

علوم و مهندسی کامپیوتر

نشریه علمی پژوهشی انجمن کامپیوتر ایران

صاحب امتیاز: انجمن کامپیوتر ایران
مدیر مسئول: دکتر جعفر حبیبی
سر دبیر: دکتر حمید سربازی آزاد

شورای علمی

- گ. آقا، استاد دانشگاه ایلینویز، امریکا
ف. ارباب، استاد سی.دبلیو.آی و دانشگاه لایدن، هلند
ن. باقرزاده، استاد دانشگاه کالیفرنیا- پروین، امریکا
ک. بدیع، دانشیار مرکز تحقیقات مخابرات، ایران
ب. بوزه، استاد دانشگاه ایالتی اورگان، امریکا
ب. پرهامی، استاد دانشگاه کالیفرنیا- سنتاباربارا، امریکا
ف. جهانیان، استاد دانشگاه میشیگان، امریکا
آ. زومایا، استاد دانشگاه سیدنی، استرالیا
ح. سربازی آزاد، دانشیار دانشگاه صنعتی شریف، ایران
ب. شیرازی، استاد دانشگاه ایالتی واشنگتن، امریکا
ا. کبیر، استاد دانشگاه تربیت مدرس، ایران
ر. صفابخش، استاد دانشگاه صنعتی امیرکبیر، ایران
ح. ر. عربنیا، استاد دانشگاه جورجیا، امریکا
ع. عدالت، استاد کالج سلطنتی لندن، انگلستان
م. فتحی، دانشیار دانشگاه علم و صنعت، ایران
م. ح. قاسمیان، استاد دانشگاه تربیت مدرس، ایران
م. قدسی، استاد دانشگاه صنعتی شریف، ایران
ع. هورسان، استاد دانشگاه ایالتی پنسیلوانیا، امریکا
ف. لو، استاد دانشگاه هنگ کنگ، چین
ع. موقر، استاد دانشگاه صنعتی شریف، ایران
ن. مهدوی امیری، استاد دانشگاه صنعتی شریف، ایران
م. ر. میبیدی، استاد دانشگاه صنعتی امیرکبیر، ایران
ک. ناکانو، استاد دانشگاه هیروشیما، ژاپن
م. ولدخوا، استاد دانشگاه گلاسگو، انگلستان

مشاوران

- ج. آرمسترانگ، دانشگاه نورث ایسترن، امریکا
ا. المقارم، دانشگاه پردو، امریکا
م. انواری، مرکز فناوری اطلاعات و ارتباطات پیشرفته شریف، ایران
ح. برنجی، ناسا، امریکا
د. بیورنر، دانشگاه دلفت، هلند
م. پدram، دانشگاه کالیفرنیا جنوبی، امریکا
ب. ذوقی، دانشگاه ای.اند.ام تگزاس، امریکا
پ. روگای، دانشگاه کالیفرنیا- دیویس، امریکا
م. قنبری، دانشگاه اسکس، انگلستان
م. صرافزاده، دانشگاه کالیفرنیا- لس آنجلس، امریکا

همکاران دفتر مجله: محمد ارجمند، شهیندخت شهنازی، لیلیا نورانی، آرش توکل

نشانی: تهران، خیابان آزادی، ضلع غربی دانشگاه صنعتی شریف، کوچه شهید ولی... صادقی، پلاک ۲۶، طبقه ۴، واحد ۱۶، دفتر انجمن کامپیوتر ایران،
نشریه علوم و مهندسی کامپیوتر. تلفن ۰۳۲۰۰۰-۶۶۰۸۷۲۲۴-۶۶۰۲۱۱۴۹، دورنگار ۶۶۰۲۱۱۴۹
پست الکترونیکی: csi-jcse@ipm.ir، تور جهان گستر: <http://www.jcse.ir>

مقالات درج شده در این نشریه صرفاً بیانگر نظرات مؤلفین آنها است و مسئولیت صحت و سقم داده‌ها و نتایج بر عهده آنها است.

لیتوگرافی، چاپ و صحافی: مرکز نشر دانشگاهی

فهرست مقالات

- بهبود پروتکل مسیریابی Tora در شبکه‌های ویژه سیار بکمک الگوریتم کولونی مورچه‌ها ۱
رعنا غزالی، مجتبی صفاری و علی موقر رحیم‌آبادی
- کاوش متون فارسی بر مبنای روش طبقه‌بندی ۸
محمد حسین سرابی و آذر شاهقلیان
- بررسی چند نقشه راه ارائه شده برای مهاجرت به SOA ۱۴
طیبه آزادی، علی معینی و ناصر مدیری
- مولد موتور گردش کار SWEG ۲۶
مرتضی یوسف صنعتی و سیدحسن میریان حسین‌آبادی
- کاربردهای آتوماتای یادگیر سلولی در پردازش تصویر ۳۷
احمدعلی آیین، سیدحمید امیری و حمید بیگی
- یک روش کارآمد برای محافظت از پیاده‌سازی سخت‌افزاری الگوریتم رمزنگاری AES در مقابل حمله تحلیل تفاضلی
توان ۵۳
مسعود معصومی و سید مجتبی دهنوی
- معرفی فیلترهایی مناسب برای بهسازی گفتار در حوزه موجک ۶۳
هدی کاویانی شهرضایی، شیما طبیبیان و احمد اکبری

بهبود پروتکل مسیریابی Tora در شبکه‌های ویژه سیار بکمک الگوریتم کولونی مورچه‌ها

رعنا غزالی مجتبی صفاری علی موقر رحیم‌آبادی

دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران

چکیده

الگوریتم TORA یکی از پروتکل‌های بر حسب نیاز مسیریابی در شبکه ویژه سیار می‌باشد. ما پیشنهاد ترکیب پروتکل TORA با الگوریتم Ant با داده و هدف از ارائه و بررسی پروتکل Ant-TORA، استفاده از الگوریتم هوشمند Ant برای تشخیص و استفاده از مسیرهای کوتاه‌تر از نظر زمانی است. همچنین به‌کارگیری چنین الگوریتمی که یک راه حل چندمسیره^۱ برای مسیریابی در شبکه‌های ویژه سیار فراهم می‌کند، تحمل‌پذیری در برابر خطا را نیز افزایش خواهد داد و در نتیجه موجب بهبود کارایی شبکه می‌شود. کارایی پروتکل جدید Ant-TORA را با پروتکل پایه TORA در قالب سناریوهای شبیه‌سازی بررسی کرده‌ایم. همانطور که خواهیم دید پروتکل Ant-TORA کارایی قابل توجهی نسبت به TORA نشان می‌دهد.

کلمات کلیدی: شبکه، ویژه سیار، Ant-TORA-TORA.

۱- مقدمه

شناسه نود مقصد در آن قرار دارد را پخش می‌کند. یک نود با ارتفاع غیرخالی با استفاده از یک بسته UPD که ارتفاع خودش را در آن قرار می‌دهد پاسخ می‌دهد. نودی که یک بسته UPD را دریافت می‌کند ارتفاع خودش را به یکی بیشتر از ارتفاع تولید کننده UPD تنظیم می‌کند. نود با بیشترین ارتفاع به عنوان نود ابتدایی مسیر و نود با کمترین ارتفاع به عنوان نود انتهایی مسیر مطرح می‌شود. در این روش یگ گراف غیرحلقوی (DAG) از مبدا به مقصد ساخته می‌شود. وقتی که یک نود حرکت می‌کند و مسیر DAG شکسته می‌شود ترمیم مسیر برای دوباره ساختن DAG در نود مقصد مورد نیاز است. در فاز از بین بردن مسیر TORA یک بسته پاکسازی (CLR) را در کل شبکه برای از بین بردن مسیرهای غیرمعتبر بصورت Flooding ارسال می‌نماید.

فواید TORA کاهش پیام‌های کنترل به همسایه‌ها زمانیکه تغییرات در توپولوژی رخ می‌دهد می‌باشد. مزیت دیگر آن این است که multicasting را حمایت می‌نماید هر چند که جز عملیات اصلی نمی‌باشد. TORA برای اتصالات با وزن سبک و سازگار با multicast استفاده می‌شود. عیب این روش این است که ممکن است مسیرهای موقت نامعتبر ایجاد نماید.

Mobile ad hoc network (MANET) مجموعه‌ای از نودهای متحرک بی‌سیم است که بصورت پویا و بدون مدیر مرکزی می‌باشد.

۱-۱- الگوریتم TORA

خصوصیت اصلی TORA متمرکز ساختن پیام‌های کنترلی در یک مجموعه بسیار کوچک از نودهای نزدیک به محلی که تغییرات توپولوژی در آن اتفاق افتاده است می‌باشد. برای دستیابی به این موضوع نودها اطلاعات مسیریابی در بازه نودهای مجاور را نگهداری می‌کنند. پروتکل سه وظیفه اساسی ایجاد مسیر و ترمیم مسیر و پاک کردن مسیر را دارد. ایجاد مسیر با استفاده از بسته‌های QRY، UPD انجام می‌شود. الگوریتم ایجاد مسیر با قرار دادن یک مجموعه صفر برای ارتفاع نود مقصد و مجموعه خالی برای ارتفاع نودهای دیگر شروع می‌شود. مبدا بسته QRY که

۱-۲- الگوریتم کولونی مورچه

$$\text{Current Pheromone} = \frac{\text{Pheromone}}{\text{DR}} \times \text{MAX}(0, (1 - \frac{\text{Current Time} - \text{Last Update Time}}{\text{MT}})) \quad (1)$$

با دقت‌تر شدن در فرمول بالا در می‌یابیم که هر چه آهنگ کاهش و حداکثر زمان بیشتر باشند، مقدار فرومون در طول زمان با سرعت بیشتری کاهش می‌یابد. همچنین در صورتی که فاصله زمانی فعلی و زمان آخرین به‌روزرسانی (Last Update Time) بیشتر از حداکثر زمان (MT) شود، مقدار فرومون صفر خواهد شد.

نکته قابل توجهی که باید در نظر گرفته شود این است که پس از هر بار دسترسی به یک سطر و محاسبه مقدار فعلی فرومون، می‌توان محتوای آن سطر را در جدول Ant به‌روز رساند.

۲- مکانیزم واردسازی و بروزرسانی اطلاعات در

جدول Ant

اطلاعات مسیریابی در پروتکل Ant-TORA در سه زمان وارد جدول Ant و بروزرسانی می‌شوند. در ادامه این سه وضعیت شرح داده می‌شوند:

دریافت پیغام درخواست مسیر: هنگامی که گره میانی C بسته درخواست مسیری را که متعلق به گره A است از گره B دریافت می‌کند، در صورتیکه قبلاً زوج (A, B) را به عنوان (Destination Node, Next Node) در جدول Ant نیابد، چهارتایی زیر را در جدول اضافه می‌کند:

(A, B, 1, Current Time)

در صورتیکه این زوج مرتب قبلاً وجود داشته باشد، چهارتایی زیر را در جدول اضافه می‌کند:

(A, B, 1+Δ, Current Time)

که Δ مقدار پیشین فرومون مسیر C به A از طریق B است و همانند فرمول بالا قابل حصول می‌باشد.

دریافت پیغام بروزرسانی مسیر: هر گاه گره C پیغام بروزرسانی مسیری را که از گره مقصد X صادر شده است، از گره D دریافت دارد، در صورتیکه قبلاً زوج (X, D) را به عنوان (Destination Node, Next Node) در جدول Ant نیابد، چهارتایی زیر را در جدول اضافه می‌کند:

(X, D, 1, Current Time)

در صورتیکه این زوج مرتب قبلاً وجود داشته باشد، چهارتایی زیر را در جدول اضافه می‌کند:

(X, D, 1+Δ, Current Time)

که Δ مقدار پیشین فرومون مسیر C به X از طریق D است و همانند فرمول بالا قابل حصول می‌باشد.

همچنین اگر این پیغام بروزرسانی مسیر از طرف یکی از گره‌های میانی صادر شده باشد نیز چنین درج اطلاعاتی در جدول Ant انجام می‌شود؛ با این تفاوت که در این حالت از آنجا که یک گره میانی این بروزرسانی را صادر کرده است، باید جلوی اشتباه در فهم کوتاه‌ترین مسیر را بگیریم. یعنی ممکن است پیغام درخواست و به تبع آن پیغام بروزرسانی متناظر با آن درخواست از مسیری کوتاه‌تر، دیرتر از پیغام بروزرسانی که مربوط به مسیر طولانی‌تری بوده، ولی از گره میانی صادر شده است به گره C برسد. در نتیجه ما می‌توانیم از پارامتر طول مسیر هم برای محاسبه متغیر فرومون استفاده کنیم.

دریافت بسته پاک کننده مسیرهای نامعتبر: هر گاه گره C بسته پاک کننده را که متعلق به گره A است و به مقصد گره X ارسال شده است را برای تحویل به

الگوریتم کولونی مورچه‌ها بر پایه کولونی مورچه‌ها در طبیعت می‌باشد. مورچه‌ها قادر هستند کوتاهترین مسیر بین لانه و منبع غذا را بکمک ترشح فرومون بیابند. در واقع مورچه‌ها در حین راه رفتن ماده‌ای بنام فرومون ترشح می‌نمایند و سایر مورچه‌ها نیز از طریق فرومون ترشح شده توسط مورچه‌های دیگر مسیر را می‌یابند. فرض کنیم مورچه‌ها به نقطه‌ای برسند که باید تصمیم بگیرند سمت چپ بروند یا سمت راست و چون نمی‌دانند کوتاهترین مسیر کدام است در ابتدا بصورت تصادفی سمت چپ یا راست را انتخاب می‌کنند و فرض کنیم ۵۰٪ سمت چپ و ۵۰٪ سمت راست را انتخاب نمایند و تمام مورچه‌ها با سرعت ثابت حرکت می‌کنند، کوتاهترین مسیر توسط مورچه‌های بیشتری پیموده می‌شود لذا فرومون ترشح شده در آن مسیر بیشتر است. پس از دوره زمانی کوتاه تفاوت فرومون موجود در دو مسیر زیاد شده بطوریکه در دفعات بعدی مورچه‌ها مسیری را که فرومون بیشتری دارد با احتمال بالاتری انتخاب می‌نمایند.

۱-۳- الگوریتم Ant-TORA

مؤلفه اصلی پروتکل Ant-TORA، جدولی به نام جدول Ant^۲ است. از جدول Ant برای نگهداری مسیرهای مختلف و تعیین مسیرهای مناسب به سمت مقصد خاص در مواقع لازم استفاده می‌شود. هر سطر این جدول شامل چهار عنصر است: Destination Node: این عنصر، گرهی را مشخص می‌کند که دانستن مسیری به آن مد نظر است. باید توجه داشت که این عنصر در بین سطرها جدول Ant یکتا نیست. یعنی ممکن است در جدول ما چندین سطر مختلف با Destination Node یکسان وجود داشته باشند که هر کدام، مسیر خاصی را به سمت این گره مشخص می‌کنند.

Next Node: این عنصر، اولین گره از یک مسیر به سمت گره Destination Node را نشان می‌دهد. به عبارت دیگر ما با دیدن یک سطر از جدول Ant در می‌یابیم که برای رسیدن به گره Destination Node باید از گره Next Node بگذریم. به این ترتیب، دو عنصر Destination Node و Next Node با یکدیگر یک سطر یکتا را در جدول مشخص می‌کنند.

Pheromone: این عنصر، مقدار فرومون موجود در مسیر Destination Node از گره Next Node را در زمان Last Update Time مشخص می‌نماید. به عبارت دیگر، مقدار فرومون معیاری برای سنجش میزان تازگی و میزان استفاده از مسیر به دست می‌دهد. این متغیر از نوع حقیقی مثبت است.

Last Update Time: این عنصر، آخرین زمانی را نشان می‌دهد که این سطر از جدول به‌روز رسانده شده و مقدار فرومون آن تغییر کرده است. به عبارت دیگر با نگاه کردن به یک سطر ما می‌فهمیم که در لحظه Last Update Time، میزان فرومون فلان مقدار بوده و پس از آن تغییری داده نشده است. این متغیر یک عدد صحیح مثبت است که زمان را بر حسب میلی ثانیه نشان می‌دهد.

ما از دو ثابت دیگر به نام‌های آهنگ کاهش^۳ یا DR و بزرگترین زمان ممکن^۴ یا MT در کنار جدول Ant استفاده می‌کنیم. با به کارگیری این دو ثابت ما می‌توانیم مقدار فعلی فرومون را در هر لحظه تعیین کنیم. ثابت DR، مقداری بزرگتر از یک دارد. ثابت MT هم هر مقداری می‌تواند داشته باشد؛ اما برای عملیاتی شدن سیستم باید مقدار آن را در محدوده طول زمان شبیه‌سازی در نظر گرفت. مقدار MT, DR به ترتیب 1.05 و 200000 فرض شده است.

به عنوان یک قانون کلی، مقدار فرومون یک مسیر در هر لحظه برابر است با:

جدول Ant خود را به روز می‌رساند و هم بهترین مسیر فعلی را جهت ارسال بسته‌های داده‌ای انتخاب می‌کند.

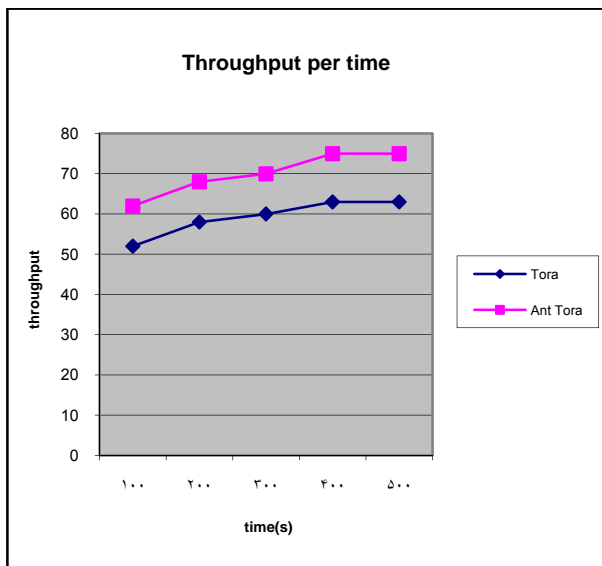
۴- نتایج شبیه‌سازی

از نرم‌افزار شبیه‌ساز شبکه NS2 (Network Simulator) نسخه ۲.۲۷ انجام شده است. برای شبیه‌سازی پروتکل جدید و مقایسه آن با پروتکل TORA از چندین سناریو استفاده شده است. در این سناریوها محدوده محیط شبیه‌سازی مربعی به طول ۱۰۰۰ متر فرض شده است که شامل ۳۰ نود متحرک است. همچنین نحوه حرکت نودها از الگوی Random Waypoint پیروی می‌کند. سرعت حرکت نودها دارای توزیع یکنواخت در یک بازه معین صفر تا ۲۰ m/s است. نود با سرعت مشخص شروع به حرکت به سمت مقصد می‌کند، پس از رسیدن به مقصد برای مدتی در آنجا می‌ماند که به این زمان Pause time (زمان توقف) می‌گویند؛ سپس نقطه دیگری را به عنوان مقصد بعدی انتخاب می‌نماید. نودها این کار را تا پایان شبیه‌سازی تکرار می‌کنند. زمان توقف ۲۵۰ m/s فرض شده است و از مدل CBR جهت ترافیک استفاده شده است.

۴-۱- توان عملیاتی

توان عملیاتی (Throughput) عبارت است از نسبت بسته‌های دریافت شده به بسته‌های ارسال شده.

در نمودار زیر توان عملیاتی دو پروتکل TORA، Ant-TORA در طول زمان شبیه‌سازی با هم مقایسه شده‌اند. همانطور که در شکل پیداست توان عملیاتی دو پروتکل TORA و Ant-TORA با گذشت زمان افزایش پیدا می‌کند؛ علت آن است که به مرور زمان تعداد مسیرهای موجود افزایش یافته و نودها بسته‌های کمتری را از دست می‌دهند.



شکل ۱- نمودار توان عملیاتی بر حسب زمان

توان عملیاتی با افزایش زمان توقف نودها، افزایش می‌یابد. زیرا با افزایش زمان توقف، نودها تحرک کمتری دارند و در نتیجه شکست پیوند کمتر پیش می‌آید پس توان عملیاتی بالا می‌رود. همانگونه که نمودار مشخص می‌کند در حالتی که توقف نودها کم است توان عملیاتی Ant-TORA بهتر از TORA است. زیرا در این

گره میانی بعدی D از گره B تحویل بگیرد در صورتیکه هر یک از زوج‌های (X, D) و (A, B) را در جدول Ant بیابد، چهارتایی‌های زیر را در جدول اضافه می‌کند:

$$\Delta_1 = \Delta_1 \neq 0.1$$

(A, B, Δ_1 , Current Time)

$$\Delta_2 = \Delta_2 \neq 0.1$$

(X, D, $1 + \Delta_2$, Current Time)

Δ_1 و Δ_2 مقادیر پیشین مسیرهای متناظر هستند. هرگاه میزان فرمون در جدول Ant صفر شود چهارتایی متناظر با آن از جدول حذف می‌گردد.

۳- مکانیزم بازیابی اطلاعات از جدول Ant برای

یافتن مسیر

گره نوعی C در چند حالت به جدول Ant خود مراجعه می‌کند و از اطلاعات آن استفاده می‌کند. این حالات به ترتیب شرح داده خواهند شد.

هنگامی که گره C بخواهد بسته‌ای به گره D ارسال نماید. در این حالت در پروتکل TORA، گره C شروع به ایجاد و ارسال پیام درخواست مسیر می‌کند. اما در پروتکل پیشنهادی Ant-TORA، ابتدا جدول Ant برای یافتن تمامی سطرهایی که Destination Node آن‌ها برابر D است، جستجو می‌شود به این ترتیب، جستجوی بسیار وقت‌گیر حجم زیاد اطلاعات سازمان‌نیافته، با جستجوی سریع یک ستون از جدول Ant با اندیس D جایگزین شده است.

اگر یک سطر در جدول Ant یافت شود، باید توجه داشت که با توجه به مکانیزم درج در جدول Ant، اگر مسیری به یک مقصد خاص در این جدول یافت شود، به احتمال زیاد مسیر متناظر با آن سطر وجود خواهد داشت.

پس از یافتن مسیری یا مسیرهای جایگزینی به سمت گره D (مثلاً از طریق گره E) در جدول Ant، حافظه نهان را جستجو نکنیم. به جای آن، آدرس گره بعدی را در بسته برابر گره E قرار داده و بسته را ارسال نماییم. گره E پس از دریافت این بسته به دلیل ناقص بودن مسیر مبدا آن، دوباره جدول Ant خود را جستجو می‌نماید و نظیر گره قبل، بسته را به گره بعدی تحویل می‌دهد. این کار آنقدر تکرار می‌شود تا بسته به مقصد برسد.

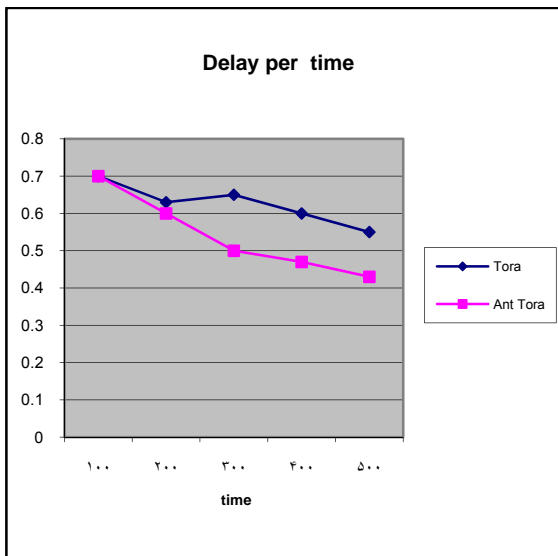
اما اگر چند سطر در جدول Ant یافت شود، فرمون فعلی هر یک از این مسیرها بر اساس فرمول ۱ محاسبه می‌شود. مسیری که بیشترین فرمون فعلی را دارد، برای ارسال بسته‌ها انتخاب می‌شود.

ما برای بهبود بیشتر کارایی TORA در هنگام شکستگی پیوند، Ant-TORA را بدین گونه طراحی کرده‌ایم که گره C پس از تشخیص شکستگی پیوند جدول خود را جستجو می‌کند. اگر مسیر دیگری به سمت مقصد با فرمون بزرگتر از صفر یافت شد. اگر چند مسیر به سمت مقصد یافت شد، فرمون‌های فعلی آن‌ها را محاسبه می‌نماییم و مسیری را که بیشترین فرمون را دارد و در نتیجه تازه‌ترین مسیر است و بیشترین استفاده از آن شده است را انتخاب نمایم. در صورتی که هیچ مسیری در جدول Ant یافت نشود، فرایند درخواست مسیر برای یافتن مسیری برای بسته داده‌ای فراخوانی می‌شود.

هنگامی که گره میانی C، پیام درخواست مسیر گره A را دریافت کرده است و مسیرهای مختلفی به گره مقصد X در جدول Ant خود سراغ دارد. در این حالت گره C، فرمون این مسیرها را محاسبه می‌نماید و مسیرهای متناظر با هر یک از این سطرها را پیدا می‌کند. برای هر مسیر، یک پیام بروزسانی مسیر به گره مبدا فرستاده می‌شود. طبیعی است که متناظر با هر پیام بروزسانی، وزن جاری مسیرها هم بروزسانی خواهد شد. گره مبدا پس از دریافت این بروزسانی‌ها هم

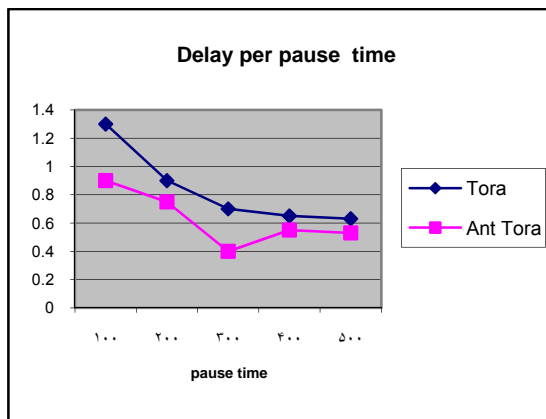
۴-۲- تاخیر

تاخیر (Delay) عبارت است از متوسط اختلاف زمانی بین ارسال بسته‌ها در نود مبدا تا دریافت بسته‌ها در نود مقصد که به آن اصطلاحاً تاخیر انتها به انتها (End-to-End Delay) نیز می‌گویند. تاخیر ارسال بسته‌ها با افزایش زمان کاهش می‌یابد (البته نه به صورت یکنواخت). علتی که می‌توان برای آن متصور بود این است در جدول Ant به مرور زمان مسیرهای اصلی مشخص می‌گردند. در نتیجه نودها سریعتر می‌توانند مسیری به مقصد بیابند و واضح است که هر چه تعداد مسیرها بیشتر باشد احتمال یافتن مسیر کوتاه‌تر افزایش می‌یابد؛ این موضوع باعث کاهش تاخیر می‌شود. آنچه از نمودار پیداست این است که تاخیر Ant-TORA از TORA کمتر است.



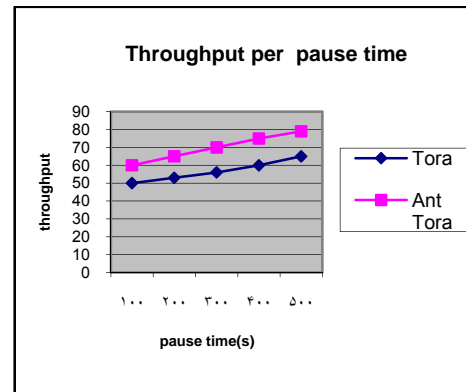
شکل ۵- نمودار تاخیر بر حسب زمان

نمودار تاخیر بر حسب زمان توقف نودها به صورت زیر است. تاخیر ارسال بسته‌ها با افزایش زمان توقف نودها، کاهش می‌یابد. زیرا با افزایش زمان توقف، نودها تحرک کمتری دارند و در نتیجه گسست ارتباط کمتر پیش می‌آید. پس متوسط زمان تاخیر کم می‌شود. همانگونه که نمودار مشخص می‌کند تاخیر Ant-TORA بهتر از TORA است (کمتر از آن است) زیرا Ant-TORA مسیرهای دیگری به مقصد دارد و نیاز به ارسال مجدد پیام درخواست ندارد. به جای آن مستقیماً مسیر دیگری را جایگزین کرده و باعث صرفه‌جویی در وقت می‌شود.



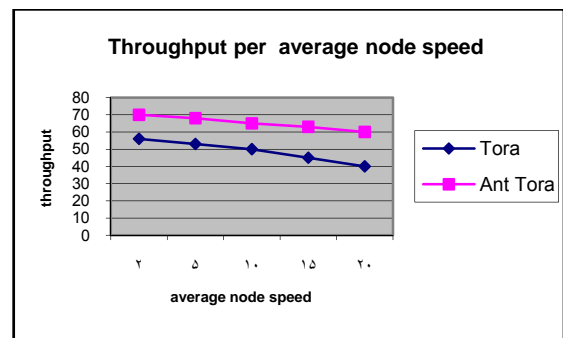
شکل ۶- نمودار تاخیر بر حسب زمان توقف

حالت شکست پیوند به تناوب رخ می‌دهد و Ant-TORA چون مسیرهای دیگری به مقصد دارد نیاز به ارسال مجدد بسته درخواست ندارد. بنابراین توان عملیاتی آن بالاتر است ولی در حالتی که نودها زمان توقف بیشتری دارند شکست پیوند کم اتفاق می‌افتد. بنابراین عملکرد دو پروتکل شبیه به هم می‌شود.



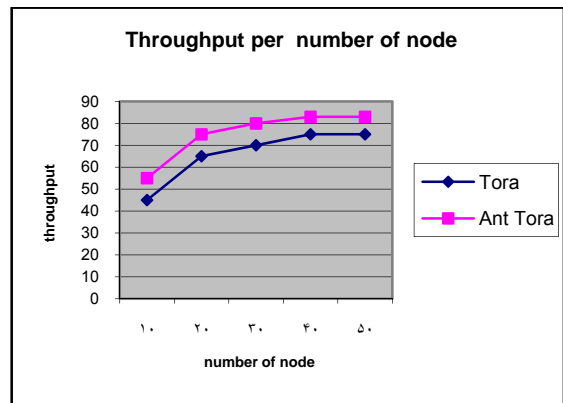
شکل ۲- نمودار توان عملیاتی بر حسب زمان توقف

طبق نمودار، توان عملیاتی با افزایش سرعت نودها، کاهش می‌یابد زیرا با افزایش سرعت، شکست پیوند بیشتر رخ می‌دهد. پس توان عملیاتی پایین می‌آید. در حالتی که سرعت نودها زیاد شود توان عملیاتی Ant-TORA بهتر از TORA است. زیرا Ant-TORA مسیرهای مختلفی به مقصد دارد.



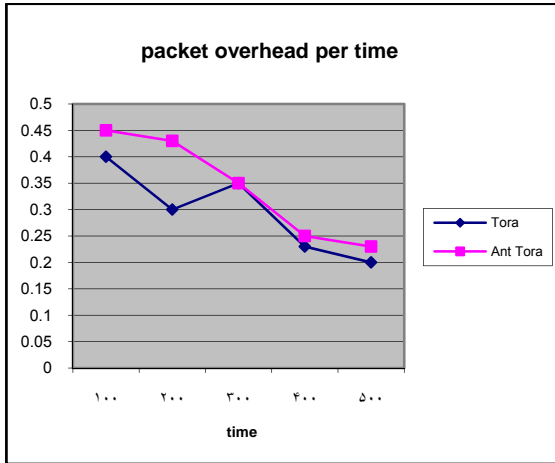
شکل ۳- نمودار توان عملیاتی بر حسب سرعت نودها

توان عملیاتی با افزایش تعداد نودها، افزایش می‌یابد زیرا احتمال وجود مسیرهای بیشتر و کوتاه‌تر افزایش می‌یابد. براساس نمودار، توان عملیاتی Ant-TORA و TORA تقریباً یکسان است.



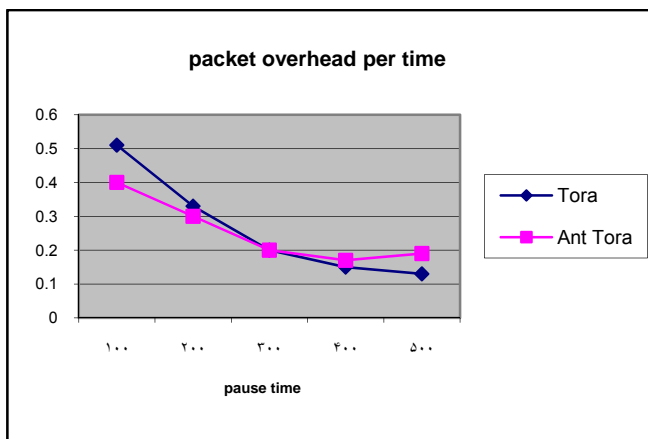
شکل ۴- نمودار توان عملیاتی بر حسب تعداد نودها

مسیر، روزرسانی مسیر، پاک کننده مسیره‌های نامعتبر و بسته‌های Ant نمونه‌هایی از بسته‌های کنترلی هستند. همانطور که در شکل پیداست سربرار بسته‌ها با گذشت زمان کاهش می‌یابد؛ علت آن است که به مرور زمان تعداد مسیره‌های موجود در جدول Ant افزایش پیدا می‌کند. بنابراین نیاز کمتری به ارسال بسته‌های کنترلی برای یافتن مسیر دارند. نکته دیگری که مشاهده می‌گردد این است که سربرار بسته‌ها در Ant-TORA اندکی بیشتر از TORA است. علت این امر، ارسال بسته‌های Ant در شبکه است که باعث افزایش نسبی بسته‌های کنترلی می‌شود.



شکل ۹- نمودار سربرار بر حسب زمان

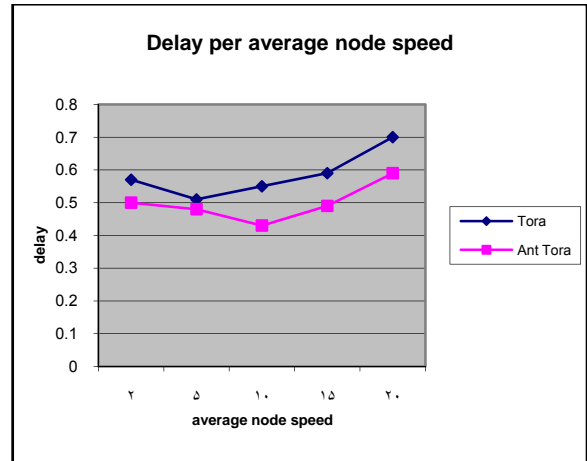
نمودار سربرار بسته‌ها بر حسب زمان توقف نودها به صورت زیر است سربرار بسته‌ها با افزایش زمان توقف نودها، کاهش می‌یابد. زیرا با افزایش زمان توقف، نودها تحرک کمتری دارند و در نتیجه شکست پیوند کمتر پیش می‌آید. پس نیاز به ترمیم مسیر یا درخواست مسیر جدید کم می‌شود و این یعنی کاهش ارسال بسته‌های کنترلی. همانگونه که نمودار مشخص می‌کند وقتی زمان توقف نودها کم است، سربرار بسته‌ها در Ant-TORA کمتر از TORA است زیرا Ant-TORA مسیره‌های دیگری به مقصد دارد و نیاز به ارسال مجدد درخواست مسیر ندارد.



شکل ۱۰- نمودار سربرار بر حسب زمان توقف

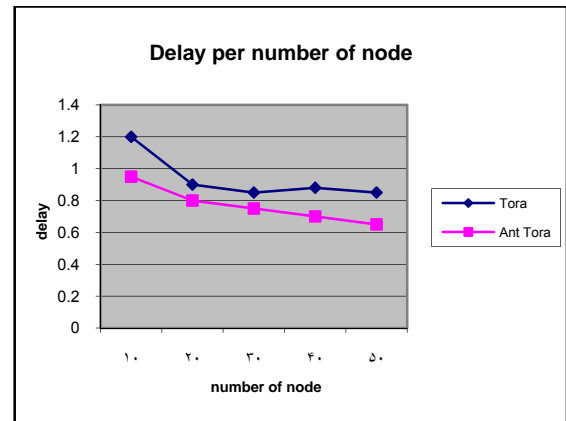
سربرار بسته‌ها با افزایش سرعت نودها، افزایش می‌یابد؛ همچنین بر اساس نمودار، وقتی که سرعت حرکت نودها زیاد است، سربرار بسته‌ها در پروتکل Ant-TORA کمتر از پروتکل TORA است. ولی وقتی که سرعت حرکت نودها کم است، سربرار بسته‌ها در هر دو پروتکل تقریباً یکسان می‌باشد. علت این امر، شکست

نمودار تاخیر بر حسب متوسط سرعت نودها در زیر آمده است. طبق نمودار، تاخیر با افزایش سرعت نودها، افزایش می‌یابد زیرا با افزایش سرعت، شکست پیوند بیشتر رخ می‌دهد پس تاخیر در ارسال بسته‌ها بیشتر می‌شود. همانگونه که نمودار مشخص می‌کند تاخیر Ant-TORA کمتر از TORA است زیرا Ant-TORA مسیره‌های مختلفی به مقصد دارد و همچنین احتمال یافتن مسیر کوتاهتری به مقصد در آن بیشتر است؛ همین امر باعث می‌شود بسته‌ها مدت زمان کمتری در صف ارسال یا در طول مسیر ارسال باشند.



شکل ۷- نمودار تاخیر بر حسب میانگین سرعت نودها

تاخیر ارسال بسته‌ها بر حسب تعداد نودها در نمودار زیر آورده شده است تاخیر ارسال بسته‌ها با افزایش تعداد نودها، کاهش می‌یابد. زیرا در این حالت احتمال وجود مسیره‌های مختلف و کوتاه بیشتر است. بنابراین بسته‌ها زیاد معطل نمی‌مانند. همانند نمودارهای قبل، در این نمودار نیز مشخص است که تاخیر Ant-TORA از TORA کمتر است. در حالت کلی می‌توان گفت که تاخیر ارسال بسته‌ها در پروتکل Ant-TORA از پروتکل TORA کمتر است زیرا داشتن مسیره‌های مختلف به مقصد باعث کاهش زمان انتظار ارتباط می‌شود.



شکل ۸- نمودار تاخیر بر حسب تعداد نودها

۳-۴- سربرار بسته‌ها

سربرار بسته‌ها (Packet Overhead) عبارت است از نسبت بسته‌های کنترلی ارسال شده توسط نودها به کل بسته‌های ارسالی. بسته‌های کنترلی به بسته‌هایی گفته می‌شود که برای ساخت و ترمیم مسیر استفاده می‌شوند. بسته‌های درخواست

مسیریابی در ANT_TORA کمتر از TORA می‌باشد و سریار مسیریابی ANT_TORA کمتر از TORA می‌باشد.

سریع پیوندها در سرعت زیاد است. وقتی که شکست پیوند رخ می‌دهد، برای ترمیم مسیر اگر مسیر جایگزینی موجود نباشد، نیاز است که بسته‌های کنترلی ارسال شوند.

مراجع

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial intelligence*, Oxford University Press, 1999.

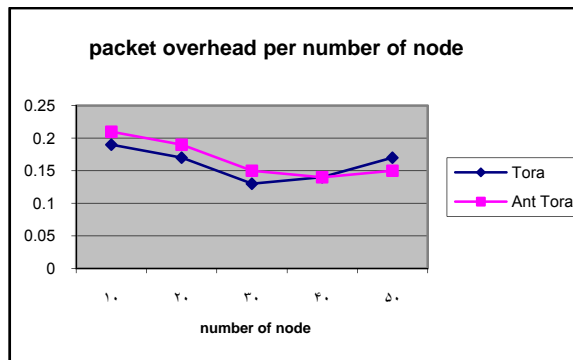
[2] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multihop wireless ad hoc network routing protocols," *Proc. of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 85-97, 1998.

[3] M. Dorigo, and G. D. Caro, "The ant colony optimization meta-heuristic," *New Ideas in Optimization*, pp. 11-32. McGraw-Hill, London, 1999.

[4] K. Fall, and K. Varadhan, *The ns Manual*, Nov 2000.

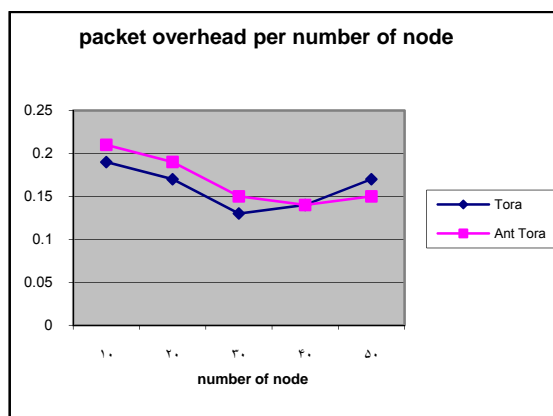
[5] V. Park, and M. S. Corson, "Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification," <http://tools.ietf.org/html/draft-ietf-manet-tora-spec-04>.

[6] J. P. Macker, and M. S. Corson, "Mobile ad hoc networking and the IETF," *Mobile Computing and Communications Review*, vol. 2, no. 1, pp. 9-14, 1998.



شکل ۱۱- نمودار سریار بر حسب میانگین سرعت نودها

در اینجا نمودار سریار بسته‌ها برای تعداد متغیر نودها رسم شده است. از نمودار می‌توان استنباط کرد که با افزایش تعداد نودها از ۱۰ نود به ۳۰ نود، سریار بسته‌ها کاهش می‌یابد. ولی با افزایش بیشتر نودها به ۶۰ نود، سریار بسته‌ها اندکی افزایش می‌یابد. این مطلب برای هر دو پروتکل تقریباً یکسان است. علتی که برای این امر وجود دارد این است که وقتی تعداد نودها کم است، تراکم نودها کم است. بنابراین احتمال اینکه نودها در فضای قابل دسترس یکدیگر قرار گیرند، کم می‌شود و برای ایجاد ارتباط نیاز به ارسال بسته‌های کنترلی بیشتری است. وقتی تعداد نودها از حد خاصی بیشتر می‌شود، هم شکست در پیوندها زیاد اتفاق می‌افتد هم بسته‌های کنترلی در محدوده گسترده‌تری پخش می‌شوند که همین امر باعث افزایش سریار بسته‌ای می‌شود.



شکل ۱۲- نمودار سریار بر حسب تعداد نودها



رعنا غزالی در حال حاضر مدرس دانشگاه جامع علمی کاربردی می‌باشد که از سال ۱۳۸۴ مشغول به تدریس می‌باشد. مدرک کارشناسی در رشته مهندسی کامپیوتر نرم‌افزار را در سال ۱۳۷۹ از دانشگاه آزاد اسلامی واحد تهران جنوب اخذ نموده و مدرک کارشناسی ارشد در رشته مهندسی کامپیوتر نرم‌افزار را در سال ۱۳۸۴ از دانشگاه آزاد واحد اراک اخذ نموده است. همچنین سه مقاله در کنفرانس بین‌المللی انجمن کامپیوتر ایران به چاپ رسانده است و دو کتاب تحت عنوان سیستم عامل‌ها و ریاضی در علم کامپیوتر را تالیف نموده است. زمینه‌های مورد علاقه ایشان شبکه‌های بی‌سیم سیار و هوش مصنوعی می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

ra_arpanet2000@yahoo.com

مجتبی صفاری تا زمان چاپ مقاله بیوگرافی نویسنده دریافت نشده است.

آدرس پست‌الکترونیکی ایشان عبارت است از:

m_saffari@yahoo.com



علی موقر رحیم‌آبادی در حال حاضر استاد دانشکده مهندسی کامپیوتر در دانشگاه صنعتی شریف است که در سال ۱۳۷۳ در این دانشگاه به عنوان استادیار شروع به کار کرد. او مدرک کارشناسی را در رشته مهندسی برق از دانشکده فنی دانشگاه تهران در سال ۱۳۵۶ و مدارک کارشناسی ارشد و دکتری را در رشته مهندسی کامپیوتر، اطلاعات و کنترل از دانشگاه میشیگان به ترتیب در سال‌های ۱۳۵۸ و ۱۳۶۴ دریافت نمود. او در

برای بهبود توان عملیاتی و تاخیر ارسال بسته‌ها، ما پیشنهاد ترکیب پروتکل TORA با الگوریتم Ant را داده و ضمن معرفی این الگوریتم مکانیزم درخواست مسیر و بروزرسانی اطلاعات جداول مسیریابی و روند مسیریابی بکمک این روش را شرح دادیم.

نتایج شبیه‌سازی نشان داده است که روش ANT_TORA نسبت به TORA درحالتی که تحرک نودها بالاست کارایی بهتری داشته و همچنین میزان تاخیر

۵- نتیجه‌گیری

موسسه تحقیقاتی INRIA فرانسه به عنوان محقق در سال ۱۳۶۳، در آزمایشگاه‌های بل به عنوان محقق فنی در سال‌های ۱۳۶۴ و ۱۳۶۵ و در دانشگاه میشیگان به عنوان مدرس در سال‌های ۱۳۶۶ الی ۱۳۶۸ مشغول به کار بود. زمینه‌های مورد علاقه حرفه‌ای ایشان مدلسازی کارایی و اتکاپذیری، درستی‌یابی و اعتبار سنجی، شبکه‌های بی‌سیم سیار، و سیستم‌های بی‌درنگ توزیع شده می‌باشد. ایشان عضو ارشد سازمان‌های IEEE و ACM نیز هستند. آدرس پست‌الکترونیکی ایشان عبارت است از:

movaghar@sharif.edu

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۸/۶/۴

تاریخ اصلاح: ۸۹/۶/۱۶

تاریخ قبول شدن: ۸۹/۷/۱۸

نویسنده مرتبط: دکتر علی موقر رحیم‌آبادی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران.

-
- ¹ Multipath
 - ² Ant Table
 - ³ Decline Rate
 - ⁴ Max Time

کاوش متون فارسی بر مبنای روش طبقه‌بندی

محمد حسین سرایی آذر شاهقلیان

دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان، اصفهان، ایران

چکیده

سازمان‌دهی اطلاعات On-line در قالب طبقه‌بندی مستندات زبان طبیعی در دسته‌های از قبل مشخص شده یکی از روش‌های مهم مدیریت اطلاعات محسوب می‌شود. با توجه به اهمیت موضوع و کاری که در این زمینه برای زبان‌های دیگر دنیا انجام گرفته است، نیاز به طبقه‌بندی متون فارسی به خوبی احساس می‌شود. در این مقاله از یادگیری ماشین برای ارائه روشی در طبقه‌بندی متون فارسی استفاده می‌شود. روش ارائه شده، تحت سیستم نرم‌افزاری طبقه‌بند متون فارسی، طراحی و پیاده‌سازی شده است. سیستم طبقه‌بند متون فارسی در فاز یادگیری، مجموعه‌ای از متون آموزشی را برای استخراج ویژگی‌های دسته‌ها بررسی می‌کند تا خصوصیات اصلی هر دسته را بدست آورد. به‌طوریکه در فاز تست سیستم طبقه‌بند متون فارسی، این ویژگی‌های مختص دسته، برای طبقه‌بندی متون دسته‌بندی نشده به کار می‌روند. از ریشه‌یابی برای کاهش بعد بردارهای ویژگی استفاده می‌شود. دقت روش پیشنهادی روی مجموعه جمع‌آوری شده‌ای از اخبار فارسی در هفت دسته مورد آزمایش قرار گرفته است. نتایج حاصله نشان می‌دهد که طبقه‌بند پیشنهادی برای دسته‌های اقتصادی، سیاسی و ورزشی دقت بسیار خوبی دارد. در سایر دسته‌ها نیز نتایج مورد قبول می‌باشد.

کلمات کلیدی: یادگیری ماشین، Text Mining، الگوریتم KNN، زبان فارسی.

۱- مقدمه

این وظیفه بطور معمول تحت عنوان دسته‌بندی متون^۲ مورد بررسی قرار گرفته و به عنوان زیرمجموعه‌ای متن‌کاوی محسوب می‌شود. از جمله کاربردهای طبقه‌بندی می‌توان به طبقه‌بندی صفحات وب، شاخص‌گذاری آیت‌های خبری در منابع مختلف اینترنتی و دسته‌بندی موضوعی در زمینه‌های تجارت، پزشکی و بیوانفورماتیک اشاره نمود.

روش یادگیری ماشین^۳ با ساختن قوانین طبقه‌بندی می‌تواند بر همه این کمبودها فائق آید. بر مبنای این روش، تعدادی از مستندات آموزشی طبقه‌بندی شده به صورت دستی، داده می‌شود. قوانین طبقه‌بندی متن باید به نحوی آموزش داده شوند که بتوانند مسأله یادگیری بانظارت (supervised) را به راحتی حل کنند [۳]. فرایند استنتاج عمومی به صورت اتوماتیک، قوانین را بوسیله یادگیری مشخصه‌های دسته‌ها از مجموعه مستندات از قبل برچسب‌گذاری شده، می‌سازد.

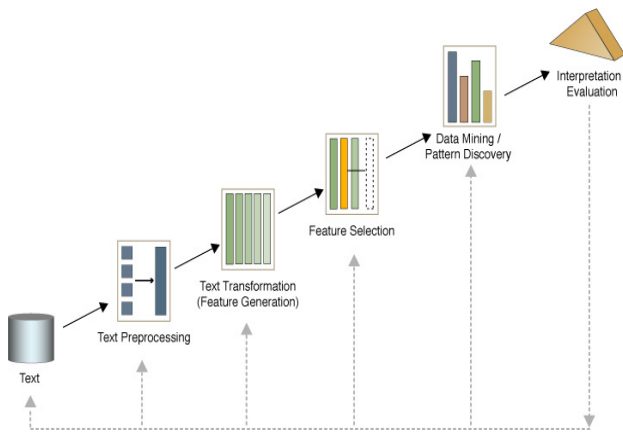
طبقه‌بندی متن که اغلب کلاسه‌بندی^۴ نامیده می‌شود، متون زبان طبیعی را به یک یا بیشتر دسته‌های از قبل معرفی شده بر اساس محتوی نسبت می‌دهد [۳][۴]. مراحل انجام کار طبقه‌بندی به این صورت است که ابتدا پردازش

واضح است که در سازمان‌های جدید، دانش^۱ فاکتور کلیدی برای رقابت سالم می‌باشد. دستیابی به موفقیت و باقی‌ماندن در صحنه رقابت به صورت گسترده به قابلیت یافتن اطلاعات مفید در زمان مناسب بستگی دارد. از طرفی رشد اطلاعات و تکنولوژی‌های ارتباطی به صورت بیش از اندازه باعث افزایش اطلاعات دردسترس شده است. لذا، فاکتور کلیدی، تنها خود اطلاعات نیست؛ بلکه قابلیت مدیریت و اعمال نفوذ بر این اطلاعات به صورت موفق، از اهمیت بالایی برخوردار است [۱].

توجه به این نکته ضروری است که بخش قابل توجهی از اطلاعات موجود در پایگاه‌های داده‌ای متنی، ذخیره شده‌اند. لذا از Text Mining برای مقایسه متون مختلف، رتبه‌بندی مهمترین و مرتبطترین متون و یا یافتن الگوها و رفتارهای بین متون مختلف استفاده می‌شود. متن‌کاوی عهده‌دار این وظایف می‌باشد [۲]. یکی از این وظایف طبقه‌بندی مستندات زبان طبیعی به طبقه‌های از قبل مشخص شده می‌باشد که از جمله روشهای مهم برای سازماندهی اطلاعات On-line می‌باشد.

فازهای یادگیری ماشین برای طبقه‌بندی متن مطابق شکل ۱ می‌باشد. همانگونه که در شکل مشاهده می‌شود، طی ۵ فاز، سیستم یادگیری ماشین برای انجام طبقه‌بندی به کار گرفته می‌شود. این فازها عبارتند از:

- پیش‌پردازش (Preprocessing)
- تولید ویژگی (Feature Generation) (Feature Selection)
- انتخاب ویژگی (Feature Selection)
- اعمال الگوریتم‌های داده‌کاوی / متن کاوی (Data Mining/Pattern Discovery)
- تفسیر و ارزیابی (Interpretation /Evaluation)



شکل ۱- فازهای سیستم یادگیری ماشین برای طبقه‌بندی

۴- آماده‌سازی متون سیستم طبقه‌بند متون فارسی

داده‌های ورودی به برنامه طبقه‌بند متون، عموماً متونی هستند که از سایت‌های خبری فارسی جمع‌آوری شده‌اند. بررسی کلی روی هفت طبقه زیر می‌باشد:

- ۱- اجتماعی ۲- اقتصادی ۳- پزشکی ۴- سیاسی ۵- فرهنگی ۶- مذهبی
- ۷- ورزشی

برای هر یک از این طبقه‌ها ۱۰۰ متن در نظر گرفته شده است. از هر کدام از این ۱۰۰ متن، ۸۰ متن برای فاز یادگیری و ۲۰ متن دیگر برای فاز تست استفاده خواهد شد. با توجه به اینکه این متون در ویرایشگرهای فارسی متفاوتی تایپ شده است و نیز در هنگام بارگذاری روی اینترنت ممکن است دچار تغییراتی شده باشد، لازم است در ابتدا بازبینی کلی روی متون انجام شود. قسمتی از این بازبینی به شیوه دستی انجام شده است که اهم آنها عبارتند از:

- دسته‌بندی متون در گروه‌های مربوطه و تعیین اسم متون بر مبنای "شماره + حرف اختصاری نمایانگر آن گروه".
- برطرف نمودن غلط‌های املایی تا حد امکان.
- چک کردن رعایت فاصله صحیح بین کلمات.
- یکسان نویسی حروف و علائم ریاضی.
- معرفی افعال، پیشوندها، پسوندها، حروف ربط، اضافه و نشانه، افعال ربطی، علائم نقطه‌گذاری و ضمیر به صورت فایل‌های جداگانه به عنوان ورودی سیستم [۷].

۵- پیش‌پردازش سیستم طبقه‌بند متون فارسی

پس از آماده‌سازی اولیه متون، فاز پیش‌پردازش انجام می‌شود. در واقع پیش‌پردازش، اولین گام در جهت تطابق مستندات متنی با نمایش آنها در یک

زبان طبیعی^۵ انجام شده، سپس با به کارگیری روش یادگیری اتوماتیک، این نتایج تفسیر می‌شود. وظایفی نظیر بازیابی متن و طبقه‌بندی متن از این تفسیر استفاده می‌کنند [۳] [۵]. هدف نهایی، طبقه‌بندی متون در تعداد ثابتی از دسته‌های از قبل معرفی شده می‌باشد. با توجه به این تفاسیر طبقه‌بندی‌متون به عنوان زیرمجموعه‌ای از Text Mining محسوب می‌شود.

۲- مروری بر دیدگاه‌ها و روش‌های موجود طبقه‌بندی متون

هنگام بررسی طبقه‌بندی متون، اشاره به انواع طبقه‌بندی‌های موجود لازم است. در کل طبقه‌بندی به دو دسته انحصاری و غیر انحصاری تقسیم می‌شود. در طبقه‌بندی انحصاری، هر شیء دقیقاً به یک دسته وابسته می‌شود. در حالیکه در طبقه‌بندی غیر انحصاری می‌تواند به چند دسته اختصاص یابد و اصطلاحاً در این مورد گفته می‌شود که همپوشانی دارد. برای مثال، دسته بندی گروهی از افراد با وزن یا قد یکسان، از نوع انحصاری است و دسته‌بندی گروهی از افراد دارای بیماری، غیرانحصاری می‌باشد. زیرا یک شخص می‌تواند بطور همزمان به چند بیماری دچار شود.

طبقه‌بندی انحصاری می‌تواند به دو زیر شاخه تقسیم شود:

۱- ذاتی یا intrinsic / Unsupervised

۲- خارجی: extrinsic یا Supervised

تفاوت میان این دو دسته در این است که در دسته دوم، دسته‌های از قبل مشخص شده را برای طبقه‌بندی اشیاء (یعنی مستندات) مورد استفاده قرار می‌دهد. در حالیکه اولین گروه، یک راه پیش‌بینی شده را برای طبقه‌بندی مشخص می‌کند.

زیرشاخه ذاتی^۶ خود به دو زیر شاخه سلسله‌مراتبی و partitional تقسیم می‌شود. سلسله‌مراتبی را "nested sequence of partitions" می‌نامند، در حالیکه partitional یک single partition است. لازم به ذکر است که اصطلاحات خوشه‌بندی و خوشه‌بندی سلسله‌مراتبی به ترتیب برای Unsupervised و partitional و طبقه‌بندی سلسله‌مراتبی به کار رفته است. در این مقاله، روش Supervised استفاده شده است.

۳- روش پیشنهادی برای طبقه‌بندی متون فارسی

روش پیشنهادی برای طبقه‌بندی متون فارسی بر اساس روش یادگیری ماشین استوار است. در روش یادگیری ماشین، دو فاز آموزش و تست وجود دارد. در فاز آموزش دسته‌های از قبل مشخص شده‌ای را برای یادگیری ماشین استفاده می‌کنند و معنای هر طبقه برای سیستم یادگیری ماشین مشخص می‌شود. در فاز تست، مستندات شناخته نشده‌ای به سیستم داده می‌شود. سیستم به طور اتوماتیک آن متن را به طبقه‌ای که بیشتر شباهت دارد نسبت می‌دهد.

قانون اصلی یادگیری ماشین این است که یک فضای وسیع از فرضیات ممکن و دانش پیشین نگه‌داشته شده بوسیله یادگیر را جستجو نموده و بهترین طبقه ممکن را برای متن تست بدست آورد. وظیفه یادگیر جستجو در این فضا برای جایگذاری فرضیاتی است که بیشترین سازگاری را با نمونه‌های آموزشی موجود دارد.

عموماً در طراحی الگوریتم یادگیری ماشین باید انتخابی‌هایی انجام شود. این انتخاب‌ها شامل انتخاب نوع داده آموزشی، تابع هدف، نمایش آن و یک الگوریتم برای یادگیری این تابع از نمونه های آموزشی می‌باشد [۶].

از فرمول‌های تولید ویژگی Entropy, LTC, IDF برای تولید ویژگی‌های هر متن استفاده می‌شود. البته فرکانس خام هم برای مقایسه و ارزیابی به کارگیری این فرمول‌های تولید ویژگی نسبت به کار با فرکانس تکرار کلمات در نظر گرفته می‌شود. اگر f_{ik} به عنوان وزن کلمه i در متن k باشد، راه‌های متعددی برای تصمیم‌گیری در مورد آن وجود دارد. اما بیشتر روش‌ها بر مبنای یکی از دو روش زیر استوار است:

بیشترین تعداد باری که کلمه در متن اتفاق می‌افتد.

بیشترین تعداد باری که کلمه در همه متون موجود در مجموعه اتفاق می‌افتد.

$$\text{Tf*IDF Weighting: } a_{ik} = f_{ik} * \log\left(\frac{N}{n_i}\right) \quad (1)$$

Ltc- Weighting:

$$a_{ik} = \frac{\log(f_{ik} + 1.0) * \log\left(\frac{N}{n_i}\right)}{\sqrt{\sum_{j=1}^M \left[\log(f_{ij} + 1.0) * \log\left(\frac{N}{n_i}\right) \right]^2}} \quad (2)$$

Entropy-Weighting:

$$a_{ik} = \log(f_{ik} + 1.0) * \left(1 + \frac{1}{\log(N)} \sum_{j=1}^N \left[\frac{f_{ij}}{n_i} \log\left(\frac{f_{ij}}{n_i}\right) \right] \right) \quad (3)$$

به طوریکه $\frac{1}{\log(N)} \sum_{j=1}^N \frac{f_{ij}}{n_i} \log\left(\frac{f_{ij}}{n_i}\right)$ Average uncertainty یا

entropy کلمه i می‌باشد.

در این فرمول‌های وزن‌دهی، تعاریف زیر وجود دارند:

f_{ik} : فرکانس کلمه i در متن k

N : تعداد متون در مجموعه

M : تعداد کلمات مجموعه پس از انجام عملیات کاهش و حذف کلمات اضافی

n_i : مجموع تعداد دفعاتی که کلمه i در هر مجموعه اتفاق افتاده است

۷- انتخاب ویژگی متون در سیستم طبقه‌بند متون

فارسی

در قسمت قبل، ویژگی‌های هر کدام از کلمات موجود در متون پیدا شد. با توجه به اینکه هنوز تعداد این ویژگی‌ها بسیار زیاد است باید از بین آنها، تعدادی انتخاب شوند به طوری که این ویژگی‌ها، ویژگی‌های مؤثری در طبقه‌بندی باشند. انتخاب ویژگی‌ها باعث کاهش سربار فضا و زمان برای پیاده‌سازی الگوریتم‌های طبقه‌بندی خواهد شد. برای انتخاب ویژگی در سیستم طبقه‌بند متون فارسی از یک حد آستانه^۹ استفاده می‌شود که این حد آستانه قابل تنظیم می‌باشد. البته با توجه به اینکه ثابت شده است تنها ۳۳٪ کلمات موجود در متن برای طبقه‌بندی مناسب هستند، پیشنهاد می‌شود که این حد آستانه بین ۰ تا ۳۰ انتخاب شود. انتظار می‌رود در این محدوده بهترین جواب‌ها بدست آیند.

