علوم و مهندسی کامپیوتر
نشریه علمی پژوهشی انجمن کامپیوتر ایران
مجلد ۴ شماره ۱ (الف)، ۱۳۸۵
صفحات ۲۶-۳۲
مقاله کوتاه

نشریه علمی پژوهشی انجمن کامپیوتر ایران

طراحی تکاملی شمای پایگاه دادهٔ رابطه‌ای بر پایه اندازه‌گیری نرم‌افزار

امین روزبهانی سید حسن میریان

دانشکده مهندسی کامپیوتر دانشگاه صنعتی شریف، تهران، ایران

چکیده

طراحی شمای پایگاه دادهٔ رابطه‌ای از مراحلی است که نقش طراح در آن بسیار مؤثر است. ابزارهای موجود در این زمینه طراح را به سمتی راهنمایی می‌کنند که پایگاه داده‌ای بدون آنومالی و درست از نظر نحو ایجاد کند. در همین موارد نیز ابزارها اغلب ناقص هستند. اما این که یک پایگاه داده در نهایت از جهت کیفیت پاسخگوی نیازها خواهد بود یا نه مسأله‌ای است که به آن کمتر توجه شده است و به همین دلیل امکان هدایت طراح برای ایجاد محصولی بهتر به صورت سیستماتیک وجود ندارد. در این مقاله با استفاده از روشی مبتنی بر مبانی اندازه‌گیری نرم افزار معیارهایی برای اندازه‌گیری کیفیت شمای پایگاه دادهٔ رابطه‌ای به دست می‌آید. سپس بر اساس مدلی که می تواند کیفیت شمای پایگاه داده را با استفاده از معیارها تخمین زند، روشی برای طراحی شمای پایگاه داده به صورت تکاملی ارائه می‌شود. در این گونه طراحی سعی می‌شود کمترین تعداد حالات برای رسیدن به پاسخ مناسب بررسی شود.

واژه‌های کلیدی: طراحی پایگاه دادهٔ رابطه‌ای، ارزیابی کیفیت، اندازه‌گیری نرم افزار، الگوریتم ژنتیک.

۱- مقدمه

طراحی پایگاه داده مراحل مختلفی دارد که ابتدا از مدل سازی معنایی شروع می‌شود. مهم ترین هدف این مرحله شناخت صفت‌ها و وابستگی‌های بین آن‌ها است. پس از آن طراحی منطقی انجام می شود. دو روش برای طراحی منطقی وجود دارد: روش بالا به پایین و روش سنتز رابطه‌ای. در روش بالا به پایین، پس از مدل سازی معنایی به مجموعه‌ای از رابطه‌های خوش طرح می‌رسیم. در روش سنتز رابطه‌ای، مجموعه صفت‌های موجود در نظر گرفته می شود، سپس با توجه به مجموعه قواعد جامعیتی و محدودیت‌های روی صفات و به وسیلهٔ روش‌های نرمال سازی، صفات به مجموعه‌های جداگانه تبدیل می شوندکه تشکیل شماهای رابطه می دهند. در این روش ممکن است یک رابطهٔ جهانی در نظر گرفته شود و با سنتز این رابطه، رابطه‌های کوچک‌تری به وجود آید.

در روش ترکیبی ابتدا با روش بالا به پایین شمای پایگاه داده به دست می‌آید سپس با دخالت دادن برخی محدودیت‌های جامعیتی که در مدل سازی معنایی داده‌ها منظور نشده‌اند، رابطه‌ها نرمال می‌شوند. بعد از طراحی منطقی، طراحی فیزیکی خواهد بود. در طراحی فیزیکی دیگر به مسائل معنایی پرداخته نمی‌شود. طراحی فیزیکی به عواملی مانند حجم و ماهیت داده‌ها، تعداد کاربران، تعداد و ماهیت تراکنش‌ها بستگی دارد. همچنین نوع سامانه مدیریت پایگاه داده (DBMS)و امکانات آن در این مرحله بسیار تأثیرگذار است.

در این مراحل طراح همواره باید سعی کند طراحی انجام شده ارضا کنندهٔ نیازمندی‌های مورد نظر باشد. بنابراین گاهی لازم است برخلاف مفاهیم نرمال-سازی، رابطه‌ها را غیرنرمال کرد. در بسیاری موارد نیز طراحی‌های مختلفی را می-توان برای یک مسأله ارائه کرد و تصمیم‌گیری در مورد این که کدام مناسب‌تر است امری دشوار است. ابزارها و روش‌هایی وجود دارد که در مرحله طراحی کمک کننده‌اند. این روش‌ها و ابزارها (به طور نمونه نرمال‌سازی) محصول نهایی را کمتر مد نظر دارند و معیارهایی برای طراحی خوب دارند که آیندهٔ پایگاه داده در آن‌ها مطرح نیست. این مشکل به این دلیل است که معیارهای مناسب و قابل اندازه‌گیری برای پایگاه دادهٔ رابطه‌ای وجود ندارد و روش مناسب و قابل قبول برای ارزیابی کیفیت پایگاه داده ارائه نشده است.

مسأله این است که با داشتن مجموعه‌ای از صفت‌ها و وابستگی تابعی بین آنها، شمای پایگاه داده‌ای به دست آوریم که با توجه به شاخص‌ها کیفیت خوبی داشته باشد. با داشتن مدلی برای ارزیابی کیفیت یک شمای پایگاه داده، می‌توان طراحی

خوب را مشخص کرد. بنابراین می‌توان روشی برای طراحی پایگاه داده به وجود آورد که مبتنی بر این نوع ارزیابی کیفیت باشد. ساده‌ترین روش، جستجوی تمام حالت‌های ممکن از شمای پایگاه داده با داشتن صفات و وابستگی‌های بین آن‌ها است[1]. اما معمولا این حالت‌ها بسیار زیاد خواهند بود. از آن جا که هر پایگاه داده افرازی از مجموعهٔ صفات است، این مساله شبیه به مسائل افراز مجموعه[2] است[1]. تعداد افرازهای مجموعه‌ای با n عضو، n-امین عدد بل[3] است که پیچیدگی زمانی آن نمایی است و مثلا برای n=۳۰ تعداد افرازها در حدود $۸/۴\times۱۰^{۲۳}$ است[2]. بنابراین باید از روشی دیگر استفاده کرد که طراحی قابل قبولی را در زمانی مناسب ارائه کند. الگوریتم‌های تکاملی[4] در این زمینه قابل استفاده هستند[3].

