

1995118841

HIDDEN MARKOV MODELS AND NEURAL NETWORKS FOR FAULT DETECTION IN DYNAMIC SYSTEMS

Padhraic Smyth
Jet Propulsion Laboratory, MS 238-420
California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109
pjs@galway.jpl.nasa.gov

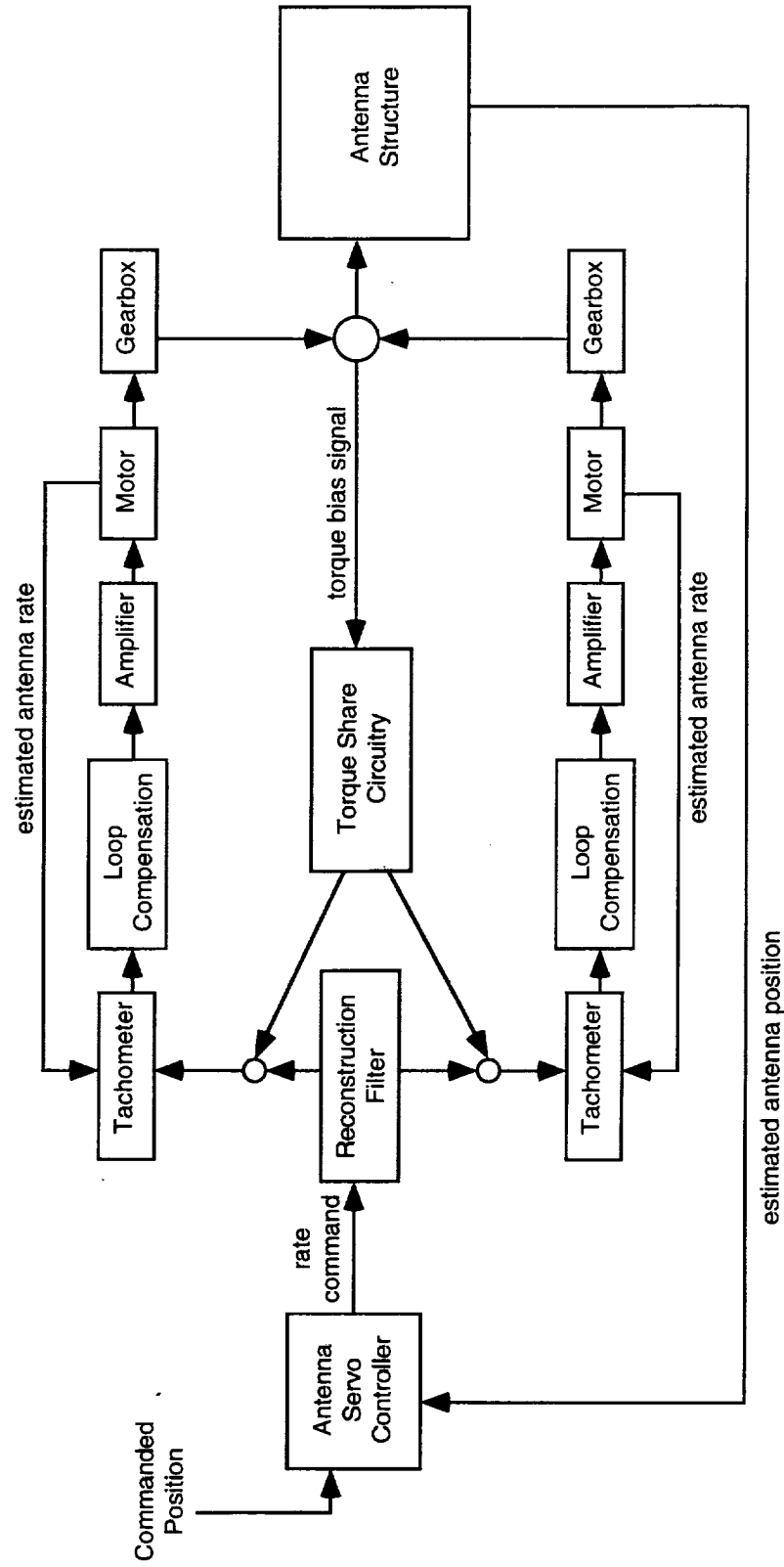
40900
p. 15

JPL Neural Network Workshop, May 11-13, 1994

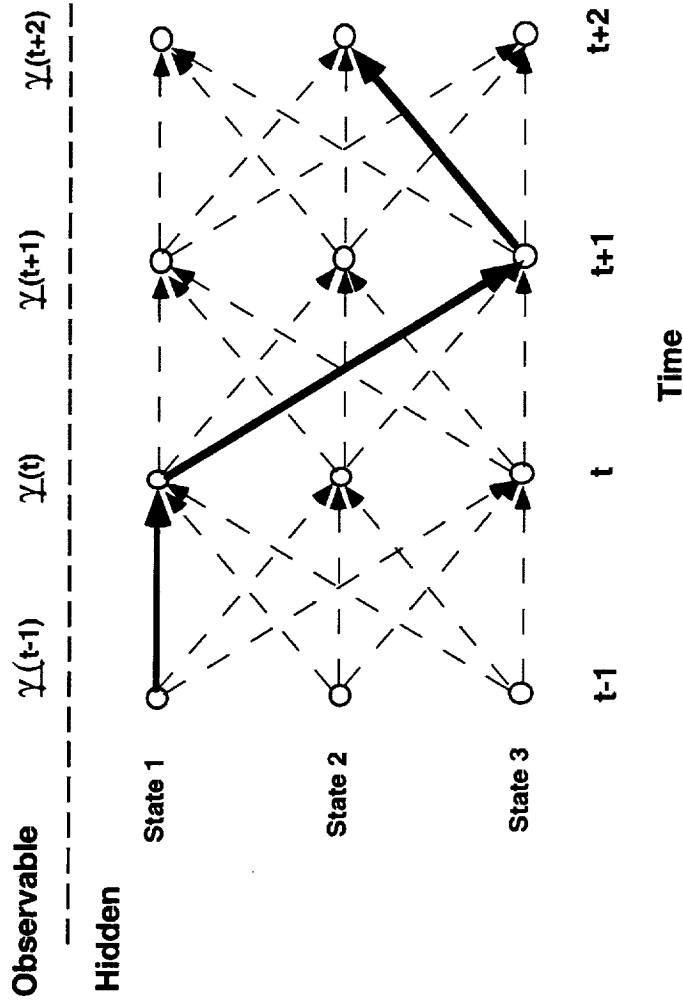
OVERVIEW

- **Neural Network + Hidden Markov models (HMMs):**
 - networks for discrimination and probability estimation
 - embedding networks in HMM's
 - application to fault detection (different from speech)
- **Application to Deep Space Network (DSN) Antenna Monitoring:**
 - on-line fault detection in large 34 meter ground antenna
 - discriminative vs. generative models for novelty detection
 - experimental evaluation
- **Conclusions and Application Status**

34 meter Beam Waveguide Antenna Pointing System



HIDDEN MARKOV MODEL BASICS



- **Explicit Assumptions (first order):**

1. Present state only depends on previous state.
2. Observables are independent over time *given* the states.

BASIC HIDDEN MARKOV EQUATIONS

Let $\Phi_t = \{\theta_t, \theta_{t-1}, \dots, \theta_0\}$.

and $\Gamma_{t-k} = \{\theta_t, \theta_{t-1}, \dots, \theta_{t-k+1}\}$.

- Probability of Current State given Past Observed Data:

$$p(\omega_j^t | \Phi_t) = \frac{1}{C_t} p(\theta_t | \omega_j^t) \sum_{i=1}^m a_{ij} p(\omega_i^{t-1} | \Phi_{t-1})$$

where

$$C_t = \sum_{j=1}^m \left[p(\theta_t | \omega_j^t) \sum_{i=1}^m a_{ij} p(\omega_i^{t-1} | \Phi_{t-1}) \right]$$

- Probability of Past State given Observed Data to Present

$$p(\omega_j^{t-k} | \Phi_t) = \frac{p(\omega_j^{t-k} | \Phi_{t-k}) p(\omega_j^{t-k} | \Gamma_{t-k})}{\sum_{i=1}^m p(\omega_i^{t-k} | \Phi_{t-k}) p(\omega_i^{t-k} | \Gamma_{t-k})}$$

NEURAL NETWORKS FOR PROBABILITY ESTIMATION

- **Theoretical Results:**
 - Theory shows that networks can approximate $p(\omega_i|\text{input features})$
 - Must use appropriate loss function: mean squared error or cross entropy
 - Results are asymptotic, assume global minimum in weight space.
- **Links with Conventional Statistics**
 - Feedforward networks can be considered a generalization of logistic regression: logistic nature of output is appropriate form for approximating posterior probabilities from exponential families.
- **Practical Consequences**
 - Practical results suggest that networks do a decent job of probability estimation.
 - Networks are better at probability estimation than competing non-parametric models (e.g., near-neighbor, decision tree methods).