۸- پیاده‌سازی الگوریتم KNN

پس از انجام مراحل پیش‌پردازش، تولید ویژگی و انتخاب ویژگی در سیستم طبقه‌بند متون فارسی که در قسمت‌های قبلی تشریح شد، نوبت به پیاده‌سازی الگوریتم‌های طبقه‌بندی می‌رسد.

قالب مناسب می‌باشد. ثابت شده است که تنها ۳۳٪ کلمات در یک متن مفید هستند و می‌توان از آنها برای استخراج اطلاعات استفاده نمود [۸]. اغلب کلمات در راستای رساندن منظور و هدف اصلی استفاده می‌شوند و بعضاً تکراری می‌باشند. در نتیجه هدف از این فاز، یافتن کلمات مفید^۷ و چشم‌پوشی از کلمات بی‌فایده می‌باشد. در این فاز، عملیات کلی زیر انجام می‌شود:

الف: تبدیل هریک از متون به برداری از کلمات

ب: پیدا کردن پایان جملات با استفاده از تشخیص افعال و حروف ربط

ج: اطمینان از یکسان بودن کاراکترهای الفبای فارسی (یکسان‌سازی برخی از کاراکترها مانند "ی" و "ک" از لحاظ کد اسکی)

د: تفکیک جملات از یکدیگر

ه: از بین بردن کلمات بی‌فایده (حروف ربط، اضافه، نشانه، علائم نقطه‌گذاری، ضمائر، افعال ربطی، شبه‌جمله‌ها و ...)

و: انجام عملیات کاهش و ریشه‌یابی^۸ با توجه به دو گروه کلمات، افعال و اسامی:

✓ بررسی افعال

- بررسی پیشوندهای فعل در دو قسمت با تغییر معنی و بدون تغییر معنی. مثال: فراگرفتن = فرا + گرفتن

- حذف شناسه افعال و بررسی امکان پذیر بودن کاهش با استفاده از جدول بن و مصدر.

- جایگزین نمودن فعل با مصدرش با استفاده از جدول افعال.

مثال: می‌گفتم ← گفتم ← گفت ← گفتن

✓ بررسی اسامی

- کاهش علامات جمع (مانند ها، های، ...)

- حذف ضمائر متصل

- حذف پیشوند

- حذف پسوندهای اسم

- مشتق

- مرکب و ساده

لازم به ذکر است که در کاهش اسامی از منطق فازی استفاده شده است و بهترین کاهش بدست آمده برای کلمه به عنوان جایگزین کلمه استفاده می‌شود.

۶- تولید ویژگی متون در سیستم طبقه‌بند متون

فارسی

پس از انجام فاز پیش‌پردازش در سیستم طبقه‌بند متون فارسی، متون از رشته‌ای از کلمات به بردارهایی از کلمات تبدیل می‌شوند. در این بردارها کلمات بی‌فایده وجود ندارند. البته این بردارها، شامل کلمات یکتایی نیستند [۹]. برای برطرف کردن این مشکل و آماده‌سازی یک ورودی مناسب برای الگوریتم یادگیری طبقه‌بندی متون، از متد تولید ویژگی استفاده شده است. در این متد، کلمات تکراری دسته‌بندی شده و فرکانس آنها محاسبه می‌شود. اینکار بوسیله پیاده‌سازی یک hash table انجام می‌گیرد. Hash table یک نگاشت از کلیدها (key) به ارزش‌ها (value) را فراهم می‌آورد. در اینجا کلمات به عنوان کلید و فرکانس آنها ارزش محسوب می‌شوند.

لازم به ذکر است که همه کلماتی که از فاز پیش‌پردازش به دست آمده‌اند، برای طبقه‌بندی متن لازم نیستند. در این قسمت باید ویژگی‌های شاخص هر متن را استخراج نموده تا بتوان به کمک آنها متون جدید را به بهترین کلاس تطبیق داد.

۹- ارزیابی الگوریتم KNN

پس از پیاده‌سازی الگوریتم KNN باید با استفاده از متون تست، کارایی این الگوریتم مورد ارزیابی قرار گیرد. همانگونه که در قسمت سه گفته شد این متون از روی منابع اینترنتی مختلفی جمع‌آوری شده است و سعی شده است که از نظر پراکندگی به‌طور مناسبی انتخاب شوند. داده‌های ارزیابی شامل ۱۴۰ متن برای تست می‌باشد که برای هر دسته ۲۰ متن تست در نظر گرفته شده است. این متون به عنوان ورودی تست به برنامه داده می‌شود. ارزیابی الگوریتم طبقه‌بند براساس پارامترهای زیر صورت می‌گیرد [12][13]:

$$\begin{aligned} \Delta) \text{ Recall} &= a / (a + c) \\ \text{Precision} &= a / (a + b) \\ \text{Fall out} &= b / (b + d) \\ \text{Error rate} &= (b + c) / (a + b + c + d) \end{aligned}$$

بطوریکه

a: تعداد نمونه‌های عضو کلاس و درست تشخیص داده شده

b: تعداد نمونه‌های عضو کلاس و اشتباه تشخیص داده شده

c: تعداد نمونه‌های غیرعضو کلاس و درست تشخیص داده شده

d: تعداد نمونه‌های غیر عضو کلاس و اشتباه تشخیص داده شده

$$n = a + b + c + d \text{ تعداد کل متن‌های تست}$$

با اجرای برنامه روی داده‌های تست، دقت برای الگوریتم‌های فوق بدست آمده است. برای هر دسته، بهترین الگوریتم به‌صورت پررنگ مشخص شده است. نتایج در جدول ۱ نشان داده شده است. دقت الگوریتم و نمودار کیفیت به‌ترتیب در شکل‌های ۲ و ۳ نمایش داده شده است. همانطور که در جدول ۱ ملاحظه می‌شود دقت الگوریتم KNN برای هر هفت دسته نمایش داده شده است. برای دسته‌های اقتصادی، اجتماعی، فرهنگی، پزشکی و سیاسی بر اساس فرمول تولید ویژگی LTC بهترین دقت را بدست می‌آورد. دقت الگوریتم در بهترین حالت برای دسته ورزشی ۱۰۰٪ است که فرمول تولید ویژگی TxIDF آنرا تولید کرده است. دسته مذهبی در مقایسه با سایر دسته‌ها دقت خوبی را نشان نمی‌دهد و دقت الگوریتم با کاربرد TxIDF برابر با ۶۰٪ می‌باشد.

در شکل ۲، نمودار میله ای دقت الگوریتم KNN برای هر دسته آورده شده است. همانطور که در شکل قابل مشاهده است، همانطور که مشاهده می‌شود دقت الگوریتم برای دسته ورزشی برای کلیه روش‌های تولید ویژگی تقریباً تغییرات چندانی ندارد و در دسته مذهبی تغییرات چندانی محسوس نمی‌شود.

جدول ۱- دقت الگوریتم KNN برحسب فرمول‌های تولید ویژگی

دسته	Raw			TxIDF			LTC			Entropy		
	K Minimax	Average	Smallest	K Minimax	Average	Smallest	K Minimax	Average	Smallest	K Minimax	Average	Smallest
اقتصادی	۷۳.۳۳	۷۵	۵۵	۶۵	۴۰	۶۰	۱۰۰	۹۰	۱۰۰	۶۵	۹۰	۵۵
اجتماعی	۴۱.۶۷	۴۰	۴۰	۴۰	۱۵	۵۰	۳۰	۷۵	۳۰	۳۵	۶۵	۲۵
فرهنگی	۶۰.۰۰	۷۰	۲۰	۶۰	۱۰	۷۵	۸۰	۱۰۰	۷۵	۷۵	۳۵	۵۵
پزشکی	۷۷.۶۷	۸۸	۶۰	۸۰	۴۴	۸۰	۸۰	۹۶	۶۸	۹۲	۹۲	۷۶
مذهبی	۳۶.۲۵	۲۵	۴۰	۳۰	۱۰	۳۵	۴۰	۴۰	۳۵	۴۰	۴۰	۴۰
سیاسی	۵۷.۵۰	۵۵	۳۵	۴۰	۰	۴۰	۹۵	۸۵	۷۵	۷۵	۶۵	۷۵
ورزشی	۸۴.۵۸	۷۵	۹۰	۸۰	۱۰۰	۷۰	۸۰	۸۰	۸۰	۸۵	۹۰	۸۵
جمع	۴۳۱	۴۲۸	۳۴۰	۳۹۵	۲۱۹	۴۱۰	۵۰۵	۵۶۶	۴۶۳	۴۷۶	۴۷۷	۴۱۱

KNN به خاطر سادگی و مؤثر بودن آن در طبقه‌بندی متون بسیار به‌کار برده می‌شود. مبنای کار این الگوریتم، مقایسه متن تست داده شده با متون آموزشی داده شده و بدست آوردن میزان شباهت بین آنها می‌باشد [۱۰]. متون آموزشی با n ویژگی موجود می‌باشد. هر متن به عنوان یک نقطه در یک فضای n بعدی نمایش داده می‌شود. هنگامی که متن ناشناخته‌ای را الگوریتم دریافت می‌کند، فضای الگو را برای یافتن متون آموزشی که شبیه آن متن ناشناخته باشند، جستجو می‌کند. در اینجا از فاصله اقلیدسی به عنوان معیار شباهت استفاده شده است [۱۱].

فاصله اقلیدسی بین دو متن یا دو نقطه $X=(x_1, x_2, x_3, \dots, x_n)$ و $Y=(y_1, y_2, y_3, \dots, y_n)$ با استفاده از فرمول (۴) محاسبه می‌شود:

$$D = \sum_{i=1}^n \sqrt{(x_i - y_i)^2} \quad (4)$$

برای ارزیابی شباهت با کمک گرفتن از فاصله اقلیدسی از سه روش زیر استفاده شده است:

۱- Smallest Distance

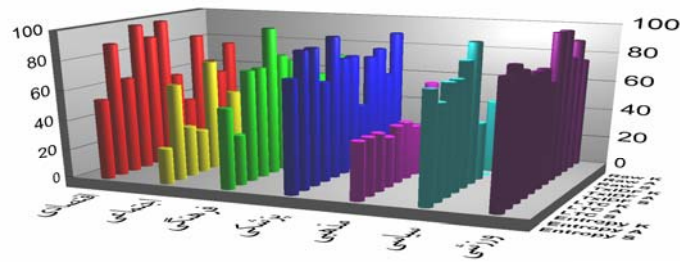
در این روش فاصله متن تست با نزدیک‌ترین متن در هر کلاس، بدست آمده و از بین فاصله‌های بدست‌آمده مینیمم بین آنها انتخاب می‌شود. متن تست به این کلاس تعلق خواهد داشت.

۲- Average Distance

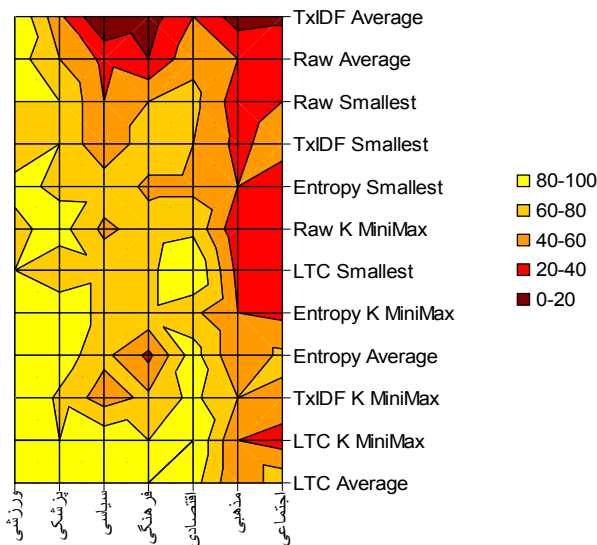
در این روش برای هر دسته، میانگین مقادیر بردارهای ویژگی بدست آمده و بردار میانگین ویژگی کلاس محاسبه می‌شود. متن تست به کلاس با کمترین فاصله با بردار میانگین کلاس تعلق دارد.

۳- K_MinMax

در این روش فاصله متن را با تمام متون موجود محاسبه کرده و مجموعه‌ای با k متن دارای کمترین فاصله با متن تست موردنظر را انتخاب نموده و سپس کلاسی که دارای بیشترین متن در این مجموعه است، به عنوان کلاس آن متن تست انتخاب می‌شود.



شکل ۲- نمودار دقت دسته برحسب الگوریتم KNN



شکل ۳- نمودار کیفیت دقت الگوریتم و دسته

۱۰- نتیجه گیری

در گذشته بیشتر کاری که برای طبقه‌بندی متون انجام گرفته است بر روی متون زبانهای انگلیسی و چینی بوده است. این مقاله روشی برای طبقه‌بندی متون فارسی ارائه داده است. روش کلی بر مبنای روش یادگیری ماشین استوار است که دو فاز یادگیری و تست را در بر گرفت. الگوریتم طبقه‌بند، اطلاعات در مورد طبقه‌ها را پس از انجام بررسی‌های لازم در فاز یادگیری بدست آورد. با توجه به جدید بودن تجزیه و تحلیل متون فارسی، نیاز به کار بر روی الگوریتم‌هایی لازم بود که زبان فارسی را پردازش نمایند تا در نهایت بتوان بردار ویژگی متن را برای الگوریتم‌های طبقه‌بندی معمول فراهم آورد. آنچه در این مقاله مورد تاکید قرار گرفته‌شد، الگوریتم‌هایی بودند که برای زبان فارسی ساخته شده‌اند بطوریکه با حفظ معنی مفید کلمات در عملیات کاهش آنها را به نحو مناسب برای مراحل بعدی طبقه‌بند آماده کنند. پس از ساخته شدن بردار ویژگی، الگوریتمی که بتواند با این داده‌ها، که از متون فارسی بدست آمده‌اند طبقه‌بندی مناسبی را انجام دهد، مطرح شد. در این مقاله از این نمونه الگوریتم‌ها می‌توان به K MinMax اشاره کرد، که با توجه به سربار تقریباً مساوی با دیگر الگوریتم‌ها از بازده بسیار خوبی برخوردار بوده است.

طبقه‌بند نه تنها برای پیدا کردن موضوع متن، که در فیلتر کردن متون با توجه به محتوای نسبی آنها نیز کاربرد دارد. از مواردی که به عنوان ادامه کار می‌توان پیشنهاد داد بهبود روش پردازش زبان طبیعی جهت تولید بردار کلمات

همانگونه که در شکل ۳ مشاهده می‌شود، روش LTC Average برای دسته‌های ورزشی، پزشکی سیاسی، فرهنگی و اقتصادی بهترین جواب را برمی‌گرداند. این ناحیه روی شکل با رنگ زرد (۸۰-۱۰۰) مشخص شده است. بدترین جواب هم توسط رنگ قرمز (۰-۲۰) برای الگوریتم TxIDF برای کلاس‌های سیاسی، فرهنگی، مذهبی و اجتماعی می‌باشد. با توجه به شکل ۲ می‌توان نتیجه‌گیری کرد که روش LTC Average بهترین جواب‌ها را تولید کرده است. جدول کیفیت دقت دسته بر حسب الگوریتم برای روش LTC Average به صورت جدول ۲ محاسبه شده است.

جدول ۲- کیفیت دسته بر حسب الگوریتم LTC Average

ErrorRate	FallOut	Recall	Precision	کلاس
۷.۵۹	۴.۰۳	۷۱.۴۳	۷۵	اجتماعی
۲.۰۷	۱.۵۹	۹۴.۷۴	۹۰	اقتصادی
۲.۰۷	۰.۸۴	۹۲.۳۱	۹۶	پزشکی
۴.۱۴	۲.۴	۸۵	۸۵	سیاسی
۹.۶۶	۰	۵۸.۸۲	۱۰۰	فرهنگی
۸.۹۷	۸.۸۲	۸۸.۸۹	۴۰	مذهبی
۲.۷۶	۳.۱	۱۰۰	۸۰	ورزشی



محمد حسین سرایی مسئول آزمایشگاه پایگاه داده‌ای هوشمند، داده‌کاوی و بیوانفورماتیک در دانشکده برق و کامپیوتر دانشگاه صنعتی اصفهان می‌باشد. حوزه اصلی تحقیقاتی ایشان پایگاه داده‌ای هوشمند، متن‌کاوی، temporal داده‌کاوی، بیوانفورماتیک و تجارت الکترونیک می‌باشد. ایشان دارای مقالات متعددی در کنفرانس‌ها و مجلات معتبر داخلی و خارجی می‌باشند. دکتر سرایی مدرک دکتری خود را از دانشگاه منچستر، انگلستان در علوم کامپیوتر و مدرک کارشناسی‌ارشد خود را از دانشگاه Wyoming آمریکا در رشته مهندسی کامپیوتر و مدرک کارشناسی خود را از دانشگاه شهید بهشتی تهران در رشته علوم کامپیوتر اخذ نموده‌اند. آدرس پست‌الکترونیکی ایشان عبارت است از:

saraee@cc.iut.ac.ir



آذر شاهقلیان کارشناس گروه کامپیوتر و IT دانشکده برق و کامپیوتر دانشگاه صنعتی اصفهان از سال ۱۳۸۲ می‌باشد. همچنین ایشان عضو آزمایشگاه پایگاه داده‌ای هوشمند، داده‌کاوی و بیوانفورماتیک در دانشکده برق و کامپیوتر دانشگاه صنعتی اصفهان می‌باشد. حوزه اصلی تحقیقات ایشان در زمینه داده‌کاوی و شبکه‌های اجتماعی است. آذر شاهقلیان مدرک کارشناسی‌ارشد خود را از دانشگاه آزاد نجف‌آباد در زمینه مهندسی کامپیوتر نرم‌افزار و مدرک کارشناسی خود را از دانشگاه صنعتی اصفهان در رشته مهندسی کامپیوتر اخذ نموده‌اند. آدرس پست‌الکترونیکی ایشان عبارت است از:

shahgholian@gmail.com

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۷/۱۱/۲۶

تاریخ اصلاح: ۹۰/۴/۲۶

تاریخ قبول شدن: ۹۰/۵/۳

نویسنده مرتبط: دکتر محمد حسین سرایی، دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان، اصفهان، ایران.

دقیق تر، بررسی الگوریتم‌های SVM و شبکه عصبی بر روی زبان فارسی و همچنین بررسی روش‌های تقویت بردار ویژگی مناسب برای زبان فارسی اشاره کرد.

مراجع

- [1] T. Joachimes, *Learning to classify Text using support Vector Machines: Methods, Theory, and Algorithms*, Kluwer Academic Publishers, Boston-Dordrecht-London, 2002.
- [2] J. Han, and M. Kamber, *Data Mining Concepts and Techniques*, Elsevier, 2006.
- [3] R. Feldman, and J. Sanger, *The Text Mining Handbook, Advanced Approach in Analyzing Unstructured Data*, Cambridge University Press, 2007.
- [4] D. Chiang, H. Keh, H. Huang, and D. Chyr, "The Chinese text categorization system with association rule and category priority," *Expert System with Applications* vol. 35, no. 1-2, pp. 102-110, 2008.
- [5] T. M. Valdivia, M. G. Vega, and A. U. Lopez, "LVQ for text categorization using a multilingual linguistic resource," *Neurocomputing*, vol. 55, no. 3-4, pp. 665-679. 2003.
- [6] M. T. Mitchell, *Machine Learning*, McGraw-Hill Companies Inc., USA, 1997.
- [7] ح. انوری و ح. احمدی، *دستور زبان فارسی*. چاپ ششم، چاپخانه بهرام، تهران، ۱۳۷۰.
- [8] L. Liu, J. Kang, J. Yu, and Z. Wang, "A Comparative Study on Unsupervised Feature Selection Methods for Text Clustering," *Proc. of the Conference on Natural Language Processing and Knowledge Engineering*, pp. 597- 601, 2005.
- [9] Z. Li, Z. Xiong, Y. Zhang, C. Liu, K. Li, "Fast text categorization using concise semantic analysis," *Pattern Recognition Letters*, vol. 32, no. 3, pp. 441-448, 2011.
- [10] S. Tan, "An effective refinement strategy for KNN text classifier," *Expert Systems with Application*, vol. 30, no. 2, pp. 290-298, 2006.
- [11] R. J. Roiger, and M. W. Geatz, *Data Mining: A Tutorial-based Primer*, Addison-Wesley, 2003.
- [12] S. Alvarez, *An exact analytical relation among recall, precision, and classification accuracy in information retrieval*, Technical Report BCCS-02-01, Computer Science Department, Boston College, 2002.
- [13] Y. Yang, *An evaluation of Statistical Approaches to Text Categorization*, Kluwer Academic Publishers, Netherlands, 2000.

¹ Knowledge

² Text Categorization

³ Machine Learning Approach

⁴ Text Classification

⁵ Natural Language Processing

⁶ Intrinsic

⁷ Keyword

⁸ Stemming

⁹ Threshold

بررسی چند نقشه راه ارائه شده برای مهاجرت به SOA

طیبه آزادی^۱ علی معینی^۲ ناصر مدیری^۱

^۱ دانشکده فنی و مهندسی، دانشگاه آزاد اسلامی واحد تهران شمال، تهران، ایران
^۲ دانشکده فنی، دانشگاه تهران، تهران، ایران

چکیده

متدها و روش‌های مختلف مهاجرت سیستم‌های موروثی به معماری سرویس‌گرا اخیراً مورد توجه خاص قرار گرفته است. در این مقاله چند نقشه راه ارائه شده است. این نقشه راه‌ها شامل نقشه راه IowaSOA، یبکن، IBM و CBDDI هستند. این نقشه راه‌ها شرح داده شده، سپس به بررسی نقاط قوت و ضعف آنها با استفاده از روش SWOT در چارچوب BSC پرداخته می‌شود.

کلمات کلیدی: SOA، نقشه راه، حاکمیت، SWOT، عوامل موفقیت.

۱- مقدمه

یکی از تکنولوژی‌های که در زمینه طراحی و معماری سیستم‌های اطلاعاتی در فناوری اطلاعات امروزه بیشتر در مورد آنها صحبت شده «معماری بر اساس سرویس»^۱ است که به صورت SOA رایج گردیده است. معماری سرویس‌گرا یک فرم از معماری فنی است که مبتنی بر اصول سرویس‌گرایی است وقتی که این معماری از طریق تکنولوژی وب سرویس شکل می‌گیرد، منجر به افزایش کارایی فرایندهای کسب و کار و اتوماسیون حوزه‌های سازمانی می‌گردد [۱]. در حقیقت رشد به سوی SOA و پذیرش آن از سوی سازمان‌های مختلف با مقیاس‌های مختلف طی ۸۰ سال گذشته در حال افزایش می‌باشد البته رهبران IT انتظار دارند بر اساس منافع ارزیابی شده که به پذیرش کنندگان اولیه قول داده‌اند افزایش بیشتری در پذیرش SOA ببینند. امروزه کمتر کسی است که افزایش خدمات SOA را در عبارتهای: کارایی، قابل استفاده مجدد، چابکی و بهره‌وری یک سازمان انکار کند [۲]. در مقاله ابتدا به بیان تعریف SOA و چالش‌های آن پرداخته می‌شود سپس نقشه راههای مطرح شده را بیان و سپس به مقایسه آنها با روش SWOT^۲ در قالب BSC^۳ پرداخته می‌شود.

۲- تعریف و مفهوم SOA

SOA چنانچه در منابع مختلف تعریف شده یک روش بهبود یافته‌ای متمایل به IT و معماری سیستم‌های اطلاعاتی است که روی مجموعه‌ای از سرویس‌ها پایه‌گذاری شده که با یکدیگر مرتبط هستند. بیشترین تعاریف ویژه کسب و کار معماری سرویس‌گرا را به صورت استراتژی تعریف می‌کنند که توابع را از طریق استفاده مجدد سرویس‌های کسب و کار پیاده‌سازی می‌کند [۳]. معماری سرویس‌گرا سبک معماری است که بر تعریف خوب، اتصالات سست، درشت دانه^۴، کسب و کار محور^۵، سرویس قابل استفاده مجدد مشترک تأکید دارد. معماری سرویس‌گرا شامل سیاست‌ها، تجارب و چارچوب‌هایی است که کارکردهای سیستمی را قادر می‌سازد که بصورت مجموعه‌ای از سرویس‌های توزیع شده و مورد استفاده درخواست کننده قرار گیرند. این سرویس‌ها با تعریف یک واسط استاندارد از پیاده‌سازی مجزا شده‌اند. برای معماری سرویس‌گرا تعاریف متنوع و بعضاً مختلفی ارائه شده است: IBM این گونه معماری سرویس‌گرا را تعریف می‌کند؛ معماری سرویس‌گرا شیوه‌ای از معماری برای ایجاد یک معماری فناوری اطلاعات سازمانی است، که اصول سرویس‌گرایی را برای رسیدن به ارتباط محکم بین کسب و کار و سیستم اطلاعاتی که از کسب و کار پشتیبانی می‌کنند، ارائه

تحلیل کنند، متدولوژی را توسعه دهند، برنامه‌ها را استقرار بدهند و پشتیبانی کنند، و ارتباط با شرکاء/ مشتریان/ با تأمین کنندگان را بررسی کنند.

- حرکت به SOA کار بسیار بزرگی است که بهتر است بطور افزایشی انجام شود، نیاز به تغییر در چگونگی معماری و ترکیب برنامه‌های کاربردی مبتنی بر سرویس دارد در حالیکه سرمایه‌های فناوری موجود را به حداکثر می‌رساند.

چالش دیگر در زمینه اتخاذ SOA، چگونگی حرکت به سوی SOA می‌باشد. سوالاتی مانند از کجا باید شروع کرد، مراحل اتخاذ به چه صورت باشد و در هر مرحله به چه اهدافی باید رسید، چه کاری را در چه مرحله‌ای انجام داد، با نقشه راه پاسخ داده خواهد شد. که در قسمت بعد در این خصوص بحث می‌شود.

۴- نقشه راه‌های پیشین

نقشه راه مانند قطب‌نمایی می‌ماند که سبب هدایت مسیر اتخاذ و خارج نشدن از مسیر اصلی می‌باشد. شرکت‌های مطرحی چون IBM، بیکن و... نقشه‌راه‌های ارائه داده‌اند که برخی از این نقشه‌راه‌ها بیان شده و با استفاده از روش SWOT مورد تحلیل قرار می‌گیرند.

۴-۱- نقشه راه IowaSOA

نقشه‌راه IowaSOA در پنج سطح تکامل و از سه منظر سازمان، فناوری و حاکمیت بررسی شده است که مراحل آن بصورت مختصر بیان می‌شود [۹].

الف) سطح اول تکامل

- سازمان

در این سطح، سازمان تصمیم می‌گیرد تا متعهد به اجرای SOA شود. معمولاً با طرح اولیه‌ای شروع می‌شود، که در آن عملکرد برای رسیدن به نیازهای کسب و کار که به طور مستقیم همسو با فعالیت‌های SOA پیاده‌سازی شده است. در این سطح، مفاهیم اولیه معرفی می‌شوند، پیش‌نویس اولیه معماری مرجع SOA تعریف شده، و کاربران درباره استانداردهای در مورد SOA یاد می‌گیرند. فرآیندهای اولیه و سیاست‌ها به منظور حمایت از این سطح تکامل در سازمان تعریف می‌شوند.

با این فرض که همه شعب سازمان باید SOA را درک کنند، بپذیرند، و فعالیت‌های SOA را درون سراسر شعب در نظر بگیرند پروژه سرویس اولیه فرآیند یادگیری ساختار راه‌حل SOA آغاز می‌شود، و تیمی برای این تعامل از شرکت کنندگان از رشته‌های مختلف تشکیل شود.

• فناوری اطلاعات

از منظر فناوری اطلاعات، SOA باید پیش‌نویس شود. همه برنامه‌های کاربردی، سرویس‌ها یا اجزا که می‌خواهند در آینده اجرا شوند باید با معماری مرجع منطبق شوند. معماری مرجع SOA باید به عنوان خصوصیات داخلی ملاحظه شود که هر کسی از آن پیروی کند.

از دیدگاه معماری، جنبه آمادگی مهم SOA؛ تعریف، مشخص کردن و پیاده‌سازی پیش‌نویس مدل اولیه سرویس است.

پیش‌نویس اولیه مدل دامنه مشترک باید طراحی شود. مدل دامنه مشترک متشکل از انتزاع ورودی‌های کسب و کار شعب است.

بر اساس معماری مرجع SOA، ارزیابی و بررسی معماری زیرساخت‌های فعلی فناوری اطلاعات توصیه شده است. سازمان باید بسنجد که آیا زیرساخت‌ها برای حمایت از اجرای SOA آماده هستند.

از نقطه نظر فناوری باید مجموعه‌ای از فناوری‌ها، استانداردهای صنعتی و

می‌دهد. IBM بر یکپارچه‌سازی به عنوان هدف اصلی معماری سرویس‌گرا تمرکز دارد. تعاریف دیگری برای SOA وجود دارد که می‌توانید در منابع ببینید [۴، ۵].

سرویس‌ها هسته SOA، بسته‌بندی دارایی‌های کسب و کار را به خوبی تعریف می‌کنند. آنها مبتنی بر شبکه شرح داده شده با استفاده از زبان رابط مبتنی بر استاندارد، مانند زبان توصیفی وب سرویس (WSDL)، و دسترس‌پذیر از طریق پروتکل‌های مبتنی بر وب از قبیل پروتکل دسترسی شیء ساده (SOAP) هستند. هر مشتری قادر به ایجاد ارتباط با SOAP می‌تواند به سرویس‌ها دسترسی داشته باشد. با این حال، سرویس‌ها معمولاً توسط موتورهای زبان اجرای فرآیند کسب و کار (BPEL)، پورتال‌ها، و یا برنامه‌های کاربردی دسترس‌پذیر می‌باشند [۶].

۳- چالش‌ها در انتقال به SOA

فرآیند مهاجرت به SOA می‌تواند از دو چشم‌انداز مختلف مطالعه شود: یکی چشم‌انداز IT (تکنیکی) و دیگری از چشم‌انداز کسب و کار سازمان. برای اولی باید روی بحث‌های تکنیکی پیاده‌سازی سیستم‌های اطلاعاتی سازمان در چارچوب معماری سرویس‌گرا تمرکز کرد، در حالی که دومین چشم‌انداز عامل‌های داخلی و خارجی کسب و کار را مطالعه می‌کند که ممکن است روی موفقیت یا شکست پروژه اتخاذ SOA در یک کمپانی تأثیر بگذارند

موارد بسیاری در طول چرخه حیات یک پروژه وجود دارند که منجر به شکست آن می‌شوند. نمونه‌هایی از آنها عبارتند از [۷]:

- عدم تعریف شفاف نیازمندی‌ها، که باعث برآورده نشدن انتظارات می‌شود.
 - قطع لبه و یا فناوری جدید که باعث مشکلات پیش بینی نشده می‌شود.
 - طراحی فناوری ضعیف مانع از تغییر راه‌حل یا مقیاس‌پذیری در آینده می‌شود.
 - کنترل ضعیف تغییر اجازه می‌دهد به تغییر درخواست‌های که باعث شکست پروژه می‌شود.
 - تغییر برخی اولویت‌ها باعث انحراف توجه از کار اصلی می‌شود.
- از آنجا که اجرای SOA می‌تواند رویداد تغییر فرهنگ باشد، اصول اساسی تغییر مدیریت باید مانع خروج آن از مسیر اصلی شود. در اینجا خلاصه‌ای از نکات آورده شده است:

۱. آدرس دهی سیستماتیک سمت - بشری
۲. پشتیبانی سطح بالا
۳. دخالت هر لایه
۴. ارتباط واضح مورد کسب و کار
۵. ایجاد مالکیت، نصب عوامل تغییر
۶. ارتباط پیام بطور واضح و مستمر
۷. ارزیابی اثرات فرهنگی
۸. راهنمایی مسائل فرهنگی
۹. آماده‌سازی برای غیر منتظره‌ها
۱۰. تعریف WIIFM^۲ برای هر فرد

سازمان‌ها باید از چند واقعیت مهم SOA، برای برطرف کردن هر تصور یا درک نادرستی در مورد بهترین روش برای انتقال آرام به SOA آگاه باشند، از جمله:

- SOA سبک معماری است که زمان زیادی است وجود دارد. در حالی که راه‌های جدیدی برای تحقق بخشیدن به SOA وجود دارد، از جمله استفاده از فناوری‌ها وب سرویس‌ها [۸].

- موفقیت SOA از گسترش نرم‌افزار بیشتر است. سازمان‌ها باید مدل‌های سرمایه‌گذاری و حاکمیت خود را ارزیابی کنند، تکنیک‌ها را طراحی و تجزیه و

برای تولید باشند. از دیگر خصیصه‌های که باید ذکر شود پردازش مبتنی بر سیاست و مدیریت نسخه سرویس است.

• حاکمیت

سیاست‌های کنترلی از دیدگاه حاکمیت قابلیت‌های بیشتری به نمونه کارهای سرویس‌های بهبود یافته اضافه می‌کنند.

معیارهای پیشرفته‌ای باید طراحی و اجرا شوند. در اولین مرحله از تکامل، معیارها بیشتر روی کد تمرکز داشتند. معیارها در سطح دوم بیشتر سرویس‌گرا می‌باشند. معیارها قادر به ارزیابی انطباق از پروژه SOA، در قسمت (اجزاء، سرویس‌های وب، و غیره) و یا در کل (پروژه SOA)، با معیارهای که می‌تواند حجم کار و نحوه استفاده از سرویس با خودش را اندازه‌گیری کند، هستند.

سیاست‌های که اکنون در سراسر معماری سازمانی گسترده هستند، عبارتند از تست‌های غیرکارکردی مانند کارایی، failover، متعادل کردن بار و امنیت، و سیاست قابلیت استفاده مجدد. حاکمیت زمان اجرا باید قادر بر مدیریت مبتنی بر سیاست با مقید کردن سرویس‌ها (بر مصوبات و مردودها، SLAs^۱، و غیره)، مدیران سیاست، مجریان سیاست، قراردادهای زمان اجرا، برنامه‌های کاربردی باشد.

ج) سطح سوم تکامل

• سازمان

مدیران تمام شعب مرتبط باید از همه پیشرفت‌ها و موفقیت‌های پروژه‌های سرویس‌های در حال پیاده‌سازی آگاه باشند. آنها باید ارزش کسب و کار را درک کنند و حامی قوی طرح‌های SOA در شعب خود باشند. همچنین هر شعبه درگیر باید دفتر مدیریت پروژه SOA ایجاد کند که استانداردها و اصول فرآیندهای مدیریت پروژه در داخل شعب را تعریف و حفظ کند.

• فناوری اطلاعات

شعب مرتبط باید اتخاذ معماری مدل محور (MDA) در پروژه‌های خود را شروع کنند. اتخاذ MDA باید دستورالعمل‌های برای مشخصات ساختار ارائه کند. خروجی باید یک مدل تعریف شده به عنوان مدل مستقل از پلتفرم (PDM) با استفاده از زبان خاص دامنه (DSL) اجرا شود.

هر شعبه همچنین باید ابرمدل SOA که چکیده یک مدل کسب و کار، مدل سرویس‌ها (دارای پلتفرم مستقل)، و مدل پلتفرم وابسته است، را ایجاد کند. ابرمدل SOA باید این چکیده‌ها را اتصال بدهد و آنها را بطور یکپارچه عملیاتی کند. SOA شامل بعضی یا همه معماری‌های اطلاعات شعب است. در صورت امکان، معماری اطلاعات سازمان باید طیف گسترده‌ای از داده‌ها به شیوه‌ای سلسله مراتبی تعریف کند.

از نقطه نظر فناوری، پذیرش ترکیب سرویس‌ها در این سطح شعب‌ها را برای ترکیب فرآیندها، سرویس‌ها و برنامه‌های کاربردی متشکل از سرویس‌های با پیچیدگی کمتر از محیط‌های متنوع بدون ترس از تفاوت این محیط‌ها قادر خواهد ساخت. سرویس ترکیب باید به صورت ترکیب واسط‌های دانه درشت و اجزای پردازش پیام درک شود.

با گنجاندن هم‌اجرای، موتور قوانین برای جدا کردن فرآیندهای کسب و کار از آنچه که فرآیندهای کسب و کار مجاز به انجام هستند با توجه به قوانین شعب ضروری است.

• حاکمیت

مجدداً، سیاست‌های کنترل بررسی می‌شوند، ویژگی‌هایی مانند مدیریت تغییر باید اضافه شوند، ممکن است برخی سیاست‌ها در طول زمان تغییر کنند. هنگامی که قوانین جدید، نیازمندی‌ها یا بهبودهای سیاست موجود وجود می‌آیند، باید با روشی حفظ، ایجاد و ارتباطات چنین تغییراتی را اداره کرد. این تغییرات باید در روش اتصالات سست بدون تاثیر بر پیاده‌سازی سرویس‌ها، چه سیاست‌های وابسته

چارچوب‌ها ارزیابی شوند. در این سطح تکامل، فعالیت‌های فنی R&D مرتبط به SOA مانند SOAP، XML، WS-* و استانداردهای WSDL، وجود دارد.

در صورت مقیاس‌پذیری محیط، فناوری‌های مناسب را باید انتخاب کرد. برای سرویس‌های زیرساخت مشترک اهدافی باید مانند واسطه‌ها (ESB)، زمان طراحی و حاکمیت زمان اجرا مدیریت وب سرویس، و غیره، مشکل و مسائل مدیریت، نظارت و امنیت از طریق گواهی دیجیتال، و غیره در نظر گرفته شود.

• حاکمیت

نقشه راه SOA مهم است زیرا سطوح تکامل را ارزیابی می‌کند، پیشرفت و موقعیت را روی نقشه و وظایف باقی‌مانده را مشخص می‌کند. نقشه راه همچنین محدوده کلی طرح و تاریخ‌های نقاط عطف مهم را نشان می‌دهد.

سیاست‌های کنترل اولیه باید در جای خود باشند، یعنی استراتژی ارتباطات در سازمان، استانداردها و رویه‌های مدیریت پروژه، مدیریت ریسک و چگونگی بحران و درگیری باید درون یک تیم یا پروژه پشتیبانی شوند.

پیش‌نویس اولیه معیارهای نرم‌افزار قبل از شروع پروژه لازم است. سنجش و موفقیت مدیریت، بهره‌وری کد مهم هستند.

حاکمیت دارای منظر فنی و فناوری است که باید برای مدیریت یا پیاده‌سازی همسو باشد. سیاست‌های اساسی باید برای اصول راهنمایی، چرخه حیات سرویس، تست عملکرد، مدیریت سرویس، و نظارت تعریف شوند.

در نظر داشته باشید که پیش‌نویس اولیه حاکمیت زمان طراحی؛ برای پیاده‌سازی ویژگی‌های کشف، اجرای سیاست‌ها، فرآیندهای انتشار سرویس، جستجو سرویس مبتنی بر ویژگی، ابزار و IDE با ویژگی‌های انتشار و ادغام UDDI، و غیره نوشته شده است.

ب) سطح دوم تکامل

• سازمان

در این سطح، ادارات، برنامه‌ها یا واحدهای کسب و کار باید همسو و متعهد با مدیریت اجرایی مرتبط با SOA شوند. تیم چند رشته‌ای ایجاد شده در سطح اول تکامل برای پشتیبانی پروژه‌های بیشتر SOA باید گسترش یافته، و یا جدید ایجاد شود. در این سطح، پروژه سرویس اولیه باید کامل شود. مدیریت باید استراتژی‌های ارتباطات در مورد نحوه تبادل پیام SOA در سراسر سازمان تاسیس نماید.

• فناوری اطلاعات

پیش‌نویس اولیه معماری مرجع SOA باید در این مرحله بیشتر تکامل یابد و رسمی شود. با تجربه سرویس پروژه اولیه، معماری مرجع از طریق این تجربیات بروز شود. معماری مرجع اکنون باید فرآیندهای کسب و کار را از واحدهای کسب و کار و یا ادارات شرکت کننده در طرح‌های SOA مرتبط به پروژه‌های سرویس اضافی در حال اجرا هستند، ترکیب کند. یکی دیگر از جنبه‌های کلیدی گنجانیدن معماری رویداد محور (EDA) در معماری مرجع است که تکامل خوبی به SOA برای قابلیت تعامل ناهمگام خود و انعطاف‌پذیری آن در زمان قطع اتصالات و خرابی تحویل پیام می‌دهد.

پیش‌نویس مدل سرویس اولیه باید در این زمان بروز شده و باید شامل مدل سرویس واحد کسب و کار و نسخه بروز شده مدل دامنه مشترک مبتنی بر موجودیت‌ها و تحولات مشترک جدید که برای پروژه‌های سرویس اضافی انتخاب شده‌اند، باشد.

پایه‌های، زیرساخت مشترک باید گسترش یابند و در این مرحله پیاده‌سازی شود؛ سرویس‌های مثل واسطه‌ها باید نصب شوند، استقرار یابند و همچنین باید حداقل یک مجموعه‌ای از سرویس‌های در حال اجرا، امنیت از طریق گواهی دیجیتال، رجیستری سرویس‌ها وجود داشته باشند. سرویس‌های گسترده مانند دسترس‌پذیری بالا و مدیریت حجم کار نیز باید تنظیم شده، تست شده و آماده

شده برای سطح چهار آماده برای حمایت از خود کشف کننده و خود ترمیم کننده است، حاکمیت باید در تمام سیاست‌ها، استانداردها و فرآیندهای که فناوری اطلاعات را پشتیبانی می‌کند را داشته باشد.

ذ) سطح پنجم تکامل

• سازمان

شعب سازمان بلادرنگ شده‌اند که در آن کسب و کار می‌تواند مانیتور شود، مدیریت شود، وضعیت حوزه‌های کلیدی سازمان نمایش داده شود، و مبتنی بر رویدادها بلادرنگ، سازمان فعالانه واکنش نشان دهد. مدیران و مجریان قادر به فهرست کردن وضعیت فعلی سازمان هستند، آنها می‌توانند از طریق داشبورد به زیر وارد شوند که عملکرد واحدهای کسب و کار را نشان بدهد و از طریق ویدادهای بلادرنگ سازمان قادر به واکنش از طریق اقدامات از پیش تعریف شده بطور خودکار یا دستی باشد.

• فناوری اطلاعات

معماری مرجع IowaSOA ساختارهای وابسته را با اجازه دادن به همه سرویس‌ها از همه شعب برای استفاده از طریق یک نقطه ورودی منفرد، مثل پورتال و یا داشبورد شبیه سیستم ترکیب می‌کند. از این پورتال، مصرف کنندگان درخواست سرویس‌ها را خواهند داد و SOA مراقب، معین کردن چه سرویس‌ها یا تعاملی از سرویس‌ها باید در راستای تحقق درخواست مصرف کننده فراخوانی بشوند، است. این سرویس‌های هوشمند فراتر از شعبه، شعب و یا مرزهای کشور خواهند رفت. سرویس‌ها در یک محیط وابسته، حتی در سراسر کشور باز می‌باشند و شبکه می‌شوند. از نقطه نظر فناوری، فدراسیون باید از طریق سرورهای فدراسیون و روابط صحیح پیکربندی شود

• حاکمیت

استانداردهای سیاسی، پیکربندی، مدیریت، اجراء و عملیات روزانه خود نیاز به بروز رسانی برای تهیه ویژگی‌های وابسته دارد و باید برای استانداردهای فدراسیونی حاکمیت ملی توافق کنند که قادر به قابلیت تعامل و هماهنگی بین کشورهای مختلف و حاکمیت فدرال خواهند بود. پیاده‌سازی رجیستری فدراسیونی یک ضرورت به منظور ذخیره، نسخه، نگهداری، مدیریت و اداره محتوی مرتبط به اطلاعات وابسته، سیاست‌ها، و شناسایی در یک روش امن است. فناوری‌های مثل استانداردهای رجیستری ebXML و زبان نشانه‌گذاری کنترل دسترسی پیشرفته OASIS قابل ملاحظه هستند.

۴-۲- نقشه راه روش بیکن^۹

اتخاذ سراسری SOA می‌تواند فرآیند طولانی باشد، اما فرآیند اتخاذ می‌تواند برای سازمان از همان مرحله اولیه نتیجه بخش باشد که برای ادامه تضمین اتخاذ SOA و درک ارزش SOA موفقیت‌های ارائه شده و اولیه حیاتی هستند. "وسیع فکر کن، جزئی شروع کن، سریع توسعه بده" ایده مناسبی است که زمان طرح‌ریزی تلاش‌های گسترده باید در ذهن‌تان باشد.

در این روش توصیه شده است که فرآیند اتخاذ SOA به خاطر مدت طولانی اجراء آن با یک طرح تحویل فزیندی شده استنتاج شود. برای سازمان بزرگ با چندین محصول، چندین واحد کسب و کار متعدد و حضور در بازار جهانی، نقشه راه باید سه فاز مختلف پوشش سریع ۱۸-۲۴ ماه را بررسی کند، همانطور که در شکل یک نشان داده شده است [۱۰].

برخی آمادگی‌ها و برنامه‌ریزی‌های قبل از توسعه پلتفرم SOA جدید، بایستی انجام و اجراء شود. این زمان شامل فاز پیچیدگی سنتی که سهامداران اولیه را درگیر می‌کند، می‌باشد. در ابتدا اهداف اصلی ایجاد یک تیم اولیه، ایجاد دیدگاهی

به امنیت و چه وابسته به فرآیند، انجام شود. یکی دیگر از جنبه‌های کلیدی حاکمیت سازمانی اضافه کردن قابلیت‌های عملکرد نظارت بر مدیریت سیاست و پیروی از استانداردهای حاکمیت ملی است.

معیارهای استانداردها و مقررات با معیارهای فرآیند کسب و کار که درک درستی از سرویس یا سیاست، در ارتباط با فرآیندهای کسب و کار شعب باید بروز شوند.

از نقطه نظر فنی نیز سیاست‌ها برای حمایت از مدل‌سازی فرآیند کسب و کار بروز خواهند شد، آنها همچنین ساختار سلسله مراتبی اطلاعات در سراسر شعب و چگونه معماری آن و مونتاژ آن را برای نشان دادن چیزهای معنی‌دار در سیاست‌هایی که پیاده‌سازی شده‌اند، منظور خواهند کرد.

مخازن ابر داده قابل جستجو برای فعال کردن یک راه حل کامل SOA مهم هستند. اطلاعات ابر داده فراتر از مفهوم UDDI، نیاز به جستجو و پرس و جو، نه تنها در وب سرویس‌ها، بلکه محتوا، الگوهای XML، و خود ابر داده اطلاعاتی دارند، ابر داده اطلاعاتی یک نقش اصلی فعال‌سازی SOA واقعی است.

در این مرحله، حاکمیت باید فراتر از شعبه برود و باید با شعب دیگر و یا سازمانها در محدوده یا خارج از محدوده و حاکمیت فدرال تماس و عملیاتی داخلی داشته باشد.

د) سطح چهارم تکامل

• سازمان

در این زمان شعبه به طور کامل سرویس‌گرا شده است، همه کمیته‌ها، پروژه‌ها فعالیت‌های خود را از طریق سیاست‌ها و استانداردهای تعریف شده SOA اداره می‌کنند. همه فرآیندهای کسب و کار، برنامه‌ها، داده‌ها به عنوان سرویس‌ها پیاده‌سازی شده‌اند که کاملاً قابل استفاده مجدد برای دستیابی به چابکی کسب و کار هستند. برنامه‌های کاربردی پیچیده مرکب از سرویس‌های هستند که اجازه دارند تا در اکوسیستم SOA شرکت کنند.

• فناوری اطلاعات

معماری اطلاعات طراحی و اجراء شده در سطح سه یکپارچگی معنایی را ارتقاء می‌دهد که در آن محل داده‌ها و ساختار کنونی و سلسله مراتب آن مهم است، اما همچنین آن داده زمینه و وابستگی معنایی روی سناریوی ارتباطی می‌دهد. کشف قابلیت‌های دینامیکی، مانند فهرست سرویس، باید به معماری پشته اضافه شود. سرویس‌ها در فهرست سرویس‌های ثبت می‌شوند که می‌تواند طبقه‌بندی شوند. با اضافه کردن فهرست سرویس‌ها در بین تامین کنندگان و مصرف کنندگان، سرویس‌ها مقیاس‌پذیرتر می‌شوند، سرویس‌ها سریع بروز رسانی می‌شوند؛ مصرف کنندگان قادر به انتخاب تامین کنندگان در زمان اجراء هستند و انفصال کلی بین مصرف کنندگان و ارائه دهندگان بدست می‌آید.

در این برهه، قابلیت‌های خود ترمیم کننده در معماری SOA بیشتر تکامل می‌یابد.

از دیدگاه فنی، سرویس‌ها و مدیران سرویس باید قادر به نظارت مطابق با SLA و معیارهای تعریف شده در زبانی مانند کسب و کار باشند.

• حاکمیت

در هر سطح تکامل، سیاست‌ها بهبود یافته، قابلیت‌ها و ویژگی‌های بیشتر اضافه می‌شوند. در سطح چهار تکامل، سیاست‌ها نیز باید در مرحله بهبود مستمر باشند و استانداردهای ملی برای واژگان معنایی را رعایت کنند. حاکمیت در این نقطه کامل است و تنها باید حفظ بهبود فرآیندهای جاری خود را تکرار کند. معیارها اکنون قادر خواهند بود کسب و کار را اندازه گیری و ارزیابی کنند.

حاکمیت باید بتواند همه جنبه‌های تعریف شده در چشم‌انداز فناوری اطلاعات این سطح از تکامل را در نظر بگیرد. از آنجا که معماری و فن‌آوری تعریف

می‌کشد. فاز یک ساختن زیر ساخت هسته SOA را درون سازمان شروع می‌کند. فاز دو روی ارائه برنامه‌های کاربردی اضافه شده و سرویس‌ها برای چندین واحد کسب و کار و توابع سراسری سازمان ساخته می‌شود. طرح اتخاذ بایستی تعریف کند چگونه سازمان به استفاده مجدد سرویس‌ها گسترده‌تر در سراسر واحدهای کسب و کار دست می‌یابد.

در طی این فاز است که شما، دیدگاه‌های ارزشمند و واقعی نسبت به استفاده مجدد، کشف، انتشار و کنترل سرویس‌ها بدست می‌آورید. تلاش مضاعف در سراسر سازمان می‌تواند بیشتر برای آمادگی تعریف کردن و بازسازی کردن سرویس‌های مشترک باشد. در مجموع طرح اتخاذ بایستی سرویس‌های مشترکی را برای امنیت، شناسایی، ورود و کیفیت پیاده‌سازی سرویس‌ها تعریف کند.

فاز سه اهداف - بلند مدت استراتژی SOA و ادامه موفقیتش درون سازمان را نشان می‌دهد. بیشترین تمرکز روی پویایی کشف سرویس‌ها، شامل کشف سرویس‌های خارجی در سازمان‌های مشارکتی است. در یک نقطه، بیشترین برنامه‌های کاربردی گسترده سازمانی و روی پلتفرم SOA گسترده می‌شوند.

در فاز سه بایستی مشارکت‌های گسترده سازمان در تعریف زیرساخت‌های SOA، و سرویس‌های مشترک گسترده میان واحدهای کسب و کار، شرکاء و مشتریان را مشاهده می‌کنید. بطوری که در نسخه‌های قبلی بیکن نمایان شد، SOA همچنین منافع بالقوه‌ای کاهش هزینه نگهداری و توسعه برنامه‌های کاربردی را در خود دارد. این منافع در طی فاز سه درک خواهند شد.

۴-۳- روش IBM

استراتژی SOA نباید با روش یکباره^{۱۰} در محیط فناوری اطلاعات موجود جایگزین شود، بلکه باید نقشه راه تکاملی و افزایشی باشد. اغلب جایگزینی کلی غیر ممکن است هنگامی که اکثریت مردم در سازمان فناوری اطلاعات مشغول نگهداری سیستم‌های در حال اجرا می‌باشند. بنابراین نقشه راه باید یک روند تکرار شونده را منعکس کند [۱۲].

سازمان دارای گزینه‌های متعددی برای ورود به معماری سرویس‌گرا است. این گزینه‌ها مشخص می‌کنند که چقدر مدل SOA به کسب و کار نفوذ می‌کند و سطوح اتخاذ را تعریف می‌کند. گزینه‌ها به شرح زیر هستند:

- اتخاذ اولیه: سازمان‌های که می‌خواهند خطرات را کاهش بدهند، در ابتدا باید اعتبار و آمادگی ارزیابی فن آوری را انجام بدهند که تاثیر کسب و کار و تکنیک را در دامنه تعریف شده تجزیه و تحلیل کنند. در نهایت، ارزش کسب و کار و تکنیک بکار برده شده در این محدوده می‌تواند با پیامدهای واقعی سازمان قیاس شود، این معمولاً به معنای تعهد عمیق‌تر برای حرکت به SOA تفسیر می‌شود. این آزمایش‌های پابلوت اولیه شامل ایجاد و ارائه سرویس‌های از عملیات کسب و کار موجود در برنامه‌های کاربردی جدید یا موجود می‌باشد. این آزمایش‌ها برای اعتبارسنجی اولیه چندین نقطه تصمیم‌گیری از قبیل موارد زیر استفاده می‌شود:

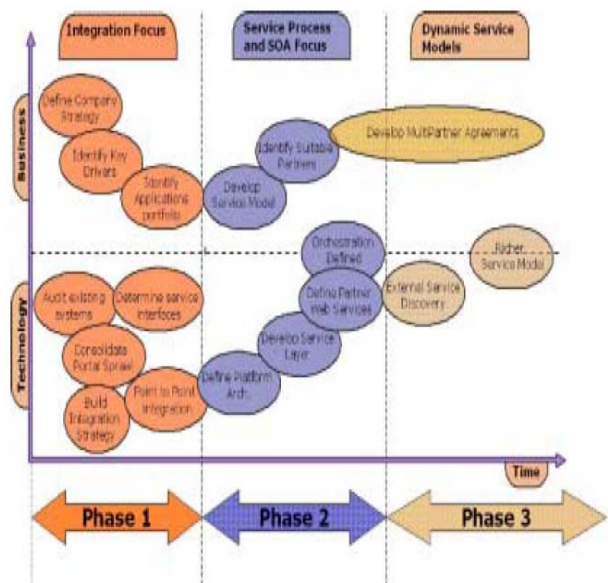
- توانایی برای انتقال سیستم‌های موروثی موجود. این ممکن شامل راه‌حل‌های تکنیکی مانند پیام، آداپتورها و اتصال دهنده‌ها باشد، یا شامل هدایت همکاری با فروشندگان باشد که بتواند محصولاتی برای یکپارچگی سرویس‌گرا ارائه دهند.

- قابلیت‌های نیازمندی‌های غیر کارکردی مثل کارایی، امنیت، قابلیت مدیریت و دسترس پذیری شکل‌دهی.

- ساختار سازمانی نیاز به حمایت سیر تکامل سازمانی دارد، بخصوص آنهایی که شکاف‌های مهارتی و ساختار حاکمیت مؤسسات را نشان می‌دهند.

- اتخاذ مسیر کسب و کار: در این سطح، سازمان مسیر کسب و کار را

برای هر سازمان مورد نظر و چگونگی کمک به SOA و برپایی فونداسیون معماری است. مرکز تعالی نقش مهمی را در روزهای اول بازی می‌کند. فعالیت‌های این فاز باید امکان‌پذیر بودن تلاش را تعیین کنند و سازمان را برای پذیرش معماری و پیاده‌سازی که مد نظر دارند، آماده کند. به طور کلی فعالیت‌ها شامل اجرای جزئیات رسیدگی سیستم‌های موروثی و برنامه‌های کاربردی سراسر سازمان، شامل آنهایی که لازم است مهاجرت کنند و آنهایی که می‌خواهند یکپارچه شوند، است. اولین استراتژی سرویس‌ها و برنامه‌های کاربردی باید تعریف شوند و فرآیندهای کسب و کار بایستی مشخص شوند. همچنین زمانی برای توسعه مفاهیم ضعیف پیاده‌سازی است که نواحی ریسک در سرویس‌های پیشنهادی را نشان می‌دهند، بخصوص اگر این اولین بار است که سازمان فناوری مرتبط به SOA را استفاده می‌کند.

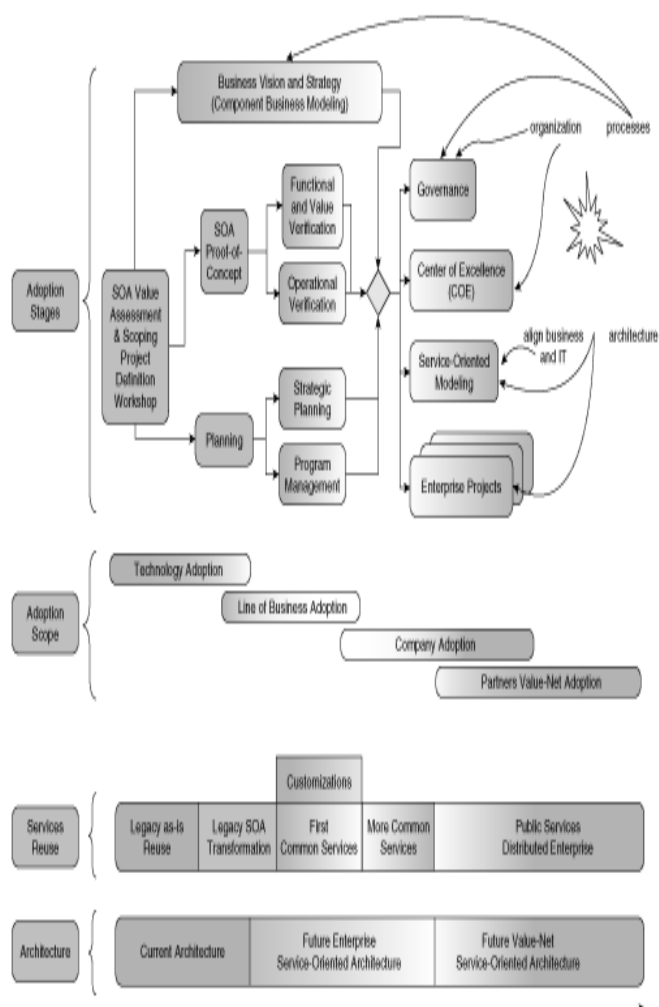


شکل ۱- نقشه راه اتخاذ SOA - روش بیکن (مرجع [۱۱])

شکل ۱ نقشه راهی سطح بالا را برای اتخاذ SOA نشان می‌دهد. محور عمودی فاصله بین ملاحظات کسب و کار و ملاحظات فناوری را نشان می‌دهد. ملاحظات کسب و کار روی نصف بالای نمودار سیر تکاملی از استراتژی تعریف شده از طریق توسعه مدل سرویس‌ها، برای توسعه چندین توافقات مشارکتی را نشان می‌دهد. وابستگی‌های فناوری روی نیمه پایین نمودار بحث‌های را مثل توسعه واسط‌های سرویس از طریق سیر تکاملی معماری و هماهنگ کننده سرویس، کشف سرویس و توسعه مدل‌های سرویس غنی‌تر را نشان می‌دهد [۱۱].

فاز یک فرآیند اتخاذ روی یکپارچگی تمرکز می‌کند. در طی این زمان است که اولین قابل تحویل‌های واقعی از فرآیند اتخاذ SOA ایجاد می‌شود. این فاز معمولاً بیش از شش ماه نیست و باید شامل تلاش گسترده تقریباً بین ۱۶ تا ۲۰ هفته‌ای باشد. هدف از این فاز، معرفی SOA به سازمان، بدست آوردن برخی سرویس‌های قابل استفاده مجدد در واحد کسب و کار و استفاده SOA برای اولین بار برای تحویل راه‌حلی برای مشکلات برخی استراتژی‌ها است. طرح اتخاذ بایستی شامل استراتژی برای دسترسی به داده‌ها تشکیلات در مقابل سرویس‌ها و بدست آوردن یکپارچگی سریع برای برنامه‌های کاربردی موروثی با استفاده از وب سرویس‌ها باشد. خود پروژه بایستی نسبتاً پیچیدگی کمتری داشته باشد بدون اینکه بی‌محتوا باشد.

