سیستم خبره عیب‌یابی و تشخیص علائم در پستهای انتقال نیرو

حیدرعلی شایانفر - داریوش زارع - احمدرضا میرزائی
دانشگاه علم و صنعت

چکیده :

در این مقاله سیستم خبره‌ای بنام " EXSFALP " معرفی میگردد که جهت عیب‌یابی و تشخیص علائم در پستهای فشار قوی طراحی شده است. برای پیاده سازی این سیستم از یک پوسته سیستم خبره (Expert System Shell) بنام Xiplus استفاده شده است . هدف از طراحی چنین سیستمی فراهم نمودن اطلاعات لازم برای افراد غیر متخصص است تا این افراد بتوانند با تبادل اطلاعات از طریق ردیابی آلارم‌های داخل پست ، عیوب موجود را تشخیص دهند. همچنین این سیستم میتواند بعنوان یک مشاور در کنار افراد متخصص مورد استفاده قرار گیرد.

شرح مقاله :

به محض اینکه خطائی در پست اتفاق می‌افتد ، ثبات های نشان دهنده ، دیاگرام‌های سمبلیک ، علائم رله‌ها و فرمان قطع دژنکتورها فعال شده و به اپراتور هشدار می‌دهند و او را از بروز خطا آگاه می‌سازند. از روی این علائم میتوان میزان وخامت اوضاع در هنگام بروز خطا را مورد ارزیابی قرار داد. بعلاوه این علائم اپراتورها و مسئولین پست را هدایت می‌کنند تا اقدامات لازم را برای برقدار کردن مجدد پست انجام دهند. چنانچه برقراری مجدد برق بطور اتوماتیک غیرممکن باشد ، راه دیگری وجود ندارد مگر اینکه به گروه تعمیرات

اعلام شود تا وضعیت را پیگیری و تعمیرات لازم را انجام دهند همچنین با توجه به محل عیب و درجه اهمیت آن ممکن است قابلیت اطمینان سیستم بطور قابل ملاحظه‌ای کاهش یابد.

با در نظر گرفتن تمامی این مسائل ، نکات زیر خاطر نشان میگردد :

الف - عواقب برخی از عیوب در پست ممکن است برای مصرف کننده ها بسیار خطرناک باشد.

ب - فاصله مکانی پست از آزمایشگاهی که گروه تعمیرات در آنجا مستقر هستند گاهی اوقات مشکل آفرین بوده و مسائلی را بدنبال خواهد داشت.

ج - اصولا" پستها بندرت دچار ایراد میگردند ، از اینرو بکار گرفتن تمام وقت یک متخصص در محل پست ، امری غیرقابل توجیه است . مضافا" اینکه برخی از عیوب بگونه‌ای هستند که برای برطرف کردن آنها یک سری اعمال ساده وجود دارد که میتواند براحتی توسط تکنیسین های غیرمتخصص نیز انجام شود .

د - متخصصین برای اینکه بتوانند عیب دستگاهی را برطرف کنند میبایست اطلاعات مشخصی را در مورد آن داشته باشند و کسب این اطلاعات در مواقع اضطراری موجب اتلاف وقت و تاخیر در امر راه‌اندازی میشود .

با توضیحاتی که داده شد ، استفاده از یک سیستم خبره کاملا" منطقی به نظر میرسد. سیستمی که از طریق میکروکامپیوتری با هزینه‌ای اندک بطور دائم در پست نصب میگردد و دائما" تمام علائم و حفاظتهای لازم را کنترل میکند. با بکارگیری چنین سیستم خبره‌ای امکان داشتن دیدگاه سریعتر و دقیقتر برای آگاهی از وضعیت پیش از بروز عیب در پست ، میسر میگردد و قبل از اعزام متخصصین به محل میتوان اقدامات لازم جهت فائق آمدن بر عیوب را انجام داد. در نتیجه بطور چشمگیری در وقت صرفه‌جویی شده و بر قابلیت اعتماد کل سیستم قدرت و کیفیت خدمات به مصرف کننده ، اثر مطلوبی بر جای خواهد گذاشت.

۱ - توصیف سیستمهای خبره :

سیستمهای خبره بخشی از رشته گسترده هوش مصنوعی بوده و به حل مسائلی می‌پردازند که نیاز به هوش و تخصص انسانی دارند .

یکی از علل بوجود آمدن این سیستمها ، کنجکاوی و توجه به شیوه حل مشکلاتی است که هرچند خود انسان کفایت حل آنها را دارد ، لیکن برنامه‌ریزی آن بر روی کامپیوتر بسادگی قابل اجرا نیست . منظور از سیستم خبره یک برنامه کامپیوتری است که در یک محدوده خاص ، درجه‌ای از خبره‌گی قابل مقایسه با انسان را در حل مسائل از خود نشان می دهد .

