

تنظیم پارامتر چکه در ساماندهنده ترافیک توسط اتوماتای یادگیر

یونس سیفی محمد رضا میبدی

آزمایشگاه سیستم های نرم افزاری

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی امیرکبیر

چکیده: این مقاله یک مدل سطح چکه‌ای با تنظیم پارامتر چکه متغیر را ارائه نموده و اقدام به بررسی کارآبی آن در هنگام اجرای UPC در شبکه ATM می‌نماید. از آنجائیکه قرارداد اولیه صورت پذیرفته در CAC برای کاربردهای مهم دقیق نمی‌باشد و از سوی دیگر در انواع گوناگونی از منابع مانند منابع انفحاری تخمین مشخصات ترافیک مشکل و دارای پیچیدگی بسیار می‌باشد آنگونه که باید روش‌های ساماندهی شناخته شده قادر نیستند بصورتی کارآمد پارامترهای توافق شده را کنترل نمایند. لذا سعی شده است از تنظیم پویای این پارامترها استفاده شود. نتایج شبیه‌سازی‌های صورت گرفته نشان می‌دهند که روش جدید از جنبه‌های گوناگونی مانند درصد سلونیهای دوربینخته شده و بهره‌وری استفاده از منابع بهتر از نوع قدیمی سطح چکه‌ای عمل می‌نماید.

کلمات کلیدی : انتقال حالت ناهمگام، اتوماتای یادگیر، سطح چکه‌ای، ساماندهی ترافیک، کنترل پارامتر کاربری

(۱) مقدمه

یکی از راهکارهای عمده در شبکه‌های ارتباطی پیشرفتنه است و توسط CCITT بعنوان بهترین راه حل پیاده‌سازی سرویس‌های B-ISDN^۱ بیشنهاد شده است [۱]. یکی از عمده‌ترین قابلیت‌های این شبکه پشتیبانی از نوع سرویس‌ها با نرخهای گوناگون، مشخصات و کیفیت‌های سرویس^۲ متنوع می‌باشد. ATM با استفاده از تکنولوژی مولتی‌پلکس از میزان بهره‌وری از منابع شبکه را بهبود می‌بخشد. با این وجود اگر شبکه دارای مکانیزم ساماندهی مناسب نباشد، کیفیت سرویس به دلیل تداخلهای تصادفی پیش‌آمده در ترافیک و همچنین بروز مشکل در گره‌های سوئیچ شبکه تنزل خواهد یافت. از آنجائیکه کنترل ترافیک در تکنولوژی ATM مبحثی مهم می‌باشد، CCITT در توصیه‌های I.371^۳ بیشنهادات و قراردادهای لازم را عرضه نموده است. بدلیل اینکه در شبکه‌های ATM ساختار کanal مشخصی در نظر گرفته نشده است، ترافیک کاربر می‌تواند براحتی منابع شبکه را اشغال نماید و منجر به بروز تراکم گردد. مکانیزم‌های کنترل درخواست ورود^۴ می‌توانند تعیین نمایند که آیا پذیرفته شدن یک اتصال با کیفیت سرویس معین می‌تواند منجر به بروز تداخل در سایر اتصالات گردد یا خیر. حال پس از پذیرفته شدن یک اتصال در صورت بروز هرگونه تخلف از پارامترهای اولیه مکانیزم کنترل پارامتر کاربری^۵ اقدام به انجام عمل متناسب با تخلف صورت گرفته می‌نماید. در مجموع هر اندازه که برقراری قرارداد با جزئیات بیشتری صورت پذیرد، رفتار سرویس را به شکل کارآمدتری می‌توان ساماندهی نمود. ولی از سوی دیگر هنگامیکه قرارداد بسیار پیچیده می‌شود اعمال UPC بصورتی کارآمد مشکل می‌گردد و مهمتر آنکه بدست آوردن چنین مشخصاتی برای سرویس‌های خاص، مانند ویدئو بسیار مشکل می‌شود. الگوریتم سطح چکه‌ای^۶ و انواع مختلف الگوریتم‌های پنجه‌ر^۷ بعنوان دو روش کارآمد برای اعمال UPC در شبکه ATM در نظر گرفته شده‌اند. هر چند که الگوریتم‌های پنجه‌ر در شبکه‌های بسته‌ای عملکرد بسیار خوبی دارند، اما آنها مناسب برای شبکه‌های با سرعت بالای ATM نیستند. روش‌های UPC بسیاری وجود دارند که کنترل نرخ میانگین و حداقل را انجام

¹ - Quality Of Service (QoS)

² - Call Admission Control (CAC)

³ - Usage Parameter Control (UPC)

⁴ - Leaky Bucket (LB)

⁵ - Windowing Algorithms

می‌دهند. در [۲] چندین مکانیزم ساماندهی پیشنهاد شده که سطل چکه‌ای بعنوان موثرترین آنها معرفی گردیده است. این الگوریتم می‌تواند با تغییر، ساماندهی و شکل‌دهی نرخ ترافیک ورودی به شبکه اقدام به حفاظت از کیفیت سرویس موجود نماید. پیاده‌سازی سخت‌افزاری این مکانیزم بسیار سریع و مقرون به صرفه می‌باشد.

اما از سوی دیگر هنگامیکه سطل چکه‌ای برای ساماندهی منابع انفجاری مورد استفاده قرار می‌گیرد تخمین مشخصات یک منبع انفجاری مانند نرخ میانگین، بعلت طبیعت آماری انفجار منجر به بروز خطاها می‌شود که با کاهش دوره‌های تخمین افزایش می‌یابند. در ضمن افزایش فاصله تخمین منجر به بروز عدم پویائی مورد نیاز شبکه ATM می‌شود [۳].

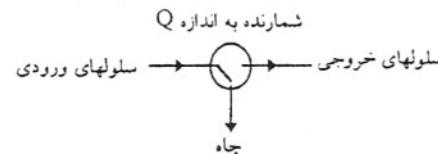
از زمان ارائه الگوریتم سطل چکه‌ای تا حال، روش‌های بهبودیافته بسیاری ارائه شده است. در [۴] کارآیی سطل چکه‌ای در شبکه ATM مورد مطالعه و بررسی قرار گرفته است و در [۲] نیز عملکرد کنترل جریان چهار نوع الگوریتم پنجه را در شبکه ATM مورد مطالعه و بررسی قرار داده است. در [۵] براساس سطل چکه‌ای قدیمی مکانیزم جدیدی ارائه شده است که دارای قابلیت تطبیق بیشتری نسبت به مشخصات ترافیک می‌باشد. در حال حاضر نیاز مبرمی به یک مکانیزم کنترل پویای UPC در شبکه‌های سرعت بالا که در ضمن تضمین کیفیت سرویس، استفاده بهینه را از منابع شبکه بعمل آورده [۶] احساس می‌شود.