فاز دوم روی موفقیت فاز اول ساخته می‌شود و بیشترین تمرکز را روی فرآیندهای سرویس‌ها، چارچوب کار و قابلیت جایابی دارد. سازمان حالا اطمینان دارد که کارآمدگی و موفقیت در راه است. هشت تا ۱۲ ماه این فاز معمولاً طول



شکل ۲- نقشه راه IBM (مرجع در [۱۳])

۴-۴-۱- معرفی فازهای نقشه راه

اتخاذ SOA وظیفه مدیریت تغییر پیچیده نیاز به تغییر در فشار ممارست دامنه گسترده‌ای از اندازه تحت نظم کسب و کار و فناوری اطلاعات دارد. توانایی‌های سازمان‌های بزرگ برای ایجاد تغییرات عمیق با پیچیدگی برنامه‌های کاربردی موجود و نمونه‌های فناوری محدود شده است و نیاز مستمر به پشتیبانی کافی برای تغییرات فرآیند کسب و کار و نیازمندی‌های اطلاعات دارد. برای بعضی سازمان‌ها، بنابراین بدست آوردن تکامل SOA بدون از طریق برنامه‌ریزی که اهداف توأم پشتیبانی کسب و کار متوالی و تکامل SOA را نشان بدهد، و اطمینان بدهد که توازن معقولی بین آنهاست، ممکن نیست. در نتیجه تحول یک سازمان برای SOA برنامه چند ساله خواهد بود [۸]. یک تکنیک اصلی در مدیریت تحول SOA فازبندی فعالیت‌ها است. در هر فاز هدف مشخصی برای بهبود فرآیند و ارزیابی توسعه تنظیم می‌شوند که پیشرفت بهبود ممارستی را ممکن می‌سازد.

فازها قادرند برای ما:

- مدیریت یا کنترل ریسک
- مدیریت سرمایه
- کنترل انتظارات
- همسو کردن ظرفیت وابستگی‌ها
- هماهنگ کردن و مدیریت کار گروه‌های متمایز (که بنوعی اتصال سست با تعریف وابستگی‌های گروه سراسری در محدوده‌های فاز خواهد بود).

شناسایی خواهد کرد و فرآیندهای چابک و انعطاف‌پذیر اولویت‌بندی می‌شوند، SOA تلاش می‌کند ارزش کسب و کار را افزایش دهد. البته، ممکن است سازمان در حال حاضر این اولویت‌ها را شناسایی کند و یا مسئله حیاتی کسب و کار را برای حل داشته باشد. در این موارد، شما هنوز هم نیاز به بررسی قابلیت‌های کاربرد SOA برای حل مسئله مهم دارید. این شامل یک فاز ارزیابی گسترده اولیه و شناسایی معیارهای کلیدی و عوامل موفقیت بحرانی است.

• اتخاذ سازمانی: این سطح از اتخاذ شامل ساخت نمای کسب و کار یک سازمان سرویس‌گرا است، با اولویت‌بندی کامل پروژه‌ها بر اساس ارزش کسب و کار، فازهای معماری و پیاده‌سازی دنبال می‌شود. شما نیاز به دسته‌بندی فعالیت‌های سازمانی به حوزه‌های کسب و کار جداگانه و اجزای تشکیل دهنده سازمان دارید. این دسته‌بندی ممکن است در حال حاضر درون سازمان یا به صورت مدل صنعتی موجود باشد. در این مرحله، شما باید انجمن حاکمیت SOA با توانمندی لازم برای نظارت، تعریف، و اجازه تغییر به سرویس‌های درون سازمان را تاسیس نماید.

• اتخاذ سازمان-و-شریک-شبکه: در این سطح، تحول گسترده‌ای از مدل‌های کسب و کار موجود یا استقرار مدل‌های کسب و کار جدید وجود دارد که نه تنها شامل سازمان، بلکه شرکای تجاری خود، تامین‌کنندگان، یا مشتریان می‌باشد. سپس سازمان می‌تواند نقش‌های مناسب برای ارائه ارزش خود را انتخاب کند، که یک ارائه‌کننده، مصرف‌کننده، کارگزار، خبرخوان، مبادله‌کننده سرویس، یا هر ترکیبی از این نقش‌ها باشد.

برای هر یک از اجزاء و سرویس‌ها کسب و کار اولویت‌بندی شده، نقشه راه فازهای مختلفی از توسعه پروژه فناوری اطلاعات، با فازهای شروع، جزئیات، اجرا، و تست و تولید دنبال کند، مانند نمونه مشخص شده در فرآیند یکپارچه رشنال (RUP). با این حال، هر یک از این فازها شامل فعالیت‌های جدید مربوط به شناسایی اجزاء سرویس‌ها و درک آنهاست. شکل ۲ دیدگاه کلی از نقشه راه را نشان می‌دهد، مراحل اتخاذ و فعالیت‌های مربوطه را ببینید [۱۳].

این نمودار جامع نیست اما نشان‌های از گام‌های بالقوه دارد که شما می‌توانید دنبال کنید.

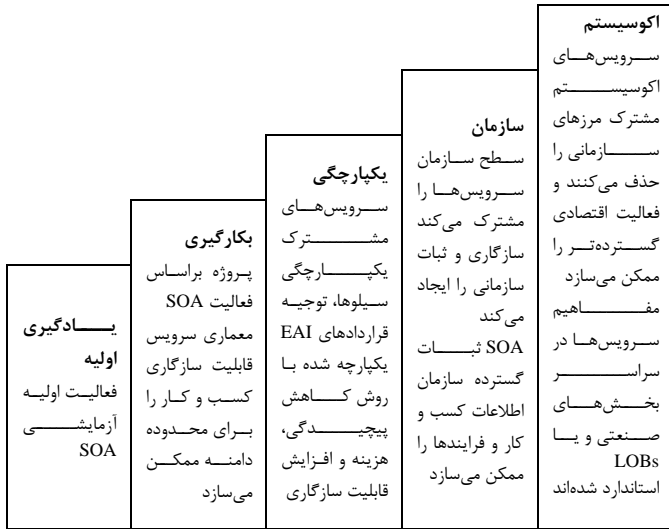
روش IBM بصورت فلوچارتی از نقشه راه حرکت به سمت SOA موجود بود که براساس این فلوچارت راجع به این نقشه راه صحبت می‌شود. مستند زیادی برای نقشه راه IBM یافت نشد بنابراین براساس شکل آن، نقشه راه را تفسیر می‌شود. این روش بر خلاف روش‌های پیشین به جزئیات مهمتری توجه کرده است. استراتژی کسب و کار را در طول مسیر قرار دارد، بعد از ارزیابی به اثبات مفاهیم و برنامه‌ریزی پرداخته است. پس از آن اثبات مفاهیم عملکرد و ارزش بازبینی و بازبینی عملیاتی را انجام می‌دهد و در مرحله بعد از برنامه‌ریزی به برنامه‌ریزی استراتژیک و مدیریت برنامه

توجه کرده و پس از پایان این مراحل به حاکمیت، مرکز تعالی و مدل‌سازی سرویس‌گرا و پروژه‌های سازمانی پرداخته است. همچنین سیر تکاملی سازمان را نشان می‌دهد که در سه بعد دامنه پذیرش، استفاده مجدد سرویس‌ها، معماری می‌باشد.

۴-۴-۲- نقشه راه ارائه شده CBDI

در ابتدا فازهای این نقشه راه و سپس مدل نقشه راه و مدل جریان نقشه راه را بیان می‌کنیم [۱۴].

۴-۲- مدل فاز اولیه



شکل ۳- فازهای اتخاذ SOA - الگوی اولیه (مرجع در [۱۴])

۴-۴-۳- فازهای نقشه راه افزایشی

مدل اولیه فازی به طور گسترده‌ای قابل اجرا است. با این حال در شرایطی ساختار فازی نیاز به تغییر دارد. ما می توانیم این‌ها را به عنوان الگوهای مشترک تشخیص بدهیم. جدول ۲ ساختارهای فاز جایگزین را نشان می‌دهد. در هر یک از این الگوها فرصت زیادی برای سفارشی سازی، به خصوص در فازهای rollout اصلی - سازمان، اکوسیستم، یکپارچگی و برنامه وجود دارد. توجه داشته باشید که برخی از این الگوها گسترش تدریجی دامنه را شامل نمی‌شود، و براساس تجربه که فعالیت SOA را داخل اکوسیستم قرار بدهد، نمی‌باشد.

جدول ۲- ساختارهای فاز جایگزین (مرجع در [۱۴])

الگو	نیازمندی‌ها	ساختار فاز کاندید
اتخاذ سازگاری (الگوی اولیه)	ایجاد قابلیت‌های فنی و شایستگی، اثبات توجه، درگیری به تدریج کسب و کار در برنامه ریزی و حاکمیت	<ul style="list-style-type: none"> یادگیری اولیه بکارگیری یکپارچگی سازمان اکوسیستم
اتخاذ حاکمیت	ایجاد سطح مناسبی از استاندارد و به اشتراک گذاری قابلیت‌های تکنیکی سراسر شعب‌های عمومی. توجه کنید شعب مجزا یا شعبه باید رویکرد استاندارد سازمان را استفاده کنید	<ul style="list-style-type: none"> جامعه مورد علاقه استانداردهای متداول پیوستن به حاکمیت
M&A	ادغام SOA در M&A برای هر دو تسهیل یکپارچه‌سازی کسب و کار و برپایی موثر مدل کسب و کار جدید مبتنی بر اصول سرویس‌ها را طرح ریزی می‌کند	<ul style="list-style-type: none"> مقتضی پشتکار ۱۰۰ روز اول یکپارچگی
برنامه اصلی	قابلیت‌های SOA مورد نیاز به خصوص برای برنامه اصلی	<ul style="list-style-type: none"> چارچوب مرجع مسیر باریک برنامه

مدل فاز اولیه برای راهنمایی برخی سازمان‌ها بزرگ در ساخت نقشه راه SOA توسعه داده شده است و در استفاده شرکت و بخش‌های اداری زیادی است. تمرکز فازهای نقشه راه هماهنگ کردن مجموعه گسترده‌ای از قابلیت‌های مورد نیاز برای ارائه و استفاده سرویس‌های است که خروجی‌های خاصی می‌دهند. نیاز به تغییر گسترده برای اتخاذ SOA به طور کامل دارید، که بسیار برای تعمد مرحله و تحمیل خروجی‌های طبیعی توصیه شده است، به طوری که یک سازمان یا تشکیلات در یک روش کنترل شده برای صلاحیت کامل و گنجایشی با مدیریت ریسک نمو می‌کند. مدل فاز اولیه به الگوی نگاشت می‌شود که می‌تواند در بیشتر سازمانها مشاهده شود. در مراحل اولیه SOA نیازمندی‌های اولیه، برای برپایی قابلیت و صلاحیت تکنیکی است. تا زمانی که صلاحیت‌های اولیه تنظیم و اثبات نشده‌اند هر تغییری در پروژه سازمان نامناسب و نادرست است. برای این منظور قابلیت‌های کلیدی باید در این مکان قرار بگیرند برای اطمینان از اینکه تلاش‌های اولیه بطور موفق درون برخی محدودیت‌ها تعریف شوند. هنگامی که درست و صحیح شروع کنید، روش استراتژیک با درگیری بیشتر کسب و کار، امکان‌پذیرتر است.

جدول ۱- مدل فاز اولیه (مرجع در [۱۴])

فاز	خروجی‌های اولیه	استراتژی‌های کلیدی
یادگیری اولیه	مهارت‌های اولیه در گروه اصلی اثبات مفاهیم سرویس‌های اولیه و فقط اتوماسیون واحدها پیاده‌سازی مرجع	آزمایش برپایی مرکز تعالی معیار سازی و یادگیری از منابع زیاد اثبات مفاهیم پروژه‌ها ارزیابی فروشندگان، اثبات و آزمایش
بکارگیری	پروژه‌های ساختار یافته بهتر صلاحیت SOA تکنیکی برای دامنه باریک	پروژه متمرکز شده بر SOA پروژه ESB یادگیری پروژه‌های هادی و تجربه مشترک کاهش معماری مرجع سطح سازمان
یکپارچگی	کاهش تلاش یکپارچگی استفاده مجدد سرویس‌های کلیدی بدون رشد بیشتر در یکپارچگی نقطه به نقطه گسترش صلاحیت تکنیکی SOA برای پشتیبانی استفاده گسترده‌تر از الگوها	تمرکز کنترل معماری و حاکمیت کلی یکپارچگی فعالیت سرویس‌ها با فرصت مدل داده منجر به معماری سرویس گسترده‌ی فناوری اطلاعات منجر به فعالیت تکامل معماری مرجع براساس تجربیات
سازمان	به اشتراک‌گذاری سرویس‌ها سازگاری اطلاعات کسب و کار و فرآیندها	مشارکت کسب و کار الحاق کسب و کار و برنامه‌های فناوری اطلاعات مدل کسب و کار منجر به معماری سرویس پوشش سیستم های موجود و ارائه نمونه کارهای سرویس‌های کسب و کار اصلی برای استفاده سازمان
اکوسیستم	به اشتراک گذاری فرآیندهای کسب و کار سازگاری اطلاعات مشترک	عمودی) سرویس‌های استاندارد صنعتی پروتکل‌های زیرساخت مشترک، درگاه‌ها، مدیریت محیط هماهنگی معماری

جدول ۳- مدل جریان اولیه (مرجع در [۱۴])

محدوده قابلیت	تولید
ابزار مدیریت شامل چشم انداز، استراتژی، بودجه، شارژ، اندازه گیری و نظارت و برنامه‌های اتخاذ SOA و مدیریت	مدیریت SOA
مدل های کسب و کار و طرح‌های تحول کسب و کار که کسب و کار سرویس‌گرا و یکپارچه سازی و محرک الزامات برای معماری سرویس را شکل می‌دهد	طراحی کسب و کار
معماری سرویس و طرح نمونه کار سرویس (SPP)	معماری
معماری زیرساخت‌های عملیاتی و پیاده‌سازی برای پشتیبانی وضعیت‌های زمان اجرای چرخه حیات سرویس می‌باشد. شامل رجیستری زمان اجرا، نظارت، وساطت، امنیت، هماهنگی، سیاست و مدیریت است.	زیرساخت عملیاتی
چرخه حیات معماری پشتیبانی و پیاده‌سازی برای پشتیبانی برنامه‌ریزی برای وضعیت‌های تحویل سرویس می‌باشد. شامل رجیستری، مخزن و مدیریت سرمایه، ابزارهای مدل سازی، ابزارهای توسعه و مونتاژ می‌باشد.	زیرساخت تولید
چارچوب مرجع شامل مدل مفهوم، معماری مرجع و فرآیند برای فعال و هماهنگی تحویل و اجرای سرویس‌های فدرالی	چارچوب مرجع و فرآیند
تعریف نقش و مسئولیت‌های مورد نیاز برای ایجاد، اجرا، مدیریت و اداره محیط سرویس‌ها	سازمان
قابلیت اجرای کلاس‌های پروژه و یا برنامه، و به عنوان الگوهای پروژه تعریف می‌شود	پروژه‌ها و برنامه‌ها

• ایجاد بسته‌های (packages) کاری ذاتی سازمان

۴-۴-۵- مدل جریان اولیه

مدل جریان اولیه، مانند مدل فاز اولیه برای راهنمای برخی سازمان‌ها بزرگ در ساخت نقشه راه SOA توسعه داده شده است و در حال استفاده شرکت‌های زیاد و شعب دولتی است.

نقشه راه جریان باید برای مهارت و هماهنگی کار سهامداران در رشته‌های مشترک (زیرساخت‌ها، معماری و غیره) برای ایجاد یک روش سازمانی حد وسط SOA برای فعالیت‌های مورد نیاز برای برنامه‌ریزی، توسعه و ارائه قابلیت‌های مورد نیاز SOA تعریف شده در فازها سازماندهی شود.

هر جریان گونه‌ای متفاوت از قابلیت‌های مدیریتی دارد، و ممکن است افق‌های زمانی متفاوت داشته باشد. ابزار نصب و برگزاری دوره‌های آموزشی ممکن است نسبتاً کوتاه مدت؛ ارائه و اجرای پروژه‌های معماری ممکن است میان مدت و تحویل منافع کسب و کار ممکن است دراز مدت باشد.

۵- تجزیه و تحلیل مدل‌ها

تحلیل SWOT در فرآیند تشکیل استراتژی جزء بسیار مهم محسوب می‌شود. تحلیل فرصت‌ها و تهدیدات خارجی اساساً برای ارزیابی این امر به کار می‌رود که بررسی شود، آیا سازمان می‌تواند از فرصت‌ها استفاده کرده و تهدیدات را به حداقل برساند و همچنین این تجزیه و تحلیل نقاط ضعف برای بررسی عملکرد داخلی شرکت‌ها (مانند روندهای کاری اثر بخش و تحقیق و توسعه) حائز اهمیت است. همچنین این روش تحلیلی را به همراه سایر ابزارهای استراتژیک مانند QFD^{۱۱} و BSC (کارت امتیازی متوازن) نیز می‌توان بکار برد. در نتیجه تحلیل SWOT قادر است به سازمان‌ها کمک کند وضعیت کلی خود را نسبت به سایر رقبا مورد سنجش و ارزیابی قرار بدهند [۱۵].

از آنجایی که نقشه راه برای اتخاذ SOA در سازمان است. بنابراین اهداف را در سه سطح مدیران، سازمان، اجزا قالب روش کارت امتیازی متوازن نمایش می‌دهیم. یکی از روشهای ارزیابی روش کارت امتیازی متوازن می‌باشد، در کارت امتیازی متوازن چهار جنبه شامل جنبه مالی، جنبه مشتری، جنبه فرآیند، جنبه یادگیری وجود دارد. اهداف مورد نظر هر جنبه مشخص و با شاخص‌ها اندازه گیری می‌شود که آیا به این اهداف رسیده‌ایم یا خیر [۱۵]. در این مقاله به طریق دیگری از کارت امتیازی متوازن استفاده شده است. از آنجایی که معیارهای مستند و مستدل برای ارزیابی نقشه راه وجود ندارد از قیاس بین چند نقشه راه برای کسب نتایج بهره گرفته شده است. برای این کار ماتریس SWOT را با کارت امتیازی متوازن ترکیب کرده تا بهتر بتوان قیاس را انجام داد. اهداف در قالب کارت امتیازی متوازن نشان داده شده است.

چهار جنبه کارت امتیازی متوازن متناسب با هدف مقاله به منظر مالی، منظر سازمان، منظر فرآیند، منظر حاکمیت تغییر کرده است. منظر مالی دیدگاه و اهداف مدیران، سهامداران و به نوعی چشم‌انداز استراتژیک سازمان را شامل می‌شود. هدف اصلی منظر سازمان شفاف سازی است. از آنجا که هدف از اتخاذ SOA مقیاس‌پذیری، جابجایی کد، حذف مرزها و... است. بنابراین سازمان به دنبال اهداف شفاف سازی می‌باشد. شفاف سازی شامل شفاف سازی زمان، مکان، انتقال، دسترسی، مهاجرت، جابجایی منابع (Relocation)، تکرار، همروندی، خرابی باشند. همه این موارد و اهداف مرتبط به منابع انسانی سازمان در این سطح قرار می‌گیرند. منظر فرآیند به نوعی به دنبال همان اهداف سازمانی می‌باشد که شامل شفاف سازی زمان، مکان، انتقال، دسترسی، همروندی است. که در سطح جزئی‌تر

۴-۴-۴- جریان‌های نقشه راه

تحول SOA بر سراسر دیسپلین‌های درون فناوری اطلاعات و کسب و کار تاثیر می‌گذارد. در بعضی سازمان‌ها بسیاری از واحدهای تشکیلاتی وجود دارند که درگیر عمل و بهبود فن آوری هستند و نیاز برای تغییر هماهنگی سازمان دارد. مفهوم جریان را برای ارائه تمرکز موضوع چشم انداز سهامداران که مستقل از ساختار فعلی سازمان است، استفاده می‌کنیم. این به فعالیت‌ها مدیریت تغییر، آسانتر کردن بیشتر مدیریت و هماهنگی، فراهم کردن تمرکز بیشتر بر موضوعات خاص را اجازه می‌دهد. از این جریان‌ها برای موارد زیر استفاده می‌کنیم:

- سازماندهی برنامه مدیریت تغییر
- جدایی نقطه نظر و دیدگاه‌ها از یکدیگر به عبارت دیگر ارائه تقسیم‌بندی منطقی دیدگاه‌ها
- تسهیل اختلاف همکاری سازمان بر توسعه قابلیت‌ها ارائه
- تعریف قابلیت‌های مورد نیاز برای حمایت از هر فاز تبدیل SOA

جدول ۴- اهداف اتخاذ SOA در چارچوب BSC

اهداف	اقدامات پیشنهادی
بازگشت سرمایه‌گذاری، کاهش هزینه‌ها، مدل سرمایه، استفاده مجدد سرمایه‌های موروثی (فناوری اطلاعات)، کاهش زمان بازاریابی، کاهش کل زمان مالکیت، ارزش منافع SOA سازمانی	تعریف محدوده پروژه‌ها، درگیر کردن سهامداران، مشخص کردن استراتژی سازمان، تعریف شفاف نیازمندی‌ها، مستندسازی طرح و بودجه، مستندسازی استراتژی فناوری اطلاعات، آنالیز ارزش منافع SOA سازمانی
چابکی، پشتیبانی چندین نوع مشتری، مقیاس‌پذیری، انعطاف‌پذیری کسب و کار و فناوری اطلاعات، همسویی کسب و کار و فناوری اطلاعات، کاهش هزینه‌های سخت افزار، نفوذ مهارت‌های توسعه، فراهم کردن پلی بین فناوریهای ناسازگار، کاهش هزینه‌های مدیریت، قابلیت نگهداری و پشتیبانی	ارزیابی آمادگی سازمان، فرهنگ سازی، درک افراد از SOA، شناسایی سیستم‌های دارای پتانسیل، امکان‌سنجی پروژه و تأیید عملکرد، اعتبارسنجی عملیاتی، پایلوت اولیه، مستند سازی کسب و کار موجود، کشف و تعیین فرآیندهای کسب و کار، آنالیز پتانسیل SOA سازمانی، تکامل معماری مرجع
اجتماع سرویس، موازی در توسعه، دسترس‌پذیری سرویس، انتخاب، کیفیت، کاهش هزینه، مشخص کردن خصوصیات سرویس (شامل رفتاری، سیاسی، ساختاری)	انتزاع سرویس، اتصال سست سرویس، مخزن سرویس، رجیستری سرویس، معماری SOA، به اشتراک گذاری استراتژی سرویس، مدل داده استاندارد، متدلوژی تکرار شونده توافق روی رابط سرویس‌ها، مستند سازی سرویس، امکان‌پذیر بودن سرویس، کلیات طراحی SOA سازمانی
امنیت، سیاستها، وجود استانداردها و پلتفرم باز، مدیریت موثر چرخه حیات اجزاء سرویس و فرآیند کسب و کار (برنامه‌ریزی، انتشار، کشف، رجیستری، نسخه، مدیریت و امنیت)، موافقت‌های سطح سرویس SLA، پروسیجر تست SOA، یکپارچه‌سازی	سازمان متمرکز، برپایی مرکز تعالی، رجیستری SOA، نظارت، کیفیت سرویس، اعمال سیاست‌ها، شناسایی و استفاده مناسب از پروتکل‌های امنیتی، مدل حاکمیت، زیرساخت سازی، ارزیابی ریسک‌ها، مشخص کردن نقاط عطف، کشف و رجیستری و فرآیند انتشار سرویس، SLA

یعنی فرآیند و سرویس اجرا می‌شوند. به دنبال اهداف خود- استار (self-stars) نیز هست، Self-Star سیستم‌ها شامل self-managing, self-healing, self-optimizing, self-configuring است. در مورد سرویس‌ها به دنبال چنین اهدافی هستند، همچنین در قسمت مزایای SOA این مطلب بیان شده است. که به دنبال استقلال زیاد سرویس‌ها، با اتصال سست و انتزاع سرویس‌ها هستند. از اهداف دیگر کشف و رجیستری سرویس، جستجوی سرویس برای قابلیت استفاده مجدد سرویس هستند. در منظر حاکمیت، اهداف غیر کارکردی مانند امنیت، سیاست، یکپارچگی و... را مورد توجه قرار دارد.

ماتریس SWOT دارای چهار بخش نقاط قوت و ضعف عوامل درونی و فرصت‌ها و تهدیدهای عوامل بیرونی است. ما در تحقیق خود از عوامل بیرونی صرف‌نظر می‌کنیم. فقط بر عوامل درونی تکیه کرده و نقاط ضعف و قوت هر نقشه راه بررسی می‌شود.

اهداف مورد نظر برای اتخاذ SOA در قالب کارت امتیازی متوازن رسم شده است. سپس عواملی که هر نقشه راه برای رسیدن به این اهداف انجام داده یا نداده در ماتریس SWOT مشخص می‌گردند. و در آخر جداول SWOTها با هم مقایسه می‌شوند. در ابتدا کارت امتیازی متوازن اهداف اتخاذ SOA را در جدول ۴ رسم می‌شود.

حال با توجه به اهداف بالا نقشه‌راه‌های ارائه شده در قالب SWOT در جداول جداگانه‌ای بیان می‌شوند. در ابتدا نقشه راه IowaSOA بررسی می‌شود. اعمالی که در خصوص برآوردن اهداف انجام داده در نقاط قوت و موارد انجام نداده در نقاط ضعف قرار می‌گیرد. چنانچه بیان شد نقشه راه IowaSOA با سیر تکامل آن بیان شده است و بر سه جنبه سازمان، حاکمیت و فناوری بسیار تأکید دارد. که در هر سیر تکامل یکسری اهداف و اعمال را مورد توجه قرار داده است که بصورت مختصر در جدول ۵ آنها را بیان می‌کنیم.

روش بیکن به صورت سه فاز بیان شده است که در فاز اول به تمرکز بر یکپارچگی، معرفی SOA و شناسایی سرویس‌های قابل استفاده مجدد و زیرساخت‌سازی SOA می‌پردازد. در فاز دوم فرآیند سرویس را مد نظر قرار می‌دهد و بر شناسایی، کشف، انتشار و کنترل سرویس تمرکز می‌کند. در فاز سوم بر پویایی کشف سرویس شامل شناسایی سرویس‌هایی خارجی سازمان‌های شریک و استفاده از آنها تمرکز دارد. همانطور که می‌بینید این نقشه راه بسیار سطح بالا بود است. برخی از عوامل شکست مانند بودجه و زمانبندی، تعریف شفاف نیازمندی‌ها، مدل حاکمیت و طبقه‌بندی SLA در آن مشهود است. جدول ۶ مشخصات نقشه‌راه بیکن را نشان می‌دهد.

روش IBM از روش‌های است که مستندی کاملی در این زمینه در دسترس نویسنده نبوده است. بنابراین تحلیل آن براساس مستندات کم و شکل آن صورت می‌پذیرد، اگر نکاتی از آن در این تحلیل قرار نگرفته است به این علت می‌باشد. از ظاهر فلوجارت اتخاذ SOA، IBM پیداست تقریباً نقشه راه کامل‌تری از نقشه راه‌های قبلی است. در آن استراتژی و دیدگاه کسب و کار را مد نظر قرار داده است و جامع‌تر بنظر می‌رسد. مشخصات این نقشه راه در جدول ۷ بیان شده است.

روش CBDI بصورت جدولی و اطلاعات کم نقشه راه مدل اتخاذ اولیه و افزایشی و مدل جریان اولیه و افزایشی را توضیح داده است و براساس عناوین و اهداف بیان شده ما گزینه‌های مناسب را انتخاب و در نقاط قوت جدول SWOT آن قرار داده‌ایم. آن چنان که پیداست بیشتر اهداف منظرهای مورد نظر را پوشش داده است.

همانطور که جداول SWOT نشان می‌دهند؛ روش ایوو در چهار منظر BSC برخی از اهدافی را مد نظر قرار نداده است یکی از علل موفقیت در اتخاذ SOA عامل بودجه است برای حمایت مالی، سیاسی، فرهنگی نیاز به حمایت سهامداران و مدیران می‌باشد که هیچ بحثی در این خصوص انجام نشده است. ضعف دیگر در منظر مالی اتخاذ SOA، بیان شفاف نیازمندی‌ها است.

در منظر سازمان معایب آن شامل ارزیابی آمادگی سازمان، فرهنگ سازمانی است. مسلماً برای اتخاذ هر پروژه‌ای نیاز به امکان‌سنجی پروژه است ولی امکان‌سنجی پروژه با امکان‌سنجی سازمان متفاوت است. ممکن است در اجرای پروژه سیاست‌ها، بومی بودن، مسائل اجتماعی زیاد مطرح نباشد ولی در امکان‌سنجی سازمانی بسیاری از این مسائل قابل شهود است که ایوو به آن کمتر اهمیت داده است.

یک دیگر از علل شکست یک پروژه مقاومت افراد بخصوص کارکنان در برابر تغییر است که ایوو این مسئله را هم نادیده گرفته است. از منظر سرویس مهمترین ایراد وارد بر آن انتزاع سرویس است که این نیز یکی از مهمترین علل موفقیت است. در منظر حاکمیت ایوو یکی از مسایل کمتر توجه شده نظارت است که نظارت نیز یکی از علل موفقیت می‌باشد که در این نقشه‌راه دیده نشده است.

روش بیکن در منظر مالی یا همان منظر مدیران و سهامداران بسیار ضعیف عمل نموده است بسیاری از اهداف مهم در اتخاذ SOA را نادیده گرفته است.

جدول ۵- جدول SWOT برای نقشه راه آیووا

ضعفها	قوتها	
آنالیز ارزش منافع م. س. گ. سازمانی، آنالیز بودجه و زمان‌بندی، درگیر کردن سهامداران، مستندسازی استراتژی فناوری اطلاعات، تعریف شفاف نیازمندی‌ها	برنامه‌ریزی استراتژیک، مدیریت برنامه، ارزیابی ارزش م. س. گ.، تعریف محدوده پروژه، زیرساخت مشترک سرمایه	منظر مالی
ارزیابی آمادگی سازمان، فرهنگ سازی (درک افراد از م. س. گ.)، مستندسازی کسب و کار موجود، شناسایی و معین کردن فرآیندهای کسب و کار، آنالیز پتانسیل م. س. گ. سازمانی، پایلوت اولیه، همسویی فناوری کسب و کار	امکان سنجی پروژه و تایید عملکرد، اعتبارسنجی عملیاتی، معماری مرجع م. س. گ.، فراهم کردن منابع با مهارت‌های درست، سرمایه‌گذاری سطح سازمانی در صورت لزوم، شناسایی اجزاء دارای پتانسیل تبدیل به سرویس	منظر سازمان
انتزاع سرویس، به اشتراک‌گذاری سرویس، توافق روی رابطه سرویس، مدل داده استاندارد، اتصال سست سرویس	مدیریت سطح سرویس، رجیستری سرویس، مدلسازی سرویس‌گرا، استفاده مجدد، فناوری مرتبط برای پیاده‌سازی و اجزاء چرخه حیات سرویس، دایرکتوری و مخزن سرویس	منظر سرویس(فناوری)
برپایی مرکز تعالی، نظارت، ارزیابی ریسک‌ها، مشخص کردن نقاط عطف، کیفیت و یکپارچگی	پیاده‌سازی و بروزرسانی مدل حاکمیت، مدیریت سیاست، کشف و رجیستری سرویس، فرآیند انتشار سرویس، مدیریت هماهنگی، سیاست و ارتباطات، تعریف اصول سیاست‌های امنیتی، تعریف معیارها و طبقه‌بندی SLA، مدیریت زیرساخت‌های مشترک، چارچوب مشترک برای نیازمندی‌های غیر کارکردی (مانند امنیت، انعطاف‌پذیری، دسترسی‌پذیری و...)	منظر حاکمیت

جدول ۶- جدول SWOT برای روش بیکن

ضعفها	قوتها	
آنالیز طرح (تعریف محدوده پروژه)، آنالیز بودجه و زمان‌بندی، بازگشت سرمایه‌گذاری، چشم‌انداز استراتژی کسب و کار، مستندسازی استراتژی فناوری اطلاعات، تعریف شفاف نیازمندی‌ها	درگیری سهامداران، آنالیز ارزش م. س. گ.	منظر مالی
پایلوت اولیه	امکان‌سنجی، ارزیابی آمادگی سازمان، فرهنگ سازی، مشخص کردن فرآیند-های کسب و کار، برپایی و تکامل معماری مرجع، شناسایی سیستم‌های دارای پتانسیل، همسویی فناوری و کسب و کار، مدل‌سازی تکرار شونده	منظر سازمان
توافق روی رابط سرویس‌ها، اتصال سست سرویس، طراحی مدل م. س. گ.، مدل داده استاندارد، مدل‌سازی تکرار شونده	تعریف استراتژی سرویس، توسعه واسط‌های سرویس، هماهنگ کننده سرویس، کشف سرویس، قابلیت استفاده مجدد سرویس، کشف و انتشار و کنترل سرویس، پویایی کشف سرویس، یکپارچگی سرویس	منظر سرویس(فناوری)
مدل حاکمیت، وضعیت و ارزیابی ریسک‌ها، آنالیز محدودیت‌ها، مشخص کردن نقاط عطف، تهیه چارچوبی برای نیازمندی‌های غیر کارکردی (مانند انعطاف‌پذیری، دسترسی‌پذیری و...)، کیفیت، تعریف معیار و SLA، مدیریت زیرساخت مشترک، نظارت	ایجاد مرکز تعالی، تمرکز روی یکپارچگی، سیاست‌های امنیتی، مدیریت سرویس	منظر حاکمیت

جدول ۷- جدول SWOT روش IBM

ضعفها	قوتها	
مستند سازی طرح (تعریف محدوده پروژه)، زمان‌بندی، آنالیز ارزش منافع م. س. گ.، مستندسازی استراتژی فناوری اطلاعات، تعریف شفاف نیازمندی‌ها	در نظر گرفتن استراتژی و دیدگاه کسب و کار، برنامه‌ریزی، پذیرش شرکاء، اعتبارسنجی اولیه	منظر مالی
	اثبات مفاهیم، بازبینی عملیاتی، همسویی کسب و کار و فناوری اطلاعات، اعتبارسنجی اولیه، تمرکز بر سیستم‌های موروثی، شناسایی سیستم‌های دارای پتانسیل (فرآیندهای چابک و انعطاف‌پذیر)، مدلسازی معماری، شناسایی کسب و کار، فرهنگ سازی سازمان، پذیرش کسب و کار، ایجاد نمای کسب و کار سرویس‌گرا، ساخت مدل اولیه موروث as-is	منظر سازمان
انتزاع سرویس، اتصال سست سرویس، رجیستری سرویس، مخزن سرویس، مدل داده استاندارد، توافق روی رابط سرویس‌ها	قابلیت استفاده مجدد سرویس‌ها، توزیع سرویس‌ها، معماری سرویس‌گرا، پذیرش فناوری	منظر سرویس (فناوری)
مدیریت سیاست، کشف و رجیستری و انتشار سرویس، اصول سیاست‌های امنیتی، تعریف معیارها و طبقه‌بندی SLA، اتخاذ چارچوب مشترک برای نیازمندی‌های غیر کارکردی (مانند امنیت، انعطاف‌پذیری، دسترسی‌پذیری و...)، نظارت، مشخص کردن نقاط عطف، کیفیت و یکپارچگی، ارزیابی ریسک‌ها، مدیریت زیرساخت‌های مشترک	مدل حاکمیت، مرکز تعالی	منظر حاکمیت

جدول ۸- SWOT برای روش CBDI

ضعفها	قوتها	
مستند سازی طرح (تعریف محدوده پروژه)، زمان بندی، زمان بندی، آنالیز ارزش منافع م. س. گ، مستند سازی استراتژی فناوری اطلاعات، تعریف شفاف نیازمندیها	مدیریت سرمایه، در نظر گرفتن چشم انداز، استراتژی بودجه، مدیریت، اندازه گیری و نظارت، برنامه های اتخاذ، مدیریت، مدیریت سرمایه	منظر مالی
ارزیابی آمادگی سازمان، فرهنگ سازی سازمانی، شناسایی سیستم های دارای پتانسیل، امکان سنجی پروژه و تایید عملکرد، آنالیز پتانسیل م. س. گ	همسو کردن ظرفیت وابستگی ها به مهارت های اولیه، اثبات مفاهیم، معیار سازی و یادگیری از منابع، همسویی کسب و کار و فناوری اطلاعات، به اشتراک گذاری فرآیندهای کسب و کار، تعریف نقش ها و مسئولیت ها (برای ایجاد، اجراء، مدیریت و اداره محیط سرویس ها)	منظر سازمان
انتزاع سرویس، متدلوژی تکرار شونده	اتصال سست، سرویس های اولیه، کنترل معماری، استفاده مجدد سرویس های کلیدی، مدل داده منجر به معماری سرویس، اشتراک گذاری سرویس ها، سازگاری اطلاعات مشترک، مدیریت چرخه حیات سرویس (رجیستری، زمان اجراء، نظارت وساطت و...)، ابزار مدل سازی سرویس	منظر فرآیند (سرویس)
ارزیابی ریسک ها، مشخص کردن نقاط عطف	کنترل و مدیریت ریسک ها، برپایی مرکز تعالی، حاکمیت کلی یکپارچگی، مدیریت محیط، هماهنگی معماری، پروتکل های زیرساخت مشترک، استاندارد سازی و به اشتراک گذاری تکنیکی، رجیستری سرویس، امنیت، هماهنگی سیاست و مدیریت سرویس	منظر حاکمیت

SOA باشد تا بتواند تصمیم گیری بهتر جهت کسب اهداف و اتخاذ SOA داشته باشند. در کارهای آتی نویسندگان در نظر دارند نقشه راهی برای اتخاذ SOA ارائه دهند.

منابع

[1] M. Galinium and N. Shahbaz, *Factors Affecting Success in Migration of Legacy Systems to Service-Oriented Architecture (SOA)*, Master thesis Department of informatics, 2009.

[2] T. Erl, "SOA Principle of Service Design," *Online Upper Saddle River: Pearson Inc*, <http://www.whatissoa.com/p10.asp>, 2007.

[3] IBM Company, "Transparent Decision Services in SOA," <http://www.ilog.com/promotions/ppc/soa/?acc=gmarcomeu&gp=GaLP&source=cpc&cmpn=RulesSOAEnglish>, 2009.

[4] Chatterjee, Sandeep, and Webber, "Developing Enterprise Web Services: An Architect's Guide," *Upper Saddle River*, Prentice Hall, 2004.

[5] D. Linthicum, "What Level Is Your SOA? Choose for what you need and maybe a little better," <http://webservices.sys-con.com/read/47277.htm>, 2004.

[6] Sun Microsystems, "Assessing Your SOA Readiness," www.web.sun.com, 2004.

[7] M. Kavis, "SOA Critical Success Factors," <http://it.toolbox.com/blogs/madgreek/SOA-critical-success-factors-221189>, 2007.

[8] C. Lawrence, "Modernizing Legacy Systems using an SOA Approach," http://www.ibm.com/developerworks/library/ws-soadaptleg/index.html?S_TACT=105AGX20&S_CMP=EDU, 2007.

[9] Integrated Software Specialists, and Iowa Department of Administrative Services, "IT Enterprise Service-Oriented Architecture adoption roadmap," 2006.

بودجه و زمان بندی، آنالیز طرح، بازگشت سرمایه گذاری، تعریف شفاف نیازمندیها، چشم انداز کسب و کار از مهمترین معایب در منظر مالی روش بیکن می باشد. در منظر سرویس یکی از معایب وارده به این روش اتصال سست سرویس می باشد. و در منظر حاکمیت یکی از مهمترین علل موفقیت که مدل حاکمیت است، نادیده گرفته شده است همینطور نداشتن چارچوبی برای نیازهای غیر کارکردی یکی از ضعف های دیگر این روش است.

روش IBM با توجه به مستندات کم و شکل آن مورد تجزیه و تحلیل قرار گرفته است. در روش IBM از منظر مالی معایبی همچون تعریف شفاف نیازمندیها، مستند سازی طرح و زمان بندی را دارا می باشد. در منظر سرویس معایب انتزاع سرویس، اتصال سست سرویس، رجیستری و مخزن سرویس که از عوامل حیاتی موفقیت هستند، دیده می شود. برای منظر حاکمیت مسائلی همچون مدیریت سیاستها، اصول و استانداردهای امنیتی، SLA، چارچوبی برای نیازهای غیر کارکردی مانند انعطاف پذیری، دسترس پذیری، مدیریت زیرساخت مشترک و... در این روش در نظر گرفته نشده است.

روش CBDI که روش های مدل اتخاذ اولیه و افزایشی و مدل جریان اولیه و افزایشی را شامل می شود. در منظر مالی مستند سازی یا آنالیز طرح و زمان بندی، تعریف شفاف نیازمندیها، آنالیز ارزش منافع SOA جز معایب مطرح هستند. ارزیابی آمادگی سازمان، فرهنگ سازی سازمانی، شناسایی سیستم های دارای پتانسیل، امکان سنجی پروژه و تایید عملکرد نقاط ضعفی هستند که در منظر سازمان این روش وجود دارند. از منظر سرویس معایبی همچون انتزاع سرویس و متدلوژی تکرار شونده را مشاهده می شود. در منظر حاکمیت ارزیابی ریسک ها و مشخص کردن نقاط عطف را شاهد هستید.

۶- نتیجه گیری

در این مقاله چهار نقشه راه برای مهاجرت به SOA ارائه شدند و مورد بررسی قرار گرفته اند، همانطور که از جداول SWOT هر چهار نقشه راه پیداست هر کدام از آنها معایب و مزایایی خودشان را دارند که بطور خلاصه آنها مرور شدند. هر چهار نقشه راه اهداف را به طور کامل برآورده نمی کنند روش IBM و CBDI نقاط ضعف کمتری نسبت به روش بیکن و ایوو دارند. می توان بر اساس این نتایج بیان کرد هر چهار روش را برای اتخاذ SOA کامل نیستند و نیاز به نقشه راه جامعتری برای اتخاذ SOA در سازمان است.

همچنین انتظار می رود که این تحلیل بتواند کمک موثری به افراد اتخاذ کننده

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۹/۱۲/۷

تاریخ اصلاح: ۹۰/۵/۳

تاریخ قبول شدن: ۹۰/۵/۱۲

نویسنده مرتبط: طیبه آزادی، دانشکده فنی و مهندسی، دانشگاه آزاد اسلامی واحد تهران شمال، تهران، ایران.

- ¹ Service Oriented Architecture
² Strength, Weakness, Opportunity, Treat
³ Blanced Scorecard
⁴ Coarse-Grained
⁵ BPEL
⁶ Cutting edge
⁷ What is it for me?
⁸ Service Language Agreements
⁹ BEACON
¹⁰ Big Bang
¹¹ Quality Function Deployment

[10] The Beacon, "Service Oriented Architecture – delivering agility to the Travel domain," www.traventec.com/upload/documents, 2004.

[11] B. Traventec, "Roadmap for Adoption of Service Oriented Architecture," www.traventec.com/upload/documents, 2004.

[12] H. M. Hess, "Aligning technology and business: Applying patterns for legacy transformation," *IBM Systems Journal*, vol. 44, no. 1, pp. 25-45, 2005.

[13] R. Shah, M. Fiammante, and K. Jones, "SOA Project planning Aspects," *IBM Press*, www.informit.com, 2005.

[14] D. Sportt, "An Update to the SOA Adoption Roadmap Framework," *CBDJ Journal*, pp 20-34, 2008.

[15] R. Kaplan and N. David, *The Execution Premium: Linking Strategy to Operation for Competitive Advantage*, Harvard Business School Press, 2008.

[۱۶] ف. سید صالحی و ل. کفائی، "مدیریت استراتژی"، نشرسیمیا، ۱۳۸۹.



طیبه آزادی مدرک کارشناس کامپیوتر در گرایش مهندسی نرم‌افزار را از دانشگاه شهید چمران اهواز دریافت کرد. در حال دریافت درجه کارشناسی ارشد از دانشگاه تهران است. تحقیقات او در زمینه‌های مختلف معماری سرویس‌گرا می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

engineer.azadi@gmail.com



علی معینی مدرک کارشناسی و کارشناسی ارشد خود را از پردیس دانشکده‌های فنی دانشگاه تهران در سال ۱۳۶۵ اخذ کرده است. تحصیلات خود را در مقطع دکتری در دانشگاه ساسکس انگلستان در سال ۱۳۷۵ در زمینه بکارگیری کامپیوتر در تحلیل سیستم‌های دینامیکی غیرخطی ادامه داده است. علایق پژوهشی وی در زمینه اتوماتان‌های سلولی، محاسبات هوشمند، الگوریتم‌های تصادفی و بکارگیری اینگونه روش‌ها در تحلیل سیستم‌های اقتصادی - مدیریتی است. وی اکنون عضو گروه مهندسی پردیس دانشکده‌های فنی است و مسئولیت مرکز فناوری اطلاعات و فضای مجازی دانشگاه تهران را نیز عهده‌دار است.

آدرس پست‌الکترونیکی ایشان عبارت است از:

moeini@ut.ac.ir



ناصر مدیری مدرک فوق لیسانس در میکروالکترونیک از دانشگاه ساوث همپتون، انگلستان در سال ۱۹۸۶ دریافت کرد. او درجه دکترای در شبکه‌های کامپیوتری از دانشگاه ساسکس انگلستان در سال ۱۹۸۹ دریافت کرد. او سخنرانی در گروه مهندسی کامپیوتر در دانشگاه آزاد اسلامی زنجان، ایران است. علایق پژوهشی او عبارتند از مراکز عملیات شبکه، چارچوب برای تضمین امنیت شبکه، سازمان‌های مجازی، RFID، کنترل دسترسی، چرخه عمر محصول توسعه و چهارچوب برای شبکه‌های ایمن.

آدرس پست‌الکترونیکی ایشان عبارت است از:

nassermodiri@yahoo.com

مولد موتور گردش کار SWEG

مرتضی یوسف صنعتی^۱ سید حسن میریان حسین آبادی^۲

^۱ دانشکده مهندسی، دانشگاه بوعلی سینا، همدان، ایران

^۲ دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران

چکیده

بسیاری از سازمان‌ها برای انجام مسئولیت‌های خود از گردش کارهای مختلفی استفاده می‌کنند. لذا تولید کنندگان نرم‌افزار با توجه به این موضوع، از موتورهای گردش کار نرم‌افزاری برای سرویس‌دهی به اعمال پایه‌ای و سیاست‌های گردش کارهای موجود در سازمان‌ها استفاده می‌نمایند. از طرف دیگر تولید نرم‌افزارهای موتور گردش کار که کلیه نیازهای گردش کار سازمان‌ها را پاسخ دهد، نیاز به صرف هزینه و زمان زیادی دارد. در این میان با حضور روش‌های مهندسی خط محصول^۱ برای تولید نرم‌افزار مانند روش تولید نرم‌افزار مبتنی بر خانواده^۲، هزینه تولید بسیار کاهش یافته و در زمان بسیار کمتری نرم‌افزار با کیفیتی بسیار مطلوب ایجاد می‌گردد. در این مقاله نحوه ایجاد نرم‌افزارهای موتور گردش کار، با استفاده از روش مبتنی بر خانواده بررسی شده و با ارائه یک زبان مدل‌سازی مخصوص اعضا خانواده موتورهای گردش کار، مشکلات تولید چنین نرم‌افزارهایی به میزان قابل ملاحظه‌ای کاهش یافته است.

کلمات کلیدی: تولید نرم‌افزار مبتنی بر خانواده، زبان مدل‌سازی کاربرد، موتور گردش کار.

۱- مقدمه

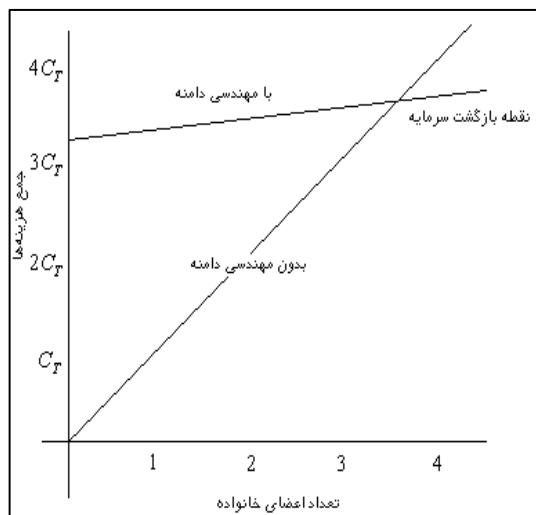
محصول، محصولات میانی، زمان لازم برای تولید، کیفیت محصول، هزینه، قابلیت استفاده مجدد، انعطاف‌پذیری و قابلیت تغییر می‌باشد. البته در این میان روش‌های دیگری نیز وجود دارند که می‌توانند در زمان کوتاه و با هزینه کم سامانه‌هایی مطلوب ایجاد نمایند که یکی از این روش‌ها، روش تولید نرم‌افزار مبتنی بر خانواده است. این روش دارای مزایای مهمی است که در بخش بعدی بطور کامل در مورد آن توضیح داده خواهد شد.

در روش تولید نرم‌افزار مبتنی بر خانواده اصل بر شناخت خانواده‌ای از نرم‌افزارها است که دارای مشابهت‌های زیادی هستند و می‌توان اعضای آن‌ها را با این روش تولید نمود. در این میان یکی از دامنه‌های مطلوب، دامنه سامانه‌های حاوی گردش کار می‌باشد. در مورد دامنه گردش کار و جزئیات مربوط به آن در بخش‌های بعدی توضیح داده شده است. ولی لازم به ذکر است که اعضای این دامنه بسیار زیاد هستند و دارای مشابهت‌های بسیاری نیز می‌باشند.

این دو خاصیت کلیدی باعث می‌شود تا با استفاده از روش مورد بحث بتوان سامانه‌های گردش کار را در زمانی کوتاه و با هزینه‌ای اندک تولید نمود. هدف این

در مهندسی نرم‌افزار روش‌های متفاوتی برای تولید سامانه‌های نرم‌افزاری وجود دارد که هر یک دارای مزایا و معایبی می‌باشند و بسته به شرایط و اهداف، می‌توان یکی از آن‌ها را به خدمت گرفت. از جمله آن‌ها می‌توان به روش‌های مبتنی بر شی‌گرایی مانند RUP^۳ و روش‌های ساخت‌یافته مانند SSADM^۴ اشاره نمود. در تمام روش‌های موجود مراحل و گام‌های پیش‌بینی‌شده‌ای وجود دارد که با انجام آن‌ها می‌توان به محصول نهایی دست یافت. در این میان نیروها و کارشناسان با ایفای نقش‌های خود، تعامل با دیگر نقش‌ها و همکاری با دیگران انجام مراحل را عهده‌دار می‌باشند. بعلاوه در همه روش‌ها گام‌های اساسی تولید نرم‌افزار از جمله تحلیل، طراحی و پیاده‌سازی وجود دارد.

ابتدا نیازهای مشتری تبیین می‌شود، سپس طراحی انجام شده و در نهایت پیاده‌سازی صورت می‌پذیرد. بعد از این مرحله نیز تست انجام خواهد شد. در این میان نکته حائز اهمیت، تعداد مراحل، تعداد نقش‌ها و نیروهای لازم برای تولید



شکل ۱- وجود یا عدم وجود مهندسی دامنه

در این روش اگر هزینه لازم برای انجام مهندسی دامنه I فرض گردد، هزینه تولید هر نرم‌افزار پس از انجام مهندسی دامنه C_f می‌شود. لذا اگر مهندسی دامنه با موفقیت انجام شده باشد، $C_f < C$ خواهد بود. پس کل هزینه $I + C_f * N$ می‌شود. نشان داده شده است که برای $N > 3$ میزان کل هزینه در صورت داشتن مهندسی دامنه موفقیت‌آمیز، کمتر از عدم وجود مهندسی دامنه است، پس $C * N < I + C_f * N$. این مطلب در شکل ۱ قابل مشاهده می‌باشد.

همچنین در مورد زمان، می‌توان گفت که زمان لازم برای تولید اولین محصول توسط این روش، بدلیل انجام مهندسی دامنه زیادتر از دیگر روش‌ها می‌باشد، یعنی اگر زمان لازم برای انجام مهندسی دامنه، تولید محیط و توصیف کاربرد توسط زبان مدل‌سازی به ترتیب T_d ، T_e و T_s باشد، لذا $T < T_d + T_e + T_s$ خواهد بود. اما بدلیل این‌که مهندسی دامنه و تولید محیط یکبار صورت می‌پذیرد، پس برای تولید محصولات بعدی زمان لازم فقط T_s یعنی همان زمان لازم برای توصیف کاربرد توسط زبان مدل‌سازی خواهد بود، لذا کاربردهای بعدی سریعتر از سایر روش‌ها، تولید می‌شوند و در نهایت $T_d + T_e + T_s * N < T * N$ می‌باشد.



شکل ۲- نقش‌ها در روش تولید نرم‌افزار مبتنی بر خانواده [۵]

در روش مورد بحث نقش‌های متفاوتی فعالیت می‌کنند. این نقش‌ها در شکل ۲ نشان داده شده‌اند. یکی از آن نقش‌ها مهندس دامنه است. او خانواده‌ای را که ارزش سرمایه‌گذاری داشته باشد، پیدا می‌کند. یعنی دامنه را مشخص می‌نماید.

مقاله بررسی و تولید مولد موتور گردش کار با استفاده از روش تولید نرم‌افزار مبتنی بر خانواده است. ابتدا روش تولید نرم‌افزار مبتنی بر خانواده به همراه مزایا و معایب آن بررسی می‌شود. سپس جزئیات مربوط به دامنه گردش کار مطرح خواهد شد. همچنین مزایای اعمال این روش بر روی دامنه کار نیز بوضوح قابل مشاهده خواهد بود. سپس زبان گردش کار، معماری مربوط به مولد موتور گردش کار و مثالی از نحوه ایجاد یک موتور نیز ارائه می‌گردد.

۲- روش تولید نرم‌افزار مبتنی بر خانواده

در بیشتر سازمان‌ها تحویل دیرهنگام سامانه‌های مورد نیاز و نیز گران بودن تولید نرم‌افزارها از مهمترین مشکلات به شمار می‌رود. از این رو وجود روش‌هایی که بتوانند با سرعت و بدون هزینه زیاد نیازهای مشتریان را پاسخگو باشند، ضروری به نظر می‌رسد. یکی از این روش‌ها، روش تولید نرم‌افزار مبتنی بر خانواده است که در آن فرایندها و عناصر خاصی جهت رسیدن به اهداف مهمی که از جمله آن‌ها افزایش کارایی پروژه تولید [۱]، بهبود کیفیت محصول و خدمات سازمان‌ها [۱]، تولید سریع، دقیق و کم هزینه سامانه‌ها می‌باشد، تعبیه شده است.

به عبارت دیگر این روش الگویی برای پروژه‌های تولید نرم‌افزار است که در آن از فشار بین تولید سریع و مهندسی مطمئن و دقیق کاسته شده است. به عبارت دیگر این روش یک پروژه سیستماتیک برای توسعه مجموعه‌ای از نرم‌افزارها است که دارای خصوصیات مشترک زیادی می‌باشند [۲] و می‌تواند در حل مشکلات زیادی به مشتریان و توسعه‌دهندگان نرم‌افزار کمک نماید. در زیر تعدادی از این مشکلات ذکر شده‌اند.

۱. عدم رضایت از کارایی پروژه یا محصول موجود
۲. نیاز به کاهش هزینه و زمان تولید
۳. کمبود کارمند
۴. نیاز به پاسخگویی سریع به درخواست‌کننده نرم‌افزار
۵. پیچیدگی مدیریت و نگهداری بسیاری از محصولات نرم‌افزاری

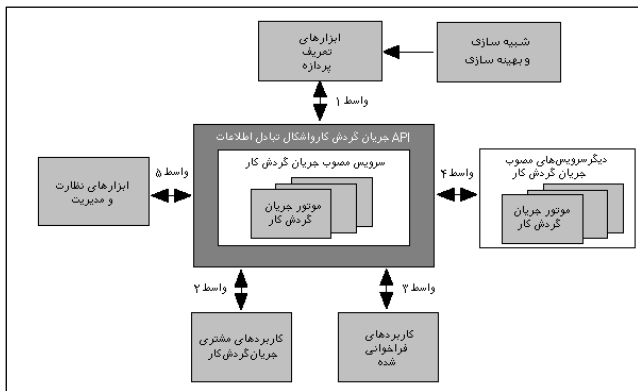
نکته قابل توجه این است که بطور نمونه شرکت‌های بزرگی مانند HP، Nokia، Philips با استفاده از این روش توانسته‌اند زمان و هزینه مهندسی را کاهش داده و نرخ تولید را از ۳ به ۵۰ برسانند [۳].

به طور کلی در این روش خانواده‌ای از نرم‌افزارها شناسایی می‌گردد. منظور از خانواده، مجموعه‌ای از نرم‌افزارهایی است که دارای مشترکات زیادی هستند یا به عبارت دیگر مشترکات آن‌ها می‌توانند هسته اصلی هر عضو این مجموعه را تشکیل دهند. بنابراین با شناخت خانواده و اعضای آن و تحلیل جنبه‌های مشترک می‌توان گام‌های ابتدایی این روش را به انجام رساند. اما قبل از تحلیل بررسی جنبه‌های اقتصادی موضوع جهت سرمایه‌گذاری روی خانواده از اهمیت ویژه‌ای برخوردار است. از این رو پس از سرمایه‌گذاری، تحلیل دامنه و انجام یکسری فعالیت‌ها و تولید محصول میانی، ابزارهایی تولید می‌گردند که توسط آن‌ها به راحتی می‌توان هر عضوی از خانواده مورد نظر را به سرعت، دقت و کیفیت مطلوب تولید نمود. یکی از دلایلی که این روش به سرعت می‌تواند محصول خود را تولید نماید، زمان لازم برای تولید یک عضو از خانواده است.

در مقایسه روش تولید نرم‌افزار مبتنی بر خانواده با سایر روش‌ها می‌توان به کمتر بودن هزینه و زمان تولید در این روش دست یافت. اگر تولید هر محصول (T) واحد زمان ببرد و نیز تولید N محصول مد نظر باشد، زمان لازم برای تولید آن‌ها در سایر روش‌ها $T * N$ خواهد بود. شبیه همین مطلب نیز در مورد هزینه تولید نیز برقرار است، یعنی اگر هزینه لازم برای تولید محصول C فرض شود و N محصول مورد نیاز باشد، لذا کل هزینه برابر $C * N$ خواهد بود و این در حالی است که در روش مورد مطالعه بحث هزینه و زمان به گونه‌ای دیگر می‌باشد.

۳- سامانه مدیریت گردش کار و مدل مرجع آن

همه سازمان‌ها دارای اهدافی هستند و برای رسیدن به آن اهداف، پروژه‌های متفاوتی را اجرا می‌نمایند. این پروژه‌ها از فعالیت‌های سازمانی که برای رسیدن به هدف با ترتیب مشخصی اجرا می‌شوند، تشکیل شده‌اند. از این رو خودکارسازی این پروژه‌ها کمک شایانی به افزایش سرعت انجام کارها و کارایی سازمان‌ها می‌نماید [۶]. بنابراین خودکارسازی تمام یا قسمتی از پروژه‌ها در مدتی که اسناد، اطلاعات یا وظایف از یک همکار به همکار دیگر جهت انجام عملیات بر طبق مجموعه‌ای از قوانین سازمانی فرستاده می‌شود، گردش کار نامیده می‌شود. ولی گردش کار باید مدیریت شود که این امر بر عهده سامانه‌های مدیریت گردش کار است که در این میان مفهوم مهم مدیریت گردش کار به عنوان یک فناوری کلیدی برای هماهنگ نمودن پروژه‌های تجاری متفاوت نقش بسزایی را ایفا می‌نماید [۷]. پس سامانه‌های مدیریت گردش کار علاوه بر مدیریت، قابلیت تعریف پروژه، اجرا، مدل‌سازی [۸] و هماهنگ کردن آن را دارا می‌باشند. لازم به ذکر است که پروژه‌های مورد بحث باید از روی مدل‌های سازمانی توصیف شوند [۸]. همچنین این سامانه‌ها فعالیت‌ها و روند آن‌ها را مدیریت نموده، در زمان‌های معین یا شرایط معین اعمال لازم را انجام می‌دهند.



شکل ۴- مدل مرجع سامانه‌های گردش کار [۹]

شکل ۴ مدل مرجع اغلب سامانه‌های مدیریت گردش کار را نشان می‌دهد که در مورد هر یک از اجزاء مدل بطور مختصر توضیح داده می‌شود.

خدمت مصوب گردش کار

خدمت مصوب گردش کار یک محیط اجرایی را فراهم می‌آورد که در آن پروژه ایجاد می‌شود، فعال می‌گردد و از یک یا چند موتور گردش کار بهره می‌گیرد و مسئولیت تعامل با منابع خارجی را در صورت لزوم بر عهده دارد.

موتور گردش کار

موتور محیطی اجرایی برای نمونه گردش کار فراهم می‌کند و نوعاً امکانات زیر را نیز فراهم می‌آورد:

- تفسیر تعریف پروژه‌ها
- ایجاد، کنترل و نمونه‌سازی پروژه‌ها
- ورود همکار به سامانه و خروج از آن
- انجام اعمال مدیریتی
- انجام اعمال نظارتی
- ناوبری بین فعالیت‌های ترتیبی یا موازی

در اصل مهمترین جزء سامانه همین موتور است و مهمترین کارها را موتور انجام می‌دهد.

