ISWHM: Tools and Techniques for Software and System Health Management

Johann Schumann
Ole J. Mengshoel

SHM TIM, 9/2011

ISWHM for Aircraft

- ISWHM monitors the on-board software and sensors in order to:
  - Detect, in integrated (not isolated) fashion, failures in hardware components *and software*
  - perform on-board reasoning to detect most likely root cause(s)
  - annunciate failures or initiate mitigation actions
- Non-functional requirements:
  - small computational footprint
  - few false positives and false negatives
  - techniques/tools for V&V of ISWHM

ISWHM for Aircraft?!

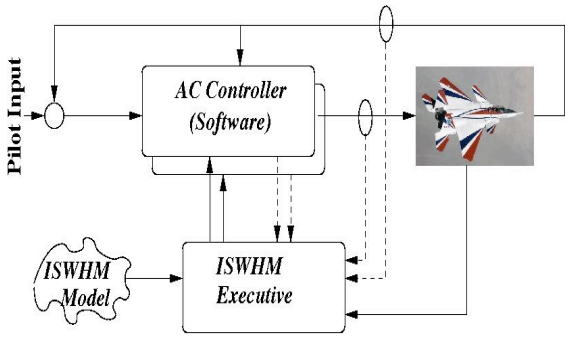
- Traditional fault management monitors AC components in isolation
- Qantas A380 engine explosion: pilots had to deal with hundreds of diagnostic message before landing
- Lack of *integration*: Several messages contradicted each other and could have led to critical situations

ISWHM for Aircraft?!

- Several predator crashes attributed to software anomalies or configuration problems
  - 2003: high pitch-up command after autopilot disengage caused “several oscillations exceeding the data link... and eventual loss of control”. A “software anomaly[..] set the pitch stick at 9deg..”
  - 2000: Pilot issued “Program mode” command that erase EEPROM data (incl. weight, link frequencies) during flight
  - 2000: Watchdog timer was “left out” of UAS software causing loss of telemetry and then loss of control
  - ...

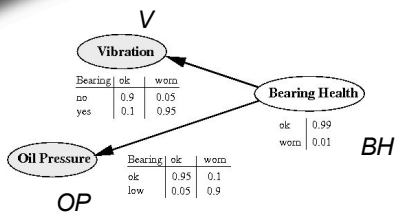
http://usaf.aib.law.af.mil

ISWHM Architecture



- ISWHM monitors
  - software
  - OS
  - hardware sensors
- ISWHM uses Bayesian networks for modeling

Bayesian Networks I

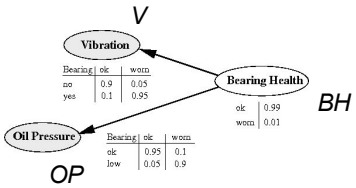


- Vibration, Oil Pressure: observable sensor variables (discrete)
- Bearing Health: unobservable health node
- Typical queries might include
  - Given sensor readings: what is the probability that the bearing is OK?
  - Which effects can a worn bearing have?

- A Bayesian network is a graphical representation of a multivariate probability distribution
- Roughly speaking:
  - Nodes = random variables
  - Edges = conditional dependencies
- Compact and modular representation of joint probability:
  - Conditional probability tables (CPTs) exponential in number of parents

Bayesian Networks II

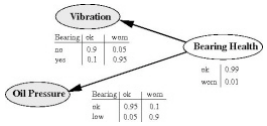
- Probability distribution of p(*BH,V,OP*)



- Lambda: presented evidence (0, 1)
- Example:
  - vibration, oil pressure, good bearing
    $\theta_{v|ok}\theta_{ok}\theta_{op|ok} = 0.1 * 0.99 * 0.95 = 0.09405$
  - no vibration, oil pressure, good bearing
    $\theta_{\sim v|ok}\theta_{ok}\theta_{op|ok} = 0.9 * 0.99 * 0.95 = 0.84645$

