

ارزیابی و مقایسه تاثیر الگوریتم‌های مدیریت صف در عملکرد شبکه‌های کامپیوتری

فرناز امین^۱، کیارش میزانیان^۲

^۱گروه کامپیوتر - دانشگاه یزد، Farnaz.amin@stu.yazduni.ac.ir

^۲گروه کامپیوتر - دانشگاه یزد، K.mizanian@yazduni.ac.ir

چکیده - 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 می‌باشد. با کمک این مکانیزم‌ها، وقوع تراکم کنترل شده و از کاهش کارایی شبکه جلوگیری می‌شود. در این مقاله، عملکرد الگوریتم‌های صف‌بندی مختلف، بر اساس نتایج شبیه‌سازی با هم مقایسه شده است. برای مقایسه این الگوریتم‌ها سه معیار را در نظر گرفتیم: توان عملیاتی شبکه، نرخ گم شده بسته‌ها و پیچیدگی پیاده‌سازی. در این مقاله به بررسی مفاهیم و مکانیزم‌های مدیریت فعال صف و زمانبندی در شبکه‌های IP پرداختیم و عملکرد هر کدام از این الگوریتم‌ها را از نظر توانایی در افزایش کارایی شبکه و فراهم کردن عدالت بیشتر ارزیابی شده‌اند.

کلید واژه - زمانبندی، کارایی شبکه، کنترل ازدحام، مدیریت صف.

۱ - مقدمه

با توجه به اینکه مسیریاب یک شبکه محلی گلوگاه ارتباطی آن شبکه با اینترنت است، فراهم کردن کارایی آن مسأله مهمی است. یکی از مشکلاتی که باعث کاهش کارایی یک مسیریاب می‌شود وقوع ازدحام در آن است [۱]. ازدحام هنگامی رخ می‌دهد که ترافیک رسیده به یک اتصال از بیشینه ظرفیت آن اتصال تجاوز کند. اما قرار دادن صف‌های بزرگ برای مقابله با ازدحام که اغلب اوقات هم پر باشد باعث کاهش کارایی می‌شود؛ زیرا این امر باعث افزایش قابل ملاحظه متوسط تأخیر در شبکه می‌شود. به این منظور روش‌هایی برای اجتناب از وقوع ازدحام و نیز مقابله با آن ارائه شده‌اند. برخی از این راهکارها روش‌هایی برای مدیریت صف و زمان بندی پردازش بسته را بکار می‌گیرند.

با توجه به اینکه انتخاب مناسب الگوریتم مدیریت صف تاثیر زیادی بر کارایی شبکه دارد. در این مقاله کارایی الگوریتم‌های صف‌بندی که در نرم‌آبزار شبیه‌ساز ns2 پیاده‌سازی شده‌اند را با هم مقایسه کرده و به بیان ویژگی‌های هر کدام می‌پردازیم.

۲ - مدیریت صف

تا کنون روش‌های متفاوتی برای مقابله با مشکل ازدحام بکار گرفته شده است. با مشکل ازدحام یا با فراهم کردن روش‌ها و الگوریتم‌هایی برای اجتناب از ازدحام در فرستنده و یا با بکار بردن تکنیک‌های به منظور مدیریت و اختصاص ظرفیت اتصال

صف‌ها، محلهایی هستند که بسته‌های اطلاعاتی در آنها نگهداری شده و یا دور انداخت^۱ می‌شوند. منظور از مدیریت صف^۲ الگوریتمی است که نحوه ورود بسته به صف‌های مختلف را بررسی می‌کند. به عبارت دیگر تصمیم‌گیری می‌کند که کدام بسته می‌تواند به صف مربوط وارد شود و کدام بسته باید دور ریخته شود. آنچه در مورد الگوریتم‌های صف اهمیت دارد، حجم بسته‌های عبور داده شده و بسته‌های دور ریخته شده و به تبع آن، محاسبه کارایی^۳ شبکه است. مسائل مرتبط با کارایی در شبکه‌ها از اهمیت ویژه‌ای برخوردارند. وقتی صدها یا هزاران کامپیوتر به هم متصل می‌شوند، تعامل پیچیده آن‌ها با تبعات غیرمنتظره‌ای همراه است و در اکثر مواقع، این پیچیدگی‌ها منجر به کارایی بسیار ضعیف شبکه می‌شود و هیچ‌کس هم نمی‌داند که علت چیست. منشاء برخی از مشکلات کارایی به استفاده بیش از حد منابع موجود در شبکه برمی‌گردد. اگر یکباره ترافیکی بیش از توان و ظرفیت مسیریاب به آن تحویل داده شود، ازدحام^۴ پدید آمده و موجب کاهش کارایی می‌شود.

