A BREATHING CIRCUIT ALARM SYSTEM BASED ON NEURAL NETWORKS

Joseph A. Orr, PhD, and
Dwayne R. Westenskow, PhD

Orr JA, Westenskow DR. A breathing circuit alarm system based on neural networks.

J Clin Monit 1994;10:101–109

ABSTRACT. Objective. The objectives of our study were (1) to implement intelligent respiratory alarms with a neural network; and (2) to increase alarm specificity and decrease false-alarm rates compared with current alarms. **Methods.** We trained a neural network to recognize 13 faults in an anesthesia breathing circuit. The system extracted 30 breath-to-breath features from the airway CO_2, flow, and pressure signals. We created training data for the network by introducing 13 faults repeatedly in 5 dogs (616 total faults). We used the data to train the neural network using the backward error propagation algorithm. **Results.** In animals, the trained network reported the alarms correctly for 95.0% of the faults when tested during controlled ventilation, and for 86.9% of the faults during spontaneous breathing. When tested in the operating room, the system found and correctly reported 54 of 57 faults that occurred during 43.6 hr of use. The alarm system produced a total of 74 false alarms during 43.6 hr of monitoring. **Conclusion.** Neural networks may be useful in creating intelligent anesthesia alarm systems.

KEY WORDS. Equipment: alarms; circuits. Monitoring: practical alarms.

ABSTRAKT. Ziel. Die Ziele unserer Studie waren (1) die Implementation intelligenter Alarme für die Beatmungsüberwachung mit Hilfe neuronaler Netze und (2) die Verbesserung der Alarm-Spezifität sowie die Reduktion der Zahl der Fehl-Alarme im Vergleich zu derzeitigen Alarmsystemen. **Methoden.** Wir trainierten ein neuronales Netzwerk zur Erkennung von 13 verschiedenen Fehlertypen in Anästhesie-Beatmungskreisläufen. Das System ermittelte 30 Merkmale aus Messungen der CO2-Konzentration, des Flows und des Druckes im Atemweg. Wir erzeugten Trainings-Datensätze für das neuronale Netz durch wiederholte Induktion von Fehlern der 13 Typen bei 5 Experimenten an Hunden (616 Fehlersituationen insgesamt). Die Datensätze wurden zum Training des Netzwerkes nach der Methode der Fehlerrückübermittlung genutzt. **Ergebnisse.** In Tierexperimenten erzeugte das trainierte Netzwerk korrekte Alarme in 95% der Fälle bei kontrollierter Beatmung und in 86.8% bei Spontanatmung. Im Operationsaal detektierte das System 54 der 57 innerhalb von 43.6 Betriebsstunden aufgetretenen Fehler Das Alarmsystem erzeugte in dieser Zeit insgesamt 74 Fehlalarme. **Schlußfolgerung.** Die Ergebnisse weisen darauf hin, daß Neuronale Netzwerke bei der Entwicklung von intelligenten Anästhesie-Alarmsystemen sinnvoll eingesetzt werden können.

RESUMEN. Objetivo. Los objetivos de nuestro estudio fueron (1) implementar alarmas respiratorias inteligentes mediante redes neuronales; y (2) aumentar la especificidad de las alarmas y disminuir la incidencia de falsas alarmas, comparando con las alarmas actualmente en uso. **Mètodos.** Entrenamos una red neural para reconocer 13 fallas en un circuito respiratorio de anestesia. El sistema extrajo 30 caracterìsticas, respiraciòn a respiraciòn, desde el CO_2 de vìa aèrea, y desde las señales de flujo y presiòn. Creamos información para entrenar la red introduciendo 13 fallas repetidamente en 5 perros (616 fallas en total). Usamos la informaciòn para entrenar la red neuronal usando el algoritmo de propagaciòn retrògrada del error. **Resultados.** En animales, la red entrenada reportò correctamente las alarmas para el 95% de las fallas durante ventilaciòn controlada, y para 86.9% de las fallas durante ventilaciòn espontànea. Al ser probada en el pabellòn quirùrgico, el sistema identificò y reportò correctamente 54 de las 57 fallas que ocurrieron durante 43.6 horas de uso. El sistema de alarma produjo un total de 74 falsas alarmas durante las 43.6 horas de monitorizaciòn. **Conclusiòn.** Las redes neuronales pueden ser ùtiles para crear sistemas de alarma inteligentes para anestesia.

