Dynamic Causal Hidden Markov Model Risk Assessment

Michael Sievers and Azad M. Madni

Abstract Understanding system vulnerabilities to risk factors during operation is essential for developing dependable systems. By implication, assessing in-use risk factors requires monitoring system parameters that contribute to making probabilistic inferences. We argue, however, that naïve use of statistical data without regard to causality can yield surprising and often erroneous risk predictions. Making reliable risk predictions is further complicated by lack of full awareness of system states and the existence of unobservable parameters in complex systems. Overly conservative risk assessment leads to increased life-cycle cost and reduced system availability resulting from overly aggressive preventive maintenance or replenishment strategies, while overly optimistic risk assessment can lead to even higher life-cycle cost and potential harm when otherwise preventable failures occur. This paper discusses a causality-aware, dynamic risk assessment model based on hidden Markov model construct. This model employs the concept of hidden system states that account for otherwise unexplainable observations. The model is continuously evaluated during system operation and updated when new observations warrant reevaluation.

Keywords Risk assessment · Markov model · Causality · Causal modeling · Probabilistic inference

1 Introduction

> Telling a risk analyst to 'just specify the likelihood,' is like telling a homeless person to 'just get a house' (Ferson 2005).

M. Sievers (✉) · A. M. Madni
University of Southern California, Los Angeles, CA, USA
e-mail: michael.sievers@usc.edu

A. M. Madni et al. (eds.), *Recent Trends and Advances in Model Based Systems Engineering*, https://doi.org/10.1007/978-3-030-82083-1_13

Nomenclature

a	State transition probability
β	Covariance of two system events
O	Set of observations
π	Initial state distribution
PN	Probability necessity
S	Finite set of states

Risk analyses methods generally predict events that can potentially occur and the impact of those events on system behavior. A commonly used approach by NASA and the US Department of Defense assesses and tracks risk by assigning qualitative values to the likelihood and severity of events (DoD 2014; NASA 2010). An obvious question is whether the relevant data needed for risk assessment is available, especially when the events of interest may have minimal or no prior history of occurrence (Huff 1954). Furthermore, the events selected for analysis are usually based on analyst judgment and reflect analyst's biases, not on hard evidence. When complex systems are involved, uncertainty in the models used may mask system realities, thereby resulting in questionable potentially misleading conclusions.

Several authors have noted that Bayesian modeling methods can potentially help in understanding the true nature of events and event impacts (Ferson 2005; Homayoon 2009; NASA 2010; Baru 2016). When applied appropriately, Bayesian models can accurately converge on the right parameters which influence or are indicators of risk. This is an expected outcome in that Bayesian models account for both parameter and model uncertainties. However, the proper application of the Bayesian approach needs to clearly distinguish between correlation and causation.

Briefly, Bayes' theorem, $P(A \,\&\, B) = P(A|\, B)P(B)$, has been successfully applied in multiple, disparate domains. Of course, if improperly applied, it can lead to surprising, potentially erroneous conclusions. Consider the oft cited example of ice cream sales and drowning. If the data collected considers only number of ice cream sales and number of drownings, then Bayes shows a correlation between increased ice cream sales and people drowning. That is, if A is "drowning" and B is "ice cream sales," then as the priori, $P(B)$, and likelihood, $P(A|\, B)$, increase so does the apparent correlation $P(A \,\&\, B)$. Obviously, this is flawed reasoning because both ice cream sales and people swimming increase in hot weather. Bayes is not at fault here; rather, it is that the "wrong" data set was used in the analysis. While finding correlations is relatively easy, understanding causality is far more difficult (Pearl 2001, Pearl 2009).