HYBRID HMM/NEURAL NETWORK MODELS

- **Duality of Observed data term:**
 - Update equations are valid when terms are scaled by constants
 - By Bayes' rule can write:

$$p(\omega_j^t | \Phi_t) = \frac{1}{K_t} \frac{p(\omega_j^t | \theta^t)}{p(\omega_j)} \sum_{i=1}^m a_{ij} p(\omega_i^{t-1} | \Phi_{t-1})$$

- **Estimation of $p(\omega_j^t | \theta^t)$ terms:**
 - $p(\omega_j^t | \theta_t)$ = posterior probability of class j given inputs θ_t .
 - Train a feedforward network with MSE or CE loss functions.
 - Simple 12 input, 8 hidden units, 4 output units (normal + 3 fault conditions) feedforward network trained using conjugate gradient descent.
 - Cross-validation indicated that network size was not important.

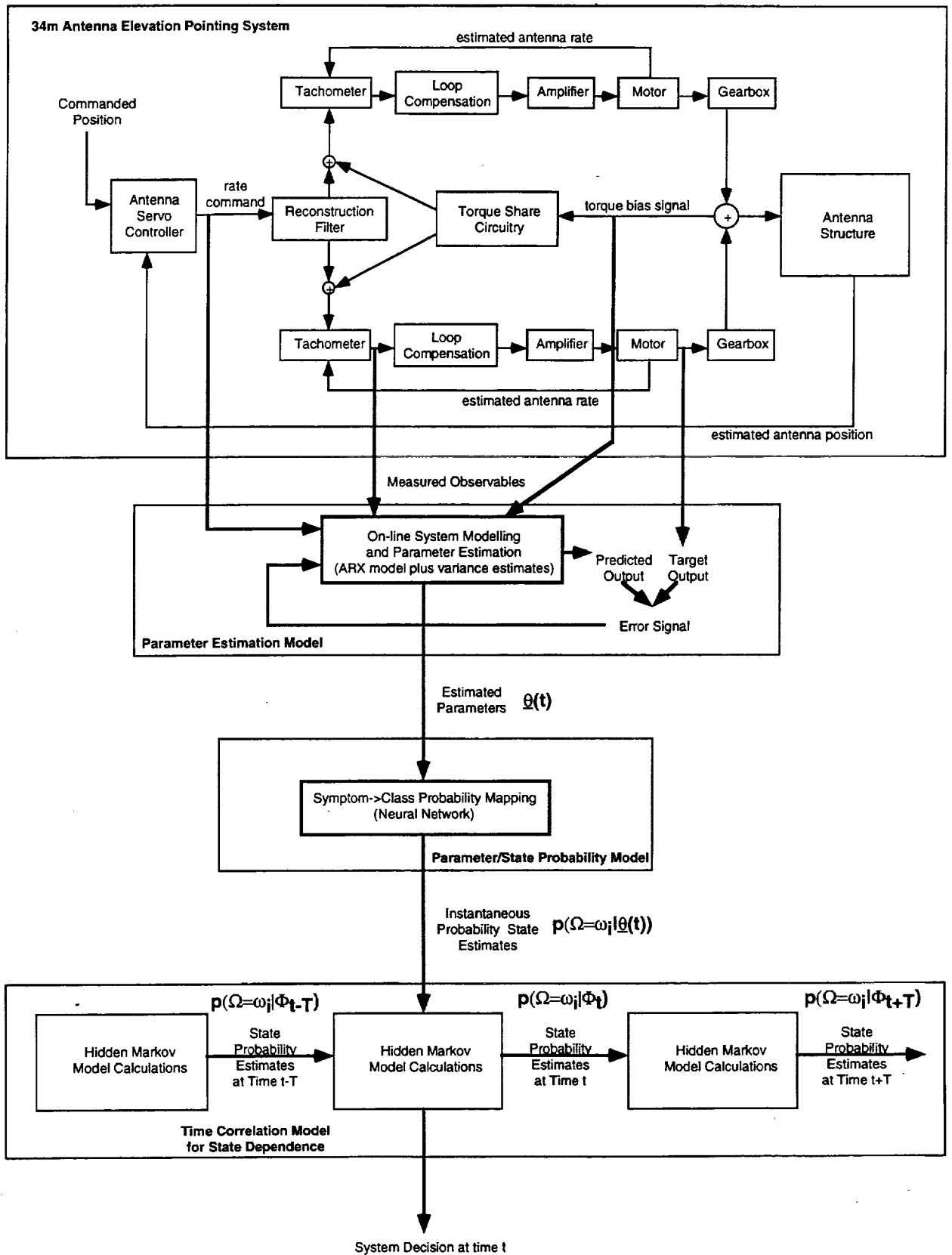
HYBRID HMM/NN FOR FAULT DETECTION

- **Key Ingredients:**
- States are known a priori, correspond to distinct physical states of system, e.g., normal, fault conditions.
