

الحمد لله الذي  
خلقنا من  
الحمم



دانشگاه تربیت مدرس  
دانشکده مهندسی برق و کامپیوتر

پایان نامه دوره دکتری مهندسی برق - کنترل

# کنترل ترافیک برای تضمین کیفیت سرویس (QoS) در شبکه‌های با سرویس متمایز

مهدی علینقی زاده اردستانی

استاد راهنما:

دکتر محمد تقی حمیدی بهشتی

استاد مشاور:

دکتر سعید جلیلی

زمستان 1388

## تشکر و قدردانی

پس از حمد و ثنای الهی بر خود لازم می‌دانم از زحمات و راهنمایی‌های استاد گرامی جناب آقای دکتر بهشتی صمیمانه تشکر و قدردانی نمایم.

همچنین از پدر و مادر عزیزم سپاسگزارم و از خداوند می‌خواهم همانطور که آنان مرا در کودکی مورد لطف و عنایت خود قرار دادند ایشان را در سایه رحمت و مغفرت خویش قرار دهد.

مهدی علینقی‌زاده اردستانی

زمستان 1388

## چکیده

پدیده تراکم یکی از مشکلات مهم پیش روی طراحان شبکه‌های کامپیوتری است و در سالهای گذشته باعث کاهش کارایی شبکه‌ها شده است. این کاهش کارایی به خصوص در دوره فعلی که بایستی کیفیت سرویس برای برخی مشتریان تضمین شود نمود بیشتری دارد. تاکنون روش‌های مختلفی به منظور پیشگیری و کنترل این پدیده پیشنهاد شده‌اند که بیشتر روش‌هایی شهودی هستند. در کنار این روش‌ها، از اواخر دهه 1990 میلادی، ایده به کارگیری مفاهیم نظریه کنترل در حل مسائل کنترل ازدحام مطرح شده که هدف آن استفاده از ابزارهای مهندسی کنترل در تحلیل و طراحی کنترلگرهای تراکم مناسب برای شبکه‌ها به عنوان سیستم‌های حلقه بسته است. با نگرش سیستمی و استفاده از نظریه کنترل می‌توان ساختار حلقه بسته فرایند انتقال داده‌ها در شبکه‌های کامپیوتری را بدین صورت در نظر گرفت که در آن کنترلگر به منظور پیاده‌سازی در مسیریاب‌های شبکه طراحی شده و بقیه شبکه از دید مسیریاب، به عنوان سیستم هدف کنترل تعریف شود. این پایان‌نامه پس از طرح مسئله کنترل ازدحام در شبکه‌های با سرویس متمایز، به طراحی کنترلگر مقاوم برای ترافیک شبکه با سرویس متمایز با در نظر گرفتن تاخیر زمانی می‌پردازد.

در این پایان‌نامه با طراحی کنترلگر مقاوم برای کنترل ازدحام در شبکه‌های با سرویس متمایز، از مزایای کنترلگر مقاوم مانند حساسیت کم به تغییرات پارامترهای شبکه و همچنین تضعیف اغتشاش اعمالی به شبکه استفاده شده است که این دو مساله به خصوص با توجه به اینکه پارامترهای شبکه پیوسته در حال تغییر بوده و در عمل بسیاری از جریان‌های داده از منطق کنترل تراکم یکسانی پیروی نمیکنند و در عمل مانند اغتشاش عمل می‌کنند، بسیار مهم است. برای سرویس‌های با کیفیت بالا یک کنترلگر و برای سرویس‌های معمولی نیز دو کنترلگر پیشنهاد شده است که کارایی آنها طی سناریوهای متعددی ارزیابی شده است. جهت طراحی برخی کنترلگرها از نامساوی‌های ماتریسی خطی استفاده شده است تا حل مسائل پیچیده امکان پذیر باشد.

با توجه به نیاز شبکه‌های نسل آینده به استفاده از فیدبک غیر صریح در سرویس‌های معمولی و پیشنهاد بیشتر تحقیقات مشابه گذشته به این امر، دو نوع کنترلگر، کنترلگر مقاوم زمان گسسته (RDTC) و کنترلگر مقاوم مبتنی بر تاخیر (RDBC) با فیدبک غیر صریح جهت سرویس‌های معمولی طراحی شده است. همچنین در این پایان‌نامه با در نظر گرفتن تاخیر انتقال در شبکه، کنترلگر برای دینامیک شبکه با وجود تاخیر طراحی شد.

شبیه‌سازی‌های متعدد در شبیه‌ساز شبکه ns2 نشان داد که کنترلگرهای طراحی شده برای فیدبک غیر صریح از نظر مقاومت در برابر تغییرات تاخیر شبکه، تعقیب طول صف مرجع، استفاده حداکثر از پهنای باند، کاهش تاخیر انتها به انتها و بهره شبکه از کنترلگرهای کنونی و متداول بهتر عمل می‌کنند.