BH	V	OP	$p(BH,V,OP)$
ok	v	op	$\lambda_{ok}\lambda_v\lambda_{op}\theta_{v\|ok}\theta_{ok}\theta_{op\|ok}$
ok	v	~op	$\lambda_{ok}\lambda_v\lambda_{\sim op}\theta_{v\|ok}\theta_{ok}\theta_{\sim op\|ok}$
ok	~v	~op	$\lambda_{ok}\lambda_{\sim v}\lambda_{\sim op}\theta_{\sim v\|ok}\theta_{ok}\theta_{\sim op\|ok}$
ok	~v	op	$\lambda_{ok}\lambda_{\sim v}\lambda_{op}\theta_{\sim v\|ok}\theta_{ok}\theta_{op\|ok}$
~ok	v	op	$\lambda_{\sim ok}\lambda_v\lambda_{op}\theta_{v\|\sim ok}\theta_{\sim ok}\theta_{op\|\sim ok}$
~ok	~v	op	$\lambda_{\sim ok}\lambda_{\sim v}\lambda_{op}\theta_{\sim v\|\sim ok}\theta_{\sim ok}\theta_{op\|\sim ok}$
~ok	v	~op	$\lambda_{\sim ok}\lambda_v\lambda_{\sim op}\theta_{v\|\sim ok}\theta_{\sim ok}\theta_{\sim op\|\sim ok}$
~ok	~v	~op	$\lambda_{\sim ok}\lambda_{\sim v}\lambda_{\sim op}\theta_{\sim v\|\sim ok}\theta_{\sim ok}\theta_{\sim op\|\sim ok}$

Compilation to Arith. Circuit



$$
\begin{aligned}
f = \; & \lambda_{ok}\lambda_v\lambda_{op}\theta_{v|ok}\theta_{ok}\theta_{op|ok} + \\
& \lambda_{ok}\lambda_v\lambda_{\sim op}\theta_{v|ok}\theta_{ok}\theta_{\sim op|ok} + \\
& \lambda_{ok}\lambda_{\sim v}\lambda_{\sim op}\theta_{\sim v|ok}\theta_{ok}\theta_{\sim op|ok} + \\
& \lambda_{ok}\lambda_{\sim v}\lambda_{op}\theta_{\sim v|ok}\theta_{ok}\theta_{op|ok} + \\
& \lambda_{\sim ok}\lambda_v\lambda_{op}\theta_{v|\sim ok}\theta_{\sim ok}\theta_{op|\sim ok} + \\
& \lambda_{\sim ok}\lambda_{\sim v}\lambda_{op}\theta_{\sim v|\sim ok}\theta_{\sim ok}\theta_{op|\sim ok} + \\
& \lambda_{\sim ok}\lambda_v\lambda_{\sim op}\theta_{v|\sim ok}\theta_{\sim ok}\theta_{\sim op|\sim ok} + \\
& \lambda_{\sim ok}\lambda_{\sim v}\lambda_{\sim op}\theta_{\sim v|\sim ok}\theta_{\sim ok}\theta_{\sim op|\sim ok},
\end{aligned}
$$

$$f = \sum_{E} \prod_{\theta_{s|x}} \theta_{s|x} \prod_{\lambda_s} \lambda_s$$

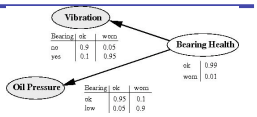
- Traditional reasoning algorithms are recursive and graph-based
- Probabilities can be expressed using polynomials *f*
- *f* can be represented as an arithmetic circuit (Kantorovich tree)
- Benefit: time/memory bounded computation of posterior distributions for all non-evidence random variables

BN for ISWHM

- A Bayesian ISWHM model
  - is a static BN (i.e., no temporal/dynamic BN)
  - uses a discretized BN
  - processes data from
    - software sensors
    - operating system/middleware
    - hardware sensors
  - posterior distributionss on health nodes provide information about their status
  - additional reasoning method for: sensitivity analysis and affected components

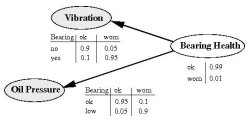
ISWHM Model

Types of Bayesian network nodes:



- command node *C*: external inputs (ground truth)
- sensor node *S*: process measurements that can be noisy, bad or unreliable
- health node *H*: reflects health of sensor or (software) component
- status node *U*: unobservable status
- behavioral node *B*: used to detect behavioral patterns (e.g., oscillation)

