پ-۶})$$

$$q_{old} = q$$

مراجع

1. www.basircyber.com

۲. محمدحسین یغمایی مقدم، (۱۳۸۸)، "شبکه‌های کامپیوتری و اینترنت"، جلد ۱، چاپ اول، انتشارات دانشگاه فردوسی مشهد.

3. Alvarez T. (2012), "Design of PID Controllers for TCP/AQM Wireless Networks", Proceedings of the World Congress on Engineering, WCE, London, U.K.

۴. شاهرخ زاده ب، (۸۹)، پایان‌نامه ارشد، "بررسی و ارزیابی کارایی روش های انتها-به-انتهای TCP در شبکه های سیار"، دانشکده کامپیوتر، دانشگاه آزاد اسلامی واحد تهران جنوب.

۵. ابوالمعصومی ا. بهشتی م. جلیلی س.، (۸۵)، "کنترل ازدحام در شبکه های کامپیوتری مبتنی بر TCP"، دوازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، ص ۲، تهران.

6. Floyd S. Jacobson V., (1993) "Random early Detection gateways for congestion avoidance", *IEEE/ACM Trans on Networking*, vol. 1, no. 4, pp. 397-413.

7. Christiansen M. Jeffay K. Ott D. Smith F., (2000), "Tuning RED for Web Traffic", in *Proceedings of ACM/SIGCOMM*, pp. 139-150.

8. Feng W. Shin K. Kandlur D. Saha D., (2002), "The BLUE active queue management Algorithm", *IEEE/ACM Trans. Networking*, vol. 10, no.4, pp. 513-528

9. Wyrowski B. and Zukerman M., (2002) "GREEN: An active queue man algorithm for a self managed Internet", in *Proceedings of ICC , New York*, vol. 4, pp. 2368-2372

10. Athuraliya S. Li V. H. Low S. H. Yin Q., (2001) "REM: Active Queue Management", *IEEE Network*, vol. 15, no.3, pp. 48-53

11. Jacobson V., (1988), "Congestion avoidance and control", *proceeding of SIGCOM88, ACM*, pp. 278-299

12. Jacobson V., (1993), "Modified TCP congestion avoidance algorithm", *IEEE/ACM Trans on Networking*, vol. 1, no. 4, pp. 597-615.

13. Wang R. Valla M. Sanadidi M.Y. Ng B. Gerla M., (2002), "Efficiency/friendliness tradeoffs in TCP Westwood", IEEE Symposium on Computers and Communications, Taormina, Italy.

14. Barzamini R., Shafiee M., (2011), "LMI Based Switching Congestion Controller for Multiple Bottleneck Packet Switching Networks", *Journal of American Science*, vol. 76, pp. 254-261
15. Tanenbaum A. S., (2003), "*Computers Networks*", 4'th Edition, Prentice-Hall Inc. New Jersey.
16. Misra C. V. W-B Gong, Towsley D., (2000) "Fluid-based analysis of a network of AQM router supporting TCP flows with an application to RED", in *Proc. of ACM SIGCOMM'00*, vol. 30, no. 4, pp.151-160.
17. Holot C. V. Misra V. Towsley D. Gong W., (2001), "On designing improved controllers for AQM routers supporting TCP flows", in *Proceeding of INFOCOM*, vol. 3, pp. 1726-1734.
۱۸. مرادی ا.، پایان نامه ارشد: "کنترل ازدحام در شبکه های TCP با استفاده الگوریتم های کنترل کلاسیک"، دانشگاه فنی، دانشگاه تربیت مدرس.
19. Yanfei F. Fengyuan R. Chuang L., (2003), "Design a PID Controller for Active Queue management", in *proc of IEEE, ISCC'03*, pp. 1530-1346.
20. Aström K. Hägglund T., (1995), "PID Controllers: Theory, Design and Tuning", Instrument Society of America.
21. Zhang W.D. (1996), PhD thesis, "The Robust Control of Time Delay Process", Zhejiang University, Zhejiang, China.
22. Cicco L. Mascolo S. Niculescu S., (2011), "Robust stability analysis of Smith predictor-based congestion control algorithms for computer networks", *Automatica* 47, pp. 1685–1692
23. Demers A. Keshav S. Shenker S., (1989), "Analysis and Simulation of a Fair Queueing Algorithm", *Journal of Internetworking Research and Experience*, In *Proceedings of the ACM SIGCOMM '89*, vol. 1, pp. 1-12
۲۴. جلیلی ر. حسینی س.م.، (۲۰۰۷) پایان نامه ارشد، "SRED الگوریتمی برای مدیریت پویای صف در حفاظ"، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف.
25. Lin D. Morris R., (1997), "Dynamics of Random Early Detection", In *Proceedings of the ACM SIGCOMM '97*, pages 127-137.
26. Nagle J., (1984) "Congestion Control in IP/TCP Internetworks", RFC 896, FACC.

