Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 9, September 2014, pg.404 – 412

RESEARCH ARTICLE

DESIGN AND ANALYSIS OF VARIOUS APPROACHES TO DETECT INTRUSION

Sonia
Department of Computer
Science & Engineering
R.N College of
Engineering & Management,
Rohtak
sonia.hooda92@gmail.com

Vikram Nandal
Department of Computer
Science & Engineering
R.N College of Engineering &
Management,
Rohtak
vikramcse@live.com

ABSTRACT

We use computers for banking and investing to shopping and communicating with others through email or chat programs. Computer and network of computer become the very important part of companies, organization and government sector. A lot of important information is stored in computers and transferred across networks and the internet. Unauthorized users may try to break into systems to have access to private information. This brings the need of a system that can detect and prevent those harmful activities. Intrusion detection systems (IDSs) monitor networks and/or systems to detect malicious activities. That helps us to re-act and stop intruders. There are two types of IDSs, network-based IDSs and host-based IDSs. A network-based IDS monitor's network traffic and activities to find attacks, and a host-based IDS monitors activities in a computer system to detect malicious actions. This thesis is a research on using Bayesian techniques in implementing a network-based IDS that can tell us a computer process is normal (harmless) or abnormal (harmful). We combine three techniques to build a IDS. In our system k2 algorithm is applied which main purpose is to incrementally add a node to a network, it means start with a single node than add another node to complete a network. Bayesian methods utilize a search-and-source procedure to search the space of DAGs, and use the posterior density as a scoring function and finally to construct a data structure called a junction tree which can be used to calculate any query through message passing on the tree. In a past a lot of research is done on a IDS but for a wireless network is little. We have combine three techniques to make sure that network is safe from unauthorized access and attacks.

1. INTRODUCTION

1.1 Intrusion Detection Systems

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusions, defined as attempts to compromise the confidentiality, integrity, availability, or to bypass the security mechanisms of a computer network. Intrusions are caused by attackers accessing the systems from the Internet, authorized users of the systems who attempt to gain additional privileges for which they are not authorized, and authorized users who misuse the privileges give them. Intrusion Detection Systems (IDSs) are software or hardware products that automate this monitoring and analysis process.

1.2 Use of Intrusion Detection Systems:

Intrusion detection allows organizations to protect their systems from the threats that come with increasing network connectivity and reliance on information systems. Give the level and nature of modern network security threats, the question for security professionals should not be whether to use intrusion detection, but which intrusion detection features and capabilities to use. IDSs have gained acceptance as a necessary addition to every organization's security infrastructure. Despite the documented contributions intrusion detection technologies make to system security, in many organizations one must still justify the acquisition of IDSs. There are several compelling reasons to acquire and use IDSs:

1. To prevent problem behaviors by increasing the perceived risk of discovery any punishment for those who would attack or otherwise abuse the system.
2. To detect attacks and other security violations that is not prevented by other security measures.
3. To detect the deal with the preambles to attacks (commonly experienced as networks probes and other "doorknob rattling" activities).
4. To document the existing threat to an organization.
5. To act as quality control for security design and administration, especially of large and complex enterprises.
6. To provide useful information about intrusions that do take place, allowing improved diagnosis, recovery, and correction of causative factors.

2. RELATED WORK

2.1 Anomaly Detection:

Anomaly detection techniques establish a "normal activity profile" for a system; we could, in theory, flag all system states varying from the established profile by statistically significant amounts as intrusion attempts [Krister, Johansen (2003)]. However, if we consider that the set of intrusive activities only intersects the set of anomalous activities instead of being exactly the same, we find a couple of interesting possibilities:

Anomalous activities that are not intrusive are flagged as intrusive.

Intrusive activities that are not anomalous result in false negatives (events are not flagged intrusive, though they actually are).

The main issues in anomaly detection systems thus become the selection of threshold levels so that neither of the above 2 problems is reasonably magnified, and the selection of features to monitor. Anomaly detection systems are also computationally expensive because of the overhead of keeping track of, and possibly updating several system profile metrices. A block diagram of typical anomaly detection.