**کلید واژه:** کنترل مقاوم، تاخیر زمانی، تضعیف اغتشاش، شبکه با سرویسهای متمایز.

## فهرست مطالب

| صفحه    | عنوان  |
|---------|--|
| د.....  | فهرست شکل ها.....  |
| 2.....  | <b>فصل 1- مقدمه</b> .....  |
| 2.....  | 1-1- پیشگفتار.....   |
| 2.....  | 2-1- تاریخچه.....  |
| 3.....  | 3-1- هدف از انجام تحقیق.....   |
| 4.....  | 4-1- نوآوری تحقیق.....   |
| 5.....  | 5-1- ساختار گزارش.....   |
| 7.....  | <b>فصل 2- مفاهیم کنترل ازدحام</b> .....                                |
| 7.....  | 1-2- مقدمه.....  |
| 7.....  | 2-2- ازدحام.....   |
| 7.....  | 1-2-2- عوامل بوجود آورنده ازدحام.....                                  |
| 8.....  | 2-2-2- کنترل ازدحام.....   |
| 9.....  | 3-2-2- روش های تشخیص ازدحام.....                                       |
| 9.....  | 4-2-2- کنترل ازدحام در پروتکل TCP.....                                 |
| 10..... | 1-4-2-2- کنترل ازدحام TCP در سمت منابع.....                            |
| 12..... | 2-4-2-2- کنترل ازدحام TCP در سمت مسیر یاب گلوگاه شبکه.....             |
| 13..... | 3-2- مدل TCP و تحلیل پایداری.....                                      |
| 14..... | 1-3-2- مساله کنترل AQM.....  |
| 14..... | 2-3-2- قوانین کنترل پایداری در AQM.....                                |
| 15..... | 4-2- کیفیت خدمات.....  |
| 15..... | 1-4-2- نیازها.....   |
| 16..... | 2-4-2- راه کارهای تامین کیفیت خدمات.....                               |
| 16..... | 3-4-2- خدمات مجتمع.....  |
| 17..... | 4-4-2- خدمات متمایز.....   |
| 17..... | 5-2- مدل جریان سیال.....   |
| 19..... | 6-2- استراتژی های کنترلی برای سرویسهای متمایز.....                     |
| 20..... | 1-6-2- استراتژی کنترل سرویس با کیفیت بالا.....                         |
| 21..... | 2-6-2- استراتژی کنترل سرویس معمولی.....                                |
| 22..... | 3-6-2- استراتژی کنترل سرویس بهترین تلاش.....                           |
| 24..... | <b>فصل 3- تاریخچه پژوهش در مورد کنترل ازدحام سرویسهای متمایز</b> ..... |

|    |  |
|----|--|
| 24 | 1-3- مقدمه   |
| 25 | 2-3- کنترلگرهای ازدحام برای شبکه سرویس متمایز                            |
| 25 | 1-2-3- کنترلگر غیرخطی تطبیقی   |
| 27 | 2-2-3- کنترلگر غیرخطی مبتنی به بازگشت به عقب                             |
| 29 | 3-2-3- کنترلگر مقاوم   |
| 31 | 4-2-3- کنترلگر مد لغزشی  |
| 32 | 5-2-3- کنترلگر مد لغزشی مرتبه دوم  |
| 34 | 6-2-3- کنترلگر مقاوم توزیع شده   |
| 36 | 3-3- بررسی اجمالی روش‌های AQM  |
| 37 | 1-3-3- روش‌های ابتکاری   |
| 37 | 2-3-3- روش‌های مبتنی بر نظریه کنترل                                      |
| 39 | 3-3-3- روش‌های مبتنی بر بهینه سازی                                       |
| 40 | 4-3-3- روش‌های ترکیبی  |
| 40 | 4-3- نتیجه گیری  |
| 43 | فصل 4- طراحی کنترلگر مقاوم سرویسهای متمایز                               |
| 43 | 1-4- طراحی کنترلگر مقاوم برای سرویس با کیفیت بالا                        |
| 43 | 1-1-4- مقدمه   |
| 45 | 2-1-4- طراحی کنترلگر   |
| 46 | 3-1-4- تحلیل پایداری کنترلگر   |
| 47 | 2-4- طراحی کنترلگر مقاوم برای سرویس معمولی                               |
| 48 | 1-2-4- روش اول - فیدبک TCP با تاخیر ثابت                                 |
| 48 | 1-1-2-4- مقدمه   |
| 50 | 2-1-2-4- طراحی کنترلگر   |
| 52 | 3-1-2-4- تحلیل پایداری سیستم کنترل صف طراحی شده RDTC                     |
| 53 | 4-1-2-4- آنالیز حساسیت   |
| 55 | 5-1-2-4- پیاده سازی کنترلگر پیشنهادی در FPGA و تجزیه و تحلیل زمانبندی‌ها |
| 58 | 2-2-4- روش دوم - فیدبک TCP با در نظر گرفتن تاخیر                         |
| 58 | 1-2-2-4- کنترلگر فیدبک حالت  |
| 59 | 2-2-2-4- کنترلگر فیدبک خروجی   |
| 61 | 3-2-2-4- طراحی کنترلگر   |
| 63 | 4-2-2-4- تحلیل پایداری   |
| 65 | 5-2-2-4- آنالیز حساسیت   |
| 66 | 6-2-2-4- پیاده سازی کنترلگر RDTC در FPGA و تجزیه و تحلیل زمانبندی‌ها     |
| 69 | فصل 5- بررسی نتایج شبیه سازی   |
| 69 | 1-5- شبیه سازی کنترلگر مقاوم سرویس با کیفیت بالا                         |

|    |   |              |
|----|---|--------------|
| 69 | سناریوی اول - تنظیم طول صف                        | 5-1-1        |
| 71 | سناریوی دوم - تعقیب طول صف                        | 5-1-2        |
| 72 | شبیه سازی کنترلگر مقاوم سرویس معمولی              | 5-2          |
| 72 | شبیه سازی کنترلگر فیدبک TCP با تاخیر ثابت         | 5-2-1        |
| 86 | شبیه سازی کنترلگر فیدبک TCP با در نظر گرفتن تاخیر | 5-2-2        |
| 92 | <b>نتیجه گیری و پیشنهادات</b>                     | <b>فصل 6</b> |
| 92 | نتیجه گیری  | 6-1          |
| 93 | پیشنهادات   | 6-2          |
| 95 | <b>فهرست مراجع</b>                                |              |

