



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

موضوع ارائه نوشتاری

ارائه روشی برای تخمین میزان اطمینان پذیری سیستم‌ها مبتنی بر شبکه‌های بیز

نگارش

امیرمحمد پیرحسینلو

استاد درس

دکتر رضا صفابخش

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

تقدیم به

پدر و مادر عزیز و مهربانم

که در سختی‌ها و دشواری‌های زندگی همواره یآوری دلسوز و فداکار
و پشتیبانی محکم و مطمئن برایم بوده‌اند.

امیرمحمد پیرحسینلو

دی ماه سال ۱۳۹۷

چکیده

در این پژوهش روشی برای اندازه گیری میزان اطمینان پذیری سیستم^۱ها با استفاده از شبکه های بیز^۲ و الگوریتم^۳ ک^۲ ارائه شده است. روش های پیشین در این زمینه نیازمند شبکه های از قبل ساخته شده هستند که برای ساخت این شبکه ها، نیاز به یک فرد متخصص در زمینه سیستم مورد نظر است. در روش ارائه شده، ساخت شبکه بیز به صورت خودکار به وسیله داده های از پیش تهیه شده درباره رفتار سیستم صورت می گیرد، سپس از روی شبکه به دست آمده میزان اطمینان پذیری سیستم تخمین زده می شود. از الگوریتم ک^۲ که یک الگوریتم داده کاوی و یادگیری ماشین است برای پیدا کردن روابط میان اجزای سیستم^۴ استفاده می شود. این الگوریتم شامل یک تابع امتیازدهی^۵ و یک تابع اکتشافی^۶ است و اجرای آن نیاز به نظارت انسانی ندارد.

¹ system reliability

² Bayesian network

³ K2 algorithm

⁴ system component

⁵ scoring function

⁶ heuristic function

فهرست مطالب

۱- مقدمه	۱
۱-۱ تعریف اطمینان پذیری سیستم	۱
۲-۱ روش های اندازه گیری اطمینان پذیری سیستم ها	۱
۱-۲-۱ روش سنتی	۲
۲-۲-۱ روش نوین مبتنی بر شبکه های بیز	۲
۲- ساختار شبکه های بیز	۳
۱-۲ اجزای شبکه های بیز و نحوه نمایش	۳
۲-۲ جدول احتمالات اجزای شبکه	۴
۳- ساخت خودکار شبکه ی بیز	۵
۱-۳ الگوریتم ک ۲	۵
۲-۳ روش محاسبه تابع امتیازدهی	۶
۴- تخمین میزان اطمینان پذیری سیستم	۷
۵- ارزیابی روش پیشنهادی	۷
۶- نتیجه گیری	۱۰
منابع و مراجع	۱۱

فهرست نمادها

نماد	مفهوم
$p(A)$	احتمال رخداد واقعه A
$p(H E)$	احتمال رخداد واقعه H به شرط رخداد واقعه E
X_i	گره ای در شبکه بیز که بیانگر جزء i ام سیستم است.
u	حد بالای تعداد والدهای یک جزء سیستم
n	تعداد کل اجزای یک سیستم
f	تابع امتیازدهی در الگوریتم ک ۲
π_i	مجموعه شامل والدهای کاندیدا برای جزء i ام
j	یک والد در π_i
k	یک نمونه از والد X_j
α_{ijk}	تعداد دفعاتی که X_i, X_j هر دو در حالت k قرار دارند.
q_i	تعداد کل مجموعه والدهای X_i
d_i	تعداد حالاتی که یک جزء می تواند داشته باشد (عملکرد درست معادل موفقیت و عملکرد نادرست معادل شکست)
α_{ij}	جمع α_{ijk} ها به ازای k های مختلف
$P(X_i = 1)$	احتمال درست عمل کردن جزء i ام
t	اندازه داده های مربوط به رفتار سیستم
A_m	تعداد اشتباهات الگوریتم ک ۲.
A_{FP}	تعداد روابط اشتباهی که الگوریتم ک ۲ در نظر می گیرد.
A_{FN}	روابطی که در سیستم واقعی وجود دارند اما الگوریتم ک ۲ آنها را کشف نمی کند.
A_T	تعداد کل یال ها (روابط) در شبکه بیز ساخته شده
ρ	درصد خطای الگوریتم ک ۲

۱- مقدمه

امروزه با توجه به گسترش سامانه های نرم افزاری در سراسر دنیا و همچنین پیچیده شدن این سیستم ها و رشد آنها در زمان، نیاز به روشی برای اندازه گیری میزان اطمینان پذیری این سیستم ها احساس می شود. تاکنون روش های زیادی ارائه شده اند و از تکنیک های مختلفی برای حل مساله استفاده کرده اند. در این بخش ابتدا به تعریف اطمینان پذیری سیستم ها پرداخته سپس چند تا از روش ها به همراه مثال توضیح داده خواهد شد.