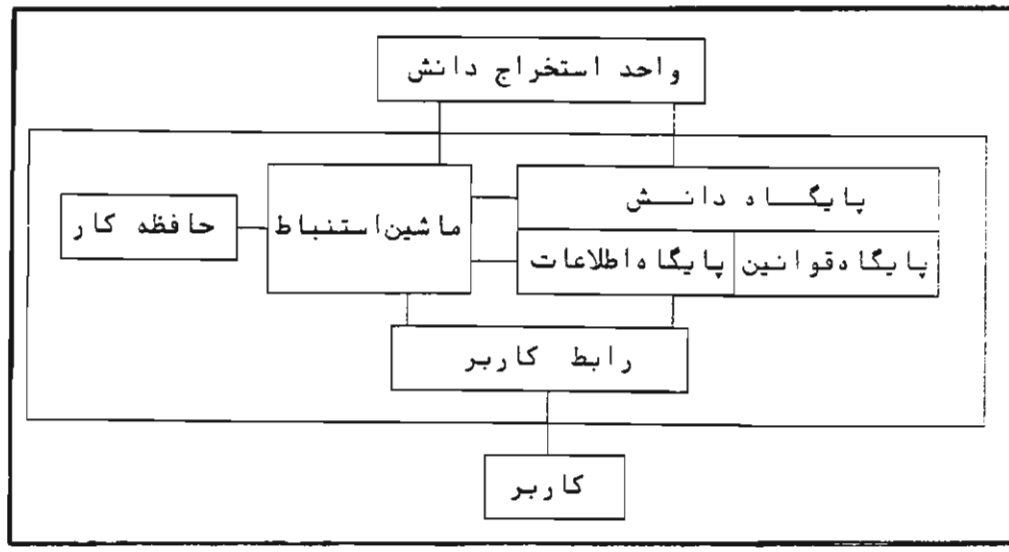
این نوع برنامه‌ها از مجموعه معلومات و دانش افراد خبره و نیز راههای استدلالی ایشان در استفاده از این معلومات برای انجام عمل تخصصی خویش ، بهره می‌جویند و اغلب با اطلاعات غیردقیق و غیرقطعی سر و کار دارند . چنین سیستمی باید در حد امکان قادر به ارائه توضیحاتی پیرامون تصمیمات خود باشد .

این توضیحات می‌بایست تا جای امکان برای شخص ناآشنا با کامپیوتر قابل درک باشد . از خصوصیات دیگر سیستمهای خبره قابلیت انعطاف آنهاست . از آنجائیکه این سیستمها با مجموعه دانش انسانی سر و کار دارند و این مجموعه ممکن است در هر لحظه تغییر کند . حقایقی به آن اضافه گردد و یا قواعدی از آن کاسته شود ، لازم است که یک سیستم خبره ، همچون فرد متخصص ، خود را با این تغییرات تطبیق داده و از مجموعه دانش جدید در انجام فعالیت خود بهره گیرد.

برای استفاده از سیستمهای خبره الزاما" احتیاج به سخت افزار و یا کامپیوتر بخصوصی نمی‌باشد . اما گاهی برای برنامه سازی چنین سیستمهایی از زبانهای کامپیوتری بخصوصی مانند لیسپ (Lisp) و پرولوگ (Prolog) استفاده میشود. [1، 2] از سالهای ۱۹۷۰ تا به امروز سیستمهای خبره متعددی در زمینه‌های تخصصی گوناگون مورد استفاده قرار گرفته‌اند که از جمله معروفترین و ابتدائی‌ترین آنها DENDRAL [3] / MYCIN [4] می‌باشند.

۲ - ساختار سیستمهای خبره [5] :

شکل (۱) ساختار یک سیستم خبره را نشان میدهد .



شکل ۱- ساختار سیستمهای خبره

همانطور که در این شکل دیده می‌شود یک سیستم خبره از واحدهای زیر تشکیل شده است .

۱-۲- واحد استخراج دانش (Knowledge Elicitaion) :

وظیفه این واحد کسب دانش از متخصصین و سازمان دادن آن می‌باشد. اکتساب دانش بعلت مشکلی که متخصصین در توضیح تخصص خود با آن مواجهند ، یکی از دشوارترین مراحل در ساختن سیستمهای خبره بشمار می‌رود. از آنجائیکه معلومات چنین اشخاصی طی سالها تجربه کسب شده است ، این گونه علوم برای متخصص حالت طبیعی و غیرقابل توضیح پیدا می‌کند. برای اکتساب این معلومات از روشهای گوناگونی استفاده میشود که از آن جمله میتوان به موارد زیر اشاره نمود .