## فهرست شکل ها

| صفحه | عنوان  |
|------|--|
| 13   | شکل 1-2 - دینامیک خطی شده TCP.....   |
| 14   | شکل 2-2 - سیستم کنترل AQM.....   |
| 15   | شکل 3-2 - سیستم حلقه بسته کنترل AQM.....   |
| 18   | شکل 4-2 - مدل جریان سیال.....  |
| 19   | شکل 5-2 - مقایسه مدل جریان سیال با شبیه سازی OPNET.....                                      |
| 20   | شکل 6-2 - استراتژی کنترلی در هر پورت خروجی.....  |
| 28   | شکل 1-3 - طول صف و سیگنال کنترلی برای سرویس با اولویت بالا.....                              |
| 28   | شکل 2-3 - طول صف و سیگنال کنترلی برای سرویس معمولی.....                                      |
| 30   | شکل 3-3 - مدل خطی برای طراحی کنترلگر سرویس با کیفیت بالا.....                                |
| 30   | شکل 4-3 - بلوک دیاگرام کنترلگر ترافیک معمولی.....  |
| 32   | شکل 5-3 - تاثیر تاخیر بر روی عملکرد کنترلگرهای با کیفیت بالا و معمولی [43].....              |
| 33   | شکل 6-3 - عملکرد کنترلگرهای با کیفیت بالا و معمولی [45].....                                 |
| 34   | شکل 7-3 - شبکه $n \times n$ گره پیشنهادی.....  |
| 34   | شکل 8-3 - استراتژی کنترل در هر یک از گره ها.....   |
| 36   | شکل 9-3 - مشخصات طول صف 5 گره پس از اعمال استراتژی کنترل.....                                |
| 38   | شکل 10-3 - مدل کنترل فیدبک کنترل تراکم با استفاده از AQM.....                                |
| 44   | شکل 1-4 - ساختار کنترلگر مقاوم.....  |
| 47   | شکل 2-4 - نمودار بود تابع حلقه باز سرویس با اولویت بالا.....                                 |
| 52   | شکل 3-4 - مکان هندسی قطب های $K(s)$ .....  |
| 53   | 4-4 - نمودار بود تابع تبدیل حلقه باز RDTC.....   |
| 53   | شکل 5-4 - نمودار بود $ \Delta(jw)V(jw) $ .....   |
| 54   | شکل 6-4 - نمودار نایکویست سیستم حلقه باز با تغییرات تاخیر از 240 تا 460 میلی ثانیه.....      |
| 54   | شکل 7-4 - نمودار نایکویست سیستم حلقه باز با تغییرات ظرفیت از 3000 تا 7000 بسته در ثانیه..... |
| 54   | شکل 8-4 - نمودار نایکویست سیستم حلقه باز با تغییرات N از 18 تا 88.....                       |
| 63   | شکل 9-4 - مکان هندسی قطب و صفرهای کنترلگر RDTC.....  |
| 63   | شکل 10-4 - مکان هندسی قطب و صفرهای کنترلگر زمان پیوسته RDTC.....                             |
| 64   | شکل 11-4 - نمودار بود تابع تبدیل حلقه باز کنترلگر RDTC.....                                  |
| 64   | شکل 12-4 - نمودار بود $ \Delta(jw)V(jw) $ برای کنترلگر RDTC.....                             |
| 65   | شکل 13-4 - نمودار حاشیه بهره حلقه باز سیستم نسبت به تغییرات تاخیر.....                       |
| 65   | شکل 14-4 - نمودار حاشیه بهره حلقه باز سیستم نسبت به تغییرات بار شبکه.....                    |
| 65   | شکل 15-4 - نمودار حاشیه بهره حلقه باز سیستم نسبت به تغییرات ظرفیت شبکه.....                  |
| 69   | شکل 1-5 - طول صف اولویت بالا برای ورودی پواسون با متوسط 200 و واریانس 300 بسته.....          |
| 70   | شکل 2-5 - طول صف اولویت بالا برای ورودی نرمال بین 300 تا 500.....                            |
| 70   | شکل 3-5 - طول صف اولویت بالا برای ورودی نرمال بین 300 تا 600.....                            |



|    |  |
|----|--|
| 71 | شکل 4-5- طول صف برای ورودی پواسون با میانگین 400 و واریانس 300                           |
| 71 | شکل 5-5- خطا برای ورودی پواسون با میانگین 400 و واریانس 300                              |
| 72 | شکل 6-5- توپولوژی شبیه سازی  |
| 73 | شکل 7-5- طول صف گلوگاه در تاخیر 240 میلی ثانیه   |
| 74 | شکل 8-5- پهنای باند گلوگاه در تاخیر 240 میلی ثانیه                                       |
| 75 | شکل 9-5- طول صف گلوگاه در تاخیر 160 میلی ثانیه   |
| 76 | شکل 10-5- پهنای باند گلوگاه در تاخیر 160 میلی ثانیه                                      |
| 77 | شکل 11-5- طول صف گلوگاه در تاخیر 360 میلی ثانیه  |
| 78 | شکل 12-5- پهنای باند گلوگاه در تاخیر 360 میلی ثانیه                                      |
| 79 | شکل 13-5- طول صف گلوگاه با وجود اعمال اغتشاش در تاخیر 240 میلی ثانیه                     |
| 80 | شکل 14-5- پهنای باند گلوگاه با وجود اعمال اغتشاش در تاخیر 240 میلی ثانیه                 |
| 81 | شکل 15-5- طول صف گلوگاه با وجود اعمال اغتشاش در تاخیر 360 میلی ثانیه                     |
| 82 | شکل 16-5- پهنای باند گلوگاه با وجود اعمال اغتشاش در تاخیر 360 میلی ثانیه                 |
| 83 | شکل 17-5- انحراف معیار تاخیر انتها به انتها بر حسب فرکانس نمونه برداری                   |
| 86 | شکل 18-5- طول صف و پهنای باند برای تاخیر انتها به انتها 120 میلی ثانیه                   |
| 87 | شکل 19-5- طول صف و پهنای باند برای تاخیر انتها به انتها 240 میلی ثانیه                   |
| 87 | شکل 20-5- طول صف و پهنای باند برای تاخیر انتها به انتها 340 میلی ثانیه                   |
| 87 | شکل 21-5- طول صف و پهنای باند برای تاخیر انتها به انتها 440 میلی ثانیه                   |
| 88 | شکل 22-5- طول صف و پهنای باند برای تاخیر انتها به انتها 120 میلی ثانیه با اغتشاش 1.5Mbps |
| 88 | شکل 23-5- طول صف و پهنای باند برای تاخیر انتها به انتها 240 میلی ثانیه با اغتشاش 1.5Mbps |
| 88 | شکل 24-5- طول صف و پهنای باند برای تاخیر انتها به انتها 340 میلی ثانیه با اغتشاش 1.5Mbps |
| 88 | شکل 25-5- طول صف و پهنای باند برای تاخیر انتها به انتها 440 میلی ثانیه با اغتشاش 1.5Mbps |
| 89 | شکل 26-5- طول صف و پهنای باند برای تاخیر انتها به انتها 120 میلی ثانیه با اغتشاش 3Mbps   |
| 89 | شکل 27-5- طول صف و پهنای باند برای تاخیر انتها به انتها 240 میلی ثانیه با اغتشاش 3Mbps   |
| 89 | شکل 28-5- طول صف و پهنای باند برای تاخیر انتها به انتها 340 میلی ثانیه با اغتشاش 3Mbps   |
| 90 | شکل 29-5- طول صف و پهنای باند برای تاخیر انتها به انتها 440 میلی ثانیه با اغتشاش 3Mbps   |