- Observable-state conditional dependencies, $p(\omega_j^t | \theta_t)$ are learned by neural network from suitably generated training data.
- HMM transition probabilities are a function of system MTBF and other long term characteristics:

$$a_{11} = 1 - \frac{\tau}{\text{MTBF}}$$

- Only a single model is used: purpose is to infer “hidden” state sequence, i.e.,

estimate $p(\omega_j^t | \Phi_t)$



**SUMMARY OF EXPERIMENTAL RESULTS OBTAINED AT DSS-13 34M ANTENNA
IN REAL-TIME UNDER NORMAL AND FAULT CONDITIONS**

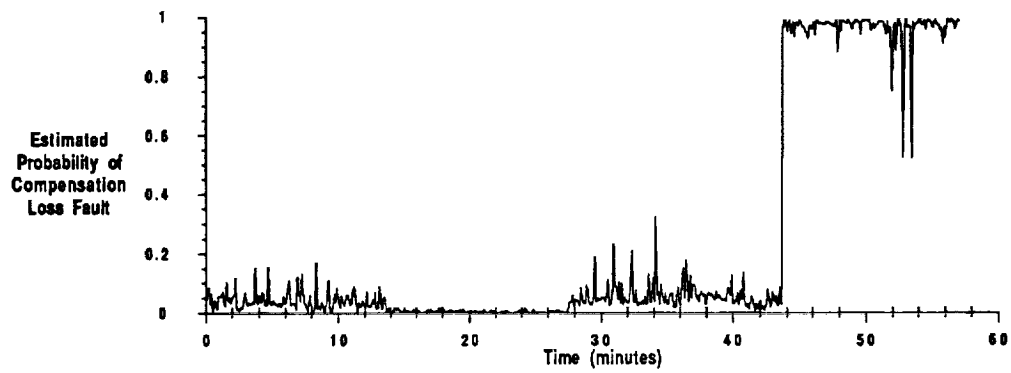
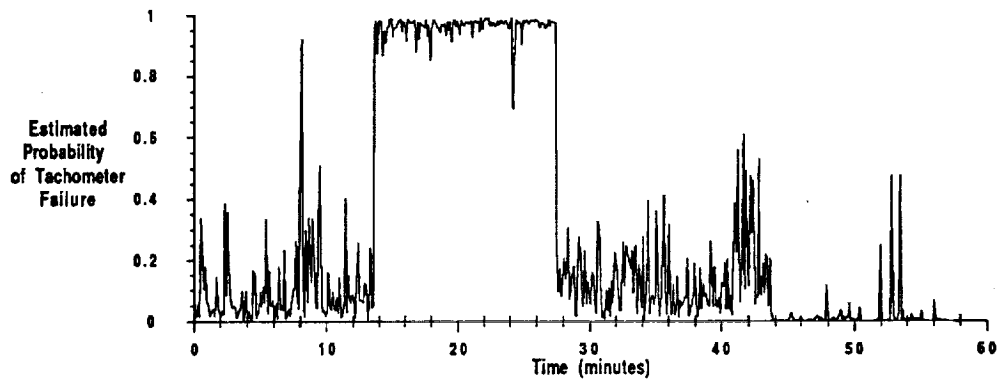
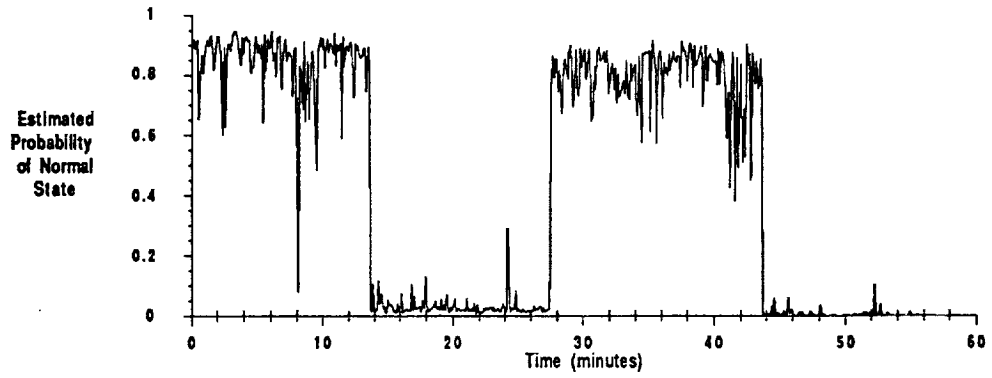
Class	Without Markov model		With Markov model	
	Gaussian	Neural	Gaussian	Neural
Normal Conditions	0.36	1.72	0.36	0.00
Tachometer Failure	27.78	0.00	2.38	0.00
Compensation Loss	34.21	0.00	43.16	0.00
All Classes	16.92	0.84	14.42	0.00