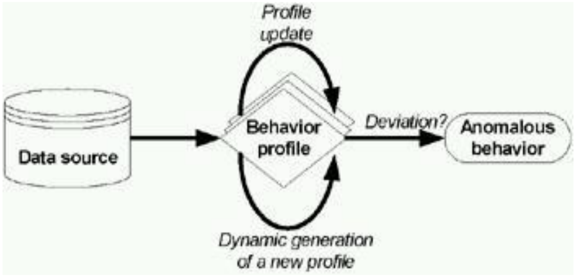


Figure 2.1 Anomaly Detection Systems

There have been a few approaches to anomaly intrusion detection systems, namely:

- STATSTICAL APPROACHES
- PREDICTIVE PATTERN RECOGNITION
- NEURAL NETWORK

2.2 Misuse Detection:

It uses a pre known signature or pattern to compare with incoming traffic. In the signature detection there are several methods to detect the intrusion patterns. The detection approaches, such as expert system, pattern recognition are grouped on the misuse. The concept behind misuse detection that these systems are not unlike virus detection systems – they can detect many or all known attack patterns, but they are of little use for as yet unknown attack methods. An interesting point to note is that anomaly detection systems try to detect the complement of "bad" behavior. Misuse detection systems try to recognize known "bad" behavior. The main issues in misuse detection systems are how to write a signature that encompasses all possible variations of the pertinent attack, and how to write signatures that do not also match non – intrusive activity. A block diagram of typical misuse detection system is given below.

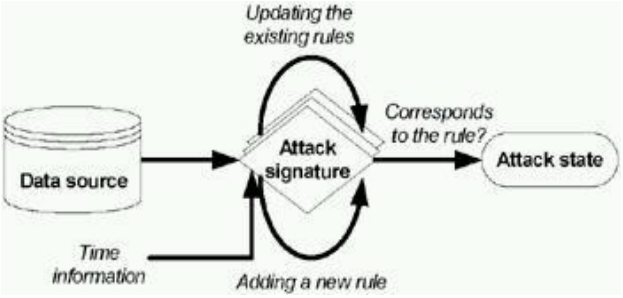


Figure 2.2 Misuse Detection system

- EXPERT SYSTEMS
- KEYSTROKE MONITORING
- MODEL BASED INTRUSION DETECTION
- NETWORK BASED INTRUSION DETECTION
- HOST BASED INTRUSION DETECTION

3. PROPOSED SCHEME

A Bayesian network is used to model a domain containing uncertainty. It is a directed acyclic graph (DAG) where each node represents a discrete random variable if Interest.

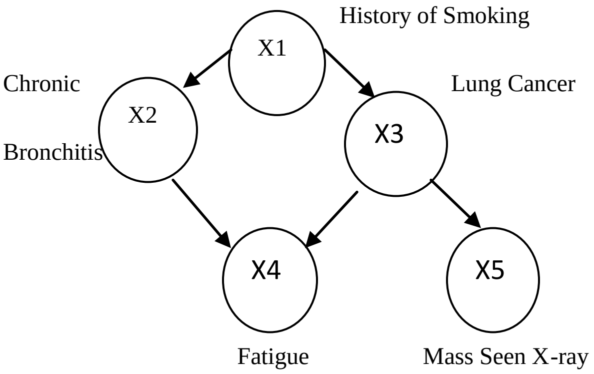


Fig 4.1 A Simple Example of Bayesian Network

Each node contains the states of the random variable that it represents and a conditional probability table (CPT). The CPT of a node contains probabilities of the node being in a specific state given the states of its parents. The parent-child relationship between nodes in a Bayesian network indicates the direction of causality between the corresponding variables. That is, the variable represented by the child node is causally dependent on the ones represented by its parents.