ISWHM Model: Edges



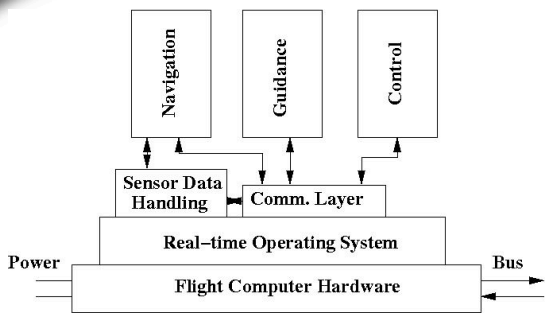
- $\{H,C\} \rightarrow U$
- $\{C\} \rightarrow U$
- $\{H,U\} \rightarrow S$
- $\{H\} \rightarrow S$
- $\{U\} \rightarrow S$

- BN edges reflect the structure of the system/software and capture substantial design knowledge.
- Fault analysis reveals a number of network patterns
- Bayesian health models can be constructed in a modularized manner

Demonstration Architecture

- For ISWHM demonstration we set up a small workstation-based aircraft GN&C model
  - F16 Simulink plant model (available from Mathworks website); C code for Simulink model + external functions have been generated using Mathwork's RTW
  - An emulated RTOS kernel, which features multiple processes, shared resources, semaphores, etc. to carry the GN&C software, the ISWHM executive, and the plant. No device drivers were emulated.
  - Simplified GN&C software with capabilities to externally inject failures
  - External logging and data analysis

Typical GN&C Architecture



Typical architecture:

- PowerPC 750, RAM, Flash,
- IObus: MIL 1553 or CAN bus (automotive)
- OS: Real-time: VxWorks, RTLinux, OSEK compliant, …

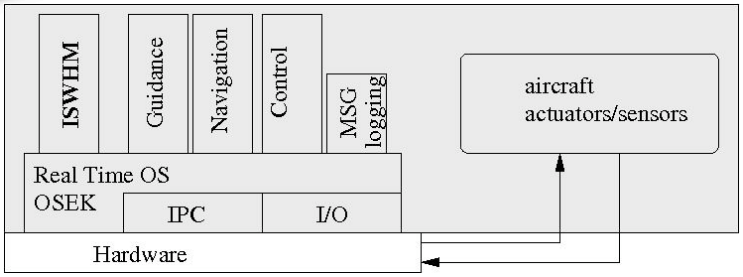
- Software components run periodically, at different frequencies:
  - G: 2Hz
  - N: 10Hz-100Hz
  - C: 100Hz
- Software components:
  - Run in different processes
  - Use comm layer for inter-process communication

OSEK

- OSEK is a standard for a real-time OS, which is heavily used in the automotive industry. We used the open source OSEK/Trampoline emulator.
- Features: multiple processes, rate-monotonic scheduling, semaphores, interrupt processing on multiple levels, CAN-bus communication, highly portable, *small* footprint
- Does not feature: dynamic processes, space separation

OSEK: Offene Systeme und deren Schnittstellen fuer die Elektronik in Kraftfahrzeugen

Demonstration Architecture



- GN&C reside in individual processes
- ISWHM executive is a separate process, which reads information about sensors, signals, OS information from shared resources
- For demonstration purposes, the plant model is implemented as a process

Sensors and Preprocessing

Software	
errors	flagged errors and exceptions
memsize	used memory
quality	signal quality
reset	filter reset (for navigation)
Software Intent	
fs_write	intent to write to FS
fork	intent to create new process(es)
malloc	intent to allocate memory
use_msg	intent to use message queues
use_sem	using semaphores
use_recursion	using recursion
Operating system	
cpu	CPU load
n_proc	number of processes
m_free	available memory
d_free	percentage of free disk space
shm	size of available shared memory
sema	information about semaphores
realtime	missed deadlines
n_intr	number of interrupts
l_msgqueue	length of message queues

- Software and Sensor signals are time series
- Signals are obtained from many sources
- We perform feature extraction (e.g., min, moving average, FFT,...) to use discrete BN

Analyzed Fault Scenarios

- For ISWHM we did a detailed analysis of 14s scenario groups, which were motivated by real-world incidents
  - Bayesian ISWHM model for scenario group
  - Required SW/sensor information
  - Experiments and results

•File system (data storage) based faults
•Inverted/crossed signals
•Bad signal ranges
•Inappropriate Signal Handling
•Resource Allocation/Blocking
•Priority Inversion
•Transient signals
•Bad/overlapping deadbands
•Component-based diagnosis
•Navigation: geometry-based fault detection
•date line crossing
•Euler angles
•Signal quality and bias
•Byzantine logic for redundant components