Failures in the patient breathing circuit cause many preventable anesthesia accidents. Hose disconnects are the most common failure and result in apnea [1–3]. Other failures include leaks in hose connections and leaks around the endotracheal tube cuff (hypoventilation) [1,4,5], airway obstructions (barotrauma or hypoventilation) [6], and incompetent nonrebreathing valves (CO_2 breathing) [4,7]. All of these problems are harmless if detected and corrected quickly.

Neural networks should be capable of identifying these faults during anesthesia. Recent applications show that neural networks can process complex combinations of variables to classify events or objects into multiple categories (e.g., voice and image recognition). These pattern recognition capabilities should allow neural networks to process airway signals, creating specific failure classifications whenever the signals are abnormal. Neural networks should be able to provide problem-specific alarm messages, while keeping the number of false alarms low.

We developed a neural network based alarm system that automatically detects and identifies the location of 13 breathing circuit failures. In an animal study, we measured the true-positive rate with which the neural networks identified the faults. In a clinical study, we measured the system's false-positive alarm rate.

From the Department of Anesthesiology, University of Utah, Salt Lake City, UT.

Received July 13, 1992, and in revised form Sep 22, 1992. Accepted for publication July 2, 1993.

Address correspondence to Dr Orr, Department of Anesthesiology, University of Utah, 50 N Medical Dr, Salt Lake City, UT 84132

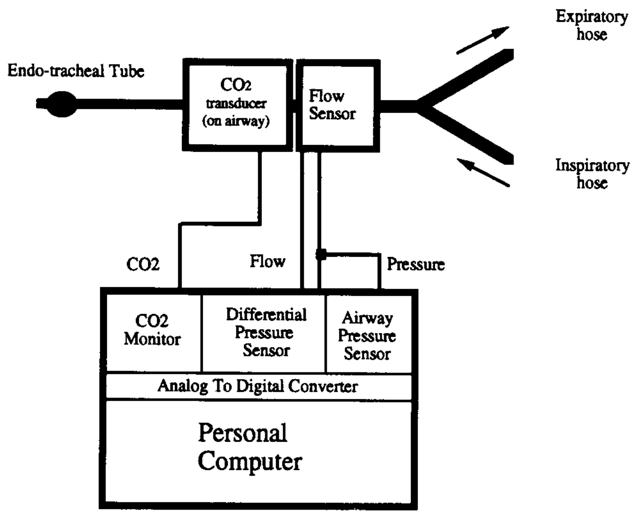

Fig 1. Location of sensors.

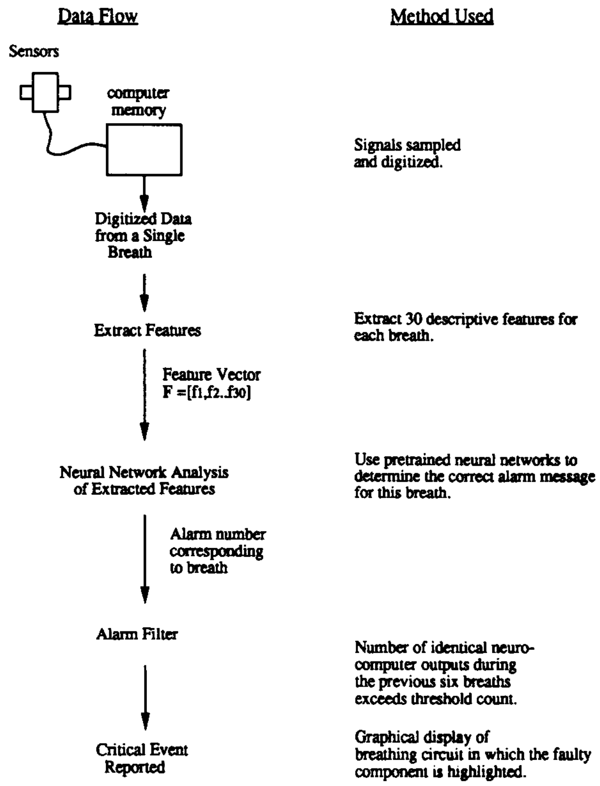

Fig 2. Data flow through the alarm-generating algorithm.