¹ drop

² buffer management

³ throughput

⁴ Congestion

در مسیریاب‌ها (روش‌های مدیریت صف) مقابله می‌شود. در این قسمت الگوریتم‌های ساده و پویای مدیریت صف که به منظور مقابله با ازدحام در مسیریاب‌ها ارائه شده‌اند را مرور می‌کنیم.

۲-۱ - 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 نشده اند همه بسته‌های ورودی پذیرفته می‌شوند و به محض پر شدن، همه ورودی‌ها دور انداخته می‌شوند تا وقتی که ظرفیت خالی ایجاد شود. الگوریتم‌های مختلفی که در این حوزه ارائه شده‌اند به نحوی یا سعی در عادلانه‌تر کردن مدیریت صف و یا سعی در کاهش پیچیدگی و افزایش کارایی دارند.

الگوریتم FIFO (یا Drop-Tail) 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.

مهم‌ترین معایب این روش این است که یک جریان انفجاری می‌تواند کل فضای بافر چنین صفی را اشغال کند و باعث عدم سرویس‌دهی مناسب به سایر جریان‌ها شود. ثانیاً در هنگام ازدحام بسته‌ها، این صف‌بندی به نفع جریان‌های UDP عمل می‌کند [۲].

مهم‌ترین اشکالاتی که به روش صف‌بندی Drop-Tail وارد می‌شود این است که این روش بین بسته‌هایی که از مبادی مختلف رسیده تمایزی قایل نمی‌شود، در نتیجه ممکن است برخی از آن مبادی سهم زیادی از پهنای باند یک مسیریاب را اشغال کنند، در الگوریتم FQ^1 سعی شده این اشکال رفع شود [۳]، به این ترتیب که برای هر جریان عبور کننده از مسیریاب یک صف جداگانه در نظر گرفته می‌شود. سپس به این صف‌ها به طریق چرخشی 2 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. اگرچه الگوریتم FQ مزایای خوبی برای حل مسأله کنترل ازدحام دارد، اما پیچیدگی پیاده سازی آن مانع از بکارگیری آن در شبکه‌های با سرعت بالا می‌شود.

۲-۲ - مدیریت پویای صف

مدیریت پویای صف (AQM^3) به معنی مدیریت صف به منظور اجتناب از پر شدن و حبس 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 نگه داشتن متوسط طول صف، بسته‌ها را سریعاً و پیش از اینکه صف سرریز کند، دور می‌ریزند [۴].

در میان روش‌های مدیریت پویای صف، الگوریتم RED^4 از همه مشهورتر است و در واقع یک الگوریتم کلاسیک در این حوزه به شمار می‌آید. الگوریتم‌هایی که پس از RED ارائه شده‌اند، عمدتاً ضعف‌های RED را جبران نموده‌اند؛ به عنوان مثال برخی از این الگوریتم‌ها سعی داشته‌اند که عدالت بیشتری نسبت به RED فراهم کنند و برخی دیگر روش‌های متفاوتی برای به محاسبه پارامترها ارائه کرده‌اند.

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 خبردهی 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 بوسیله کنترل متوسط طول صف است. در الگوریتم RED با رسیدن هر بسته، متوسط طول صف محاسبه و با دو آستانه کمینه^۵ و بیشینه^۶ 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 که P تابعی از متوسط طول صف (avg) خواهد بود.