# فصل اول

## مقدمه

## فصل 1- مقدمه

### 1-1- پیشگفتار

با گسترش روزافزون اینترنت و تقاضا برای سرویسهای حساس به زمان مانند صوت و تصویر نیاز به طراحی و استفاده از الگوریتمهای کنترلی مناسب برای کنترل ازدحام شبکه امری ضروری به نظر می‌رسد. ازدحام در یک شبکه هنگامی رخ می‌دهد که تعداد بسته‌های عبوری از شبکه بیش از حد ظرفیت شبکه باشد. وقتی شبکه دچار ازدحام می‌شود، بسته‌هایی از بین می‌روند، پهنای باند تلف می‌شود، تاخیرهای نامعقول پدید می‌آید و به طور کلی کارایی شبکه کاهش می‌یابد.

از آنجا که همواره منابع سخت‌افزاری محدود هستند، افزایش پهنای باند و یا ظرفیت حافظه مسیریاب‌ها در هنگام وقوع ازدحام جوابگو نمی‌باشند. اما با کنترل دینامیکی ازدحام شبکه می‌توان کیفیت سرویس را تا حد قابل قبولی بالا برد.

### 1-2- تاریخچه

ساختار سرویس‌های متمایز برای بالا بردن کیفیت سرویس در شبکه‌های IP<sup>1</sup> در سال 1998 توسط Blake پیشنهاد شد. وظیفه اصلی سرویس‌های متمایز استانداردسازی مجموعه ساده‌ای از مکانیزم‌ها برای کار با بسته‌های اطلاعاتی است که دارای اولویت‌های مختلفی هستند. این اولویت‌ها توسط یک DS Field در هدر IP هر بسته مشخص می‌شوند. ذکر این نکته لازم است که در سرویس‌های متمایز، دسته‌بندی بسته‌های دیتا در لبه‌های شبکه انجام می‌شود تا از پیچیدگی عملکردی در هسته شبکه کاسته شده و مقیاس عملکردی بهتری ارائه دهد.

از طرف دیگر هیچ اندازه‌گیری مشخصی برای اطمینان از این که اولویت‌ها بر اساس کیفیت سرویس مطلوب به بسته‌ها اختصاص داده می‌شود وجود ندارد. بنابراین ساختار سرویس‌های متمایز استاندارد تنها یک کیفیت سرویس ابتدایی و ناقص فراهم می‌کند و هیچ تضمین کمی در این مورد وجود ندارد. با افزایش حجم اطلاعات همزمان ارسالی توسط این سرویس‌ها نیاز به معرفی مکانیزم‌های کنترلی برای جلوگیری از ازدحام در این شبکه‌ها احساس شد. به همین خاطر مهندسان با ارائه روشهای مختلف کنترلی چه در سمت فرستنده‌ها و چه در قسمت روترها<sup>2</sup> سعی در کاهش بار ترافیکی این سرویس‌ها داشته‌اند.

<sup>1</sup> Internet Protocol

<sup>2</sup> Routers

در سال‌های اخیر به منظور بهبود الگوریتم‌های مدیریت فعال صف<sup>1</sup> (AQM) [1]، روش‌های کنترل کلاسیک مانند کنترلگرهای PI نیز بکار گرفته شده‌اند که به منظور تعیین پارامترهای این کنترلگرها از مدل ریاضی شبکه استفاده شده است.

همانگونه که مشاهده می‌شود، همه الگوریتم‌های کنترل ازدحام مطرح شده در اینترنت با بسته‌های داده و ترافیک شبکه به صورت یکسان برخورد کرده و اولویت خاصی برای کاربردهای مختلف قائل نمی‌شوند. در صورتی که سرویس‌های متمایز جهت تضمین کیفیت خود به ویژگی‌های مختلفی نیاز دارند. به عنوان مثال یک فایل داده‌ای اطلاعاتی نیازمند رسیدن تمامی بسته‌های آن به مقصد به صورت صحیح می‌باشد (بدون مطرح بودن تاخیر) و برای بسته‌های یک مکالمه‌ی صوتی مهمترین مسئله رسیدن بسته‌ها با کمترین تاخیر است (حتی اگر بسته‌ای/بسته‌هایی در میان هزاران بسته صوتی از بین برود). البته پروتکل‌هایی در زمینه‌ی پشتیبانی از سرویس‌های متمایز در اینترنت وجود دارد که از آن جمله می‌توان به IntServ و DiffServ اشاره نمود.