۱-۱ تعریف اطمینان پذیری سیستم

میزان اطمینان پذیری یک سیستم عبارت است از:

چقدر احتمال دارد یک سیستم وظایفش را در یک دوره زمانی مشخص تحت شرایطی خاص درست انجام دهد. همانطور که از تعریف پیداست، میزان اطمینان پذیری سیستم یک عدد بین صفر تا یک است و در صورتی که صفر باشد یعنی سیستم هیچگاه درست کار نمی کند و در صورتی که یک باشد، سیستم همواره درست کار خواهد کرد.

۱-۲ روش های اندازه گیری اطمینان پذیری سیستم ها

روش های مدنظر در دو دسته جای می گیرند:

- روش های ابتدایی که اکثر مهندسان از آن استفاده می کردند،
- روش های نوین تر که بر مبنای شبکه های بیز شکل گرفته اند.

۱-۲-۱ روش سنتی

در این روش، ابتدا مهندسان سعی می کردند که روابط بین اجزای سیستم را شناسایی کرده و آنها را به وسیله مدل های گرافیکی مانند درخت خطا^۱، گراف شبکه^۲ و ... مدل کنند. سپس، با استفاده از روش هایی مانند جدول درستی^۳، مجموعه های مسیر کمینه^۴ و ... میزان اطمینان پذیری سیستم را تخمین بزنند^[۳].

۱-۲-۲ روش نوین مبتنی بر شبکه های بیز

پیش از آن که وارد این مبحث شویم، لازم است ذکر شود که حتی اگر روش سنتی دقت بالایی هم داشته باشد همچنان در آن یک مشکلی وجود دارد: این روش نیازمند آن است که قبل از استفاده، ساختار سیستم و روابط بین اجزا معلوم شده باشد که برای سیستم ها پیچیده با اجزای بسیار زیاد، اینکار دشوار و تا حدی غیر ممکن است.

برای حل این مشکل، شبکه های بیز پیشنهاد شدند. مطالعات انجام شده بر اساس شبکه های بیز نشان دادند که این شبکه ها کارایی بسیار بالایی نسبت به روش های پیشین در پیدا کردن روابط بین اجزای سیستم و ارائه یک مدل از سیستم دارند. از مزیت دیگر شبکه بیز می توان به سادگی آن اشاره کرد.

برای تخمین میزان اطمینان پذیری سیستم ها با استفاده از شبکه های بیز، اجزای سیستم و روابط بین آنها و همچنین میزان وابستگی اجزا به یکدیگر باید مشخص باشد. به همین دلیل، باید یک متخصص در زمینه سیستم مورد نظر حضور داشته باشد تا شبکه را بسازد. تعداد متخصصان محدود است و پیدا کردن آنها دشوار و هزینه بر است، از طرفی دیگر، اشتباهات غیر عمدی را که ممکن است توسط این افراد رخ دهد نباید نادیده گرفت. در کنار تمامی موارد یاد شده، سیستم های پیچیده به طور مداوم در حال تکامل هستند؛ برای مثال، بعضی از اجزای سیستم با بعضی از اجزای دیگر جایگزین می شوند و یا ممکن است دو جزء با هم ترکیب شوند و یک جزء جدید به وجود آید و ... پس بعد از هر تغییر در شبکه بیز، متخصص زمینه باید دوباره آن را اصلاح کند، در نتیجه، نیاز است که متخصص زمینه همواره در کنار سیستم حضور داشته باشد. حال راهکار حل این مشکل چیست؟

یکی از راهکارها این است که شبکه بیز به صورت خودکار ساخته شود. برای اینکار رفتار سیستم را در شرایط و زمان های مختلف ثبت می کنیم و از داده ی به دست آمده همراه الگوریتم های موجود روشی را برای ساخت خودکار شبکه به کار می بریم. رفتار سیستم در یک لحظه شامل وضعیت اجزای آن و وضعیت کل سیستم است. وضعیت کل

¹ fault tree

² network graph

³ Boolean truth table

⁴ minimal cut sets

⁵ domain expert

سیستم یعنی این که آیا کل سیستم درست کار می کند یا نه. وضعیت هر جزء هم مانند وضعیت کل سیستم تعریف می شود.