27. Ryu S. Rump C. Qiao, (2004), “Advances in Active Queue Management (AQM) based TCP congestion control” *Telecommunication Systems*, vol. 25, no. 3, pp. 317-351.
28. Long G.E. B. Fang J.S. Sun and Z.Q. Wang, (2010), “Novel Graphical Approach to Analyze the Stability of TCP/AQM Networks” *Acta Automatica Sinica*, vol. 36, pp. 314-321.
29. Hollot C.V. and Y. Chait, (2001), “Nonlinear stability analysis for a class of TCP/AQM networks”, In Proceedings of the 40th IEEE Conference on Decision and Control, Orlando, USA.
30. hlorari, (1989), “Robust **Process** Control”, Prentice Hall.
31. Hollot C. V. Vishal Misra, Donald Towsley, Weibo Gong,, (2002), “Analysis and Design of Controllers for AQM Routers Supporting TCP Flows”, *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, vol. 47, no. 6, pp. 945-959
32. Li Yu a, MaodeMab n. WeidongHuc ZiboShi a. YantaiShu, (2011), “Design of parameter tunable robust controller for active queue management based on Hinf control theory”, *Journal of Network and Computer Applications* 34, pp. 750–764
33. Yananaka K. and shimemura E., (1973), “Effects of Mismatched Smith Predictor on stability in systems with Tim-delay”, *Automatica*, vol. 23, pp.787-791.
34. Zhang, W.D., Y. X. Sun, and X. M. Xu., (1998), “Two Degree-of-Freedom Smith Predictor for Process with Time-delay” *Automatica*, 34(10), pp. 1279-1282
35. N.Abet and K.Yamanaka, (2003), “Smith Predictor Control and Internal Model Control”, SICE Annual Conference in Fukui, Fukui University, Japan.
36. Morari, M. and E. Zafiriou, (1989), “Robust Process Control”, Prentice Hall, Englewood Cliffs, NY.
37. Horn, I.G et al, (1996), “Improved Filter Design in Internal Model Control”, *Ind. Eng. Chem. Res.*, 35, pp. 3437-3441
38. Dahlin, E. B., (1968), “Designing and tuning Digital Controllers”, *Instr. and Contr. Systems*, 41, pp. 77-83
39. Palmor Z.J., (1980), *Int. J. Control* 32, 937.
40. Dongkwon Lee, Moonyong Lee, Suwhan Sung, Indium Lee, (1999), “Robust PID tuning for Smith predictor in the presence of model uncertainty”, *Journal of Process Contorl* 9, pp. 79-85
41. Giglio A., (2007), master thesis, “Router-based Congestion Control through Control Theoretic Active Queue Management”, Stockholm, Sweden

Abstarct

Wide use of Transmission Control Protocol (TCP) in Internet and developing volume of sent data through the Web cause studying some issues like network congestion that is becoming more attractive for researchers in the past decades. Increasing packets lost rate and decreasing network efficiency make basic challenges for users and has decreased the reliability of network. In this thesis, at first congestion issue and importance of congestion control is explained and then discussed about suggested solution using control theory which means designing controllers by math concepts and place on routers which causes to decrease packet drop probability. To get the desired result, smith predictor is implemented as the best structure to guarantee stability and encounter delay in proper form and its operation will be evaluate to control the system. Subsequently Dahlin controller, PID controller and Robust PID controller has placed in this structure to compare the result of operations. Smith structure changes by Using concept of Equivalent Gain Plus Time Delay For robust design till entrance of parametric unknown uncertainties, existent PID controllers in this structure exhibit acceptable operation. The new control structure named Robust PID controller. Since this study related to network data system for more comprehension we supposed a topology with two nodes (routers) and some senders and receivers which has simulated in NS simulator to have more real results. The result shows Robust PID controller has the best operation in deferent condition of network compare to other controllers and preserve the queue length on desired value and use minimum capacity of router buffer for data transmission. To implement controllers in NS, analog to digital conversion technique is used. After preparation the related code, controllers have placed on related routers and the results has evaluated.

Key words:

Congestion, Robust PID, router, Smith predictor, TCP



University of shahrood

Faculty of Electrical & Computer Engineering

MSc. Thesis in Control Engineering

Robust PID Congestion Controller of networks

Mansoureh Peyvandi

Supervisor:

Dr. alfi

March 2015