Percentage misclassification rates for Gaussian and neural models
both with and without Markov component.

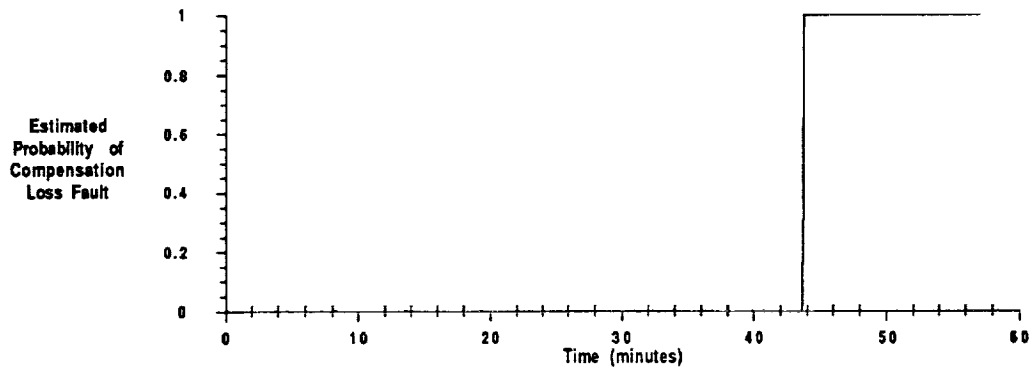
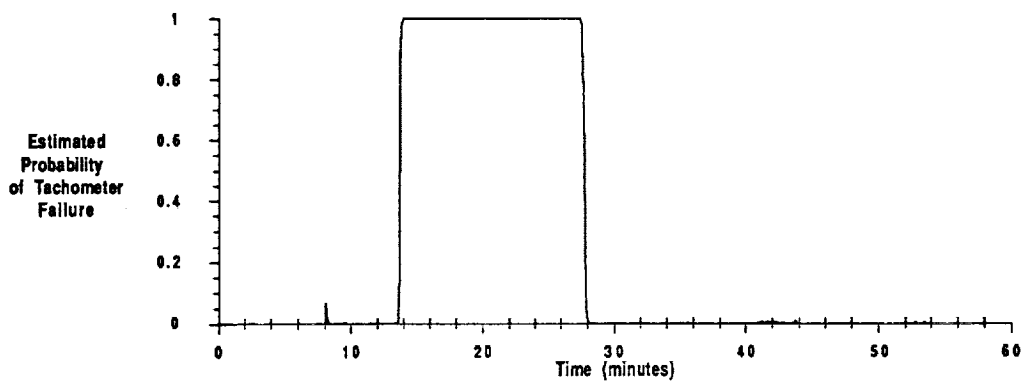
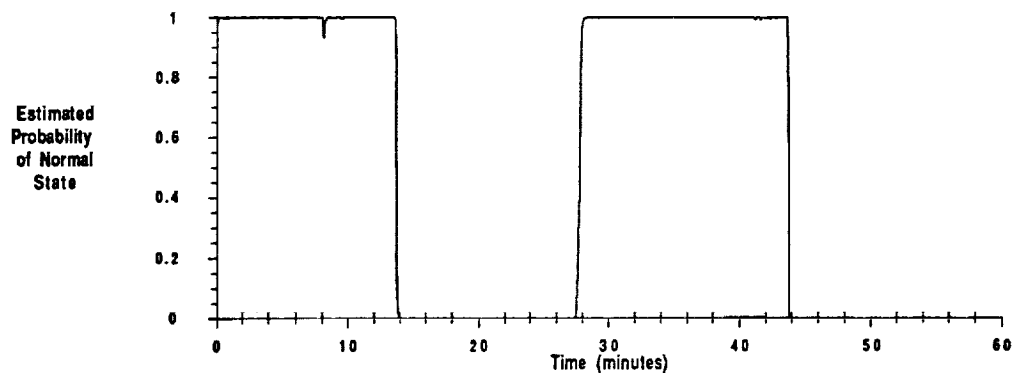
Class	Without Markov model		With Markov model	
	Gaussian	Neural	Gaussian	Neural
Normal Conditions	-2.44	-1.97	-2.46	-4.24
Tachometer Failure	-0.40	-3.52	-0.42	-4.22
Compensation Loss	-0.82	-3.48	-1.39	-4.71
All Classes	-0.87	-2.29	-1.02	-4.34