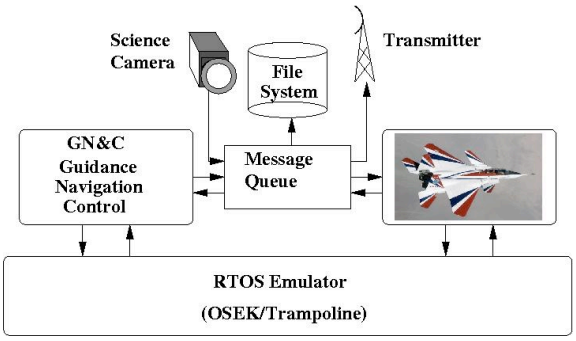
Analyzed Fault Scenarios

1. File system (data storage) based faults
2. Inverted/crossed signals
3. Bad signal ranges
4. Inappropriate Signal Handling
5. Resource Allocation/Blocking
6. Priority Inversion
7. Transient signals
8. Bad/overlapping deadbands
9. Component-based diagnosis
10. Navigation: geometry-based fault detection
11. date line crossing
12. Euler angles
13. Signal quality and bias
14. Byzantine logic for redundant components

USRA - Research Institute for Advanced Computer Science

Data storage / message queue

- Overfull data storage (file system) can cause systemwide problems (see SPIRIT problem)
- Likewise logging of messages into an almost full storage or slow (blocking) telemetry) can cause problems

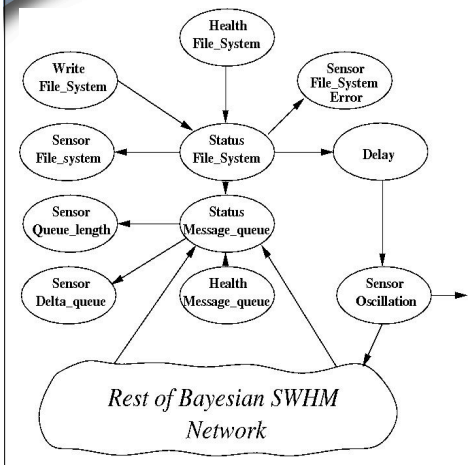


"bad" SW design: logging into file system uses a blocking write

Bad scenarios:
messages can be lost
PIO-style oscillation of the entire aircraft can occur

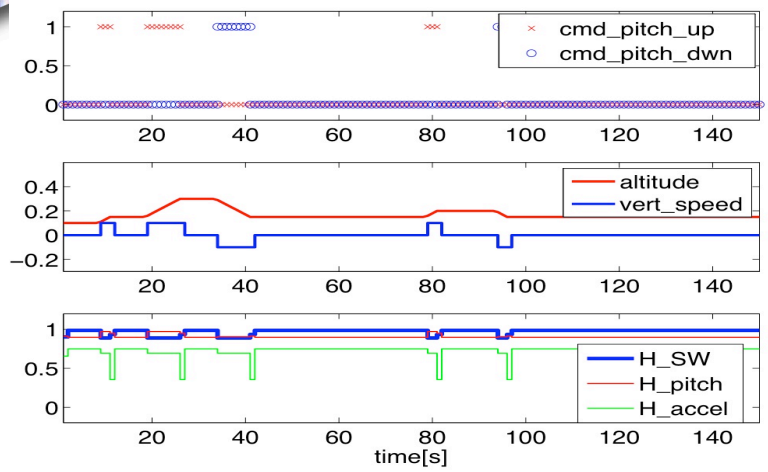
NOTE: no Error message

ISWHM model



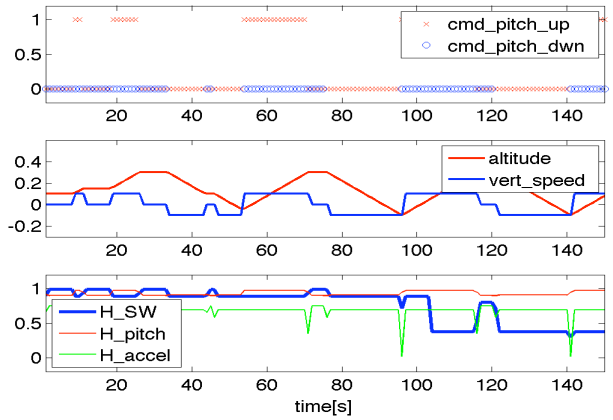
- Shown: modular extension of BN
- a systemwide "vibration / oscillation" sensor uses Fast Fourier transformation (FFT) to detect vibration and or oscillation