At the heart of most forms of risk assessment are so-called weak assertions of the form: if event A occurs, then event B occurs. If B occurs, then there is a higher probability that A also occurs. Weak assertions are expressed by Bayes:

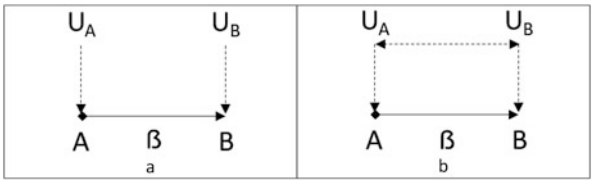

Fig. 1 Two simple models of causality in which exogenous variables U_A and U_B are connected to each other and to endogenous variables A and B with dashed lines. ß represents the direct effect A has on B

$$P\left(A|B\right) = P(A)\frac{P\left(B|A\right)}{P\left(B\right)} \tag{1}$$

and are the basis for probabilistic risk assessment (PRA) (NASA 2010).

When considering risk though, simply observing a collection of parameters without understanding causality can lead to false alarms or misinterpretation of potentially hazardous situations. While this seems reasonable, unfortunately, in complex systems, requisite observations and state knowledge for making sound decisions may not be available. Moreover, systems that degrade with time may invalidate priors that frequentists depend on or prevent proper updating of subjectivists' beliefs.

Pearl (2009) describes a causality construct that represents the probability that an event, $B{=}b$, will occur whenever action, $A{=}a$, is enforced over the entire population as $P(B = b|\, do(A = a))$. In essence, $do(A)$ implies a controlled experiment with randomized A.

Pearl shows that causality can be associated with directed graphs in which nodes represent observed or unobserved system factors connected by a term that represents the causal effect of one factor on another. The model comprises so-called exogenous variables that are not influenced by other system variables but have an impact on other system variables called endogenous variables. Figure 1 shows two simple examples based on Pearl's paper. In Fig. 1a, $Cov(A, B) = \beta$ and in Fig. 1b, $Cov(A, B) = \beta + Cov(U_A, U_B)$. Note that in some situations $Cov(U_A, U_B) = 0$ in which case the covariance is β as in Fig. 1a.

The evaluation of probabilities needed for causality needs more care than simply collecting data and looking at frequencies of occurrence. For example, Bayesian analyses are strongly influenced by the assumptions made on prior probabilities as shown in Eq. 1. As sample size increases, the sensitivity to those priors is reduced. However, in the case of causality, sensitivity to prior causal assumptions remains strong regardless of sample size. Moreover, hidden and indirect effects confound faithfully representing the relationships between events and actions. Recalling the example of eating ice cream and drowning, a naïve statistical analysis will conclude a strong correlation.

One solution for reducing the likelihood of false correlations uses Pearl's concept of probability necessity, PN. Under the assumption that event, A, is monotonic relative to action, B, then

$$PN = \frac{P(A|B) - P\left(A|B'\right)}{P(A|B)} + \frac{P\left(A|B'\right) - P\left(A|do\left(B'\right)\right)}{P(B, A)} \tag{2}$$

Equation 2 subtracts the likelihood that event, A, occurs even when action B does not. In the case of eating ice cream, the likelihood of drowning will be roughly the same regardless of ice cream consumption which eliminates confounding and incorrect bias.

2 Risk

Loosely, risk assesses the likelihood some event will occur and the impact that event has on a system or on a system's environment. Assessments run the gamut of subjective analyses by subject matter experts (SMEs) to more rigorous and formal mathematical constructs. SME risk assessments are essential during system formulation, design, and test phases because hard data are usually not available. While far from perfect, methods have been created that help mitigate the impact of SME bias and incorrect assumptions that often underlie subjective assessments. Also, while some systems are heavily instrumented for post-deployment data collection, that data may not always be useful for evaluating the cause of a particular event or the probability that an unexpected and dangerous event is likely to occur in the near future.