هم اکنون شبکه های زیادی برای کاربردها و سیستم های مختلف ساخته شده اند. برای مثال:

گرن و هلمینن شبکه ای برای سامانه کنترل نیروگاه های اتمی ارائه داده اند [۱].

آماساکی شبکه ای برای ارزیابی کیفیت نرم افزار ارائه داده است [۶].

در دو مثال ذکر شده ، برای ایجاد ساختار شبکه بیز و مقداردهی احتمالات اولیه نیاز به یک عامل انسانی (متخصص زمینه) است.

۲- ساختار شبکه های بیز

۲-۱ اجزای شبکه های بیز و نحوه نمایش

شبکه های بیز نوعی از گراف های غیر مدور (directed acyclic graph) هستند که در این بخش گره های گراف همان اجزای سیستم و یال های گراف، روابط بین اجزا را مشخص می کنند. یال های این گراف وزن دار هستند و وزن یال ها متناسب با میزان وابستگی دو جزئی است که در دو سر یال قرار دارند.

فرض کنید یک یال یکطرفه از جزء الف به جزء ب وجود دارد. معنای این یال این است که عملکرد جزء ب به جزء الف وابسته است. برعکس این موضوع صادق نیست زیرا یال یکطرفه است. در صورتی که یال بین الف و ب دوطرفه باشد، هر دو در عملکرد یکدیگر تاثیرگذار هستند. وجود یال جهت دار از طرف جزء الف به سمت جزء ب بیانگر این است که جزء ب به الف وابسته است؛ جزء الف را والد^۲ و جزء ب را فرزند^۳ می نامیم. نبود یال بین دو جزء بیانگر عدم وابستگی^۴ آنها به طور مستقیم است.

در شبکه های بیز یکی فرض های بنیادین این است که رابطه ی میان دو جزء از سیستم به صورت احتمالاتی با عددی در بازه صفر و یک تعیین می شود. هر چه قدر این عدد بیشتر باشد نشان دهنده ی این است که این دو جزء تاثیر بیشتری روی یکدیگر می گذارند. به عبارتی دیگر، موفقیت آمیز بودن عملکرد یکی به دیگری وابستگی بیشتری دارد.

همانطور که در فهرست نمادها در صفحه ج مشخص است، $p(A)$ بیانگر احتمال رخداد واقعه A است. در این زمینه به دلیل اینکه می خواهیم احتمال درست عمل کردن یک جزء سیستم را نمایش دهیم، از $p(A)$ به عنوان احتمال

¹ dependent

² parent

³ child

⁴ independent

درست عمل کردن جزء A از سیستم استفاده می کنیم. حال در صورتی که دو جزء E و H داشته باشیم که از H یک یال جهت دار به سمت E وجود داشته باشد، آنگاه به دلیل وابستگی E به H، احتمال موفقیت آن به صورت شرطی به شکل

$$p(E|H) \quad (۱-۳)$$

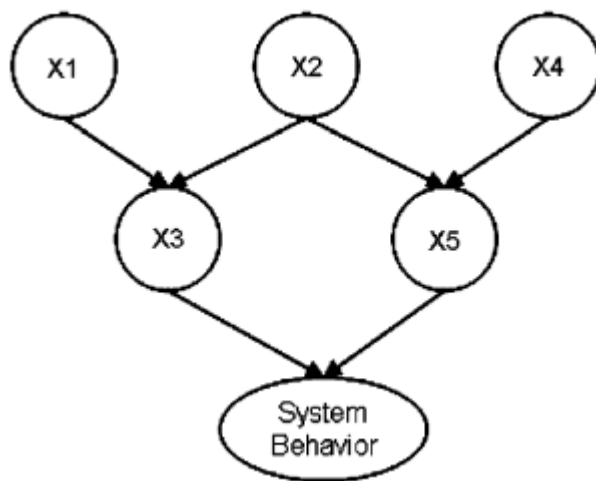
بیان می شود. اگر جزء E به اجزای دیگری نظیر F , G و ... نیز وابسته باشد، رابطه ذکر شده به شکل زیر در می آید:

$$p(E|H,G,F,...) \quad (۲-۳)$$

۲-۲ جدول احتمالات اجزای شبکه

به هر جزء از شبکه بیز یک جدول که هر سطر آن حاوی یک عدد احتمالاتی بین صفر و یک است نسبت داده می شود. از بخش قبل به یاد داریم که طبق فرمول ۲-۳، احتمال موفقیت یک جزء سیستم یا به عبارت دیگر، درست عمل کردن آن، به درست عمل کردن والد هایش وابسته است. جدول احتمالات برای یک جزء حاوی محاسبه رابطه ۲-۳ به ازای تمام حالات عملکرد والد های آن جزء است. طبق تعریف، اگر یک جزء m والد داشته باشد، آنگاه 2^m سطر در جدول احتمالات آن جزء وجود خواهد داشت. اجزایی که والد ندارند، جدول آنها تنها یک عدد دارد و این عدد را سازنده شبکه بیز تعیین می کند. همینجا می توان مشاهده کرد که اندازه جدول احتمالات به صورت نمایی رشد می کند.

در شکل زیر یک شبکه بیز را مشاهده می کنید:



شکل ۱- یک شبکه بیز ساده

¹ conditional probability table (CPT)

همانطور که در شکل یک مشاهده می کنید، اجزای $X1$ و $X2$ و $X4$ اجزای مستقل سیستم و اجزای $X3$ و $X5$ اجزای وابسته سیستم هستند. والد های $X5$ شامل مجموعه $\{X2, X4\}$ است. رفتار کل سیستم هم به صورت یک جزء نمایش داده شده است. با محاسبه جداول احتمالات از بالا به پایین و در نهایت محاسبه جدول احتمال آخرین جزء (رفتار سیستم)، می توان میزان اطمینان پذیری سیستم را تخمین زد.

۳- ساخت خودکار شبکه ی بیز

همانطور که در بخش ۱-۲-۲ گفته شد، ساخت شبکه توسط یک متخصص زمینه پر هزینه و دشوار است و بعضا اشتباهات غیر عمدی رخ می دهد. از این رو در روش پیشنهادی در این پژوهش، ابتدا شبکه بیز به صورت خودکار به وسیله الگوریتم ک ۲ و داده هایی که از قبل در مورد رفتار سیستم جمع آوری شده است ساخته می شود، سپس برای شبکه ساخته شده جداول احتمالات محاسبه و در انتها میزان اطمینان پذیری سیستم تخمین زده می شود.

۳-۱ الگوریتم ک ۲

ساخت خودکار شبکه توسط الگوریتم ک ۲ انجام می شود. ک ۲ یک الگوریتم جستجوی اکتشافی حریصانه است که برای پیدا کردن روابط میان متغیر های ورودی اش و پیدا کردن بهترین مجموعه والد برای یک متغیر، از یک تابع امتیاز دهی و یک تابع اکتشافی استفاده می کند.

- تابع امتیازدهی: الگوریتم با استفاده از این تابع به مجموعه ای از اجزا که می خواهند به عنوان والد برای یک جزء انتخاب شوند امتیاز می دهد. سپس مجموعه ای که بیشترین امتیاز را دارد به عنوان مجموعه والد برای جزء مورد نظر انتخاب می شود.
- تابع اکتشافی: الگوریتم برای آن که تمام حالت های موجود برای انتخاب والد برای یک جزء را جستجو نکند (تعداد حالات بر حسب تعداد اجزا نمایی است)، از این تابع استفاده می کند. به فرض می خواهیم مجموعه والد های جزء X_i را پیدا کنیم. الگوریتم ابتدا از مجموعه تهی شروع می کند، سپس اجزای X_{i-1}, \dots, X_1, X_0 را به ترتیب از X_0 تا X_{i-1} به مجموعه اضافه می کند و زمانی متوقف می شود که اضافه کردن یک جزء باعث افزایش امتیاز مجموعه نشود و یا تعداد والد های وارد شده به مجموعه از یک مقدار ثابتی (u) بیشتر شود. این مقدار ثابت یک حد بالا است و می توان قبل اجرای الگوریتم آن را تنظیم کرد. پس از توقف اجرا، مجموعه به دست آمده به عنوان مجموعه والد جزء مورد نظر انتخاب می شود. روند ذکر شده برای همه ی اجزای سیستم اجرا می شود.