در این راستا مکانیزم‌های گوناگونی ارائه شده که عمدۀ آنها سعی دارند به گونه‌ای تشخیص دهنده رفتار منبع چگونه است و براساس این تشخیص اگر منبع بصورت غیرعادی رفتار نمود سلوهای آنرا علامت‌گذاری نموده و یا دور بریزند. ساماندهنده پیشنهادی در [۷] از این نوع می‌باشد. استه در بعضی روش‌های پیشنهاد شده مانند [۶] نیز سعی گردیده به اشکال گوناگونی مانند تنظیم نرخ تولید توکن عملکرد ساماندهنده بهبود یابد. در مکانیزم ساماندهی پیشنهادی در این مقاله سعی می‌گردد پارامتر چکه بگونه‌ای تنظیم شود که میزان بهره‌وری از منابع شبکه افزایش یابد. در این روش همواره از بین چندین پارامتر چکه از پیش تعیین شده یکی بعنوان نرخ چکه جاری در نظر گرفته می‌شود و انتخاب آن نیز بگونه‌ای است که تعداد سلوهای دور ریخته شده توسط ساماندهنده کاهش یابد.

این مقاله به صورت زیر سازماندهی شده است: در بخش‌های ۲، ۳ و ۴ توضیحاتی در مورد سطل چکه‌ای ساده، مدل ترافیک انفجاری و اتوماتیک یادگیر ارائه می‌گردد. سپس در بخش‌های ۵ و ۶ ساماندهنده‌های جدیدی با استفاده از اتوماتیک یادگیر و منطق فازی پیشنهاد می‌گردد و در نهایت در بخش‌های ۷ و ۸ نتایج شبیه‌سازی و مقایسات صورت پذیرفته عرضه می‌شود.

(۲) مکانیزم سطل چکه‌ای

ساده‌ترین نوع مکانیزم سطل چکه‌ای یک شمارنده می‌باشد که بینگام تولید هر سلوی یک واحد افزایش می‌یابد و دائمًا با نرخ ثابت m تا زمانیکه شمارنده ثابت است، کاهش می‌یابد (شکل ۱). زمانی که شمارنده به یک مقدار آستانه Q برسد سلوی ورودی دور ریخته می‌شود (با آنکه بعنوان سلوی اضافه مطابق با [۸] علامت زده می‌شود).



شکل ۱) مکانیزم سطل چکه‌ای

نرخ چکه m معمولاً بصورت $m = c \cdot m$ با نرخ میانگین مرتبط می‌باشد که در این رابطه c ثابتی بزرگتر یا مساوی یک است و m نیز نرخ میانگین مورد توافق می‌باشد، اگر انحراف بصورت $m = d \cdot m$ تعریف شود ($d = m/n$ نرخ میانگین واقعی منبع می‌باشد) در آنصورت اثبات گردیده است بازای هر مقدار d که از c کوچکتر باشد، سطل چکه‌ای قادر به عمل نمی‌باشد [۹] مقدار c (و در نتیجه مقدار d) دارای اهمیت بسیاری است زیرا توسط آن شدت کنترل مشخص می‌گردد و اهمیت این موضوع بدین دلیل است که هر اندازه کنترل اعمال شده توسط رویه ساماندهی قویتر باشد پهنانی باند بیشتری حفظ می‌شود [۱۰].

(۳) مدل ترافیک یک منبع انفجاری

یک منبع وقتی انفجاری نامیده می‌شود که دارای حالات فعال و بیکار باشد. منابع صوتی و انتقال داده با سرعت بالا نمونه‌هایی از منابع انفجاری هستند. جهت مدل نمودن این منابع می‌توان از زنجیره مارکوف دو حالته استفاده نمود. حالات ON و OFF دوره‌های فعال و بیکار را نشان می‌دهند. مشخصات منبع را می‌توان با مجموعه پارامترهای زیر بیان نمود:

$$\begin{aligned} m &: \text{نرخ بیت میانگین} \\ b &: \text{میزان انفجاری بودن منبع} \\ T_{OFF} &: \text{میانگین مدت زمان بیکاری} \\ p &: \text{نرخ بیت حداقل} \\ L &: \text{طول متوسط انفجار (برحسب سلوی)} \\ T_{ON} &: \text{میانگین مدت زمان انفجار} \end{aligned}$$

در اینجا فرض شده است که دوره‌های فعالیت و بیکاری هر دو دارای توزیع نمایی با میانگین‌های T_{ON} و T_{OFF} هستند. بر طبق [۹]: $b=p/m$, $T_{ON}=(L.n_{cell})/p$, $T_{OFF}=T_{ON}(b-1)$

در اینجا n_{cell} نشانده‌نده طول سلول برحسب بایت می‌باشد، که برابر ATM استاندارد ۵۳ می‌باشد.

۴) مروری بر اتوماتای یادگیر

اتوماتای یادگیر یک ماشین حالت متناهی است که با یک محیط تصادفی دائما در حال محاوره می‌باشد و از طریق یک فرآیند یادگیری سعی در آموختن عمل بهینه مورد نیاز محیط دارد. در هر حالتی اتوماتا بطبقه بردار احتمالات یک عمل را انتخاب می‌نماید و سپس از یک تابع خروجی استفاده می‌نماید تا تعیین نماید پاسخ محیط به این عمل چیست (جریمه یا پاداش). اتوماتا با درنظر گرفتن این پاسخ و بوسیله یک تابع انتقال، حالت خود را به حالتی جدید که لزوماً با حالت قبل متفاوت نیست تغییر می‌دهد. اتوماتا حتی در زمانیکه اطلاعات کمی از سیستم در دسترس باشد، می‌تواند برای پیشگوئی رفتار آن سیستم عمل نماید [۱].

اتوماتاهای یادگیر را می‌توان به لحاظ ساختاری به دو دسته عمده ساختار ثابت و ساختار متغیر تقسیم نمود. در نوع اول هنگام استفاده از اتوماتا بردار احتمالات تغییر نمی‌نماید و فقط براساس یک ترتیب از پیش تعریف شده انتقال بین اعمال صورت می‌گیرد. ولی در نوع دوم بردار احتمالات دائما در حال تغییر می‌باشد و برحسب آنکه یک عمل از محیط جریمه یا پاداش دریافت دارد احتمال مرتبط با آن کاهش و یا افزایش می‌یابد.