Dynamic assessment of system state from post-deployment data can provide insights into design weaknesses and aid in scheduling maintenance and replenishment activities. This is not a matter of simply collecting large quantities of data and doing a statistical analysis because dependencies in complex systems can be difficult to untangle. For example, suppose there is a risk of event occurring when exogenic variable $P1 > x$ but never when endogenic variables $P2 < y$ and $P3 = true$. A correct assessment of event risk depends on knowing the correlation of $P1$ to $P2$ and $P3$. Knowing only that an event occurs based on the value of $P1$ is akin to correlating increased ice cream sales to drowning while disregarding the correlation with increased swimming and summer temperatures.

2.1 Hidden Markov Causality Risk Model

Traditionally, risk assessments are used by managers and engineers in tradespace and early design evaluations and focus attention on design changes needed for

removing or reducing the likelihood of serious, undesirable future events. While using risk assessments in the design process is essential, it is equally important to understand post-deployment system vulnerabilities. That is, systems must be monitored during operation so that risks of serious or dangerous events can be estimated. As previously noted, naïve data collection without consideration of causality will not suffice as a reliable predictor of risk. What is needed is the creation of models in which prior probabilities account for probability necessity as in Eq. 2.

A state machine construct is a natural model for evaluating system risk in which states represent a notion of vulnerability, transitions occur as the result of system events, and outputs result from system state and system events. In an ideal world, the Markov property holds, i.e., the conditional probability distribution of future states depends only on the present state and events and not on the trajectory taken to arrive at that state. When the Markov property holds, we can create an understanding of the state space and the probability distribution for future states based on straightforward statistical analysis of observations made.

In the real world, there might be ambiguity in the knowledge of system state or uncertainty that an event will have the expected result. That is, some aspects of a system may be unobservable or hidden. Hidden Markov models (HMMs) accommodate uncertainty by including hidden states associated with initially unknown observation and transition distributions. HMMs are trained during system use and over time refine distributions by evaluating how well a model predicts system behavior. In essence, continuous model updates are a Bayesian process that improves HMM priors and consequently improves the reliability of the predictions made by the model.

HMMs for real-time vulnerability assessment of network cyberattacks are not new (Årnes et al. 2005; Liu and Liu 2016). Conceptually, these assessments involve the creation of a HMM-based attack graph in which states represent a method of attack and transition probabilities reflect the difficulty or vulnerability of an attack causing an unwanted operational change. Monitors collect an observation sequence that is used for evaluating how well the HMM predicts that sequence but can also be used in determining a state probability distribution (belief state). That is, given an observation sequence, O, it is possible to predict $P(O|\, model)$ using the *forward algorithm* as well as the belief state distribution after observing the sequence, O. Additionally, the state sequence can be determined using the Viterbi algorithm which determines the most probable path the model takes as each observation is made. The state sequence is useful in understanding the events that caused the system to arrive in its current state, that is, it provides the notion of causality. Additionally, if $P(O|\, model)$ is below a threshold, then it is likely that there is a new hidden state at play or the model parameters need adjustment.

In a similar vein, a more general risk model can be created. This model comprises known states that represent system conditions, transitions, and observations associated with system conditions. The model is augmented with hidden system conditions and initially unknown transition and observation probabilities. For example, Fig. 2 shows a HMM comprising two known and two hidden system conditions. The

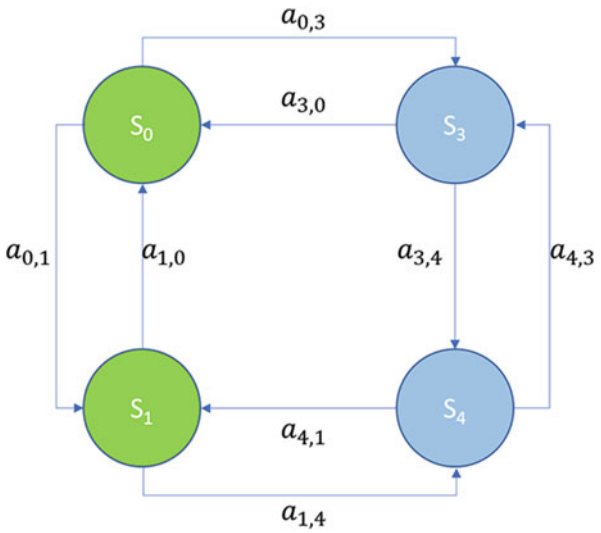

Fig. 2 A four-state Markov model comprising two observable states, S_0 and S_1, and two hidden states, S_3 and S_4

transition and observation probabilities associated with the hidden states must be nonzero but can be arbitrarily small.