¹ heuristic function

کارایی و دقت بالای الگوریتم ک ۲ اثبات شده است. مرتبه زمانی آن $O(n^2t)$ است که n تعداد اجزای سیستم و t اندازه داده های جمع آوری شده حاوی رفتار سیستم است. همانطور که از مرتبه زمانی آن پیداست، مرتبه زمانی بر حسب تعداد اجزای سیستم درجه دو و بر حسب اندازه ورودی درجه یک است که خیلی خیلی کمتر از مرتبه زمانی نمایی است.

بر خلاف روش های قبلی، این روش وابسته به یک سیستم خاص نیست زیرا ورودی الگوریتم ک ۲ تنها داده های قبلی در مورد رفتار سیستم است که این سیستم هر چیزی می تواند باشد.

شبه کد الگوریتم ک ۲ :

Algorithm K2 (T, u)

Input: A dataset T of historical observations on system S , an upper bound u for the number of parents

Output: A full BN B .

1. For each column i in dataset T
 - Create node X_i and add it to B .
 - $\pi_i = \phi$ for node X_i .
 - Calculate $f(i, \pi_i)$ using empty set ϕ
 - While the size of $\pi_i \leq u$
 - Let X_z be a node preceding node X_i
 - Calculate $f(i, \pi_i \cup \{X_z\})$ using X_z
 - If the new score is better than the previous score
 - Add X_z to π_i permanently
2. Return B

۲-۳ روش محاسبه تابع امتیازدهی

تابع امتیازدهی که در الگوریتم ک ۲ استفاده شده است به شرح زیر است:

$$f(i, \pi_i) = \prod_{j=1}^{q_i} \frac{(d_i - 1)!}{(\alpha_{ij} + d_i - 1)!} \prod_{k=1}^{d_i} \alpha_{ijk}!$$

(۳-۳)

از تابع رابطه ۳-۳ برای امتیازدهی به هر مجموعه والد کاندیدا برای هر کدام از اجزای سیستم استفاده می شود. این تابع به ازای هر X_j در π_i ، تعداد حالاتی را که هر دوی X_i و X_j در یک وضعیت هستند محاسبه می کند که با علامت α_{ijk} نمایش داده می شود. در این جا چون یک سامانه را مورد بررسی قرار می دهیم و این سامانه یا درست کار می کند و یا عملکرد درستی ندارد، دو حالت بیشتر نداریم: موفقیت و شکست. پس d_i برابر ۲ می شود. سایر نمادهای به کار رفته در رابطه ۳-۳ در فهرست نمادها در صفحه ج توضیح داده شده اند.

۴- تخمین میزان اطمینان پذیری سیستم

پس از پیدا شدن والد‌های هر جزء توسط الگوریتم ک ۲، نوبت به محاسبه جداول احتمالات می رسد. می توان محاسبات را در کنار اجرای الگوریتم ک ۲ انجام داد و یا اینکه پس از اجرا، به صورتی جداگانه با توابع بازگشتی پیاده سازی محاسبه را انجام داد. در انتها جدول احتمالات جزء رفتار سیستم^۱ (مانند آخرین جزء در شکل ۱) به دست می آید و جمع احتمالات در آن بیانگر میزان اطمینان پذیری سیستم است.

۵- ارزیابی روش پیشنهادی

در دو بخش کارایی روش مورد بحث در این پژوهش مورد ارزیابی قرار می گیرد:

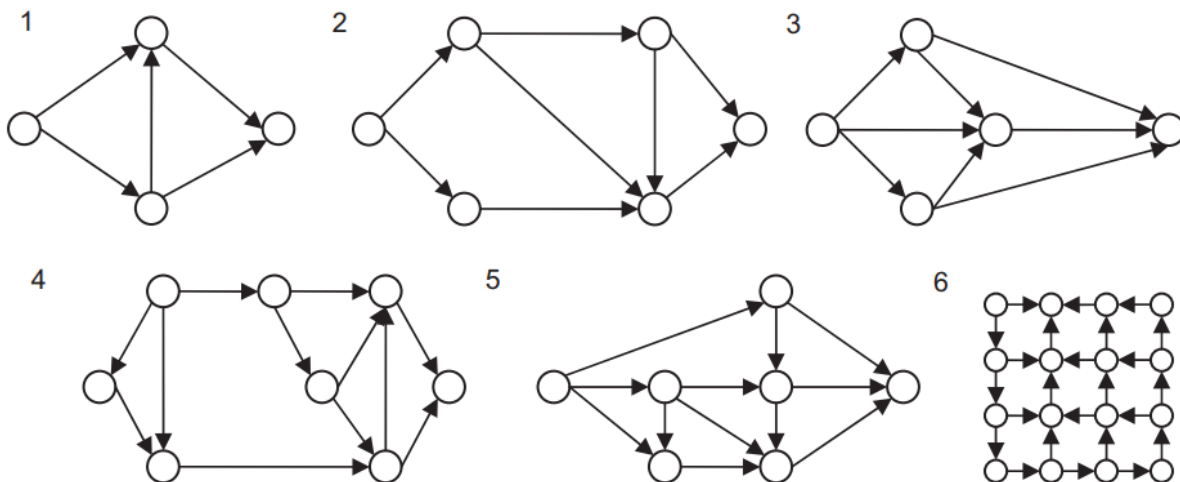
- کارایی الگوریتم ک ۲
- به دست آوردن جداول احتمال برای اجزای سیستم