هر الگوریتم یادگیری را می‌توان بصورت زیر بیان نمود:

$$P(n+1) = T[P(n), \alpha(n), \beta(n)]$$

$\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه خروجی‌های اتوماتا

$\beta = \{\beta_1, \beta_2, \dots, \beta_r\}$ مجموعه ورودی‌های اتوماتا

$P = \{p_1, p_2, \dots, p_r\}$ بردار احتمالات

اگر تابع T خطی باشد الگوریتم یادگیری خطی نامیده می‌شود و در غیراينصورت غیرخطی خوانده می‌شود. اينde اصلی تمامی الگوریتمهاي یادگیر بدين صورت است که اگر الگوریتم در تکرار n -ام عمل a_i را انتخاب نماید و پاسخ مطلوبی برای آن از محیط دریافت نماید در آنصورت احتمال (n) p_i مربوط به عمل a_i افزایش می‌يابد و احتمالات مربوط به سایر اعمال کاهش می‌يابد. در صورت دریافت جواب غيرمطلوب، (n) p_i کاهش می‌يابد و احتمالات مربوط به سایر اعمال افزایش می‌يابد. بنابراین:

• هنگام دریافت پاسخ مطلوب

$$p_j(n+1) = p_j(n) - f_j[p_j(n)]; \quad \forall j; \quad j \neq i$$

$$p_i(n+1) = p_i(n) + \sum_{j=1}^r f_j[p_j(n)]$$

• هنگام دریافت پاسخ نامطلوب

$$p_j(n+1) = p_j(n) - g_j[p_j(n)]; \quad \forall j; \quad j \neq i$$

$$p_i(n+1) = p_i(n) + \sum_{j=1}^r g_j[p_j(n)]$$

تابع f_j و g_j توابعی غیرمنفی می‌باشند و از آنها با عنوان تابع پاداش و جریمه نام برده می‌شود. هم اکنون مطالعات بسیاری برروی الگوریتمهاي یادگیر خطی به علت قابلیت انعطاف بالای آنها انجام می‌شود. برای این نوع از الگوریتمها تابع پاداش و جریمه بصورت زیر تعریف می‌شوند:

$$f_j[p_j(n)] = ap_j(n); \quad 0 < a < 1$$

$$g_j[p_j(n)] = \frac{b}{r-1} - bp_j(n); \quad 0 \leq b < 1$$

در روابط فوق r تعداد اعمال اتوماتا، a پارامتر پاداش و b پارامتر جریمه می‌باشند. براساس مقادیر مختلفی که برای پارامترهای یادگیری a و b در نظر گرفته می‌شود الگوریتمهاي مختلفي بدست می‌آيد که در زیر مواردی از آنها نام برده شده است الگوریتم یادگیری $L_{SN,5}$ نامیده می‌شود اگر $a=b$ باشد و L_{RI} خوانده می‌شود اگر $b=0$ باشد.

هر دو نوع اتوماتای فوق از نوع اتوماتاهای با ساختار متغیر هستند. اتوماتاهایی که در ادامه به تشریح آنها خواهیم پرداخت از نوع ساختار ثابت می‌باشند.

اتوماتای $L_{SN,5}$

اتوماتای $L_{SN,5}$ دارای ۵ حالت و ۵ عمل می‌باشد. اتوماتای $L_{SN,5}$ اقدام به نگهداری تعداد موفقیتها و شکستهای دریافت شده بازای هر عمل می‌نماید و فقط زمانیکه تعداد شکستهای بیش از تعداد موفقیتها گردد (یا از یک مقدار حداقل N تجاوز نماید) اقدام به تغییر عمل می‌نماید. اینکار با توسعه دادن فضای حالت به 5^N و تعیین قواعد انتقال از یک حالت به حالت دیگر صورت می‌پذیرد.

N حافظه مرتبط با هر عمل گفته می‌شود و در مجموع اتوماتا دارای حافظه‌ای برابر $5N$ می‌باشد. هنگام دریافت هر پاسخ مطلوب، حالت اتوماتا به صورت عمیق‌تری در حافظه آن قرار می‌گیرد و اگر پاسخ نامطلوب دریافت گردد آن حالت بیشتر از حافظه اتوماتا خارج می‌شود.

اتوماتای L_{5N} دارای $5N$ حالت $\varphi_1, \varphi_2, \dots, \varphi_{5N}$ و پنج عمل a_1, a_2, \dots, a_5 می‌باشد. حالتی $\varphi_{(i-1)N+K}$ مربوط به عمل a_i ($i=1\dots 5$) می‌باشد و هنگام قرارگرفتن در هر یکی از این حالات عمل a_i اجرا می‌گردد. انتقال حالت بهنگام دریافت پاسخ مطلوب بصورت زیر می‌باشد

$$\varphi_i \rightarrow \varphi_{i+1} \text{ if } i \neq KN, K=1\dots 4 \quad \varphi_i \rightarrow \varphi_{(i+1)N} \text{ if } i=KN, K=1\dots 4$$

$$\varphi_{5N} \rightarrow \varphi_1$$

$$a_i \rightarrow a_{(i \bmod 5)+1}$$

از آنجاییکه این اتوماتا دارای پنج عمل می‌باشد هنگام تعویض اعمال نیاز به وجود یک راهکار از پیش تعریف شده است تا عمل بعدی بدون بروز هیچگونه ابهامی انتخاب گردد. در اینجا اعمال بصورت ترتیبی عوض می‌شوند. بدین صورت که پس از عمل اول عمل دوم انتخاب می‌گردد و درنهایت پس از آخرین عمل مجدداً عمل اول اجرا می‌گردد. با توجه به این توضیحات، بهنگام دریافت پاسخ نامطلوب نیز عملکرد اتوماتا بصورت زیر است:

$$\varphi_i \rightarrow \varphi_{i+1} \text{ if } i \neq KN, K=0\dots 5 \quad \varphi_i \rightarrow \varphi_{(i+1)N} \text{ if } i=KN, K=0\dots 4$$

$$\varphi_{5N} \rightarrow \varphi_1$$

$$a_i \rightarrow a_{(i \bmod 5)+1}$$

اتوماتای $G_{5N,5}$

این نوع از اتوماتا دارای 5 عمل می‌باشد و با آموزش رفتار گذشته سیستم سعی در تغییر اعمال می‌نماید. در این اتوماتا هنگام دریافت پاسخ مطلوب، انتقال حالات بصورت زیر می‌باشد:

$$\varphi_i \rightarrow \varphi_{i+1} \text{ if } i \neq KN+1, K=0\dots 4$$

$$\varphi_i \rightarrow \varphi_i \text{ if } i=KN+1, K=0\dots 4$$

برخلاف اتوماتای $L_{5N,5}$ اتوماتای $G_{5N,5}$ پیش از آنکه از عملی به عمل دیگر انتقال یابد عمل جاری را حداقل N بار انجام می‌دهد و درنهایت هنگام دریافت پاسخ نامطلوب از محیط، انتقال حالات بشکل زیر صورت می‌پذیرد:

$$\varphi_i \rightarrow \varphi_{i+1} \text{ if } i \neq 5N$$

$$\varphi_{5N} \rightarrow \varphi_1$$

در این نوع از اتوماتا نیز مانند اتوماتای L_{5N} تعویض اعمال بشکل ترتیبی صورت می‌گیرد. و پس از تعویض هر عمل، عمل بعدی انتخاب می‌گردد.