Experimental Results



- Nominal Case

Experimental Results



- oscillation starts at t=20s; detection at around t=100s

Bad signal handling

- Harrier: experimental Autolander caused near-crash (NASA Ames)
- cause: handling of bad radar altimeter data

```
if (RA==OK){
    last_reading = RA; return RA;
    }
else
    return last_reading;
```

- What happened?

Bad signal handling

- ISWHM takes multiple information sources into account
  - radar altimeter
  - position of stick
  - pressure altimeter
  - length and severity of error signals
- Bayesian reasoning combines these sensor readings and produces most likely explanation (in this case: do not trust radar altimeter reading)
- similar example: Mars Polar Lander

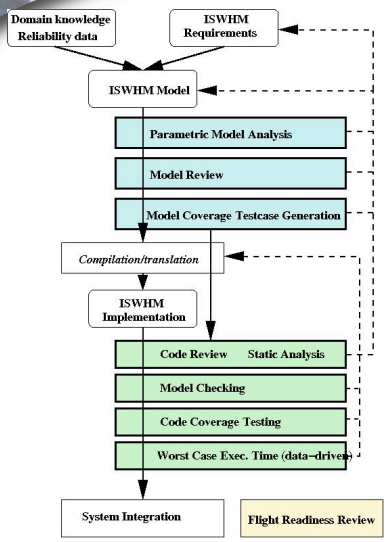
Component-based FDIR

- as seen at the Qantas A380 incident, the diagnostic messages from indivdiual subsystems, although individually correct, can contradict each other or can produce nonsense
- The modular Bayesian network can perform systemwide FDIR (for SW and subsystems) in a statistically sound manner
- Alternatively, a ISWHM model can take diagnostic warning messages as (potentially unreliable) sensor signals

Emergent Behavior

- The ISWHM presented so far is capable of detecting and diagnosing (multiple) pre-analyzed failures
- Emergent behaviors
  - have not been modeled during design
  - show up when system must work in novel environment or due to unexpected component interaction
- ISWHM cannot per se diagnose emergent behavior
- ISWHM can detect/diagnose violations of (generic) performance properties. (Assumption: emergent behavior manifests itself though some kind of performance degradation or violation of safety property
- Example: overlapping deadband in satellite controller: substantial fuel burn without getting anywhere

V&V



- V&V of ISWHM is very important
- V&V must be performed on model level *and* on code level
- V&V requires tailored tools/approaches for yet "unproven" algorithms
- most V&V tools/approaches can also be used for IVHM and FDIR using different modeling approaches (e.g., Fault trees, table-driven, rule-based)

ISWHM Applications

- Aircraft (in particular UAS): complex software, necessity for autonomous fault detection (e.g., when link is broken)
- (small) spacecraft: low footprint, single-string architecture, long signal latencies and low telemetry bandwidth makes on-board ISWHM important
- automotive industry
- monitoring of critical infrastructure (cybersecurity) requires fast processing of large amounts of distributed data in real-time
- malware detection in mobile devices
- software health monitoring in notebooks
- ...

Recent Papers

- PHM 2011 (Conference on Prognostics and Health Management, Montreal, Sept 2011) J. Schumann, T. Mbaya, O. Mengshoel "Bayesian Software Health Management for Aircraft Guidance, Navigation, and Control"
- SMC-IT 2011. A. Srivastava and J. Schumann "The Case for Software Health Management", IEEE
- SMC-IT 2011. J. Schumann and O. Mengshoel and T. Mbaya "Integrated Software and Sensor Health Management for small Spacecraft" IEEE
- Software Health Management Workshop (during SMC-IT 2011) O. Mengshoel and J. Schumann "Software Health Management with Bayesian Networks"