- اکتساب دانش از روی نوشتار
- اکتساب دانش بوسیله مصاحبه آزاد با متخصص
- ارائه یک مسئله به متخصص و مشاهده و ضبط راه حل آن
- زیر نظر قرار دادن و تحلیل وظایف یک متخصص به هنگام انجام کارهای روزمره.

۲-۲- واحد پایگاه دانش (Knowledge Base) :

پایگاه دانش همانطور که از نامش پیداست ، بخشی است که در آن دانش یک سیستم خبره انباشته میشود . این واحد یکی از مهمترین بخشهای سیستم خبره

است . زیرا کیفیت این بخش است که تعیین کننده چگونگی برخورد کاربر یا متخصص با سیستم میباشد . آنچه که در این میان از اهمیت بسیار زیادی برخوردار است ، نحوه نمایش دانش میباشد . نمایش دانش بصورت کارا و مؤثر کلید موفقیت تمام سیستمهای خبره است . در حال حاضر چهار روش اصلی برای نمایش دانش و ارتباط میان آنها وجود دارد که عبارتند از :

الف - قواعد تولید (production rules) - بصورت قانونهای " اگر ... آنگاه..." نمایش داده میشود که این قواعد الگوهایی را مشخص میکنند و نتیجه آنها ممکن است یک عمل یا یک گزاره باشد.

ب - گزاره‌های منطقی (Production Logic) - این روش در زبان پرولوگ استفاده میشود.

ج - شبکه‌های معنایی (Semantic nets) - در این شبکه‌ها ارتباط میان موضوعات دامنه با گره‌های شبکه نمایش داده میشود .

د - نمایش کالبدی (Frame based) -

۳-۲- ماشین استنباط (Inference engine) :

این واحد در سیستمهای خبره برنامه‌ای است که قواعد و دانش انباشته شده در پایگاه دانش را تحلیل کرده و به نتیجه گیری منطقی راه مییابد . بطور کلی طریقه جستجو در پایگاه دانش به دو صورت پیشرو (Forward chaining) و پس رو (Backward chaining) انجام میشود . در استدلال پیشرو مرحله تحلیل با بررسی حکم‌ها و داده‌های اولیه شروع میشود. پس از آن سیستم از دانش خود برای رسیدن به هدف استفاده میکند . همچنین امکان دارد که چنین سیستمی در حین پیشرفت تحلیلی خود احتیاج به داده‌های بیشتری پیدا کند. این نیاز از طریق پرسش از کاربر و دست یابی به داده‌های جدید برطرف میشود. در انتها سیستم به هدف مورد نظر خود خواهد رسید. در استدلال پس رو سیستم با نزدیک‌ترین هدفی که به آن برخورد میکند ، استنتاج خود را شروع کرده و در جهت معکوس حرکت میکند. این بار نیز در صورت نیاز سؤالاتی از کاربر میشود و پاسخ هایی که از این راه بدست میآید امکان رسیدن به هر یک از اهداف نهایی را تعیین میکند. هدف جدیدی که به نظر امکان پذیرتر از بقیه اهداف میآید انتخاب شده و این مرحله

چندین مرتبه تکرار میشود تا اینکه یکی از اهدافی که بیش از بقیه درست به نظر می‌رسد بعنوان حل مسئله به کاربر ارائه میشود. سیستمهای استنباطی پیشرو معمولا" در روبوت‌ها (robots) و برنامه‌ریزیهای مدیریتی بکار رفته و سیستمهای پس رو در زمینه‌های انتخاب ، عیب یابی و تشخیص ، مورد استفاده قرار می‌گیرند.

۴-۲- رابط کاربر (User Interface) :

بخشی از سیستم است که محاوره میان کاربر و سیستم را امکان‌پذیر می‌سازد. در واقع انجام اعمال دریافت اطلاعات و اعلام نتایج از طریق این واحد انجام می‌شود. همچنین از این طریق است که کاربر چگونگی پیشنهاد سیستم را مورد سئوال قرار می‌دهد . این واحد گاهی بصورت گرافیکی و گاهی با استفاده از زبان طبیعی با کاربر رابطه برقرار می‌کند.