اتوماتای کرینسکی

این نوع از اتوماتا نیز همانند موارد پیشین دارای پنج عمل می‌باشد. عملکرد این اتوماتا هنگام دریافت پاسخ مطلوب، بصورت زیر است:

$$\varphi_i \rightarrow \varphi_{i \bmod N+1} \text{ if } i \neq KN+1$$

$$\varphi_i \rightarrow \varphi_{(i \bmod N-1)} \text{ if } i=KN$$

$$\varphi_i \rightarrow \varphi_i \text{ if } i=KN+1, K=0\dots 4$$

عملکرد این نوع از اتوماتا بهنگام دریافت پاسخ نامطلوب از محیط مانند اتوماتای $L_{5N,5}$ می‌باشد. و می‌توان آنرا بصورت زیر تشریح نمود:

$$\varphi_i \rightarrow \varphi_{i+1} \text{ if } i \neq KN, K=1\dots 4$$

$$\varphi_i \rightarrow \varphi_{(i+1)N} \text{ if } i=KN, K=1\dots 4$$

$$\varphi_{5N} \rightarrow \varphi_N$$

$$a_i \rightarrow a_{(i \bmod 5)+1}$$

در اینجا مجموعاً جهت رفتن از یک عمل به عمل دیگر نیاز به دریافت N پاسخ نامطلوب متواتی می‌باشد. و در ضمن تعویض اعمال نیز مانند روشهای پیشین بصورت ترتیبی صورت می‌پذیرد.

۵) ساماندهی ترافیک با استفاده از سطل چکه‌ای دارنده اتوماتای یادگیر

^۱ کارهای گوناگونی در زمینه استفاده از اتوماتای یادگیر به منظور ساماندهی ترافیک در طول سالهای ۱۹۹۴ الی ۱۹۹۷ توسط واسیلاکوس

^۲ و اتسیس صورت پذیرفته است. در اولین کار انجام شده از نوعی اتوماتای یادگیر بنام SELA در جهت بهبود عملکرد سطل چکه‌ای استفاده گردیده است. عملکرد این ساماندهنده آنگونه است که با استفاده از تابع توزیع منبع ON-OFF مقدار ایده‌آل شمارنده LB محاسبه می‌شود و پس از آن این مقدار با مقدار واقعی شمارنده مقایسه می‌شود. هر اندازه اختلاف بیشتری بین این دو مقدار وجود داشته باشد منبع از پارامترهای مورد توافق خود بیشتر تخطی داشته است. با توجه به این تعریف دو عمل برای اتوماتا درنظر گرفته شده است که برطبق یکی از آنها منبع عملکردی عادی (عمل ۱) و برطبق دیگری عملکردی غیرعادی (عمل ۲) دارد. اگر میزان اختلاف گفته شده از یک مقدار آستانه‌ای تجاوز نماید احتمال عمل ۲ افزایش می‌یابد و اگر به یک مقدار آستانه بررسد در آنصورت تمامی سلوهای منبع دور ریخته می‌شوند. حال هر زمان که عمل ۱ انتخاب شود سلوهای ورودی به سطل چکه‌ای بدون هیچ تغییری از آن عبور می‌نمایند و اگر احتمال این عمل به یک مقدار مشخص بررسد در آنصورت کلیه احتمالات اتوماتا مجدداً مقداردهی اولیه می‌شوند.

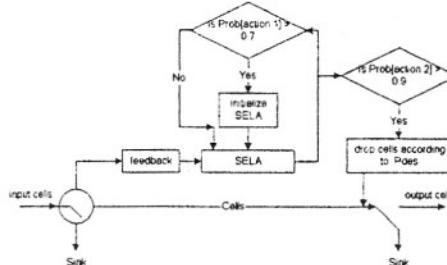
¹

- A. V. Vasilakos

²

- A. F. Atlasis

با توجه به شبیه‌سازی‌های صورت گرفته در [۱۱] روش جدید منجر به کاهش قابل توجهی در میزان سلولهای دور ریخته شده و کارآبی سطل می‌شود. لازم است در اینجا یادآوری شود که این روش در گرههای ابتدایی شبکه و بعارت واضح‌تر در محل تولید ترافیک بکار گرفته شده است. عملکرد این ساماندهنده در شکل ۲ نمایش داده شده است.

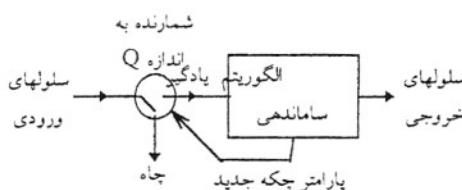


شکل ۲) مکانیزم ساماندهی

کار بعدی صورت گرفته در این راستا توسط واسیلاکوس و اتسیس در سال ۱۹۹۵ صورت گرفت [۱۰] و در آن ساماندهنده فوق‌الذکر بجای گرههای ابتدایی در گرههای میانی شبکه جای گرفت. شبیه‌سازی‌های صورت گرفته با استفاده از این روش نیز نشان می‌دهد که عملکرد سطل چکه‌ای جدید به مراتب بهتر از انواع رایج می‌باشد.

آخرین کار صورت گرفته در این زمانه در سال ۱۹۹۷ با تعیینتابع توزیع شمارنده سطل چکه‌ای منابع ترافیک ویدیوئی توسط اتسیس و واسیلاکوس صورت پذیرفت [۱۲]. در اینحالت نیز با محاسبه مقدار واقعی و ایدهآل شمارنده LB و اختلاف آنها تعیین می‌شود که آیا عملکرد منبع برطبق قرارداد بوده است یا نه. و در صورتیکه هرگونه تخلف مشاهده شود سلولهای منبع خاطی دور ریخته می‌شوند. در اینحالت نیز شبیه‌سازی‌ها حاکی از بهبود عملکرد و کارآبی بهتری نسبت به سطل چکه‌ای ساده می‌باشد.

در نوع ساد سطل چکه‌ای پارامتر چکه ثابت است. این امر باعث می‌شود هنگامیکه سطل چکه‌ای برای ساماندهی منابع انفجاری مورد استفاده قرار می‌گیرد و بعلت طبیعت انفجاری منبع تخمین مشخصات آن مانند میانگین سادگی انجام نمی‌گیرد و همیشه متضمن خطای می‌باشد و باعث دور ریزی سلولها بیش از مقدار مورد توافق در قرارداد می‌شود. در روش پیشنهاد شده از سطل چکه‌ای، نرخهای چکه مختلفی در یک محدوده مجاز درنظر گرفته شده است. هر یک از این نرخها متناظر با یکی از اعمال اتوماتا می‌باشد. اتوماتا دائماً با شناسائی مشخصات منبع سمعی می‌نماید بهترین نرخ را برای پارامتر چکه به گونه‌ای انتخاب نماید که میزان سلولهای دور ریخته شده کاهش یابد و عملکرد ساماندهنده ترافیک بهبود یابد. با توجه به توضیحات فوق نحوه عملکرد سطل چکه‌ای جدید مطابق با شکل ۳ می‌باشد.