P(X1=no)=0.8	P(X1=yes)=0.2
P(X2=absent|X1=no)=0.95	P(X2=present|X1=no)=0.05
P(X2=absent|X1=yes)=0.75	P(X2=present|X1=yes)=0.2 5
P(X2=absent|X1=no)=0.99995	P(X3=absent|X1=no)=0.00005
P(X3=absent|X1=yes)=0.997	P(X3=absent|X1=yes)=0.003
P(X4=absent|X2=absent,	P(X4=present|X2=absent,
X3=absent)=0.95	X3=absent)=0.5
P(X4=absent|X2)=absent	(X4=oresent|X2)=absent
X3=present)=0. 5	P(X3=present)=0.5
P(X4=absent|X2)=present	P(X4=present|X2)=present
P(X3=absent)=0.9	P(X3=absent)=0.1
P(X4=absent|X2)=present	P(X4=present|X2)=present
P(X3=present)=0.25,	P(X3=present)=0.75,
P(X5=absent|X3=absent)=0.98	P(X5=present|X3=absent)=0.02
P(X5=absent|X3=present)=0.4	P(X5=present|X3=present)=0.6

Fig 4.2 The Probabilities Associated With Above Network

Related Approaches:

A. K2 Algorithm

B. Bayesian Recognition

C. Junction Tree Inference

3.1 K2 Algorithm:

In our system k2 algorithm is applied which main purpose is to incrementally add a node to a network, it means start with a single node than add another node to complete a network.

Algorithm K2 used in learning step needs:

The K2 algorithm taken from [aEH93] is included below. This algorithm heuristically searches for the most probable belief{network structure given a database of cases.

```
procedure K2;
fInput: A set of n nodes, an ordering on the nodes, an upper bound u on the
number of parents a node may have, and a database D containing m cases.g
fOutput: For each node, a printout of the parents of the node.g
for i:= 1 to n do
    _i := ;;
    Pold := f(i; _i); fThis function is computed using Equation 20.g
    OKToProceed := true;
    While OKToProceed and j_ij < u do
        let z be the node in Pred(xi) - _i that maximizes f(i; _i [ fzg);
        Pnew := f(i; _i [ fzg);
        if Pnew > Pold then
            Pold := Pnew;
            _i := _i [ fzg;
        else OKToProceed := false;
    end fwhileg;
    write('Node: ', xi, ' Parent of xi: ',_i);
end fforg;
end fK2g;
```

where:

i : set of parents of node xi

qi = j_ij

i : list of all possible instantiations of the parents of xi in database D. That is, if p1; : : : ; ps are the parents of xi then i is the Cartesian product p1

x= xp

of all the possible values of attributes p1.

ri = Vij

Vi : list of all possible values of the attribute xi

ijk : number of cases (i.e. instances) in D in which the attribute xi is instantiated with its kth value, and the parents of xi in i are instantiated with the jth instantiation in i.

ij = ri

k=ijk. That is, the number of instances in the database in which the parents of xi in i are instantiated with the jth instantiation in i.

3.2 Bayesian Recognition:

Bayesian methods utilize a search-and-score procedure to search the space of DAGs, and use the posterior density as a scoring function. There are many variations on Bayesian application [9] like greedy heuristic, combined with techniques to avoid local maxima in the posterior density (e.g., greedy search with random restarts or best first searches).Bayesian approaches are capable of dealing with incomplete records in the database [10]. The most serious drawback to the Bayesian approaches is the fact that they are relatively slow.

3.3 Junction Tree:

Junction tree inference will create a network through which we can easily determine the unauthorized users or applications. The goal of junction tree algorithm (JTA) is to define a potential representation of the graph such that, coupled with a suitable algorithm to modify these potentials, will result in the marginal of individual or groups could be obtained directly from the modified potentials.

3.4 Framework For An Adaptive Intrusion Detection System:

We propose a framework for an intrusion detection system using Bayesian network which combines k2 learning process, Bayesian Recognition and Junction Tree

Now we describe how framework starts and perform intrusion detection over a intrusion detection dataset. Working is described in six steps that are given below:

- Training data set is uploaded in our Bayesian IDS framework which contains normal signatures or connections and signature of known attacks. It is basically used to train the framework so that it can work on a testing data set.
- Testing data set is uploaded in our Bayesian IDS framework which contains thousands of computer connections. We have to detect the intrusion in this dataset.
    - K2 learning process lists the values of different features of computer connections that are present in a testing dataset.
    - Bayesian recognition uses these values for classifying the chances of attacks. It lists the output according to two classes: normal/anomaly and also with their corresponding parameter values for different features of all the computer connections present in a testing data set.
    - Junction Tree produces the final result in the form of records. It mainly uses the result of Bayesian Recognition and produce output according to the actual and predicted classes for each host with respective ID numbers.