۳ - انتخاب ابزار :

برای پیاده سازی سیستم‌های خبره نرم‌افزارهای آماده‌ای تحت عنوان پوسته سیستم‌های خبره (Expert system shell) وجود دارند که در آنها امکانات تعریف و ایجاد پایگاه دانش در مورد یک زمینه خاص ، بعلاوه مکانیزم‌های استنباط ، رابط کاربر و واحد توضیحات وجود دارد . در حال حاضر پرطرفدارترین پوسته‌ها که بر روی کامپیوترهای PC قابل اجرا می‌باشند عبارتند از :

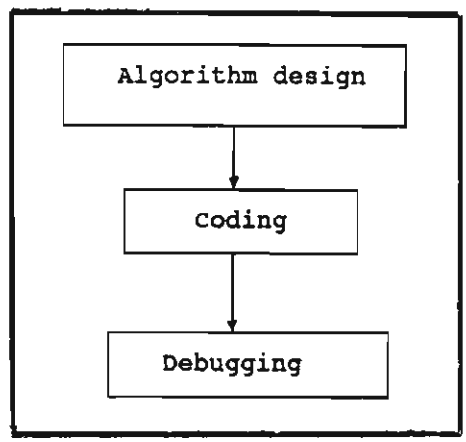
EXSYS ، XIPLUS ، CRYSTAL و KES

برای پیاده سازی سیستم EXSFALP با توجه به نوع مسئله ، ترجیح داده شد که از پوسته XIPLUS استفاده شود . XIPLUS از تجربیات عملی ناشی از ساخت سیستم‌های خبره کاربردی ، استخراج شده است . زبان طراحی شده در آن بسیار ساده و بخوبی قابل فراگیری است و علاوه بر آن ، امکانات نیرومندی را برای کاربردهای پیشرفته دار است.

۴ - توسعه سیستم خبره Exsfalp :

طراحی و پیاده‌سازی برنامه‌های هوشمند معمولا" شامل روشهای متفاوتی نسبت به برنامه‌های عددی می‌باشد . برای نشان دادن این تفاوت‌ها ، مسئله تشخیص خطا

در یک سیستم قدرت را بوسیله هر دو نوع برنامه مورد بررسی قرار می‌دهیم. هدف این است که بوسیله مشاهده علائم رله‌ها و نیز فرمان قطع دژنکتورها ، عنصر معیوب در یک سیستم قدرت را مشخص نمائیم.
شکل (۲) طرحی اجمالی از روند پیاده سازی یک برنامه معمولی را نشان میدهد.



شکل ۲- روند پیاده سازی یک برنامه عددی

در یک برنامه معمولی غالبا" اجرای عملیات بصورت متوالی انجام شده و انعطاف کمی در ترتیب انجام مراحل مختلف آن وجود دارد . بر طبق مدل فوق مرحله اول شامل طراحی الگوریتم مسئله می‌باشد. با این توضیح که هیچ الگوریتم مشخصی در زمینه تشخیص خطا وجود ندارد و بسته به نوع مسئله ممکن است راه حل‌های متفاوتی ارائه گردد.

اصولا" تشخیص خطا بر مبنای علائم رله‌ها و فرمان قطع دژنکتورها انجام می‌شود. برای اینکار سیستم را به چند منطقه که از نظر حفاظتی از یکدیگر مجزا می‌باشند ، تقسیم کرده وسپس الگوریتم تشخیص زیر را در مورد هر منطقه بکار می‌بندیم.

عناصر تحت حفاظت مربوط به کلیه رله‌های عملکرده در یک منطقه و نیز ارتباط بین آنها را مشخص می‌کنیم . به این ترتیب که ابتدا یکی از رله‌های عملکرده را انتخاب کرده و کلیه عناصر سیستم که بروز عیب در آنها می‌توانسته منجر به عملکرد رله مورد نظر شود را تعیین می‌کنیم . پس از آنکه این کار را در مورد تمام رله‌های عملکرده انجام دادیم ، آنگاه عنصر یا عناصر حفاظت شده مشترک بین آنها بدست آمده و ارتباط بین رله ها مشخص می‌گردد . در این

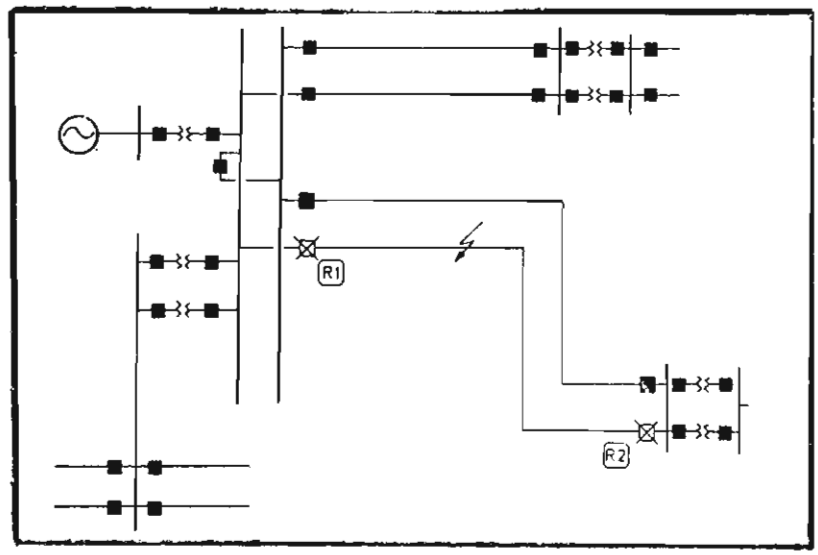
شرایط سه حالت ممکن است بروز نماید .