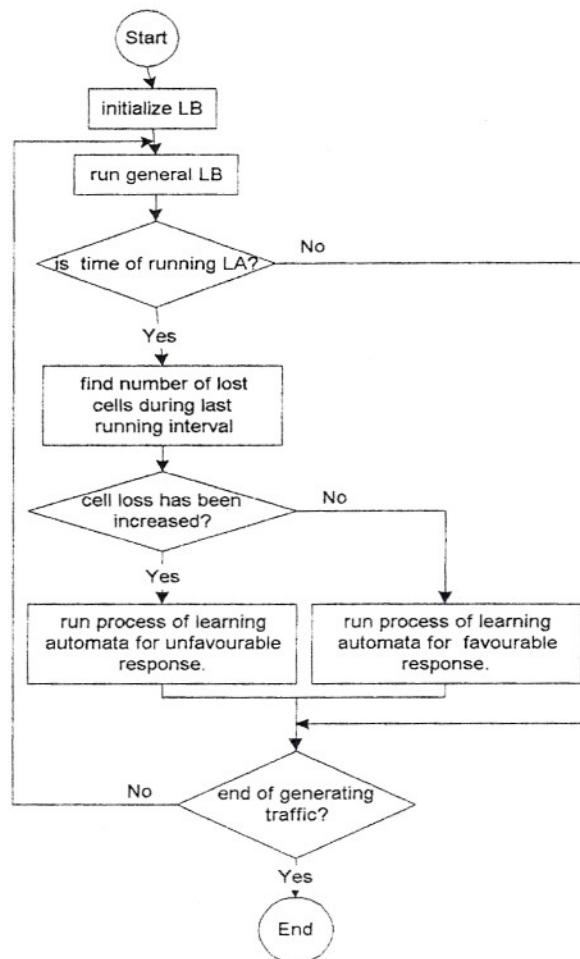


شکل ۳) سطل چکه‌ای با پارامتر چکه متفاوت

جزئیات بیشتر این روش بصورت نمودار جریان در شکل ۴ نمایش داده شده است. مطابق این نمودار اتوماتا ابتدا مدت زمانی مشخص را صرف آموزش اولیه می‌نماید. پس از آن در فواصل زمانی مشخص اتوماتا اجرای می‌شود و در هریار اجرا، تعداد سلولهای دور ریخته شده در فاصله زمانی گذشته محاسبه می‌گردد. اگر این تعداد نسبت به دفعه ماقبل آن بهبود یافته باشد به معنای پاسخ مطلوب ارزیابی می‌شود. حال برحسب آنکه چه نوعی از اتوماتا (ساختار ثابت یا متغیر) مورد استفاده قرار گرفته باشد این پاسخ مطلوب در احتمال و یا حالت اعمال تاثیر می‌گذارد. اما اگر تعداد سلولهای دور ریخته شده نسبت به دفعه ماقبل آن افزایش یافته باشد به منزله پاسخ نامطلوب درنظر گرفته می‌شود و متناسب با ساختار اتوماتا تاثیر خود را بر احتمالات و حالات اعمال می‌نماید و در صورت تعویض عمل اتوماتا اندازه سطل جدیدی را انتخاب می‌نماید. مدت زمانی که برای آموزش اولیه اتوماتا بکار می‌رود و همچنین فواصل زمانی اجرای اتوماتا مقادیری ثابت هستند که می‌توان آنها را متناسب با منابع ترافیک گوناگون تنظیم نمود. روش جدید در مقایسه با روش‌های متناسب بر اتوماتای یادگیر [۷] و [۱۰-۱۲] از دو جنبه متفاوت است. اولاً معیار عملکرد مطلوب و یا نامطلوب منبع در روش‌های پیشین انحراف از مقدار ایدهآل شمارنده سطل چکه‌ای می‌باشد ولی در روش‌های پیشنهادی معیار جدیدی بر مبنای تعداد سلولهای گمشده در بازه‌های زمانی پیشین تعریف شده است. با استفاده از این معیار تصمیم‌گیری‌ها بیشتر بر واقعیت منطبق است و عملکرد اتوماتا و اینکه آیا از قرارداد خود تخطی داشته است را می‌توان براساس یک معیار عینی (مانند میانگین سلولهای دور ریخته شده در یک بازه زمانی) بیان نمود. ثانیا در این روش اتوماتاهایی کاملاً متفاوت با انواع طراحی شده توسط اتسیس و واسیلاکوس بکار گرفته شده است.

در روش پیشنهادی، ATM به عنوان یک سرویس مورد توجه قرار گرفته است تا یک بستر ارتباطی. اگر ATM بعنوان یک بستر درنظر گرفته شده باشد روش‌های پیشنهادی که از اتوماتاهای با ساختار ثابت استفاده می‌کنند مناسب‌تر می‌باشد. بدلیل اینکه پیاده‌سازی اتوماتاهای با ساختار

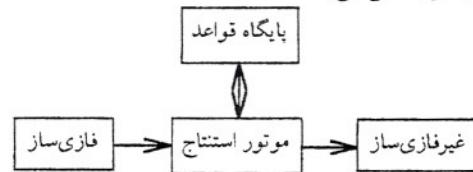
متغیر در سخت افزار بسیار کند عمل می نماید استفاده از آنها در شبکه های پرسرعی مانند ATM مناسب نمی باشد. یکی از مزایای پیاده سازی سخت افزاری روش های پیشنهادی بیشتر شدن قابلیت انعطاف ساماندهنده جهت استفاده به همراه کاربردهای گوناگون می باشد. زیرا در این حالت ساماندهنده خود را مناسب با عملکرد کاربرد جدید تنظیم می نماید و نیازی به تغییر ساماندهنده بدلیل تغییر کاربرد نمی باشد.



شکل ۴) نمودار عملکرد سطح چکه ای جدید

۶) ساماندهی ترافیک با استفاده از منطق فازی

همانگونه که در مباحث پیشین بیان شد هدف از بکار بردن اتماتای یادگیر تنظیم پارامترهای گوناگون سطح چکه ای در جهت بهبود عملکرد ساماندهی منابعی که شناخت کاملی از آنها در دسترس نیست می باشد. علاوه بر استفاده از اتماتای یادگیر به کمک سایر روشها نیز می توان اقدام به تنظیم این پارامترها نمود که از مهمترین آنها می توان به منطق فازی اشاره نمود. با استفاده از یک کنترل کننده فازی می توان با بهره گیری از مجموعه های از متغیرهای زبانی ورودی و تعدادی قواعد کنترل فازی اقدام به تعیین یک عمل کنترلی نمود. در ادامه به بیان چگونگی طراحی یک ساماندهنده ترافیک فازی پرداخته و نحوه کاربرد آنرا در شبکه های ATM مورد مطالعه قرار می دهیم. ساماندهنده فازی مطابق شکل ۵ دارای چهار بخش اساسی می باشد:



شکل ۵) اجزای مختلف ساماندهنده ترافیک فازی

- فازی ساز: پارامتر ورودی سیستم را به متغیر زبانی مناسب جهت استفاده در موتور استنتاج تبدیل می کند.
- پایگاه قواعد فازی: در بردارنده تعدادی قواعد کنترل فازی است که بصورت زبانی جهت تشریح رویه کنترل نوشته شده است.