در قسمت شبه کد الگوریتم ک ۲ ذکر شد که مرتبه زمانی الگوریتم برحسب تعداد اجزای سیستم چند جمله ای و برحسب اندازه داده های جمع آوری شده خطی است. دلیل چند جمله ای بودن مرتبه زمانی آن است که برای تمام اجزایی که می توانند به عنوان والد برای یک جزء انتخاب شوند، تمام داده های پیشین سیستم یکبار مورد ارزیابی قرار می گیرند و هر جزء تنها با اجزای پیشین خود (اجزای با اندیس کمتر) مورد ارزیابی قرار می گیرد.

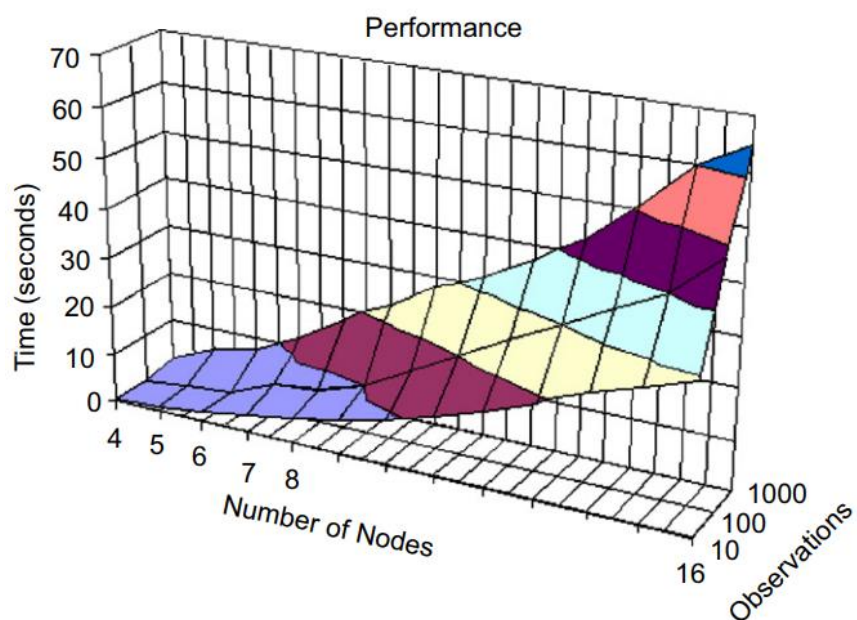
برای ارزیابی الگوریتم ک ۲ از شبکه های شکل ۲ در صفحه بعد استفاده شده است. با توجه به نمودار ۱ در صفحه بعد، زمان اجرای الگوریتم با افزایش تعداد اجزا رشد درجه دو دارد.

دقت روش مورد نظر به میزان دقیق بودن داده های ورودی الگوریتم ک ۲ و تعداد آنها بستگی دارد. هر قدر که تعداد داده ی بیشتری در اختیار الگوریتم قرار دهیم، دقت آن بالاتر می رود، در نتیجه، شبکه بیز ساخته شده تطابق بیشتری با سیستم واقعی دارد و احتمالات هم با دقت بیشتری محاسبه می شوند.