وی همچنین به بررسی اعضای متفاوت دامنه می‌پردازد و مشترکات و تمایزات آن‌ها را به خوبی درمی‌یابد. این عمل تحلیل مشترکات و تمایزات نامیده می‌شود.

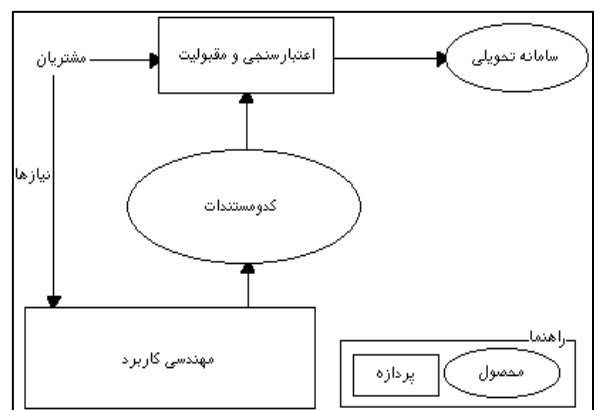
گام بعدی، مدل‌سازی اعضای خانواده و ایجاد مدل سراسری است که بر طبق تحلیل‌های بدست آمده صورت می‌گیرد. مهندس دامنه با طراحی زبانی که زبان مدل‌سازی کاربرد نامیده می‌شود، خانواده را مدل می‌نماید. تاکنون مهندس دامنه دو محصول ایجاد کرده است. اول مستندات مربوط به تحلیل مشترکات و تمایزات و دوم زبان مدل‌سازی کاربرد. ولی برای استفاده از زبان و تولید محصول توسط آن، دو راه حل وجود دارد. اولین راه حل ایجاد یک مترجم برای زبان مدل‌سازی است تا با استفاده از آن بتوان توصیفاتی را که نوشته می‌شود ترجمه، و در نهایت به یک محصول عملیاتی تبدیل کرد. راه حل دوم ایجاد یک ترکیب‌کننده است.

کار این ترکیب‌کننده الحاق کردن اجزاء کوچک نرم‌افزاری است که در تناظر با دستورات زبان مدل‌سازی می‌باشند. پس از الحاق، محصول عملیاتی مطلوب تولید می‌گردد. در صورت استفاده از ترکیب‌کننده چند نکته حائز اهمیت است. ابتدا باید یک کتابخانه از اجزاء نرم‌افزاری پیاده‌سازی شده که در اعضای دامنه وجود دارند ایجاد کرد و دوم یک تناظر بین دستورات زبان مدل‌سازی و اجزاء نرم‌افزاری برقرار نمود تا ترکیب‌کننده بتواند با توجه به تناظر اجزاء مطلوب را انتخاب نماید.

پس از تولید مترجم یا ترکیب‌کننده، نکته مهم ایجاد محیطی برای استفاده و توصیف نمودن کاربرد است. تولید این محصول نیز بر عهده مهندس دامنه است و وی باید محیطی را جهت ایجاد توصیف نرم‌افزار در اختیار دیگر نقش‌ها قرار دهد. تنها مشکل باقیمانده چگونگی فرایند تولید نرم‌افزار با استفاده از این محصولات است که در این میان مهندس دامنه با طراحی فرایند تولید نرم‌افزار و مستندسازی آن، مشکل را رفع می‌نماید. در این مرحله تقریباً مهندس دامنه تمام نقش خود را ایفا نموده است و نوبت به ایفای نقش توسط مهندس کاربرد می‌رسد.

مهندس کاربرد نیازهای مشتری را خوبی تحلیل می‌نماید. سپس با استفاده از زبان مدل‌سازی تولیدی، نرم‌افزار مورد نیاز مشتری را در محیطی که برای وی توسط مهندس دامنه تولید شده است، توصیف می‌نماید. در نهایت با استفاده از مترجم یا ترکیب‌کننده نرم‌افزار مورد نیاز مشتری را تهیه می‌کند. البته قبل از تحویل، ابتدا از صحت عملکرد آن با جلب رضایت مشتری اطمینان حاصل می‌نماید. در صورت وجود کاستی و عدم رضایت مشتری، نقص‌های موجود را با بهبود و اصلاح توصیف، رفع نموده و مجدداً نرم‌افزار را تولید می‌نماید و به مشتری تحویل می‌دهد.

فرایند تولید محصول در شکل ۳ نشان داده شده است. پس از ارائه این روش، در بخش بعد سامانه مدیریت گردش کار و مدل مرجع آن مورد تحلیل و بررسی قرار خواهد گرفت.



شکل ۳- فرایند تولید محصول

ابزارهای تعریف پروژه

از این رو با توجه به روش مذکور، مهندس دامنه پس از شناخت اجمالی دامنه، یک مدل اقتصادی تهیه می‌کند تا تصمیمات بعدی طبق آن اتخاذ شود. ولی در اینجا دلیل وضوح اقتصادی بودن این کار، از تهیه مدل اقتصادی دامنه صرف‌نظر می‌گردد. گام بعدی تحلیل مشترکات و تمایزات است که مهندس دامنه باید آن را انجام دهد. پس از مطالعه روی سامانه‌های متفاوت و بررسی استانداردهای در دست تدوین برای سامانه‌های گردش کار، مشترکات و تمایزات بدست آمده به تفکیک از قرار زیر می‌باشد:

❖ مشترکات

- وجود پروژه‌های مختلف
- وجود وظایف و فعالیت‌های متفاوت
- وجود کنشگرهای سازمانی
- وجود اقلام گردش کار
- وجود تصمیم‌گیری مبتنی بر قاعده
- کنترل و نمونه‌سازی پروژه‌ها
- کنترل و نمونه‌سازی فعالیت‌ها
- وجود لیست کارها و وظایف
- تخصیص‌دهی وظایف
- وجود ابزارهای تعریف و تفسیر پروژه [۱۲] و فعالیت‌های آن
- وجود ابزارهای مربوط به تعریف و تفسیر منطق مسیرگزینی اقلام گردش کار
- ایجاد امنیت [۱۳] با هویت‌شناسی و کنترل دسترسی [۱۴] برای انجام وظایف محوله بصورت نقش محور [۱۵] یا سیاست‌های دیگر

❖ تمایزات

- تعداد و نحوه عملکرد پروژه‌ها
- تعداد و نحوه عملکرد فعالیت‌ها
- تعداد و ویژگی‌های کنشگرها
- اقلام گردش کار با صفات متفاوت
- منطق مسیرگزینی [۱۶]
- منطق اتخاذ تصمیم
- گزارش‌گیری‌های متنوع
- انجام تارنگاری
- وجود و شکل الگوهای رفتاری [۱۷، ۱۸، ۱۹] متفاوت از جمله [۲۰] And-Join, Or-Join

موارد ذکر شده که نتایج تحلیل مشترکات و تمایزات می‌باشد به عنوان ورودی گام بعدی روش مورد بحث استفاده می‌شود. مشترکاتی که بین سامانه‌های گردش کار وجود داشت، به عنوان قسمت پایه‌ای و الزامی همه اعضا در تولید آن‌ها نقش کلیدی ایفا می‌نماید. از این رو باید موارد مشترک پیاده‌سازی گردیده، در هنگام تولید یک عضو از خانواده به آن متصل گردد. اکنون باید زبانی برای تولید تمایزات اعضا طراحی گردد. پس از طراحی زبان، مترجم آن و محیط تولید، مهندس کاربرد می‌تواند با استفاده از آن، هر یک از ویژگی‌های مورد نظر را برای سامانه مطلوب مشتری، تولید نماید. لذا طراحی زبان کاری بسیار مهم است که باید با دقت انجام شود. در قسمت بعد جزئیات مربوط به زبان گردش کار توضیح داده می‌شود.

این ابزارها قابلیت تحلیل، مدل کردن و مستندسازی پروژه‌ها را فراهم می‌کنند، بطوریکه خروجی حاصل از این ابزارها توسط موتور گردش کار قابل تفسیر و اجرا است [۱۰].

ابزارهای نظارت و مدیریت

این ابزارها برای نظارت بر حسن اجرای پروژه‌ها و نیز مدیریت آن‌ها تعبیه شده‌اند و گزارش‌های متنوعی را در مورد پروژه‌ها، فعالیت‌ها، حالات آن و سایر موارد مطلوب دیگر به مدیر یا مدیران سامانه ارائه می‌نمایند.

کاربردهای فراخوانی شده

در بعضی موارد برای انجام پروژه لازم است کاربردی خاص، فراخوانی شود. در این صورت واسط کاربری خاصی برای این امر تعبیه شده است که با استفاده از آن، خدمت مصوب گردش کار نیازمندی‌های کاربرد را به وی داده و اطلاعات لازم را از آن دریافت می‌کند.

کاربردهای مشتری گردش کار

کاربردهای مشتری، کاربردهایی هستند که مشتری گردش کار را در رفع نیازهای متفاوتی از جمله یادآوری و نمایش وظایفی که به آن‌ها محول شده، پیام‌هایی که به آن‌ها رسیده، یاری می‌دهند.

دیگر خدمات‌های مصوب گردش کار

خدمت مصوب سامانه می‌تواند با خدمات‌های مصوب دیگر سامانه‌ها در ارتباط باشد. لذا واسطی نیز برای این کار تعبیه شده است تا خدمات‌های مصوب با کمک آن بتوانند نیازهای خود مانند ردوبدل اطلاعات را مرتفع نمایند. در اینجا بطور مختصر مدل مرجع توضیح داده شد که برای مطالعه بیشتر می‌توان به [۱۰، ۱۱] مراجعه نمود.

۴- بررسی خانواده سامانه‌های حاوی گردش کار

در بخش‌های قبلی، روش تولید نرم‌افزار مبتنی بر خانواده و سامانه مدیریت گردش کار به همراه مدل مرجع آن بطور مستقل مورد بررسی قرار گرفتند. در این بخش با استفاده از روش مطرح شده، خانواده سامانه‌های حاوی گردش کار مورد مطالعه قرار می‌گیرد.

تعداد اعضای این خانواده بسیار زیاد است. زیرا هر سامانه‌ای که به نوعی حاوی گردش کار باشد، عضو این خانواده خواهد بود. لذا بطور نمونه می‌توان به موارد زیر اشاره نمود.

- سامانه‌های مربوط به اتوماسیون عملکرد ادارات، انبارها، شرکت‌های خصوصی و دولتی و نهادهای اجتماعی دیگر مانند تعاونی‌ها و انجمن‌ها
- سامانه‌های مربوط به امور مزایده و مناقصه
- سامانه‌های مدیریت فعالیت‌ها در کارخانه‌های متفاوت
- سامانه‌های آموزش در دانشگاه‌ها و سایر مراکز آموزشی غیر آموزش عالی
- سامانه‌های بیمارستان‌ها

با توجه به این‌که فقط به تعدادی از سامانه‌های گردش کار اشاره شد، در حالیکه همین موارد تعداد زیادی از سازمان‌ها را تحت پوشش قرار می‌دهد، لذا تعداد اعضای این خانواده بسیار زیاد است. بنابراین با توجه به توضیحاتی که در مورد روش تولید نرم‌افزار مبتنی بر خانواده ارائه گردید، این روش می‌تواند اعضای این خانواده را با زمان و هزینه اندک ایجاد نماید.

مستقلی ندارد. بنابراین می‌توان کنش‌گرهای سازمان را گروه‌بندی نمود. البته همانطور که قبلاً ذکر شد، در رابطه با کلیات کنش‌گرهای سازمان ذکر مواردی مانند تعریف نمودار سازمانی، انتساب نقش‌ها به بخش‌های سازمان، تعریف کارمندان و انتساب نقش‌ها به آنان نیز دارای اهمیت است. به همین دلیل قواعدی نیز برای این امر تعبیه گردیده که در شکل ۸ تا شکل ۱۱ قابل مشاهده می‌باشند.

```
WorkflowActorSpecification ::=
    ExtendedAttributeDefinition
    ACTORS{ WorkflowActorDeclaration }
ExtendedAttributeDefinition ::=
    ExtendedAttributeItemDefinition
    ExtendedAttributeDefinitionList | λ
ExtendedAttributeItemDefinition ::=
    < AttributeName : AttributeType OtherProperties >
OtherProperties ::= DefaultValue AllowNull
DefaultValue ::= { DEFAULT : Value } | λ
AllowNull ::= , ALLOWNULL | λ
ExtendedAttributeDefinitionList ::=
    ExtendedAttributeItemDefinition
    ExtendedAttributeDefinitionList | λ
```

شکل ۶- تعریف نوع کنش‌گر

```
WorkflowActorDeclaration ::=
WorkflowActorItem WorkflowActorList
WorkflowActorItem ::= <Name, Description, ActorType,
    UserName, Password
    ExtendedAttributeDeclarationList>
ActorType ::= HUMAN | ROLE | ACTORSYSTEM |
    ORGANIZATIONDEPARTMENT | GROUP
ExtendedAttributeDeclarationList ::= ,
    ExtendedAttributeDeclarationItem
    ExtendedAttributeDeclarationList | λ
ExtendedAttributeDeclarationItem ::= Value | NULL
WorkflowActorList ::= WorkflowActorDeclaration | λ
```

شکل ۷- تعریف کنش‌گر

```
OrganizationChartDefinition ::=
    ORGANIZATIONCHART {
        OrganizationDepartmentItem
        OrganizationDepartmentList } | λ
OrganizationDepartmentItem ::=
    < DepartmentName, DepartmentFatherName >
OrganizationDepartmentList ::=
    OrganizationDepartmentItem
    OrganizationDepartmentList | λ
DepartmentFatherName ::=
    DepartmentName | NULL
```

شکل ۸- تعریف نمودار سازمانی

```
DepartmentHasRole ::=
    DEPARTMENTROLE {
        DepartmentHasRoleItem
        DepartmentHasRoleList } | λ
DepartmentHasRoleItem ::=
    <DepartmentName, RoleName Manager>
Manager ::= , MANAGER
DepartmentHasRoleList ::=
    DepartmentHasRoleItem DepartmentHasRoleList | λ
```

شکل ۹- تعریف نقش‌های سازمانی

۵- طراحی زبان جهت مدل‌سازی دامنه گردش کار

زبان مدل‌سازی کاربرد مبتنی بر نتایج گام تحلیل مشترکات و تمایزات طراحی شده و در صورت لزوم اصلاحاتی نیز روی آن انجام می‌شود. با توجه به دامنه سامانه‌های حاوی گردش کار و نتایج تحلیل، زبان مذکور دارای اجزاء و قواعد زیر است.

• تعریف مدل سازمان

هر سازمان دارای یک نام است و از چندین بخش تشکیل شده است که در آن بخش‌ها کارمندی به ایفای نقش می‌پردازند که هر یک از آن‌ها نیز شرح وظایفی دارند. با انجام کارها و وظایف توسط کارمندان هدف سازمان محقق می‌گردد. بنابراین در این میان انتقال اسناد، اطلاعات بین کارمندان از مهمترین مباحث گردش کار در سازمان است. البته این انتقالات باید در قالب فرآیندهای سازمانی و ارضای شرایط خاصی صورت پذیرد تا در زمان مقتضی اطلاعات لازم در دسترس فرد مجاز و مطلوب قرار گیرد. از این رو در تعریف مدل سازمانی اقلام زیر باید تعریف گردد.

- (۱) نام سازمان
- (۲) تعریف کنش‌گرها و موارد مرتبط با آن‌ها
- (۳) تعریف نرم‌افزارهای سازمان
- (۴) تعریف داده‌های مرتبط
- (۵) تعریف موجودیت‌های سازمانی
- (۶) تعریف توابع لازم برای انجام کارها
- (۷) تعریف موارد لازم برای ثبت عملکردها
- (۸) تعریف گزارشات متنوع
- (۹) تعریف پرده‌ها، فعالیت‌های آن‌ها و انتقالات بین فعالیت‌ها

```
OrganizationModel ::= < ORGANIZATIONNAME >
WorkflowActorSpecification OrganizationChartDefinition
DepartmentHasRole GroupHasMember
HumanHasRole WorkflowApplicationDefinition
WorkflowRelevantData ObjectDefinition
LibraryDeclaration LoginDefinition
ReportDefinition WorkflowProcessDefinition
```

شکل ۵- تعریف مدل سازمان

• تعریف کنش‌گرها

بطور کلی کنش‌گرها به پنج گروه تقسیم می‌شوند که عبارتند از بخش سازمانی، نقش سازمانی، کارمند، سیستم و گروه. به دلیل اینکه در سازمان‌های متفاوت، کنش‌گرها دارای خصوصیتی هستند که منحصر به آن سازمان است، لذا ابتدا باید خصوصیات کنش‌گرها بطور کلی مشخص گردد (شکل ۶). سپس همه کنش‌گرهای موجود در سازمان با توجه به خصوصیات مطلوب که قبلاً تعریف گردید، توسط قواعد موجود در شکل ۷ مشخص شوند.

ذکر این نکته حائز اهمیت است که ممکن است در میان پرده‌های سازمان لازم باشد تا جزیی از کار توسط یک نرم‌افزار خاص یا یک سیستم موجود انجام شود، بنابراین باید این سیستم به عنوان یک کنش‌گر معرفی گردد. به همین علت یک از انواع کنش‌گرها، سیستم در نظر گرفته شده است. در این میان کنش‌گر گروه نیز فقط به عنوان یک دسته‌بندی کننده برای سایر کنش‌گرهاست و هویت

```

ExtendedAttributes ::=
    EXTENDEDATTRIBUTES {
        ExtendedAttributeItem
        ExtendedAttributeList } | λ
ExtendedAttributeItem ::=
    < AttributeName , AttributeType ,
        AttributeValue,AttributeDescription >
ExtendedAttributeList ::=
    ExtendedAttributeItem ExtendedAttributeList | λ
    
```

شکل ۱۳- تعریف صفات خاص

- تعریف پردازها و فعالیت‌ها

مهمترین موجودیت در دستورزبان پردازها است. پردازها بیان کننده روال‌های کاری موجود در سازمان می‌باشند که با تعریف و اجرای آن‌ها می‌توان تمام اهداف سازمانی را محقق نمود. به عبارت دیگر، به ازای هر روال موجود در سازمان، پردازهای در دستورزبان ایجاد می‌گردد که در تعریف آن باید اطلاعات متفاوتی مشخص شود. البته با توجه به این‌که هر پردازها از تعدادی فعالیت تشکیل شده است، لذا باید فعالیت‌ها و تمام اطلاعات مربوط به آن‌ها نیز تعریف گردد. بطور کلی فعالیت‌ها به دو دسته فعالیت‌های غیرزائد و فعالیت‌های زائد تقسیم می‌شوند. فعالیت‌های زائد فعالیت‌هایی هستند که برای اتخاذ تصمیم‌های مربوط به شیوه گردش کار از جمله ایجاد حلقه، کنترل‌گر انتقالات ورودی یا انتقالات خروجی استفاده می‌شوند و این در حالی است که فعالیت‌های غیرزائد همان کارهای واقعی می‌باشند که برای تحقق گردش کار توسط کنش‌گرها صورت می‌پذیرند. قواعد مربوط به این قسمت در شکل ۱۴ نشان داده شده است.

- تعریف انتقالات

برای تحقق پردازها، باید فعالیت‌ها به ترتیب تعریف‌شده‌ای اجرا گردند. در این میان بعد از تکمیل یک فعالیت، فعالیت بعدی اجرا می‌گردد. البته این‌که فعالیت بعدی کدام فعالیت است و با حصول چه شرایطی باید اجرا شود، توسط گرافی جهت‌دار مشخص می‌گردد که در این دستور زبان گراف مذکور با استفاده از قواعد تعبیه شده قابل توصیف است که این توصیفات به نام انتقالات شناخته می‌شوند. با مراجعه به شکل ۱۵ می‌توان دستورات لازم را مشاهده نمود.

- تعریف عبارت

برای بیان شروط در تعریف انتقالات باید بتوان عبارات منطقی مطلوب را تعریف نمود که در بعضی از موارد لازم است تا با استفاده از عبارتهای حسابی، تاریخی یا رشته‌ای، عبارتهای منطقی ترکیبی ایجاد نمود که در شکل ۱۶ این قواعد نمایش داده شده‌اند.

- تعریف داده‌های مرتبط

برای متغیرهایی که در تعریف عبارات استفاده می‌شوند و در زمان اجرا مقادیر آنان مشخص می‌گردد، از دستورات مربوط به تعریف داده مرتبط استفاده می‌شود که در این میان با ذکر نام، شرح، نوع، طول و مقدار اولیه می‌توان به تعریف این متغیرها مبادرت نمود (شکل ۱۷).

- تعریف انواع داده‌ای ساده و مرکب

با توجه به اینکه برای تعریف داده‌های مرتبط و تعریف پارامترهای توابع احتیاج به ذکر نوع آنان است. لذا قواعد موجود در شکل ۱۸ برای این امر تعبیه گردیده است. همانطور که در شکل مذکور دیده می‌شود، پنج نوع داده‌ای ساده و نیز سه نوع داده‌ای ترکیبی که شامل رکورد، آرایه و مجموعه می‌باشد در نظر گرفته شده است.

```

HumanHasRole ::=
    HUMANROLE {
        HumanHasRoleItem
        HumanHasRoleList } | λ
HumanHasRoleItem ::= < HumanName , RoleName >
HumanHasRoleList ::=
    HumanHasRoleItem
    HumanHasRoleList | λ
    
```

شکل ۱۰- تعریف تناظر فرد به نقش

```

GroupHasMember ::=
    GROUPMEMBERS {
        GroupHasMemberItem
        GroupHasMemberList } | λ
GroupHasMemberItem ::=
    < GroupName , MemberItem MemberList >
MemberItem ::= ( MemberName , MemberType )
MemberList ::= MemberItem MemberList | λ
MemberType ::=
    HUMAN | ROLE | ACTORSYSTEM |
    ORGANIZATIONDEPARTMENT
GroupHasMemberList ::=
    GroupHasMemberItem
    GroupHasMemberList | λ
    
```

شکل ۱۱- تعریف اعضای گروه

```

WorkflowApplicationDefinition ::=
    WORKFLOWAPPLICATION {
        WorkflowApplicationItem
        WorkflowApplicationList } | λ
WorkflowApplicationItem ::=
    < ApplicationName , ApplicationDescription ,
        ToolName , FormalParameters ,
        ExtendedAttributes >
WorkflowApplicationList ::=
    WorkflowApplicationItem
    WorkflowApplicationList | λ
    
```

شکل ۱۲- تعریف نرم‌افزارهای فراخوانی‌شونده

- تعریف نرم‌افزارهای فراخوانی‌شونده

در پاره‌ای از موارد برای انجام اعمال، احتیاج به فراخوانی یک یا چند نرم‌افزار می‌باشد. ولی با توجه به این‌که برای فراخوانی، نرم‌افزارهای مذکور باید به موتور گردش کار شناسانده شوند تا وی در هنگام نیاز از آن‌ها استفاده نماید، بنابراین با استفاده از قواعد موجود در زبان که برای این کار تعبیه شده‌اند، نرم‌افزارهای مورد نظر به همراه نوع ورودی‌ها و خروجی‌ها به سیستم موتور گردش کار معرفی می‌گردند. سپس موتور گردش کار در موقع نیاز با ایجاد پارامترهای لازم برای ورود و خروج اطلاعات فراخوانی را بطور کامل انجام می‌دهد که قواعد مربوط به این مطلب را می‌توان در شکل ۱۲ مشاهده نمود.

- تعریف صفات خاص

در سازمان‌های متفاوت بسته به نیاز، احتیاج به تعریف صفاتی خاص است که در هنگام طراحی زبان بطور مشخص نمی‌توان آن‌ها را در نظر گرفت، ولی باید قابلیت به زبان اضافه نمود که هر سازمان بتواند صفات خاص کاری مورد نظر خویش را تعریف نماید. بنابراین قسمتی از قواعد شکل ۱۳ به این امر اختصاص داده شده است.

سایر موجودیت‌های الکترونیکی قابل چرخش، تخصیص کارها به همکاران و سایر موارد مربوط را دارا باشد. بنابراین برای ایجاد انعطاف‌پذیری در تولید اعضای دامنه مذکور، به کاربر اجازه داده شده است تا بتواند توابعی تعریف نماید و با استفاده از APIهایی که تعبیه شده است، بتواند نیازهای خود را مرتفع نماید. از این رو کاربر می‌تواند با قواعد شکل ۱۹ به این امر مبادرت ورزد.

- تعریف پارامترها

در تعریف توابع لازم است، پارامترهای ورودی و خروجی توابع با جزئیات کافی بطور مثال نوع، تعداد و مقدار پیش‌فرض مشخص شوند که با استفاده از قواعد تعبیه شده در شکل ۲۰ می‌توان تمامی پارامترهای ورودی یا خروجی توابع را بصورت جزئی تعریف نمود.

- تعریف شی

در سیستم‌های گردش کار اغلب موجودیتی در حال گردش است که این موجودیت می‌تواند سند، شرح یک دستور یا هر موجودیت الکترونیکی دیگری باشد. برای این که بتوان بطور مناسب با این پدیده برخورد کرد، عنصری به نام شی، مسئولیت تعریف موجودیت گردش کننده را برعهده دارد و با تعریف آن می‌توان به موتور گردش کار شی مورد نظر خود را شناساند. لذا با استفاده از دستورات شکل ۲۱ می‌توان عناصر مورد نظر را تعریف نمود.

```

LogicalExpression ::=
  ( LogicalExpression BooleanOperator LogicalExpression )
  | ( StringExpression BOperator StringExpression )
  | ( DateExpression BOperator DateExpression )
  | ( ArithmeticExpression BOperator ArithmeticExpression )
  | ( SetMemberName IN SetName )
  | UnaryBooleanOperator ( LogicalExpression )
  | ( BooleanConstant ) | ( DataAccess )
ArithmeticExpression ::=
  ( ArithmeticExpression AOperator ArithmeticExpression )
  | ( DateExpression - DateExpression )
  | - ArithmeticExpression
  | NumericConstant | DataAccess
DateExpression ::=
  ( DateExpression + DateExpression )
  | ArithmeticExpression + DateExpression | DateConstant
  | DataAccess
StringExpression ::=
  StringExpression + StringExpression | StringConstant
  | DataAccess
DataAccess ::=
  FunctionAccess | RecordName.RecordMemberName
  | ArrayName [ ArrayIndex ]
FunctionAccess ::= ( FNCACS : FunctionName )
BooleanOperator ::= && || EqualityOperator
BOperator ::= < | > | <= | >= | EqualityOperator
EqualityOperator ::= = | !=
AOperator ::= + | - | * | / | MOD | DIV | ^
BooleanConstant ::= TRUE | FALSE

```

شکل ۱۶- تعریف عبارات، شیوه دسترسی به داده‌ها و عملگرها

- ثبت وقایع

جهت رفع مطلوب مشکلات احتمالی در هنگام بروز خطا، لازم است که اطلاعات مربوط به نام کنش‌گرها، نوع آن‌ها، زمان انجام کار، نام پردازش، نام فعالیت، تاریخ یا یکسری دیگر از صفات که به دل‌خواه کاربر از قبل تعریف شده است نگهداری گردد و برای ردیابی عوامل خطا و ساده‌تر نمودن پردازش نگهداری از آن بهره‌بردار می‌شود (شکل ۲۲).

```

WorkflowProcessDefinition ::=
  WorkflowProcessItem WorkflowProcessList
WorkflowProcessItem ::=
  PROCESS{
    ProcessInformation, ProcessActivityList,
    TransitionDefinition,
    ProcessTransitionList }
ProcessInformation ::=
  < Name , Description ,
  ProcessResponsibleName , Priority ,
  Duration (DurationUnit) , FromDate ,
  ToDate , Documentation , Icon >
DurationUnit ::= YEAR | MONTH | DAY | HOUR | MINUTE
  | SECOND
ProcessActivityList ::=
  ACTIVITIES { ActivityItem ActivityList }
ActivityItem ::=
  <Name,Description,ActivityPerformerName,
  ObjectName , Duration (DurationUnit) ,
  FromDate , ToDate, Kind StartMode
  FinishMode Priority Icon Documentation
  ExtendedAttributes >
Kind ::= Dummy | NoDummy
Dummy ::= LoopControler | TransitionPolicy
LoopControler ::= , LOOP ( LoopKind < LoopCondition > ) | λ
LoopKind ::= REPEATUNTIL | WHILE
LoopCondition ::= LogicalExpression
TransitionPolicy ::= InputKind OutputKind
InputKind ::= , INPUT ( ActionOnInputTransition )
ActionOnInputTransition ::= ANDJOIN | XORJOIN
OutputKind ::= , OUTPUT ( ActionOnOutputTransition )
ActionOnOutputTransition ::= ANDSPLIT | XORSPLIT
  TransitionNameList
TransitionNameList ::= { TransitionName
  OtherTransitionsName } | λ
OtherTransitionName ::= , TransitionName
  OtherTransitionsName | λ
StartMode ::= AUTOMATIC | MANUAL
FinishMode ::= AUTOMATIC | MANUAL
NoDummy ::= APPLICATION ( ApplicationName ) |
  SUBFLOW( ProceName ) | FunctionAccess
  | REPORT ( ReportName ) | λ
WorkflowProcessList ::=
  WorkflowProcessItem WorkflowProcessList | λ

```

شکل ۱۴- تعریف پردازش و فعالیت

```

TransitionDefinition ::=
  TRANSITIONS { TransitionItem TransitionList } | λ
TransitionItem ::=
  < TransitionName , TransitionDescription ,
  TransitionKind ExtendedAttributes >
TransitionKind ::=
  FROM < ActivityName > TO < ActivityName >
  TransitionCondition
  | FROMLOOP < ActivityName > TO
  < ActivityName >
  | FROM < ActivityName > TOLOOP
  < ActivityName >
TransitionCondition ::= [ LogicalExpression ] | λ

```

شکل ۱۵- تعریف انتقالات

- تعریف توابع

با توجه به این که این زبان برای دامنه بزرگی از نرم‌افزارهای گردش کار طراحی شده است و باید بتواند قابلیت‌های متنوع و متفاوتی از قبیل چرخش انواع سند یا

```

LoggingDefinition ::=
  LOG { LogAttributeNameItem LogAttributeNameList } |
  λ
LogAttributeNameItem ::=
  ACTORNAME | ACTORTYPE | TIME
  | PROCESSNAME | DATE
  | ACTIVITYNAME | RELDAPA <RelevantDataName>
  | OBJ<ObjectName>
  | AEAN<ActorExtendedAttributeName>
LogAttributeNameList ::=
  , LogAttributeNameItem LogAttributeNameList |
  λ
    
```

شکل ۲۲- تعریف Log

• گزارش‌ها

در بیشتر سازمان‌ها مدیران به وجود گزارش‌های متفاوت علاقه‌مند هستند و آنان یا دیگر همکاران سازمان با استفاده از گزارش‌ها می‌توانند عملکردها را از لحاظ کمی و کیفی ارزیابی نمایند تا برای برنامه‌ریزی آتی سازمان تصمیمات درستی اتخاذ نمایند. با استفاده از قواعد دستور زبان که در شکل ۲۳ نشان داده شده است می‌توان گزارش‌های متفاوت و متنوعی را ایجاد نمود.

```

ReportDefinition ::= ReportItem ReportList
ReportItem ::=
  REPORT ReportName { ReportAttributeItem
  ReportAttributeList } | λ
ReportAttributeItem ::=
  ACTORNAME | ACTORTYPE | TIME
  | PROCESSNAME | ACTIVITYNAME | DATE
  | RELDAPA <RelevantDataName>
  | OBJ <ObjectName>
  | AEAN <ActorExtendedAttributeName>
ReportAttributeList ::=
  , ReportAttributeItem ReportAttributeList | λ
ReportList ::= ReportItem ReportList | λ
    
```

شکل ۲۳- تعریف گزارش

۶- معماری مولد موتور گردش کار SWEG^۵

پس از طراحی زبان برای تولید موتور گردش کار، باید محیطی مناسب ایجاد نمود تا با استفاده از آن مهندس کاربرد بتواند موتور دل‌خواه خود را توصیف و سپس آن را با استفاده از مترجم زبان، تولید نماید. به همین دلیل پیاده‌سازی محیط مورد ذکر از اهمیت ویژه‌ای برخوردار است. بنابراین معماری کلان نرم‌افزار سه لایه در نظر گرفته شده است که آن را می‌توان در شکل ۲۴ مشاهده نمود. معماری مورد نظر به دلایل متفاوتی از قبیل استقلال اجزاء، قابلیت اطمینان بالا، انعطاف‌پذیری و استقلال از سیستم مدیریت پایگاه‌داده‌ها بسیار مناسب می‌باشد. به طور کلی هر یک از لایه‌ها وظایف خاصی را عهده‌دار هستند که به شرح زیر است:

• لایه مدیریت کلان و واسط گرافیکی کاربر

یکی از اعمال مهمی که این لایه انجام می‌دهد، مدیریت کلان سایر لایه‌ها و اجزای آن‌ها است. البته وظایف دیگری را نیز بر عهده دارد که از آن جمله می‌توان به فراخوانی جزء مترجم و جزء سازنده موتور گردش کار جهت اجرا و نیز نظارت بر صحت عملکرد آن دو جزء اشاره نمود. همچنین این لایه وظیفه تعامل با کاربر، فراهم نمودن ابزارهای مدل‌سازی گرافیکی مناسب، چک کردن صحت منطق مدل‌سازی، تولید مدل متنی و ذخیره‌سازی و بارگذاری مجدد را انجام می‌دهد.

```

WorkflowRelevantData ::=
  WORKFLOWRELEVANTDATA {
  RelevantDataItem RelevantDataList } | λ
RelevantDataItem ::= < RelevantDataName ,
  RelevantDataDescription ,
  RelevantDataType,
  RelevantDataLength , DefaultValue >
RelevantDataType ::= ComplexDataTypeDefinition
RelevantDataList ::= RelevantDataItem RelevantDataList | λ
    
```

شکل ۱۷- تعریف داده‌های مرتبط

```

BasicDataTypeDefinition ::=
  STRING | FLOAT | REFERENCE | DATE | BOOLEAN
ComplexDataTypeDefinition ::=
  BasicDataTypeDefinition
  | RECORD RecordName { RecordMemberItem
  RecordMemberList }
  | ARRAY ArrayName ( BeginNumber .. EndNumber )
  : BasicDataTypeDefinition
  | SET SetName Of { DataConstantItem
  DataConstantList }
RecordMemberItem ::=
  RecordMemberName : RecordMemberType
RecordMemberType ::= BasicDataTypeDefinition
RecordMemberList ::=
  , RecordMemberItem RecordMemberList | λ
DataConstantItem ::= Constant
DataConstantList ::= DataConstantItem DataConstantList | λ
    
```

شکل ۱۸- تعریف انواع داده‌ای ساده و ترکیبی

```

LibraryDeclaration ::= LIBRARY { Libraries } | λ
Libraries ::= LibraryItem LibraryList
LibraryList ::= , LibraryItem LibraryList | λ
LibraryItem ::= FunctionItem
FunctionItem ::=
  FUNCTION < FunctionName , FunctionDescription ,
  ResultType , ContentFile , FormalParameters
  , ExtendedAttributes >
ResultType ::= BasicDataTypeDefinition
    
```

شکل ۱۹- تعریف توابع

```

FormalParameters ::=
  FORMALPARAMETER { Parameters } | λ
Parameters ::= InputParameters OutputParameters
InputParameters ::=
  INPUTPARAMETERS ( ParameterItem ParameterList ) | λ
OutputParameters ::=
  OUTPUTPARAMETERS ( ParameterItem ParameterList ) | λ
ParameterItem ::= RelevantDataName j ObjectName
ParameterList ::= , ParameterItem ParameterList | λ
    
```

شکل ۲۰- تعریف پارامترها

```

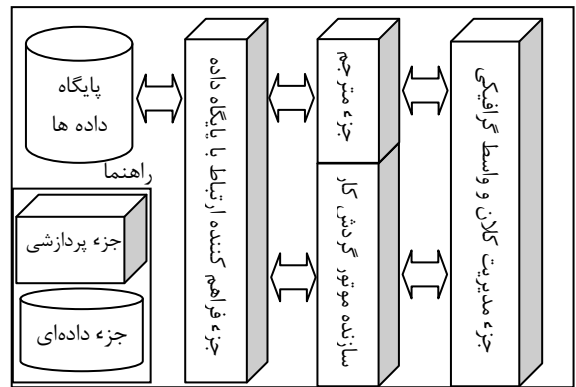
ObjectDefinition ::= OBJECTS { ObjectItem ObjectList } | λ
ObjectItem ::=
  < ObjectName , ObjectDescription ExtendedAttribute >
ObjectList ::= ObjectItem ObjectList | λ
    
```

شکل ۲۱- تعریف شیء

دیگر برای انجام اعمال مورد نظر خود از این اعمال استفاده می‌کند. به همین دلیل در سایر اجزا هیچ نمونه‌ای از توابعی که مربوط به سیستم مدیریت پایگاه داده‌های خاصی باشد، دیده نمی‌شود.

۷- بررسی مثالی از گردش کار

در این مثال شرکتی مورد بررسی قرار می‌گیرد که الگوهای رفتاری And-join و And-split در آن استفاده شده است. در این شرکت تجاری پنج نفر کار می‌کنند که نقش‌های مدیرعامل، مدیرمالی و کارشناس را بر عهده دارند. از آن پنج نفر، یک نفر مدیرعامل، یک نفر مدیرمالی و سه نفر کارشناس هستند. در ابتدا سازمان مذکور توسط دستورزبان توصیف می‌شود که این دستورات در شکل ۲۶ قابل مشاهده هستند. فرض کنید در این شرکت پروژه تجاری زیر وجود دارد.

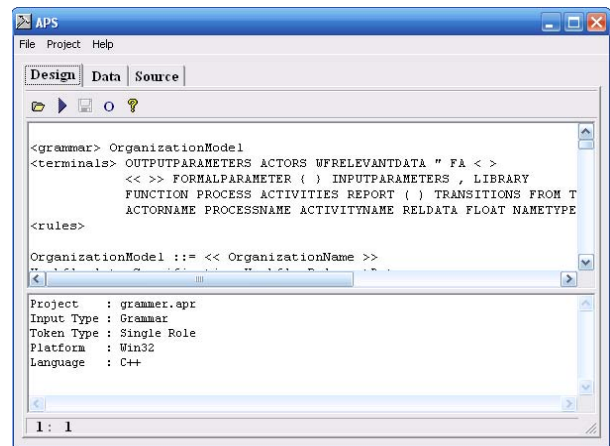


شکل ۲۴- معماری کلان نرم‌افزار تولیدکننده موتور گردش کار

• جزء مترجم از لایه دوم

این جزء یکی از مهمترین اجزاء این معماری است که وظیفه ترجمه مدل متنی را برعهده دارد و اطلاعات لازم و مفید را از مدل متنی بیرون کشیده، با استفاده از لایه سوم در پایگاه داده‌ها ذخیره می‌نماید تا در هنگام تولید موتور گردش کار از آن استفاده گردد. برای تولید مترجمی که بتواند زبان طراحی شده در بخش قبلی را ترجمه نماید، می‌توان از مترجم‌سازهایی مانند APS، PreCC، یا Yacc استفاده نمود.

در این معماری برای تولید مترجم از APS استفاده شده است. این نرم‌افزار می‌تواند با استفاده از دستورزبان یا گراف نحوی زبان مورد نظر، مترجم آن را ایجاد نماید. در این میان مشخص کردن نحوه عملکرد روال‌های مفهومی بر عهده طراح زبان خواهد بود. در شکل ۲۵ نمایی از APS نشان داده شده است.



شکل ۲۵- نمایی از نرم‌افزار APS

• جزء سازنده موتور گردش کار از لایه دوم

پس از اتمام کار جزء مترجم، این بخش با استفاده از اطلاعات موجود در پایگاه داده‌ها، موتور گردش کار مناسبی که منطبق با نیازهای کاربر و مدل منطقی ترسیم شده وی باشد، ایجاد می‌نماید. این قسمت موتوری ایجاد می‌نماید که با موتور گردش کاری که در بخش‌های قبلی بررسی شد، سازگار باشد. بنابراین موتور تولید شده دارای اجزایی مانند جزء اجراکننده نرم‌افزارها، جزء فراهم‌کننده توابع برنامه‌نویسی، جزء ردیاب فعالیت‌ها و جزء شروع‌کننده پروژه است.

• جزء فراهم‌کننده ارتباط با پایگاه داده‌ها

این جزء تمام کارهای مربوط به ذخیره‌سازی، روزرسانی و بازیابی اطلاعات را برعهده دارد. در این لایه توابع اساسی کار با پایگاه داده‌ها وجود دارد و هر جزء

```
< Corporation >
ACTORS {
  < DirectManager, "Main Role Of Corporation", ROLE,
    DirectMan, ManPass >
  < FinancialManager, "Financial Manager", ROLE,
    FinanMan, FiPass >
  < FirstEngineer, "Computer Engineer ", ROLE, FEng,
    Eng1Pass >
  < SecondEngineer, "Computer Engineer ", ROLE, SEng,
    Eng2Pass >
  < ThirdEngineer, "Computer Engineer ", ROLE, TEng,
    Eng3Pass >
  < EngineerGroup, "Group Of Engineers ", GROUP, -, - >
  < Word, "Software For Creating Document ",
    ACTORSYSTEM, None, None >
}
GROUPMEMBERS {
  <EngineerGroup, (FEng, ROLE)> <EngineerGroup,
  (SEng, ROLE)> <EngineerGroup, (TEng, ROLE)>
}
WORKFLOWAPPLICATION {
  < MsWord, "MicrosoftWord ", msword.exe,
  FORMALPARAMETER { OUTPUTPARAMETER (Path)} >
}
WORKFLOWRELEVANTDATA {
  < Result, "Result", BOOLEAN, 1, TRUE >
  < Path, "Path Of Request File", STRING, 500, "" >
  < AppName, "Word Application File Name", STRING, 10,
  "MsWord.exe" >
}
LIBRARY {
  FUNCTION < GivingIdea, "This Function Show Request
  And Giving Result And Set Result Relevant Data",
  BOOLEAN, "func.dll",
  FORMALPARAMETER {
    INPUTPARAMETERS ( AppName , Path )
    OUTPUTPARAMETER (Result)
  }
}
```

شکل ۲۶- تعریف مدل سازمان

۷-۱- پروژه در خواست مساعده

برای این که کارشناس مساعده دریافت کند، ابتدا باید فرم مربوط به تقاضای مساعده را پر نماید. سپس فرم مذکور را به مدیرعامل و مدیرمالی ارسال کند. مدیرعامل و مدیرمالی درخواست مذکور را بررسی می‌نمایند و نظر خود را نیز اعلام

قابلیت حمل و تولید موتورهای گردش کار توزیع شده، ایجاد عامل‌ها در انجام فعالیت‌ها [۱۶] از موارد مهمی هستند که باید روی آن کار شود.

مراجع

[1] K. C. Kang, V. Sugumaran, and S. Park, *Applied software product-line engineering*, Auerbach Publications, 2009.

[2] C. Kim, H. S. Chung, E. S. Cho, *Micro and macro workflow variability design techniques of component*, Information and Software Technology, In Press, Corrected Proof, Available online 31 January 2007.

[3] Biglever Software Inc., *Software Product Lines*, <http://www.softwareproductlines.com>, 2004.

[4] D. M. Weiss, and C. R. Lai, *Software Product-Line Engineering: A Family-Based Software Development Process*, Addison Wesley Professional, 1999.

[5] M. Harsu, *FAST product-line architecture process*, Software Systems Laboratory, Tampere University of Technology, Technical Reports, pp.45, 2001.

[6] M. Wang, H. Wang, and D. Xu, "The design of intelligent workflow monitoring with agent technology," *Knowledge-Based Systems*, vol. 18, no. 6, pp. 257-266, 2005.

[7] W. Tan and Y. Fan, "Dynamic workflow model fragmentation for distributed execution," *Computers in Industry*, vol. 58, no. 5, pp. 381-391, 2007.

[8] J. Jung, H. Kim, and S. Kang, "Standards-based approaches to B2B workflow integration," *Computers and Industrial Engineering*, vol. 51, no. 2, pp. 321-334, 2006.

[9] M. D. R-Moreno, D. Borrajo, A. Cesta, and A. Oddi, "Integrating planning and scheduling in workflow domains, Expert Systems with Applications," vol. 33, no. 2, pp. 389-406, 2007.

[10] D. Hollingsworth, *The Workflow Reference Model - Issue 1.1*, Technical Report Document Number TC00-1003, Workflow Management Coalition, 1995.

[11] Workflow Management Coalition, *Workflow Management Coalition Terminology & Glossary*, Technical Report Document Number WPMC-TC-1011, Workflow Management Coalition, 1999.

[12] S. Li and D. Coleman, "Modeling distributed GIS data production workflow, Computers," *Environment and Urban Systems*, vol. 29, no. 4, pp. 401-424, 2005.

[13] R. A. Botha and J. H. P. Ellof, "A security interpretation of the workflow reference model," *Information Security - from Small Systems to Management of Secure Infrastructures*, pp. 43-51, 1998.

[14] D. G. Cholewka, R. A. Botha, and J. H. P. Ellof, "A Context-sensitive Access Control Model and Prototype

می‌کنند. در صورتی که هر دو موافق باشند، مساعده مذکور پرداخت خواهد شد ولی اگر یکی از آن‌ها با درخواست مخالفت نماید، به کارشناس هیچ مبلغی پرداخت نخواهد شد. همچنین موافقت یا عدم موافقت نیز به اطلاع وی خواهد رسید. در این پردازش دو الگوی And-split و And-join مشاهده می‌شود که دستورات لازم برای تعریف پردازش مورد نظر در شکل ۲۷ ذکر گردیده است.

```

PROCESS {
  < FinancialHelp, "Financial Help Description",
    DirectMan >
  ACTIVITIES {
    < FillingRequest, "None", EngineerGroup,
      APPLICATION(MsWord, Path), MANUAL,
      MANUAL >
    < FinMIdea, "Getting Financial Manager Idea",
      Financial Manager, FUNCTION ( GivingIdea ),
      MANUAL, MANUAL >
    < DirMIdea, "Getting Direct Manager Idea",
      Direct Manager, FUNCTION ( GivingIdea ),
      MANUAL, MANUAL >
    < NegativeResult, "Your Request Not Accepted",
      EngineerGroup, MANUAL, MANUAL >
    < ToFMDM, "Sendig To F. and D. Managers",
      OUTPUT ( ANDSPLIT ) >
    < FromFMDM, "Both Managers Accepted", INPUT
      ( ANDJOIN ) >
    < PositiveResult, "Your Request Accepted",
      EngineerGroup, MANUAL, MANUAL >
  }
  TRANSITIONS {
    < SendToManagers0, "Sending to DommyActivity",
      FROM < FillingRequest > TO < ToFMDM > >
    < SendingToManager1, " Sending From Dummy
      Activity To Direct Manager for Giving
      Idea",FROM < ToFMDM > TO < DirMIdea >>
    < SendingToManager2, " Sending From Dummy
      Activity To Financial Manager for Giving
      Idea",FROM < ToFMDM > TO < FinMIdea >>
    < IdeaResult0, "Direct Managers Not Accepted",
      FROM < DirMIdea > TO < NegativeResult >
      [ Result == FALSE ] >
    < IdeaResult1,"FinancialManagers Not
      Accepted",FROM<FinMIdea>
      TO < NegativeResult > [ Result == FALSE ] >
    < IdeaResult2, "Direct Managers Accepted",
      FROM < DirMIdea >
      TO < FromFMDM > [ Result == TRUE ] >
    < IdeaResult3, "Financial Managers Accepted",
      FROM < FinMIdea >
      TO < FromFMDM > [ Result == TRUE ] >
    < PosResult,"Sending Positive Answer ",
      From<FromFMDM > TO < PositiveResult >>
  }
}
    
```

شکل ۲۷- تعریف پردازش درخواست مساعده

۸- نتیجه گیری

با توجه به گسترش روزافزون استفاده از سامانه‌های گردش کار در سازمان‌ها، جهت ارتقاء سطح کمی و کیفی در کارها، در این مقاله به بررسی روش تولید نرم‌افزار مبتنی بر خانواده، موتور گردش کار استاندارد و استفاده از روش مذکور جهت ایجاد موتورهای گردش کار پرداخته شد و در نهایت معماری کلان سامانه‌ای که بتواند موتور گردش کار دل‌خواه را ایجاد نماید مطرح گردید. ولی در این میان پیاده‌سازی کامل سامانه، مبتنی بر وب بودن آن، ایجاد مدل‌های مبتنی بر XML برای ایجاد

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۸/۳/۲۵

تاریخ اصلاح: ۹۰/۵/۳

تاریخ قبول شدن: ۹۰/۵/۱۶

نویسنده مرتبط: دکتر سیدحسین میریان حسین آبادی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران.

¹ Product Line Engineering

² Family Based Approach

³ Rational Unified Process

⁴ Structural System Analysis and Design Method

⁵ Sharif Workflow Engine Generator

⁶ Advanced Parsing System

implementation," *Proc. of the IFIP 15th Annual Working Conference on Information Security for Global Information Infrastructures*, pp. 341-350, 2000.

[15] J. Wainer, A. Kumar, and P. Barthelmeß, "DW-RBAC: A formal security model of delegation and revocation in workflow systems," *Information Systems*, vol. 32, no. 3, pp. 365-384, 2007.

[16] P. Senkul and I. H. Toroslu, "An architecture for workflow scheduling under resource allocation constraints," *Information Systems*, vol. 30, no. 5, pp. 399-422, 2005.

[17] W. M. P. van der Aalst and A. H. M. Ter Hofstede, "YAWL: Yet Another Workflow Language," *Information Systems*, vol. 30, no. 4, pp. 245-275, 2005.

[18] R. W. Weidel, *Workflow White Paper*, SoftBrands Manufacturing, 2002.

[19] C. Ouyang, M. Adams, and A. H. M. Ter Hofstede, "Yet Another Workflow Language: Concepts, Tool Support, and Application," *Handbook of Research on Business Process Modeling*, pp. 92-121, 2009.

[20] H. Lee and H. Suh, "Workflow structuring and reengineering method for design process," *Computers and Industrial Engineering*, vol. 51, no. 4, pp. 698-714, 2006.



مرتضی یوسف صنعتی مدرک کارشناسی و کارشناسی ارشد را از دانشگاه صنعتی شریف به ترتیب در سال‌های ۸۱ و ۸۳ در رشته مهندسی نرم‌افزار اخذ نمود. سپس از سال ۸۳ در دانشگاه بوعلی سینا در گروه کامپیوتر مشغول به تدریس گردید. همچنین ایشان از سال ۸۹ در مقطع دکترا در دانشگاه مک‌مستر در کشور کانادا مشغول به تحصیل است و زمینه تحقیقاتی ایشان مدیریت جریان گردش کار در پزشکی می‌باشد. آدرس پست‌الکترونیکی ایشان عبارت است از:

mysanati@basu.ac.ir



سید حسن میریان حسین آبادی مدرک کارشناسی را از دانشگاه شهید بهشتی در سال ۶۳ و مدرک کارشناسی ارشد را از دانشگاه صنعتی شریف در سال ۶۶ در رشته مهندسی نرم‌افزار اخذ نموده است. ایشان در سال ۷۱ در مقطع دکترا در دانشگاه اسکس انگلستان ادامه تحصیل داده و در سال ۷۶ پس از فراغت از تحصیل با اخذ مدرک دکترا در علوم کامپیوتر (روش‌های صوری) به عنوان استادیار در دانشکده مهندسی کامپیوتر دانشگاه صنعتی شریف مشغول به کار شده‌اند. زمینه‌های تحقیقاتی ایشان کاربرد روش‌های صوری در توصیف و تولید نرم‌افزار به ویژه با استفاده از تئوری انواع و ریاضیات ساختی، معیارها و اندازه‌گیری نرم‌افزار، معماری نرم‌افزارهای با قابلیت پیکربندی مجدد، توصیف و درستی‌یابی صوری معماری نرم‌افزار و پایگاه داده رابطه‌ای و اکس-ام-ال است.

آدرس پست‌الکترونیکی ایشان عبارت است از:

hmirian@sharif.edu

کاربردهای اتوماتای یادگیر سلولی در پردازش تصویر

احمدعلی آبین سیدحمید امیری حمید بیگی

دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران

چکیده

اتوماتای یادگیر سلولی به عنوان یک مدل برای سیستم‌هایی که از اجزاء بسیار ساده‌ای تشکیل شده‌اند مطرح شده است. در این مدل رفتار هر جزء براساس رفتار همسایگانش و نیز تجربیات گذشته‌اش تعیین می‌شود. اجزاء ساده تشکیل دهنده این مدل از طریق تعامل با یکدیگر می‌توانند رفتار پیچیده و قدرتمندی را در کاربردهای مختلف از خود نشان دهند. پردازش تصویر یکی از حوزه‌های مهم هوش مصنوعی است که در آن می‌توان از اتوماتای یادگیر سلولی در مراحل مختلف پردازش استفاده نمود. ویژگی سلولی بودن اتوماتای یادگیر سلولی می‌تواند در پردازش و انتشار اطلاعات همسایگی تصاویر مورد استفاده قرار گیرد. همچنین برای افزایش دقت و بهبود الگوریتم‌های پردازش تصویر می‌توان از خواص یادگیری و تکرار شونده‌ی اتوماتای یادگیر سلولی استفاده نمود. در این مقاله اتوماتای یادگیر سلولی با بررسی خواص و کاربردهای آن در زمینه‌های مختلف بررسی می‌شود. سپس کاربرد آن در دو زمینه آشکارسازی لبه و قطعه‌بندی تصویر که هر دو از مباحث بسیار مهم و پایه‌ای در امر پردازش تصویر هستند با معرفی دو الگوریتم جدید مورد بررسی قرار می‌گیرد. روش پیشنهادی برای آشکارسازی لبه یک روش وفقی برای آشکارسازی لبه‌های تصویر می‌باشد که بدون استفاده از روش‌های معمول پردازش تصویر و تنها به کمک ویژگی‌های آماری تصویر و استفاده از اتوماتای یادگیر سلولی لبه‌های تصویر را آشکار می‌سازد. روش قطعه‌بندی مبتنی بر اتوماتای یادگیر سلولی نیز با استفاده از اطلاعات همسایگی پیکسل‌ها به کمک یک اتوماتای یادگیر سلولی اقدام به انتشار اطلاعات یافت در طول تصویر نموده و در نهایت به یک تصویر قطعه‌بندی شده همگرا می‌شود.

کلمات کلیدی: اتوماتای یادگیر، اتوماتای یادگیر سلولی، پردازش تصویر، تشخیص لبه، قطعه‌بندی تصویر.

۱- مقدمه

همسایگی تعریف شده با یکدیگر در ارتباط هستند. هر سلول می‌تواند چند مقدار مختلف را به خود بگیرد. برای تعیین مقدار سلول در هر مرحله، از یک قانون محلی استفاده می‌شود که این قانون با توجه به مقدار یک سلول و همسایه‌های آن در مرحله کنونی، مقدار سلول را در مرحله بعد تعیین می‌کند. اتوماتای یادگیر سلولی^۱ از یک اتوماتای سلولی تشکیل شده است که هر سلول آن به یک یا چند اتوماتای یادگیر مجهز شده است و به عنوان یک مدل برای سیستم‌هایی که از اجزاء ساده‌ای تشکیل شده‌اند، مطرح شده است. در این مدل رفتار هر جزء بر اساس رفتار همسایگانش و نیز تجربیات گذشته‌اش تعیین و اصلاح می‌شود. اجزاء ساده تشکیل دهنده این مدل از طریق کنش و واکنش با یکدیگر می‌توانند رفتار پیچیده‌ای از خود نشان دهند. تاکنون اتوماتای یادگیر سلولی در کاربردهای بسیاری مانند پردازش تصویر [۱، ۱۰]، شبکه‌های سلولی سیار [۱۱]، مدلسازی شبکه‌های تجاری [۱۲] مورد استفاده قرار گرفته است.

اتوماتای یادگیر^۱ ماشینی است که می‌تواند تعدادی متناهی کنش را انجام دهد. هر کنش انتخاب شده، توسط یک محیط احتمالی ارزیابی می‌شود و نتیجه ارزیابی در قالب سیگنال پاداش یا جریمه به اتوماتا داده می‌شود و اتوماتا از این پاسخ در انتخاب کنش بعدی تاثیر می‌گیرد. هدف نهایی این است که اتوماتا یاد بگیرد تا از بین کنش‌های خود بهترین کنش را انتخاب نماید. حل مسائل NP کامل [۷]، تخصیص ظرفیت [۸] و شبکه‌های عصبی [۹] تنها چند نمونه از مسائلی هستند که در آنها از اتوماتای یادگیر برای حل مساله استفاده شده است. از سویی دیگر، اتوماتای سلولی^۲ به عنوان یک ابزار برای سنتز سیستم‌های پیچیده از چندین مولفه ساده تشکیل شده‌اند که هر کدام از این مولفه‌ها به یک واحد به نام سلول نگاشت می‌یابند. این سلول‌ها در یک شبکه منظم قرار گرفته و تحت یک رابطه

سلولی به همراه ساختار روش‌های پیشنهادی و نتایج شبیه‌سازی هر روش شرح داده می‌شود. در پایان هم مقاله با نتیجه‌گیری پایان می‌پذیرد.

۲- آتوماتای یادگیر سلولی

در این بخش نخست بطور خلاصه آتوماتای سلولی معرفی می‌شود و سپس مقدمه‌ای خواهیم داشت بر آتوماتای یادگیر و چند روش متداول یادگیری و در پایان آتوماتای یادگیر سلولی معرفی و عملکرد آن شرح داده می‌شود.

۲-۱- آتوماتای سلولی

آتوماتای سلولی در سال ۱۹۴۰ توسط دو ریاضیدان به نام‌های John Von Neumann و Stanislaw Ulam معرفی شده است [۲۲]. آتوماتای سلولی به عنوان یک ابزار برای سنتز سیستم‌های پیچیده مناسب است. این سیستم‌ها از چندین مولفه ساده تشکیل شده‌اند که هر کدام از این مولفه‌ها به یک واحد به نام سلول نگاشت می‌یابند [۲۳، ۲۴]. این سلول‌ها در یک شبکه منظم قرار می‌گیرند و تحت یک رابطه همسایگی تعریف شده با یکدیگر در ارتباط هستند. هر سلول می‌تواند k مقدار مختلف را به خود بگیرد و برای تعیین مقدار سلول در هر مرحله، از یک قانون محلی استفاده می‌شود. این قانون با توجه به مقدار یک سلول و همسایه‌های آن در مرحله کنونی، مقدار سلول را در مرحله بعد تعیین می‌کند.

۲-۲- آتوماتای یادگیر

آتوماتای یادگیر به عنوان یک روش برای حل مسایل یادگیری توسط ریاضیدان روسی به نام Tsetlin معرفی شد [۲۵]. آتوماتای یادگیر ماشینی است که می‌تواند تعدادی متناهی کنش را انجام دهد. هرکنش انتخاب شده، توسط یک محیط احتمالی ارزیابی می‌شود و نتیجه ارزیابی در قالب سیگنال پاداش یا جریمه به آتوماتا داده می‌شود و آتوماتا از این پاسخ در انتخاب کنش بعدی تاثیر می‌گیرد. هدف نهایی این است که آتوماتا یاد بگیرد تا از بین کنش‌های خود بهترین کنش را انتخاب کند. بهترین کنش، آن کنشی است که احتمال دریافت پاداش از محیط را بیشینه نماید [۲۶]. شکل ۱ چگونگی تعامل آتوماتای یادگیر با محیط را نشان می‌دهد.



شکل ۱- ارتباط بین آتوماتای یادگیر و محیط [۲۷]

محیط را می‌توان توسط سه تایی $E = \{\alpha, \beta, C\}$ نشان داد که در آن $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه ورودی‌های محیط و خروجی‌های آتوماتا، $\beta = \{0, 1\}$ مجموعه خروجی‌های محیط و ورودی‌های آتوماتا و $C = \{c_1, c_2, \dots, c_r\}$ مجموعه احتمال‌های جریمه می‌باشد. هر گاه β مجموعه دو عضوی باشد، محیط از نوع P می‌باشد. در چنین محیطی $\beta_0 = 1$ به عنوان

از طرفی، حوزه پردازش تصویر یکی از حوزه‌هایی است که در آن استفاده از آتوماتای یادگیر سلولی بسیار مورد توجه قرار گرفته است. بسیاری از الگوریتم‌های پردازش تصویر، برای دستیابی به اهداف موردنظرشان معیارهای خاصی را تعریف کرده و سپس بر اساس این معیارها و استفاده از اطلاعات همسایگی عمل می‌نمایند. هر کدام از دو جزء یاد شده (تعریف معیار و استفاده از اطلاعات همسایگی) می‌توانند به راحتی توسط یک آتوماتای یادگیر سلولی پیاده‌سازی شوند. برای استفاده از اطلاعات همسایگی پیکسل‌های واقع در تصویر می‌توان از رابطه همسایگی موجود در آتوماتای یادگیر سلولی بهره گرفت و برای تعریف معیار می‌توان یک قانون مناسب طراحی نمود. از طرفی خاصیت سلولی تصاویر و همسایگی بین پیکسل‌های آن زمینه استفاده از آتوماتای یادگیر سلولی را آسان‌تر نموده است. از جمله کاربردهایی که در آنها استفاده از آتوماتای یادگیر سلولی منجر به تولید نتایج مناسبی شده است می‌توان به بازیابی تصاویر [۲]، الگوگذاری تصاویر [۳]، قطعه‌بندی تصاویر خاکستری [۴]، تشخیص لبه [۵] و تشخیص رنگ پوست [۱۳] اشاره کرد. همچنین در زمینه‌هایی از قبیل شبیه‌سازی پدیده‌های اجتماعی مانند انتشار شایعه [۱۴]، خوشه‌بندی داده‌ها [۱۵] و جایابی مدارهای متراکم [۱۶] از آتوماتای یادگیر سلولی استفاده شده است.