A HMM is conventionally defined by:

- A finite set of states, $S = \{s_0, s_1, \ldots s_{n-1}\}$; the state at time, t, is q_t.
- A set of observations, $O = \{O_0, O_1, \ldots O_{T-1}\}$.
- State transition matrix, A, in which element $a_{i,j} = P(q_{i+1} = s_j \mid P(q_t = s_i)$.
- Observation distribution matrix, B, in which $b_j(k) = P(o_k | \, q_t = s_j)$ where $0 \leq j \leq n-1$ and
  $0 \leq k \leq T$.
- An initial state probability distribution, π, in which $\pi_j = P(q_0 = s_i)$ for $0 \leq i \leq n-1$.

2.2 *Assessing Risk*

In evaluating risk, HMM states represent hazard conditions, e.g., the condition that pressure in a tank exceeds a specified threshold. State transitions are determined by substituting the HMM parameters into Eq. 2. Equation 3 computes $a_{i,j}$ by considering whether there is a causal link between s_i and s_j if observation, o, occurs while in s_j:

$$a_{i,j} = \frac{P\left(s_j | s_i, o\right) - P\left(s_j | s_i', o\right)}{P\left(s_j | s_i\right)} + \frac{P\left(s_j | s_i', o\right) - P\left(s_j | do\left(s_i', o\right)\right)}{P\left(s_j, s_i\right)} \tag{3}$$

The HMM is developed by choosing a set of hazard conditions either randomly or through analyses such as fault tree or branch termination. Hidden states are then added and connected to the initial state set. Hidden state transition and observation probabilities are assigned nonzero, but low values so that they do not exert undue influence on model parameter initialization. However, the consequence of overly high values is that model parameter convergence could take more iterations.

The initial risk algorithm comprises five steps:

1. Determine initial values for A, B, and π; these may be set randomly if initial values are unknown.
2. Collect observations and update the initial model Baum-Welch (Baum and Petrie 1966) using Eq. 3 for evaluating transition updates.
3. Given an observation sequence, O, compute $P(O|\, model)$ using the forward algorithm, i.e., determine whether the observations match a risk scenario predicted by the model.
4. Given the state distribution determined in Step 3, use the model to predict the probability of transitioning to another risk.
5. Go back to Step 2 until $P(O|\, model)$ exceeds a threshold.

The algorithm changes once $P(O|\, model)$ is above a threshold. That is, observations are made, $P(O|\, model)$ is computed, and a risk prediction is made. $P(O|\, model)$ below a threshold is an indication of a novel condition that requires returning to Step 2. Figure 3 shows the risk algorithm flow diagram.

3 Observation Clusters

Making observations to evaluate risk in a real system is more complicated than simply collecting data from monitors. Factors such as noise, faulty monitors, and transient events can potentially create variances that need accommodation without necessarily adding to an already large state space. Moreover, some monitors may have greater influence on the state space than others in certain system operational modes. For example, a fault in an entry-descent-landing (EDL) subsystem during the early cruise phase to Mars is less important than the same fault occurring in the EDL activity.

Dealing with transient effects and certain types of noise is readily managed by requiring persistence on monitor samples. Modal information is collected as a data point in an observation. That is, rather than trust that a commanded mode has been achieved, for the purposes of risk, we depend on correlation of mode-dependent variables that represent the mode the system is actually in. Note too that variations in mode-dependent variables are also likely.