¹ system behavior component

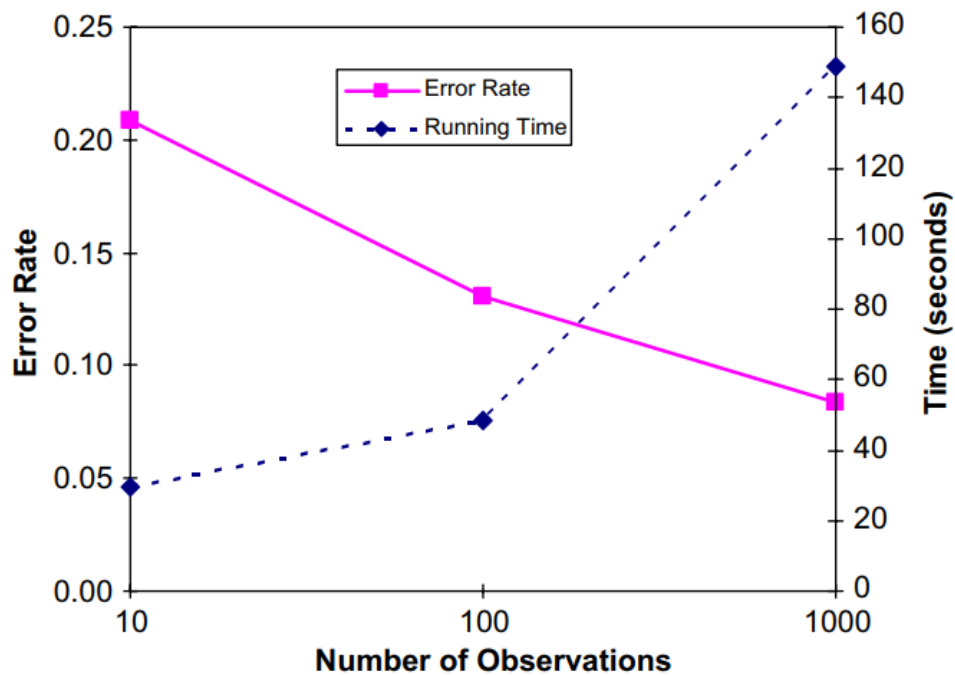


شکل ۲- شبکه هایی برای ارزیابی الگوریتم ک ۲



نمودار ۱- زمان اجرای الگوریتم ک ۲

البته نباید میزان داده ی ورودی به الگوریتم ک ۲ را بیش از حد زیاد کرد زیرا در این صورت زمان اجرای الگوریتم زیاد خواهد شد، پس باید تعادل را بین زمان اجرا و دقت مورد نیاز رعایت کرد. برای مثال در نمودار ۲ تعداد داده های ورودی صد برابر، میزان خطا نصف و زمان اجرای الگوریتم تقریباً هشت برابر شده است.