³ Active Queue Management

⁴ Random Early Discard

⁵ minimum threshold

⁶ maximum threshold

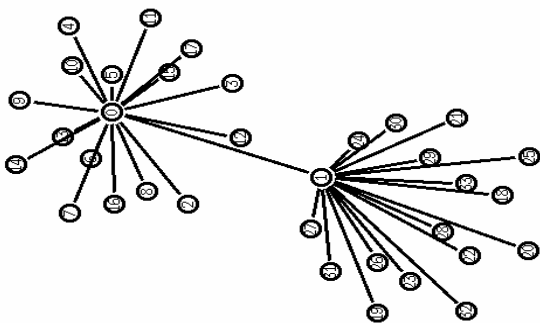
¹ Fair Queuing

² Round Robin

الگوریتم‌های صف‌بندی که در حال حاضر در بازار شبیه ساز شبکه ns2 پیاده سازی شده اند عبارتند از: Drop-Tail، RED، FQ، CBQ، DRR و SFQ. در سه سناریو مختلف این الگوریتم‌های صف را با هم مقایسه کرده‌ایم، در سناریو اول مطابق شکل (۱)، از ۱۶ گره مبدا برای ارسال بسته و ۱۶ گره مقصد استفاده کردیم، در سناریو دوم مطابق شکل (۲)، از ۳۲ گره مبدا و در سناریو سوم مطابق شکل (۳)، از ۶۴ گره مبدا استفاده کردیم و برای اینکه سناریو به شبکه واقعی نزدیک باشد، هر کدام از گره‌های مبدا در زمان تصادفی شروع به ارسال ترافیک TCP می‌کنند و در هر سه سناریو لینک $n_0 - n_1$ به عنوان گلوگاه شبکه نقش کلیدی در انتقال بسته های اطلاعاتی ایفا میکند؛ از این رو تست انواع الگوریتم‌های صف ایستا و پویا، روی این لینک پیاده‌سازی می‌شود و مشخص خواهد شد که الگوریتم‌های صف-بندی مختلف چگونه با نوبت‌دهی و زمانبندی بسته‌های دریافتی و نحوه برخورد با بسته‌های سرریز، کارایی شبکه را تحت تاثیر قرار می‌دهند. پارامترهایی که برای این شبیه‌سازی در نظر گرفتیم مطابق جدول (۱) می‌باشد.

جدول ۱: پارامترهای شبیه‌سازی

| | |
|--------------------------|-----------------|
| پهنای باند لینک گلوگاه | ۱۰ مگابیت/ثانیه |
| تاخیر انتشار لینک گلوگاه | ۱۵۰ میلی ثانیه |
| اندازه بسته | ۱۰۰۰ بایت |
| اندازه بافر | ۱۰۰ بسته |



شکل ۱: مدل شبیه‌سازی شبکه با ۱۶ گره

پیکربندی RED، برای دستیابی به یک کارایی قابل پیش بینی، کار پیچیده و مشکلی است.

روش CBQ، زیربنای کلاسی از الگوهای زمانبندی صف است که هدف از آنها رفع محدودیت‌های مدل‌های FQ می‌باشد. این روش، محدودیت‌های مدل FQ را با پشتیبانی از جریان‌هایی با احتیاجات پهنای باند مختلف حل می‌کند. در واقع، در این روش می‌توان به هر صف، درصدی مختلفی از پهنای باند درگاه خروجی را تخصیص داد. در صف‌بندی CBQ بسته‌ها به کلاس‌های سرویس مختلفی تقسیم شده (برای مثال، بلادرنگ، انتقال فایل) و سپس به صفی که برای آن کلاس سرویس در نظر گرفته شده است، انتقال می‌یابند. هر کدام از این صف‌ها یکبار در هر دوره چرخش سرویس‌دهی شده و صف‌های خالی در نظر گرفته نمی‌شوند. به این صف‌بندی، صف‌بندی بر مبنای کلاس نیز گفته می‌شود.