در ادامهٔ مقاله در بخش ۲ روشی برای ارزیابی کیفیت شمای پایگاه داده مطرح می‌شود که به کمک آن بتوان کیفیت شمای پایگاه داده را تخمین زد. سپس با طراحی یک الگوریتم ژنتیک در بخش ۳ و استفاده از تخمین کیفیت افراد[5] هر نسل، می‌تواند در هر مرحله تولید نسل، نسل‌هایی با کیفیت بهتر ایجاد کند.

۲- ارزیابی کیفیت شمای پایگاه داده

برای ارزیابی کیفیت پایگاه داده و اینکه چه چیز یک مدل خوب می‌سازد هنوز رهنمودی که به طور عمومی پذیرفته شده باشد وجود ندارد[4]. اغلب کارهای انجام شده تنها لیستی ساده از خاصیت‌هایی ارائه می‌دهند که مدل باید داشته باشد بدون این که ساختاری سیستماتیک برای ارزیابی آن‌ها ارائه دهند. به علاوه، این لیست‌ها نیز اغلب غیر ساخت‌یافته و بدون تعریف دقیق هستند. خواص کیفیتی اغلب برای تضمین کیفیت کافی نیستند چون افراد مختلف درک متفاوتی از یک مفهوم دارند. بنابراین باید اندازه‌گیری‌های کمی و کیفی برای کم کردن پیش‌داوری‌ها در فرآیند ارزیابی وجود داشته باشد.

بسترهایی ارائه شده‌اند که به موضوع کیفیت در مدل‌سازی داده و پایگاه داده رابطه‌ای به صورت سیستماتیک پرداخته‌اند[4,6].

مهندسین نرم‌افزار معیارهای زیادی برای محصولات، فرآیندها و منابع نرم‌افزاری به وجود آورده‌اند. اما متأسفانه تقریبا اغلب مطالعات در جهت ایجاد معیارهایی برای اندازه‌گیری کیفیت برنامه‌ها بوده است و در مورد مدل‌های داده و پایگاه دادهٔ رابطه‌ای کمتر تحقیقاتی وجود داشته است. این موضوع نشان می‌دهد که در مقایسه با کیفیت مهندسی نرم افزار، مفهوم کیفیت پایگاه داده کمتر درک شده است. پیشنهادهایی برای اندازه‌گیری مدل‌های مفهومی توسط گری (Gray) و دیگران[7]، مودی (Moody)[8]، جنرو (Genero) و دیگران[9] و پیاتینی (Piattini) [10] ارائه شده است.

معیارهای ارائه شده در کارهای قبل بیشتر برای مدل‌سازی به روش موجودیت-ارتباط و نمودار موجودیت-ارتباط تعریف شده‌اند. گرچه روش موجودیت-ارتباط مهم‌ترین شیوه در مدل‌سازی داده است اما نمی‌توان بدون در نظر گرفتن تئوری پایگاه‌های داده‌ای رابطه‌ای و مدل رابطه‌ای این معیارها را مناسب برای ارزیابی کیفیت این گونه پایگاه‌ها دانست. با توجه به این نکته و این که در تحقیقات، پایگاه داده رابطه‌ای کمتر مد نظر قرار گرفته است، ابتدا به ملاحظات تئوریک معیارهای مورد نیاز پرداخته خواهد شد، سپس معیارها و شاخص‌ها با بهره‌گیری از روش تعریف معیار در کارهای دیگران معرفی می‌گردند. پس از آن با طراحی و انجام آزمایشات، به اعتبارسنجی معیارها و ایجاد مدلی بر اساس شبکه‌های عصبی برای تخمین شاخص‌های کیفیتی با استفاده از معیارهای قابل اندازه‌گیری پرداخته می‌شود. از این مدل در ساخت تابع شایستگی[6] برای طراحی تکاملی استفاده خواهد شد.

۲-۱ بررسی تئوریک معیارهای مورد نیاز

بر اساس تئوری پایگاه دادهٔ رابطه‌ای، هنگام انجام اعمال در پایگاه داده، مشکلاتی ممکن است رخ دهد که به آنها آنومالی می‌گویند. این آنومالی‌ها به دلیل وجود وابستگی‌های مختلف در شماهای رابطه و پایگاه داده است. وابستگی‌های تابعی[7] و شمول[8] از مهم‌ترین نوع وابستگی‌های داخل رابطه‌ای و بین رابطه‌ای است.

برای به دست آوردن پایگاه داده‌ای که کمترین آنومالی‌ها را داشته باشد روش‌های نرمال‌سازی مطرح شده‌اند. در طراحی پایگاه داده، نرمال‌سازی رابطه‌ها عملی است که بر اساس وابستگی‌های موجود در یک رابطه یا بین رابطه‌ها انجام می‌شود. نرمال‌سازی دارای معایب و مزایایی است که ذکر آن‌ها از حوصلهٔ این مقاله خارج است. اما به طور معمول نرمال سازی به عنوان روشی برای به دست آوردن مدل فیزیکی مورد استفاده قرار می‌گیرد و باعث می‌شود آنومالی‌ها در به روزرسانی و حذف رخ ندهند. اما در برخی موارد نیز سبب می‌شود بازیابی اطلاعات با کندی انجام پذیرد.

شکل نرمال BCNF[9] با توجه به وابستگی تابعی تعریف می‌شود و مربوط به یک رابطه بدون در نظر گرفتن سایر رابطه‌هاست. شکل نرمال شمول[10] (INNF) برای تضمین این که پایگاه داده با توجه به وابستگی‌های شمول و وابستگی‌های تابعی افزونگی نداشته باشد ارائه شده است. شکل‌های قبلی نرمال تنها به رابطه‌ها به طور جداگانه توجه دارند که در آن‌ها صفات با نام‌های یکسان در رابطه‌های مختلف متمایز از یکدیگرند. شکل نرمال شمول چندین رابطه را در بر دارد. بنابراین بر اساس فرض رابطهٔ جهانی[11] اگر صفتی در دو رابطه یا بیشتر در پایگاه داده‌ای ظاهر شود، در تمام این موارد به یک مفهوم اشاره دارد و معنای یکسانی را می‌رساند. برای مطالعه بیشتر در مورد شکل‌های نرمال و وابستگی‌های مختلف، خواننده می‌تواند به مراجع معتبر در این زمینه مراجعه کند.