- related papers by team members
- E. Reed, J. Schumann, and O. Mengshoel "Verification and Validation of System health Models using Parametric testing" Infotech@aerospace
- J. Schumann, A. Bajwa, P. Berg, R. Thirumalainambi "Parametric Testing of Launch Vehicle FDDR Models", AIAA 2010
- J. Schumann, O. Mengshoel, A. Shrivastava "Who guards the Guardians? Toward V&V of Health Management Software", Runtime Verification 2010, Springer
- O. J. Mengshoel, M. Chavira, K. Cascio, S. Poll, A. Darwiche, and S. Uckun. Probabilistic model-based diagnosis: An electrical power system case study. IEEE Transactions on Systems, Man, and Cybernetics, Part A, 40(5): 874-885, 2010.

recent papers on project and related topics

Some Related Publications

- A. Choi, L. Zheng, A. Darwiche, and O. J. Mengshoel. "A Tutorial on Bayesian Networks for System Health Management'," In *Machine Learning and Knowledge Discovery for Engineering Systems Health Management*, Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, A. N. Srivastava and J. Han (Editors) , 2011.
- L. Zheng, O. J. Mengshoel, and J. Chong. "Belief Propagation by Message Passing in Junction Trees: Computing Each Message Faster Using GPU Parallelization". Proc. Uncertainty in Artificial Intelligence (UAI-11), Barcelona, Spain, July 2011.
- J. Schumann, O. J. Mengshoel, T. Mbaya. "Integrated Software and Sensor Health Management for Small Spacecraft", *Proc. Space Mission Challenges for Information Technology (SMC-IT)*, Palo Alto, CA, August 2011.
- O. J. Mengshoel, "Understanding the Scalability of Bayesian Network Inference using Clique Tree Growth Curves", *Artificial Intelligence*, 174 (2010) pp. 984–1006.
- J. Schumann, O. J. Mengshoel, A. Srivastava, and A. Darwiche. "Towards Software Health Management with Bayesian Networks", In *Proc. 2010 FSE/SDP Workshop on the Future of Software Engineering Research*, Santa Fe, NM, November 2010.
- B. W. Ricks and O. J. Mengshoel. "Diagnosing Intermittent and Persistent Faults Using Static Bayesian Networks", In *Proc. 21st International Workshop on Principles of Diagnosis (DX-10)*, Portland, OR, October 2010.
- O. J. Mengshoel, M. Chavira, K. Cascio, S. Poll, A. Darwiche, and S. Uckun, "Probabilistic Model-Based Diagnosis: An Electrical Power System Case Study." *IEEE Trans. on Systems, Man and Cybernetics, Part A*, Vol. 40, No. 5, September 2010.
- B. W. Ricks and O. J. Mengshoel. "Methods for Probabilistic Fault Diagnosis: An Electrical Power System Case Study." In *Proc. of Annual Conference of the PHM Society*, 2009.
- O.J. Mengshoel, S. Poll, and T. Kurtoglu, "Developing Large-Scale Bayesian Networks by Composition: Fault Diagnosis of Electrical Power Systems in Aircraft and Spacecraft," In *Proc. of the IJCAI-09 Workshop on Self-* and Autonomous Systems (SAS): Reasoning and Integration Challenges*, 2009.
- B. W. Ricks and O. J. Mengshoel, "The Diagnostic Challenge Competition: Probabilistic Techniques for Fault Diagnosis in Electrical Power Systems." In *Proc. of the 20th Intern. Workshop on Principles of Diagnosis (DX-09)*, Stockholm, Sweden, 2009.
- O. J. Mengshoel, "Understanding the role of noise in stochastic local search: Analysis and experiments", *Artificial Intelligence*, Volume 172, Issue 8-9, Pages 955-990, 2008.
- O. J. Mengshoel, A. Darwiche, K. Cascio, M. Chavira, S. Poll, and S. Uckun, "Diagnosing Faults in Electrical Power Systems of Spacecraft and Aircraft." In *Proc. of the Twentieth Innovative Applications of AI Conference (IAAI-08)*, Chicago, IL, 2008.
- O. J. Mengshoel, "Designing Resource-Bounded Reasoners using Bayesian Networks: System Health Monitoring and Diagnosis." In *Proc. 18th International Workshop on Principles of Diagnosis (DX-07)*, Nashville, TN, May 2007.
- O. J. Mengshoel, D. C. Wilkins, and D. Roth, "Controlled generation of hard and easy Bayesian networks: Impact on maximal clique size in tree clustering." *Artificial Intelligence* 170(16-17): pp. 1137-1174, 2006.