الف - اگر ارتباط فقط شامل یک عنصر باشد ، در آن صورت آن یک عنصر ، همان عنصر معیوب در منطقه خواهد بود.

ب - اگر دو یا تعداد بیشتری عنصر ، ارتباط بین رله ها را تشکیل دهند ، آنگاه می‌توان دریافت که تعدادی از رله‌ها وظیفه خود را بخوبی انجام نداده و عمل نکرده‌اند . در این حالت یکی از این عناصر را بعنوان عنصر معیوب احتمالی در نظر گرفته و با توجه به آن رله‌هائی که ممکن است اشتباها" عمل نکرده باشند را چک میکنیم . و این عمل را تا آنجائیکه صحت معیوب بودن یکی از این عناصر مشخص گردد ، ادامه می‌دهیم .

ج - حالت سوم این است که هیچ عنصر مشترک وجود نداشته باشد . در این صورت نتیجه می گیریم که یا بیشتر از دو خطا در منطقه اتفاق افتاده است و یا برخی از رله‌ها عملکرد نادرست داشته‌اند . لذا سعی خواهیم نمود مجموعه‌ای از رله‌های اکتیو را طوری انتخاب کنیم که عناصر تحت حفاظت آنها دارای یک عضو مشترک غیر تهی باشند . در صورت موفقیت الگوریتم تشخیص را برای هر مجموعه بکار می بریم و گرنه نتیجه می‌گیریم که رله‌ها عملکرد نادرست داشته‌اند.

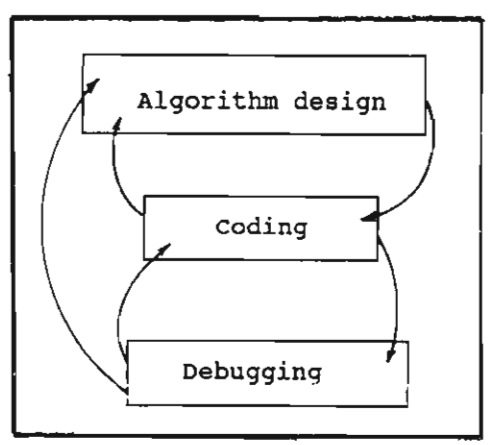
همانطور که مشاهده می‌شود الگوریتم فوق در غالب زبان طبیعی نوشته شده و به همین شکل در کامپیوتر قابل اجرا نیست . بنابراین لازم است که این الگوریتم را بصورت کد در آوریم (مرحله Coding) .برای این کار ابتدا بایستی آنرا در غالب یک زبان قابل فهم برای ماشین بنویسیم و کد اصلی آنرا به یکی از زبانهای برنامه نویسی رایج مثل فرترن ، پاسکال و یا c کامپایل نمائیم که این خود شامل مراحل کامپایل معمولی ،LINK ، LOAD و RUN می‌باشد و مستلزم تغییرات مکرر در حین انجام کار است (مرحله Debugging) . نیروی انسانی لازم برای انجام مراحل مختلف این برنامه ، تقریبا" یک و نیم نفر سال می‌باشد و نتیجه نهائی آن برنامه‌ای است که اگر به زبان فرترن نوشته شود در حدود پانزده هزار خط خواهد بود[10] . زمان اجرای این برنامه برای تشخیص خطائی که در شکل (۳) نشان داده شده است ، در یک کامپیوتر pc با پروسسور 32 بیتی حدود ده ثانیه بطول می‌انجامد.



شکل ۳- دیاگرام قسمتی از یک سیستم قدرت

همانطور که ملاحظه می‌شود ، نوشتن چنین برنامه‌ای احتیاج به نیروی انسانی زیادی دارد و علت اصلی آن این است که الگوریتم تشخیص را نمی‌توان بصورتی ساده و مختصر توصیف نمود . این نوع مسائل در واقع از نوع مسائل بدساختار (ill- structure) بوده و نمی‌توان الگوی معینی که بطور کامل مشخص کننده الگوریتم مورد نظر باشد را ارائه داد. اصولا" هنگامی که زبانهای برنامه نویسی عددی در مورد مسائلی بکار میروند که بر مبنای دانش استوارند ، کمتر قابل خواندن و قابل فهم بوده و این خود باعث می‌شود که اینگونه برنامه‌نویسی بیشتر دچار اشکال شود .