برای شکل نرمال INNF نیز مراجعی مانند [11]، [12]، و [13] مفید هستند. ایدهٔ شمای ناحلقه‌ای[12] در دههٔ ۸۰ میلادی با نام‌های مختلفی مانند شمای بدون دور، شمای درختی و شمای بدون دور α معرفی شد. شماها و پایگاه داده‌های ناحلقه‌ای خواصی دارند که باعث می‌شود پرس وجو در رابطه‌های مبتنی بر آن‌ها را بتوان با کارایی بهتری انجام داد.

اگر در انجام یک پرس وجو، بخشی از شمای پایگاه داده که در پرس وجو شرکت می‌کند γ-ناحلقه‌ای باشد، پرس وجو را می‌توان با یک عبارت پیوند یکنواخت[13] انجام داد[14]. اهمیت یکنواخت بودن در این است که وقتی دو رابطه سازگار هستند، هنگام پیوند آن‌ها هیچ تاپلی گم نمی‌شود. دو رابطه سازگار هستند اگر پرتویی از یک رابطهٔ جهانی باشند. در محاسبهٔ حاصل یک عبارت پیوند یکنواخت روی رابطه‌های سازگار هیچ تاپلی از بین نمی‌رود. بنابراین تعداد تاپل‌ها به صورت یکنواخت افزایش پیدا می‌کند. با استفاده از عبارت‌های پیوند ترتیبی[14]، تنها یک حاصل پیوند میانی باید نگهداری شود. هر عبارت پیوند یکنواخت را می‌توان به صورت یک عبارت پیوند ترتیبی نوشت. عبارت‌های پیوند یکنواخت، سبب می شوند عمل پیوند با کارایی بالا از نظر زمان و فضای مورد نیاز انجام شود[15].

۲-۲ معیارهای پایگاه دادهٔ رابطه‌ای

گام نخست ارائه معیارهایی است که اهداف واضحی را مد نظر دارند. یعنی معیارها مبتنی بر هدف باشند. شیوهٔ هدف/پرسش/معیار یا GQM [16] بستری برای به دست آوردن معیارهای درست از اهداف اندازه‌گیری است.

هدف ما «اندازه‌گیری کیفیت در پایگاه دادهٔ رابطه‌ای» است. پرسش‌هایی که می‌توان برای این هدف مطرح کرد پرسش‌های زیر هستند:

- آیا به هنگام سازی اطلاعات در پایگاه داده با هزینه و زمان اندک انجام می‌شود؟
- آیا استخراج اطلاعات از پایگاه داده با هزینه و زمان اندک انجام می شود؟

• آیا پایگاه داده از فضای ذخیره‌سازی به طور مناسب استفاده می‌کند؟

این‌ها سوالاتی هستند که در سه زمینهٔ مختلف به طور کلی مطرح شده‌اند: زمان به هنگام سازی، زمان پرس وجو در پایگاه داده، و فضای ذخیره سازی. برای پاسخ به هر یک از این سوالات ش٠اخص‌های کیفیتی تعریف می‌شوند. این شاخص‌ها معیارهایی هستند که به طور مستقیم بیان کننده میزان کیفیت پایگاه داده از سه دیدگاه متفاوت‌اند.

۲-۲-۱ شاخص‌های کیفیتی

در نظر گرفتن تعداد زیاد شاخص‌های کیفیتی و معیارها برای اندازه‌گیری آن‌ها باعث می‌شود نتایج نادرستی به دست آید. بنابراین شاخص‌ها و معیارهای مهم در نظر گرفته شده‌اند. ابتدا سه شاخص کیفیتی در پایگاه دادهٔ رابطه‌ای تعریف می‌شود که کیفیت شمای پایگاه داده را از سه جنبهٔ مهم به هنگام سازی، پرس وجو و فضای پایگاه داده در بر می‌گیرند.

هزینه واحد پرس وجو

معمولا سعی می‌شود انجام پرس وجوها در پایگاه داده با حداقل هزینه از نظر زمان، حافظه، زمان استفاده از CPU و دستگاه‌های ورودی/خروجی انجام شود. البته زمان بیشترین اهمیت را دارد. بنابراین اندازه‌گیری *هزینه واحد پرس وجو*[۱۵] برای هر پرس وجوی Q، به این صورت تعریف می‌شود:

$$\theta^q(Q) = \frac{Cost\ of\ Q}{Number\ of\ tuples\ in\ query\ result}$$

θ^q در واقع هزینهٔ به دست آوردن هر فقره اطلاع است. اگر تاپلی در نتیجه وجود نداشته باشد آن را ۱ در نظر می‌گیریم چون پرس وجوی بدون پاسخ خود یک اطلاع است. هر چه θ^q برای یک پرس وجو بیشتر باشد، نشان دهنده این است که زمان بیشتری برای به دست آوردن یک فقره اطلاع مورد نیاز است. بنابراین پرس وجویی بهتر است که θ^q برای آن کمتر باشد. می‌توان شاخص کیفیتی برای پرس وجوها در یک پایگاه داده را با استفاده از شاخص متناظر برای رابطه‌ها تعریف کرد. بنابراین *هزینه پرس وجوی پایگاه داده*[۱۶] به صورت تابعی از هزینه‌های واحد پرس وجو برای تمام پرس وجوهای معقول در پایگاه دادهٔ d تعریف می‌شود:

$$\theta^{dq}(d) = F(\theta_1^q, \theta_2^q, \Lambda\ , \theta_n^q)$$

که در آن n تعداد کل پرس وجوهای معقول در پایگاه داده است. F می‌تواند مجموع وزنی، بیشینه، میانگین و مانند آن باشد. پرس وجویی معقول گفته می‌شود که روی مجموعه‌ای از رابطه‌ها باشد که دو به دو حداقل یک صفت مشترک داشته باشند.

هزینه واحد به هنگام سازی

همانند پرس وجوها، به هنگام سازی‌ها نیز هزینه دارند. این هزینه‌ها شامل هزینه‌های مختلفی مانند هزینه انجام عمل به هنگام سازی و حفظ سازگاری پایگاه داده است. شاخص *هزینه واحد به هنگام سازی*[۱۷] برای هر رابطه به این صورت تعریف می‌شود:

$$\theta^u(r) = \frac{Cost\ of\ possible\ updates\ in\ r}{Number\ of\ changed\ tuples}$$

در این جا نیز اگر نتوان تاپلی را به هنگام کرد مقدار یک را برای مخرج کسر در نظر می‌گیریم. چون به این معنی است که به هنگام سازی باعث ناسازگاری می‌شود یا این که اطلاعی برای به هنگام سازی وجود ندارد. *هزینه به هنگام سازی پایگاه داده*[۱۸] این گونه تعریف می‌شود:

$$\theta^{du}(d) = F(\theta_1^u, \theta_2^u, \Lambda\ , \theta_n^u)$$

در این تعریف نیز n تعداد رابطه‌ها در پایگاه داده است و F می‌تواند مجموع وزنی، بیشینه، میانگین و مانند آن باشد.