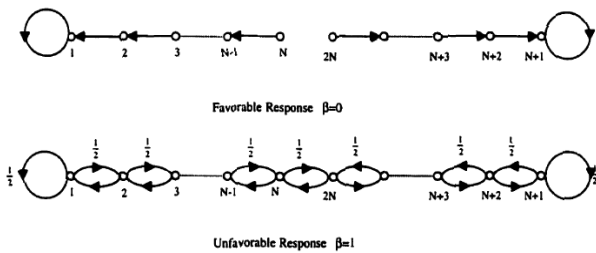
در این مقاله استفاده از آتوماتای یادگیر سلولی در پردازش تصویر در دو زمینه تشخیص لبه و قطعه‌بندی تصویر مورد بررسی قرار گرفته است و برای هر یک از آنها یک راهکار ارائه شده است. لبه‌های موجود در یک تصویر از ویژگی‌هایی است که در حوزه پردازش تصویر و بینایی ماشین بسیار پرکاربرد بوده و از مباحث پایه‌ای و بسیار مهم در این حوزه به شمار می‌رود. روش‌های زیادی از قبیل سوبل، پرویت، کنی و ... برای آشکارسازی این ویژگی مطرح شده‌اند که بیشتر آنها از خاصیت گرادیان تصویر و برخی عملیات ریاضی برای این امر استفاده می‌کنند.

در این مقاله یک آشکارساز و فقی لبه با استفاده از آتوماتای یادگیر سلولی ارائه شده است که در آن ابتدا با استفاده از تجزیه و تحلیل آماری تصویر و بررسی وجود لبه در نواحی مختلف تصویر، لبه‌های تصویر تقویت شده و نقاط غیر لبه تضعیف می‌شوند. این روش به عنوان یک مرحله پیش‌پردازش برای آتوماتای یادگیر سلولی می‌باشد. پس از آن از آتوماتای یادگیر سلولی برای تقویت و بهبود لبه‌های تقویت شده در مرحله پیش‌پردازش استفاده می‌شود. در این مرحله آتوماتای یادگیر سلولی به کمک خاصیت همسایگی و تکرارپذیری خود نوبزه‌های مرحله قبل را برطرف نموده و سبب بهبود لبه‌های آشکار شده خواهد شد.

قطعه‌بندی تصویر و ویدئو هم به عنوان یک گام پیش‌پردازش بسیار مهم برای اعمالی نظیر پیدا کردن ناحیه دلخواه در یک صحنه، تفسیر داده‌ها و بازیابی تصاویر محسوب می‌شود. تاکنون روش‌های زیادی از قبیل روش‌های مبتنی بر مدل‌سازی آماری [۱۶، ۱۷، ۱۸]، انتشار انرژی [۱۹]، افراز گراف [۲۰] و قطعه‌بندی بدون ناظر [۲۱] برای قطعه‌بندی تصاویر مطرح شده‌اند. در قسمت دوم این مقاله یک الگوریتم برای قطعه‌بندی تصاویر رنگی با استفاده از آتوماتای یادگیر سلولی معرفی می‌شود. در بسیاری از الگوریتم‌های قطعه‌بندی موجود، برای یافتن نواحی تصویر یک سری معیار تعریف و سپس بر اساس این معیارها و تشابه پیکسل‌های واقع در یک همسایگی، تعلق هر کدام از پیکسل‌ها به یک دسته تعیین می‌شود. هر کدام از دو جزء یاد شده (تعریف معیار و همسایگی) توسط یک آتوماتای یادگیر سلولی مدل‌سازی شده و در راستای قطعه‌بندی مورد استفاده قرار گرفته است. آتوماتای یادگیر سلولی از یک الگوریتم تکراری برای رسیدن به یک حالت پایدار در امر قطعه‌بندی استفاده می‌کند.

ادامه این مقاله بدین‌گونه سازماندهی شده است. در بخش ۲ آتوماتای یادگیر به همراه آتوماتای یادگیر سلولی مورد بررسی قرار می‌گیرد. در بخش ۳ و ۴ به ترتیب آشکارسازی لبه‌های تصویر و قطعه‌بندی تصاویر توسط آتوماتای یادگیر

آتوماتای Krylov نیز نوع دیگری از آتوماتا با ساختار ثابت است که گراف تغییر حالت آن در شکل ۴ نشان داده شده است. رفتار این نوع آتوماتا را به ازای پاداش و جریمه دریافتی از محیط، در شکل ۵ مشاهده می‌کنید. همانطوری که مشاهده می‌کنید رفتار این آتوماتا در محیط بدین صورت می‌باشد که به ازای دریافت پاداش در قبال اجرای یک کنش مانند دو آتوماتای قبلی عمل می‌کند ولی به ازای دریافت جریمه در قبال اجرای یک کنش، به احتمال ۰.۵ فرصت جریمه شدن آن را یک واحد افزایش می‌دهد و به احتمال ۰.۵ به اندازه یک واحد از فرصت جریمه آن کنش می‌کاهد. در کل این آتوماتا بدین صورت رفتار می‌کند که در حین اجرای هر کنش در محیط چنانچه یک پاداش دریافت کند فرض را بر این می‌گذارد که این کنش خوب است و فرصت جریمه آن را یک واحد افزایش می‌دهد ولی به ازای دریافت جریمه، به طور یقین فرض را بر این نمی‌گذارد که این کنش بد است بلکه با احتمال ۰.۵ این فرض را در نظر می‌گیرد.



شکل ۴- رفتار آتوماتای Krylov [۲۸]

۲-۲-۲- آتوماتای یادگیر با ساختار متغیر

آتوماتای یادگیر با ساختار متغیر توسط چهار تایی $E = \{\alpha, \beta, p, T\}$ نشان داده می‌شود که در آن $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه کنش‌های آتوماتا، $\beta = \{0, 1\}$ مجموعه ورودی‌های آتوماتا، $p = \{p_1, p_2, \dots, p_r\}$ بردار احتمال انتخاب هر یک از کنش‌ها و $T[\alpha(n), \beta(n), p(n)]$ الگوریتم یادگیری می‌باشد. فرض کنید که در مرحله n -ام کنش α_i انتخاب شود. این انتخاب بر اساس بردار احتمال کنش‌ها صورت گیرد. پس از اعمال کنش بر روی محیط تصادفی، پاسخ محیط دریافت می‌شود.

چنانچه محیط به کنش انتخاب شده پاداش دهد، احتمال انتخاب کنش α_i در مرحله بعد افزایش و احتمال انتخاب سایر کنش‌ها کاهش می‌یابند. اما اگر پاسخ محیط به این کنش جریمه باشد، احتمال انتخاب کنش α_i در مرحله بعد کاهش و احتمال انتخاب سایر کنش‌ها افزایش می‌یابند. نکته‌ای که باید مورد توجه قرار گیرد این است که p یک بردار احتمال است و بنابراین باید مجموع احتمال کنش‌ها در هر مرحله مساوی یک باشد. روابط (۱) و (۲) یکی از روش‌های به‌هنگام‌سازی بردار احتمال p (الگوریتم یادگیری) را نشان می‌دهد.

الف- پاداش ($\beta = 0$)

$$p_i(n+1) = p_i(n) + a[1 - p_i(n)]$$

$$p_j(n+1) = (1-a)p_j(n) \quad \forall j \neq i \quad (1)$$

ب- جریمه ($\beta = 1$)

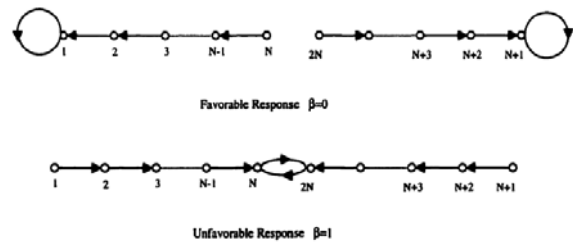
$$p_i(n+1) = (1-b)p_i(n)$$

$$p_j(n+1) = \frac{b}{r-1} + (1-b)p_j(n) \quad (2)$$

جریمه و $\beta_1 = 0$ به عنوان پاداش در نظر گرفته می‌شود. آتوماتاهای یادگیر به دو دسته کلی با ساختار ثابت و متغیر تقسیم می‌شوند.

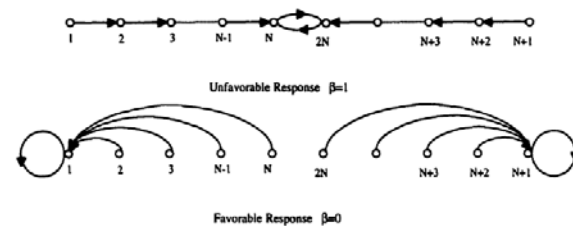
۲-۲-۱- آتوماتای یادگیر با ساختار ثابت

آتوماتای یادگیر با ساختار ثابت توسط ۵ تایی $\{\alpha, \beta, F, G, \Phi\}$ نشان داده می‌شود که در آن $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه کنش‌های آتوماتا، $\beta = \{\beta_1, \beta_2\}$ مجموعه ورودی‌های آتوماتا، $\Phi = \{\phi_1, \phi_2, \dots, \phi_s\}$ مجموعه وضعیت‌های داخلی آتوماتا، $F: \Phi \times \beta \rightarrow \Phi$ تابع تولید وضعیت جدید آتوماتا و $G: \Phi \rightarrow \alpha$ تابع خروجی که وضعیت کنونی را به خروجی بعدی می‌نگارد، می‌باشد. نمونه‌ای از آتوماتای یادگیر با ساختار ثابت آتوماتای Tsetlin است. این آتوماتا دارای $2N$ حالت و دو کنش α_1 و α_2 است. برای حالت‌های ۱ تا N کنش α_1 و برای حالت‌های $N+1$ تا $2N$ کنش α_2 انتخاب می‌شود. در شکل ۲ گراف تغییر حالت این نوع آتوماتا را به ازای پاداش و جریمه دریافتی از محیط نشان داده شده است. همانطوری که مشاهده می‌کنید رفتار این آتوماتا در محیط بدین صورت می‌باشد که به ازای دریافت پاداش در قبال اجرای یک کنش، فرصت انتخاب آن کنش را افزایش می‌دهد و به ازای دریافت جریمه در قبال اجرای یک کنش، فرصت انتخاب آن کنش را کاهش می‌دهد. در کل این آتوماتا بدین صورت رفتار می‌کند که تا زمانی که تعداد جریمه‌های دریافتی حاصل از اجرای یک کنش از تعداد پاداش دریافتی آن کنش بیشتر نشده است، همان کنش را در محیط انجام می‌دهد.



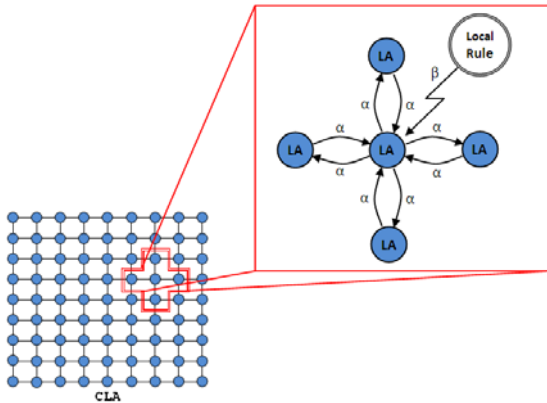
شکل ۲- رفتار آتوماتای Tsetlin [۲۸]

آتوماتای Krinsky نمونه‌ای دیگر از آتوماتای یادگیر با ساختار ثابت است. این آتوماتا نیز دارای $2N$ حالت و دو کنش α_1 و α_2 است. رفتار این آتوماتا در محیط بدین صورت می‌باشد که به ازای دریافت پاداش در قبال اجرای یک کنش به آن کنش بیشترین فرصت انتخاب را می‌دهد و به ازای دریافت جریمه در قبال اجرای یک کنش به اندازه یک واحد از فرصت انتخاب آن کنش می‌کاهد. در کل این آتوماتا بدین صورت رفتار می‌کند که در حین اجرای هر کنش در محیط چنانچه یک پاداش دریافت کند فرض را بر این می‌گذارد که این کنش خوب است و فرصت جریمه آن را تا حد بیشترین مقدار ممکن افزایش می‌دهد. شکل ۳ گراف تغییر حالت این نوع آتوماتا را بهتر نشان می‌دهد.

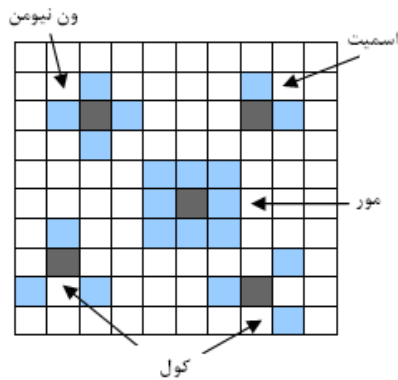


شکل ۳- رفتار آتوماتای Krinsky [۲۸]

همسایه در نظر گرفت، اما معمول ترین آنها همسایگی ون نیومن، مور، اسمیت و کول می باشد که به نزدیکترین همسایگان مشهور می باشند و در شکل ۶ نشان داده شده اند.



شکل ۵- ساختار آتوماتای یادگیر سلولی [۲۷]



شکل ۶- چند نوع همسایگی متداول

عملکرد آتوماتای یادگیر سلولی را می توان به شرح زیر بیان کرد. در هر لحظه هر آتوماتای یادگیر در آتوماتای یادگیر سلولی یک کنش از مجموعه کنش های خود را انتخاب می کند. این کنش می تواند بر اساس مشاهدات قبلی یا به صورت تصادفی انتخاب شود. کنش انتخاب شده، با توجه به کنش های انتخاب شده توسط سلول های همسایه و قانون حاکم بر آتوماتای یادگیر سلولی ارزیابی و پاداش داده و یا جریمه می شود و بدین اساس آتوماتا رفتار خود را تصحیح کرده و ساختار داخلی آتوماتا بروز می شود. معمولاً عمل بروزسانی تمام آتوماتاها به صورت همزمان انجام می شود. بعد از بروزسانی، هر آتوماتا در آتوماتای یادگیر سلولی دوباره یک کنش از مجموعه کنش های خود را انتخاب کرده و انجام می دهد.

فرآیند انتخاب کنش و پاداش دهی یا جریمه و به روز رسانی بردار احتمال آتوماتاها تا زمانی که سامانه به حالت پایدار برسد و یا یک معیار از قبل تعریف شده ای برقرار شود، ادامه می یابد. عمل بهنگام سازی ساختار آتوماتاها موجود در آتوماتای یادگیر سلولی توسط الگوریتم یادگیری انجام می شود. قوانین در آتوماتای یادگیر سلولی به سه دسته عمومی، کلی گرا و کلی گرای خارجی تقسیم می گردند. در قوانین عمومی^۸ مقدار یک سلول در مرحله بعدی، به مقدار تک تک سلول های همسایه آن سلول وابسته است ولی در قوانین کلی گرا^۹ مقدار یک سلول در مرحله بعدی، به تعداد سلول های همسایه که در حالتهای مختلف می باشند، وابسته است. در این نوع قانون برخلاف قانون عمومی، توجهی به مقدار تک تک سلول ها نمی شود. تنها تفاوتی که قوانین کلی گرای خارجی^{۱۱} با قوانین کلی گرا دارند در این

در روابط (۱) و (۲)، α پارامتر پاداش و b پارامتر جریمه هستند و میزان تغییرات در احتمال کنش ها را مشخص می کنند. با توجه به مقادیر a و b سه حالت را می توان در نظر گرفت. حالت اول زمانی که a و b با هم برابر باشند، الگوریتم را L_{RP} ^۵ می نامیم. در این الگوریتم هر کنش انتخابی به ازای دریافت پاداش از محیط، احتمال انتخاب خود را برای مراحل بعدی افزایش می دهد و از احتمال انتخاب کنش های دیگر کمی می کاهد و به ازای دریافت جریمه از محیط، احتمال انتخاب خود را برای مراحل بعدی کاهش می دهد و به احتمال انتخاب کنش های دیگر کمی می افزاید.

حالت دوم زمانی که b از a خیلی کوچکتر باشد، الگوریتم را L_{REP} ^۶ می نامیم. در این الگوریتم هر کنش انتخابی به ازای دریافت پاداش از محیط، احتمال انتخاب خود را برای مراحل بعدی افزایش می دهد و از احتمال انتخاب کنش های دیگر کمی می کاهد و به ازای دریافت جریمه از محیط، احتمال انتخاب خود را برای مراحل بعدی کاهش می دهد و به احتمال انتخاب کنش های دیگر کمی می افزاید. تفاوت این الگوریتم با روش L_{RP} در این می باشد که در این روش به ازای دریافت جریمه، مقدار افزایش احتمال کنش های دیگر با ضریب نسبتاً کوچکی ($\alpha < \beta$) صورت می گیرد، ولی در روش L_{RP} این افزایش احتمال کنش های دیگر در صورت جریمه شدن یک کنش با همان ضریب α صورت می گیرد.

حالت سوم زمانی که b مساوی صفر باشد، الگوریتم را L_{RI} ^۷ می نامیم. در این الگوریتم هر کنش انتخابی فقط به ازای دریافت پاداش از محیط احتمال انتخاب خود را برای مراحل بعدی افزایش می دهد و از احتمال انتخاب کنش های دیگر کمی می کاهد. در صورت جریمه شدن هم هیچ افزایش یا کاهش احتمالی صورت نمی گیرد. در [۲۶] توضیح کاملی از آتوماتاهای یادگیر با ساختار ثابت و متغیر به همراه روش های مختلف یادگیری و ویژگی های هر کدام بیان شده است.

۲-۳- آتوماتای یادگیر سلولی

در بسیاری از مسائلی نمی توان با استفاده از یک آتوماتای یادگیر عمل یادگیری را به درستی انجام داد بلکه قدرت اصلی آتوماتای یادگیر زمانی آشکار می شود که آنها به صورت دسته جمعی بکار روند. با توجه به این مساله و ضعف های عنوان شده برای آتوماتای سلولی، در [۶] با ترکیب این دو مدل، مدل جدیدی با نام آتوماتای یادگیر سلولی ایجاد شد.

آتوماتای یادگیر سلولی یک آتوماتای سلولی است که در هر سلول آن یک یا چند آتوماتای یادگیر قرار گرفته است. آتوماتاهای یادگیر هر سلول یکی از کنش های خود را انتخاب می کنند و در نتیجه وضعیت هر سلول با کنش های آتوماتاهای یادگیر واقع در آن تعیین می شود. در نتیجه در اصلاح رفتار یک سلول علاوه بر رفتار سلول های همسایه، تجربه گذشته آن سلول نیز تاثیر گذار خواهد بود. همانند آتوماتای سلولی، یک قانون محلی بر محیط حاکم است و این قانون تعیین می کند که آیا کنش انتخاب شده توسط آتوماتای یادگیر در یک سلول بایستی پاداش داده شود یا جریمه شود. عمل دادن پاداش و یا جریمه منجر به بروز در آوردن ساختار آتوماتای یادگیر سلولی به منظور رسیدن به خروجی مورد نظر می گردد.

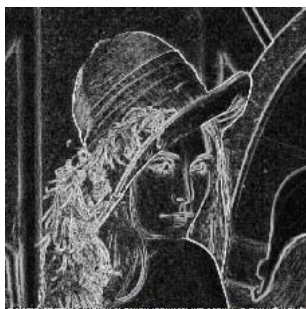
در شکل ۵ ساختار آتوماتای یادگیر سلولی به همراه ارتباط میان آتوماتاهای همسایه نشان داده شده است. در این ساختار آتوماتاهای همسایه از طریق تبادل اطلاعات مربوط به کنش انتخاب شده توسط آنها با آتوماتای مرکزی، در تصمیم گیری آتوماتای مرکزی در راستای پاداش یا جریمه مربوط به کنش انتخاب شده توسط آتوماتای مرکزی تاثیر می گذارند.

در آتوماتای یادگیر سلولی می توان از ساختارهای مختلفی برای همسایگی استفاده نمود. در حالت کلی هر مجموعه مرتب از سلول ها را می توان به عنوان

این الگوریتم بدین صورت می‌باشد که برای تمامی نقاط تصویر مقدار میانگین $\mu(i,j)$ و انحراف معیار $\sigma(i,j)$ همسایگی 3×3 آن محاسبه می‌شود. حال مقادیر این انحراف معیارهای بدست آمده به عنوان تصویر لبه‌های تصویر اصلی یا تصویر $I\sigma$ در نظر گرفته می‌شود. با این کار پیکسل‌هایی که دارای همسایگی یکنواخت می‌باشند دارای مقدار انحراف معیار کمتری و پیکسل‌هایی که دارای همسایگی غیریکنواخت می‌باشند دارای مقدار انحراف معیار بیشتری هستند. که این امر خود بیانگر لبه‌های یک تصویر است. در شکل ۷ یک تصویر و تصویر لبه آن نشان داده شده است.



الف) تصویر اصلی

ب) تصویر لبه $I\sigma$

شکل ۷- یک تصویر اصلی و تصویر لبه آن

برای اینکه از میان لبه‌های یافت شده، لبه‌های قویتر را تقویت نماییم از یک تابع غیرخطی که مقادیر بالاتر را بیشتر و مقادیر پایین‌تر را کمتر تقویت می‌کند، استفاده شده است. در اینجا از تابعی به شکل رابطه (۳) استفاده شده است.

$$f(i,j) = \sqrt{I\sigma(i,j)^\beta} \quad \beta > 2 \quad (3)$$

که در آن مقدار β بیانگر مقدار تقویت می‌باشد. هر چه این مقدار بیشتر باشد، لبه‌های ضعیف‌تری آشکار خواهد شد. در این مقاله از مقدار $\beta=3$ استفاده شده است. البته می‌توان از توابع دیگری که دارای خاصیت ذکر شده باشند هم استفاده نمود. پس از تبدیل فوق باید مقادیر $I\sigma(i,j)$ به بازه $[255,0]$ نگاشت شوند. شکل ۸ این تقویت و تضعیف لبه را نشان می‌دهد.

برای نادیده گرفتن لبه‌های بسیار ضعیف، ابتدا مقدار انحراف معیار تصویر $I\sigma$ محاسبه و σ' نامیده می‌شود و سپس این مقدار از مقادیر تمامی نقاط تصویر $I\sigma$ کم می‌شود و سپس مقادیر کوچکتر از صفر در تصویر $I\sigma$ برابر با صفر قرار داده می‌شود. این کار شبیه کاستن زمینه تصویر از تصویر اصلی می‌باشد که در اینجا بدین معناست که لبه‌های بسیار ضعیف نادیده گرفته شده است. الگوریتم زیر فرآیند فوق را به صورت کاملتری بیان می‌کند.

است که در تعیین مقدار بعدی سلول، علاوه بر حالت‌های فعلی سلول‌های همسایه آن سلول، حالت فعلی خود سلول نیز موثر است [۱]. در [۲۷] یک مدل ریاضی از رفتار اتوماتای یادگیر سلولی به همراه اثبات خاصیت همگرایی آن عنوان شده است.

۳- الگوریتم پیشنهادی برای یافتن لبه‌های تصویر

بسیاری از ویژگی‌های تصویر نظیر لبه‌ها، خطوط، انحنایها، گوشه‌ها و مرزها را می‌توان به کمک برخی اعمال ریاضی استخراج نمود. استخراج این ویژگی‌ها در تجزیه و تحلیل تصویر بسیار مهم است. هرچه فاز استخراج این ویژگی‌ها با دقت بیشتری انجام شود، فازهای تجزیه و تحلیل بعدی نیز کارآمدتر و با دقت بیشتری انجام خواهد پذیرفت. یکی از این ویژگی‌های تصویر که در حوزه پردازش تصویر و بینایی ماشین بسیار پرکاربرد بوده و از مباحث پایه‌ای و بسیار مهم در این حوزه به شمار می‌رود، لبه‌های تصویر می‌باشد.

روش‌های زیادی از قبیل سوپل [۳۲]، پرویت [۳۲]، کنی [۳۱] و ... برای آشکارسازی این ویژگی مطرح شده‌اند که بیشتر آنها از خاصیت گرادیان تصویر و برخی عملیات ریاضی برای این امر استفاده می‌کنند. بیشتر این روش‌های معمول، پارامتریک هستند یعنی برای امر آشکارسازی لبه‌های تصاویر متفاوت، به یک پارامتر خاص وابسته هستند که دقت لبه‌های مورد جستجو را بیان می‌کند. البته برخی روشها سعی می‌کنند به صورت وقتی این پارامترها را برای هر تصویر بدست آورند که در بسیاری موارد ناکارآمد بوده و نمی‌توانند لبه‌های تصویر را به خوبی آشکار کنند. از دیگر ویژگی‌های آشکارسازی لبه متداول این است که بیشتر آنها از روشهای مبتنی بر گرادیان برای آشکارسازی لبه‌های تصویر استفاده می‌کنند. در برخی از این نوع آشکارسازی لبه، این ویژگی تا حدی با برخی عملیات ریاضی ترکیب شده و توانسته یک آشکارساز لبه با خاصیت وقتی تولید کند.

در این بخش با ترکیب یک روش آماری برای آشکارسازی لبه‌های تصویر و استفاده از اتوماتای یادگیر سلولی برای بهبود لبه‌های آشکار شده، یک آشکارساز وقتی لبه ارائه شده است. در این روش ابتدا با استفاده از تجزیه و تحلیل آماری تصویر و بررسی وجود لبه در نواحی مختلف تصویر، لبه‌های تصویر تقویت شده و نقاط غیر لبه تضعیف می‌شوند. این روش که به عنوان یک فاز پیش‌پردازش برای اتوماتای یادگیر سلولی می‌باشد، بسیار قدرتمند بوده و در تقویت لبه‌های تصویر بسیار کارآمد است. پس از فاز پیش‌پردازش، از اتوماتای یادگیر سلولی برای تقویت و بهبود لبه‌های تقویت شده در فاز پیش‌پردازش استفاده می‌شود. در این فاز اتوماتای یادگیر سلولی به کمک خاصیت همسایگی و تکرارپذیری خود نوبه‌های فاز قبل را برطرف نموده و سبب بهبود لبه‌های آشکار شده خواهد شد.

در یک تصویر سطح خاکستری اگر شدت روشنایی تمامی نقاط تصویر به هم نزدیک باشد، یا به عبارت دیگر اینکه تصویر یکنواخت باشد در این صورت، چنین تصویری دارای لبه‌های کمی می‌باشد. زیرا لبه‌های تصویر، محل تغییر غیریکنواخت در سطوح خاکستری تصویر می‌باشند. مطلب فوق از دیدگاه آماری بدین معناست که انحراف معیار شدت روشنایی نقاط تصویر پایین باشد. برعکس اگر تصویر غیر یکنواخت باشد در این صورت، تعداد پیکسل‌های لبه در این تصویر بالا می‌باشد از دیدگاه آماری مطلب فوق را می‌توان بدین صورت بیان کرد که انحراف معیار شدت روشنایی نقاط تصویر بالا باشد.

در این روش با بهره‌گیری از خصوصیات آماری ذکر شده برای پنجره‌های 3×3 از تصویر، لبه‌ها با کیفیت بالایی تشخیص داده شده و تقویت می‌شوند. روش کار

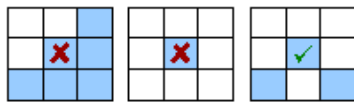
اتوماتای یادگیر با ساختار متغیر با الگوریتم یادگیری L_{REP} در نظر گرفته شده است که با همسایگی ون نیومن با اتوماتاهای همسایه در ارتباط است و هر اتوماتای یادگیر دارای دو کنش لبه‌بودن و لبه‌نبودن است. مقادیر احتمال اولیه برای هر یک از کنش‌های لبه‌بودن و نبودن از هر اتوماتای یادگیر باید از روی تصویر $I\sigma$ نهایی محاسبه شود. رابطه (۴) می‌تواند به عنوان یک رابطه که مقادیر تصویر نهایی را به مقدار اولیه احتمال اتوماتاها نگاشت می‌کند به کار رود.

$$P_{edge}(i,j) = \frac{\frac{I\sigma(i,j)}{\sigma}}{\max\left(\frac{I\sigma(i,j)}{\sigma}\right)} \quad (4)$$

$$P_{not-edge}(i,j) = 1 - P_{edge}(i,j)$$

قوانین محلی حاکم بر این اتوماتای یادگیر سلولی بدین صورت تعریف شده است که در تکرارهای متوالی بتواند پیکسل‌های نویز و غیرلبه را تضعیف و پیکسل‌های لبه را تقویت کند. از سوی دیگر بتواند پیکسل‌های لبه‌ای که در میان دو پیکسل لبه قرار گرفته‌اند ولی به عنوان پیکسل لبه تشخیص داده نشده‌اند و یا اینکه لبه ضعیفی هستند را تقویت کند و از طرف دیگر بتواند پیکسل‌هایی که لبه نیستند و به عنوان لبه قوی تشخیص داده شده‌اند را تضعیف کند. برای تقویت لبه‌ها، هنگامی که بین دو تا چهار همسایه از یک اتوماتای یادگیر کنش لبه بودن را انتخاب کرده باشند و اتوماتای یادگیر مرکزی هم آن کنش را انتخاب کرده باشد، در این صورت به کنش لبه بودن انتخابی توسط اتوماتای یادگیر مرکزی پاداش تعلق خواهد گرفت در غیر این صورت آن کنش جریمه خواهد شد. در مورد تضعیف لبه‌ها هم بدین صورت عمل می‌کند که هنگامی که بیش از چهار همسایه اتوماتای یادگیر مرکزی کنش لبه بودن را انتخاب کنند و یا اینکه هیچکدام از همسایه‌های اتوماتای یادگیر کنش لبه بودن را انتخاب نکنند ولی اتوماتای یادگیر مرکزی کنش لبه بودن را انتخاب کند در این صورت جریمه خواهد شد.

شکل ۱۰ دیدگاه شهودی بهتری نسبت به مطلب فوق ارائه می‌دهد. خانه‌های آبی‌رنگ بیانگر انتخاب کنش لبه‌بودن توسط اتوماتای یادگیر آن سلول می‌باشند. برای مثال در ستون میانی شکل ۱۰ اتوماتای میانی پیکسل منتسب خود را به عنوان یک پیکسل لبه معرفی نموده است در حالی که هیچیک از اتوماتاهای یادگیر همسایه آن پیکسل منتسب خود را به عنوان لبه معرفی ننموده‌اند. در این حالت قانون حاکم بر محیط این پیکسل را به عنوان یک پیکسل نویز در نظر می‌گیرد و اتوماتای میانی جریمه می‌شود.



شکل ۱۰- تقویت و تضعیف لبه‌ها توسط اتوماتای یادگیر سلولی

برای تضعیف و تقویت لبه‌های نامتصل هم بدین صورت عمل شده است که چنانچه فقط دو همسایه از اتوماتای یادگیر مرکزی به همراه اتوماتای یادگیر مرکزی پیکسل خود را به عنوان پیکسل لبه تشخیص دهند، در این صورت کنش انتخاب لبه اتوماتای یادگیر مرکزی پاداش می‌گیرد. در ستون سمت چپ شکل ۱۱ نحوه پاداش دادن اتوماتای یادگیر مرکزی در تقویت لبه‌های متصل نشان داده شده است. چنانچه اتوماتای یادگیر مرکزی پیکسل خود را به عنوان لبه تشخیص ندهد، کنش عدم انتخاب لبه اتوماتای یادگیر مرکزی جریمه می‌شود. نحوه جریمه کردن اتوماتای یادگیر مرکزی در صورت عدم تقویت لبه‌های متصل، در ستون سمت راست شکل ۱۱ نشان داده شده است.

For $i:1..M$ & $j:1..N$
 $(i,j) = I\sigma(i,j) - \sigma'$
 if $I\sigma(i,j) < 0$ then $(i,j) = 0$
 End



شکل ۸- تصویر تقویت شده توسط تابع f

برای تقویت مجدد لبه‌های باقیمانده از تابع غیرخطی f مرحله قبل استفاده می‌شود. شکل ۹ این تغییر نهایی در تصویر $I\sigma$ را به عنوان تصویر لبه نهایی نشان می‌دهد.



شکل ۹- تصویر لبه نهایی

همانگونه که مشاهده می‌کنید الگوریتم فوق توانسته تا حد خوبی لبه‌های تصویر را مشخص و تقویت کند و نقاط غیر لبه را از تصویر کم کند. از ویژگی‌های بارز این روش آشکارسازی لبه، خاصیت وقتی آن برای هر تصویر می‌باشد. این بدان معناست که در تصاویر مختلف با یکنواختی و شدت روشنایی متفاوت، با توجه به خصوصیات آماری شدت روشنایی پیکسل‌های همان تصویر، لبه‌های آن را به بهترین نحو آشکار می‌کند.

اگر در یک تصویر آشکارسازی لبه بسیار مشکل باشد، باز هم از آنجا که این الگوریتم با توجه به خواص آماری همان تصویر عمل می‌کند، قادر است تا لبه‌های موجود در آن را آشکار کند. تصویر لبه $I\sigma$ که تاکنون بدست آمده است برای بهبود و تقویت لبه به عنوان ورودی به یک اتوماتای یادگیر سلولی داده می‌شود تا اتوماتای یادگیر سلولی لبه‌های باکیفیت را تقویت و نویزها و نقاط غیرلبه را تضعیف و حذف کند. در بخش بعد نحوه استفاده از اتوماتای یادگیر سلولی برای این امر را به طور کامل شرح داده شده است.

۳-۱- استفاده از اتوماتای یادگیر سلولی برای بهبود لبه

در این بخش از یک اتوماتای یادگیر سلولی به همراه یک سری قوانین محلی برای بهبود لبه‌ها استفاده شده است. در این بخش هر نقطه از تصویر به عنوان یک

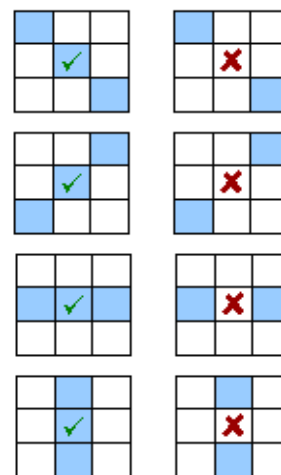
۲-۲- نتایج پیاده‌سازی

هدف از این بخش ارزیابی کارایی الگوریتم پیشنهادی می‌باشد. برای این منظور نتیجه الگوریتم روی چند تصویر نمونه ارزیابی و کارایی آن با الگوریتم‌های دیگر در این زمینه مقایسه خواهد شد. در این بخش نخست پارامترهای مربوط به الگوریتم تشخیص لبه بیان می‌شود و سپس نتایج حاصله از پیاده‌سازی الگوریتم پیشنهادی نمایش داده شده و با چند روش معروف آشکار سازی لبه مقایسه می‌شود. در این مقاله از مقدار $\beta=3$ برای تابع $f(i, j)$ بیان شده در رابطه (۳) استفاده شده است. روش یادگیری هر اتوماتای یادگیر با ساختار متغیر L_{REP} با مقدار اولیه $\alpha=0.01$ و $b=0.001$ و $r=2$ می‌باشد که به تعداد ۱۰۰۰ مرحله برای هر تصویر تکرار شده است.

شکل‌های ۱۳ تا ۱۸ نتایج تولید شده توسط الگوریتم‌های مختلف را برای چند تصویر نمایش می‌دهند. همان‌گونه که مشاهده می‌شود، روش پیشنهادی در تمامی تصاویر توانسته است لبه‌های واضح‌تری از روش سوپل استخراج کند و در مقابل روش کنی هم می‌توان ادعا کرد که در برخی موارد لبه‌ها را با دقت و جزئیات بیشتری آشکار می‌کند. از ویژگی‌های بسیار جالب این روش که از روی نتایج بدست آمده هم قابل درک می‌باشد این است که این روش آشکار سازی لبه در مقایسه با دو روش دیگر ساختار کلی تصویر را بهتر حفظ می‌کند یعنی با نگه داشتن لبه‌های مهم به گونه‌ای عمل می‌کند که ساختار لبه‌ای تصویر که بیانگر یک دید کلی از تصویر می‌باشد، بهتر حفظ شود و شکل طبیعی تری داشته باشد، در حالی که روش‌های مبتنی بر گرادیان دیگر فاقد این ویژگی هستند. عبارت فوق بدین معناست که در این روش آشکار سازی لبه، بیننده با مشاهده تصویر لبه ذهنیت بهتری از ساختار تصویر پیدا می‌کند. در شکل ۱۳ ادعای فوق کاملاً مشهود است. همانگونه که در این شکل‌ها مشاهده می‌کنید لبه‌های آشکار شده توسط الگوریتم پیشنهادی در برابر لبه‌های آشکار شده توسط الگوریتم‌های سوپل و کنی شکل طبیعی تری دارد. برای مثال اگر به لبه‌های مربوط به چشم و لب و بینی در شکل ۱۴ دقت شود، مشاهده می‌شود که الگوریتم پیشنهادی این لبه‌ها را به شکل زیباتری استخراج نموده است.

۴- الگوریتم پیشنهادی برای قطعه‌بندی تصویر

قطعه‌بندی تصویر و ویدئو به عنوان یک مرحله پیش‌پردازش برای اعمالی نظیر پیداکردن ناحیه دلخواه در یک صحنه، تفسیر داده‌ها و بازیابی تصاویر محسوب می‌شود. همچنین استاندارد MPEG-4، از قطعه‌بندی برای کدکردن ویدئو استفاده می‌کند. در این فرآیند، تصویر به تعدادی ناحیه جدا از هم تقسیم و در هر ناحیه مجموعه‌ای از پیکسل‌ها که از لحاظ چند معیار به یکدیگر شبیه هستند، قرار می‌گیرند. تاکنون روش‌های زیادی از قبیل روش‌های مبتنی بر مدل‌سازی آماری [۱۶، ۱۷، ۱۸]، انتشار انرژی [۱۹]، افراز گراف [۲۰] و قطعه‌بندی بدون ناظر [۲۱] برای قطعه‌بندی تصاویر مطرح شده‌اند. بسیاری از روش‌های قطعه‌بندی موجود مانند خوشه‌بندی پیکسل‌ها در فضای رنگی، بر روی نواحی همگن به خوبی عمل می‌کنند اما از آنجا که تصاویر طبیعی از تنوع رنگ و بافت بالایی برخوردار هستند، در مجموع کارایی این روش‌ها پایین می‌باشد. روش‌های مبتنی بر بافت، که به عنوان یک روش برای قطعه‌بندی مطرح شده‌اند، نیاز به تخمین پارامترهای مدل بافت دارند. تخمین این پارامترها پیچیده بوده و در اغلب اوقات تخمین مناسب پارامترها نیازمند وجود نواحی همگن در تصویر می‌باشد. در [۱۰، ۱۳] از اتوماتای یادگیر سلولی به منظور قطعه بندی نواحی رنگ پوست در تصاویر استفاده شده است که از اطلاعات رنگ و بافت پوست به همراه اتوماتای یادگیر سلولی بدین منظور استفاده می‌کند.



شکل ۱۱- تقویت و تضعیف لبه‌های متصل و منفصل

برای مثال پس از ۱۰۰۰ بار تکرار الگوریتم بهبود لبه توسط اتوماتای یادگیر سلولی، تصویر شکل ۱۲ به عنوان تصویر لبه نهایی تشخیص داده شده است. شکل ۱۲ تصویر لبه نهایی حاصل از اتوماتای یادگیر سلولی را به همراه معکوس آن نشان می‌دهد.



الف) تصویر لبه نهایی



ب) معکوس تصویر لبه نهایی

شکل ۱۲- الف) تصویر لبه بدست آمده توسط اتوماتای یادگیر سلولی به همراه معکوس آن در (ب)

همانگونه که مشاهده می‌کنید تصویر لبه بدست آمده از اتوماتای یادگیر سلولی دارای کیفیت بیشتری نسبت به تصویر حاصله از مرحله پیش پردازش قبلی می‌باشد. در اینجا منظور از کیفیت، در نظر گرفتن معیارهای بصری می‌باشد.



الف) تصویر اصلی

ب) روش سوبل



ج) روش کنی

د) روش پیشنهادی

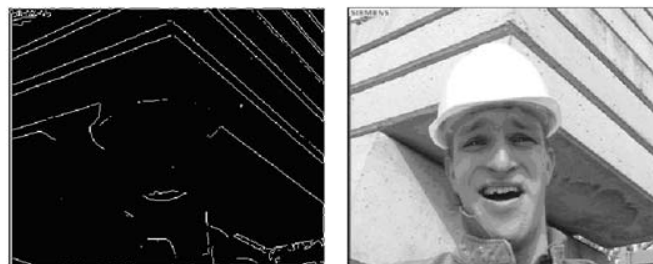
شکل ۱۳- مقایسه لبه‌های بدست آمده از چند روش با روش پیشنهادی



الف) روش کنی

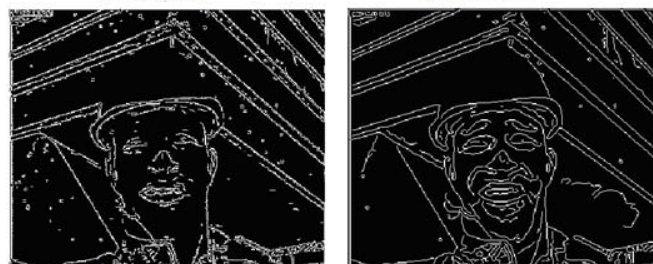
ب) روش پیشنهادی

شکل ۱۴- مقایسه لبه‌های بدست آمده به روش کنی و پیشنهادی



الف) تصویر اصلی

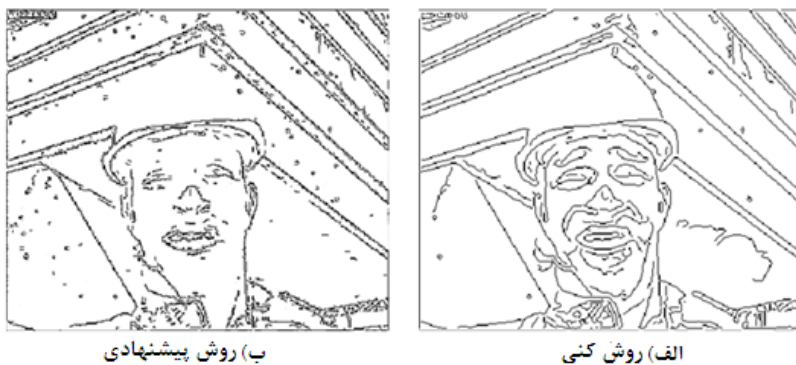
ب) روش سوبل



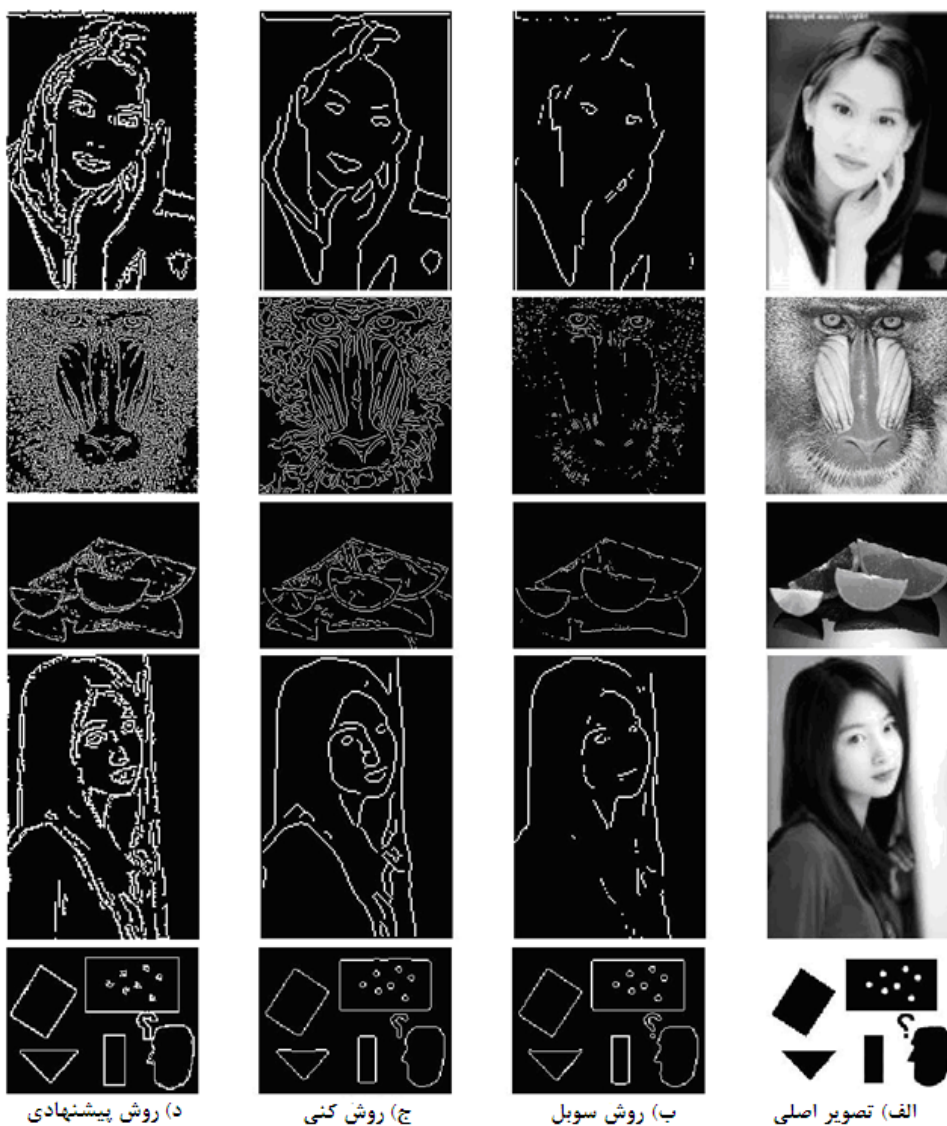
ج) روش کنی

د) روش پیشنهادی

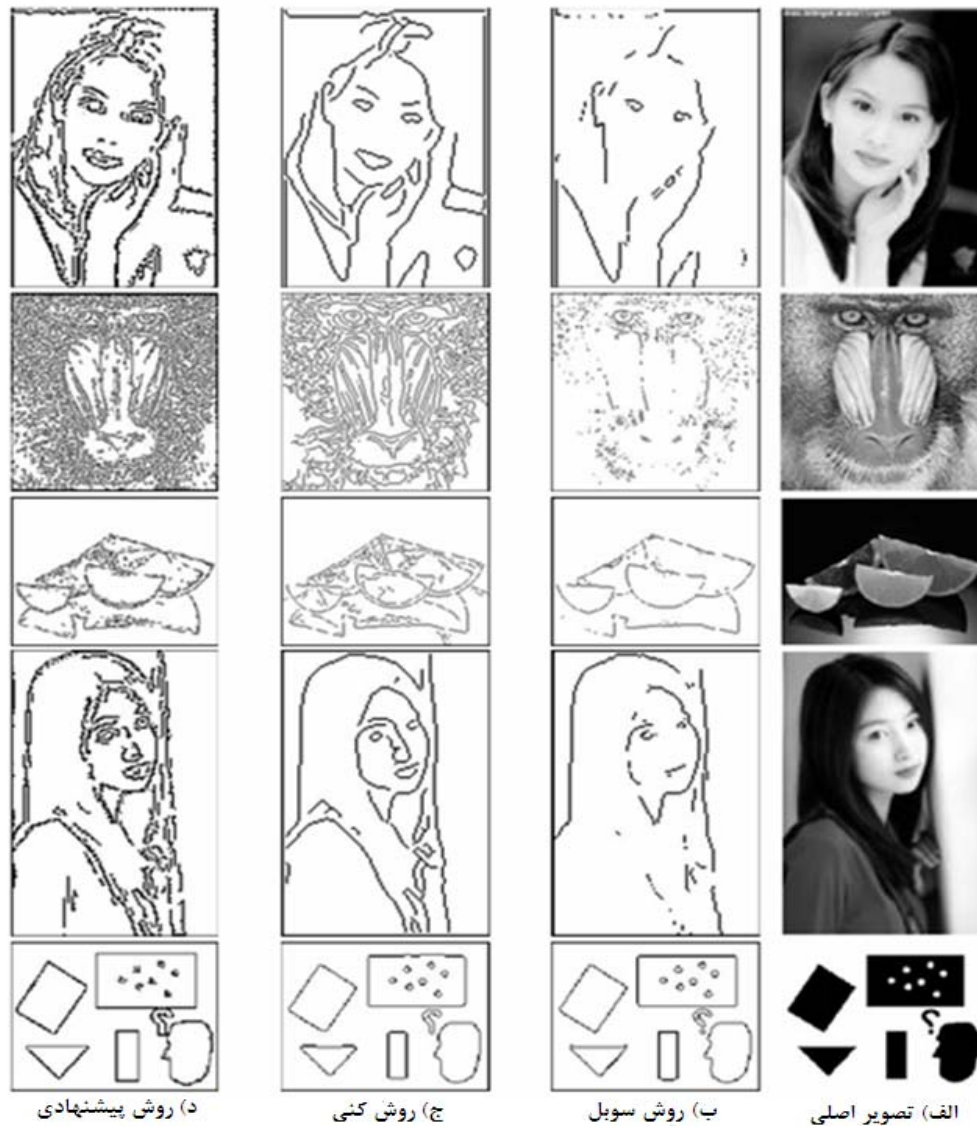
شکل ۱۵- مقایسه لبه‌های بدست آمده از چند روش با روش پیشنهادی



شکل ۱۶- مقایسه لبه‌های بدست آمده به روش کنی و پیشنهادی



شکل ۱۷- مقایسه لبه‌های بدست آمده از چند روش با روش پیشنهادی



شکل ۱۸- مقایسه لبه‌های بدست آمده از چند روش با روش پیشنهادی

تکراری برای رسیدن به خروجی مناسب استفاده می‌کند. از آنجایی که در قطعه‌بندی تصاویر، خروجی الگوریتم باید یک تصویر قطعه‌بندی شده باشد، بنابراین از چند مرحله پیش پردازش و پس پردازش برای الگوریتم مبتنی بر اتوماتای یادگیر سلولی استفاده می‌شود. این عملیات باعث تسریع همگرایی اتوماتای یادگیر سلولی خواهند شد. در این روش بدینگونه عمل می‌شود که ابتدا بر روی تصویر ورودی، قطعه‌بندی مبتنی بر اتوماتای یادگیر سلولی صورت می‌پذیرد و بر روی تصویر حاصله عمل استخراج زنجیره‌ها برای هر پیکسل و انتشار رنگ آن زنجیره برای آن پیکسل صورت می‌گیرد. این چرخه تا رسیدن به یک شرط توقف خاص ادامه می‌یابد.

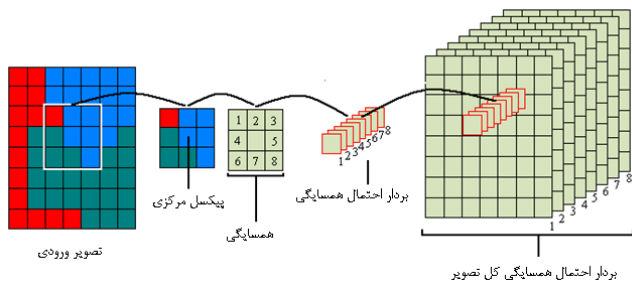
در پایان هم با اعمال یک حد آستانه عمل ادغام نواحی صورت می‌گیرد. تصویر حاصل از فاز ادغام همان تصویر قطعه‌بندی شده نهایی می‌باشد. شکل ۱۹ ساختار کلی الگوریتم قطعه‌بندی مبتنی بر اتوماتای یادگیر سلولی را نشان می‌دهد.

برای قطعه‌بندی مبتنی بر اتوماتای یادگیر سلولی، فرض می‌شود که هر پیکسل تصویر به یک سلول در اتوماتا نگاشت می‌یابد و رابطه همسایگی سلول‌ها از نوع همسایگی مور است. بنابراین هر پیکسل با پیکسل‌های واقع در همسایگی 3×3 آن

در روش پیشنهادی در [۱۰، ۱۳] قطعه بندی فقط برای یک کلاس خاص (رنگ پوست) در نظر گرفته شده است که در آن از اتوماتای یادگیر سلولی به عنوان ابزاری جهت تحلیل اطلاعات بافت نواحی استفاده شده است. در روش پیشنهادی نه تنها از اتوماتای یادگیر سلولی به عنوان ابزاری جهت پردازش بافت استفاده می‌شود بلکه به عنوان ابزاری جهت استخراج زنجیره‌ای از تشابهات پیکسلی به منظور انتشار رنگ تصویر مورد پردازش تا همگرایی به یک تصویر قطعه بندی شده استفاده می‌شود.

در این بخش یک الگوریتم برای قطعه‌بندی تصاویر رنگی با استفاده از اتوماتای یادگیر سلولی معرفی می‌شود. در بسیاری از الگوریتم‌های قطعه‌بندی موجود، برای یافتن نواحی تصویر معیارهایی تعریف و سپس بر اساس این معیارها و تشابه پیکسل‌های واقع در یک همسایگی، تعلق هر کدام از پیکسل‌ها به یک قطعه تعیین می‌شود. هر کدام از دو جزء یاد شده (تعریف معیار و همسایگی) می‌توانند به راحتی توسط یک اتوماتای یادگیر سلولی پیاده‌سازی شوند.

برای بررسی تشابه پیکسل‌های واقع در یک همسایگی می‌توان از رابطه همسایگی موجود در اتوماتای یادگیر سلولی بهره گرفت و برای تعریف معیار می‌توان یک قانون مناسب طراحی نمود. اتوماتای یادگیر سلولی از یک الگوریتم



شکل ۲۱- چگونگی انتساب آتوماتای یادگیر سلولی به تصویر به همراه بردار احتمال انتخاب کنش‌ها

که I تصویر ورودی، (x_i, y_i) مختصات همسایه شماره i پیکسل (x, y) و $D_i(x, y)$ فاصله بین پیکسل (x, y) تا همسایه i ام آن می‌باشد. احتمال انتخاب کنش‌ها با استفاده از معکوس این فاصله‌ها تعیین می‌شود. رابطه (۶) نحوه محاسبه این احتمال‌ها را نشان می‌دهد که در آن $S_i(x, y) = \frac{1}{D_i(x, y)}$ معکوس فاصله اقلیدسی پیکسل i ام تا پیکسل مرکزی و $P_i(x, y)$ احتمال انتخاب کنش i ام می‌باشد.

$$P_i(x, y) = \frac{S_i(x, y)}{\sum_{j=1}^8 S_j(x, y)} \quad (۶)$$

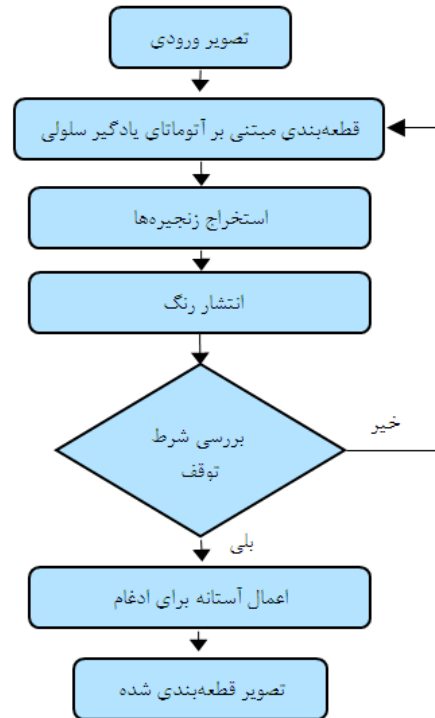
قانونی که برای پاداش یا جریمه کردن یک کنش استفاده می‌شود بدین صورت است که ابتدا فاصله پیکسل واقع در یک سلول تا تمامی سلول‌های همسایه آن سلول با استفاده از رابطه (۵) محاسبه می‌شود. میانگین این فواصل محاسبه شده و با $D_m(x, y)$ نشان داده می‌شود. فرض کنید که این سلول کنش i را انتخاب کند. بنابراین فاصله پیکسل این سلول را تا پیکسل واقع در i امین سلول همسایه آن محاسبه شده و با $D_i(x, y)$ نشان داده می‌شود. قانونی که برای این سلول به کار گرفته می‌شود، به صورت رابطه (۷) می‌باشد که C یک ضریب ثابت نزدیک به یک است.

$$\text{قانون} = \begin{cases} \text{پاداش} & c \times D_i(x, y) \geq D_m(x, y) \\ \text{جریمه} & c \times D_i(x, y) < D_m(x, y) \end{cases} \quad (۷)$$

از آنجا که در هر مرحله باید پیکسل‌های واقع در یک قطعه در فضای رنگی به هم نزدیک شوند بنابراین نیاز به روشی جهت انتشار رنگ در یک قطعه است. بدین منظور پس از هر بار انجام قطعه‌بندی مبتنی بر آتوماتای یادگیر سلولی یکبار عمل استخراج زنجیره‌ها و انتشار رنگ صورت می‌پذیرد. پس از انجام اعمال استخراج زنجیره‌ها و انتشار رنگ آنها تصویری ایجاد می‌شود که در آن پیکسل‌های واقع در یک قطعه از لحاظ رنگی به یکدیگر نزدیک می‌باشند. فرض کنید که تصویر ورودی را با I و تصویر خروجی را با I' نشان دهیم. برای تعیین مقدار پیکسل‌ها در تصویر I' ابتدا یک زنجیره از پیکسل‌ها با شروع از مختصات (x, y) استخراج می‌گردد. شبه کد مربوط به این عمل در شکل ۲۲ شبه کد مربوط به استخراج زنجیره پیکسل‌ها نشان داده شده است.

در این شبه کد LA بیانگر آتوماتای یادگیر یک سلول است و زنجیره استخراج شده برای مختصات (x, y) در pixels_list نگه داشته می‌شود. مطابق شکل ۲۲ نخست لیست پیکسل‌ها خالی است و سلول فعلی (x, y) است. نخست سلول فعلی به لیست پیکسل‌ها اضافه می‌شود و سپس سلول همسایه‌ای که مطابق با کنش انتخاب شده در سلول فعلی است، تعیین می‌گردد. سپس سلول فعلی برابر با سلول همسایه قرار می‌گیرد تا تکرارهای بعدی با این سلول ادامه یابد. مراحل ذکر شده تا

همسایه است. در هر سلول یک آتوماتای یادگیر با ساختار متغیر از نوع LRep قرار می‌گیرد. به ازای هر پیکسل همسایه، یک کنش برای آتوماتای یادگیر واقع در یک سلول تعریف می‌شود و بنابراین آتوماتای موجود در هر سلول دارای ۸ کنش برای انتخاب است. انتخاب کنش توسط هر آتوماتای متناسب به هر پیکسل بدین معنی است که آن پیکسل خود را هم قطعه با آن همسایه انتخاب شده می‌داند. در هر مرحله احتمال انتخاب کنش‌های مجاز در یک سلول با توجه به خروجی قانون به کار گرفته شده (پاداش یا جریمه) توسط روابط (۱) و (۲) تغییر می‌کنند. شکل ۲۰ نحوه تعریف کنش‌های آتوماتای یادگیر واقع در سلول مرکزی را نشان می‌دهد.



شکل ۱۹- فلوجارت روش قطعه‌بندی مبتنی بر آتوماتای یادگیر سلولی

1	2	3
4		5
6	7	8

شکل ۲۰- نحوه تعریف کنش‌های سلول مرکزی

هر کنش بیانگر این است که پیکسل واقع در یک سلول با کدام یک از پیکسل‌های همسایه آن، در یک قطعه قرار می‌گیرند. از آنجا که یک پیکسل به تنهایی نمی‌تواند یک قطعه را تشکیل دهد و این چنین پیکسلی نویز محسوب می‌شود برای سلول مرکزی واقع در شکل ۲۰ کنشی تعریف نمی‌شود. در شکل ۲۱ چگونگی انتساب آتوماتای یادگیر سلولی به تصویر به همراه بردار احتمال انتخاب کنش‌ها نشان داده شده است.