METHODS AND MATERIALS

Sensors

Figure 1 shows the neural network alarm system sensors. A mainstream monitor measures the CO_2 concentration (Novametrix, Inc., model 1260, Wallingsford, CT). A solid-state pressure transducer measures airway pressure through a 0.6-cm diameter 200-cm tube (Sensym, model SCX01DN, Sunnyvale, CA). A variable orifice pneumotach (Carlsbad Plastics, Carlsbad, CA) and differential pressure transducer (Validyne Instruments, model MP45, Northridge, CA) measure airway flow. Together these transducers added 15 ml of dead space to the airway. We auto-referenced both pressure transducers to ambient pressure every 30 min to correct for drift.

Feature Extraction

Figure 2 diagrams data flow through the alarm system computer (PC-386, Zenith Data System, St. Joseph, MI). The computer samples the output from the three sensor signals at 60 Hz with 12-bit resolution (Tecmar, Labmaster, Solon, OH). Samples are scaled and placed in a first-in–first-out buffer. The algorithm extracts the 30 features listed in Tables 1 through 3 from each breath (k) and stores them in a feature vector $\vec{F}[k]$, where k is a discrete time indicator in breaths. The computer calculates a difference vector $\vec{D}[k]$ for each breath, each of whose elements is the difference between the current value of the ith feature ($f_i[k]$), and a weighted sum of past values for the ith feature ($m_i[k]$):

$$\vec{D}[k] = \vec{M}[k] - \vec{F}[k], \tag{1}$$

where

$$\vec{M}[k] = (0.15)\vec{F}[k] + (0.85)\vec{M}[k-1]. \tag{2}$$

If an alarm condition is detected, $\vec{M}[k]$ is not updated until the condition is resolved. The system knows that an alarm has been resolved when it again recognizes normal breaths. The breath-detection algorithm forces the interval between updates to be less than 30 sec.

Neural Networks

Figure 3 shows the alarm system's five neural networks. The first network uses the feature vector $\vec{F}[k]$ to classify each breath as controlled or spontaneous. It has 30 input nodes, a layer of 30 fully interconnected hidden nodes, and two output nodes.

Neural networks 2 and 3 use the feature vector ($\vec{F}[k]$) to determine if a breath is normal or if it belongs to one of the five general alarm classes listed in Table 4.

Table 1. Features Extracted from the Capnogram

1. Phase II slope
2. Phase IV slope
3. Phase III slope
4. Phase I time
5. Phase II time
6. Phase IV time
7. Minimum partial pressure
8. Maximum partial pressure (end-tidal CO_2)
9. Time (since start of breath) that the maximum CO_2 occurs
10. Sample mean of capnogram
11. Sample variance of capnogram

Table 2. Features Extracted from the Airway Pressure Signal

1. Inspiratory pressure-volume slope
2. Expiratory pressure-volume slope
3. Pressure difference between start and end of breath
4. Maximum pressure
5. Minimum pressure
6. Pressure signal mean
7. Pressure signal variance
8. Area of pressure-volume loop

Table 3. Features Extracted from the Flow Signal

1. Inspiratory time/expiratory time
2. Inspiratory volume minus expiratory volume
3. Maximum flow during inspiration
4. Maximum flow during expiration
5. Mean inspiratory flow
6. Volume at which maximum inspiratory flow occurs
7. Volume at which maximum expiratory flow occurs
8. Respiratory rate
9. Inspired tidal volume
10. Variance of expiratory flow signal
11. Number of flow volume loops in breath

Networks 2 and 3 have 30 input nodes, 30 hidden-layer nodes, and five output nodes. These networks identify general faults that are extant when the alarm system is first activated or those faults that develop slowly over the course of many breaths.