### 1-3- هدف از انجام تحقیق

با توجه به اینکه امروزه خدمات بسیار زیادی بر بستر شبکه‌های کامپیوتری ارائه می‌گردد، تضمین کیفیت سرویس در شبکه‌های کامپیوتری اهمیت بسیار زیادی پیدا کرده است. ارائه خدماتی چون انتقال صوت، انتقال فایل، گپ<sup>2</sup>، ویدئو کنفرانس، کنترل ربات از راه دور و بسیاری خدمات دیگر هر یک الزامات خاص خود را برای تضمین کیفیت می‌طلبند که در نتیجه طراحان شبکه‌ها را به خود مشغول داشته است. بر اساس این موضوع، طراحی شبکه‌هایی با راندمان بالا و کیفیت تضمین شده برای هر سرویس از بخش‌های جذاب تحقیقاتی در این چند سال تبدیل شده است و پیش‌بینی می‌شود که این روند ادامه داشته باشد.

در سال‌های اخیر تلاش‌های فراوانی برای طراحی یک کنترلگر جامع برای سرویس‌های متمایز به منظور بالابردن کیفیت سرویس انجام پذیرفته است. هر یک از این طرح‌ها گرچه موجب بهبود عملکرد سرویس شده‌اند اما به نوبه خود دارای ضعف‌هایی نیز بوده‌اند که اکنون نیز پژوهشگران برآنند تا با ارائه راهکارهای جدید بر این مشکلات فائق آیند.

از جمله ضعف‌هایی که در اکثر کنترلگرهای ارایه شده مشاهده می‌شود عدم بررسی موضوع تاخیر در سرویس‌های شبکه می‌باشد. حال آنکه وجود تاخیر در شبکه‌های کامپیوتری امری اجتناب ناپذیر است. گاه نگارندگان مقالات بدون در نظر گرفتن تاخیر به طراحی کنترلگر پرداخته‌اند و تنها به آن بسنده کرده‌اند تا با ارائه نتایج شبیه‌سازی، قوام کنترلگر را در حضور تاخیر نشان دهند.

<sup>1</sup> Active Queue Management

<sup>2</sup> Chat

در این تحقیق وضعیت ازدحام و معیارهای کیفیت سرویس در مدل‌های ترافیکی جدید بررسی شده و الگوریتمی مناسب برای کنترل ازدحام در شبکه‌هایی با سرویس‌های متمایز ارائه می‌گردد. هرچند کنترل ازدحام تنها معیار تضمین کیفیت سرویس نمی‌باشد، اما به عنوان مرتبط‌ترین بخش با مباحث مهندسی کنترل در این تحقیق به آن خواهیم پرداخت.

در این پژوهش سعی بر آن است تا با در نظر گرفتن پدیده تاخیر در هر یک از سرویس‌های متمایز به طراحی کنترلگر مقاوم برای تنظیم طول صف در این سرویس‌ها پردازیم. به نظر می‌رسد طراحی کنترلگر مقاوم با در نظر گرفتن تاخیر گامی مثبت برای رسیدن به یک کنترلگر ایده‌آل برای تحقق عملی جلوگیری از ازدحام در شبکه‌های با سرویس متمایز باشد که بتواند کیفیت سرویس را در سطح مطلوبی تضمین نماید.

#### 1-4- نوآوری تحقیق

در پژوهش انجام شده با استفاده از روش کنترل مقاوم روش جدیدی برای طراحی کنترلگر شبکه با در نظر گرفتن تاخیر زمانی معرفی شده است. در نظر گرفتن تاخیر و حل مساله تاخیر زمانی شبکه موجب شده است تا بتوان حساسیت شبکه را، نسبت به تغییرات تاخیر در شبکه که همواره وجود دارد، کاهش داد. هرچند در معدود تحقیقات گذشته، استفاده از کنترل مقاوم برای طراحی شبکه مبتنی بر تاخیر نیز انجام شده بود، اما نتایج این تحقیق نشان می‌دهد که رویکرد پژوهش پیش رو بسیار موثرتر، جامع‌تر و کاراتر از آنهاست.

همچنین با توجه به اینکه بسیاری از جریان‌های داده در شبکه‌ها، از پروتکل‌های کنترل تراکم مورد نظر ما تبعیت نمی‌کنند، این جریانات به صورت اغتشاشاتی دائمی به ورودی شبکه اعمال می‌شوند. برای حل این معضل نیز یکی از اهداف اصلی طراحی کنترلگر مقاوم، تضعیف اثر اغتشاش در نظر گرفته شده است. نتایج تحقیق نیز نشان می‌دهد که اغتشاشات ورودی به شبکه نیز به صورت قابل ملاحظه‌ای تضعیف شده‌اند. تمرکز بر تضعیف اغتشاش در طراحی کنترلگر تراکم شبکه برای اولین بار انجام پذیرفته است و با توجه به وجود ترافیک‌های مختلف که در شبکه وجود دارند و بسیاری از آنها از الگوی کنترل تراکم مسیر یاب استفاده نمی‌کنند، این نوع طراحی در آینده بیشتر مدنظر خواهد بود.

در این تحقیق با استفاده از نامساوی‌های ماتریسی خطی مسائل پیچیده طراحی کنترلگر مقاوم حل گشت که ایده استفاده از نامساوی‌های ماتریسی خطی در حل کنترلگرهای ازدحام در شبکه توانست کمک شایانی در حل مساله نماید.

## 1-5- ساختار گزارش

در این رساله ابتدا با معرفی شبکه‌های کامپیوتری و پروتکل‌های حاکم بر آن، به بررسی الگوریتم‌ها و اصول کلی حاکم بر کنترل ازدحام پرداخته می‌شود. همچنین به معرفی انواع مکانیزم‌های کنترل ترافیک در شبکه خواهیم پرداخت.

سپس با بررسی جدیدترین طرح‌های مطرح شده برای کنترل ازدحام در سرویس‌های متمایز، نقاط ضعف و قوت هر یک از آنها را مورد بررسی قرار خواهیم داد.