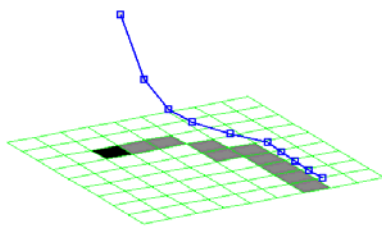
برای محاسبه احتمال اولیه انتخاب کنش‌ها در هر سلول، ابتدا فاصله اقلیدسی پیکسل واقع در آن سلول با پیکسل‌های سلول‌های همسایه در فضای رنگی RGB با استفاده از رابطه (۵) محاسبه می‌شود.

$$D_i(x, y) = \sqrt{\sum_{k=r,g,b}^3 (I(x, y, k) - I(x_i, y_i, k))^2} \quad (۵)$$

عنصر i ام لیست و وزن متناظر با آن می‌باشند. نکته‌ای که باید مورد توجه قرار گیرد این است که اگر اتوماتای واقع در یک سلول جریمه شود آنگاه لیست آن تنها شامل همان سلول است و بنابراین پیکسل آن سلول بدون تغییر به تصویر خروجی انتقال می‌یابد. وزن‌های متناظر با هر عنصر لیست باید به گونه‌ای طراحی شوند که یک روند نزولی را از ابتدای لیست تا انتهای آن طی کنند زیرا عناصر ابتدایی لیست از اهمیت بیشتری در تعیین پیکسل خروجی برخوردار هستند. رابطه (۹) یک نمونه از تعریف این وزن‌ها را نشان می‌دهد که w_i وزن عنصر i ام لیست استخراج شده برای مختصات (x, y) را نشان می‌دهد.

$$w_i = \frac{1}{2^{i-1}} \quad (9)$$

در شکل ۲۴ رفتار این تابع توزیع وزن برای یک زنجیر نمونه نشان داده شده است.



شکل ۲۴- رفتار تابع توزیع وزن برای یک زنجیر نمونه

همانگونه که در بالا شرح داده شد، رابطه (۸) قادر است تفاوت رنگی بین پیکسل‌های واقع در یک قطعه را تا حد زیادی از بین ببرد. یکی از پدیده‌هایی که می‌تواند کارایی این روش را پایین بیاورد وجود جزئیاتی مانند لبه‌های بسیار ریز که در نتیجه بافت یک قطعه به وجود می‌آیند، می‌باشد. برای کاهش اثر این گونه جزئیات از فیلتر گوسی^{۱۱} رابطه (۱۰) استفاده می‌شود.

$$h(m, n; \sigma) = \frac{\exp[-(m^2+n^2)/2\sigma^2]}{\sum_i \sum_j \exp[-(i^2+j^2)/2\sigma^2]} \quad (10)$$

که m و n سطر و ستون فیلتر h را نشان می‌دهند که در مرکز فیلتر مقدار آنها صفر خواهد بود. پارامتر انحراف معیار استاندارد فیلتر است که توسط آن میزان تاثیر پیکسل‌های همسایه در پیکسل قرار گرفته در مرکز فیلتر را مشخص می‌کنند. هرچه این پارامتر بزرگتر باشد، نقش پیکسل‌های همسایه نیز بیشتر خواهد شد. در الگوریتم پیشنهادی پس از K مرحله تاثیر همسایگی در پیکسل‌های تصویر خروجی با استفاده از رابطه (۸)، یک مرتبه فیلتر گوسی 3×3 را اعمال می‌کنیم. از آنجا که با افزایش تعداد تکرارهای الگوریتم اتوماتای یادگیر سلولی، لازم است میزان تغییرات تصویر خروجی حاصل از اعمال این فیلتر کاهش یابد، بنابراین پارامتر σ در هر مرحله از اعمال فیلتر با استفاده از رابطه (۱۱) کاهش می‌یابد.

$$\sigma(t) = \sigma_I - \frac{\sigma_I - \sigma_F}{T} t \quad (11)$$

که $T = N/K$ و $\sigma_I > \sigma_F$ به ترتیب انحراف معیار های اولیه و نهایی فیلتر را نشان می‌دهند و N تعداد کل تکرارهای اتوماتای یادگیر سلولی است. در انتهای تکرارهای الگوریتم اتوماتای یادگیر سلولی، یک تصویر حاصل می‌شود که در آن تفاوت پیکسل‌های واقع در یک قطعه تا حد بسیار زیادی کنار

زمان جریمه شدن سلول فعلی توسط قانون تعریف شده یا ایجاد حلقه در لیست پیکسل‌ها تکرار می‌شوند.

Extract_Chain (x, y)

Set current cell to (x, y) .

Assign pixels_list to an empty set.

Repeat

Add current cell to pixels_list.

Set α to selected action in current cell.

Set current cell to (x', y') which (x', y') is the neighborhood of current cell corresponds to action α .

Until (LA in current cell is penalized or a cycle is detected in pixels_list)

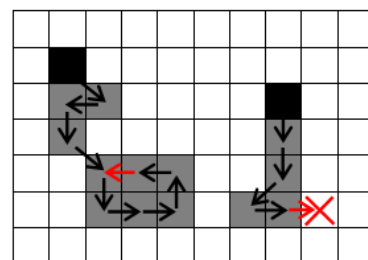
Return pixels_list.

End Extract_Chain

شکل ۲۲- شبه کد مربوط به استخراج زنجیره پیکسل‌ها

برای هر یک از پیکسل‌های موجود در مختصات (x, y) تصویر ورودی یک لیست محاسبه می‌شود که در آن عنصر ابتدایی پیکسل موجود در مختصات (x, y) است. اگر قانون تعریف شده برای اتوماتای سلولی برای پیکسل موجود در یک مختصات جریمه را انتخاب کند آنگاه لیست آن تنها دارای یک عضو (که خود پیکسل است) خواهد بود. بنابراین هر لیست دست کم یک عضو خواهد داشت.

در شکل ۲۳ دو نمونه از لیست پیکسل‌های استخراج شده توسط شبه کد شکل ۲۲ نشان داده شده‌اند. در هر لیست، خانه‌های سیاه بیانگر نقطه آغازین و خانه‌های خاکستری رنگ پیکسل‌های همسایه آن هستند. همچنین پیکان‌های نشان داده شده بیانگر رابطه همسایگی هستند. در لیست سمت راست شکل ۲۳ پیدا کردن همسایه‌ها تا زمانی که جریمه دریافت شده است ادامه پیدا کرده است. سلولی که جریمه شده است با ضربدر نشان داده شده است در حالی که در لیست سمت چپ، پیدا کردن همسایه‌ها تا ایجاد حلقه در لیست ادامه یافته است.



شکل ۲۳- دو مثال از زنجیره استخراج شده

پس از تعیین لیست همسایگی برای هر سلول، مقدار پیکسل‌ها در تصویر خروجی با استفاده از میانگین وزن‌دار لیست متناظر با آن پیکسل محاسبه می‌شوند. این عمل متناظر با عمل انتشار رنگ می‌باشد که در شکل ۱۹ بیان شده است. رابطه (۸) نحوه محاسبه مقدار هر پیکسل را با توجه به زنجیره استخراج شده برای آن پیکسل نشان می‌دهد.

$$I'(x, y) = \frac{\sum_{i=1}^{|L(x,y)|} w_i I\{L_i(x,y)\}}{\sum_{i=1}^{|L(x,y)|} w_i} \quad (8)$$

که در آن I تصویر ورودی، I' تصویر خروجی، $|L|$ بیانگر تعداد اعضای مجموعه، لیست استخراج شده برای مختصات (x, y) و $L_i(x, y)$ و w_i به ترتیب

آن با الگوریتم‌های دیگر در این زمینه مقایسه خواهد شد. برای بررسی عملکرد الگوریتم پیشنهادی از پایگاه داده تصاویر رنگی Berkeley^{۱۴} [۳۰] استفاده شده است. الگوریتم پیشنهادی را می‌توان تا تعداد از پیش تعیین شده‌ای تکرار نمود و یا اینکه تعداد تکرارهای آنرا تا زمانی در نظر گرفت که تغییر ایجاد شده در کنش‌های انتخابی اتوماتای یادگیر سلولی از یک عدد کوچک کمتر باشد.

اتوماتای یادگیر مورد استفاده در سلول‌ها، از نوع L_{REP} با پارامترهای $a = 0.07$ و $b = 0.01$ می‌باشد که برای به‌هنگام‌سازی احتمال انتخاب کنش‌های خود از روابط (۱) و (۲) استفاده می‌کند. برای انحراف معیار فیلتر گوسی، مقدار ابتدایی آن مقدار ۱ انتخاب می‌شود و برای مقدار پایانی آن از مقدار ۰.۵ استفاده شده است. این فیلتر بعد از هر ۱۰ تا ۱۵ تکرار بر روی تصویر اعمال می‌شود. برای پارامتر T در این فیلتر اگر تعداد تکرارها مشخص باشد، از آن استفاده می‌شود در غیر این صورت از عدد ۱۰۰ استفاده می‌شود.

در شکل ۲۵ نتایج حاصل از اجرای الگوریتم را مشاهده می‌کنید. در شکل مزبور، تصاویر (الف) تصویر ورودی، تصاویر (ب) نتیجه قطعه‌بندی با الگوریتم k-means، تصاویر (ج) نتیجه قطعه‌بندی با الگوریتم JSeg [۲۹] و تصاویر (د) نتیجه قطعه‌بندی با روش پیشنهادی را نشان می‌دهد. نکته قابل ذکر این است که در تمامی موارد الگوریتم k-means با تعداد خوشه‌های مساوی با تعداد قطعه‌های خروجی الگوریتم پیشنهادی کنش قطعه‌بندی را انجام می‌دهد.

در شکل ۲۶ کارایی الگوریتم قطعه‌بندی مبتنی بر اتوماتای یادگیر سلولی برای یک تصویر نویزی نشان داده شده و با روش‌های قطعه‌بندی k-means و JSeg مقایسه شده است.

سطر اول نتیجه قطعه‌بندی تصویر بدون نویز را نشان می‌دهد. در سطرهای دوم و سوم به ترتیب نتیجه قطعه‌بندی را برای نویزهای از نوع نمک و لفل^{۱۵} و گوسی^{۱۶} نشان داده شده است. خروجی‌های حاصل شده نشان دهنده کارایی بالای روش قطعه‌بندی به کمک اتوماتای یادگیر سلولی است.

گذاشته می‌شود. برای اینکه قطعه‌بندی نهایی صورت بگیرد، نیاز به یک فاز ادغام می‌باشد. این فاز به این صورت انجام می‌شود که یک حد آستانه^{۱۲} برای فاصله بین پیکسل‌ها تعریف می‌شود و چنانچه فاصله پیکسل‌ها از این حد آستانه کمتر شود، به عنوان پیکسل‌های یک قطعه یکسان معرفی می‌شوند. در بررسی فاصله‌ها، برای هر قطعه یک نماینده معرفی می‌شود که از طریق میانگین‌گیری بر روی پیکسل‌های آن قطعه محاسبه می‌شود. در نهایت به جای هر کدام از پیکسل‌ها، پیکسل نماینده قطعه متناظر با آن جایگزین می‌شود تا تصویر قطعه‌بندی شده حاصل شود.

نکته‌ای که در قطعه‌بندی نهایی باید مورد توجه قرار گیرد، تعیین حد آستانه مناسب ادغام است. در سایر الگوریتم‌های قطعه‌بندی نیز یک پارامتر آزاد وجود دارد که تغییر در آن تعداد قطعات ایجاد شده را تغییر خواهد داد. به عنوان مثال در الگوریتم k-means در ابتدای اجرای الگوریتم باید تعداد خوشه‌ها را تعیین کرد و سپس با استفاده از آن الگوریتم را اجرا نمود. تعداد خوشه‌ها معادل با تعداد قطعات است. همچنین در الگوریتم JSEG [۲۹]، یک پارامتر به نام حد آستانه ادغام نواحی^{۱۳} وجود دارد که در بازه صفر تا یک تعریف می‌شود. این پارامتر بر روی خروجی الگوریتم تاثیرگذار است و با تغییر آن تعداد قطعات ایجاد شده از یک تا چند قطعه تغییر می‌کند. در الگوریتم پیشنهادی نیز تغییر در حد آستانه فاصله ادغام پیکسل‌ها، تعداد قطعات ایجاد شده را تغییر می‌دهد. هر چه این پارامتر کوچکتر باشد، تعداد قطعات ایجاد شده بیشتر خواهد شد. چنانچه این پارامتر بیشتر از حداکثر فاصله موجود بین پیکسل‌ها انتخاب شود، کل تصویر به عنوان یک قطعه انتخاب می‌شود. با کاهش در این مقدار تعداد قطعات افزایش می‌یابد.

۴-۱- نتایج پیاده‌سازی

در این بخش کارایی الگوریتم پیشنهادی برای قطعه‌بندی تصویر مورد ارزیابی قرار می‌گیرد. برای این منظور نتیجه الگوریتم روی چند تصویر نمونه ارزیابی و کارایی



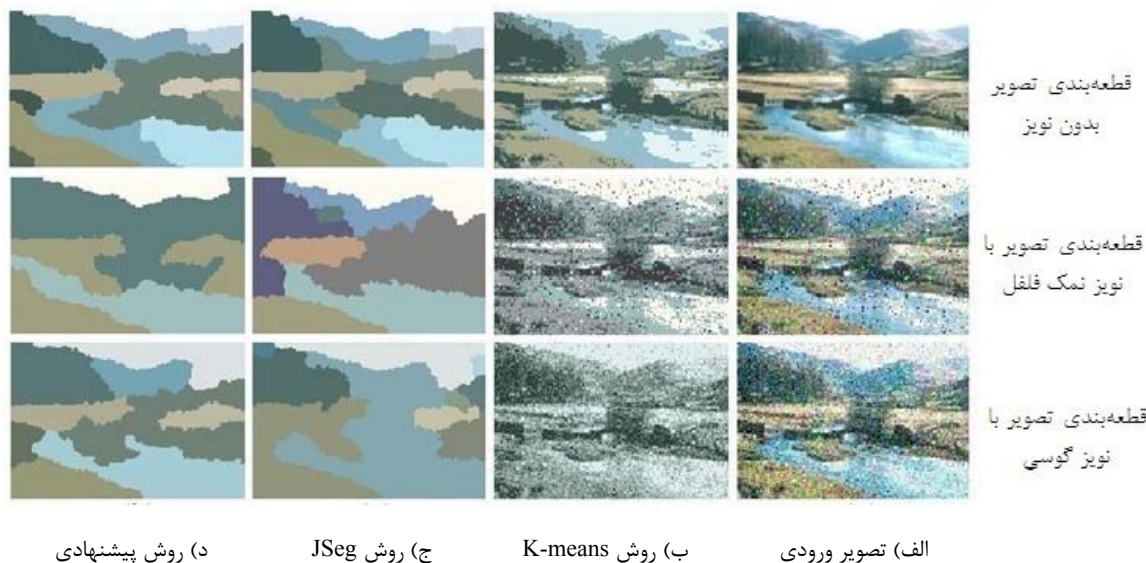
(د) روش پیشنهادی

(ج) روش JSeg

(ب) روش K-means

(الف) تصویر اصلی

شکل ۲۵- مقایسه کارایی روش قطعه‌بندی پیشنهادی با چند روش قطعه‌بندی متداول



شکل ۲۶- مقایسه کارایی روش قطعه‌بندی پیشنهادی با چند روش قطعه‌بندی متداول بر روی تصاویر نویزی

قطعه‌بندی مورد استفاده قرار گرفته است. اتوماتای یادگیر سلولی از یک الگوریتم تکراری برای رسیدن به یک حالت پایدار در امر قطعه‌بندی استفاده می‌کند.

۵- نتیجه‌گیری

شاخه‌های مختلف علم کامپیوتر در کنار هم می‌توانند نقاط ضعف و قدرت هم را پوشش داده و سبب ایجاد الگوریتم‌های بسیار کارآمد در حوزه‌های مختلف علوم شوند. مبحث پردازش تصویر هم که از شاخه‌های بسیار مهم علم کامپیوتر می‌باشد از این قاعده مستثنی نمی‌باشد. در این مبحث با استفاده از تکنیک‌های بسیار قدرتمند علوم دیگر مانند یادگیری ماشین و ... می‌توان قدرت الگوریتم‌های این حوزه را افزایش داد. یکی از شاخه‌های علم یادگیری ماشین که به علت ذات همسایگی سلولی و تاثیرپذیری همسایگی و همچنین سلولی بودن تصویر به مقدار زیادی در حوزه پردازش تصویر نفوذ کرده است، مبحث اتوماتاهای یادگیر سلولی می‌باشد که تاکنون در بسیاری از موارد مانند استخراج ویژگی‌ها، قطعه‌بندی، تجزیه و تحلیل، فشرده‌سازی و ... بسیار موفق بوده و توانسته سبب بهبود الگوریتم‌های معمول پردازش تصویر شود. به علت گستردگی دو شاخه پردازش تصویر و یادگیری ماشین می‌توان با ترکیب صحیح الگوریتم‌های یادگیری ماشین و پردازش تصویر قدرت الگوریتم‌های پایه در حوزه پردازش تصویر را به میزان قابل توجهی افزایش داد.

در این مقاله ابتدا اتوماتای یادگیر سلولی به همراه ویژگیها و خواص آن بررسی شده و سپس کاربرد آن در دو حوزه تشخیص لبه و قطعه‌بندی تصویر بررسی شده و برای هر یک از آنها یک روش جدید مبتنی بر اتوماتای یادگیر سلولی ارائه شده است. برای تشخیص لبه‌های تصویر یک آشکارساز وقتی لبه با استفاده از اتوماتای یادگیر سلولی ارائه شده است که در آن ابتدا با استفاده از تجزیه و تحلیل آماری تصویر و بررسی وجود لبه در نواحی مختلف تصویر، لبه‌های تصویر تقویت شده و نقاط غیر لبه تضعیف می‌شوند. این روش به عنوان یک مرحله پیش‌پردازش برای اتوماتای یادگیر سلولی می‌باشد. پس از آن از اتوماتای یادگیر سلولی برای تقویت و بهبود لبه‌های تقویت شده در مرحله پیش‌پردازش استفاده می‌شود. در ادامه یک الگوریتم برای قطعه‌بندی تصاویر رنگی به کمک اتوماتای یادگیر سلولی معرفی شده است. در بسیاری از الگوریتم‌های قطعه‌بندی موجود، برای یافتن نواحی تصویر یک سری معیار تعریف و سپس بر اساس این معیارها و تشابه پیکسل‌های واقع در یک همسایگی، تعلق هر کدام از پیکسل‌ها به یک دسته تعیین می‌شود. هر کدام از دو جزء بیان شده توسط یک اتوماتای یادگیر سلولی مدلسازیشده و در راستای

مراجع

[۱] م. ر. مبینی و ح. بیگی و م. طاهرخانی، "اتوماتای یادگیر سلولی و کاربردهای آن در پردازش تصویر"، مجموعه مقالات اولین کنفرانس ریاضیات و ارتباطات، مرکز تحقیقات مخابرات، ۱۳۷۹.

[۲] م. رفیع خوارزمی و م. ر. مبینی، "بازیابی تصاویر توسط اتوماتای یادگیر سلولی"، مجموعه مقالات دومین کنفرانس بینایی ماشین و پردازش تصاویر، ص. ۲۷۰-۲۶۱، ۱۳۸۱.

[۳] ا. طاهری‌نیا و ح. بیگی، "الگوگذاری تصاویر دیجیتال با استفاده از اتوماتای یادگیر سلولی"، مجموعه مقالات پنجمین کنفرانس بینایی ماشین و پردازش تصاویر، ۱۳۸۷.

[۴] م. رفیع خوارزمی و م. ر. مبینی، "قطعه بندی تصاویر توسط اتوماتای یادگیر سلولی"، مجموعه مقالات دهمین کنفرانس برق، ص. ۲۹۸-۳۰۶، ۱۳۸۱.

[۵] ا. آبین و ح. بیگی، "الگوریتم وقتی مبتنی بر اتوماتای یادگیر سلولی برای یافتن لبه‌های تصویر"، مجموعه مقالات سیزدهمین کنفرانس ملی انجمن کامپیوتر ایران، ۱۳۸۶.

[۶] م. ر. مبینی و ف. مهدی‌پور، "کاربرد اتوماتای یادگیر سلولی در جایابی مدارهای متراکم"، مجموعه مقالات هشتمین کنفرانس ملی انجمن کامپیوتر ایران، ص. ۲۰۳-۱۹۵، ۱۳۸۱.

[7] B. Oommen, and D. Croix, "Graph Partitioning Using Learning Automata," *IEEE Transactions on Computers*, vol. 45, no. 2, pp 195-208, 1996.

- [20] J. Shi, and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905, 2000.
- [21] Y. Deng, and B. S. Manjunath, "Unsupervised Segmentation of Color-Texture Regions in Images and Video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 800-810, 2001.
- [22] D. F. J. George, and S. E. George, "Cellular Automata Cryptography Using Reconfigurable Computing Source," *Proc. of the 16th International Conference on Developments in Applied Artificial Intelligence*, pp. 104-111, 2003.
- [23] N. H. Packard, and S. Wolfram, "Two-dimensional Cellular Automata," *Journal of Statistical Physics*, vol. 38, pp. 901-946, 1985.
- [24] Mitchell, *Computation in cellular automata: A selected review*, Technical report, Santa Fe Institute, Santa Fe, NM, USA, 1996.
- [25] M. Tsetlin, "On the Behavior of Finite Automata in Random Media," *Automation and Remote Control*, vol. 22, pp. 1345-1354, 1961.
- [26] K. S. Narendra, and M. A. L. Thathachar, *Learning Automata: An Introduction*, Prentice Hall, 1989.
- [27] H. Beigy, and M. R. Meybodi, "A Mathematical Framework for Cellular Learning Automata," *Advances in Complex Systems*, vol. 7, no. 3-4, pp. 295-320, 2004.
- [28] M. Agache, *Families of Estimator-Based Stochastic Learning Algorithms*, M.Sc. Thesis of Computer Science, School of Computer Science, Carleton University, Ottawa, Ontario, 2000.
- [29] Y. Deng, and B. S. Manjunath, "Unsupervised Segmentation of Color-Texture Regions in Images and Video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 800-810, 2001.
- [30] D. Martin, C. Fowlkes, D. Tal, J. Malik, "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," *Proc. of the 8th International Conference on Computer Vision*, pp. 416-423, 2001.
- [31] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679-698, 1986.
- [32] R. C. Gonzalez, and R. E. Woods, *Digital Image Processing*, Second Edition, Prentice-Hall, 2002.
- [8] B. Oommen, and T. Roberts, "Continuous Learning Automata Solutions to the Capacity Assignment Problem," *IEEE Transactions on Computers*, vol. 49, no. 6, pp 608-620, 2000.
- [9] M. R. Meybodi, and H. Beigy, "A Note on Learning Automata-Based Schemes for Adaptation of BP Parameters," *Journal of Neurocomputing*, vol. 48, no. 4, pp. 957-974, 2002.
- [10] A. A. Abin, M. Fotouhi, and S. Kasaei, "Skin Segmentation Based on Cellular Learning Automata," *Proc. of the Advances in Mobile Computing and Multimedia*, pp 254-259, 2008.
- [11] H. Beigy, and M. Meybodi, "Asynchronous Cellular Learning Automata," *Journal of Automatica*, vol. 44, no 5, pp 1350-1357, 2008.
- [12] M. R. Meybodi, and M. Khojasteh, "Application of Cellular Learning Automata in Modeling of Commerce Networks," *Proc. of the 6th Annual International Computer Society of Iran Computer Conference*, pp 284-295, 2001.
- [13] A. A. Abin, M. Fotouhi, and S. Kasaei, "A Dynamic Cellular Learning Automata-Based Skin Detector," *Multimedia System Journal*, vol. 15, no. 5, pp. 309-323, 2009.
- [14] M. R. Meybodi, and M. Taherkhani, "Application of Cellular Learning Automata in Modeling of Rumor Diffusion," *Proc. of the 9th Conference on Electrical Engineering, Power and Water Institute of Technology*, pp. 102-110, 2001.
- [15] R. Rastegar, M. Rahmati, and M. R. Meybodi, "A Clustering Algorithm using Cellular Learning Automata based Evolutionary Algorithm," *Proc. of the International Conference on Adaptive and Natural Computing Algorithms*, pp. 144-150, 2005.
- [16] Y. Delignon, A. Marzouki, and W. Pieczynski, "Estimation of Generalized Mixtures and its Application in Image Segmentation," *IEEE Transactions on Image Processing*, vol. 6, no. 10, pp. 1364-1376, 1997.
- [17] D. K. Panjwani, and G. Healey, "Markov Random Field Models for Unsupervised Segmentation of Textured Color Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 10, pp. 939-954, Oct. 1995.
- [18] J. P. Wang, "Stochastic Relaxation on Partitions with Connected Components and its Application to Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 6, pp. 619-636, 1998.
- [19] W. Y. Manjunath, and B. S. Ma, "Edge Flow: a Framework of Boundary Detection and Image Segmentation," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 744-749, 1997.



احمد علی آبین مدرک کارشناسی خود را در رشته مهندسی کامپیوتر گرایش نرم‌افزار از دانشگاه علم و صنعت ایران در سال ۱۳۸۴ اخذ نموده و در سال ۱۳۸۷ موفق به اخذ مدرک کارشناسی ارشد در رشته مهندسی کامپیوتر گرایش هوش مصنوعی از دانشگاه صنعتی شریف شده است. ایشان از سال

۱۳۸۹ مقطع دکتری خود را در دانشگاه صنعتی شریف آغاز نموده است و هم‌اکنون در آزمایشگاه سیستم‌های هوشمند دانشکده مهندسی کامپیوتر در حوزه یادگیری ماشین در حال پژوهش می‌باشد. زمینه‌های تحقیقاتی ایشان شامل یادگیری ماشین و پردازش تصویر می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

abin@ce.sharif.edu



سیدحمید امیری کارشناسی خود را در سال ۱۳۸۵ در دانشگاه شهید باهنر کرمان در رشته مهندسی کامپیوتر گرایش نرم‌افزار به پایان رسانده و در سال ۱۳۸۷ موفق به اخذ مدرک کارشناسی‌ارشد از دانشگاه صنعتی شریف در گرایش هوش مصنوعی شده است. وی هم‌اکنون به عنوان دانشجوی دکترا در آزمایشگاه پردازش تصویر و بینایی ماشین دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف مشغول به تحقیق می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

s_amiri@ce.sharif.edu



حمید بیگی تحصیلات خود را در مقاطع کارشناسی و کارشناسی‌ارشد مهندسی کامپیوتر به ترتیب در سال‌های ۱۳۷۱ و ۱۳۷۴ از دانشگاه شیراز و در مقطع دکتری مهندسی کامپیوتر در سال ۱۳۸۳ از دانشگاه صنعتی امیرکبیر به پایان رسانده است و هم‌اکنون دانشیار دانشکده مهندسی دانشگاه صنعتی شریف می‌باشد. زمینه‌های پژوهشی ایشان شامل الگوریتم‌های موازی، الگوریتم‌های یادگیری، هوش محاسباتی و کاربردهای آن در شبکه‌های کامپیوتری می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

beigy@ce.sharif.edu

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۹/۷/۲۱

تاریخ اصلاح: ۹۰/۵/۳

تاریخ قبول شدن: ۹۰/۵/۱۷

نویسنده مرتبط: دکتر حمید بیگی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران.

- ¹ Learning Automata
- ² Cellular Automata
- ³ Cellular Learning Automata
- ⁴ Clustering
- ⁵ Linear Reward Penalty
- ⁶ Linear Reward Epsilon Penalty
- ⁷ Linear Reward Inaction
- ⁸ General
- ⁹ Totalistic
- ¹⁰ Outer totalistic
- ¹¹ Gaussian Filter
- ¹² Threshold
- ¹³ Region Merge Threshold
- ¹⁴ The Berkeley Segmentation Dataset and Benchmark
- ¹⁵ Salt & Pepper
- ¹⁶ Gaussian

یک روش کارآمد برای محافظت از پیاده‌سازی سخت‌افزاری الگوریتم رمزنگاری AES در مقابل حمله تحلیل تفاضلی توان

مسعود معصومی^۱ سید مجتبی دهنوی^۲

^۱ دانشکده فنی، دانشگاه آزاد اسلامی واحد اسلامشهر، تهران، ایران
^۲ دانشکده علوم ریاضی و کامپیوتر، دانشگاه تربیت معلم، تهران، ایران

چکیده

مدل متداول و سنتی ارزیابی سیستم‌های رمزنگاری، امنیت را از منظر توابع ریاضی بکار رفته در آن مورد بررسی قرار می‌دهد. این روش آثار فیزیکی جانبی پیاده‌سازی و استفاده از این توابع در دنیای واقعی را در نظر نمی‌گیرد. یک مدل واقعی‌تر، امنیت ابزار رمزنگاری را از دید حملات کانال جانبی یا حملاتی که از اطلاعات مرتبط با پیاده‌سازی فیزیکی توابع رمزنگاری استفاده می‌کنند نیز مورد توجه قرار می‌دهد. حمله تحلیل تفاضلی توان نوع قدرتمند و منحصر بفردی از حملات کانال جانبی است که از توان مصرفی تراشه در حال رمز کردن اطلاعات برای شکستن الگوریتم رمز و بدست آوردن کلید آن استفاده می‌کند. در این مقاله روش جدید و کارآمدی برای محافظت از الگوریتم رمزنگاری AES پیشنهاد شده که قادر به افزایش قابل توجه مقاومت الگوریتم در مقابل این حمله با هزینه سخت‌افزاری بسیار پایین در مقایسه با سایر روش‌های گزارش شده تاکنون است. روش جدید مبتنی بر استفاده از ریاضیات میدان‌های مرکب است در حالیکه تنها باعث ۷٪ افزایش مساحت اشغالی روی تراشه خواهد شد بدون آنکه باعث کاهش فرکانس کاری یا تغییر در الگوریتم شده و خدشه‌ای به استاندارد بودن آن وارد شود. موثر بودن روش پیشنهادی با استفاده از نتایج عملی حاصل از پیاده‌سازی بر روی تراشه Xilinx Spartan-II FPGA تایید شده است.

کلمات کلیدی: سیستم رمزنگاری AES، حملات کانال جانبی، حمله تحلیل توان، پیاده‌سازی سخت‌افزاری، FPGA.

۱- مقدمه

خطر برای سیستم‌های رمزنگاری معاصر از ناحیه هدف قرار دادن نقاط ضعف پیاده‌سازی و استفاده از اطلاعات جانبی ناشی است و حملات آماری و الگوریتمی در درجه بعدی اهمیت قرار می‌گیرند. در نتیجه پیاده‌سازی صحیح یک الگوریتم رمز الزاما بمعنای امن بودن آن نیست [۱]. پس از انتشار مقاله P. Kocher در سال ۱۹۹۶ با عنوان "حمله تحلیل تفاضلی توان" انواع مختلفی از این گونه حملات پیشنهاد شده و توسعه یافته است [۲-۴]. حملات تحلیل توان نوع خاصی از حملات کانال جانبی هستند که از وابستگی توان مصرفی ابزار یا تراشه در حال رمز کردن اطلاعات به اطلاعات در حال پردازش و/یا عملیات در حال انجام برای شکستن الگوریتم و بازیابی کلید رمز استفاده می‌کنند. تحلیل توان خانواده‌ای

در مدل متداول ارزیابی سیستم‌های رمزنگاری سیستم امن به سیستمی اطلاق می‌شود که حتی در صورت دسترسی کامل به اطلاعات ردوبدل شده بین رمزنگار و رمزگشا نتوان کلید رمز را بازیابی کرد. در اواسط دهه نود نوع جدیدی از حملات رمزشکنی موسوم به حملات کانال جانبی^۱ معرفی شد که از اطلاعات ناشی از یک سخت‌افزار در حال پردازش اطلاعات نظیر تشعشعات الکترومغناطیسی یا توان مصرفی آن برای شکستن رمز و استخراج کلید استفاده می‌کند. این گونه حملات بخاطر سادگی و موثر بودن بسرعت گسترش پیدا کردند تا آنجا که اکنون مهمترین

تنها ۷٪ بیشتر از نسخه محافظت نشده مساحت اشغال می‌کند ضمن آنکه باعث کاهش سرعت عملیات رمز نیز نخواهد شد. علاوه بر آن، بر خلاف بسیاری از روش‌های گزارش شده تاکنون، روش پیشنهادی در این مقاله منحصر به پیاده‌سازی سخت‌افزاری یا FPGA نبوده و قابل بکارگیری در پیاده‌سازی‌های نرم‌افزاری و نیز بستری نظیر کارتهای هوشمند، پردازشگرهای سیگنال دیجیتال و ... می‌باشد. مهمتر اینکه نتایج بدست آمده در این مقاله بروشنی نشان داد که بر خلاف روش‌های پیشنهادی تاکنون که عمدتاً بدنیاال مقابله با حمله با وارد کردن یک عامل تصادفی خارجی یا استفاده از یک سخت‌افزار جانبی هستند می‌توان الگوریتم‌هایی طراحی کرد که بطور ذاتی در مقابل حمله تحلیل توان مقاوم باشند. قابل ذکر است که عمده مقالات ارائه شده در این زمینه بخاطر سادگی پیاده‌سازی حمله، کارت‌های هوشمند و ریزپردازنده‌ها را هدف قرار گرفته و مقالات کمی در مورد ارزیابی امنیتی FPGA در مقابل حمله تحلیل توان وجود دارد [۱، ۱۵]. در ادامه مقاله ابتدا در بخش دوم کاربرد ریاضیات میدانهای مرکب در پیاده‌سازی موثر الگوریتم AES را تشریح می‌کنیم. در بخش سوم به تشریح نحوه پیاده‌سازی عملیات پایه ریاضی در میدان‌های مرکب خواهیم پرداخت. در بخش چهارم ایده جدید را ارائه می‌دهیم. بخش پنجم مقاله به نحوه پیاده‌سازی عملی و آزمایشگاهی حمله، تنظیمات دستگاههای بکار رفته و نتایج بدست آمده اختصاص داده شده است. در نهایت به بررسی برخی چالشهای موجود در مقابله با حملات تحلیل توان و جمع بندی کلی موضوع پرداخته و نتایج نهایی را ارائه می‌دهیم.

۲- استفاده از ریاضیات میدان‌های مرکب برای پیاده‌سازی AES

در پیاده‌سازی AES Rijndael تقریباً نیمی از تاخیر هر دور هنگام انجام تبدیل جانشینی بایت‌ها و مربوط به تاخیر SBoxهایی است که بصورت LUT^۹ پیاده‌سازی شده‌اند. تاخیر غیر قابل کاهش LUTها عاملی است که مانع از افزایش سرعت مدار خواهد شد [۱۶]. راه‌های دیگری نیز برای پیاده‌سازی SBoxها بصورت سخت‌افزاری و بدون نیاز به LUT وجود دارد که در صورت استفاده از آنها می‌توان هر دور را با استفاده از Subpipelinig به تعدادی زیر دور تقسیم کرده و سرعت مدار را افزایش داد. از آنجا که SBox الگوریتم Rijndael شامل معکوس ضربی عناصر روی میدان $GF(2^8)$ است راه‌های مختلفی برای پیاده‌سازی معکوس ضربی بدون نیاز به LUT و افزایش سرعت مدار مانند استفاده از الگوریتم گسترش یافته اقلیدس^{۱۰} یا استفاده از قضیه فرمت^{۱۱} پیشنهاد شده است. اما یکی از موثرترین این راه‌ها استفاده از میدان‌های مرکب^{۱۲} برای محاسبه و پیاده‌سازی معکوس ضربی عناصر روی میدان محدود یا پیاده‌سازی مدار معادل SBox است [۱۷، ۲۱]. با استفاده از میدان‌های مرکب می‌توان پیچیدگی سخت‌افزاری پیاده‌سازی را کاهش داد. دو زوج

$$\{GF(2^n), Q(y) = y^n + \sum_{i=0}^{n-1} q_i y^i, q_i \in GF(2)\}$$

$$\{GF((2^n)^m), P(x) = x^m + \sum_{i=0}^{m-1} P_i x^i, P_i \in GF(2^n)\}$$

را یک میدان مرکب می‌گوییم اگر:

$$1- GF(2^n) \text{ از } GF(2) \text{ بوسیله } Q(y) \text{ ساخته شود.}$$

$$2- GF((2^n)^m) \text{ از } GF(2^n) \text{ بوسیله } P(x) \text{ ساخته شود.}$$

یک میدان مرکب با $GF((2^n)^m)$ نشان داده می‌شود و با میدان $GF(2^k)$ با $k = mn$ یکرخت^{۱۳} است. لذا یک میدان مرکب را می‌توان بصورت تکراری^{۱۴} از

قدرتمند و منحصر بفرد از حملات رمزشکنی سخت‌افزاری است. بدلیل ماهیت این نوع حملات جلوگیری از آنها بسادگی میسر نیست ضمن آنکه نیاز به تجهیزات پیچیده و گران قیمت برای پیاده‌سازی حمله نداشته و بسادگی و در زمان کوتاه قادر به شکستن رمز هستند. حملات تحلیل توان عمدتاً به دو دسته حملات ساده و تفاضلی تقسیم‌بندی می‌شوند. حمله آنالیز توان ساده^۲ مستقیماً از اندازه‌گیری‌های توان مصرفی وسیله حین انجام عملیات رمزنگاری استفاده می‌کند. الگوی توان یا انرژی مصرفی سخت‌افزار می‌تواند اطلاعات مهمی در مورد دستورالعمل‌ها، توالی اجرای آنها و حتی عملوند در اختیار هکر قرار بدهد. حمله تحلیل تفاضلی توان^۳ (DPA) بمراتب موثرتر از حمله تحلیل توان ساده است و در واقع نوعی آزمایش آماری است که کلید رمز با انجام آزمایش‌های آماری بر روی صدها و شاید هزاران داده نمونه و اندازه‌گیری توان مصرفی متناظر با آنها حدس زده می‌شود. مقاومت در برابر این حمله بسادگی میسر نیست زیرا حتی در صورت افزودن نویز به سیستم، نویز اضافه شده به مصرف توان با افزایش تعداد نمونه‌ها و متوسط‌گیری از آنها فیلتر می‌شود. علاوه بر آن نوع قدرتمند دیگری از این حملات موسوم به تحلیل همبستگی توان^۴ نیز بطور گسترده‌ای مورد مطالعه و استفاده قرار گرفته است [۱، ۳]. در سال‌های اخیر امنیت الگوریتم استاندارد پیشرفته رمزنگاری (AES) در مقابل حمله تحلیل توان مورد توجه ویژه قرار گرفته است زیرا این الگوریتم از جمله مهم‌ترین و پرکاربردترین الگوریتم‌های رمزنگاری معاصر است که در کاربردهای متنوعی از جمله کارت‌های هوشمند وب سرورها تلفن‌های سلولی شبکه‌های ATM و ... مورد استفاده قرار می‌گیرد [۵، ۶]. در نتیجه تحقیقات و مطالعات انجام شده انواع مختلفی از این حملات و نیز روش‌های مقابله با آنها پیشنهاد شده است. از جمله مهم‌ترین روش‌های مقابله می‌توان به روش‌های تصادفی سازی مانند پالس ساعت تصادفی شده [۷]، رمزکردن حافظه، تصادفی کردن توان مصرفی و از بین بردن همبستگی بین توان مصرفی تراشه و منبع تغذیه خارجی اشاره نمود. علاوه بر آن روش‌های دیگری از قبیل پیاده‌سازی با منطق مکمل^۵، منطق SABL^۶، و منطق غیر همزمان^۷ نیز می‌توان نام برد. [۸، ۹، ۱۰]. متأسفانه غالب این تکنیک‌ها غیر کارآمد و غیر موثر و یا غیر مقاوم در مقابل حملات تحلیل توان از مرتبه بالاتر هستند. بعنوان مثال تکنیک‌های پیشنهاد شده در [۱۱] تقریباً دو برابر مساحت یک پیاده‌سازی محافظت نشده سطح اشغال می‌کنند و یا دو برابر آن توان مصرفی می‌کنند و تکنیک پیشنهادی در [۱۰] سه برابر مساحت اشغال کرده و بازدهی را تا یک چهارم تقلیل می‌دهد. روش شناخته شده دیگر مقابله ماسک گذاری بمفهوم استفاده از اعداد تصادفی برای ماسک کردن داده‌های ورودی و میانی برای شخص کنجکاو است. مشکل اساسی این روش نیز آن است که نیاز به یک مسیر داده موازی جداگانه برای ماسک گذاری و ماسک برداری است که باعث افزایش قابل ملاحظه مساحت اشغالی روی تراشه و کاهش بازدهی خواهد شد ضمن آنکه تضمین کننده امنیت کامل پیاده‌سازی در مقابل حمله تحلیل توان تفاضلی نیز نیست [۱۲، ۱۳، ۱۴]. در این مقاله روش ساده جدید و کارآمدی برای خنثی کردن تحلیل توان تفاضلی و نیز تحلیل همبستگی توان پیشنهاد شده که از خصوصیات ریاضی الگوریتم AES استفاده می‌کند بدون آنکه تغییری در الگوریتم ایجاد کرده یا خدشه‌ای به استاندارد بودن آن وارد سازد. روش مزبور مبتنی بر نوعی تصادفی سازی در ریاضیات میدانهای مرکب^۸ است و نیازی به استفاده از عامل خارجی برای مقابله با حمله ندارد. بمنظور ارزیابی میزان موثر بودن روش پیشنهادی دو نسخه محافظت نشده و محافظت شده با روش جدید AES را بر روی تراشه Xilinx Spartan-II FPGA پیاده‌سازی کرده و نتایج را از حیث میزان مقاوم بودن در مقابل حمله و نیز کارایی پیاده‌سازی مورد ارزیابی قرار دادیم. نتایج بدست آمده نشان داد که نسخه محافظت شده علاوه بر فراهم آوردن امنیت بالا در مقابل حمله تحلیل توان

اگر q یک عنصر از میدان $GF(2^8)$ باشد آنگاه نگاشت یکرخست^{۱۵} و معکوس آن بفرم ضرب ماتریسی $\delta^{-1} * q$ و $\delta * q$ با ارزش‌ترین بیت و q_0 کم‌ارزش‌ترین بیت باشند این دو ضرب ماتریسی را می‌توان با استفاده از گیت‌های XOR بشکل زیر پیاده‌سازی نمود.

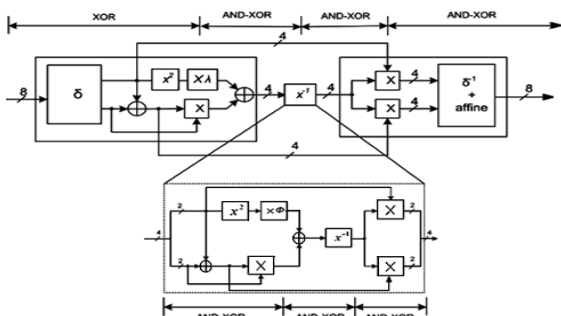
$$\delta \times q = \begin{bmatrix} q_7 \oplus q_5 \\ q_7 \oplus q_6 \oplus q_4 \oplus q_3 \oplus q_2 \oplus q_1 \\ q_7 \oplus q_5 \oplus q_3 \oplus q_2 \\ q_7 \oplus q_5 \oplus q_3 \oplus q_2 \oplus q_1 \\ q_7 \oplus q_6 \oplus q_2 \oplus q_1 \\ q_7 \oplus q_4 \oplus q_3 \oplus q_2 \oplus q_1 \\ q_6 \oplus q_4 \oplus q_1 \\ q_6 \oplus q_1 \oplus q_0 \end{bmatrix}$$

$$\delta^{-1} \times q = \begin{bmatrix} q_7 \oplus q_6 \oplus q_5 \oplus q_1 \\ q_6 \oplus q_2 \\ q_6 \oplus q_5 \oplus q_1 \\ q_6 \oplus q_5 \oplus q_4 \oplus q_2 \oplus q_1 \\ q_5 \oplus q_4 \oplus q_3 \oplus q_2 \oplus q_1 \\ q_7 \oplus q_4 \oplus q_3 \oplus q_2 \oplus q_1 \\ q_5 \oplus q_4 \\ q_6 \oplus q_5 \oplus q_4 \oplus q_2 \oplus q_0 \end{bmatrix}$$

در میدان مرکب $GF((2^4)^2)$ هر عنصر را می‌توان بصورت $s_h x + s_l$ نشان داد جاییکه s_h و s_l عضو $GF(2^4)$ هستند و x نیز ریشه $P_2(x)$ است. طبق الگوریتم توسعه یافته اقلیدس معکوس عنصر $s_h x + s_l$ در هنگ $P_2(x)$ را می‌توان بصورت زیر محاسبه نمود:

$$(s_h x + s_l)^{-1} = s_h \theta x + (s_h + s_l) \theta \quad (2)$$

که در آن $\theta = (s_h^2 \lambda + s_h s_l + s_l^2)^{-1}$ طبق رابطه فوق معکوس ضربی هر عنصر روی $GF(2^8)$ را می‌توان با استفاده از معماری نشان داده شده در شکل ۲ روی میدان مرکب $GF((2^4)^2)$ بدست آورد. ضرب‌کننده‌های در $GF(2^4)$ را می‌توان به ضرب‌کننده‌های روی $GF((2^2)^2)$ و سپس $GF(2)$ تجزیه^{۱۶} نمود که عمل ضرب در $GF(2)$ براحتی و با گیت AND قابل انجام است. شکل ۳ نحوه تجزیه ضرب‌کننده روی $GF(2^4)$ را نشان می‌دهد و چنانچه از شکل پیداست این ضرب‌کننده را می‌توان با ۲۱ گیت XOR و ۹ گیت AND پیاده‌سازی نمود در حالیکه ۴ گیت XOR و یک گیت AND در مسیر بحرانی آن وجود دارد، اما معکوس کننده ضربی روی $GF(2^4)$ را نشان نمی‌دهد [۱۶].



شکل ۲- پیاده‌سازی تبدیل جانشینی بایت‌ها با استفاده از میدان مرکب [۱۶]

میدان‌های مرتبه پایین‌تر ساخت و محاسبات روی میدان اصلی را روی میدان‌های مرتبه پایین‌تر انجام داده و هزینه سخت‌افزاری را کاهش داد. برای مثال می‌توان میدان مرکب $GF(2^8)$ را از میدان $GF(2)$ با استفاده از چند جمله‌ای‌های زیرساخت.

$$\begin{cases} GF(2) \Rightarrow GF(2^2): & P_0(X) = X^2 + X + 1 \\ GF(2^2) \Rightarrow GF(2^2)^2: & P_1(X) = X^2 + X + \phi \\ GF(2^2)^2 \Rightarrow GF(2^2)^2)^2: & P_2(X) = X^2 + X + \lambda \end{cases} \quad (1)$$

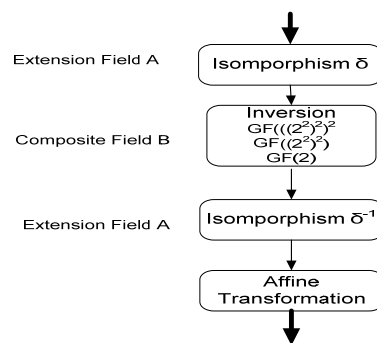
که $\Phi = \{10\}_2$ و $\lambda = \{1100\}_2$ البته برای آنکه عناصر $GF(2^8)$ دقیقاً بر روی عناصر متناظر خود در میدان مرکب تصویر شوند یک تبدیل $x \times f(x) = \delta$ یک تبدیل خطی affine نیز لازم است. δ یک ماتریس 8×8 متناظر با $m(x) = x^8 + x^4 + x^3 + x + 1$ است و می‌تواند بفرم زیر باشد.

$$\delta = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

برای بازگشت از میدان مرکب به میدان $GF(2^8)$ به عکس تبدیل فوق و δ^{-1} نیاز داریم. محاسبه معکوس عناصر در $GF(2^8)$ که عملیات اصلی در تبدیل جانشینی بایت‌ها و نیز معکوس آن است بیشترین سخت‌افزار را در پیاده‌سازی هر دور الگوریتم AES مصرف می‌کند. ماتریس δ^{-1} متناظر با ماتریس δ فوق می‌تواند بشکل زیر باشد.

$$\delta^{-1} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

شکل ۱ نشان دهنده انجام عملیات محاسبه معکوس بر روی میدان $GF(2^8)$ با استفاده از ریاضیات میدان‌های مرکب است.



شکل ۱- انجام عملیات محاسبه معکوس بر روی میدان $GF(2^8)$ با استفاده از ریاضیات میدان‌های مرکب

$$\begin{aligned} k_H &= (q_3q_2)(11)_2 + (q_1q_0)(11)_2 \\ k_H &= (q_3x+q_2)(x+1) + (q_1x+q_0)(x+1) \\ k_H &= q_3x^2 + (q_3+q_2)x + q_2 + q_1x^2 + (q_1+q_0)x + q_0 \end{aligned} \quad (3)$$

جایگزینی $x^2 = x + 1$ منجر به نتیجه زیر خواهد شد.

$$\begin{aligned} k_H &= q_3(x+1) + (q_3+q_2)x + q_2 + q_1(x+1) + (q_1+q_0)x + q_0 \\ k_H &= (q_3+q_3+q_2+q_1+q_1+q_0)x + (q_3+q_2+q_1+q_0) \\ k_3x + k_2 &= (q_2+q_0)x + (q_3+q_2+q_1+q_0) \in GF(2) \end{aligned}$$

همین روال را می‌توان برای تجزیه k_L روی میدان $GF(2)$ بکار برد.

$$\begin{aligned} k_L &= q_H \lambda_H \phi \\ k_L &= (q_3q_2)(11)_2 (10)_2 \\ k_L &= (q_3x+q_2)(x+1)x \\ k_L &= q_3x^3 + q_2x^2 + q_1x + q_0 \end{aligned}$$

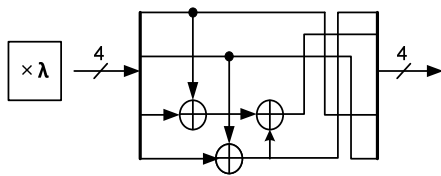
مجدداً $x^2 = x + 1$ را می‌توان طبق معادله جایگزین کرده و بطریق مشابه نیز به این نتیجه می‌رسیم که $x^3 = 1$ لذا خواهیم داشت.

$$\begin{aligned} k_L &= q_3(1) + q_2(x+1) + q_3(x+1) + q_2x \\ k_L &= (q_3+q_2+q_1)x + (q_3+q_2+q_1) \\ k_1x + k_0 &= (q_3)x + (q_2) \in GF(2) \end{aligned} \quad (4)$$

با ترکیب دو رابطه (۳) و (۴) می‌توان به منطق لازم برای پیاده‌سازی ضرب در مقدار ثابت λ رسید.

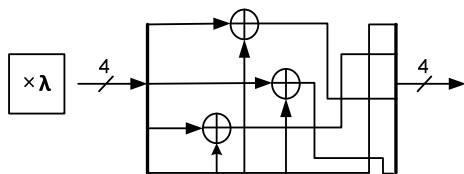
$$\begin{aligned} k_3 &= q_2 \oplus q_0 \\ k_2 &= q_3 \oplus q_2 \oplus q_1 \oplus q_0 \\ k_1 &= q_3 \\ k_0 &= q_2 \end{aligned} \quad (5)$$

با استفاده از معادله (۵) می‌توان مدار شکل ۴ را برای پیاده‌سازی سخت‌افزاری بلوک ضرب در مقدار ثابت $\lambda = \{1100\}_2$ در نظر گرفت که در این شکل مقدار λ است.

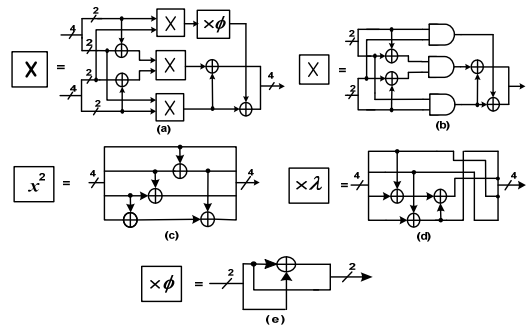


شکل ۴- پیاده‌سازی سخت‌افزاری بلوک ضرب در مقدار ثابت $\lambda = 12$

همین روال را می‌توان برای پیاده‌سازی ضرب در مقدار ثابت $\lambda = \{1111\}_2$ در نظر گرفت و به مدار شکل ۵ رسید.



شکل ۵- پیاده‌سازی سخت‌افزاری بلوک ضرب در مقدار ثابت $\lambda = 15$



شکل ۳- پیاده‌سازی بلوک‌های شکل ۴- (a) ضرب‌کننده در $GF(2^4)$, (b) ضرب‌کننده در $GF(2^2)$, (c) مربع‌کننده در $GF(2^4)$; (d) و (e) ضرب‌کننده‌های ثابت [۱۶].

۳- پیاده‌سازی سخت‌افزاری عملیات ریاضی در میدان مرکب

از آنجا که برای درک و پیاده‌سازی ایده پیشنهادی نیاز به درک نحوه پیاده‌سازی برخی از عملیات ریاضی مانند محاسبه معکوس، جمع، ضرب و مربع در میدان‌های مرکب داریم لذا در این بخش بطور مختصر نحوه تحقق چنین عملیاتی را شرح می‌دهیم. هر چند جمله‌ای دلخواه را می‌توان بصورت $bx + c$ نشان داد جاییکه b نیمه بالا و c نیمه پایین را نشان می‌دهد. بنابراین هر عدد باینری q در میدان گالوا را می‌توان به دو جز تقسیم نمود و به شکل $q_Hx + q_L$ نشان داد [۲۱]. برای مثال $q = \{1011\}_2$ را می‌توان بصورت $\{11\}_2 + \{10\}_2x$ نشان داد جاییکه $q_H = \{10\}_2$ و $q_L = \{11\}_2$ است. متعاقباً می‌توان q_H و q_L را بترتیب بصورت $\{0\}_2 + \{1\}_2x$ و $\{1\}_2 + \{1\}_2x$ تجزیه کرد. با استفاده از این ایده می‌توان عملیات منطقی برای پیاده‌سازی بلوک‌های شکل ۲ را تعیین کرده و بدست آورد. تشریح جزئیات پیاده‌سازی تمام بلوک‌های شکل ۲ خارج از موضوع این مقاله است ولی برای درک بهتر مساله و روشن شدن موضوع به ذکر نحوه پیاده‌سازی دو بلوک ضرب در مقادیر ثابت $(\times\lambda)$ و $(\times\phi)$ می‌پردازیم. سایر بلوک‌ها نیز بطریق مشابه قابل پیاده‌سازی هستند. برای سادگی، مقداری از عمق ریاضی مساله کاسته شده تا خواننده بهتر با عملیات انجام شده در درون تبدیل جانشینی بایت‌ها آشنا شود.

فرض کنیم $k = \{k_3 k_2 k_1 k_0\}_2$ باشد جاییکه $k = q\lambda$ و $k = \{k_3 k_2 k_1 k_0\}_2$ و $q = \{q_3 q_2 q_1 q_0\}_2$ عناصر $GF(2^4)$ هستند.

$$k = \{k_3 k_2 k_1 k_0\} = k_Hx + k_L = \{q_3 q_2 q_1 q_0\}(1100)_2$$

جاییکه $\lambda_H = \{00\}_2$ و $\lambda_L = \{11\}_2$

$$k = (q_Hx + q_L)(\lambda_Hx + \lambda_L)$$

از آنجا که $\lambda_L = \{00\}_2$ است می‌توان از آن صرف‌نظر کرد.

$$k = q_H \lambda_H x^2 + q_L \lambda_H x$$

با جایگزینی $x^2 = x + \phi$ از معادله (۲) خواهیم داشت.

$$\begin{aligned} k &= q_H \lambda_H (x + \phi) + q_L \lambda_H x \\ k &= (q_H \lambda_H + q_L \lambda_H)x + (q_H \lambda_H) \phi \in GF(2^2) \end{aligned}$$

k_H و k_L را می‌توان به $GF(2)$ برد.

$$k_H = q_H \lambda_H + q_L \lambda_H$$

تبدیل جانشینی بابت‌ها قابل ترجمه به گیت های منطقی است و هزینه سخت افزاری چندانی را به پیاده ساز تحمیل نمی‌کند.

۴- ایده جدید

تحلیل توان برای بازیابی کلید از وابستگی توان مصرفی ابزار رمزنگاری به مقادیر میانی در حین اجرای الگوریتم استفاده می‌کند. بنابراین برای جلوگیری و ناکام گذاشتن حمله این وابستگی باید تا حد ممکن از بین برود. همانگونه که گفته شد چالش اصلی در مقابله با حمله تحلیل توان در ماسک کردن تبدیل جانشینی بابت هاست جائیکه هر عنصر با معکوس ضربی خود در میدان $GF(2^8)$ جایگزین می‌شود. با در نظر گرفتن این حقیقت و با مطالعه ساختار ریاضی الگوریتم Rijndael می‌توان دریافت که سه پارامتر $\{\Phi, \lambda, \delta\}$ منحصر بفرد نیستند زیرا تعداد بسیار بیشتری از یک یکرختی روی میدان $GF(2^8)$ وجود دارد. دو تبدیل یکرخت نامیده می‌شوند چنانچه هر دو بازای یک ورودی مشخص خروجی یکسانی تولید کنند. چند جمله ای $P_1(x) = x^2 + x + \Phi$ روی میدان $GF(2^2)$ تحویل ناپذیر است اگر و فقط اگر $1 < \Phi < 4$. همچنین نیز روی $GF(2^4)$ تحویل ناپذیر است اگر و فقط اگر $7 < \lambda < 16$ باشد. همچنین تنها یک چند جمله ای تحویل ناپذیر روی $GF(2)$ وجود دارد که همان $P_2(x) = x^2 + x + 1$ است.

بنابراین شانزده ترکیب مختلف برای $\{\Phi, \lambda\}$ وجود دارد که همگی منجر به جواب یکسانی خواهند شد. همچنین با جستجو در مجموعه تبدیلات خطی از $GF(2^8)$ بر روی خودش یا تمام خود یکرختی‌ها^{۱۷} بر روی $GF(2^8)$ اثبات کرد که تعداد $\prod_{i=0}^{2^8-1} (2^8 - 2^i)$ از چنین تبدیلاتی و به همین تعداد ماتریس تبدیل δ/δ^{-1} وجود دارد [۲۳، ۲۲]. طبیعتاً تعداد قابل ملاحظه‌ای یکرختی بر مبنای مجموعه‌های $\{\Phi, \lambda, \delta, \delta^{-1}\}$ برای Rijndael وجود خواهد داشت که می‌توان برخی از آنها را انتخاب کرد بدون آنکه باعث تغییر در ساختار ریاضی یا خروجی تبدیل جانشینی بابت‌ها بشود. لازم به ذکر است که تمام انتخاب‌های ممکن الزاماً منجر به یکرختی میدانی مناسب برای Rijndael نخواهد شد. ما برای پیاده‌سازی ایده پیشنهادی ۳۲ مجموعه مختلف از $\{\Phi, \lambda, \delta, \delta^{-1}\}$ را که قابل پیاده‌سازی با حداقل گیت‌های منطقی هستند را انتخاب کردیم. بعنوان مثال سه مجموعه زیر چنین ویژگی دارند. برای سادگی مقادیر ماتریس‌های δ و δ^{-1} و نیز Φ و λ بصورت دسیمال نشان داده شده‌اند.

$$\Phi = 2, \lambda = 15$$

$$\delta = \{160, 126, 114, 162, 182, 84, 16, 217\}^T$$

$$\delta^{-1} = \{46, 28, 174, 2, 122, 26, 144, 75\}^T$$

$$\Phi = 3, \lambda = 12$$

$$\delta = \{160, 222, 172, 174, 202, 238, 44, 227\}^T$$

$$\delta^{-1} = \{102, 212, 230, 162, 10, 234, 176, 233\}^T$$

$$\Phi = 3, \lambda = 10$$

$$\delta = \{160, 126, 172, 2, 20, 132, 130, 99\}^T$$

$$\delta^{-1} = \{190, 132, 62, 106, 98, 2, 112, 141\}^T$$

ایده پیشنهادی تصادفی‌سازی در یکرختی تبدیل جانشینی بابت‌هاست به اینصورت که در ابتدا تعداد مشخصی مثلاً ۳۲ مجموعه از $\{\Phi, \lambda, \delta, \delta^{-1}\}$ تعیین شده و انتخاب می‌شوند. ۳۲ مجموعه SBox بر مبنای این ۳۲ مجموعه مقادیر مطابق با آنچه در شکل ۲ نشان داده شد تولید کرده و ذخیره می‌کنیم. عبارات دیگر ۳۲ مجموعه SBox با ورودی - خروجی مشابه ولی ساختار درونی متفاوت خواهیم داشت. در هنگام رمز کردن هر قالب ۱۲۸ بیتی داده یا حتی برای هر بابت داده یکی از این ساختارها برای رمزنگاری توسط یک شمارنده اعداد شبه تصادفی

یا بعنوان مثال دیگر ضرب در مقدار ثابت $(\times \Phi)$ روی میدان $GF(2^2)$ را در نظر می‌گیریم جائیکه $\Phi = \{10\}_2$. فرض کنیم $k = q\Phi$ باشد جائیکه $k = \{k_1 k_0\}_2$ می‌گیریم $q = \{q_1 q_0\}_2$ و $\Phi = \{10\}_2$ عناصری از $GF(2^2)$ هستند.

$$k = k_1 x + k_0 = (q_1 q_0)(10)_2 = (q_1 x + q_0)(x)$$

$$k = q_1 x^2 + q_0 x$$

جایگزینی x^2 با $x+1$ به نتیجه زیر منجر خواهد شد.

$$k = q_1(x+1) + q_0 x$$

$$k = (q_1 + q_0)x + (q_1) \in GF(2)$$

بنابراین منطق مورد نیاز برای پیاده سازی ضرب در ثابت $\Phi = \{10\}_2$ برابر خواهد بود با:

$$k_1 = q_1 + q_0$$

$$k_0 = q_1$$

(۶)

بنابراین مدار پیاده سازی ضرب در ثابت $\Phi = \{10\}_2$ بشکل ۶ خواهد بود.