Networks 4 and 5 use the differential feature vector ($\vec{D}[k]$) to place each breath into one of the 14 specific alarm classes listed in Table 4. Networks 4 and 5 have 30 input nodes, 40 hidden-layer nodes, and 14 output nodes. The specificity of networks 4 and 5 should be higher, because their inputs are the difference between the current features and those of the last "normal"

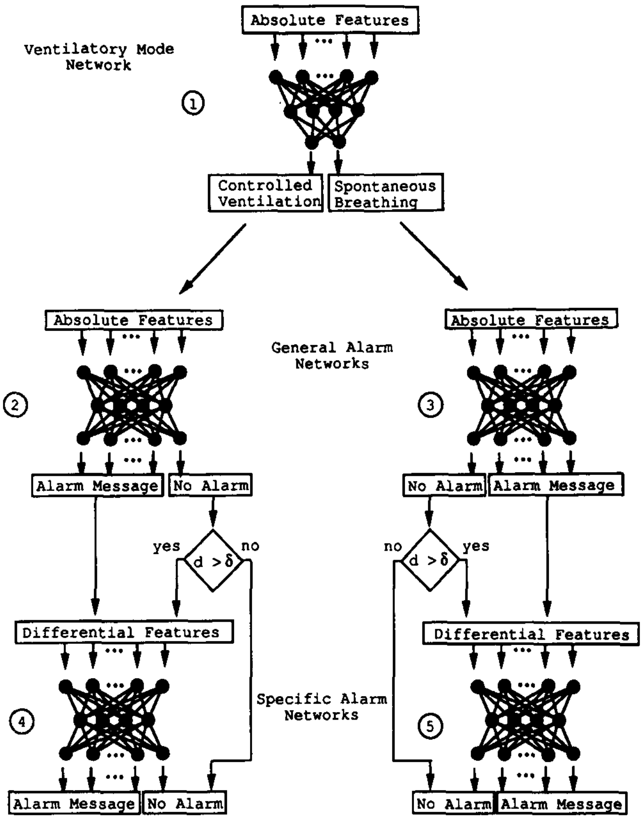

Fig 3. Neural network architecture. The first network classifies each breath according to mode of ventilation (controlled or spontaneous). Networks 2 and 3 use absolute features to classify each breath as normal or abnormal. If the breath is abnormal, one of four messages is generated. Networks 4 and 5 use differential features to generate more specific alarm messages (see Table 4).

breath. Network 4 (or 5) becomes active if network 2 (or 3) identifies an abnormal breath, or if a measure of change, $d[k]$, exceeds a threshold δ, indicating a significant change in the time trend of the features. This threshold varies according to the stability of d—that is, according to the normal breath-to-breath difference in the features.

$$d[k] = \sum_{i=1}^{30} \frac{(m_i[k] - f_i[k])^2}{v_i[k-1]}, \tag{3}$$

where

$$v_i[k] = 0.1(f_i[k] - m_i[k])^2 + 0.9v_i[k-1] \tag{4}$$

$$\delta = m_d + 0.1v_d, \tag{5}$$

where

$$m_d[k] = 0.1d[k] + 0.9m_d[k-1], \tag{6}$$

Table 4. General and Specific Alarm Classes

	Critical Count
GENERAL ALARM MESSAGES	
1. Hose disconnection	2
2. Breathing circuit obstruction	3
3. Valve leak	3
4. Endotracheal tube problem	3
5. No alarm	1
SPECIFIC ALARM MESSAGES	
1. Endotracheal tube obstruction	3
2. Endotracheal tube cuff leak	3
3. Endotracheal tube disconnection	2
4. Inspiratory hose obstruction	3
5. Inspiratory hose leak	4
6. Expiratory hose obstruction	3
7. Expiratory hose leak	5
8. Inspiratory valve leak	3
9. Inspiratory valve obstruction	4
10. Expiratory valve leak	3
11. Expiratory valve obstruction	3
12. Inspiratory hose disconnection	2
13. Expiratory hose disconnection	2
14. No alarm	3