در فصل چهارم کنترلگر مقاوم پیشنهادی برای ترافیک سرویس‌های با کیفیت بالا<sup>1</sup> و سرویس‌های معمولی<sup>2</sup> مطرح می‌شود و در نهایت در فصل پنجم با ارائه نتایج شبیه‌سازی نشان می‌دهیم که این کنترلگرها به خوبی می‌توانند طول صف این سرویس‌ها را با وجود تاخیر، بر روی مقدار مرجع تنظیم نمایند. در فصل پایانی نیز با جمع‌بندی نتایج این پژوهش اشاره‌ای به فعالیت‌های باز تحقیقاتی در این زمینه خواهیم داشت.

---

<sup>1</sup> Premium Services

<sup>2</sup> Ordinary Services

## فصل دوم

### مفاهیم کنترل ازدحام

## فصل 2- مفاهیم کنترل ازدحام

### 2-1- مقدمه

در این فصل راجع به مفاهیم ازدحام در شبکه‌ها مطالب کوتاهی ارائه شده و سپس انواع الگوریتم‌های کنترل ازدحام در نقاط مختلف شبکه معرفی می‌شوند. همچنین در ادامه مفاهیم کیفیت خدمات و سرویسهای متمایز بیان می‌شود.

### 2-2- ازدحام

وقتی به بخشی از زیر شبکه، تعداد بسیار زیادی بسته تحویل داده شود، کارایی آن کاهش می‌یابد. بدین وضعیت ازدحام گفته می‌شود. هرگاه تعداد بسته‌هایی که توسط ماشین‌های میزبان به شبکه وارد می‌شوند متناسب با ظرفیت حمل شبکه باشند، تمام این بسته‌ها به استثنای آن‌هایی که در اثر خطای انتقال آسیب می‌بینند، تحویل مقصدشان خواهند شد و تعداد بسته‌های تحویلی متناسب با تعداد بسته‌های ارسالی است. ولی به محض افزایش بی‌رویه ورود داده‌ها به شبکه، مسیریاب‌ها قادر نیستند از عهده سرویس‌دهی به آن‌ها برآمده و در این میان تعدادی بسته از دست می‌روند. هر چه ترافیک شبکه زیادتر شود، این مساله حادثتر می‌شود و در ترافیک بسیار بالا، شبکه کارایی خود را به طور کامل از دست می‌دهد و هیچ بسته‌ای به مقصد نخواهد رسید.

### 2-2-1- عوامل بوجود آورنده ازدحام

عوامل مختلفی می‌توانند باعث بوجود آمدن ازدحام در شبکه‌ها شوند که در ادامه به بررسی بعضی از آن‌ها پرداخته می‌شود:

#### 2-2-1-1- عبور چند بسته ورودی از یک خط خروجی:

اگر از چند خط ورودی به طور همزمان اطلاعاتی به مسیریاب وارد شوند که تمامی‌شان بخواهند به یک خط خروجی بروند، صفی از اطلاعات بوجود خواهد آمد. اگر برای ذخیره آن‌ها حافظه کافی در اختیار مسیریاب نباشد، این بسته‌ها از بین می‌روند و فرستنده باید دوباره آن‌ها را ارسال کند که باعث ایجاد ازدحام و کاهش سرعت انتقال اطلاعات می‌شود.

اگر حافظه مسیریاب‌ها را زیادتر کنیم، در این صورت طول صف را زیادتر کرده‌ایم و این کار باعث می‌شود در حالی که بسته‌ها در صف منتظر ارسال هستند، مهلت رسیدنشان به مقصد نیز تمام شود و فرستنده به دلیل عدم دریافت سیگنال تصدیق، بسته‌ها را دوباره ارسال کند. تمام این بسته‌های تکراری نیز دوباره در



صف‌های ارسال در مسیر یاب‌ها قرار می‌گیرند و باعث افزایش هرچه بیشتر بار شبکه می‌شوند و ازدحام افزایش می‌یابد.

## 2-1-2-2- پردازنده‌های کم‌سرعت:

مسیر یاب‌های دارای پردازشگر کند نیز باعث بوجود آمدن ازدحام می‌شوند. اگر مسیر یاب‌ها در اجرای اموری مانند صف‌بندی بافرها، تازه‌سازی جداول مسیریابی و غیره کند باشند، با وجود بالا بودن ظرفیت حمل خطوط، صف‌هایی تشکیل خواهد شد.

## 2-1-2-3- پهنای باند کم خطوط انتقال:

خطوط ارتباطی شبکه نیز در صورتی که پهنای باند پایینی نسبت به سایر اجزای شبکه داشته باشند می‌توانند باعث ایجاد ازدحام در شبکه شوند.

در کل به مناطقی از مسیر که باعث ایجاد ازدحام می‌شوند، گلوگاه<sup>1</sup> می‌گویند. باید به این نکته توجه کرد که نمی‌توان تنها با اصلاح یک قسمت از شبکه، سرعت انتقال اطلاعات را بالا برد زیرا این کار فقط باعث جابجایی گلوگاه‌های شبکه می‌شود. بلکه باید بین قطعات مختلف سیستم توازن ایجاد کرد. یک موقع گیرنده قادر به تحمل بار ارسال شده نیست و یک موقع شبکه نمی‌تواند این بار را تحمل کند. این دو مقوله کاملاً متفاوت از یکدیگر می‌باشند.

## 2-2-2- کنترل ازدحام

اگر بافرهای آزاد در اختیار مسیر یاب نباشد از بسته‌های تازه صرف نظر می‌کند. با این کار فرستنده آن‌ها را دوباره ارسال می‌کند. این کار شاید چندین بار اتفاق افتد و وضع را بدتر کند. از آنجا که فرستنده تا از رسیدن یک بسته به مقصد مطمئن نشود، نمی‌تواند آن را از بافر پاک کند، ازدحام در سمت گیرنده باعث ازدحام در سمت فرستنده نیز می‌شود. همچنین اگر یک فرستنده اطلاعات را سریع‌تر از توان دریافت گیرنده ارسال کند، باعث غرق شدن گیرنده در دریای اطلاعات می‌شود.