چنانچه $\Phi = \{11\}_2$ باشد سخت‌افزار مورد نیاز برای ضرب در مقدار ثابت Φ

می‌توان بصورت زیر بدست آورد.

$$k = k_1 x + k_0 = (q_1 q_0)(11)_2 = (q_1 x + q_0)(x+1)$$

$$k = q_1 x^2 + (q_0 + q_1)x + q_0$$

جایگزینی x^2 با $x+1$ به نتیجه زیر منجر خواهد شد.

$$k = q_1(x+1) + q_0 x$$

$$k = (q_1 + q_0)x + (q_1) \in GF(2)$$

بنابراین مدار پیاده سازی ضرب در ثابت $\Phi = \{11\}_2$ در میدان $GF(2^2)$ برابر

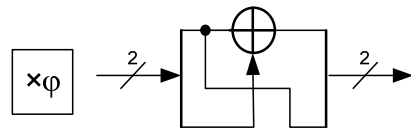
خواهد بود با:

$$k_1 = q_0$$

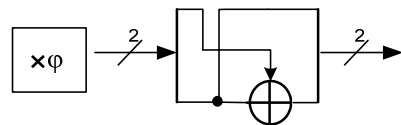
$$k_0 = q_0 + q_1$$

(۷)

و مدار منطقی لازم برای تحقق این پیاده‌سازی بصورت شکل ۷ خواهد بود.



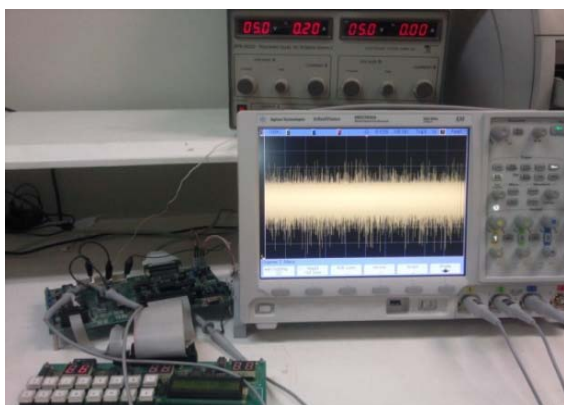
شکل ۶- پیاده‌سازی ضرب در مقدار ثابت $\Phi = \{10\}_2$ در میدان $GF(2^2)$



شکل ۷- پیاده‌سازی ضرب در مقدار ثابت $\Phi = \{11\}_2$ در میدان $GF(2^2)$

با جمع‌بندی مطالب گفته شده در بخش ۲ و ۳ و نیز با توجه به معماری شکل ۲ می‌توان چنین نتیجه گرفت که هر نوع تغییر در مقادیر مجموعه $\{\Phi, \lambda, \delta, \delta^{-1}\}$ یا عبارات دیگر هر نوع تغییر بر این مبنا در معماری ساختار

نیز افزایش دقت حمله فرکانس کار تراشه را تا ۱۰ کیلو هرتز کاهش داده، اندازه گیری‌ها را ده بار تکرار کرده و از نمونه‌های بدست آمده متوسط گرفتیم.



شکل ۸- تجهیزات آزمایشگاهی بکار گرفته شده برای پیاده‌سازی حمله

در فرم اولیه حمله تحلیل تفاضلی توان به AES، یک تابع انتخاب D بعنوان تابع محاسبه کننده مقدار یک بیت b که جزئی از یک بردار میانی S_1 است در نظر گرفته می‌شود. می‌توان b را بشکل زیر محاسبه کرد.

$$b = \text{one output bit of } S_1 (P_1 \oplus K_1) \quad (8)$$

در معادله (۸)، S_1 نشان دهنده تبدیل جانشینی بیت‌ها در دور اول، P_1 معرف i امین متن آشکار تصادفی و K_1 اولین زیر کلید است. بمنظور افزایش نسبت سیگنال به نویز و نیز دقت حمله بجای در نظر گرفتن یک بیت از خروجی تبدیل جانشینی بیت‌ها، یک مجموعه چهار بیتی مورد بررسی قرار گرفت بدین مفهوم که تابع انتخاب هنگامی که وزن همینگ خروجی بیش از چهار باشد مقدار یک و در غیر اینصورت مقدار صفر را برمیگرداند. مرجع [۲۴] نشان داده است که به این ترتیب میزان قله‌های ثانویه^{۱۸} و قله‌های شبح^{۱۹} در منحنی‌های تفاضل توان کاهش می‌یابند.

نتایج آزمایشات انجام شده برای مشاهدات تفاضل توان برای کلیدهای درست و غلط متناظراً در شکل‌های ۹ و ۱۰ نشان داده شده است. آزمایشات بروشنی قابل اندازه‌گیری بودن وزن همینگ خروجی تبدیل جانشینی بیت‌ها را نشان داد. بازیابی بیت اول کلید نیاز به تقریباً ۱۰۰۰ اندازه‌گیری توان بازای ۱۰۰۰ زوج متن آشکار- متن رمز داشت. مقدار زیر کلید صحیح چنانچه در شکل پیداست برابر 0x3C است.

برای پیاده‌سازی حمله تحلیل همبستگی توان در ابتدا با استفاده از FPGA تعداد N متن آشکار را رمز می‌کنیم. سپس مصرف توان در اولین پالس ساعت یا پالس ساعتی که عملیات جانشینی بیت‌ها در آن انجام می‌شود را اندازه‌گیری می‌کنیم. اندازه‌گیری‌ها ده مرتبه تکرار شده و متوسط‌گیری می‌شود تا اثر نویز های الکترونیک به حداقل برسد. سپس حداکثر مقدار مصرف توان را بازای هر متن آشکار بدست آورده و در یک ماتریس ذخیره می‌کنیم. از اینرو یک ماتریس ستونی N تایی مصرف توان بازای تمام متون آشکار خواهیم داشت که به آن ماتریس مصرف توان سراسری^{۲۰} می‌گوییم. برای اولین تبدیل جانشینی بیت‌ها یک تابع انتخاب D در نظر می‌گیریم.

تابع انتخاب در اینجا تعداد بیت‌هایی است در رجیستر خروجی تبدیل جانشینی بیت‌ها بازای هر ورودی تغییر می‌کند. سپس با استفاده از یک ابزار شبیه‌سازی مقدار D را بازای تمام ۲۵۶ کلید ممکن پیش بینی کرده و ذخیره می‌کنیم. لذا یک ماتریس پیش بینی بابعاد $N \times 256$ خواهیم داشت که به آن ماتریس

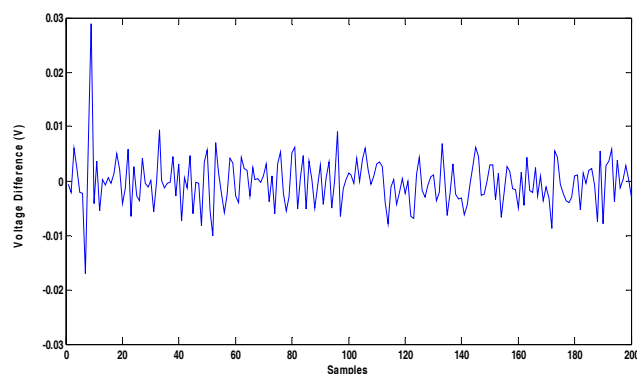
انتخاب می‌شود. بنابراین برای یک ورودی مشخص هر بار مقادیر باینری متفاوتی پردازش شده و الگوی مصرف توان ثابت نخواهد بود در حالیکه خروجی یکسان است. چنانچه در بخش ۴ نشان داده شد ایجاد ۳۲ SBox تصادفی بشکلی که ذکر شد چندان هزینه بر نخواهد بود زیرا این تغییرات براحتی قابل ترجمه به گیت‌های منطقی است. ماتریس δ در ورودی SBox مانند بردار ماسک ورودی عمل کرده و در هر مرحله با تغییر آن داده‌های میانی که در درون تبدیل جانشینی بیت‌ها مورد پردازش قرار می‌گیرد متفاوت خواهد بود. از اینرو حتی اگر چنانچه بارها یک ورودی چندین بار اجرا شود پروفایل مصرف توان و داده‌هایی که برای حمله تحلیل تفاضلی توان جمع‌آوری می‌شوند متفاوت با یکدیگر خواهد بود که این مساله باعث مشکل شدن کار شخص کنجکاو در اجرای موفقیت آمیز حمله خواهد بود. همین امر باعث کاهش نسبت سیگنال به نویز (SNR) خواهد شد و همانطور که می‌دانیم کیفیت حمله بستگی به میزان نسبت سیگنال به نویز دارد. برای جلوگیری از نشت اطلاعات هنگام ذخیره کردن خروجی تبدیل جانشینی بیت‌ها در رجیستر مربوطه، یک شمانده تصادفی دیگر دو مقدار δ^1 دیگر را انتخاب کرده و مقدار پیش نهایی بصورت همزمان و موازی در آنها ضرب می‌شود. این امر باعث غیر قابل تشخیص بودن نحوه مصرف توان در مرحله نهایی خروجی تبدیل جانشینی بیت‌ها و جلوگیری از حمله شخص کنجکاو به این نقطه خواهد شد. پیاده‌سازی ضرب همزمان در FPGA بسادگی قابل تحقق است زیرا قابلیت انجام عملیات همزمان جزو ویژگی‌های این تراشه است. چنانچه به روش پیشنهادی توجه کنیم متوجه این نکته اساسی خواهیم شد که روش مزبور منحصر به پیاده‌سازی سخت‌افزاری نبوده و قابل اعمال به پیاده‌سازی‌های نرم‌افزاری و سایر بسترها مانند کارت‌های هوشمند، پردازشگرهای سیگنال دیجیتال و سایر توکن‌های امنیتی نیز هست. هیچ پارامتر دیگری بجز کلید رمز مورد پردازش قرار نمی‌گیرد و چنانچه در بخش بعد نشان خواهیم داد این روش علاوه بر فراهم آوردن امنیت بالا در برابر حمله تحلیل توان هزینه پیاده‌سازی کمی را به کاربر تحمیل کرده و باعث کاهش سرعت سیستم نیز نخواهد شد و لذا تاثیر کمی بر میزان بازدهی سیستم خواهد داشت.

۵ - حمله به یک سیستم واقعی

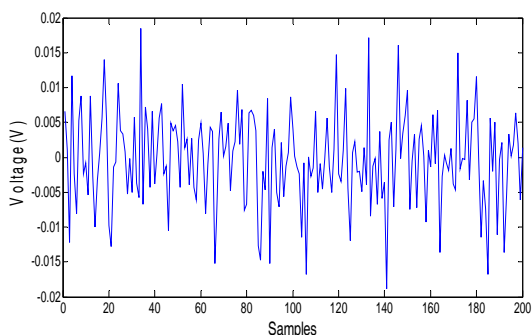
برای بررسی میزان موثر بودن روش پیشنهادی، دو نسخه از الگوریتم AES، یک نسخه محافظت شده و یک نسخه محافظت نشده در یک معماری حلقه بشکلی که تنها یک قالب از داده‌ها بطور همزمان مورد پردازش قرار می‌گیرند با استفاده از کدهای قابل سنتز زبان توصیف سخت‌افزار Verilog پیاده‌سازی شده و بر روی تراشه Xilinx Spartan-II FPGA سنتز شدند. کلید رمز در هر دو الگوریتم ۱۲۸ بیتی بوده است. پیاده‌سازی هسته رمزنگار در نسخه محافظت نشده بر روی این تراشه، 983 CLB slices یا ۴۱٪ مساحت روی تراشه را اشغال نمود در حالیکه همان پیاده‌سازی برای نسخه محافظت شده با ۳۲ ساختار SBox، ۱۰۵۰ CLB slices مساحت اشغال کرد که تنها ۷٪ بیشتر از مقدار قبلی و بسیار کمتر از مقادیر گزارش شده برای سایر روش‌ها در ادبیات مربوطه تاکنون است. ابزار سنتز سرعت پالس ساعت را در هر دو حالت 42.2 MHz نشان داد. تجهیزات اصلی اندازه‌گیری چنانکه در شکل ۸ نشان داده شده است شامل یک برد FPGA، یک اسیلوسکوپ دیجیتال MSO7034A ساخت کارخانه Agilent با پهنای باند 350 MHz و فرکانس نمونه برداری 2 GS/sec است. برد شامل دو تغذیه جداگانه 3.3 V برای تغذیه I/O و یک تغذیه 2.5 V برای تغذیه هسته تراشه است. در آزمایشات انجام شده تنها تغذیه هسته مورد بررسی و اندازه‌گیری قرار گرفت. یک مقاومت کم اهم (۱۰ اهم) بین تغذیه برد و زمین قرار داده شد تا تغییرات جریان یا تغییرات توان اندازه‌گیری شود. بمنظور کاهش نویز الکترونیک و سوئیچینگ و

آزمایشات فوق برای نسخه محافظت شده تکرار شد. شکل‌های ۱۲ و ۱۳ نشان دهنده منحنی‌های تفاضل توان برای حدس صحیح و یک حدس غلط از زیر کلید اول هستند. همانگونه که از این دو شکل مشخص است زیر کلید صحیح قابل تشخیص از زیر کلید اشتباه نیست زیرا نویز زیادی در سیستم وجود دارد. شکل‌های ۱۴ و ۱۵ نشان‌دهنده منحنی مشاهده توان در نسخه حفاظت شده بازی دو ورودی یکسان است که همانطور که از شکل پیداست با اینکه در هر دو حالت ورودی برابر است اما تفاضل مصرف توان اصلا شبیه به یکدیگر نیست. شکل ۱۶ نشان‌دهنده آزمایش همبستگی برای نسخه محافظت شده پس از ثبت مصرف توان ۱۰۰۰ متن رمز شده است. همانگونه که از شکل پیداست هیچ پیک واضحی در شکل دیده نمی‌شود و تنها باندی از اعداد تصادفی دیده می‌شود. آزمایش فوق مجدداً برای ۶۰۰۰ متن آشکار و رمز شده معادل آن تکرار گردید و باز هم کلید صحیح قابل تشخیص از کلیدهای اشتباه نبود. قابل ذکر است که در این کار حملات توان از مرتبه بالاتر مورد بررسی قرار نگرفتند و تحقیق در مورد میزان مقاومت روش پیشنهادی در برابر چنین حملاتی از جمله برنامه‌های آتی تحقیقاتی است. برای مقاوم تر کردن هر چه بیشتر پیاده‌سازی در مقابل حمله توان می‌توان روش ارائه شده در این مقاله را با سایر تکنیک‌های تصادفی‌سازی مصرف توان مانند اجرای روال‌های شبه تصادفی یا جایجا کردن ترکیب کرد. همانطور که می‌دانیم ترتیب ۱۶ بایتی که در هر قالب وارد تبدیل جانشینی بایت‌ها می‌شود اهمیتی ندارد و می‌توان ترتیب آنها را جایجا کرد. از اینرو می‌توان با عوض کردن ترتیب ورود و خروج بایت‌ها به این تبدیل شخص کنجکاو را دچار گمراهی بیشتری کرد. اجرای روال‌های تصادفی باعث کاهش بازدهی خواهد شد اما برهم زدن ترتیب بایت‌ها باعث کاهش بازدهی نخواهد شد.

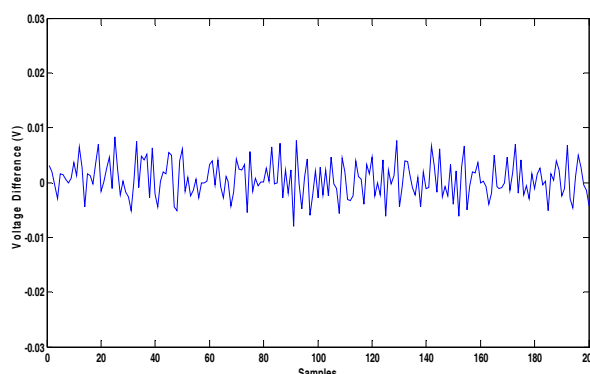
پیش‌بینی سراسری^{۲۱} می‌گوییم و عناصر آن اعداد صحیح بین ۰ تا ۷ هستند. سپس ضریب همبستگی بین ماتریس مصرف توان و تمام ستون‌های ماتریس پیش بینی (متناظر با تمام ۲۵۶ حدس کلید) را محاسبه می‌کنیم. اگر حمله موفقیت آمیز باشد انتظار می‌رود تا تنها یک مقدار که همان حدس صحیح از کلید است مقدار کاملاً بزرگتری در مقایسه با دیگران داشته باشد. نتیجه آزمایش حمله همبستگی برای اولین بایت کلید در شکل ۱۱ نشان داده شده است و چنانچه از شکل پیداست مقدار ضریب همبستگی در حدس صحیح یا 0x3C بصورت یک پیک کاملاً واضح است. آزمایشات ما نشان داد که پس از تکمیل اندازه گیری‌ها کشف کامل ۱۲۸ بیت کلید با استفاده از یک کامپیوتر چهار هسته‌ای با فرکانس 2.5 GHz تقریباً در دو ساعت امکان‌پذیر است.



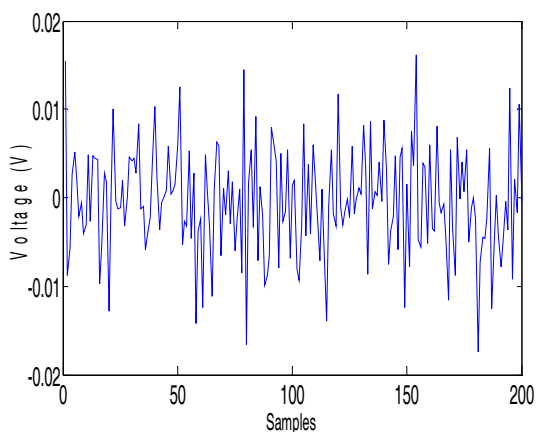
شکل ۹- منحنی تفاضل توان برای یک حدس صحیح از زیر کلید اول



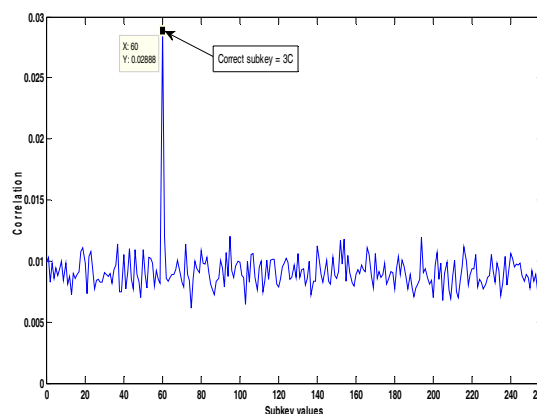
شکل ۱۲- منحنی تفاضل توان برای حدس صحیح از زیر کلید اول در نسخه محافظت شده



شکل ۱۰- منحنی تفاضل توان برای یک حدس غیر صحیح از زیر کلید اول



شکل ۱۳- منحنی تفاضل توان برای حدس غیر صحیح از زیر کلید اول در نسخه محافظت شده

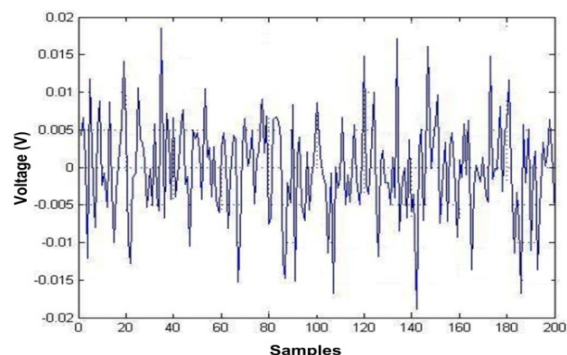


شکل ۱۱- حدس صحیح زیر کلید اول با استفاده از حمله تحلیل همبستگی توان

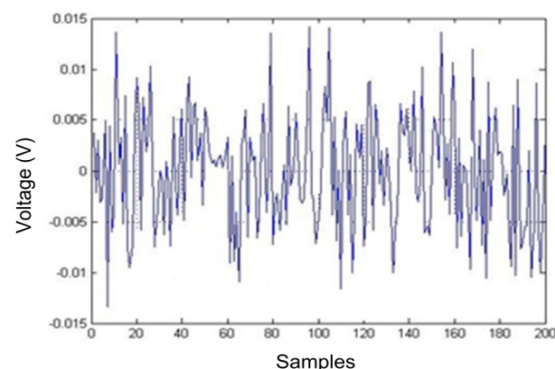
کردن منحنی مصرف توان و یا تغییرات در سطح گیت‌ها و ترانزیستورها هستند. این روش‌ها معمولا بدون هماهنگی با کارخانه سازنده FPGA بسختی قابل پیاده‌سازی هستند اما برخی روش‌های مبتنی بر تغییر در معماری پیاده‌سازی قابل تحقق در FPGA هستند. برخی از روش‌ها که بطور حسی و شهودی روش‌های موثری هستند مانند تصادفی کردن اجرای کدها یا افزودن ماژول‌های مصرف کننده توان بطور تصادفی مانند تاخیر دهنده‌ها یا شیفت دهنده‌های تصادفی و یا فرورونده‌های جریان بسختی می‌توانند در مقابل این حمله مقاومت کنند. روش‌های جدید سعی در از بین بردن وابستگی مصرف توان ابزار رمزنگاری با مقادیر سیگنال‌ها در گره‌های درونی مدار با استفاده از تصادفی‌سازی یا هموار کردن مصرف توان دارند اما همچنانکه گفته شد هیچ کدام از این روش‌ها نمی‌توانند امنیت کاملی را در مقابل این حملات فراهم آورند و تنها باعث مشکل‌تر شدن کار شخص کنجکاو و نیاز وی به اندازه‌گیری‌های بیشتر خواهند شد. تصادفی کردن مصرف توان عمدتا با ماسک کردن یعنی تصادفی کردن مقادیر در گره‌های میانی مدار بدون آنکه مقدار نهایی تغییر کند یا در سطح گیت استفاده از ماسک بیت‌هایی که باعث متعادل شدن احتمال گذار در خروجی گیت‌ها می‌شود انجام می‌شود. هموار کردن منحنی مصرف توان در سطح مدار با استفاده از منطق‌هایی مانند منطق تفاضلی پویا^{۲۳} که از آن به منطق دو ریلی با پیش شارژ^{۲۳} نیز تعبیر می‌شود و باعث می‌شود تا هر گیت منطقی در هر سیکل تنها یک بار شارژ شود. البته افزایش مقاومت در برابر حمله تحلیل توان بدون هزینه بدست نخواهد آمد. ماسک کردن داده‌ها در سطح الگوریتم باعث افزایش سطح اشغالی بر روی تراشه حداقل یک و نیم برابر و ماسک کردن در سطح ترانزیستور یا گیت بین دو تا پنج برابر باعث افزایش مساحت اشغالی روی تراشه خواهد شد. مثلا همان منطق دو ریلی با پیش شارژ که در مورد آن بحث شد باعث افزایش سطح تا سه برابر خواهد شد. استفاده از منطق تفاضلی یعنی استفاده از "10" بجای "0" و "01" بجای "1" باعث کاهش نسبت سیگنال به نویز و ثابت ماندن وزن همیتگ اطلاعات رد و بدل شده خواهد شد اما هزینه سخت‌افزاری را تقریبا تا صد درصد افزایش می‌دهد. برخی تکنیک‌ها مانند طراحی سلول‌های اختصاصی تا حدی باعث کاهش هزینه مقابله می‌شود اما همچنان هزینه بالاست. افزودن فیلتر برای ماسک کردن مشخصات طیفی یکی از راه‌های موثر برای مخفی کردن وابستگی توان مصرفی دستورالعمل‌ها به عملوندهاست. باید توجه داشت که شخص باید در ارائه روش مقابله و ادعا در مورد کارایی روش خود کاملا محتاط باشد زیرا بسیاری از روش‌ها که بلحاظ ظاهری یا حتی تئوری در مقابل حملات کانال جانبی مقاوم هستند در عمل در مقابل روش‌های پیچیده حمله چندان تاب نمی‌آورند و مغلوب می‌شوند. سوال اساسی اینجاست که آیا می‌توان این تکنیک‌ها را بیشتر بهینه‌سازی کرد و اینکه آیا می‌توان تکنولوژی جدیدی برای مقابله با این حملات ابداع نمود؟ سوال مهم دیگر آن است که آیا می‌توان راه دیگری برای سنجش میزان مقاومت در مقابل حمله بدون پیاده‌سازی حمله بر مبنای ابزارهای آماری یا هر ابزار دیگر برای ارزیابی پیاده‌سازی پیدا کرد؟ آیا می‌توان معیار دیگری برای اندازه‌گیری میزان مقاومت بر مبنای پارامترهای طراحی مانند فاکتور فعالیت^{۲۴} یا پروفایل مصرف توان پیدا کرد؟ با چه معیار یا معیارهایی می‌توان میزان مقاومت دو پیاده‌سازی مختلف را با یکدیگر مقایسه نمود؟ این سوالات نمونه‌هایی از سوالات مهم و زمینه‌های تحقیقاتی باز در مورد استفاده از FPGA بعنوان ماژول رمز است که هنوز پاسخ چندان روشنی به آنها داده نشده است.

۷- نتیجه‌گیری

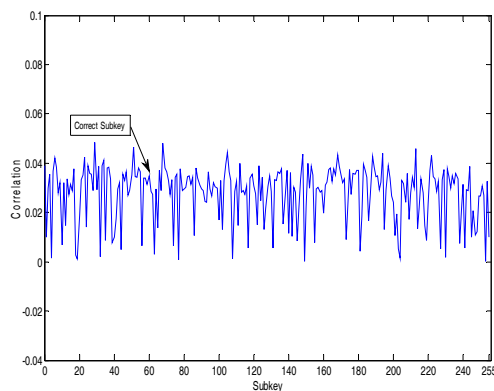
حملات کانال جانبی از جمله مهمترین تهدیدات برای امنیت سیستم‌های رمزنگاری معاصر هستند. اهمیت این حملات بگونه‌ای بوده است که بمحض مطرح



شکل ۱۴- منحنی مشاهده توان در نسخه حفاظت شده بازای یک ورودی دلخواه



شکل ۱۵- منحنی مشاهده توان در نسخه حفاظت شده بازای همان ورودی پس از اجرای مجدد الگوریتم



شکل ۱۶- منحنی تحلیل همبستگی توان در نسخه حفاظت شده

۶- چالش‌های مقابله با حمله تحلیل توان

در سال‌های اخیر تحقیقات زیادی در مورد جلوگیری از حمله تحلیل توان انجام شده است. بر این مبنای روش‌های مقابله با حمله تحلیل توان را می‌توان به دو گروه عمده تقسیم کرد: روش‌های نرم‌افزاری و روش‌های سخت‌افزاری. عمده روش‌های مطرح شده روش‌های نرم‌افزاری بوده و سعی در مقابله با حمله با استفاده از روش‌های مقابله در سطح الگوریتم دارند مانند ماسک کردن دیتا با استفاده از اعداد تصادفی. برخی از این روش‌ها قابل اعمال به سخت‌افزارهای اختصاصی مانند FPGA نیز هستند. روش‌های مقابله سخت‌افزاری عمدتا شامل روش‌های یکنواخت

Cryptographic Hardware and Embedded Systems, pp. 252-263, 2000.

[8] J. J. A. Fournier, S. Moore, H. Li, R. Mullins, and G. Taylor, "Security Evaluation of Asynchronous Circuits," *Proc. International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 125-136, 2003.

[9] S. Moore, R. Anderson, R. Mullins, G. Taylor, and J. J. A. Fournier, "Balanced Self-Checking Asynchronous Logic for Smart Card Applications," *Journal of Microprocessors and Microsystems*, vol. 27, no. 9, pp. 421-430, 2003.

[10] K. Tiri, D. Hwang, A. Hodjat, B. C. Lai, S. Yang, P. Schaumont, and I. Verbauwhede, "Prototype IC with WDDL and differential routing-DPA resistance assessment," *Proc. International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 354-365, 2005.

[11] K. Tiri, and I. Verbauwhede, "Securing Encryption Algorithms against DPA at the Logic Level: Next Generation Smart Card Technology," *Proc. International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 125-136, 2003.

[12] S. Mangard, E. Oswald, and T. Popp, *Power Analysis Attacks, Revealing the Secret of Smart Cards*, Springer, 2006.

[13] S. Mangard, N. Pramstaller, and E. Oswald, "Successfully Attacking Masked AES Hardware Implementations," *Proc. International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 157-171, 2005.

[14] N. T. Courtois, and L. Goubin, "An Algebraic Masking Method to Protect AES against Power Attacks," *Proc. International Conference on Information Security and Cryptology*, pp. 199-209, 2006.

[15] T. Wollinger, J. Guajardo, and C. Paar, "Security on FPGAs: State-of-the-Art Implementations and Attacks," *ACM Transactions on Embedded Computing Systems*, Vol. 3, No. 3, pp. 534-574, 2004.

[16] X. Zhang, and K. K. Parhi, "High-Speed VLSI Architectures for the AES Algorithm," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 12, no. 9, pp. 957-967, 2004.

[17] A. Satoh, S. Morioka, K. Takano, and S. Munetoh, "A Compact Rijndael Hardware Architecture with S-Box Optimization," *Proc. of the Advances in Cryptology*, pp. 239-254, 2001.

[18] J. P. Deschamps, J. L. Imana, and G. D. Sutter, *Hardware Implementation of Finite-Field Arithmetic*, McGraw Hill, 2009.

[19] A. Rudra, P. K. Dubey, C. S. Jutla, V. Kumar, J. R. Rao, and P. Rohatgi, "Efficient Rijndael Encryption Implementation with Composite Field Arithmetic," *Proc. of*

شدن مورد توجه ویژه صاحب نظران این رشته، کارخانه‌های سازنده و نیز موسسات مختلف استفاده کننده از این سیستم‌ها قرار گرفته‌اند. الگوریتم‌هایی نظیر الگوریتم رمزنگاری استاندارد از نظر ریاضی باندازه کافی در مقابل حملات مقاوم هستند اما این کار نشان داد هنگامیکه بر روی مازول‌های سخت‌افزاری مانند FPGA پیاده‌سازی می‌شوند در مقابل تحلیل توان کاملاً آسیب‌پذیرند. گرچه تحقق موفقیت‌آمیز این حملات بر روی تراشه‌های FPGA بمراتب سخت‌تر از کارت‌های هوشمند است اما در این تحقیق ضمن پیاده‌سازی صحیح حمله موفق به بازیابی کلید در مدت زمان حدود دو ساعت شدیم. علاوه بر آن روش ساده جدیدی بر مبنای ساختار ریاضی الگوریتم AES برای مقابله با حمله تحلیل توان ارائه دادیم که بدون آنکه خدشه‌ای به استاندارد بودن الگوریتم وارد شود باعث افزایش قابل توجه مقاومت پیاده‌سازی با هزینه پیاده‌سازی کم خواهد شد. تقریباً تمام الگوریتم‌هایی که در کارت‌های هوشمند تعبیه می‌شوند بگونه‌ای طراحی می‌شوند که در مقابل انواع حملات توان مقاوم باشند اما تقریباً هیچ کاری در زمینه آنکه این الگوریتم‌ها را بطور ذاتی در برابر حمله مقاوم کنند انجام نشده است. این پژوهش نشان داد که می‌توان الگوریتم‌هایی طراحی کرد که بطور ذاتی در مقابل این حملات مقاوم باشند. محتمل است در آینده حملاتی که ترکیبی از چند منبع اطلاعات کانال جانبی مانند ترکیب حمله تحلیل توان و القا خطا هستند همراه با روش‌های قدرتمندتر آماری و مشاهده همزمان چند عملیات بتوانند بطور جدی‌تر سخت‌افزارهای رمزنگاری را مورد تهدید قرار دهند. بررسی تئوری این مساله و نیز روش‌های مقابله با این حملات بالقوه می‌تواند از جمله موضوعات جذاب و مهم تحقیقاتی برای محققین و نیز صاحبان صنایع باشد.

مراجع

[1] M. Masoomi, M. Masoumi, and M. Ahmadian, "A Practical Differential Power Analysis Attack against an FPGA Implementation of AES Cryptosystem," *IEEE I-Society 2010*, 2010.

[2] P. Kocher, J. Jaffe, and B. Jun, "Differential Power Analysis," *Proc. Crypto*, pp. 388-397, 1999.

[3] F. X. Standaert, L. vanOldeneel, D. Samyde, and J. J. Quisquater, "Power Analysis of FPGAs, How Practical is the Attack?," *Proc. the International Conference on Field Programmable Logic and Application*, pp. 701-711, 2003.

[4] L. T. Mc Daniel, *An Investigation of Differential Power Analysis Attacks on FPGA-based Encryption Systems*, Master Thesis, Virginia Polytechnic Institute, 2003.

[5] N. T. Courtois, and L. Goubin, "An Algebraic Masking Method to Protect AES against Power Attacks," *Proc. International Conference on Information Security and Cryptology*, pp. 199-209, 2006.

[6] M. Masoumi, F. Raissi, and M. Ahmadian, "NanoCMOS-Molecular Realization of Rijndael," *Proc. International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 285-297, 2006.

[7] C. Clavier, J. S. Coron, and N. Dabbous, "Differential Power Analysis in the Presence of Hardware Countermeasures," *Proc. International Workshop on*

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۹/۱۰/۷

تاریخ اصلاح: ۹۰/۲/۱۴

تاریخ قبول شدن: ۹۰/۳/۱۱

نویسنده مرتبط: مسعود معصومی، دانشکده فنی، دانشگاه آزاد اسلامی واحد اسلامشهر، تهران، ایران.

the International Workshop on Cryptographic Hardware and Embedded Systems, pp. 171-184, 2001.

[20] A. Morioka, and A. Satoh, "A 10 Gbps Full-AES Crypto Design with a Twisted-BDD SBox Architecture," *Proc. International Conference on Computer Design*, pp. 98-103, 2002.

[21] V. Rijmen, *Efficient Implementation of the Rijndael S-Box*, Katholieke Universiteit Leuven, Dept. ESAT. Belgium.

[22] L. Xiao, and H. M. Heys, "Hardware Design and Analysis of Block Cipher Components," *Proc. International Conference on Information Security and Cryptology*, pp. 164-181, 2002.

[23] R. Lidl, and H. Niederreiter, *Introduction to Finite Field and Applications*, Cambridge University Press, 1986.

[24] T-Ha Lee, C. Canovas, and J. Cledier, "An Overview of Side-Channel Analysis Attacks," *Proc. of the ACM symposium on Information, computer and communications security* pp. 33-43, 2008.

- ¹ Side-Channel Attacks
- ² Simple Power Analysis
- ³ Differential Power Analysis
- ⁴ Correlation Power Analysis
- ⁵ Complementary Logic
- ⁶ Sense Amplifier Based Logic
- ⁷ Asynchronous Logic
- ⁸ Composite Field Arithmetic
- ⁹ Lookup Tables
- ¹⁰ Extended Euclid's Theorem
- ¹¹ Fermat Little Theorem
- ¹² Composite Field
- ¹³ Isomorphism
- ¹⁴ Iterative
- ¹⁵ Isomorphism Mapping
- ¹⁶ Decompose
- ¹⁷ Autoisomorphism
- ¹⁸ Secondary Peaks
- ¹⁹ Ghost Peaks
- ²⁰ Global Consumption Matrix
- ²¹ Global Prediction Matrix
- ²² Differential Dynamic Logic
- ²³ Precharged Dual Rail Logic
- ²⁴ Activity Factor



مسعود معصومی کارشناسی خود را در رشته مهندسی الکترونیک در سال ۱۳۷۴ از دانشگاه گیلان با عنوان دانشجوی رتبه اول اخذ نمود و در سال ۱۳۷۹ موفق به اخذ مدرک کارشناسی‌ارشد در همین رشته از دانشگاه صنعتی خواجه نصیر با عنوان دانشجوی رتبه اول گردید. نامبرده در سال ۱۳۸۵ مدرک دکترای خود را از دانشگاه صنعتی خواجه نصیر با ارائه روشی جدید برای پیاده‌سازی الگوریتم‌های رمزنگاری با استفاده از نانوالکترونیک دریافت نمود. مسعود معصومی در سال ۱۳۸۹ پس از طی یک دوره تحقیقاتی یک و نیم ساله در زمینه حملات کانال جانبی به سیستم‌های رمزنگاری دوره فرا دکترا را در دانشگاه صنعتی خواجه نصیر با موفقیت به اتمام رساند. مهمترین زمینه‌های مورد علاقه وی پیاده‌سازی الگوریتم‌های رمزنگاری، پردازش سیگنال و کدینگ بصورت سخت‌افزاری و مجتمع، حملات کانال جانبی به سیستم‌های رمزنگاری و روش‌های مقاوم‌سازی این سیستم‌ها در مقابل چنین حملاتی می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

m_masoumi@eedt.kntu.ac.ir



سید مجتبی دهنوی مدرک کارشناسی را در رشته ریاضی کاربردی از دانشگاه علم و صنعت ایران در سال ۱۳۷۹ و کارشناسی‌ارشد را در گرایش جبر و ترکیبیات از دانشگاه صنعتی امیرکبیر در سال ۱۳۸۲ دریافت نموده است. وی در حال حاضر دانشجوی دکترای رشته ریاضیات رمز در دانشگاه تربیت معلم تهران می‌باشد. زمینه‌های مورد علاقه او رمزنگاری متقارن، توابع بولی و عملگرهای پایه در رمزنگاری می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

dehnavism@tmu.ac.ir

معرفی فیلترهایی مناسب برای بهسازی گفتار در حوزه موجک

هدی کاویانی شهرضایی شیما طیبیان احمد اکبری

دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران

چکیده

تاکنون روش‌های متفاوتی برای کاهش نویز در سیستم‌های پردازش گفتار، استفاده شده‌اند که برخی از آنها در حوزه تبدیل موجک صورت گرفته‌اند. روش‌های بهسازی گفتار در این حوزه، از طریق آستانه‌بندی و یا با فیلتر کردن ضرایب نویزی، محقق می‌شود. در این مقاله، ابتدا مجموعه فیلترهای مناسبی که در روش فیلتر کردن در حوزه تبدیل موجک قابل استفاده هستند، بر اساس روابط ریاضی معرفی شده‌اند. سپس با استفاده از فیلترهای این مجموعه، دو فیلتر جدید تعریف شده است که فیلتر اول از ترکیب فیلترهای مینیمم و ماکزیمم حاصل شده است و فیلتر دوم که بر پایه فیلتر مینیمم بنا شده است، توانی متغیر با مقادیر کوچکتر از یک را به این فیلتر نسبت می‌دهد. اساس تعریف این فیلترها بر این اصل استوار است که فیلتر پیشنهادی در فریم‌های سکوت بیشترین نویز را از سیگنال حذف کرده و در فریم‌های گفتار همزمان با کاهش نویز، کمترین اعوجاج را بر روی سیگنال گفتار ایجاد کند.

کلمات کلیدی: بهسازی گفتار، تبدیل موجک، فیلتر کردن، نویز باقیمانده، انرژی سیگنال.

۱- مقدمه

آستانه‌بندی [۶-۱۰] و یا با فیلتر کردن ضرایب نویزی [۱۱-۱۳]، محقق می‌شود. در روش‌های بهبود گفتار بر پایه فیلتر کردن در حوزه موجک، سیگنال ورودی پس از پنجره‌بندی، به تعدادی زیرباند تجزیه می‌گردد سپس با اعمال فیلترهای بهبود گفتار بر روی هر یک از زیرباندها، از تأثیر نویز کاسته می‌شود. در مرحله بعد، با ترکیب زیرباندهای بهبود، سیگنال بهسازی شده نهایی حاصل می‌شود. در این مقاله ابتدا با کمک روابط ریاضی، رابطه میان انرژی سیگنال‌های ورودی و خروجی با انرژی توزیع شده در زیرباندهای درخت موجک، بررسی شده است. سپس با استفاده از همین خاصیت، مجموعه‌ای از فیلترهای قابل استفاده در این حوزه معرفی شده‌اند که از میان این فیلترها، کوچکترین آنها بیشترین نویز را از سیگنال حذف می‌کند ولی بزرگترین فیلتر کمترین اعوجاج را ایجاد می‌کند. با توجه به این اثرات فیلتر پیشنهادی اول، با استفاده از ترکیب این دو فیلتر ایجاد شده است به طوری که در فریم‌های گفتار فیلتر ماکزیمم و در فریم‌های سکوت، فیلتر مینیمم بیشترین تأثیر را داشته باشد. فیلتر پیشنهادی دوم با توجه به خاصیت حذف حداکثری نویز توسط فیلتر مینیمم تعریف شده است، به طوری که توانی متغیر به این فیلتر نسبت داده شده است که در فریم‌های گفتار مقدار فیلتر را اندکی کاهش می‌دهد تا از بروز اعوجاج در سیگنال جلوگیری شود و در

با رشد سریع سیستم‌های مخابرات صوت، بهسازی گفتار نویزی تبدیل به یک موضوع مهم و فعال تحقیقاتی شده است. هدف اصلی در حذف نویز این است که سیگنال بازسازی شده تا حد امکان به سیگنال اصلی نزدیک بوده و در عین حال خواص اصلی سیگنال همچون میزان همواری آن نیز حفظ شود. روش‌های بهسازی گفتار در دو حوزه فرکانس و زمان تدوین شده‌اند. از تکنیک‌های حوزه زمان می‌توان به مدل مخفی مارکف، فیلتر وینر و فیلتر کالمن اشاره کرد [۱-۳]. روش‌های مبتنی بر تبدیل موجک و تبدیل فوریه نیز مثال‌هایی از تکنیک‌های بهسازی گفتار در حوزه فرکانس هستند [۴-۱۳]. هدف از استفاده تبدیلات، انتقال سیگنال نویزی از حوزه زمان به حوزه دیگری است که در حوزه جدید امکان استخراج آسان‌تر و با کیفیت بهتر سیگنال اصلی از سیگنال نویزی وجود دارد. در این میان تبدیل موجک با داشتن مزایایی نظیر دارا بودن مقیاس‌های زمان-فرکانس مختلف و پیچیدگی محاسباتی پایین‌تر نسبت به تبدیل فوریه، این هدف را بهتر برآورده می‌سازد. بهسازی گفتار بر مبنای تبدیل موجک، از طریق

در بعضی از موارد، شنوندگان سیگنال نویزی اولیه را به سیگنال بهسازی شده بوسیله سیستم مبتنی بر تبدیل موجکی که در آن از آستانه‌گذاری استفاده شده است، ترجیح داده‌اند.

۲-۲- روش‌های بر مبنای فیلتر کردن

در روش‌های بر مبنای فیلتر کردن، پس از پنجره‌بندی و محاسبه ضرایب تبدیل موجک، فیلتر مورد نظر بر روی ضرایب زیرباندها اعمال می‌شود. رابطه (۱) چگونگی اعمال فیلتر را بر روی زیر باندها نشان می‌دهد:

$$\tilde{S}(m, f, j) = H(m, f) \times X(m, f, j) \quad (1)$$

که در این رابطه $X(m, f, j)$ مؤلفه j ام از زیرباندها f از فریم m از سیگنال گفتار نویزی را نشان می‌دهد و $\tilde{S}(m, f, j)$ مؤلفه j ام از زیرباندها f از فریم m از سیگنال بهبودیافته را نشان می‌دهد و $H(m, f)$ فیلتر بهبود گفتار مربوط به زیرباندها f از فریم m می‌باشد.

استفاده از روش فیلتر کردن برای بهسازی گفتار از سال‌های دورتر همواره مورد توجه بوده است ولی استفاده از آن بیشتر در حوزه تبدیل فوریه بوده است. از جمله شناخته‌شده‌ترین روش‌های این زمینه می‌توان به روش تفاضل طیف توان و استفاده از فیلترهای طیفی نظیر فیلتر وینر اشاره نمود [۴]. از جمله اثرات نامطلوب این شیوه بهبود گفتار میتوان به ایجاد اعوجاج در سیگنال و ایجاد نویز موسیقی اشاره نمود. علاوه بر این، بنا به ماهیت تبدیل فوریه و نایستانت بودن سیگنال گفتار، عملیات باید بر روی طیف دوره کوتاه سیگنال صورت گیرد که محدودیت‌هایی در تنظیم رزولوشن زمانی و فرکانسی ایجاد می‌نماید. از این رو با استفاده از فیلترها در حوزه تبدیل موجک به جای تبدیل فوریه می‌توان این اثرات نامطلوب را تا حد زیادی کاهش داد.

۳- تبدیل موجک

تبدیل موجک یکی از تبدیل‌هایی است که در سال‌های اخیر در کاربردهای مختلفی از پردازش سیگنال استفاده شده است. در این تبدیل، بر خلاف تبدیل فوریه، توابع پایه، توابع سینوسی و کسینوسی نیستند، بلکه هر مجموعه از توابع با دارا بودن شرایط خاصی می‌توانند به عنوان توابع پایه مورد استفاده قرار گیرد.

در هر تبدیل موجک، یک تابع تغییر مقیاس پایه $\phi(t)$ وجود دارد که مجموعه توابع تغییر مقیاس از انتقال این تابع پایه بدست می‌آیند:

$$\phi_k(t) = \phi(t - k); k \in Z \quad (2)$$

اگر فضای پوشانده شده توسط $\phi_k(t)$ را V_0 بنامیم، آنگاه هر تابع $f(t)$ در این فضا را می‌توانیم بصورت ترکیب خطی از توابع $\phi_k(t)$ بنویسیم:

$$f(t) = \sum_k a_k \phi_k(t); f(t) \in V_0 \quad (3)$$

این رابطه بسط $f(t)$ را در مقیاس صفر نشان می‌دهد. می‌توان آن را در مقیاس‌های دیگر نیز بسط داد. برای این کار از مجموعه‌ای از توابع که از تغییر مقیاس و انتقال تابع اصلی $\phi(t)$ بدست می‌آید، استفاده می‌شود:

فریم‌های سکوت مقدار فیلتر تقریباً بدون تغییر باقی می‌ماند تا بیشترین نویز از سیگنال حذف شود.

ساختار این مقاله به صورت ذیل است؛ در بخش بعد مرور مختصری بر روی بهسازی گفتار مبتنی بر تبدیل موجک، انجام گرفته است. در بخش سوم، به تئوری تبدیل موجک پرداخته و در ادامه در بخش چهارم، رابطه بین انرژی سیگنال‌های ورودی و خروجی با انرژی توزیع شده در زیرباندهای درخت موجک، بررسی شده است. سپس، در بخش پنجم مجموعه‌ای از فیلترهای قابل استفاده در بهسازی گفتار بر مبنای تبدیل موجک، ارائه شده‌اند. در بخش ششم، فیلترهای پیشنهادی در حوزه موجک معرفی شده است. در بخش هفتم مراحل انجام عملیات بهسازی گفتار با استفاده از فیلترهای پیشنهادی بیان شده است و در بخش هشتم نتایج پیاده‌سازی روش‌های پیشنهادی مورد ارزیابی قرار گرفته است. در انتها، در بخش پایانی، نتیجه‌گیری مطالب ارائه شده است.

۲- بهسازی گفتار مبتنی بر تبدیل موجک

روش‌های بهسازی گفتار با کمک تبدیل موجک را می‌توان به دو دسته عمده طبقه بندی کرد:

- ۱) روش‌های بر مبنای آستانه‌گذاری
- ۲) روش‌های بر مبنای فیلتر کردن

۲-۱- روش‌های بر مبنای آستانه‌گذاری

اساس کار آستانه گذاری موجک بر این امر استوار است که، برای اکثر سیگنال‌های موجود در زندگی حقیقی تعداد محدودی از ضرایب موجک در باندهای پایین، برای بازسازی تخمین خوبی از سیگنال اصلی کافی هستند. معمولاً این ضرایب در مقایسه با دیگر ضرایب مقدار بالاتری دارند و در نتیجه بوسیله حذف کردن ضرایبی که کمتر از یک مقدار حد آستانه هستند، می‌توانیم بدون از دست دادن اطلاعات مهم سیگنال، نویز را حذف کنیم.

الگوریتم این روش نویز زدایی به این ترتیب است که ابتدا سیگنال نویزی را پنجره‌بندی می‌کنیم. در ادامه، تبدیل موجک پنجره مورد نظر را محاسبه می‌کنیم. در مرحله بعد، بر مبنای یک الگوریتم آستانه‌گذار و یک مقدار آستانه، تعدادی از ضرایب موجک را حذف می‌کنیم. سپس، تبدیل معکوس موجک را بر ضرایب بدست آمده اعمال می‌کنیم. در آخر، با جمع و همپوشانی پنجره‌های بدست آمده سیگنال تمیز را محاسبه می‌کنیم.

روش آستانه گذاری موجک، ضرایب جزئیات را حذف می‌کند زیرا این ضرایب مؤلفه‌های فرکانسی بالای سیگنال را آشکار می‌کنند، در حالیکه، فرض بر این است که اطلاعات مهم سیگنال در فرکانس‌های پایین قرار گرفته‌اند و این فرض برای گروه بزرگی از سیگنال‌ها درست است. پس این بدان معنی است که در فرکانس‌های بالا، نویز تأثیر بیشتری از سیگنال اصلی دارد و به عبارت دیگر در فرکانس‌های بالا، نویز قسمت بیشتری از ضرایب را، نسبت به فرکانس‌های پایین اشغال می‌کند.

روش آستانه‌گذاری موجک، معایبی را هم برای استفاده در سیگنال‌های گفتار داراست. از آن جمله این است که قسمت‌هایی از گفتار مانند نواحی صامت که اطلاعات مهمی از سیگنال را در فرکانس‌های بالا حمل می‌کنند، در قسمت جزئیات ضرایب قرار می‌گیرند. از دیگر معایب این روش آن است که، فقط در تست‌های عینی موفق عمل میکند و این برای کاربردی نظیر گفتار که هر دو جنبه عینی و شنیداری در آن مهم هستند، کافی به نظر نمی‌رسد. زیرا به عنوان مثال،

تبدیل موجک گسسته را می‌توان هم به صورت دوتایی و هم به صورت موجک بسته‌ای در نظر گرفت که در درخت موجک بسته‌ای هم ضرایب جزئیات و هم ضرایب تخمین در سطح درخت گسترش می‌یابند [۱۳]. در کار حاضر از یک درخت بسته موجک در سه سطح و با ۸ زیرباند استفاده شده است.

۴- رابطه بین انرژی زیرباندهای تبدیل موجک با انرژی سیگنال‌های ورودی و خروجی

می‌توان ثابت کرد که انرژی سیگنال ورودی با میانگین انرژی زیرباندها در حوزه تبدیل موجک برابر است. فرض کنیم $h_a = [a_0, \dots, a_L]$ ضرایب فیلتر پایین‌گذر و $h_d = [d_0, \dots, d_L]$ ضرایب فیلتر بالاگذر تبدیل موجک باشند. از آنجا که پایه‌های موجکی که معمولا برای بهسازی گفتار استفاده می‌شوند دارای فیلترهایی با تعداد ضرایب زوج هستند، می‌توانیم فرض کنیم که L عددی زوج باشد. همچنین فرض کنیم $x = [x_1, \dots, x_N]$ سیگنال ورودی باشد. در اینصورت داریم:

$$x * h_a(j) = \sum_{k=0}^j a_k x_{j-k} \quad (12)$$

حال انرژی سیگنال در زیر باند اول به صورت زیر بدست می‌آید:

$$|x * h_a|^2 = \sum_{j=0}^{N+L-1} \left(\sum_{k=0}^j a_k x_{j-k} \right)^2 \quad (13)$$

با کمک اتحاد مربع چندجمله‌ای می‌توان نوشت:

$$|x * h_a|^2 = \sum_{j=0}^{N+L-1} \sum_{k=0}^j a_k^2 x_{j-k}^2 + 2 \sum_{j=0}^{N+L-1} \sum_{k=0}^j \sum_{n=1}^{L-k} a_k a_{k+n} x_{j-k} x_{j-k-n} \quad (14)$$

می‌توان ثابت کرد که مقدار نهایی رابطه (۱۳) به صورت رابطه (۱۵) می‌باشد:

$$|x * h_a|^2 = \left(\sum_{k=0}^L a_k^2 \right) \left(\sum_{j=0}^N x_j^2 \right) + 2 \sum_{i=1}^{L/2} \left\{ \sum_{k=0}^{L-2i} a_k a_{k+2i} \right\} \left(\sum_{j=0}^{N-2i} x_j x_{j+2i} \right) + 2 \sum_{i=1}^{L/2} \left\{ \sum_{k=0}^{L-2i+1} a_k a_{k+2i-1} \right\} \left(\sum_{j=0}^{N-2i+1} x_j x_{j+2i-1} \right) \quad (15)$$

با استفاده از رابطه (۱۰) داریم:

$$|x * h_a|^2 = \left(\sum_{j=0}^N x_j^2 \right) + 2 \sum_{i=1}^{L/2} \left\{ \sum_{k=0}^{L-2i+1} a_k a_{k+2i-1} \right\} \left(\sum_{j=0}^{N-2i+1} x_j x_{j+2i-1} \right) \quad (16)$$

با توجه به رابطه (۱۱)، انرژی سیگنال در سطح دوم به صورت زیر بدست

می‌آید:

$$|x * h_d|^2 = \left(\sum_{j=0}^N x_j^2 \right) - 2 \sum_{i=1}^{L/2} \left\{ \sum_{k=0}^{L-2i+1} a_k a_{k+2i-1} \right\} \left(\sum_{j=0}^{N-2i+1} x_j x_{j+2i-1} \right) \quad (17)$$

$$\phi_{k,j}(t) = 2^{-j/2} \phi(2^{-j}t - k); j, k \in Z \quad (4)$$

اگر فضای پوشانده شده توسط $\phi_{k,j}(t)$ را V_j بنامیم، آنگاه خواهیم داشت:

$$\dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots \quad (5)$$

ملاحظه می‌شود که بسط $f(t)$ در مقیاس‌های بالاتر (زهای بزرگتر)، در واقع یک تقریب درشت‌نمایی از آن را به ما می‌دهد. یعنی در مقیاس‌های بالاتر جزئیات را از دست خواهیم داد. برای رفع این مشکل مجموعه توابع دیگری تعریف می‌شود که تفاوت بین فضاهای V_j را بپوشاند. این مجموعه توابع، موجک‌ها می‌باشند که آنها را با $\psi_{j,k}(t)$ نشان می‌دهیم. بنابراین اگر فضای پوشانده شده با توابع $\psi_{j,k}(t)$ را W_j بنامیم، آنگاه خواهیم داشت:

$$V_j = V_{j+1} \oplus W_j \quad (6)$$

توابع $\psi_{j,k}(t)$ از تغییر مقیاس و انتقال تابع موجک مادر $\psi(t)$ به صورت زیر بدست می‌آیند:

$$\psi_{s,\tau}(x) = \frac{1}{\sqrt{s}} \psi\left(\frac{x-\tau}{s}\right) \quad (7)$$

که در آن s و τ به ترتیب پارامترهای مقیاس و تبدیل می‌باشند. در تکنیک‌های بهسازی گفتار مبتنی بر تبدیل موجک معمولا از تبدیلات موجک گسسته استفاده می‌شود. در این تبدیلات از فیلترهای بالاگذر و پایین‌گذر مرتبط با توابع موجک مادر و مقیاس استفاده می‌شود که روابط آن صورت زیر است:

$$a_{j+1}(k) = \sum_m h_a(m-2k) a_j(m) \quad (8)$$

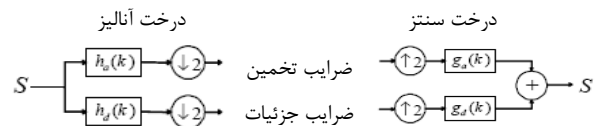
$$d_{j+1}(k) = \sum_m h_d(m-2k) d_j(m) \quad (9)$$

که h_a و h_d فیلترهای بالاگذر و پایین‌گذر و a_i و d_i به ترتیب ضرایب تخمین و جزئیات تبدیل موجک می‌باشند. روابط زیر میان ضرایب تخمین و جزئیات برقرار است.

$$\sum a_k a_{k+2i} = \delta(i) \quad (10)$$

$$a_k = (-1)^k d_{L-k} \quad (11)$$

شکل (۱) ساختاری از یک تبدیل موجک ساده با دو زیرباند را نمایش می‌دهد:



شکل ۱- ساختار درخت آنالیز و سنتز موجک

می‌توان ثابت کرد انرژی سیگنال خروجی به گونه‌ای که شرایط فوق را برآورده نماید، باید به صورت زیر باشد (جزئیات اثبات در ضمیمه آورده شده است):

$$|\hat{s}|^2 = (|x| - |n|)(|x| + a|n|); \quad -1 \leq a \leq 1 \quad (26)$$

حال مجموعه فیلترهای مناسب برای بهسازی گفتار در حوزه تبدیل موجک را به صورت رابطه (۲۷) تعریف می‌کنیم:

$$h_i = \frac{\sqrt{(|x_i| - |n_i|)(|x_i| + a|n_i|)}}{x_i}; \quad -1 \leq a \leq 1 \quad (27)$$

که h_i فیلتر زیرباند i ام و n تعداد زیرباندها است. همچنین منظور از x_i و n_i به ترتیب ضرایب سیگنال نویزی و سیگنال نویز در زیرباند i ام از درخت موجک می‌باشد. با انتخاب فیلتر به صورت رابطه (۲۷) و با توجه به رابطه:

$$|\hat{s}|^2 = \frac{1}{n} \sum_n |x_i h_i|^2 \quad (28)$$

تمام فیلترهای تعریف شده توسط رابطه (۲۷)، برای بهسازی گفتار در حوزه موجک قابل استفاده هستند.

۶- ارائه فیلترهایی مناسب در حوزه تبدیل موجک

۶-۱- فیلتر پیشنهادی اول

در این بخش قصد داریم فیلتری را معرفی کنیم که از ترکیب دو فیلتر از مجموعه فیلترهای رابطه (۲۷) بدست آمده است. برای بدست آوردن یک فیلتر ترکیبی مناسب، ابتدا باید دو فیلتر با کارایی‌های مطلوب و مختلف انتخاب کنیم و سپس با ضریبی مناسب این دو فیلتر را با هم ترکیب کنیم.

۶-۱-۱- انتخاب فیلترها برای ترکیب شدن

از میان فیلترهای بیان شده در رابطه (۲۷)، به ازای $a = -1$ فیلتری با مقدار مینیمم و به ازای $a = 1$ فیلتر ماکزیمم بدست خواهد آمد. از آنجا که هر چه مقدار فیلتر کمتر باشد، ضرایب سیگنال را بیشتر کاهش می‌دهد، قادر است مقدار نویز بیشتری را از گفتار حذف کند. این در حالی است که با اعمال فیلتری با مقدار بیشتر، به سبب تغییر کمتری که بر روی ضرایب ایجاد می‌شود، اعوجاج کمتری را شاهد خواهیم بود. اگر این فیلترها را h_{\min} و h_{\max} بنامیم، داریم:

$$h_{\min}_i = \frac{|x_i| - |n_i|}{x_i} \quad (29)$$

$$h_{\max}_i = \frac{\sqrt{|x_i|^2 - |n_i|^2}}{x_i} \quad (30)$$

که h_{\min}_i و h_{\max}_i مقادیر فیلترها در زیرباند i ام از درخت موجک می‌باشند. با توجه به خواصی که از این دو فیلتر ذکر شد، باید آنها را طوری ترکیب

حال با میانگین گرفتن از انرژی این دو سطح داریم:

$$\frac{|x * h_a|^2 + |x * h_d|^2}{2} = \left(\sum_{j=0}^N x_j^2 \right) \quad (18)$$

بنابر رابطه (۱۸)، انرژی سیگنال ورودی و خروجی با میانگین انرژی در زیرباندهای مختلف برابر است که این خاصیت به سطح‌های بالاتر درخت تجزیه موجک هم قابل تعمیم است. حال قصد داریم به کمک این خاصیت فیلترهای مناسب برای استفاده در حوزه موجک را به دست بیاوریم.