However, normal variations in samples imply the need for n-dimensional clustering in which each snapshot of monitor values is compared with a distance to observation clusters. Snapshots within a cluster are characterized by the cluster centroid. Ambiguous or unassignable snapshots are considered novel and trigger

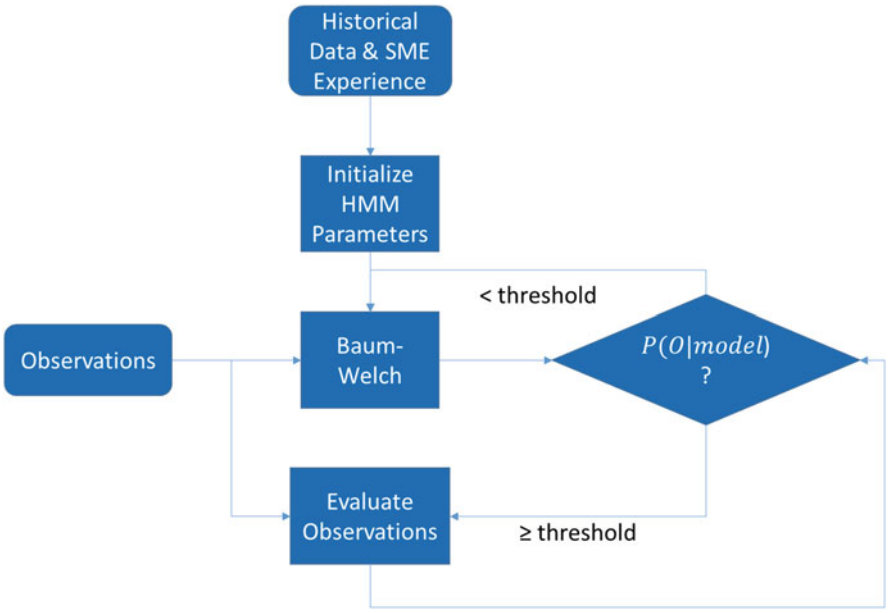

Fig. 3 Basic risk evaluation algorithm including learning iteration and observation-based risk evaluation

both a reevaluation of the cluster space and, as needed, execution of the Baum-Welsh algorithm for updating model parameters.

Not shown in Fig. 3 is the cluster step that occurs during initial observations made for updating the initial HMM parameters. Because clusters may not be known initially, clustering is performed using an expectation maximization-Gaussian mixture model (GMM) (Dempster et al. 1977). Briefly, GMM is initialized by choosing a set of clusters and randomly assigning a mean and distribution to each cluster. After initialization, the probability that each data point belongs to that cluster is computed by evaluating the proximity to the cluster centroid. The results are then used to update the clusters and repeat the probability evaluations until the probability distributions convergence.

During operation, the Mahalanobis distance from the observation to the clusters determines whether an observation belongs to a cluster. An observation is subsequently classified by the mean of the cluster it belongs to. We should note too that clusters likely will change with time, especially as the system encounters new usage, new environments, and changes. For this reason, when practical, offline GMM is periodically performed to update the cluster definitions. In this regard, it may be necessary to include heuristics for assessing the importance of certain monitor values when offline GM is not practical. Using the EDL example from above, it might be necessary to "disable" certain clusters when they no longer apply, e.g., during EDL any cluster related to cruise operation is not applicable, and any

observation that would fall into a cruise-mode cluster now falls into an observation associated with an active fault or a fault vulnerability.

4 Conclusions and Future Prospects

The prevalence of autonomous systems in automotive, aircraft, military, space, and commercial sectors is increasing making human-in-the-loop assessment of risk less and less viable. Moreover, with increasing complexity, classical diagnosis methods such as fault dictionaries that match syndromes to cause become less reliable due to false or unaccounted-for correlations. The upshot is that maintaining future systems will either become unacceptably expensive due to false alarms or, worse yet, systems will become vulnerable to serious but unpredicted risks.