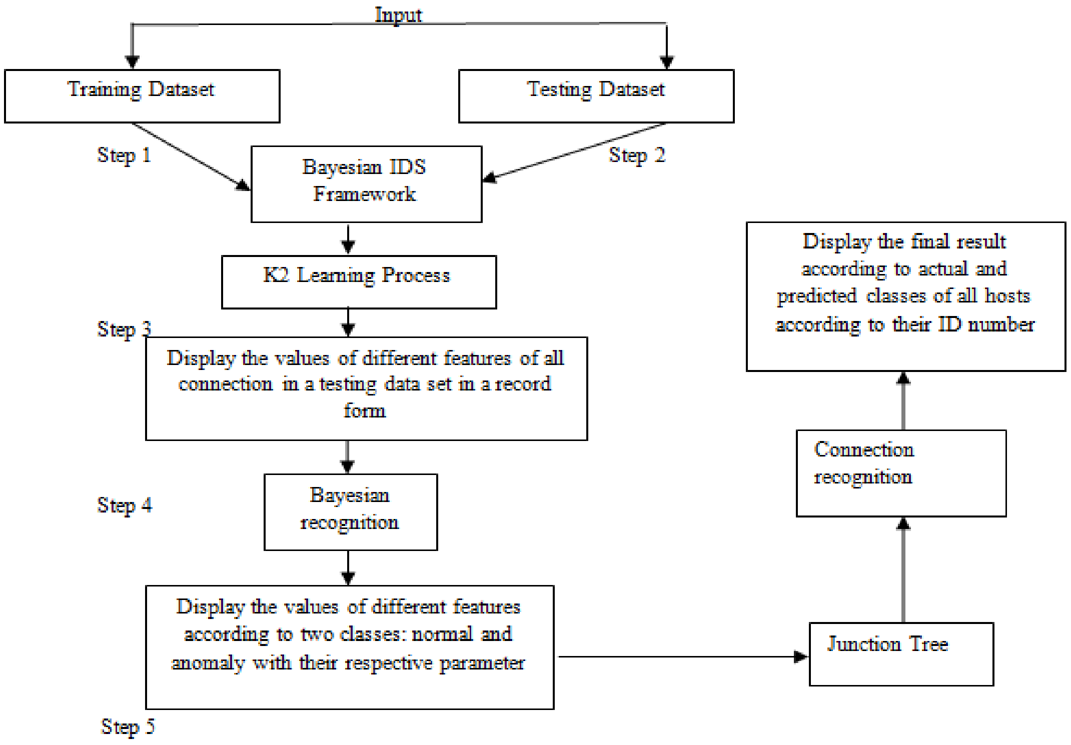

Input
Training Dataset
Testing Dataset
Step 1
Step 2
Bayesian IDS Framework
K2 Learning Process
Step 3
Display the values of different features of all connection in a testing data set in a record form
Step 4
Bayesian recognition
Display the values of different features according to two classes: normal and anomaly with their respective parameter
Step 5
Junction Tree
Connection recognition
Display the final result according to actual and predicted classes of all hosts according to their ID number

Fig 4.6 Architecture of Bayesian Intrusion Detection System

4. RESULTS AND DISCUSSION

Start intrusion detection system which contains multiple buttons.

1. Click on browse button and select the Training dataset.
2. Select testing dataset.
3. Perform k2 learning and Bayesian recognition and junction tree.