سودمندی فضای پایگاه داده

اگر Space را تابعی از مجموعهٔ صفت‌ها به اعداد طبیعی در نظر بگیریم که بیانگر میزان فضای اشغال شده توسط یک صفت باشد، *فضای رابطه*[۱۹] را برای هر رابطهٔ r(A) به این صورت تعریف می‌کنیم:

$$S^R(r) = |r| \times \sum_{A \in X} Space(A)$$

این مقدار نشان می‌دهد که یک رابطه چه مقدار فضا مصرف کرده است. *فضای صفت*[۲۰] برای هر صفت A در یک پایگاه دادهٔ d که مجموعه رابطه‌های $\{R_1(X_1),\ldots, R_n(X_n)\}$ دارد را به صورت زیر تعریف می‌کنیم:

$$S^A(A) = \left|\Upsilon_{i=1}^n \pi_A(r_i)\right| \times Space(A)$$

این مقدار حداقل فضایی است که یک صفت در پایگاه داده دارد یعنی فضای ایده‌آل برای نگهداری اطلاعات صفت A، $S^A(A)$ است. حال معیاری تعریف می‌کنیم که میزان استفاده درست از فضا در پایگاه داده را مشخص می‌کند. *سودمندی فضای پایگاه داده*[۲۱] را برای پایگاه دادهٔ d به صورت زیر تعریف می‌کنیم:

$$\theta^s(d) = \frac{\sum_{A \in \Upsilon_{i=1}^n X_i} S^A(A)}{\sum_{i=1}^n S^R(r_i)}$$

هرچه سودمندی فضای پایگاه داده بیشتر باشد نشان دهنده این است که رابطه‌ها هر یک جداگانه و نیز با در نظر گرفتن وابستگی‌های بین آن‌ها افزونگی کمتری دارند و از فضا برای نگهداری داده‌ها بهتر استفاده شده است.

۲-۲-۲ معیارهای مورد نیاز

در ادامه معیارهایی تعریف می‌شود که عوامل تعیین کننده برای شاخص‌های مطرح شده باشند. این معیارها عینی هستند و می‌توان آن‌ها را برای یک پایگاه داده، یک رابطه، و یک پرس وجو محاسبه کرد. در جدول ۱، معیارهای هر شاخص کیفیتی نشان داده شده است.

سه شاخص تعریف شده اندازه‌هایی هستند که میزان کیفیت یک پایگاه داده پیاده سازی شده را از منظری خاص نشان می‌دهند. معیارها نیز اندازه‌هایی هستند که از خصوصیات پایگاه داده طراحی شده به دست می‌آیند و مقدار آن‌ها در اندازه شاخص‌ها اثرگذارند. شاخص‌ها به دید کیفیتی نزدیک‌ترند و برای استفاده کننده از پایگاه داده قابل لمس هستند. از طرف دیگر معیارها به خصوصیت‌های ساختاری پایگاه داده وابسته‌اند. در جدول ۱، مقدار معیارهای هر شاخص بر مقدار آن شاخص تاثیر دارد. این تاثیرگذاری را با انجام آزمایشاتی در بخش‌های بعدی مورد آزمایش قرار می‌دهیم. بنابراین رابطه بین مشخصات ساختاری پایگاه داده و خصوصیت‌های کیفیتی را مورد بررسی قرار خواهیم داد تا از تاثیرگذاری معیارها بر شاخص مربوط اطمینان حاصل کنیم. اگر معیار بر شاخص مربوط تاثیر نداشته باشد، آن معیار معتبر نخواهد بود و در طراحی تکاملی مورد استفاده قرار نخواهد گرفت.

اعتبارسنجی و مدل تخمین

همانند سایر زمینه‌ها در مهندسی نرم افزار، ارائهٔ فنون و معیارها کافی نیست. نیاز به اعتبارسنجی برای اطمینان از سودمندی آن‌ها در عمل نیز هست. راه‌های گوناگونی برای انجام این گام وجود دارد و می‌توان آن‌ها را به دو دستهٔ مهم، آزمایشات[۲۲] و موارد مطالعه[۲۳] تقسیم کرد.

جدول ۱. معیارها برای هر یک از شاخص‌ها

معیارهای θ^s (دریک پایگاه داده)	معیارهای θ^q (در یک پرس و جوی معقول)	معیارهای θ^u (در یک رابطه)
تعداد صفت‌های مشترک (DBN_CATTRS) تعداد صفت‌ها (DBN_ATTRS) تعداد رابطه‌ها (DBN_RELS) تعداد نقض‌ها از INNF و BCNF (DBV_INNF) تعداد وابستگی‌های شمول با یک صفت (DBN_IND1) تعداد وابستگی‌های شمول با دو صفت (DBN_IND2)	تعداد نقض‌ها از γ-ناحلقه‌ای بودن (QV_ACYCLIC) تعداد صفت‌های مشترک (QN_CATTRS) تعداد صفت‌ها (QN_ATTRS) تعداد رابطه‌ها (QN_RELS)	تعداد نقض‌ها از INNF و BCNF (RV_INNF) تعداد وابستگی‌های شمول با یک صفت (RN_IND1) تعداد وابستگی‌های شمول با دو صفت (RN_IND2)

در انجام آزمایشات به پارامترهایی نیاز است که خارج از محدوده شمای پایگاه دادهٔ رابطه‌ای است. برای سادگی فرض می‌شود دامنهٔ تمامی صفت‌ها در پایگاه داده، زیرمجموعه‌ای یکسان از اعداد طبیعی است. این پارامترهای محیطی در لیست زیر تعریف می‌شوند:

۱. تعداد تاپل‌ها در یک رابطه (TUPLE_NUMBER)
۲. احتمال هیچ مقدار بودن صفت (NULL_PROB)، نسبت هیچ مقدار به کل مقادیر برای صفت یا به عبارت دیگر احتمال هیچ مقدار بودن صفت، تعداد هیچ مقدار نسبت به تعداد مقادیر یک صفت در یک رابطه است. این پارامتر را برای نزدیک شدن به حالت واقعی در نظر گرفته‌ایم.
۳. تعداد عناصر در مجموعه دامنهٔ صفات (DOMAIN_CARD)، یعنی مقادیر ممکن برای صفت A. مقادیر یک صفت در یک رابطه از مجموعهٔ {1,2,…,DOMAIN_CARD} انتخاب می‌شود. تأثیر این متغیر در میزان همپوشانی مقادیر یک صفت در دو رابطهٔ مجزاست.
۴. نسبت کاهش تاپل‌ها در وابستگی تابعی (TUPLE_DEC_RATIO)، نسبت کاردینالیتی دامنهٔ صفت B به صفت A وقتی B به طور تابعی به A وابسته باشد (A→B). این متغیر را به این دلیل تعریف می‌کنیم که در بسیاری مواقع صفاتی که به آن‌ها وابستگی تابعی وجود دارد تعداد مقادیر کمتری دارند.