بسیاری از مشکلات و مسایل سیستم‌های پیچیده مثل شبکه‌های کامپیوتری را می‌توان از دیدگاه نظریه کنترل بررسی کرد. از این دیدگاه راه حل‌ها به دو گروه تقسیم می‌شوند: حلقه باز و حلقه بسته. راه حل‌های حلقه باز سعی می‌کنند مساله را با طراحی خوب حل کرده و از همان ابتدا اطمینان دهند که مشکلی رخ نخواهد داد و وقتی سیستم شروع به کار و انجام فعالیت نمود، هیچگونه نظارت یا تصحیح عملکردی، ممکن نیست. تمهیداتی که برای کنترل ازدحام به روش حلقه باز پیش‌بینی می‌شوند عبارتند از: تصمیم‌گیری در

<sup>1</sup> Bottleneck

خصوص زمان پذیرش ترافیک جدید، تصمیم گیری در خصوص زمان حذف بسته‌ها، انتخاب بسته‌هایی که باید حذف شوند و تصمیم گیری در خصوص زمان بندی صحیح در شبکه. حقیقت مشترک در تمام این موارد آن است که تصمیم گیری‌ها مبتنی بر شرایط جاری شبکه نیست. در مقابل، راه حل‌های حلقه بسته مبتنی بر مفهوم حلقه‌های فیدبک هستند. این گونه راه کارهای کنترل ازدحام، سه بخش را در بر می گیرند:

1- نظارت بر سیستم به منظور تشخیص آنکه در کجا و در چه وقت ازدحام رخ داده است.

2- تحویل این اطلاعات به محلی که بایستی واکنش نشان بدهد.

3- تنظیم عملکرد سیستم برای رفع مشکل.

## 2-2-3- روش‌های تشخیص ازدحام

برای نظارت و تشخیص بروز ازدحام در زیر شبکه می توان معیارهای گوناگونی را به کار گرفت. مهمترین آن‌ها عبارتند از: درصد بسته‌هایی که به دلیل کمبود فضای بافر حذف می‌شوند، متوسط طول صف، تعداد بسته‌هایی که مهلت ارسال آن‌ها تمام شده است و از نو ارسال شده‌اند، متوسط تاخیر رسیدن بسته‌ها و انحراف معیار تاخیر بسته‌ها. در تمام این معیارها رشد اعداد نمایانگر افزایش ازدحام است.

دومین مرحله از حلقه فیدبک آن است که اطلاعاتی در خصوص ازدحام، از محل تشخیص و بروز آن به محلی که می توان کاری برای حل آن انجام شود، ارسال شود. بدیهی ترین کار این است که مسیریاب کشف کننده ازدحام بسته خاصی را برای ماشین یا ماشین‌های مبدا بفرستد و مشکل را به آن‌ها اعلام نماید که البته این بسته‌های اضافی خودشان به بار شبکه می افزایند.

با این وجود روش‌های دیگری نیز امکان پذیر است. به عنوان مثال می توان یک فیلد در هر بسته تعبیه شود تا هرگاه ازدحام از یک حد آستانه فراتر رفت، مسیریاب آن را علامت گذاری کند. هرگاه مسیریاب تشخیص بدهد که ازدحام رخ داده، در تمام بسته‌های خروجی این فیلد را علامت گذاری می کند تا به همسایه‌های خود در این خصوص هشدار دهد.

راه کار دیگر آن است که ماشین‌های میزبان و یا مسیریاب‌ها به طور متناوب بسته‌های سوال تولید و ارسال کنند و مستقیماً در خصوص ازدحام کسب آگاهی کنند. با کمک این اطلاعات می توان بسته‌ها را طوری مسیریابی کرد که از ناحیه بروز مشکل، عبور نکنند.

در تمام روش‌های مبتنی بر فیدبک، انتظار می رود که آگاهی از ازدحام موجب شود ماشین‌های میزبان از خود واکنش مناسبی نشان دهند. برای آن که این روش‌ها به درستی عمل کنند، مقیاس زمان بندی باید به درستی تنظیم شده باشد. لیکن بدست آوردن ثابت زمانی سیستم مسئله ساده‌ای نیست.

## 2-2-4- کنترل ازدحام در پروتکل TCP

شبکه‌های TCP/IP که شبکه جهانی اینترنت نیز از آن نوع است، در لایه انتقال دو پروتکل مهم دارند، یکی اتصال گرا و دیگری بی اتصال. پروتکل بی اتصال UDP بین مبدا و مقصد، اتصالی برقرار نمی کند و یک

پروتکل نامطمئن می‌باشد که از آن بیشتر برای انتقال صوت و تصویر استفاده می‌گردد. برای سایر کاربردها جهت انتقال اطلاعات از طریق اینترنت از پروتکل مطمئن و اتصال گرای TCP استفاده می‌گردد. از آنجا که انتقال اطلاعات در اینترنت غالباً از طریق پروتکل TCP انجام می‌گیرد و بار اصلی انتقال اطلاعات اینترنتی را بسته‌های TCP برعهده دارند، لذا کنترل ازدحام در TCP از اهمیت ویژه‌ای برخوردار است. جهت جلوگیری از ازدحام در TCP دو سیاست در نظر گرفته می‌شود: یکی در سمت منابع و دیگری در سمت مسیریاب‌های گلوگاه شبکه.