SFQ^۲ تغییر یافته‌ی FQ، با هدف از بین بردن محدودیت‌های آن است؛ بطوری که این روش تعداد صف‌های مورد نیاز را کاهش می‌دهد. از مهمترین معایب این روش برخورد ناعادلانه با جریان‌هایی است که با جریان‌های دیگر تصادم می‌کنند؛ بنابراین همانطور که از نام آن برمی‌آید، عدالت بصورت غیر قطعی تضمین می‌شود.

در الگوریتم DRR^۳، زمانبند تعداد بایت‌های سر صف را تعیین می‌کند؛ اگر اندازه بسته در سر صف بزرگ‌تر از buckets (تعداد کل بایت‌هایی که صف اجازه دارد در هر بار ملاقات زمانبند، ارسال کند) باشد، buckets به اندازه واحد^۴ افزایش می‌یابد، در غیر این صورت به اندازه تعداد بایت‌های بسته، کم شده و بسته ارسال می‌گردد.

۳ - پیاده‌سازی و ارزیابی

در NS2، یکی از مهمترین مؤلفه‌های شیء Simple Link، کلاس Queue است؛ این کلاس، مکانیزم مدیریت بافر را برای مسیریاب شبکه، مدل می‌کند. به این ترتیب که بسته‌های ارسالی را در بافر ذخیره کرده و زمانی که انتقال در حال انجام کامل شد، یک بسته داخل بافر را به شیء مجاور ارسال می‌کند [۵].

¹ Class Based Queuing

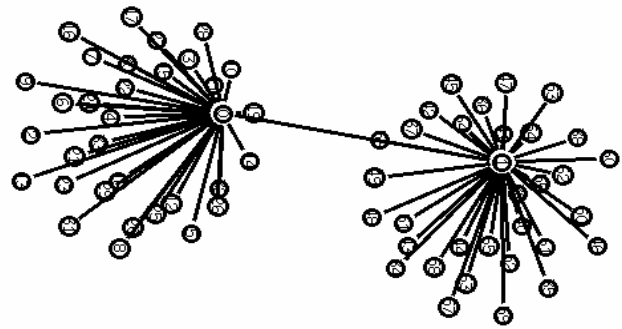
² Stochastic Fair Queuing

³ Deficit Round Robin

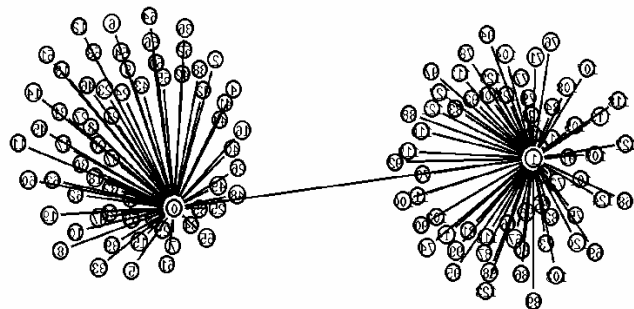
⁴ Quantum

جدول ۳: نرخ گم شدن بسته در زمان شبیه‌سازی ۱۰ ثانیه

| تعداد گره | طول صف | Drop Tail | RED | SFQ | DRR |
|--------------|-----------|--------------|------|------|-------|
| ۱۶ | ۲۵ | ۱.۶۰ | ۳.۰۶ | ۲.۱۱ | ۹۱.۵۹ |
| | ۱۰۰ | ۱.۱۵ | ۰.۸۰ | ۲.۱۱ | ۱.۵۹ |
| | ۲۵۰ | ۰.۴۷ | ۰.۸۴ | ۲.۱۱ | ۱.۵۹ |
| ۳۲ | ۲۵ | ۳.۶۲ | ۴.۴۳ | ۵.۶۵ | ۴.۴۷ |
| | ۱۰۰ | ۳.۲۳ | ۲.۷۴ | ۵.۶۵ | ۴.۴۷ |
| | ۲۵۰ | ۱.۷۵ | ۳.۰۸ | ۵.۶۵ | ۴.۴۷ |
| ۶۴ | ۲۵ | ۷.۵۵ | ۹.۰۴ | ۹.۴۴ | ۸.۵۳ |
| | ۱۰۰ | ۶.۶۷ | ۵.۶۲ | ۹.۴۴ | ۸.۵۳ |
| | ۲۵۰ | ۵.۵۹ | ۶.۴۶ | ۹.۴۴ | ۸.۵۳ |