برای ساخت مدل تخمین از شبکه‌های عصبی مصنوعی[۲۴] و برای اعتبارسنجی معیارها از تحلیل واریانس[۲۵] استفاده شده است. با استفاده از این روش میزان وابستگی یک معیار به شاخص مربوطه با وجود سایر معیارها آزمون می‌شود. آزمایشات با استفاده از برنامه‌ای به زبان جاوا و سامانه مدیریت پایگاه دادهٔ SQL Server در رایانه‌ای شخصی با پردازندهٔ Petnuim IV 2.4 MHz و با حافظهٔ 128 MB انجام شده است. برای شاخص‌های کیفیتی θ^u، θ^q و θ^{du} به ترتیب ۷۷۷، ۵۹ و ۵۳۵ فقره دادهٔ آزمایشی به دست آمده است. با انجام سه بار تحلیل واریانس، تمام معیارها به غیر از تعداد صفت‌ها برای θ^q (معیار QN_ATTRS) معتبر شناخته شدند.

از سه شبکه انتشار به عقب ۲۶ با دو لایه استفاده شده است که تابع فعال سازی۲۷ آن‌ها یک تابع سیگموید۲۸ برای لایه اول و تابع خطی برای لایه دوم است. تابع خطا میانگین مربعی[۲۹] است. تعداد نورون‌ها در لایه پنهان با سعی و خطا به دست آمده است. الگوریتم آموزش این شبکه‌ها انتشار به عقب با استفاده از روش اعتبارسنجی تقاطعی[۳۰] است.

شبکه‌های عصبی به دست آمده مدلی تخمین زننده تشکیل می‌دهند. ورودی این مدل اندازه‌هایی است که برای معیارها از مدل رابطه‌ای استخراج می‌شود و خروجی آن شاخص‌های کیفیتی است. با استفاده از این مدل می‌توان شاخص‌های یک شمای پایگاه داده را تخمین زد. از آن جا که معیارها عینی هستند، محاسبه و اندازه‌گیری معیارهای تعریف شده برای شمای پایگاه داده قابل انجام است. این اندازه‌گیری‌ها ورودی مدل تخمین خواهند بود.

۳- طراحی تکاملی

محاسبات تکاملی[۳۱] به الگوریتم‌های مستقل از دامنه برای جستجوی اتفاقی و روش‌های کمینه کردن بر اساس تکامل در طبیعت گفته می‌شود. این روش‌ها برای حل مسائلی که از نظر محاسباتی پیچیده هستند مورد استفاده قرار می‌گیرند. مفهوم طراحی تکاملی به این معنی است که مجموعه‌ای از شماهای پایگاه داده وجود دارند که یک نسل یا جمعیت را تشکیل می‌دهند. هر شما در این مجموعه یک فرد از این جمعیت است. تعداد افراد در یک جمعیت را یک عدد ثابت در نظر می‌گیریم. به هر فرد بر اساس معیارهایی، مقداری به عنوان *مقدار شایستگی*[۳۲] نسبت داده می‌شود. یکی از شاخه‌های اصلی محاسبات تکاملی *الگوریتم‌های ژنتیک*[۳۳] است. در این روش هر فرد از مقادیر ژنی تشکیل شده است که خواص آن‌ها با استفاده از عملگرهای *جهش*[۳۴] و *ترکیب*[۳۵] به نسل بعدی منتقل می‌شود[3].

هر شمای پایگاه داده مجموعه‌ای از شماهای رابطه است. صفات از ۱ تا n شماره‌گذاری می‌شوند. هر شمای رابطه با آرایه‌ای از مقادیر بولی که تعداد آنها با تعداد صفت‌های داده شده برابر است نشان داده می‌شود. j-امین عنصر این آرایه *درست* است اگر و تنها اگر j-امین صفت در این رابطه وجود داشته باشد. با استفاده از شاخص‌های مطرح شده تابعی برای تعیین مقدار شایستگی هر شمای پایگاه داده به دست می‌آوریم.

شاخص‌ها عبارتند از θ^u، هزینه واحد به هنگام سازی، θ^q، هزینه واحد پرس وجو، θ^s، سودمندی فضای پایگاه داده، θ^{du}، هزینه واحد به هنگام سازی پایگاه داده، و θ^{dq}، هزینه واحد پرس وجوی پایگاه داده. خطای شمای رابطه برای شمای رابطهٔ R به صورت زیر تعریف می‌شود:

$$E_R = a\times\theta^u(R) + c\times(1-\theta^s(\{R\}))$$

که θ^s و θ^u برای هر شمای رابطه از شبکهٔ عصبی به دست می‌آید. در محاسبهٔ θ^s در این فرمول فرض شده است که پایگاه داده تنها شامل شمای R است. *خطای شمای پایگاه داده* را نیز برای پایگاه داده D به این صورت تعریف می‌کنیم:

$$E = a\times\theta^{du}(D) + b\times\theta^{dq}(D) + c\times(1-\theta^s(D))$$

مقادیر a، b، و c ضرایب اهمیت برای هر یک از شاخص‌ها هستند که توسط کاربر مشخص می‌شود. در این جا باید توجه کرد که تنها شاخص θ^s بسته است و دو

شاخص دیگر باز هستند. بنابراین برای مقداردهی ضرایب ممکن است لازم باشد مقدار بیشینه‌ای برای شاخص‌های باز در نظر گرفت. برای تعریف شاخص‌های θ^{dq} و θ^{du} به ترتیب بیشینهٔ θ^{q} برای پرس وجوها و بیشینهٔ θ^{u} برای شماهای رابطه در پایگاه داده در نظر گرفته می‌شود.