As shown below:

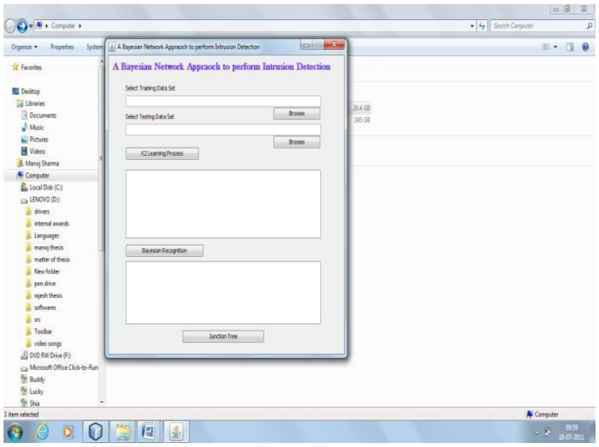
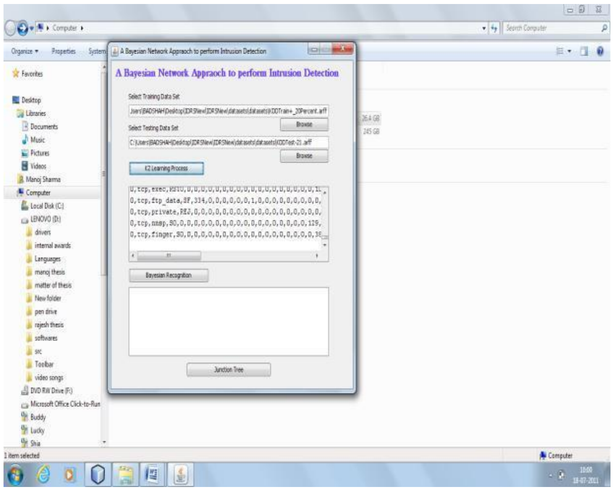

A Bayesian Network Appraoch to perform Intrusion Detection
A Bayesian Network Appraoch to perform Intrusion Detection
Select Training Data Set
Browse
Select Testing Data Set
Browse
K2 Learning Process
Bayesian Recognition
Junction Tree

Figure 1 : User Interface of Propose System

4. Now K2 learning process starts which gives the values of different features of a all connections present in a Testing Dataset as it is clearly visible on screen.


A Bayesian Network Appraoch to perform Intrusion Detection
A Bayesian Network Appraoch to perform Intrusion Detection
Browse
Browse
K2 Learning Process
Bayesian Recognition
Junction Tree

Select Training Data Set: Jsers\BADSHAH\Desktop\IDRSNew\IDRSNew\datasets\datasets\KDDTrain+_20Percent.arff

Select Testing Data Set: C:\Users\BADSHAH\Desktop\IDRSNew\IDRSNew\datasets\datasets\KDDTest-21.arff

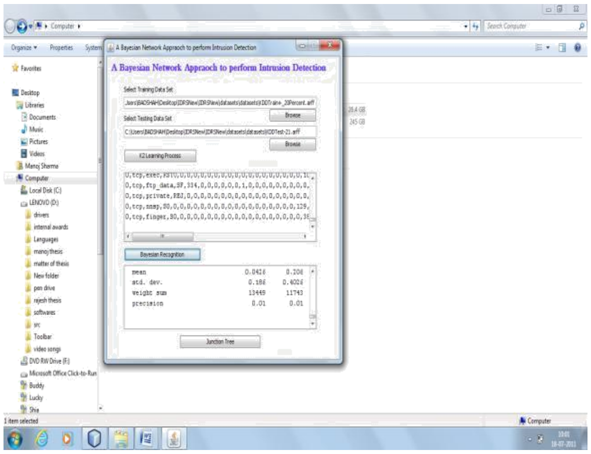
```
0,tcp,exec,RSTO,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1
0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,tcp,nnsp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,129,
0,tcp,finger,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3
```

Figure 2 : K2 Learning Process Results

5. Now system will perform Bayesian recognition which is classified in two behaviour

- Normal: indicates the system is protected.
- Anomaly : indicates something happens wrong.