بکارگیری یک سیستم خبره برای حل این مسئله با آنچه که گفته شد کاملا" متفاوت است . شکل (۴) روند پیاده سازی یک سیستم خبره را نشان میدهد .



شکل ۴- روند پیاده‌سازی یک سیستم خبره

در این مدل نیز همانند مدل قبل ، سه مرحله طراحی الگوریتم ، کدینگ و دیباگ کردن وجود دارد. ولی با این تفاوت که ارتباط بین آنها بسیار قوی بوده و ترتیب اجرای مراحل مختلف با انعطاف بیشتری انجام میشود . خروجی فاز طراحی یک پایگاه دانش خواهد بود که میتوان آنرا " به همان صورتیکه هست " در کامپیوتر اجرا نمود. بعبارت دیگر هدف از انجام فاز اول ، استخراج الگوی اصلی الگوریتم میباشد و چون این الگو مستقیما" در کامپیوتر قابل اجرا است لذا هم کاربر و هم برنامه‌نویس میتوانند چگونگی انجام کار را در هر مرحله از اجرای برنامه مشاهده نمایند . چنانچه الگوی اصلی نیازهای برنامه را برآورده سازد در آنصورت میتوان بدون هیچ تغییری پایگاه دانش را مستقیما" در ماشین پیاده کرد . ولی در غیر اینصورت انجام یکسری تغییرات جزئی در حین پیاده سازی برنامه ضروری خواهد بود.

نیروی انسانی لازم بمنظور طراحی و پیاده سازی سیستم خبره Exsfalp چیزی در حدود سه چهارم نفر- سال بوده که اکثر این زمان صرف انجام مرحله اول ، یعنی طراحی الگوریتم شده است.

نمونه‌ای از پایگاه دانش که در محیط نرم‌افزار Xiplus نوشته شده در زیر آمده است . همانطور که دیده میشود ، در این سیستم قوانین بزبان فارسی نوشته شده و کاربر میتواند از طریق زبان طبیعی با سیستم محاوره نماید.

FACT 1 : " R1 رله " is " یک رله پشتیبان برای عنصر z "

FACT 2 : " R1 رله " is " یک رله جریان زیاد جهت یاب "

FACT 3 : " R1 رله " is " y1 در موقعیت "

FACT 4 : " R2 رله " is " یک رله پشتیبان برای عنصر z "

FACT 5 : " R2 رله " is " یک رله جریان زیاد جهت یاب "

FACT 6 : " R2 رله " is " y2 در موقعیت "

rule :

IF " R1 رله " is " عمل کرده "

and " R2 رله " is " عمل کرده "

and " Y1" not= " y2 "

then " عنصر معیوب " is " z"

البته باید توجه داشت که برای برقرار بودن قانون فوق لازم است که R1 ، R2 هر دو رله‌های پشتیبان برای عنصر z باشند و چون این شرایط در fact های 1 ، 4 گنجانده شده است ، بنابراین ضرورتی به ذکر آنها در خود قانون نبوده است.

۵ - خصوصیات و ویژگیهای سیستم EXSFALP :

هنگام بروز عیب در پست ، می‌توان سیستم خبره را فعال نمود تا شروع به عیب‌یابی کند . به این ترتیب لیست تمام رله‌های موجود در پست برروی صفحه مونیتور ظاهر شده و برنامه از کاربر میخواهد تا رله‌هائی که عمل کرده‌اند را مشخص نماید . بدین ترتیب کاربر میتواند براحتی از طریق صفحه کامپیوتر رله‌های عملکرده را علامت گذاری نماید .

بعد از آن سیستم خبره شروع به عملیات کرده و سعی میکند توجیهی برای عملکرد این رله‌ها بیابد . ماشین استنتاج یک به یک فرآیندها را گرفته و مسیر لازم برای جستجو را انتخاب می‌نماید . در خلال این کار سیستم سؤالاتی را برای کاربر مطرح میکند و یا از او می‌خواهد تا عمل معینی را انجام داده و در مورد نتایج آن به سیستم اطلاع دهد . همزمان لیستی شامل تمام پاسخهای ممکن به آن سؤال مورد نظر و همچنین تمام نتایج ممکن از انجام آن عمل معین بر روی صفحه کامپیوتر ظاهر می‌شود و کاربر موظف است جوابهائی را انتخاب کند که در لیست مذکور وجود داشته و با آن مطابقت دارد. به این ترتیب سیستم بطور متقابل با کاربر وارد عمل شده تا به تشخیص صحیح دست یابد. این ارتباط همانگونه که ذکر شد بصورت سؤال و جواب بوده و لذا به زبان طبیعی بسیار نزدیک است .