هر شمای پایگاه داده را با آرایه‌ای از شماهای رابطه نشان می‌دهیم. هر شمای رابطه نیز با آرایه‌ای از مقادیر بولی نشان داده می‌شود که مقدار موجود در عنصر i-ام بیانگر این است که صفت i-ام متعلق به این رابطه است یا خیر. رابطهٔ تهی وجود ندارد (رابطه‌ای که آرایهٔ متناظر آن تنها شامل مقدار *نادرست* باشد). شماهای رابطه همواره به ترتیب صعودی خطا در آرایهٔ شمای پایگاه داده قرار می‌گیرند. بنابراین شماهای رابطهٔ بهتر در ابتدای آرایه قرار دارند. افراد (شماهای پایگاه داده) در یک نسل نیز به ترتیب صعودی خطای آن‌ها نگهداری می‌شوند. مقدار شایستگی برای هر فرد مکمل خطای آن فرد است. هر چه فرد خطای کمتری داشته باشد شایسته‌تر است. آرایهٔ متناظر پایگاه داده را *کروموزوم* می‌گوییم. آرایه بولی متناظر با یک شمای رابطه را *ژن* می‌نامیم.

هر کروموزوم با شرایط بالا، با توجه به وابستگی‌های تابعی داده شده به یک شمای پایگاه داده نگاشت می‌شود. ابتدا شماهای رابطه از هر یک از ژن‌ها به وجود می‌آید. کلید اصلی هر رابطه تعیین می‌شود. اگر وابستگی تابعی به دلیل افراز از بین رفته باشد، باید با استفاده از یک ارجاع بین شمای رابطه مبدأ و مقصد و افزودن کلید مقصد به عنوان کلید خارجی به شمای رابطهٔ مبدأ، وابستگی را احیا کرد. در این جا نیاز به یک وابستگی شمول بین دو شمای رابطه است. هرگاه نیاز به محاسبهٔ مقادیر θ^{u}، θ^{s} ، θ^{dq} و θ^{du} باشد، پس از انجام تبدیل و انجام تغییرات ذکر شده، معیارهای مورد نظر برای شماها محاسبه شده و به عنوان ورودی به شبکهٔ عصبی متناظر داده می‌شود. خروجی شبکه عصبی مقدار شاخص را تعیین می‌کند. نگاشت را تنها وقتی انجام می‌دهیم که می‌خواهیم مقدار شایستگی فرد را مشخص کنیم.

در مرحله ابتدایی و برای تولید نسل اول، افرادی با شماهای رابطهٔ یک صفتی، سپس افرادی با شماهای رابطهٔ دو صفتی و احیانا یک صفتی، و به همین ترتیب ایجاد می‌شوند. ایجاد افراد اولیه تا وقتی که تعداد افراد نسل اول به تعداد معین و ثابتی برسد ادامه پیدا می‌کند. پس از آن، شایستگی افراد محاسبه می‌شود و افراد بر اساس آن مرتب می‌شوند. اگر خطای افراد در یک جمعیت حداکثر High باشد، شایستگی هر فرد با خطای E در یک جمعیت (High-E) در نظر گرفته می‌شود. برای نجات افراد شایسته هنگام تولید نسل بعدی از یک نسل موجود، درصد کمی (۱۰ درصد) از بهترین افراد به نسل بعد منتقل می‌شوند. مقدار شایستگی برای انتخاب والدین افراد جدید به کار می‌رود. احتمال هر فرد برای انتخاب شدن به عنوان یکی از والدین، نسبت شایستگی فرد به مجموع کل شایستگی‌هاست. برای تولید یک فرزند در مرحلهٔ ترکیب، دو فرد با شرایط ذکر شده انتخاب می‌شود و عمل ترکیب روی آن‌ها انجام می‌شود. مرحله تولید نسل با استفاده از عمل ترکیب تا وقتی صورت می‌گیرد که تعداد افراد نسل جدید به تعداد معین برسد. پس از آن عمل جهش روی افراد نسل انجام می‌شود.

مرحلهٔ تولید مثل را زمانی باید متوقف کرد. می‌توان مرحله تولید مثل را به تعداد مشخصی تکرار کرد. یا تا وقتی تولید نسل را تکرار کرد که فردی با شایستگی مورد نظر به دست آید.

۳-۱ مرحلهٔ ترکیب

برای تولید فرزند از دو فرد p1 و p2 با عمل ترکیب، از ابتدای شماهای رابطهٔ هر یک از والدین شروع می‌کنیم (شروع از بهترین شماهای رابطه). کم خطاترین شمای رابطه از رابطه‌های دو والد انتخاب شده، به لیست شماهای رابطهٔ فرزند اضافه می‌گردد و از والد حذف می‌شود. اگر شمای رابطه در فرزند وجود داشته باشد، بدیهی است که به آن اضافه نمی‌شود. این عمل تا وقتی ادامه پیدا می‌کند که تعداد شماهای رابطه برابر با تعداد صفات شود، یعنی تا وقتی که تمام مکان‌های خالی در کروموزوم مقدار بگیرند. حال باید مجموعه‌های دارای اشتراک را به گونه‌ای تصحیح کرد که کروموزوم بیانگر یک افراز باشد. شماهای رابطه در هر یک از p1 و p2 صفت مشترک ندارند چون هر کدام یک افراز هستند. پس اگر دو شمای رابطه اشتراک داشته باشند، صفت یا صفات اشتراکی را از شمای رابطه‌ای که خطای بیشتری دارد حذف می‌کنیم. اگر مجموعه‌ای از صفات در فرزند وجود نداشته باشد، هر صفت در این مجموعه را به شمای رابطه‌ای اضافه می‌کنیم که وجود صفت در آن باعث می‌شود مقدار خطای آن به مقدار بیشتری کم شود. اگر افزودن صفت در هیچ یک از شماها باعث کم شدن خطا نشود، آن را به هیچ شمایی اضافه نمی‌کنیم. این عمل تا وقتی ادامه پیدا می‌کند که دیگر نتوان هیچ صفتی را به شماهای موجود اضافه کرد. اگر مجموعهٔ صفات باقیمانده تهی نباشد، تمامی صفات باقیمانده به صورت یک شما به مجموعه شماهای فرزند اضافه می‌شود. در نهایت، شماهای رابطه در فرزند بر اساس خطای آن‌ها به صورت صعودی مرتب می‌شوند.