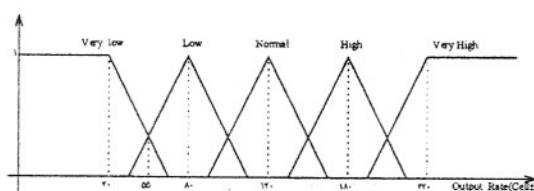
متوار استنتاج: اقدام به استنتاج عمل کنترل فازی براساس قواعد کنترل فازی و پارامترهای زبانی ورودی می‌نماید.
 • غیرفازی‌ساز: عمل کنترل فازی استنتاج شده را به یک عمل کنترل غیرفازی، و براساس یک استراتژی غیرفازی‌ساز تبدیل می‌نماید.
 اساس عمنکرد این ساماندهنده به آن صورت است که هر اندازه نرخ خروجی بیشتر شود نرخ خروجی افزایش می‌یابد و بالعکس با کاهش نرخ ورودی نرخ خروجی نیز کاهش می‌یابد. این روش کنترل علاوه بر کاهش تعداد سلولهای دوربریخته شده امکان بهره‌وری هر چه بیشتر منابع را نیز فراهم می‌کند. جهت طراحی این کنترل کننده پنج نوع متغیر زبانی برای بیان چگونگی تغییرات نرخ ورودی و یک متغیر زبانی خروجی جهت نرخ خروجی تعریف شده است.

با بکار بردن هر یک از متغیرهای زبانی ورودی بهمراه متغیر زبانی خروجی، پنج نوع روش فازی شده است که هریک از آنها بصورت مجزا مورد آزمایش قرار گرفته‌اند و درصد سلولهای دوربریخته شده توسط آنها محاسبه گردیده و نتایج حاصله در نمودار ۶ نمایش داده شده است.

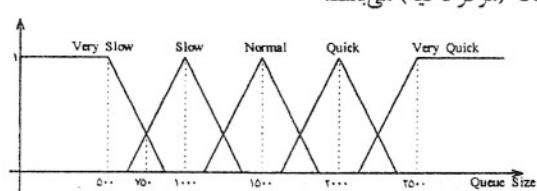
قواعد فازی بکار رفته توسط متوار استنتاج بصورت زیر است:

- Rule 1) IF input rate is very slow THEN output rate is very low.
- Rule 2) IF input rate is Slow THEN output rate is low.
- Rule 3) IF input rate is Normal THEN output rate is Normal..
- Rule 4) IF input rate is Quick THEN output rate is High..
- Rule 5) IF input rate is Very Quick THEN output rate is Very High.

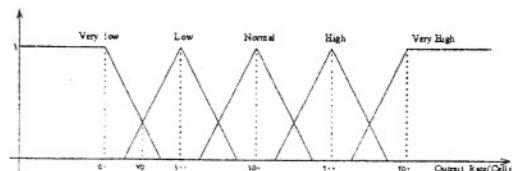
نوابع عضویت برای متغیرهای زبانی ورودی و خروجی در اشکال ۶ تا ۱۱ نمایش داده شده است. در ضمن متوار استنتاج بکار رفته از نوع COA¹ (مرکز تاحیه) می‌باشد.



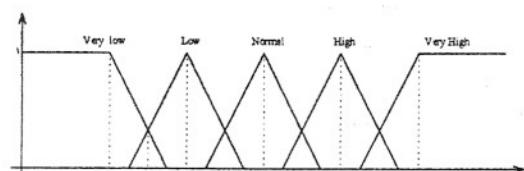
شکل ۷) متغیر زبانی نرخ خروجی نوع ۱



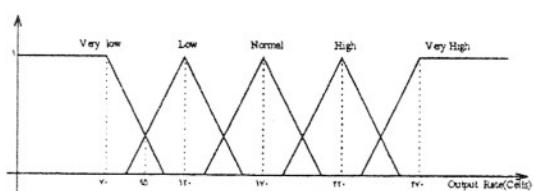
شکل ۸) متغیر زبانی نرخ ورودی



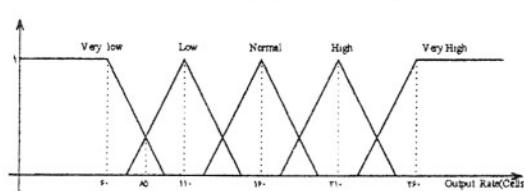
شکل ۹) متغیر زبانی نرخ خروجی نوع ۳



شکل ۱۰) متغیر زبانی نرخ خروجی نوع ۴



شکل ۱۱) متغیر زبانی نرخ خروجی نوع ۵



شکل ۱۲) متغیر زبانی نرخ خروجی نوع ۶

۷) شبیه‌سازی

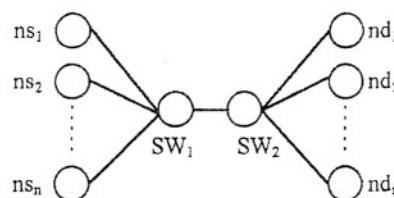
جهت انجام شبیه‌سازیها با استفاده از روش‌های پیشنهادی مبتنی بر اتوماتاهای یادگیر و همچنین روش‌های پیشنهادی مبتنی بر منطق فازی، از شبیه‌ساز ATM/HFC² تولید شده در موسسه NIST استفاده گردیده است. این شبیه‌ساز دارای قابلیت مطالعه و ارزیابی شبکه‌های ATM و می‌باشد و یک محیط مدلسازی محاوره‌ای بهمراه واسط کاربر گرافیکی برای کاربر فراهم می‌کند. موسسه NIST این ابزار را با استفاده از زبان C و سیستم XWindows که در بستر یونیکس اجرا می‌گردد تولید نموده است. با استفاده از این ابزار کاربر می‌تواند اقدام به ایجاد همبندی‌های گوناگون شبکه، کنترل پارامترهای مختلف اجزاء، مشاهده فعالیت شبکه و ثبت نمودن داده‌های حاصل از شبیه‌سازیها نماید [۱۳]. در شبیه‌سازی‌های صورت گرفته از منابع ON-OFF استفاده شده است که مشخصات آن در جدول ۱ آورده شده است.