In this paper, we have defined a modeling construct and assessment algorithm that, once trained, will provide a causality-aware assessment of risk. Our approach has the advantage of reducing the influence of false correlations, thereby enabling a more accurate understanding of system health. Moreover, there is a built-in learning process that adds new hidden states or adjusts model parameters when needed to explain a novel set of observations.

A distinguishing feature of our approach is that it relies less on individual observations and more on whether a sequence of observations fits a causality pattern. When a pattern is recognized, the model can provide a probability of the pattern as well as the probability of escaping to another pattern. Given both pieces of information, system operators can then decide whether and when repair or replacement is needed. For example, is it necessary to ground an airplane now due to a high probability of a near-term, serious fault condition, or can the airplane complete a mission and receive service later? Additionally, knowing with confidence failure risk simplifies maintenance scheduling and acquisition of spares.

It is well-known, however, that state-based models can be very large and difficult, if not practically impossible, to analyze. General approaches to managing large models comprise breaking them up into smaller models and/or using heuristics that approximate completely rigorous analyses. An issue we have not yet addressed is the impact on causality when decomposition or approximations are used. We understand that practical use of our approach will necessitate a thorough evaluation of state-space explosion.

Our primary work-to-go is to apply this concept to a realistic problem. To that end we have created an unmanned aerial vehicle (UAV) simulation in which we can "fly" multiple UAVs that are tasked with completing a reconnaissance mission. The simulation allows an arbitrary number of monitors and also allows injecting noise, transient upsets, and failures (Madni 2019).

To test-drive these concepts on realistic problems, we have created a minimum viable testbed (Madni 2019). The testbed employs an open-source infrastructure, multiple modeling and simulation methods, a library of components for rapid scenario development, software and hardware building blocks, and an open-source

repository. The testbed employs an open, extensible architecture, with the ability to incorporate both virtual models and physical systems. This testbed is different from traditional hardware-in-the-loop testbeds that employ proprietary models and focus on specific system instantiation. We intend to report our findings from testbed experimentation in a follow-on paper.

References

Årnes, A., K. Sallhammar, K. Haslum, T. Brekne, M.E.G. Moe, and S.J. Knapskog. 2005. Real-Time Risk Assessment with Network Sensors and Intrusion Detection Systems. In *Computational Intelligence and Security. CIS 2005*, Lecture Notes in Computer Science, ed. Y. Hao et al., vol. 3802. Berlin/Heidelberg: Springer.

Baru, S. 2016. Bayesian Network Based Dynamic Operational Risk Assessment. *Journal of Loss Prevention in the Process Industries* 41.

Baum, L., and E. Petrie. 1966. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *The Annals of Mathematical Statistics* 37 (6): 1554–1563.

Dempster, A., N. Laird, and D. Rubin. 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B* 39 (1): 1–38.

Department of Defense Risk Management Guide for Defense Acquisition Programs, 7th Edition, 2014

Ferson, S. 2005. Bayesian Methods in Risk Assessment. Unpublished Report Prepared for the Bureau de Recherches Geologiques et Minieres (BRGM), New York.

Homayoon, D., et al. 2009. Bayesian Inference for NASA Probabilistic Risk and Reliability Analysis. NASA Technical Report NASA/SP-2009-569, June 2009.

Huff, D. 1954. *How to Lie with Statistics*. New York: W.W. Norton & Company.

Liu, S., and Y. Liu. 2016. Network Security Risk Assessment Method Based on HMM and aTtack Graph Model. In *Proceedings of the 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, 517–522. New York: IEEE.

Madni, A.M. 2019. Minimum Viable MBSE Testbed for Exploring Models and Algorithms for System Resilience and Risk Assessment, SAE-TR-01/05/2020.

NASA Risk-Informed Decision-Making Handbook, NASA/SP-2010-576, 2010

Pearl, J. 2001. Causality Models, Reasoning, and Inference, Cambridge University Press, ISBN 0-521-77362-8.

———. 2009. Causal Inference in Statistics: An Overview. *Statistical Surveys* 3: 96–146.