۳-۲ مرحلهٔ جهش

پس از انجام عمل ترکیب، افراد جامعه بر اساس خطای آن‌ها به صورت صعودی مرتب می‌شوند. هر جهش با احتمال مشخصی انجام می‌شود. در هر جهش، یکی از صفت‌ها به صورت تصادفی انتخاب می‌شود و از شمایی که شامل آن است حذف می‌شود و به شمایی دیگر که به صورت تصادفی انتخاب شده است افزوده می‌شود. مرحله جهش به گونه‌ای انجام می‌شود که افراد با خطای بیشتر جهش بیشتری داشته باشند. برای ۱۰ درصد از بهترین جمعیت موجود (از سمت چپ) با احتمال ۰/۱ یک بار جهش صورت می‌گیرد. برای ۳۰ درصد بعدی، با احتمال ۰/۵ دو بار جهش، برای بقیه نیز پنج بار جهش با احتمال ۰/۵ انجام می‌شود.

۴- مثالی از طراحی تکاملی

طراحی ذکر شده برای مسأله‌ای با ورودی‌های نشان داده شده در جدول ۲ اجرا شده است.

صفت‌ها را از چپ به راست شماره گذاری می‌کنیم. در این مسأله هر شمای رابطه را به صورت رشته‌ای از اعداد 0 و 1 نشان می‌دهیم که طول آن به تعداد صفت‌ها، ۱۰ است. مقدار 1 موجود در مکان i-ام نشان دهنده این است که صفت i-ام در شما وجود دارد. در ابتدای اجرای الگوریتم، شماهایی با تنها یک صفت سپس شماهایی با دو و احیانا یک صفت و ... ایجاد می‌شوند. این عمل تا وقتی ادامه پیدا می‌کند که تعداد افراد به ۱۰۰ برسد. حال باید مقدار شایستگی را برای هر فرد مشخص کنیم. برای این کار ابتدا باید نگاشت هر فرد را به شمای پایگاه داده آن بر اساس وابستگی‌های موجود به دست آورد. در جدول ۳ اولین فرد در اولین نسل و شمای پایگاه دادهٔ متناظر آن نشان داده شده است.

جدول ۲. یک مثال برای طراحی تکاملی

ورودی‌های مساله	
مجموعه صفت‌ها	A, B, C, D, E, F, G, H, I, J
وابستگی‌های تابعی	A→B, A→C, A→D, A→E, A→F, F→G, F→H, H→I, H→J
ضرایب اهمیت	a=1, b=0, c=0
تعداد افراد جمعیت	100
تعداد تکرار تولید نسل	100
تعداد کل حالات	Bell(10)=115975

در این مسأله چون تنها شاخص هزینه واحد به هنگام سازی مهم بوده است، مقادیر معیارهای مربوط به این شاخص نیز محاسبه شده است. با استفاده از شبکهٔ عصبی مربوط به این شاخص، مقدار شاخص را برای هر شمای رابطه تخمین می-زنیم و با استفاده از آن، مقدار θ^{du} برای این فرد محاسبه می‌شود. بنابراین مقدار

جدول ۳. یک مثال برای طراحی تکاملی

	RV_INNF	RN_IND1	RN_IND2	θ^u
R_1 (ABCDEF)	0	0	0	5.23
R_2 (B)	0	1	0	5.79
R_3 (C)	0	1	0	5.79
R_4 (D)	0	1	0	5.79
R_5 (E)	0	1	0	5.79
R_6 (FGH)	0	0	0	5.23
R_7 (G)	0	1	0	5.79
R_8 (HIJ)	0	0	0	5.79
R_9 (I)	0	1	0	5.79
R_{10} (J)	0	1	0	5.79

p1:
1000000000 0100000000 0010000000 0001000000 0000100000
0000010000 0000001000 0000000100 0000000010 0000000001

$R_2[B] \supseteq R_1[B], R_3[C] \supseteq R_1[C], R_4[D] \supseteq R_1[D], R_5[E] \supseteq R_1[E], R_6[F] \supseteq R_1[F], R_6[G] \supseteq R_7[G], R_6[H] \supseteq R_8[H], R_8[I] \supseteq R_9[I], R_8[J] \supseteq R_{10}[J]$

خطای پایگاه داده برابر است با ۵/۷۹ (بیشینهٔ θ^u برای شماهای رابطه) و اگر بیشینهٔ خطا را ۲۰ فرض کنیم مقدار شایستگی این فرد ۱۴/۲۱=۲۰-۵/۷۹ خواهد بود. برای تمام افراد یک جمعیت این محاسبات انجام می‌شود تا مقدار شایستگی آن‌ها به دست آید. افراد بر اساس مقدار شایستگی به صورت صعودی مرتب می-شوند و سایر مراحل برای تولید نسل بعد انجام می‌شود.

نتیجه به دست آمده از انجام طراحی تکاملی شمای پایگاه داده‌ای است با شماهای رابطهٔ R_3(HIJ)، R_2(FGH) و R_1(ABCDEF). تعداد حالت‌های مختلف ۱۱۵۹۷۵ است که با استفاده از طراحی تکاملی این تعداد به ۶۵۲۲ حالت کاهش یافته است. در این مسأله، فقط هزینه به هنگام سازی در نظر گرفته شده است و به این دلیل تمام ضرایب غیر از ضریب مربوط به هزینه به هنگام سازی صفر هستند. همان طور که دیده می‌شود، شماهای رابطه در پاسخ به دست آمده به شکل نرمال BCNF هستند. به این دلیل که خطا از BCNF بودن باعث افزایش هزینه به هنگام سازی می‌شود، افراد با شایستگی بیشتر شماهایی خواهند بود که به شکل نرمال نزدیک‌تر باشند.

۵- نتیجه گیری

طراحی شمای پایگاه دادهٔ رابطه‌ای مرحله‌ای است که نقش طراح در آن بسیار مؤثر است. با استفاده از مدلی که می‌تواند کیفیت شما را تخمین بزند، روشی ارائه شد که شمای پایگاه داده‌ای با کیفیت مناسب ارائه می‌کند. این روش مبتنی بر الگوریتم ژنتیک است. بهترین شماهای موجود در هر نسل برای تولید مثل در نسل بعد استفاده می‌شوند و با این روش، نسل‌ها به سمت تولید افراد بهتر همگرا می‌شوند. فضای پاسخ برای تعدادی صفت داده شده مرتبه‌ای نمایی دارد و جستجوی تمامی این فضا برای رسیدن به پاسخ هزینه بسیاری خواهد داشت. بنابراین استفاده از یک شیوهٔ تکاملی که تنها بخش‌هایی از فضای پاسخ را جستجو می‌کند و پاسخ مناسب را با جستجوی کمتر می‌یابد کاملا قابل قبول است.