¹ - Center Of Area
² - Hybrid Fiber Coax

مقدار	پارامتر منبع
4240 usec	مدت زمان میگذین ON
3816000 usec	مدت زمان میگذین OFF
10 Mbit/s	نرخ حداکثر ترافیک ارسانی
1 Mbit/s	نرخ میگذین ترافیک ارسانی
500 sec	مدت زمان اجرا

جدول ۱) مشخصات منابع ON-OFF مورد استفاده

همبندی استفاده شده در شبیه‌سازیها مطابق با شکل ۱۱ می‌باشد.



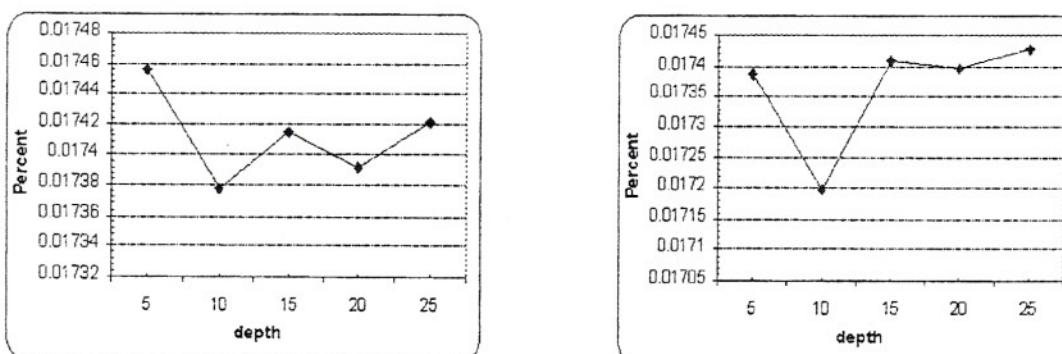
شکل ۱۱) همبندی شبکه بکار رفته

در این همبندی تعداد ۲۰۰ عدد منبع اقدام به ارسال ترافیک به مقصد می‌نمایند. از آنجاییکه ساماندهی ترافیک فقط در مبدأ انجام می‌گیرد تعداد سلولهای دور ریخته شده تنها در مبدأ محاسبه شده و مورد بررسی قرار گرفته است. این بررسی بازی انوع مختلف اتماتاهای صورت پذیرفته است و نتایج آن در انتهای این قسمت آورده شده است. جهت مقایسه ابتدا عملکرد سطل چکه‌ای ساده با مشخصات جدول ۲، توسط شبیه‌سازی مورد مطالعه قرار گرفت که برای آن متوسط سلولهای دور ریخته شده $1/270.8$ درصد بوده است که در مقایسه با نتایجی که از روش سطل چکه‌ای که در آن اتماتاهای یادگیر استفاده شده است بمراتب بالاتر بوده است.

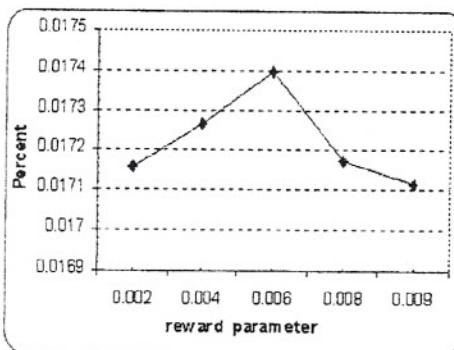
مقدار	پارامتر سطل چکه‌ای
1465000 bit/s	نرخ جکه
2500 cell	اندازه سطل
0.27087%	درصد متوسط سلولهای دور ریخته شده

جدول ۲) مشخصات سطل چکه‌ای بکار رفته

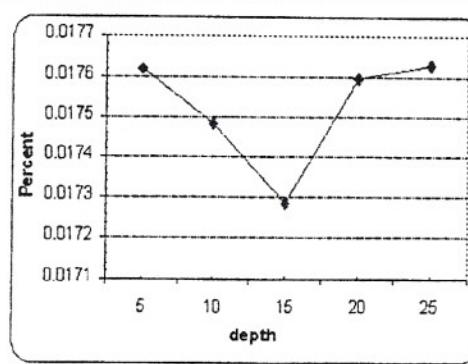
شبیه‌سازی‌های روش پیشنهادی برای اتماتاهای $L_{NS,5}$ ، $L_{NS,4}$ ، کرینسکی، $G_{NS,5}$ و ساماندهندهای فازی انجام گرفت که نتایج حاصل در نمودارهای ۱ تا ۶ ارائه گردیده است.



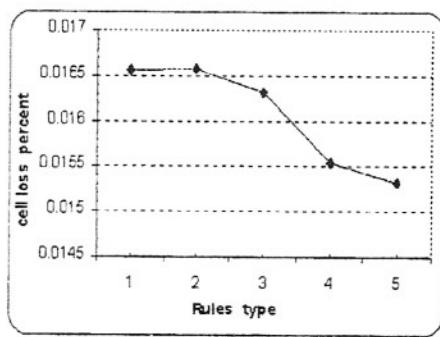
نمودار ۱) تغییرات درصد سلولهای گم شده بر حسب عمق‌های مختلف اتماتای کرینسکی
 نمودار ۲) تغییرات درصد سلولهای گم شده بر حسب عمق‌های مختلف اتماتای $L_{NS,5}$



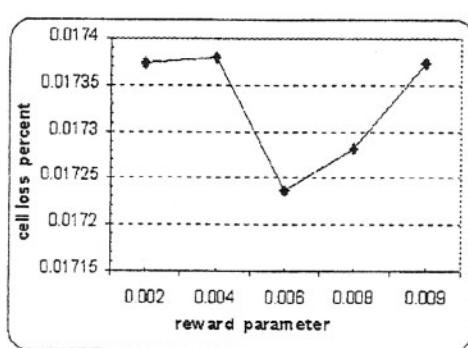
نمودار ۴) چگونگی تغییر درصد سلوهای گم شده بر حسب مقادیر مختلف پارامتر پاداش و جریمه در اتوماتای G_{NS} .



نمودار ۳) تغییرات درصد سلوهای گم شده بر حسب عمق های مختلف اتوماتای G_{NS} .



نمودار ۶) تغییرات درصد سلوهای گم شده بازی هریک از قواعد ساماندهنده فازی.



نمودار ۵) تغییرات درصد سلوهای گم شده بر حسب پارامتر پاداش در اتوماتای G_{NS} .