## 2-2-4-1- کنترل ازدحام TCP در سمت منابع

در انتقال اطلاعات از طریق اینترنت، هنگام وقوع ازدحام اگرچه لایه شبکه نیز در رفع ازدحام کمک می‌کند، اما بیشترین ازدحام را TCP رفع می‌کند. از نظر تئوری می‌توان ازدحام را به این صورت رفع نمود که تا هنگامی که بسته قبلی از شبکه خارج نشده است، بسته جدیدی وارد شبکه نشود. TCP سعی می‌کند با تغییر پویای اندازه پنجره این کار را انجام دهد. قبل از پرداختن به چگونگی واکنش TCP به ازدحام، عملکرد آن را در جلوگیری از وقوع ازدحام تشریح می‌کنیم. وقتی یک اتصال TCP برقرار می‌شود، اندازه مناسبی از پنجره ارسال باید انتخاب شود. گیرنده می‌تواند پنجره را براساس اندازه بافرش انتخاب کند. اگر فرستنده این اندازه پنجره را رعایت کند، بافر گیرنده سرریز نمی‌شود و مشکلی پیش نمی‌آید. در TCP الگوریتم‌های متعددی برای کنترل ازدحام در سمت منابع ابداع شده‌اند که در زیر به معرفی بعضی از آن‌ها خواهیم پرداخت:

### 1- الگوریتم TCP Tahoe [2]

الگوریتم TCP Tahoe در سال 1988م معرفی شد و تمامی پیاده‌سازی‌های TCP ملزم به رعایت این الگوریتم هستند. در این روش وقتی اتصالی برقرار می‌شود، فرستنده اندازه پنجره ازدحام را با اندازه توافق شده برای طول بسته که در هنگام برقراری اتصال بین فرستنده و گیرنده، بر روی آن به توافق رسیده‌اند، تنظیم می‌کند و سپس یک بسته با طول حداکثر می‌فرستد. اگر اعلام وصول این بسته قبل از انقضای مهلت مقرر دریافت شد، فرستنده اندازه پنجره ازدحام را دو برابر می‌کند و دفعه بعدی دو بسته ارسال می‌کند. مادامی که پس از ارسال هر قطعه، تصدیق دریافت آن بسته قبل از انقضای مهلت مقرر به فرستنده برسد، پنجره ازدحام برای ارسال بعدی دو برابر می‌شود. وقتی پنجره ازدحام معادل با اندازه  $n$  بسته باشد و تمام بسته‌های ارسالی سر موعد اعلام وصول شوند، به پنجره ازدحام معادل با طول  $n$  بسته اضافه خواهد شد [2].

در کل اگر در هر بار ارسال تمام بسته‌ها اعلام وصول شوند، طول پنجره ازدحام دو برابر خواهد شد. به این شروع اصلاحات شروع آهسته گفته می‌شود ولیکن هرگز عمل نمی‌کند زیرا افزایش آن نمایی است. پنجره ازدحام آنقدر افزایش می‌یابد، تا اندازه‌اش به مقداری به نام سطح آستانه برسد. مقدار اولیه این پارامتر

64 کیلوبایت است. پس از رسیدن اندازه پنجره ازدحام به سطح آستانه، دیگر پس از هر ارسال موفق، پنجره ازدحام دو برابر نمی‌شود بلکه به صورت خطی افزایش می‌یابد. یعنی بعد از هر ارسال موفق به اندازه طول یک بسته به پنجره ازدحام اضافه می‌کند. به این فاز، فاز دوری از ازدحام گفته می‌شود. این کار ادامه می‌یابد تا اینکه بسته‌ای مفقود شود. در این هنگام پارامتر آستانه به نصف مقدار آخرین پنجره ازدحام تنظیم می‌شود و سپس اندازه پنجره ازدحام برابر با اندازه یک بسته قرار داده می‌شود و فرستنده دوباره به فاز شروع آهسته برمی‌گردد.

اگر همه ارسال‌ها موفقیت‌آمیز باشد، اندازه پنجره ازدحام آنقدر رشد خواهد کرد تا آن‌که به دلیل انقضای مهلت اعلام وصول بسته‌ها، مقدار آستانه نصف شود و فرستنده به فاز شروع آهسته برگردد و یا آن‌که اندازه پنجره ازدحام به اندازه پنجره گیرنده برسد. در این نقطه رشد اندازه پنجره ازدحام متوقف شده و مادامی که طول پنجره گیرنده تغییر نکند و مهلت اعلام وصول قطعات ارسالی منقضی نشود، اندازه پنجره ازدحام ثابت خواهد ماند. در [3] و [4] سعی شده تا با استفاده از مدل ریاضی این الگوریتم، میزان بافر بهینه انتخاب شود.

## 2- الگوریتم TCP Reno [5]

در الگوریتم TCP Reno مکانیزم بازیابی سریع نسبت به الگوریتم TCP Tahoe اضافه شده است. در این روش به ازای مفقود شدن هر بسته به حالت شروع آهسته نمی‌رویم بلکه در این الگوریتم وقتی که سه اعلام وصول تکراری دریافت شد، فرستنده بسته مفقود شده را دوباره ارسال می‌کند و پنجره ازدحام را به نصف تقلیل می‌دهد و به جای رفتن به فاز شروع آهسته به فاز بازیابی می‌رود. وقتی اولین اعلام وصول غیر تکراری دریافت شود فرستنده از فاز بازیابی خارج شده و به فاز دوری از ازدحام می‌رود.

الگوریتم بازیابی سریع در TCP Reno فقط در وقتی که یک بسته از یک فرستنده مفقود می‌شود، بهینه است. زیرا هنگامی که چندین بسته از همان پنجره مفقود شوند TCP Reno برای زمان انقضای مهلت آن‌ها صبر می‌کند. در [6] نشان داده شده است که TCP Reno در حالتی که چندین بسته از یک پنجره مفقود می‌شوند بهره‌برداری کمتری از پهنای باند نسبت به TCP Tahoe دارد.

در [7] مدل اتفاقی دقیق و ساده حالت ماندگار TCP Reno ارائه شده است. این مدل به صورت تابعی از رفتار دور ریز و تاخیر رفت و برگشت بیان شده است. در [8] نیز پایداری محلی و پایداری فراگیر TCP Reno در رابطه با حالات مختلف جریان ورودی بحث شده و ناحیه پایداری مناسب بررسی شده است.

## 3- الگوریتم TCP SACK [9]