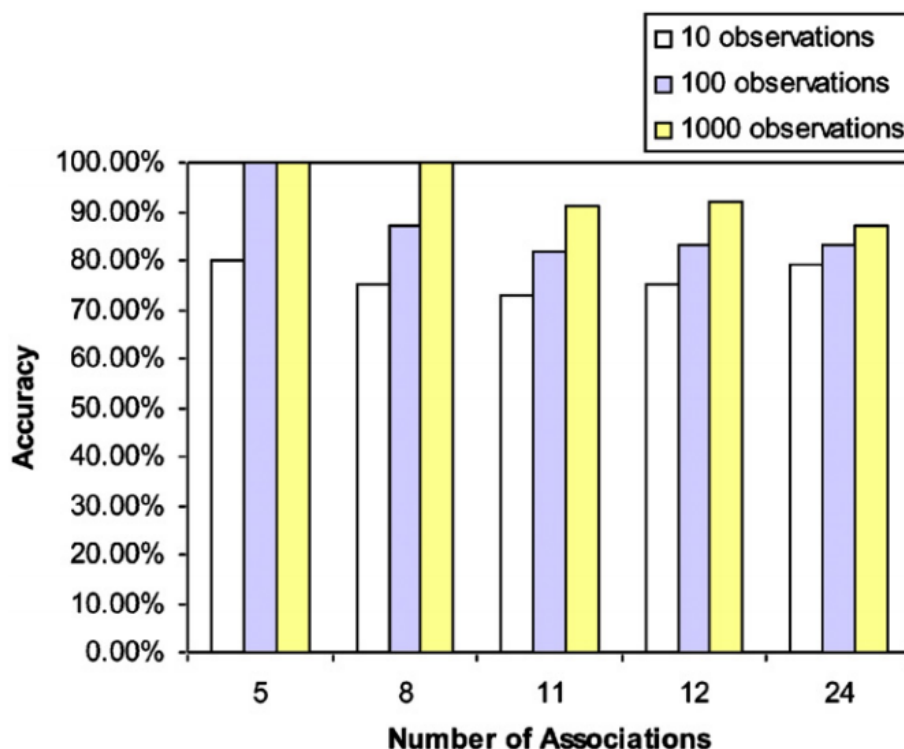


نمودار ۲- نمودار کارایی در برابر دقت الگوریتم ک ۲ برای شبکه ی شماره شش در شکل ۲

همانطور که در جدول زیر مشاهده می کنید، با افزایش تعداد داده های ورودی به الگوریتم ، شبکه بهتری به دست حاصل می شود و احتمالات واقعی تر خواهند بود. دقت الگوریتم ک ۲ با توجه به رابطه زیر به دست می آید:

$$\rho = \frac{A_M}{A_T} = \frac{A_{FP} + A_{FN}}{A_T} \quad (۱-۴)$$

در رابطه ۱-۴، A_{FP} بیانگر تعداد روابط اشتباهی است که الگوریتم ک ۲ در نظر می گیرد و A_{FN} بیانگر روابطی است که در سیستم واقعی وجود دارند اما الگوریتم ک ۲ آنها را کشف نمی کند. با توجه به رابطه ۱-۴ و شبکه های شکل ۲، میزان دقت الگوریتم ک ۲ در نمودار ۳ نمایش داده شده است.



نمودار ۳- میزان دقت الگوریتم ک ۲ در برابر تعداد مشاهده های سیستم و اجزای آن

۶- نتیجه گیری

در این پژوهش، روش های زیادی برای اندازه گیری میزان اطمینان پذیری سیستم ها مورد بررسی قرار گرفت. وجه مشترک این روش ها نیاز به نیروی انسانی (متخصص زمینه) برای ساخت شبکه یا مدل گرافیکی بود. در روش پیشنهاد شده در این پژوهش، نیاز به عامل انسانی از بین رفت ولی در عوض نیاز به داده در مورد رفتار های پیشین سیستم در شرایط مختلف است تا الگوریتم ک ۲ با استفاده از آن ها شبکه بیز سیستم را بسازد. در روش پیشنهاد شده از الگوریتم ک ۲ برای ساخت خودکار شبکه بیز استفاده شد و در ادامه از روی شبکه به دست آمده نحوه محاسبه جداول احتمال و در نهایت نحوه محاسبه میزان اطمینان پذیری سیستم توضیح داده شد. برای بهبود روش فعلی می توان از متخصص زمینه برای ساخت بخشی از شبکه که روابط بین اجزا در آن از قبل مشخص است استفاده کرد. به عبارتی نقش متخصص زمینه را تا حدی پر رنگ تر جلوه دهیم تا بتوان بخشی از سیستم را که از روابط میان اجزا در آن قسمت مطمئن هستیم استفاده کنیم تا دقت الگوریتم بالاتر رود.

- [1] Gran BA, Helminen A. "A Bayesian belief network for reliability assessment," Safecomp 2001 2001;2187:35–45.
- [2] Bishop, C. M. *PATTERN RECOGNITION AND MACHINE LEARNING*. NEWYORK, SPRINGER-VERLAG.
- [3] Coyle T, Arno RG, Hale PS. "Application of the minimal cut set reliability analysis methodology to the gold book standard network,". In: Proceedings of the industrial and commercial power systems technical conference; 2002. p. 82–93.
- [4] Doguc, O. and J. E. Ramirez-Marquez. "A generic method for estimating system reliability using Bayesian networks," Reliability Engineering & System Safety 94(2): 542-550. 2009
- [5] Langseth, H. and L. Portinale (2007). "Bayesian networks in reliability." Reliability Engineering & System Safety 92(1): 92-108.
- [6] Amasaki S, Takagi Y, Mizuno O, Kikuno T. "A Bayesian belief network for assessing the likelihood of fault content,". In: Proceedings of the 14th international symposium on software reliability engineering; 2003. p. 215–26.
- [7] Mahadevan, S., et al. (2001). "Bayesian networks for system reliability reassessment." Structural Safety 23(3): 231-251.
- [8] Russell, S. J. and P. Norvig , *Artificial intelligence : a modern approach*. Boston, Pearson.