گاهی ممکن است در یک جستجو و یا در جستجوی دیگری ، سؤال مشابهی در پایگاه دانش ظاهر گردد. به محض اینکه سیستم سؤالی را پرسید پاسخ آن در پایگاه داده‌ها نگهداری می‌شود و چنانچه بعداً در حین پرس و جو همین سؤال در پایگاه داده‌ها یافت شود دیگر پرسیده نمی‌شود و اجرای برنامه بر طبق همان پاسخ

ادامه خواهد یافت. ضمن اجرای برنامه ممکن است سیستم گزارشی را به کاربر ارائه دهد ، که در آن علت عملکرد بعضی از رله‌ها و آخرین نتایج بدست آمده توضیح داده می‌شود. به غیر از این ، سیستم می‌تواند تمام مراحلی را که برای تعیین آخرین تشخیص انجام شده و همچنین نتایج اعمالی که کاربر انجام داده به انضمام وضعیت کل سیستم را نیز ارائه دهد. نهایتا" وقتی که عملکرد کلیه رله‌ها توجیه شد، تمام عملیات متوقف می‌گردد و در نتیجه خسارات احتمالی وارده به تجهیزات برآورده شده و کارهائی که می‌بایست برای تعمیرات صورت گیرد نیز گزارش می‌شود.

بمنظور آشنائی بیشتر باExsfalp در اینجا یک نمونه از محاوره میان کاربر و سیستم آورده شده است . در آغاز اجرای برنامه لیستی از کلیه رله‌های موجود در پست بر روی صفحه مونیتور ظاهر می‌شود. (در مقابل پاسخهائی که کاربر به هر پرسش داده است علامت << قرار داده شده است)

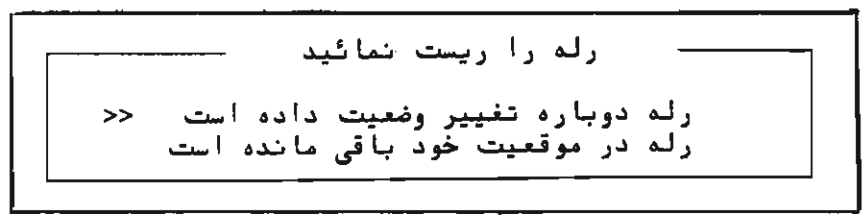
```
رله‌های عمل کرده را مشخص نمائید

رله جریان زیاد ترانسفورماتور۱
رله جریان زیاد ترانسفورماتور۲
رله بوخهلتس ترانسفورماتور۱ <<
رله بوخهلتس ترانسفورماتور۲
رله دیفرانسیل ترانسفورماتور۱
رله دیفرانسیل ترانسفورماتور۲
```

در این مورد کاربر ورودی سوم یعنی " رله بوخهلتس ترانسفورماتور ۱ " را انتخاب می‌کند با این کار سیستم خبره در مورد وضعیت این رله از کاربر سؤالاتی را می‌پرسد .

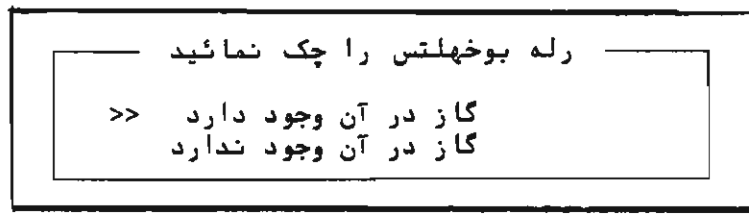
```
کدام مرحلهٔ رله عمل کرده است ؟

مرحلهٔ اول رله عمل کرده است (هشدار) <<
مرحلهٔ دوم رله عمل کرده است ( قطع )
```

در اینجا سیستم از کاربر می‌خواهد که رله را ریست (reset) نماید و نتایج احتمالی اینکار را بصورت منوی زیر در اختیار کاربر قرار می‌دهد تا او

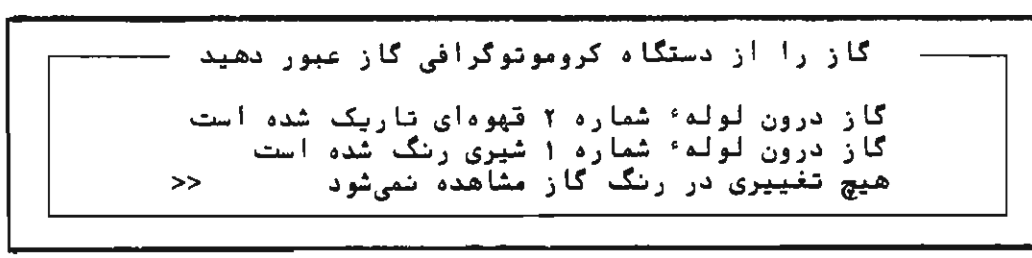
یکی از آنها را مجددا" انتخاب نماید .