6. The result of Bayesian recognition is a input to construct a junction tree.



A Bayesian Network Appraoch to perform Intrusion Detection

Select Training Data Set: .Jsers\BADSHAH\Desktop\IDRSNew\IDRSNew\datasets\datasets\KDDTrain+_20Percent.arff

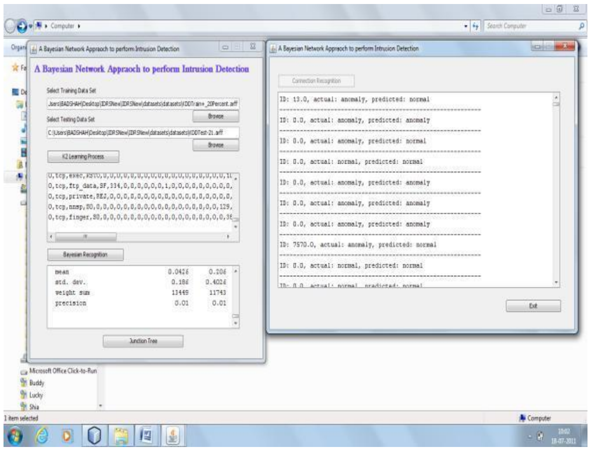
Select Testing Data Set: C:\Users\BADSHAH\Desktop\IDRSNew\IDRSNew\datasets\datasets\KDDTest-21.arff

```
0,tcp,exec,RSTO,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1
0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,tcp,nnsp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,129,
0,tcp,finger,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3
```

```
mean                          0.0426     0.206
std. dev.                      0.186    0.4026
weight sum                     13449     11743
precision                       0.01      0.01
```

Figure 3 : Bayesian Recognition Process Results

7. Junction tree produce a final result which means we can able to determine that attack is made or not.


A Bayesian Network Appraoch to perform Intrusion Detection
Browse
Browse
K2 Learning Process
Bayesian Recognition
Junction Tree
A Bayesian Network Appraoch to perform Intrusion Detection
Connection Recognition
Exit

```
ID: 13.0, actual: anomaly, predicted: normal
------------------------------------------------------------
ID: 0.0, actual: anomaly, predicted: anomaly
------------------------------------------------------------
ID: 0.0, actual: anomaly, predicted: normal
------------------------------------------------------------
ID: 0.0, actual: normal, predicted: normal
------------------------------------------------------------
ID: 0.0, actual: anomaly, predicted: anomaly
------------------------------------------------------------
ID: 0.0, actual: anomaly, predicted: anomaly
------------------------------------------------------------
ID: 0.0, actual: anomaly, predicted: anomaly
------------------------------------------------------------
ID: 7570.0, actual: anomaly, predicted: normal
------------------------------------------------------------
ID: 0.0, actual: normal, predicted: normal
------------------------------------------------------------
ID: 0.0, actual: normal, predicted: normal
```

Figure 4 : Junction Tree Process Results

It is the last module of our system which starts working by constructing a network.

5. CONCLUSION:

By combining three approaches namely K2 learning process, Bayesian recognition and junction tree inference we concluded that the performance of intrusion detection system increases as compared when these approaches are used independently. This proposed technique is more secured and less time consuming.

REFERENCES:

[1] W Arbaugh., N. Shankar, Wan Y.C.J., "Your 802.11 Wireless Network Has No Clothes", University of Maryland, Mar. 2001.

[2]. R. Kumar, Isukapalli, V. Karunya, V. Raju, "Security in Mobile Computing Systems." INTRUSION DETECTION IN WIRELESS NETWORKS.

[3]. J. Krister and L. Stephen. "Network Security: Bayesian Network."

[4]. T. F. Lunt, R. Jagannathan, "IDES: The Enhanced Prototype C a Realtime Intrusion-Detection Expert System". Technical Report SRI-CSL-88-12, SRI International, Menlo Park, CA, 1988. Intrusion Detection (BNIDS) May. 2003.

[5]. M. Esposito, C. Mazzariello, "Evaluating Pattern Recognition Techniques in Intrusion Detection Systems". The 7th International Workshop on Pattern Recognition in Information Systems, pp. 144-153, 2005.

[6]. R. Goldman, "A Stochastic Model for Intrusions." In Symposium on Recent Advances in Intrusion Detection (RAID), 2002.

[7]. D.M. Chickering, "Learning Equivalence Classes of Bayesian Network Structure", Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, Reed College, Portland, Oregon, USA, pp. 150-157, 1996.