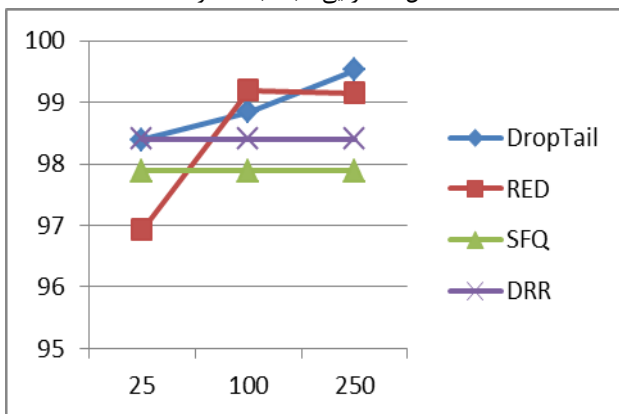


شکل ۲: مدل شبیه‌سازی شبکه با ۳۲ گره



شکل ۳: مدل شبیه‌سازی شبکه با ۶۴ گره

شکل ۴: کارایی شبکه با ۱۶ گره

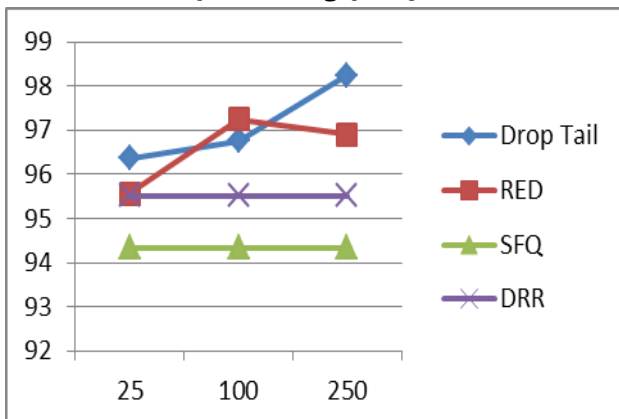


نتایج شبیه‌سازی در جدول (۱) و (۲) نشان داده شده‌است،
باتوجه به این نتایج نمودارهایی [شکل‌های (۴)، (۵) و (۶)] جهت
مقایسه این نتایج رسم نمودیم.

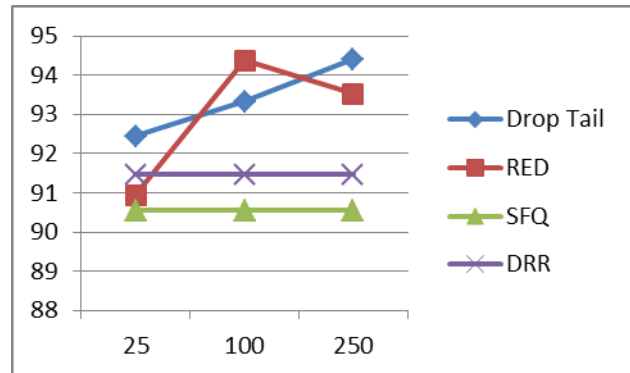
جدول ۲: کارایی شبکه در زمان شبیه‌سازی ۱۰ ثانیه