Logarithm of Mean Squared Error for Gaussian and neural models
both with and without Markov component (more negative is better).

Without hidden Markov model



With hidden Markov model



DETECTING NOVEL STATES

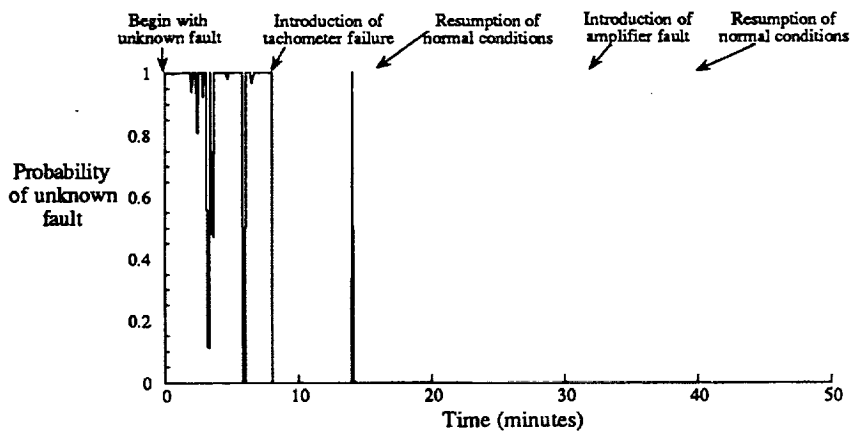
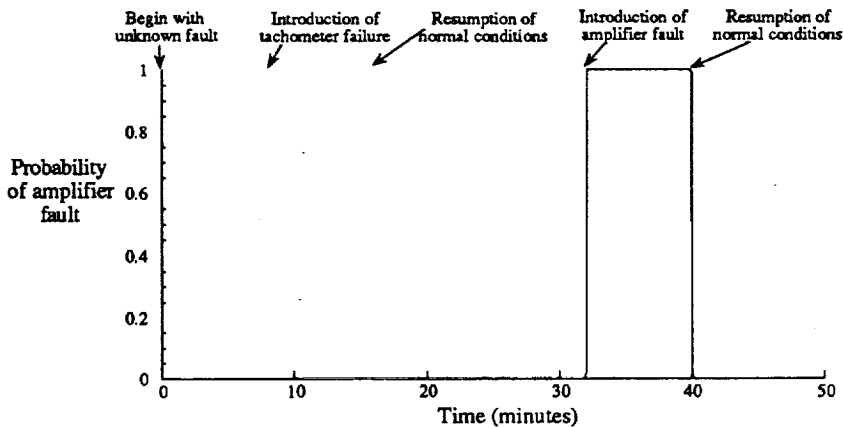
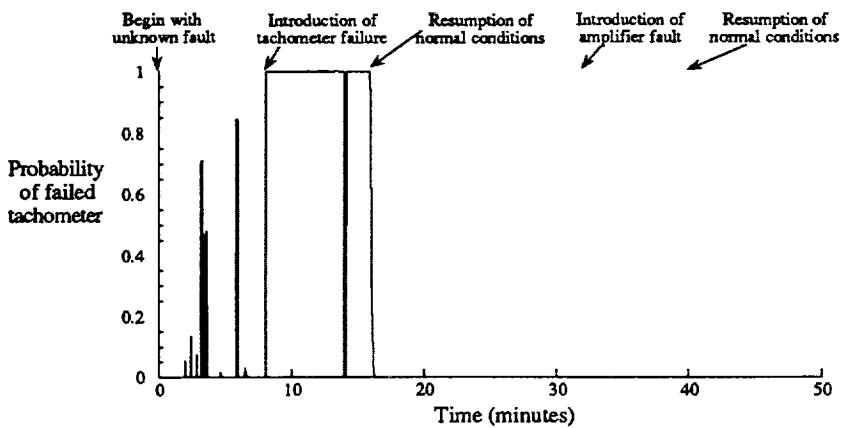
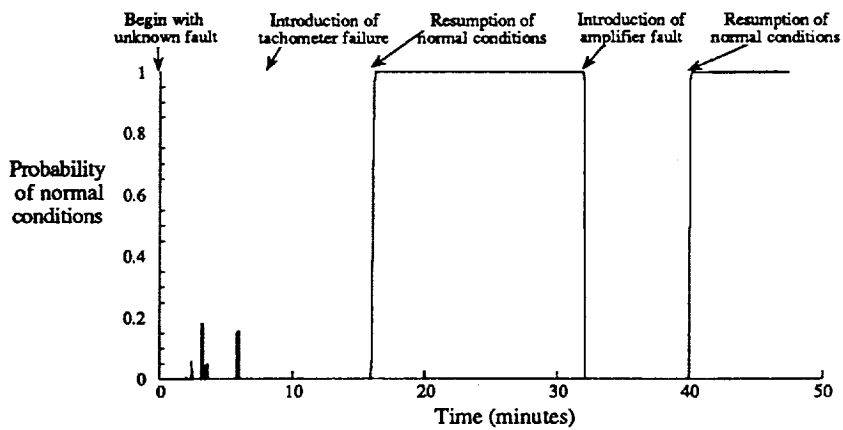
- **Basic Problem**
 - In fault detection, it is highly likely that the set of known faults is *not exhaustive*.
- **Solution**
 - Let ω_{m+1} be the “novel” state
 - Let $p_d(\omega_i|x, \omega_1, \dots, \omega_m)$ be the discriminative probabilities among the m known states
 - If we can define, $p(x|\omega_1, \dots, \omega_m)$, $p(x|\omega_{m+1})$, and $p(\omega_{m+1})$, then

$$p(\omega_i|x) = p_d(\omega_i|x, \omega_1, \dots, \omega_m)p(\omega_1, \dots, \omega_m|x)$$

and

$$p(\omega_{m+1}|x) = 1 - p(\omega_1, \dots, \omega_m|x)$$

- $p(x|\omega_{m+1})$ is determined a priori, e.g., a non-informative prior density.



CONCLUSION

- **Summary**
 - Neural networks plus HMMs can provide excellent detection and false alarm rate performance in fault detection applications
 - Modified models allow for novelty detection
- **Key Contribution of Neural Network Model:**
 - Excellent non-parametric discrimination capability
 - A good estimator of posterior state probabilities, even in high-dimensions, thus, can be embedded within overall probabilistic model (HMM).
 - Simple to implement compared to other non-parametric models.
- **Application Status:**
 - NN/HMM monitoring model is currently being integrated with the new DSN antenna controller software: will be on-line monitoring a new DSN 34m antenna (DSS-24) by July.