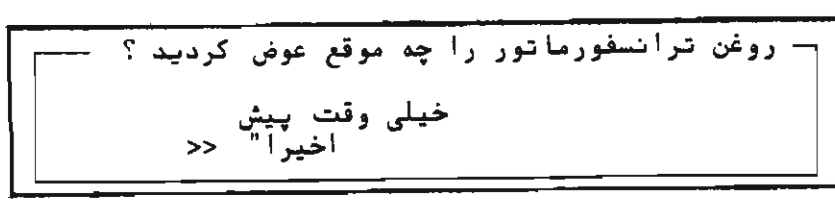
کاربر عبارت اول را انتخاب کرده است وعمل بدی بهمراه منوی مربوطه بصورت زیر بر صفحه کامپیوتر ظاهر می‌شود .



حال فرض می‌کنیم که کاربر عبارت " گاز در آن وجود دارد " را انتخاب کرده باشد و عمل بعدی بصورت زیر به اجرا در می‌آید.



در اینجا عبارت آخر انتخاب می‌شود و متعاقب آن سؤال بعدی پرسیده می‌شود.



جواب این سؤال " اخیرا " میباشد که با انتخاب آن عملکرد رله توجیه شده و در غالب عبارات زیر بر روی صفحه کامپیوتر ظاهر میگردد .

هوای خالص در داخل رلهٔ بوخهلتس موجود است . این هوا میبایست در طول مدت تعویض روغن در ترانسفورماتور محبوس مانده و سپس از طریق گردش طبیعی روغن به سمت رله بوخهلتس حرکت کرده باشد . عجالتا" ترانسفورماتور باید تحت ولتاژ قرار گیرد تا تمام هوای باقیمانده از بین برود . همچنین در صورت نیاز میتوان آنرا زیر بار نیز قرار داد .

نتیجه :

در این مقاله سیستم خبره‌ای برای عیب‌یابی در پستهای انتقال نیرو ارائه شده است. هدف اصلی از ساختن چنین سیستمی این بوده است که دانش فنی مربوط به بهره‌برداری از پست و تجارب عملی در مورد رفع عیب تجهیزات موجود در پست را از طریق یک برنامه کامپیوتری ، بصورت کد در آورده و آن را بکار گیریم . ساختار حل مسئله بگونه‌ای است که کاملا" در چهارچوب یک سیستم خبره میگنجد . در صورتیکه روشهای برنامه نویسی عددی کارآئی چندانی در مورد آن ندارند . قابلیت توسعه سیستم این امکان را میدهد که هرگاه تجهیزات جدیدی وارد پست میشود ، براحتی اطلاعات مربوط به آن را به برنامه افزود .

قدردانی :

در اینجا لازم است ، از مساعدت‌های صمیمانه آقای مهندس ابادر میرزائی و همکاران ایشان در مرکز رلیاژ توانیر تشکر و قدردانی گردد .

منابع :

1- Wilensky R., " Common Lisp Craft , Norton & Co.,1986

2- Clocksin W.F.and Mellish C.S.," Programming in Prolog "
Springer - verlag 1981

3- Buchanan B.G .and Feigenaum E.," DENDRAL AND METADENDRAL : Their Application Dimension " Artificial Intelligence , Vol.11, 1978

4- Buchanan B.G and SHORTLIFFE E.H," Rule Based Frame System : The Mycin Experiments of the Stanford Heuristic Programming Project " Addison - Wesley , Reading , MA,1984.

5- P.Jackson -" Introduction to Expert System " Second edition, Addison Wesely Publishing Company , 1990

6- k.Matsumoto et al." Fault diagnosis of a power System Based on a description of the Structure and function of the Relay System Expert Systems ,Vol.2 , no.3 , July 1985 .

7- Chihiro Fukui,Junzo Kawakami , " An Expert System for fault section Estimation Using Information from Protective Relays , and circuit Breakers ".IEEE Trans.on power Delivery,Vol-1,no.4,oct.1986.

8- C. A. Protopapas , K.P.Psaltiras , A.V.Machias , An Expert System for substation Fault Diagnosis and Alarm Processing ",IEEE Transaction on Power Delivery Vol.6 , no.2 , April 1991

9- Kevin Tomsovic , chen - ching Liu,paul Ackerman , steve pope , "An Expert System as a Dispatchers Aid for the Isolation Pope , of line Section Fault " , IEEE Trans . On Power Delivery , Vol -2 , no.3 , July 1987 .

10- B . F, Wollenberg , Toshiaki Sakaguchi," Artificial Intelligence in Power System Operations" Proccedings Of the IEEE ,Vol.75, no.12 , Dec.1987.