آن‌چه در این تحقیق قابل توجه است تعریف و اعتبارسنجی معیارهایی برای ارزیابی پایگاه داده رابطه‌ای و استفاده از الگوریتم ژنتیک در ایجاد روشی برای طراحی پایگاه‌دادهٔ رابطه‌ای است. این روش که با استفاده از مدلی بر پایه اندازه‌گیری نرم‌افزار برای ارزیابی کیفیت شمای پایگاه داده ساخته شده است، امکان طراحی خودکار شمای پایگاه داده را فراهم می‌آورد. در این روش، تکنیک‌هایی که به طور کلاسیک در طراحی شمای پایگاه دادهٔ رابطه‌ای وجود دارند و اغلب اکتشافی[۳۶] هستند، به صورت داده‌های آموزشی به شبکهٔ عصبی منتقل شده و از آن برای ارزیابی‌های بعدی استفاده می‌شود. از الگوریتم ژنتیک نیز، به عنوان روشی برای یافتن شمای پایگاه دادهٔ مناسب به صورت تکاملی و با جستجوی تعداد حالات کم در فضای پاسخ استفاده شده است.

روش مطرح شده می‌تواند پایه اصلی برای ابزاری باشد که با داشتن شرایط یک مسأله، یعنی صفت‌ها، وابستگی بین آن‌ها و همچنین ضرایب اهمیت برای شاخص‌های کیفیتی، به طور خودکار مناسب‌ترین شمای پایگاه داده را طراحی کند. البته برای رسیدن به این هدف نیاز است مدل تخمین زننده با ورودی‌های بیشتری از جمله وابستگی‌های پیچیده‌تر آموزش داده شود و پارامترهای محیطی بیشتری برای تطبیق با شرایط واقعی درنظر گرفته شود.

مراجع

[1] W. A. Greene, "Genetic algorithms for partitioning sets," *International Journal on Artificial Intelligence Tools*, Vol. 10, pp. 225-241, 2001.

[2] M. Klazar, Bell numbers, "their relatives, and algebraic differential equations," *Journal of Combinatorial Theory Series A*, Vol 102, No.1, pp. 63--87, 2003.

[3] D. Beasley, D. R. Bull, and R. R. Martin, "An Overview of Genetic Algorithms:Part I, Fundamentals," *University Computing*, Vol. 15, No. 2, pp. 58-69, 1993.

[4] D.L. Moody, "What makes a good data model? Evaluating the quality of entity relationship models," *Proceedings of the 13th International Conference on the Entity Relationship Approach*, 1994.

[5] G. S. J. Krogstie, O.I. Lindland, "Towards a deeper understanding of quality in requirements engineering," *Proceedings of the 7th International Conference on Advanced Information System Engineering (CAISE)*, pp. 82-95, 1995.

[6] R. Schuette and T. Rotthowe, "The guidelines of modelling - an approach to enhance the quality in information models," *Proceedings of the Seventeenth International Conference on Conceptual Modeling*, pp. 240-254, Singapore, 1998.

[7] R. Gray, B. Carey, N. McGlynn, and A. Pengelly, "Design metrics for database systems," *BT Technology Journal*, Vol. 9, No. 4, pp. 69-79, 1991.

[8] D. Moody, "Metrics for evaluating the quality of entity relationship models," *Proceedings of the International Conference on Conceptual Modeling*, pp. 211-225, 1998.

[9] M. Genero, M. Piattini, C. Carlo, and M. Serrano, "Measures to get better quality databases," *Proceedings of the International Conference on Enterprise Information Systems*, pp. 49-55, 2000.

[10] C. Polo and F. Ruiz, *Advanced Database Technology and Design*, Chapter 14: Database Quality Eds, Mario Piattini and Oscar Diaz, pp. 485-509, Artech House, 2000.

[11] M. Casanova, R. Fagin, and C. Papadimitriou, "Inclusion dependencies and their interaction with functional

dependencies," *Journal of Computer and System Sciences*, Vol. 21, No. 1, pp.29-54, 1984.

[12] M. Levene, "Justification for inclusion dependency form," *IEEE Transactions on Knowledge and Data Engineering*, Vol.12, pp. 281-291, 2000.

[13] J. Mitchell, "The imlication problem for functional and inclusion dependencies," *Information and Control*, Vol. 56, pp. 154-173, 1983.

[14] R. Fagin, "Acyclic database schemes (of various degrees): A painless introduction," *Proceedings of CAAP83 the 8th Colloquium on Trees in Algebra and Programming*, pp. 65-89, Springer-Verlag, LNCS 159, 1983.

[15] F. Maier and M. Yannakakis, "On the desirability of acyclic database schemes," *Journal of the ACM*, Vol. 30, pp. 479-513, 1983.

[16] V. Basili and D. Weiss, "A methodology for collecting valid software engineering data," *IEEE Transactions on Software Engineering*, Vol. 10, pp. 728-738, 1984.

سیدحسن میریان‌حسین‌آبادی 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، تئوری انواع و ریاضیات ساختی، معیارها و اندازه‌گیری نرم‌افزار، معماری نرم‌افزارهای با قابلیت پیکربندی مجدد، توصیف و درستی‌یابی صوری معماری نرم‌افزار و پایگاه داده‌های اکس-ام-ال است.

آدرس پست الکترونیک: hmirian@sharif.edu

امین روزبهانی مدرک کارشناسی و کارشناسی ارشد خود را از دانشگاه صنعتی شریف در سالهای ۸۱ و ۸۳ در رشته مهندسی نرم‌افزار اخذ نموده است. ایشان هم اکنون به عنوان مهندس ارشد نرم‌افزار در شرکت پسنا به کار مشغول است.

آدرس پست الکترونیک: ruzbahan@yahoo.com

.

---

[1] Exhaustive
[2] Set Partitioning
[3] Bell Number
[4] Evolutionary
[5] Individual
[6] Fittness
[7] Functional Dependency
[8] Inclusion Dependency
[9] Boyce-Codd Normal Form
[10] Inclusion Normal Form
[11] Universal Relation Assumption
[12] Acyclic Scheme
[13] Monotone Join Expression
[14] sequential join expression
[15] Unit Query Cost
[16] Database Query Cost
[17] Unit Update Cost
[18] Database Update Cost
[19] Relation Space
[20] Attribute Space
[21] Database Space Utilization
[22] Experiment
[23] Case Study
[24] Artificial Neural Networks
[25] Analysis of Variance
[26] Backpropagation
[27] Activation function
[28] Sigmoid
[29] Mean Squared Error
[30] Cross Validation
[31] Evolutionary Computation
[32] Fitness Value
[33] Genetic Algorithms
[34] Mutation
[35] Crossover
[36] Heuristic