| تعداد گره | طول صف | Drop Tail | RED | SFQ | DRR |
|--------------|-----------|--------------|-------|-------|-------|
| ۱۶ | ۲۵ | ۹۸.۳۹ | ۹۶.۹۳ | ۹۷.۸۸ | ۹۸.۴۰ |
| | ۱۰۰ | ۹۸.۸۴ | ۹۹.۱۹ | ۹۷.۸۸ | ۹۸.۴۰ |
| | ۲۵۰ | ۹۹.۵۲ | ۹۹.۱۵ | ۹۷.۸۸ | ۹۸.۴۰ |
| ۳۲ | ۲۵ | ۹۶.۳۷ | ۹۵.۵۶ | ۹۴.۳۴ | ۹۵.۵۲ |
| | ۱۰۰ | ۹۶.۷۶ | ۹۷.۲۵ | ۹۴.۳۴ | ۹۵.۵۲ |
| | ۲۵۰ | ۹۸.۲۴ | ۹۶.۹۱ | ۹۴.۳۴ | ۹۵.۵۲ |
| ۶۴ | ۲۵ | ۹۲.۴۵ | ۹۰.۹۵ | ۹۰.۵۵ | ۹۱.۴۶ |
| | ۱۰۰ | ۹۳.۳۲۶ | ۹۴.۳۷ | ۹۰.۵۵ | ۹۱.۴۶ |
| | ۲۵۰ | ۹۴.۴۰ | ۹۳.۵۳ | ۹۰.۵۵ | ۹۱.۴۶ |

شکل ۵: کارایی شبکه با ۳۲ گره



شکل ۶: کارایی شبکه با ۶۴ گره



۶- نتیجه گیری

با بررسی نرخ ارسال و دریافت بسته‌ها (برای هر یک از جریان‌ها) در هر یک از مدل‌های صف نتایج حاصل می‌گردد که این نتایج بصورت نمودار در قسمت‌های مربوطه رسم شده‌اند.

مطابق نمودارهای رسم شده مشاهده می‌شود که با افزایش پیچیدگی شبکه و بالا رفتن ترافیک بطور کلی کارایی شبکه کاهش می‌یابد ولی صف Drop-Tail در هر سه سناریو بالاترین کارایی را داشته و همان‌طور که پیش‌بینی می‌شد با افزایش اندازه بفر کارایی این الگوریتم نیز افزایش می‌یابد و نرخ گم شدن بسته کاهش یافته، ولی همان‌طور که می‌دانیم چون این الگوریتم بصورت FIFO عمل می‌کند از نظر برقراری عدالت بین جریان‌ها مناسب نمی‌باشد.

افزایش اندازه بفر تاثیر چندانی بر کارایی الگوریتم‌های SFQ و DRR نداشته است، مطابق نمودارهای رسم شده کارایی الگوریتم RED نیز به اندازه بفر وابسته است. بهره‌وری RED بسیار حساس به انتخاب پارامترهای آن است. انتخاب نادرست پارامترهای RED، حتی ممکن است به بهره‌وری حتی پایین‌تر از نیز منجر شود.

الگوریتم SFQ نسبت به بقیه الگوریتم‌ها پایین‌ترین کارایی را دارد ولی می‌دانیم که از نظر برقراری عدالت بین جریان‌ها SFQ بهترین کارایی را دارد.

سپاسگزاری

این مقاله با حمایت مرکز تحقیقات مخابرات ایران انجام شده است.

مراجع

- [1] S. Floyd & V. Jacobson, "Random early detection gateways for congestion avoidance", *IEEE-ACM Transactions on Networking*, pp. 397-413, August 1993.
- [۲] حسین پدram، احسان ملکیان، علی‌رضا زارع‌پور، "شبکه‌های کامپیوتری"، انتشارات نص، چاپ پنجم، ۱۳۸۴.
- [3] Demers, A., Keshav, S. and Shenker, S., "Analysis and Simulation of a Fair Queueing Algorithm", *Journal of Internetworking Research and Experience*, September 1990, 3-26; also *In Proceedings of the ACM SIGCOMM '89*, pp 1-12, Sept. 1989.
- [4] G.Thiruchelvi, Raja. "A Survey On Active Queue Management Mechanisms". In *IJCSNS*, Vol.8, No.12, pp.130-145, Dec.2008.
- [5] Teerawat Issariyakul and Ekram Hossain, "Introduction to Network Simulator NS2", Springer,2009.