۵- مجموعه فیلترهای مناسب برای بهسازی گفتار در حوزه تبدیل موجک

فرض کنیم x سیگنال نویزی ورودی باشد. در این صورت داریم:

$$x = s + n \quad (19)$$

که s معرف سیگنال تمیز و n سیگنال نویز می‌باشد. هدف یافتن سیگنال خروجی \hat{s} می‌باشد به طوری که این سیگنال به حد کافی به s نزدیک باشد. این کار معمولاً با استفاده از مینیمم کردن معیار مجموع مربعات خطا صورت می‌گیرد که رابطه آن به صورت زیر است:

$$E = (s - \hat{s})^2 \quad (20)$$

ولی از آنجا که بعد از عملیات فیلتر کردن در حوزه موجک، بدست آوردن تخمینی از سیگنال خروجی در حوزه زمان کار بسیار دشواری است و از سوی دیگر با میانگین‌گیری از انرژی زیرباندهای درخت موجک می‌توانیم تقریبی از انرژی سیگنال خروجی بدست آوریم، از معیار زیر استفاده می‌کنیم:

$$E = |s|^2 - |\hat{s}|^2 \quad (21)$$

واضح است که با مینیمم کردن این رابطه، رابطه (۲۰) نیز مینیمم خواهد شد. بنابراین فیلتر پیشنهادی باید به گونه‌ای عمل کند که انرژی سیگنال خروجی هر چه بیشتر به $|s|^2$ نزدیک باشد. می‌دانیم:

$$|s|^2 = |x|^2 + |n|^2 - 2xn \quad (22)$$

بنابراین برای مینیمم شدن رابطه (۲۱)، مقدار $|\hat{s}|^2$ باید تابعی درجه دوم از $|x|$ و $|n|$ باشد. همچنین با توجه به خاصیت نویز جمع شونده n شرایط زیر باید برقرار باشند:

$$1) |x| = |n| \Rightarrow |\hat{s}|^2 = 0 \quad (23)$$

$$2) |n| = 0 \Rightarrow |\hat{s}|^2 = |x|^2 \quad (24)$$

$$3) 0 \leq |\hat{s}|^2 \quad (25)$$

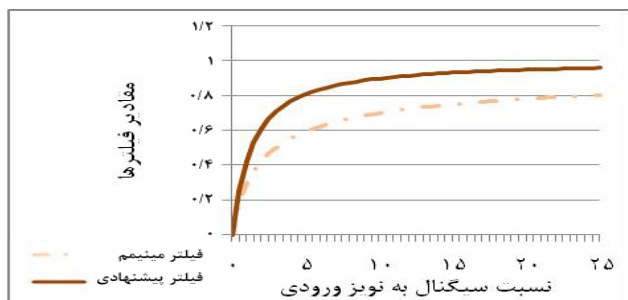
با توجه به این خواص، ضریبی که به این صورت تعریف می‌شود برای استفاده در رابطه (۳۳) مناسب می‌باشد.

۶-۲- فیلتر پیشنهاد دوم

با توجه به آنچه در بخش‌های قبل ذکر شد، از میان فیلترهای بیان شده در رابطه (۲۷)، به ازای $a = -1$ فیلتری با مقدار مینیمم بدست خواهد آمد. از آنجا که هر چه مقدار فیلتر کمتر باشد، ضرایب سیگنال را بیشتر کاهش می‌دهد، قادر است مقدار نویز بیشتری را از گفتار حذف کند. ولی عیب استفاده از این فیلتر، ایجاد اندکی اعوجاج در فریم‌های گفتار می‌باشد. برای حل این مشکل باید مقدار فیلتر را در این فریم‌ها اندکی افزایش دهیم. برای این منظور از فیلتری تواندار به صورت رابطه (۳۵) استفاده می‌کنیم که مشابه این فیلتر قبلاً در حوزه تبدیل فوریه مطرح شده و نتایج خوبی را به دنبال داشته است [۵].

$$h = h_{\min}^{1-h_{\min}} \quad (35)$$

شکل (۲) نمودار مقادیر دو فیلتر h و h_{\min} را نشان می‌دهد:



شکل ۲- نمودار مقادیر فیلتر مینیمم و فیلتر پیشنهادی دوم بر حسب میزان نسبت سیگنال به نویز ورودی

مقایسه این دو فیلتر نشان می‌دهد که در نسبت‌های سیگنال به نویز پایین تفاوت محسوسی بین دو فیلتر مشاهده نمی‌شود و هر چه مقدار نسبت سیگنال به نویز ورودی بیشتر باشد تفاوت بین این دو فیلتر بیشتر خواهد شد. بنابراین این فیلتر جدید در فریم‌های سکوت که نسبت سیگنال به نویز پایین‌تر است مقدار نویز را نسبت به فیلتر قبلی افزایش نمی‌دهد، در حالیکه در فریم‌های گفتار با افزایش مقدار فیلتر از ایجاد اعوجاج در سیگنال جلوگیری می‌کند.

۷- بهسازی گفتار با استفاده از فیلترهای پیشنهادی

مراحل انجام عملیات بهسازی گفتار بوسیله فیلتر پیشنهادی اول در شکل (۳) نشان داده شده است:

برای انجام عملیات بهسازی گفتار توسط هر کدام از فیلترهای پیشنهادی، ابتدا سیگنال ورودی را پنجره‌بندی کرده و سپس ضرایب تبدیل موجک آن را محاسبه می‌کنیم. در مرحله بعد مقدار انرژی نویز، به‌روز رسانی شده تا از آن برای بدست آوردن مقدار فیلتر استفاده شود. به‌روز رسانی مقدار انرژی نویز از طریق روش تخمین نویز پیوسته صورت می‌گیرد. روش کار به این صورت است که ابتدا یک تخمین اولیه از نویز در نظر می‌گیریم و سپس در هر مرحله با اعمال فیلتر بدست آمده از فریم قبلی، تخمین جدیدی از نویز بدست آورده و با ضریبی ثابت، آن را با تخمین انرژی نویز اولیه ترکیب می‌کنیم [۷]. سپس با داشتن انرژی نویز

کنیم که در فریم‌های سکوت سیگنال h_{\min} بیشترین تأثیر را داشته باشد، چون در این بخش هدف تنها حذف نویز از سیگنال است. از سوی دیگر در این فریم‌ها دوبار استفاده از این فیلتر، نویز بیشتری حذف خواهد شد و در نتیجه بهتری را شاهد خواهیم بود. می‌توان ثابت کرد که این کار معادل با استفاده از دوم این فیلتر است. زیرا از آنجا که با یکبار استفاده از این فیلتر در قسمت‌های سکوت، میزان انرژی سیگنال و نویز به یک نسبت کاهش می‌یابد، اگر بخواهیم بار دیگر از این فیلتر و با مقادیر جدید استفاده کنیم داریم:

$$\text{new_h_min}_i = \frac{|k||x_i| - |k||n_i|}{|k||x_i|} \quad (31)$$

حال با فاکتورگیری از $|k|$ و ساده کردن آن از صورت و مخرج داریم:

$$\text{new_h_min}_i = \frac{|x_i| - |n_i|}{|x_i|} = h_{\min}_i \quad (32)$$

بنابراین مقدار فیلتر با پارامترهای جدید تغییر نمی‌کند و می‌توانیم دوباره همان مقدار قبلی را استفاده کنیم. بنابراین دو بار استفاده از فیلتر معادل با استفاده از h_{\min}^2 خواهد بود. پس فیلتر پیشنهادی را به صورت ترکیب خطی h_{\max} و h_{\min}^2 تعریف می‌کنیم.

$$h_i = \alpha_i h_{\max}_i + (1 - \alpha_i) h_{\min}_i^2 \quad (33)$$

در این رابطه منظور از α_i ، ضریب فیلتر در زیرباند نام از درخت موجک می‌باشد.

۶-۱-۲- یافتن ضریب مناسب

با توجه به خواص دو فیلتر h_{\max} و h_{\min} ، ضریب α را باید به گونه‌ای انتخاب کنیم که در فریم‌های سکوت سیگنال، این مقدار به صفر میل کند با این هدف که فیلتر h_{\min}^2 بیشترین تأثیر را داشته باشد. از سوی دیگر این انتخاب باید به گونه‌ای باشد که در فریم‌های گفتار ضریب α به یک نزدیک شود تا تأثیر h_{\max} را افزایش دهد. با این توضیحات α را به صورت رابطه زیر تعریف می‌کنیم:

$$\alpha_i = \frac{(|x_i|^2 - |n_i|^2)}{|x_i|^2} \quad (34)$$

انتخاب α بصورت رابطه (۳۴)، دارای خواص زیر است:

- (۱) از آنجا که مقدار تفاضل انرژی سیگنال و نویز در هر فریم یکی از عوامل مشخص‌کننده سکوت یا گفتار بودن فریم است، انتخاب α به صورت تابعی از این مقدار، تا حدی نشان‌دهنده احتمال سکوت یا گفتار بودن فریم موردنظر خواهد بود.
- (۲) مقدار α بین ۰ و ۱ تغییر می‌کند.
- (۳) با انتخاب توان دوم، α در فریم‌های سکوت تا حد زیادی به صفر نزدیک می‌شود و تأثیر h_{\max} در این فریم‌ها به حداقل می‌رسد. در حالیکه در فریم‌های گفتار که کاهش نویز با کمترین اعوجاج هدف ماست، هر دو فیلتر اثرگذار خواهند بود.

در سیگنال بهبودیافته، دخیل هستند. این معیار، عددی بین ۰ و ۴/۵ به سیگنال ورودی تخصیص می‌دهد. هر چه این عدد به ۴/۵ نزدیکتر باشد، کیفیت سیگنال بهبودیافته به سیگنال گفتار تمیز نزدیکتر خواهد بود [۱۴].

جداول (۱)، (۲)، (۳)، (۴) و (۵)، میزان بهبود نسبت سیگنال به نویز ورودی در حضور نویز سفید، صورتی، ماشین، کارخانه و همه‌ها، با نرخ‌های سیگنال به نویز متفاوت را به کمک فیلتر مینیمم، ماکزیمم، وینر و فیلترهای پیشنهادی، نشان می‌دهد.

جدول ۱- نتایج ارزیابی میزان بهبود نسبت سیگنال به نویز ورودی برای نویز سفید

فیلتر پیشنهادی دوم	فیلتر پیشنهادی اول	فیلتر وینر	فیلتر ماکزیمم	فیلتر مینیمم	SNR ورودی
۱۰/۵۰۹ db	۱۰/۱۷ db	۹/۹۴۳ db	۷/۶۹۱ db	۹/۳۴۳ db	۰ db
۹/۳۸۳ db	۹/۳۶۷ db	۸/۹۹۱ db	۷/۱۱۲ db	۷/۲۳۲ db	۵ db
۸/۳۸۹ db	۸/۱۷۲ db	۸/۰۷۱ db	۶/۵۵ db	۶/۳۹۶ db	۱۰ db
۷/۵۶۸ db	۷/۴۶۲ db	۷/۳۳۵ db	۶/۰۳ db	۵/۰۷۱ db	۱۵ db
۶/۹۰۷ db	۶/۶۰۲ db	۶/۹۰۷ db	۵/۵۹۳ db	۴/۰۶۱ db	۲۰ db
۶/۳۹۷ db	۶/۱۰۳ db	۶/۲۱ db	۵/۲۴۲ db	۳/۲۶۹ db	۲۵ db

جدول ۲- نتایج ارزیابی میزان بهبود نسبت سیگنال به نویز برای نویز صورتی

فیلتر پیشنهادی دوم	فیلتر پیشنهادی اول	فیلتر وینر	فیلتر ماکزیمم	فیلتر مینیمم	SNR ورودی
۸/۴۰۲ db	۸/۵۶۵ db	۸/۴۰۳ db	۶/۵۷۸ db	۷/۸۱۷ db	۰ db
۷/۸۶۶ db	۷/۸۶۹ db	۷/۸۶۶ db	۶/۱۰۹ db	۶/۵۱۶ db	۵ db
۷/۴۲۲ db	۷/۳۰۵ db	۷/۴۲۲ db	۵/۷۴۱ db	۵/۵۵ db	۱۰ db
۶/۸۹۷ db	۵/۸۴۷ db	۶/۸۹۷ db	۵/۴۳ db	۴/۸۱۹ db	۱۵ db
۶/۳۳۱ db	۶/۴۵۷ db	۶/۳۳۱ db	۵/۱۷۹ db	۴/۳۴۱ db	۲۰ db
۵/۸۷ db	۶/۱۰۳ db	۵/۸۷ db	۴/۹۲۶ db	۳/۷۵۸ db	۲۵ db

جدول ۳- نتایج ارزیابی میزان بهبود نسبت سیگنال به نویز ورودی برای نویز ماشین

فیلتر پیشنهادی دوم	فیلتر پیشنهادی اول	فیلتر وینر	فیلتر ماکزیمم	فیلتر مینیمم	SNR ورودی
۷/۶۴۲ db	۷/۸۸۶ db	۷/۵۴۸ db	۶/۶۸۲ db	۶/۳۰۷ db	۰ db
۶/۸۵۲ db	۶/۲۹۵ db	۶/۷۸۷ db	۶/۱۱۴ db	۴/۷۰۶ db	۵ db
۶/۳۳۴ db	۶/۷۸۵ db	۶/۲۷۷ db	۵/۷۱۱ db	۳/۴۹۹ db	۱۰ db
۵/۹۹۷ db	۶/۳۰۶ db	۵/۹۴۴ db	۵/۴۳۷ db	۲/۴۹۴ db	۱۵ db
۵/۷۵۷ db	۵/۸۵۹ db	۵/۶۹۹ db	۵/۲۱ db	۲/۷۰۵ db	۲۰ db
۵/۵۲۲ db	۵/۴۱۶ db	۵/۴۴۷ db	۴/۹۵۴ db	۲/۰۸ db	۲۵ db

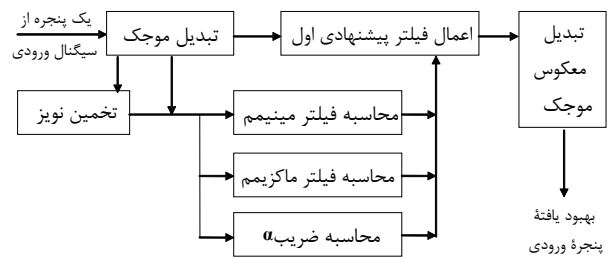
جدول ۴- نتایج ارزیابی میزان بهبود نسبت سیگنال به نویز ورودی برای نویز کارخانه

فیلتر پیشنهادی دوم	فیلتر پیشنهادی اول	فیلتر وینر	فیلتر ماکزیمم	فیلتر مینیمم	SNR ورودی
۶/۹ db	۷/۰۵۸ db	۶/۰۱۹ db	۴/۳۳۳ db	۷/۰۵ db	۰ db
۶/۴۷۳ db	۶/۲۷۵ db	۵/۶۱ db	۴/۱ db	۶/۰۴۹ db	۵ db
۶/۱۲۲ db	۵/۸۰۱ db	۵/۲۸۸ db	۳/۹۱۴ db	۵/۴۶۸ db	۱۰ db
۵/۸۳۳ db	۵/۴۱۱ db	۵/۰۲۵ db	۳/۷۵۴ db	۴/۸۵۸ db	۱۵ db
۵/۶۱۲ db	۵/۰۸۷ db	۴/۷۹۲ db	۲/۶۰۹ db	۴/۳۶۱ db	۲۰ db
۵/۳۸۳ db	۴/۷۹۳ db	۴/۵۶۳ db	۴/۵۶۳ db	۳/۹۲۲ db	۲۵ db

جدول ۵- نتایج ارزیابی میزان بهبود نسبت سیگنال به نویز ورودی برای نویز همه‌ها

فیلتر پیشنهادی دوم	فیلتر پیشنهادی اول	فیلتر وینر	فیلتر ماکزیمم	فیلتر مینیمم	SNR ورودی
۵/۸۶۱ db	۷/۱۷۷ db	۴/۹۸۷ db	۳/۲۰۵ db	۶/۸۶۶ db	۰ db
۵/۶۵۱ db	۷/۳۹۸ db	۴/۷۷۲ db	۳/۰۸۱ db	۵/۹۸۵ db	۵ db
۵/۵۱۱ db	۷/۱۱۷ db	۴/۶۰۷ db	۲/۹۸۴ db	۵/۲۵۴ db	۱۰ db
۵/۳۸۸ db	۶/۷۹۵ db	۴/۴۶ db	۲/۸۹۸ db	۴/۶۱۹ db	۱۵ db
۵/۲۶۲ db	۶/۴۲۱ db	۴/۳۲۶ db	۲/۸۱۹ db	۴/۰۶۶ db	۲۰ db
۵/۱۱۲ db	۶/۰۴۵ db	۴/۱۸۶ db	۲/۷۲۴ db	۳/۶۰۲ db	۲۵ db

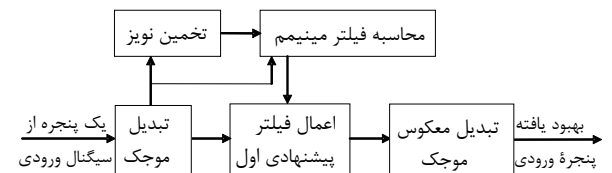
مقدار فیلترهای پایه و در مرحله بعد، مقدار فیلتر نهایی را در هر زیرباند محاسبه کرده و آن را بر ضرایب موجک اعمال می‌کنیم. سپس با کمک تبدیل معکوس موجک، فریم‌های بهبودیافته حاصل می‌شوند. با همپوشانی و جمع این فریم‌ها سیگنال بهسازی شده نهایی حاصل می‌شود.



شکل ۳- سیستم کاهش نویز با استفاده از فیلتر پیشنهادی اول

شکل (۴) مراحل انجام این عملیات را برای فیلتر پیشنهادی دوم نشان

می‌دهد:



شکل ۴- سیستم کاهش نویز با استفاده از فیلتر پیشنهادی دوم

۸- نتایج پیاده‌سازی

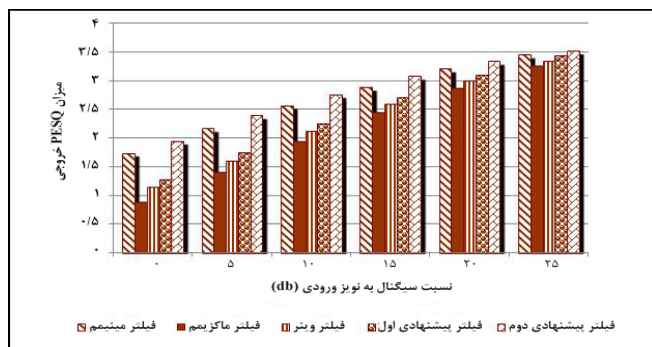
در این قسمت به مقایسه کارایی فیلترهای پیشنهادی با فیلترهای مینیمم، ماکزیمم و فیلتر وینر، پرداخته شده است.

تعداد سطوح درخت آنالیز، برای هر دو فیلتر بانک مورد آزمایش، سه سطح می‌باشد. مقایسات بر روی شش سیگنال گفتار تمیز مختلف (مربوط به سه زن و سه مرد) از پایگاه داده TIMIT انجام شده‌اند. برای ساخت سیگنال نویزی از پنج نوع نویز سفید، صورتی، همه‌ها، کارخانه و ماشین با نسبت‌های سیگنال به نویز ۰، ۵، ۱۰، ۱۵، ۲۰ و ۲۵ دسیبل، استفاده شده است و برای بدست آوردن نتایج بین ۶ جمله متوسط گیری شده است. برای طول قاب از اندازه ۲۵۶ نمونه استفاده شده است، همپوشانی میان قاب‌ها ۵۰٪ می‌باشد.

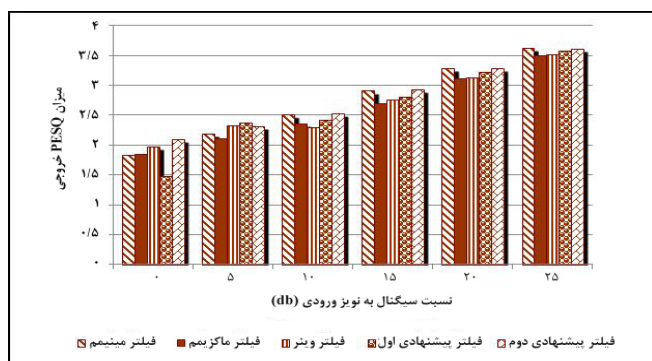
به منظور انجام مقایسات از معیارهای عینی نسبت سیگنال به نویز (SNR) و معیار ارزیابی شنیداری کیفیت گفتار (PESQ) استفاده شده است. برای محاسبه معیار عینی نسبت سیگنال به نویز از رابطه زیر استفاده شده است:

$$SNR = \frac{\sum_{i=1}^{SignalLength} S(i)^2}{\sum_{i=1}^{SignalLength} [S(i) - \hat{S}(i)]^2} \quad (36)$$

در رابطه بالا S و \hat{S} به ترتیب سیگنال تمیز و سیگنال بهبود یافته می‌باشند. معیار PESQ، یکی از معیارهای ارزیابی شنیداری کیفیت گفتار است که در محاسبه آن پارامترهای مختلفی از قبیل میزان اعوجاج، اکو، خفگی و نویز باقیمانده



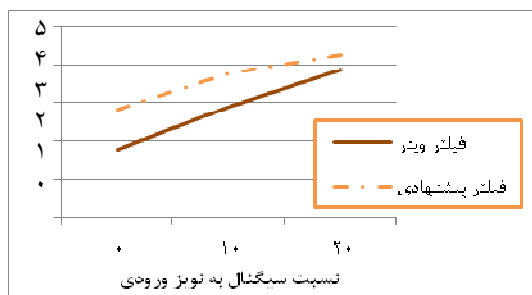
شکل ۸- مقایسه فیلترها از لحاظ میزان PESQ خروجی در حضور نویز جمع‌شونده کارخانه



شکل ۹- مقایسه فیلترها از لحاظ میزان PESQ خروجی در حضور نویز جمع‌شونده همه‌مه

گرچه معیار PESQ در برخی موارد برتری فیلتر مینیمم را نشان می‌دهد، ولی با توجه به اینکه این فیلتر با ارزیابی صورت گرفته توسط معیار نسبت سیگنال به نویز برتر قابل توجهی را نشان نمی‌دهد، نمی‌توان آن را برتر دانست. با این توضیح می‌توان گفت که همانند نتایج معیار نسبت سیگنال به نویز، فیلتر پیشنهادی اول بهترین نتیجه را داشته و به‌طور متوسط حدود ۰/۵ واحد برتری را نشان می‌دهد. بعد از این فیلتر، فیلتر پیشنهادی دوم به خصوص در نسبت‌های سیگنال به نویز بالاتر بهترین نتیجه را داشته و به‌طور متوسط ۰/۲ واحد برتری را نشان می‌دهد.

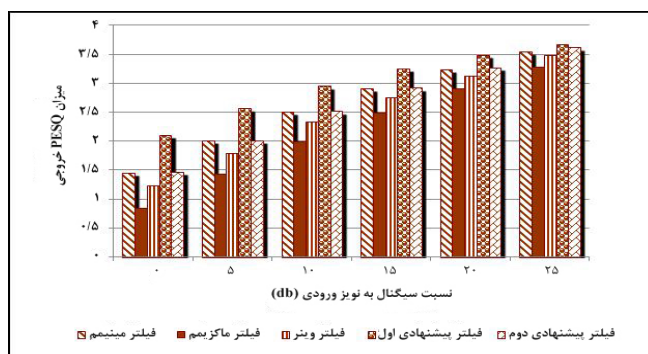
با توجه به برتری فیلتر پیشنهادی اول، تست شنوایی را بر روی فیلتر پیشنهادی اول و در مقایسه با فیلتر وینر انجام داده‌ایم. آزمایش‌ها بر روی دو سیگنال (۱ مرد و ۱ زن) از پایگاه داده TIMIT انجام شده که با سه نوع نویز سفید، ماشین و همه‌مه تخریب شده‌اند. شکل‌های (۱۰)، (۱۱) و (۱۲) نتایج مربوط به این آزمایش را نشان می‌دهد. بطوریکه هر چه مقدار نویز باقیمانده کمتر بوده، عدد بالاتری به آن نسبت داده شده است.



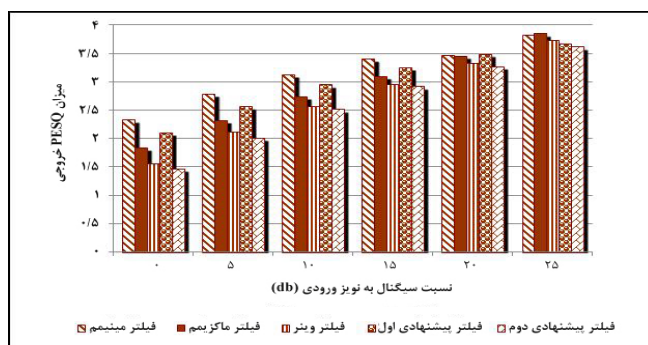
شکل ۱۰- مقایسه دو فیلتر وینر و فیلتر پیشنهادی اول، از لحاظ مقدار نویز باقیمانده کمتر در حضور نویز جمع‌شونده سفید

نتایج حاکی از آن است که فیلتر پیشنهادی اول، تقریباً تمام موارد بر دیگر فیلترها برتری داشته است بطوریکه، مثلاً در مورد نویز همه‌مه این تفاوت بین ۱ تا ۴ دسیبل بوده است. به‌طور متوسط می‌توان گفت که فیلتر پیشنهادی دوم حدود ۲ دسیبل از دیگر فیلترها برتر بوده است. همچنین، می‌توان دید که این برتری در نسبت‌های سیگنال به نویز پایین، محسوس‌تر بوده و کارایی فیلتر بالاتر است. در مورد فیلتر پیشنهادی دوم هم می‌توان گفت که نسبت به دیگر فیلترها، غیر از فیلتر پیشنهادی دوم، برتر بوده است و به‌طور میانگین حدود ۱ دسیبل برتری داشته است. این فیلتر بر خلاف فیلتر پیشنهادی اول، در نسبت‌های سیگنال به نویز بالاتر بهتر عمل می‌کند.

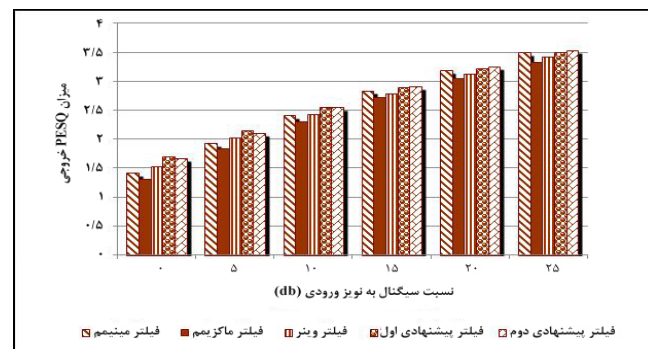
در ادامه به بررسی فیلترهای پیشنهادی با معیار PESQ خواهیم پرداخت که نتایج آن در شکل‌های (۵)، (۶)، (۷)، (۸) و (۹) نمایش داده شده است.



شکل ۵- مقایسه فیلترها از لحاظ میزان PESQ خروجی در حضور نویز جمع‌شونده سفید



شکل ۶- مقایسه فیلترها از لحاظ میزان PESQ خروجی در حضور نویز جمع‌شونده صوتی



شکل ۷- مقایسه فیلترها از لحاظ میزان PESQ خروجی در حضور نویز جمع‌شونده ماشین

کارایی فیلترهای پیشنهادی را بیش از این بهبود بخشید که این کار در حیطه فعالیت‌های پژوهشی آتی قرار خواهد گرفت.

مراجع

[1] H. Sameti, and L. Deng, "Nonstationary-State Hidden Markov Model Representation of Speech Signals for Speech Enhancement," *Signal Processing*, vol. 82, no. 2, pp. 205-227, 2002.

[2] J. Chen, J. Benesty, Y. Huang, and S. Doclo, "New Insights Into the Noise Reduction Wiener Filter," *IEEE Transaction on Audio, Speech and Language Processing*, vol. 14, no. 4, pp. 1218-1234, 2006.

[3] T. K. Kumar, "Optimum Cascaded Design for Speech Enhancement Using Kalman Filter," *Proc. World Academy of Science, Engineering and Technology*, 2008.

[4] T. Lotter, P. Vary, "Noise Reduction By Maximum A Posteriori Spectral Amplitude Estimation With Supergaussian Speech Modeling," *Proc. of the International Workshop on Acoustic Echo and Noise Control*, pp. 83-86, 2003.

[5] B. Ayad, "A Noise Suppressor," <http://www.freepatentsonline.com/EP1242992>.

[6] S. Tabibian, and A. Akbari, "Noise Reduction from Speech Signal based on Wavelet Transform and Kullback-Leibler Divergence," *Proc. of the International Symposium on Telecommunications*, pp. 787-791, 2008.

[7] M. Johnson, X. Yuan, and Y. Ren, "Speech Signal Enhancement through Adaptive Wavelet Thresholding," *Speech Communication*, vol. 49, no. 2, pp. 123-133, 2007.

[8] Q. Fu, and E. Wan, "Perceptual Wavelet Adaptive Denoising of Speech," *Proc. of the Euro Conference on Speech Communication and Technology*, pp. 577-580, 2003.

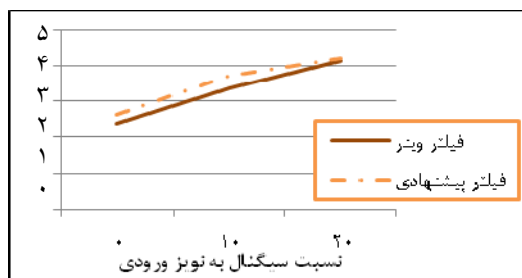
[9] C. A. Medina, and R. Alcaim, "Wavelet Denoising Of Speech Using Neural Networks For Threshold Selection," *Electronics Letters*, vol. 39, no. 25, pp. 1869-1871, 2003.

[10] M. Bahoura, and J. Rouat, "A New Approach for Wavelet Speech Enhancement," *Proc. of the Euro Conference on Speech Communication and Technology*, pp. 1937-1940, 2001.

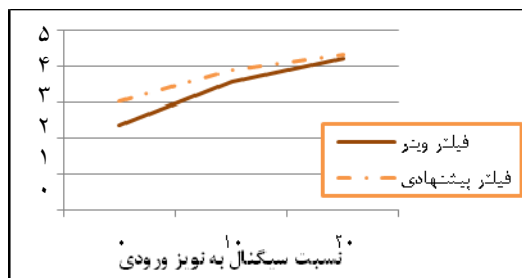
[11] Y. Shao, and C. Chang, "A Generalized Time-Frequency Subtraction Method For Robust Speech Enhancement Based On Wavelet Filter Banks Modeling Of Human Auditory System," *IEEE Transactions On Systems, Man, And Cybernetics*, vol. 37, no. 4, pp. 877-889, 2007.

[12] M. Rahmani, M. Mohammadi, and A. Akbari, "Background Noise Control for Speech Enhancement," *Proc. of the Conference on Computer Society of Iran*, 2006.

[13] F. Ykhlef, A. Guessoum, and D. Berkani, "Combined Spectral Subtraction and Wiener Filter Methods in Wavelet



شکل ۱۱- مقایسه دو فیلتر وینر و فیلتر پیشنهادی اول، از لحاظ مقدار نویز باقیمانده کمتر در حضور نویز جمع‌شونده ماشین



شکل ۱۲- مقایسه دو فیلتر وینر و فیلتر پیشنهادی اول، از لحاظ مقدار نویز باقیمانده کمتر در حضور نویز جمع‌شونده مهمه

مشاهده می‌شود که در هر سه نویز فیلتر پیشنهادی نسبت به فیلتر وینر به‌خصوص در مورد نسبت‌های سیگنال به نویز پایین، نتیجه بهتری را داشته است، بطوریکه در نسبت سیگنال به نویز ۰، حداقل ۰/۲ و حداکثر ۱/۱ واحد افزایش را شاهد هستیم.

مقایسه این دو فیلتر از لحاظ قابلیت فهم، تفاوت قابل ملاحظه‌ای را نشان نمی‌دهد و اکثر شنوندگان به هر دو فیلتر، مقادیر مساوی نسبت داده‌اند.

همچنین، در تست‌های شنوایی انجام شده، ۸۲٪ از شنوندگان سیگنال بهسازی شده با کمک فیلتر پیشنهادی را ترجیح داده‌اند و فقط در ۴٪ موارد، فیلتر وینر برتری داده شده است.

با توجه به این نتایج می‌توان گفت که فیلتر پیشنهادی اول، بهترین نتیجه را نسبت به دیگر فیلترها، به‌خصوص در نسبت‌های سیگنال به نویز پایین‌تر داراست.

۹- نتیجه‌گیری

در این مقاله مجموعه فیلترهای قابل استفاده در حوزه تبدیل موجک، با کمک روابط ریاضی، معرفی شدند و با کمک فیلترهای این مجموعه، دو فیلتر جدید در این حوزه تعریف شد که فیلتر اول از ترکیب دو فیلتر مینیمم و ماکزیمم و دیگری از تواندار کردن فیلتر مینیمم حاصل شده است. ارزیابی‌ها، برتری فیلتر پیشنهادی اول را به خصوص در مورد نسبت‌های سیگنال به نویز پایین تأیید می‌کند. بطوریکه افزایش حدود ۲ دسیبل، در میزان بهبود نسبت سیگنال به نویز، مشاهده می‌شود. میزان بهبود معیار PESQ نیز با بکارگیری فیلتر پیشنهادی اول، در حدود ۰/۵ واحد افزایش داشته است.

در نسبت‌های سیگنال بالاتر، کارایی خوب فیلتر پیشنهادی دوم محسوس است، به‌طوریکه افزایش حدود ۲ دسیبلی در نسبت سیگنال به نویز و حدود ۰/۲ واحد افزایش در مورد میزان PESQ خروجی، مشاهده می‌شود.

به نظر می‌رسد با توجه به نتایج خوب این فیلترها، بتوان با انتخاب α بهینه‌تر، در مورد فیلتر پیشنهادی اول و انتخاب مناسب‌تر توان فیلتر، در مورد فیلتر پیشنهاد دوم،



هدی کاویانی دارای مدرک کارشناسی ریاضی محض و فارغ‌التحصیل مقطع کارشناسی ارشد رشته مهندسی کامپیوتر گرایش هوش مصنوعی و رباتیک از دانشگاه علم و صنعت ایران می‌باشد که تحقیقات او در زمینه بهسازی گفتار است.

آدرس پست‌الکترونیکی ایشان عبارت است از:

hoda.kaviani@gmail.com



شیمای طبیبیان دانشجوی مقطع دکترای رشته هوش مصنوعی و رباتیک در دانشگاه علم و صنعت ایران می‌باشد. تحقیقات او در زمینه پردازش گفتار است که بر روی موضوعات بهسازی گفتار و واژه‌یابی گفتار تمرکز بیشتری یافته است.

آدرس پست‌الکترونیکی ایشان عبارت است از:

shimatabibian@iust.ac.ir



احمد اکبری دانشیار دانشگاه علم و صنعت در رشته مهندسی کامپیوتر می‌باشد. او مدرک دکترای خود را در سال ۱۹۹۵ از دانشگاه رن فرانسه اخذ کرده است. زمینه کاری او بر روی تحلیل سیگنال گفتار، بازشناسی گفتار گسسته، بهسازی گفتار، سیستم‌های گفتگو، واسط ارتباط انسان و کامپیوتر مبتنی بر گفتار، کاربرد صدا روی وب، شبکه‌های کامپیوتری و امنیت شبکه متمرکز شده است.

آدرس پست‌الکترونیکی ایشان عبارت است از:

akbari@iust.ac.ir

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۸/۳/۲۵

تاریخ اصلاح: ۸۹/۵/۲۶

تاریخ قبول شدن: ۸۹/۷/۶

نویسنده مرتبط: دکتر احمد اکبری، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران.

"Domain for Noise Reduction," *Proc. of the International Symposium on Control, Communications, and Signal Processing*, 2006.

[14] A. Graps, "An Introduction to Wavelet", *IEEE Computational Science and Engineering*, vol. 2, no. 2, pp. 57-61, 1995.

[15] "Objective Measurement of Active Speech Level," *ITU-T, Geneva*, P. 56, March 1993.

ضمیمه الف

در اینجا قصد داریم انرژی سیگنال خروجی را به گونه‌ای تعریف کنیم که سه شرط مورد نظر برقرار باشد:

$$1) |x| = |n| \Rightarrow |\hat{s}|^2 = 0 \quad (\text{الف-۱})$$

$$2) |n| = 0 \Rightarrow |\hat{s}|^2 = |x|^2 \quad (\text{الف-۲})$$

$$3) 0 \leq |\hat{s}|^2 \quad (\text{الف-۳})$$

از رابطه (الف-۱) می‌توان نتیجه گرفت:

$$|\hat{s}|^2 = (|x| - |n|) \times P \quad (\text{الف-۴})$$

که منظور از P تابعی درجه اول از |x| و |n| است. حال برای برآورده شدن شرط دوم، P باید به صورت زیر تعریف شود:

$$P = (|x| + a|n|) \quad (\text{الف-۳})$$

بنابراین:

$$|\hat{s}|^2 = (|x| - |n|) \times (|x| + a|n|) \quad (\text{الف-۴})$$

در ادامه برای بدست آوردن مقدار a، از شرط سوم کمک می‌گیریم که نتیجه می‌شود:

$$-1 \leq a \leq 1 \quad (\text{الف-۵})$$

بنابراین انرژی سیگنال خروجی باید به صورت زیر باشد:

$$|\hat{s}|^2 = (|x| - |n|)(|x| + a|n|); -1 \leq a \leq 1 \quad (\text{الف-۶})$$

راهنمای تهیه مقالات



نشریه علمی پژوهشی انجمن کامپیوتر ایران

نسخه نهائی مقالات ارسالی برای چاپ در نشریه علوم و مهندسی کامپیوتر باید بر طبق اصول مطرح شده در این راهنما تهیه شده باشد. رعایت این اصول در نسخه اولیه نیز قویاً توصیه می‌شود. مقالات می‌تواند به یکی از دو زبان فارسی یا انگلیسی ارسال گردد.

۱- ساختار مقاله

- عنوان: کوتاه و معرف محتوای مقاله باشد و از ۱۵ کلمه تجاوز نکند.
- نام نویسندگان و مؤسسه محل اشتغال آنان: از ذکر عناوین خودداری شود.
- چکیده فارسی: حاوی تعریف مسأله، روش حل، و نتایج مهم باشد و از ۱۵۰ کلمه تجاوز نکند.
- واژه‌های کلیدی: حداکثر ۱۰ کلمه
- بدنه اصلی مقاله: بدنه اصلی با «مقدمه» شروع و با «نتیجه‌گیری» خاتمه می‌یابد. بخش‌ها و زیربخش‌های بدنه اصلی باید شماره‌گذاری شوند.
- شماره «مقدمه» یک خواهد بود.
- تشکر و قدردانی (در صورت نیاز).
- مراجع: مراجع به ترتیبی که در متن به آنها رجوع می‌شود آورده شوند. نام مؤلفان مراجع در صورت لزوم در متن بصورت فارسی آورده شود. رجوع به مراجع با ذکر شماره آنها در داخل کروشه ([]) انجام پذیرد.
- پیوست‌ها (در صورت نیاز)
- واژه‌نامه (در صورت نیاز)
- برای مقالات فارسی، عنوان مقاله، نام نویسندگان، مؤسسه محل اشتغال، چکیده، و واژه‌های کلیدی به زبان انگلیسی نیز در صفحه‌ای جداگانه داده شود. برای مقالات انگلیسی این اطلاعات به زبان فارسی نیز در صفحه‌ای جداگانه داده شود.

۲- معادله‌ها، شکل‌ها، جدول‌ها، و عکس‌ها

- معادله‌ها باید با فاصله کافی از بالا و پائین تایپ و به صورت متوالی شماره‌گذاری شوند. شماره معادله در پرانتز در انتهای سمت راست سطر حاوی معادله قرار داده شود. معادلات دستنویس به هیچ شکل قابل قبول نیستند.
- شکل‌ها و جدول‌ها باید دارای شماره و عنوان باشند. در شکل‌ها شماره و عنوان در زیر شکل و در جدول‌ها در بالای شکل قرار می‌گیرد. اعداد و متون روی شکل‌ها و جدول‌ها باید دارای اندازه مناسب و کاملاً خوانا باشند.
- اعداد و کلمات روی شکل‌ها و جدول‌ها در مقالات فارسی به زبان فارسی باشند.
- عکس‌ها سیاه و سفید، برقی، و با کیفیت عالی باشند.

۳- نحوه نگارش مراجع

در لیست مراجع انواع مختلف مرجع‌ها به شکل زیر نوشته شوند:

- [۱] ب. مقدم، ا. تقوی، و ن. طاهری، **آشنائی با شبکه‌های کامپیوتری**، چاپ دوم، انتشارات نصر، تهران، ۱۳۷۵.
- [۲] ی. براون، **مقدمه‌ای بر شبکه‌های عصبی**، ترجمه م. ع. آرام، انتشارات فجر، مشهد، ۱۳۷۰.
- [۳] **راهنمای کاربران حسابرس**، شرکت پردازش رایانه‌ای ایران، تهران، ۱۳۶۵.
- [۴] ج. عارف، **استنتاج فازی بوسیله شبکه‌های عصبی**، پایان‌نامه کارشناسی ارشد هوش مصنوعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شهر، ۱۳۷۴.
- [۵] ج. حسینی، و ح. ربانی، «تشخیص چهره انسان در تصویر»، **نشریه امیرکبیر**، سال هشتم، شماره ۴۲، ص ۱۲۵-۱۴۷، ۱۳۷۷.

[۶] ج. حسینی، و.ح. ربانی، «تشخیص چهره انسان در تصویر»، در مجموعه مقالات هفتمین کنفرانس سالانه انجمن کامپیوتر ایران، ص ۲۲۴-۲۳۲، ۱۳۸۰.

- [7] M. A. Ahmadi, and M. H. Rahimi, *Fuzzy Set Theory*, New Jersey: Prentice-Hall, 1995.
- [8] M. A. Ahmadi, M. H. Rahimi, and A. Fatemi, "Evidence-Based Recognition of 3D Objects," *IEEE Trans. Patt. Anal. and Mach. Intell.*, vol. 12, no. 10, pp. 811-835, 1994.
- [9] A. Taheri, "On-Line Fingerprint Verification," *Proc. IEEE Intl Conf. Pattern Recognition*, pp. 752-758, 1992.
- [10] M. A. Ahmadi, *On-line Fingerprint Verification*, Ph. D. Dissertation, MIT, Cambridge, MA, 1982.
- [11] A. J. Washington, "The Fingerprint of MalcomX," <http://www.dermatoglyphics.com>, June 2003.
- [12] International Biometrics Group, <http://www.biometricgroup.com>, May 2003.

۴- واژه‌نامه

هر واژه خارجی در واژه‌نامه در انتهای مقاله با شماره‌ای مشخص شود و شماره در معادل فارسی آن واژه در متن، بصورت بالانویس آورده شود.

۵- آماده‌سازی مقاله

- مقاله را با نرم افزار WORD XP تحت ویندوز XP تایپ و با چاپگر لیزری روی کاغذ A4 چاپ نمائید.
- متن چکیده به صورت تک ستونی با طول سطر ۱۸ سانتیمتر و متن مقاله به صورت دو ستونی با طول هر ستون ۸۷ میلیمتر و فاصله دو ستون ۶ میلیمتر تایپ شود. حاشیه‌ها از بالا و پائین برابر ۲۰ میلیمتر و از طرفین برابر ۱۵ میلیمتر اختیار شود.
- فاصله عنوان مقاله در صفحه اول از بالای صفحه برابر ۸۵ میلیمتر باشد و عنوان وسط چین شود.
- کلیه عناوین بصورت پررنگ با قلم «**Nazanin** یا **B Nazanin**» تایپ شوند، اندازه قلم عنوان مقاله ۱۸، عناوین سطح اول ۱۴، و عناوین سطح دوم و سوم ۱۲ انتخاب شوند.
- متن چکیده‌ها با قلم «**Nazanin** یا **B Nazanin**» اندازه ۹، متن مقاله با قلم «**Nazanin** یا **B Nazanin**» اندازه ۱۰، و کلمات و متن انگلیسی با قلم **Times New Roman** اندازه ۹ تایپ شوند.
- تمام متن بصورت تک فاصله تایپ شود. اسامی نویسندگان از عنوان مقاله و اسامی نویسندگان از عناوین محل اشتغال نویسندگان دو خط فاصله داشته باشد. بالای هر عنوان یک سطر فاصله قرار داده شود.
- سعی شود تعداد صفحات مقاله از ۳۰ صفحه بیشتر نباشد.

۶- نحوه ارسال مقاله

- شش نسخه از مقاله چاپ شده با چاپگر لیزری و دارای کیفیت عالی به دفتر نشریه ارسال گردد. ارسال اولیه مقاله می‌تواند از طریق پست الکترونیکی نیز به آدرس سردبیر انجام شود، اما از طریق دورنگار قابل قبول نیست.
- مقاله ارسالی برای نشریه علوم و مهندسی کامپیوتر نباید در جای دیگری به چاپ رسیده باشد و یا در زمان بررسی توسط نشریه برای چاپ به نشریه دیگری ارسال گردد. نویسندگان باید به‌مراه مقاله نامه‌ای دال بر این مطلب نیز ارسال دارند.
- پس از قبول مقاله، نسخه نهائی تصحیح شده مقاله باید هم بصورت چاپ شده و هم در فایل کامپیوتری در قالب‌های DOC و PDF روی یک دیسکت به نشریه ارسال گردد. همچنین فرم «**اختصاص حق انتشار**» باید توسط نویسندگان تکمیل، امضاء و ارسال شود.
- در نسخه نهائی باید بیوگرافی و عکس کلیه مؤلفین نیز در انتهای مقاله قرار داده شود.

- [2] M. A. Ahadi, M. H. Rahimi, and A. Fatemi, "Evidence-Based Recognition of 3D Objects," *IEEE Trans. Pattern Analysis and Mach. Intell.*, Vol. 12, No. 10, pp. 18-25, 1994.
- [3] A. Taheri, "On-line Fingerprint verification," *Proc, IEEE Int'l Conf. Pattern Recognition*, pp. 752-759, 1992.
- [4] M. A. Ahadi, *On-line Fingerprint verification*, Ph.D. Dissertation, University of Tehran, Tehran, Iran, 1994.
- [5] M. A. Washington, "The Fingerprint of Malcom X," <http://www.dermatologyphicisc.com>, June 2004.
- [6] International Biometrics Group, <http://www.Biometrics.com>, May 2003.

4. Manuscript preparation

Typing and Printout: The manuscript should be prepared using Microsoft Word and should be printed on A4 size paper using a laser printer.

Fonts: Use Times New Roman font type. The font size must be 9 for the Abstract and 10 for the main text. All the titles should be made bold. The paper title must have a font size of 18, first-level sub-titles a size of 14, second-and third-level sub-titles a size of 12.

Layout: The paper should be typed in 2-column single space format. The top margin for the first page should be equal to 85mm, each column length should be 87mm, and the spacing between columns 6mm. Only the abstract is to be typed in a single column format. Two space line must be allowed between the paper title and authors' names, and one space line should be allowed above each section title or sub-title.

5. Paper Submission

Exclusive Submission: Submission of a paper to JCSE implies that it has not been published previously, that it is not under consideration for publication elsewhere, and that if accepted, it will not be published elsewhere in any language. Explicit announcement of this matter must be made in a letter to the editor.

Paper Length: The paper is expected to be no longer than 30 pages.

Submission Process: Authors are requested to submit their papers electronically in PDF format to the editor-in-chief at csi-jcse@ipm.ir with message subject of "Paper Submission". All other relevant correspondence should also be addressed to the this email.

Review Process: Each manuscript will be reviewed by at least 3 reviewers and their constructive criticism will be forwarded to the authors.

Final Manuscript: After the acceptance of a paper, the authors should provide the Journal with two copies of the final manuscript and its Word computer file on a diskette. In addition, the authors should sign and submit the form for the transfer of copyright to the Computer Society of Iran.

INFORMATION FOR AUTHORS

The final manuscript of the English papers accepted for publication in the CSI JCSE should be prepared in accordance with this guide. It is, however, strongly recommended that preparation of the initial manuscript also follows the instructions given here. Manuscript may be in English or Farsi.

1. Paper Organization

- Title: The title should be short (at most 15 words) and indicative of the paper contents.
- Authors: the authors' names (initial and last name) and their affiliations should appear next.
- Abstract: the abstract should include the problem explanation, methods used for solution, and the significant results; and should not be longer than 150 words.
- Keywords: The keywords should be relatively independent and together optimally characterize the paper. Include 5 to 10 keywords.
- Text: the main body of the paper should start with Introduction and end with concluding remarks. All sections and sub-sections should be numbered. The number for introduction is 1.
- Acknowledgement: If required, Acknowledgements appear after the concluding remarks.
- References: All publications cited in the text should be presented in the order they are cited in a numbered list of References following Acknowledgements. In the text refer to references by the reference number in square brackets on the line.
- Appendices: If appendices are necessary, they are placed after the list of References.
- Farsi Section: the title, authors' names and affiliations, abstract, and keywords should also be given in Farsi a separate sheets (for papers written in English). For the non-Farsi speaking authors, the journal will supply this section.

2. Figures, Tables, Photographs and Equations

- Figures and Tables: Each figure or table must have a number and a caption. In figures, the number and the caption appear under the figure while in tables, they appear over it. The size of text and numbers in tables and figures must be suitable to allow high legibility. Do not use any type of shading in computer generated illustrations.
- Photographs: High quality glossy black and white photographs must be supplied as they are to be reproduced.
- Equations: Equations are to be numbered consecutively. The number of each equation should appear in parentheses in the right-most end of the equation line. Sufficient space should be allowed above and below each equation.

3. References

The format for various types of references should be as follows:

- [1] M. A. Ahadi, and M. H. Rahimi, *Fuzzy Set Theory*, New Jersey, Prentice-Hall, 1995.

Introducing Speech Enhancement Filters in Wavelet Domain

Hoda kaviani Shahrezaei

Shima Tabibian

Ahmad Akbari

Department of Computer Engineering, Iran University of Science and Technology, Tehran, Iran

ABSTRACT

Speech enhancement methods have been developed in both frequency and time domains. Wavelet transform based methods and Fourier transform based techniques are examples for frequency domain. Speech enhancement in wavelet domain is achieved through the thresholding or filtering noisy wavelet coefficients. In this paper, a wavelet based speech enhancement method is presented that uses filtering approaches for removing additive noise. For this purpose, a set of suitable filters in the wavelet domain is introduced. In the next stage, using these filters, two new filters for applying on noisy wavelet coefficients are introduced. The first filter is obtained by combination of two other filters. These two filters have minimum and maximum values in comparison to the values of other filters in the introduced set of filters. The Second filter is achieved by raising the minimum filter to a predefined power. These filters remove most of the noise in the silence frames and minimize the signal distortion in the speech frames.

Keywords: Speech Enhancement, Wavelet Transform, Filtering, Residual Noise and Signal.

Protection of the AES Algorithm Against Differential Power Analysis by Using Randomization in Composite Fields

Massoud Masoumi¹

Seyed Mojtaba Dehnavi²

¹Engineering Department, Islamshahr Azad University, Tehran, Iran

²Arithmetic and Computer Science Department, Tarbiat Moalem University, Tehran, Iran

ABSTRACT

This paper describes a novel approach for implementation of the AES algorithm which provides a significantly improved strength against differential power analysis with a minimal additional hardware overhead. The new proposed method is based on randomization in composite field arithmetic which entails an area penalty of only 7% while does not decrease the working frequency, does not alter the algorithm and keeps perfect compatibility with the published standard. The efficiency of the proposed technique was verified by practical results obtained from real implementation on a Xilinx Spartan-II FPGA.

Keywords: AES Algorithm, Differential Power Analysis Attack, Hardware Implementation, Composite Field Arithmetic.

Cellular Learning Automata in Image Processing Applications

Ahmad Ali Abin

Seyed Hamid Amiri

Hamid Beigy

Department of Computer Engineering, Sharif University of Technology, Tehran, Iran

ABSTRACT

Cellular learning automata (CLA) is known as a model for systems that have been made from very simple components. In this model, the behavior of each component is determined based on the past experiences and the behavior of its neighbors. Simple components interacting through this model can show complex and powerful behaviors in various applications. Image processing, is one of the important areas in which cellular learning automata can be used at different stages of processing. Cellular specificity of cellular learning automata can be used in image processing for propagation of neighborhood information. Also, its cellular properties can be used to increase the accuracy of image processing algorithms. In this paper, we study the properties of cellular learning automata and its applications in various fields of study and its application in the field of edge detection and image segmentation introducing two new algorithms are studied. In this paper, an adaptive method is proposed to detect image edges. This method, unlike the conventional image processing techniques, detects edges of the image using statistical features of the input image in cooperation with cellular learning automata. In the proposed segmentation method, cellular learning automata propagate the texture information of input image until converging to a segmented image. Simulations show enhancements in results on both edge detection and image segmentation using cellular learning automata.

Keywords: Learning Automata, Cellular Learning Automata, Image Processing, Edge Detection, Segmentation.

SWIG Workflow Engine Generator

Morteza Yousef Sanati¹

Seyed Hassan Mirian HosseinAbadi²

¹Engineering Faculty, Bu-Ali Sina University, Hamedan, Iran

²Computer Engineering Department, Sharif University of Technology, Tehran, Iran

ABSTRACT

Different work flows are used in most organizations for performing their responsibilities. Software developers use Workflow Engines to computerized basic tasks and policies of existing work flows in the software applications. On the other hand, developing a general purpose Workflow Engine which can manage all organizations' needs is a costly and time consuming activity. Product Line Engineering, e.g. Family Based Approach, is a discipline for developing high quality software in a reasonable amount of time and cost. In this paper, we investigate the Family Based approach in the Workflow Engine domain and present an Application Modeling Language for modeling the members of this domain. Using such a language reduces the difficulties of developing Workflow engines.

Keywords: Family Based Approach, AML, Workflow Engine.

Surveying Some Roadmaps for Adoption Service Oriented Architecture

Tayyebbeh Azadi¹

Ali Moeini²

Nasser Modiri¹

¹Faculty of Engineering, Islamic Azad University-Tehran North Branch, Tehran, Iran

²Faculty of Engineering, University of Tehran, Tehran, Iran

ABSTRACT

There are many different methods and techniques about adoption Service Oriented Architecture. This paper explains roadmaps include Iowa SOA, Beacon, IBM and CBDI. Then these roadmaps evaluate with SWOT method in BSC framework. At the end we result strength and weakness them.

Keywords: SOA, Roadmap, Governance, SWOT, Success Factors.

Towards Persian Text Mining: A Categorisation Approach

Mohammad Hossein Saraee Azar Shahgholian

Electrical and Engineering Department, Isfahan University of Technology, Isfahan, Iran

ABSTRACT

With the rapid growth of online information, there is a growing need for tools that help in finding, filtering and managing the high dimensional data. Automated text categorization is a supervised learning task, defined as assigning category labels to new documents based on likelihood suggested by a training set of labelled documents. Real-world applications of text categorization often require a system to deal with tens of thousands of categories defined over a large taxonomy. Since building these text classifiers by hand is time consuming and costly, automated text categorization has gained importance over the years. Although, text categorisation techniques have been implemented in this in a number of natural languages, none has been reported for Persian language. In this paper, machine learning method is utilised to categorise Persian text. A Persian text categorization tool is designed and implemented based on this approach. This tool, examines a set of learning texts in the learning phase to extract the features for each group and finally to obtain the primary features of groups. Stemming is used to decrease the dimension of feature vectors. Accuracy of the proposed method is tested on a number of Persian texts in different groups, including political, economic, social, religious, cultural, sports and medical texts. The results show a significant accuracy for categorising economic, political and sport group and acceptable results for the rest of the text groups.

Keywords: Machine Learning, Persian Text Mining, Persian Text Categorisation, KNN Algorithm.

Improving Tora Routing Protocol in MANETs using ANT Colony Algorithm

Rana Ghazali

Mojtaba Safari

Ali Movaghar Rahimabadi

Department of Computer Engineering, Sharif University of Technology, Tehran, Iran

ABSTRACT

A novel routing scheme for mobile ad hoc networks (MANETs), which combines the on-demand routing capability of Ad Hoc TORA routing protocol with a distributed topology discovery mechanism using ant-like mobile agents is proposed in this paper. The proposed hybrid protocol reduces route discovery latency and the end-to-end delay by providing high connectivity without requiring much of the scarce network capacity. On the one side the proactive routing protocols in MANETs like Destination Sequenced Distance Vector (DSDV) require to know, the topology of the entire network. Hence they are not suitable for highly dynamic networks such as MANETs, since the topology update information needs to be propagated frequently throughout the network. These frequent broadcasts limit the available network capacity for actual data communication.

Through extensive simulations in this paper it is proved that the proposed Ant-TORA hybrid routing technique, is able to achieve reduced end-to-end delay compared to conventional TORA routing protocols.

Keywords: Routing Protocol, Mobile Ad Hoc Networks, Ant Colony.

The CSI Journal on Computer Science and Engineering

Vol. 8

No. 1 & 3 (a)

2010

ABSTRACTS

- **Improving Tora Routing Protocol in MANETs using ANT Colony Algorithm** 1
Rana Ghazali, Mojtaba Safari and Ali Movaghar Rahimabadi

- **Towards Persian Text Mining: A Categorisation Approach** 2
Mohammad Hossein Saraee and Azar Shahgholian

- **Surveying Some Roadmaps for Adoption Service Oriented Architecture** 3
Tayyebeh Azadi, Ali Moeini and Nasser Modiri

- **SWIG Workflow Engine Generator** 4
Morteza Yousef Sanati and Seyed Hassan Mirian HosseinAbadi

- **Cellular Learning Automata in Image Processing Applications** 5
Ahmad Ali Abin, Seyed Hamid Amiri and Hamid Beigy

- **Protection of the AES Algorithm Against Differential Power Analysis by Using Randomization in Composite Fields** 6
Masoud Masoumi and Seyed Mojtaba Dehnavi

- **Introducing Speech Enhancement Filters in Wavelet Domain** 7
Hoda Kaviani Shahrezaei, Shima Tabibian and Ahmad Akbari

The CSI Journal on Computer Science and Engineering

A quarterly publication of Computer Society of Iran (CSI)

Editor-in-Chief

H. Sarbazi-Azad, Associate Professor, Sharif University of Technology, and IPM, Tehran, Iran.

Editorial Board

G. Agha, Professor, University of Illinois at Urbana-Champaign, USA.
H. Arabnia, Professor, University of Georgia, USA.
F. Arbab, Professor, CWI and Leiden University, The Netherlands.
K. Badie, Associate Professor, Iran Telecommunication Research Center, Iran.
N. Bagherzadeh, Professor, University of California at Irvine, USA.
B. Bose, Professor, Oregon State University, USA.
A. Edalat, Professor, Imperial College, UK.
M. Fathi, Associate Professor, Iran University of Science and Tech., Iran.
M. H. Ghassemian, Professor, Tarbiat Modarres University, Iran.
M. Ghodsi, Professor, Sharif University of Technology, Iran.
A. R. Hurson, Professor, Pennsylvania State University, USA.
F. Jahanian, Professor, University of Michigan, USA.
E. Kabir, Associate Professor, Tarbiat Modarres University, Iran.
F. C. M. Lau, Professor, University of Hong Kong, Hong Kong.
A. Movaghar, Professor, Sharif University of Technology, Iran.
N. Mahdavi-Amiri, Professor, Sharif University of Technology, Iran.
R. Meybodi, Professor, Amirkabir University of Technology, Iran.
K. Nakano, Professor, Hiroshima University, Japan.
M. Ould-Khaoua, Professor, University of Glasgow, UK.
B. Parhami, Professor, University of California at Santa Barbara, USA.
R. Safabakhsh, Professor, Amirkabir University of Technology, Iran.
H. Sarbazi-Azad, Associate Professor, Sharif University of Technology, and IPM, Iran.
B. Shirazi, Professor, Washington State University, USA.
A. Zomaya, Professor, The University of Sydney, Australia

International Advisory Board

A. Almagarmid, Purdue University, USA.
M. Anvari, University of California at Berkeley, USA.
J. Armstrong, Northeastern University, USA.
H. Berenji, NASA, USA.
D. Bjorner, Delft University, The Netherlands.
M. Ghanbari, University of Essex, UK.
M. Pedram, University of Southern California, USA.
P. Rogaway, University of California at Davis, USA.
M. Sarrafadeh, Northwestern University, USA.
B. Zoghi, Texas A&M University, USA.

Assistants

M. Arjomand (Editorial Assistant)
S. Shahnazi (Publication Assistant)
L. Nourani (Publication Assistant)
A. Tavakkol (Webmaster)

Disclaimer: Publication of papers in CSI-JCSE does not imply that the editorial board, reviewers, or CSI-JCSE accept, approve or endorse the data and conclusions of authors.