برطبق نمودار ۱ اتوماتای G_{NS} با عمق حافظه ۱۰ منجر به کمترین میزان دور ریختگی سلول می شود. برای عمق های ۱۵ و ۲۰ عملکرد اتوماتا تقریباً یکسان است. با افزایش عمق درصد سلوهای گم شده روبه فزوئی می گذارد. اتوماتای کرنسکی برطبق نمودار ۲ بدترین عملکرد را برای عمق ۵ و بهترین عملکرد را برای عمق ۱۰ تولید می کند. با افزایش عمق درصد سلوهای دور ریخته شده به تدریج روبه از دیگر اتوماتا می گذارد. با توجه به نمودار ۳ اتوماتای G_{NS} در عمق ۱۵ بهترین عملکرد را دارا می باشد. بازی عمق های دیگر عملکرد بدتر شده و از عملکرد بهینه خود دور می شود. برطبق نمودار ۴ اتوماتای G_{NS} بازی که پارامتر پاداش $R = 0.006$ است بدترین عملکرد را عرضه می دارد اما هر اندازه که بر میزان پارامتر پاداش افزوده می شود مشاهده می گردد که عملکرد آن بهبود می یابد. بدیهی است که این روند صعودی نمی تواند دائم ادامه یابد و در جانی متوقف خواهد شد که جهت بدست آوردن آن نیاز به شبه سازی های پیشتری است.

اتوماتای G_{NS} بازی ای پاداش $R = 0.006$ دارای عملکرد بهینه است و در سایر نقاط منجر به دور ریزی سلوهای زیادتری می شود که استفاده از آن مقادیر را غیرضروری می کند. با مقایسه نمودار ۶ با نمودارهای شماره ۱ تا ۵ مشاهده می شود که عملکرد روش فازی بهتر از روش های یادگیر پیشنهادی می باشد. البته نوع اتوماتای ۱۵-عملی جهت ساماندهی ترافیک به منظور تنظیم اندازه سطلهای نیز پیشنهاد گردیده است که عملکرد آن بهتر از انواع ۵-عملی مشابه می باشد. در این اتوماتا فاصلی که بین اعمال اتوماتا وجود دارد کاهش یافته و از سوی دیگر گردیده است که عملکرد آن یافته است این امر منجر به آن گردیده است که عملکرد اتوماتا و تغییر اعمال آن تا حد زیادی به عملکرد روش های فازی که در آنها متغیر های زبانی ورودی و خروجی حول یک نقطه در تغییر هستند نزدیکتر شود و از سوی دیگر کارآبی اتوماتا در جهت کاهش تعداد سلوهای دور ریخته شده افزایش یابد.

پس از بررسی کارآبی اتوماتای یادگیر در ساماندهی ترافیک در ادامه به مقایسه این روشها با ساماندهی ترافیک پیشنهاد شده در [۶] پرداخته می شود. مطابق نمودار ارائه شده در شکل A این مقاله هنگامیکه نرخ توکنها در محدوده $0.05 \leq R \leq 0.06$ برای نرخ میانگین منبع تغییر می کند عملکرد اتوماتای یادگیر بهتر از روش فازی می باشد زیرا در این حالت احتمال دور ریختن سلوهای در ساماندهنده دارای اتوماتای یادگیر در محدوده $0.017 \leq R \leq 0.0175$ تغییر می کند اما محدوده تغییر این احتمال برای روش فازی محدوده $0.017 \leq R \leq 0.0177$ است. در ضمن با تنظیم مناسب نسبت پارامتر چکه و نرخ میانگین منبع می توان این بهبود را در تمامی محدوده ها بدست آورد.

۸) نتیجه گیری و کارهای آتی

در این مقاله روش سطلهای چکه ای معمولی و سطلهای چکه ای مقایسه شده اند. جهت تنظیم پارامتر چکه ای از اتوماتاهای یادگیر با ساختار ثابت مانند کرنسکی و ساختار متغیر نظری G_{NS} و منطق فازی استفاده گردیده است. نتایج حاصل از شبیه سازی های بیانگر بهبود قابل توجهی در عملکرد سطلهای چکه ای می باشد. با توجه به انواع گوناگون اتوماتای یادگیر و تنوعات بسیاری که می توان از جهت تعداد اعمال،

ساختار اتوماتا و یا حتی ترکیب انواع گوناگون از اتوماتا با یکدیگر بدست آورده می‌توان بهبودهای بیشتری در عملکرد اتوماتها ایجاد نمود. از جمله این کارها می‌توان تغییر تعداد اعمال اتوماتا، تنظیم مناسب فواصل زمانی اجرای اتوماتا، بررسی چگونگی تنظیم اتوماتا برای انواع منابع ترافیک و ترکیب انواع اتوماتا را نام برد.

(۹) فهرست مراجع

- 1- CCITT, Recommendation I.121 : "Broadband aspects of ISDN ", Blue Book, Vol. III. 7, Geneva, Switzerland, 1989.
- 2- E. P. Rathgeb, "Modeling and performance comparison of policing mechanisms for ATM networks", IEEE J-SAC April 1991 pp 325-334.
- 3- A.I. Elwalid, D. Mitra, "Analysis and design of rate-base congestion control of high speed networks, I: stochastic fluid models, access regulation", Queueing Systems 9 1991, pp. 29-63.
- 4- Milena Butto, Elisa Cavallero and Alberto Tonietti, "Effectiveness of the leaky bucket policing mechanism in ATM networks", IEEE J. Select. Areas. Commun., vol.9, no.3, pp. 335-342 April 1991.
- 5- Irfan Khan, et al, "Traffic control in ATM networks", Computer Networks and ISDN System 27(1994)85-100.
- 6- Z. Jiang, Z. Liu, "An improved algorithm of usage parameter control in ATM networks", ICCT'96, vol.1 pp. 24-27.
- 7- Y. Seifi , M.R. Meybodi , "Traffic policing in ATM networks using two action l_{tp} learning automata", Technical report, Computer eng. dep., Amirkabir University of Technology, Tehran, Iran, 2001.
- 8- G. Gallassi, G. Rigolio, L. Fratta, "ATM: Bandwidth assignment and bandwidth enforcement policies", IEEE GLOBECOM 1989, pp.178801793.
- 9- J. A. S. Monteiro, M Gerla, L. Fratta, "Leaky bucket input rate control in ATM networks", ICCC 1990, New Delhi, India, 1990, pp. 370-376.
- 10- A. V. Vasilakos, A. F. Atassis, "Effectiveness of the LB-SELA policing mechanism in an ATM network node", IEEE GLOBECOM'95, November 1995, pp. 627-631.
- 11- A. V. Vasilakos, A. F. Atassis, "LB-SELA: rate-based access protocol for ATM networks", IEEE INFOCOM June 1994, pp. 1552-1559.
- 12- A. F. Atassis, G.I. Stassinopoulos, A. V. Vasilakos, "Leaky bucket mechanism with learning algorithm for ATM traffic policing", second IEEE symposium on computer and communications, July 1997, pp 68-72.
- 13- N. Golmie, F. Mouveaux, L. Hester, Y. Saintillan, A. Koenig, D. Su, "The NIST ATM/HFC network simulator version 4.0", high-speed networks technology group, advanced networks technology division, information technology laboratory, NIST, December 1998.