and

$$v_d[k] = 0.1(d[k] - m_d[k])^2 + 0.9v_d[k-1]. \quad (7)$$

Alarm Filtering

To reduce the number of false alarms, networks 4 and 5 must find the same fault in several of the past six breaths, before the alarm system gives a message to the user. We refer to the number of breaths for which a fault must be identified (within the previous six breaths) as the "critical count." Networks 2 and 3 must find the same fault in several of the past four breaths. Table 4 shows the minimal number (critical count) for each alarm message. These numbers reflect the reliability with which the networks identify each failure. In the case that both a specific alarm and general alarm are active, only the specific alarm message is reported.

Animal Study Protocol

We anesthetized 5 mongrel dogs (10 to 25 kg) with intravenously injected sodium thiopental (25 mg/kg) and used 0.5 to 1.5 vol% halothane to keep their mean arterial blood pressure near 100 mm Hg. We used a Narkomed anesthesia machine with an oxygen fresh gas flow of 1 to 3 L/min (Drager, Telford, PA). We adjusted the tidal volume to keep the $\mathrm{P_{ETCO_2}}$ between 30 and 40 mm Hg using an Ohio model 7000 ventilator (Ohmeda, Madison, WI). To prevent spontaneous breathing during the controlled ventilation portion of the experiments, we gave bolus doses of sodium thiopental.

Table 5. Ventilator Settings Used in Animal Testing

Breaths Per Minute	I:E Ratio
10	1:1.5
	1:2.5
15	1:1.5
	1:2.5
20	1:1.5
	1:2.5

We created all 13 specific faults listed in Table 4 in each dog experiment—at least once at each of the six ventilator settings listed in Table 5 and twice during spontaneous breathing. We maintained each fault for six breaths, then let the breathing circuit return to steady state. When the elements of $v[k]$ (eq 4) changed by less than 5% from one breath to the next, we created the next fault. The computer stored the feature vector $f_i[k]$ for each breath in a sequential file.

Neural Network Training

We used the data from the first 4 dogs to train the neural networks. We hand-coded a version of the Backward Error Propagation algorithm in Pascal and used the complete database (3,432 feature vectors) 300 times each, with the events presented in random order, with a learning rate of 0.001 and a momentum of 0.8 to train the network [8,9]. It took a RISC-based workstation 11.8 hr to train the five networks (DN10000, Apollo Computers, Chelmsford, MA). At the completion of training, we installed the resulting network interconnection weights in the five networks. When the network ran on a 16-MHz PC-386 computer, with a math coprocessor, it took 69.5 msec to classify each breath (floating point neural network implementation in Pascal).

We measured the performance of the neural network by presenting to it the feature vectors from the fifth dog (data not used in training). The "confusion" matrix in Figure 4 shows the faults created (rows) and the faults reported by the networks (columns). This training and testing process was repeated five times, until the feature

Event Created \ Event Reported	Endotracheal tube obst.	Endotracheal tube leak	Endotracheal tube disc.	Inspiratory hose obst.	Inspiratory hose leak	Expiratory hose obst.	Expiratory hose leak	Inspiratory valve leak	Inspiratory valve obst.	Expiratory valve leak	Expiratory valve obst.	Inspiratory hose disc.	Expiratory hose disc.	Specific alarm indeterminate
Endotracheal tube obst.	47													
Endotracheal tube leak		48	1											
Endotracheal tube disc.			47											
Inspiratory hose obst.				43					2					1
Inspiratory hose leak					47									1
Expiratory hose obst.						45					2			2
Expiratory hose leak							47							
Inspiratory valve leak								39		6				2
Inspiratory valve obst.				2				1	43					2
Expiratory valve leak				1				1		43				
Expiratory valve obst.											47			
Inspiratory hose disc.							1					42	3	
Expiratory hose disc.												3	48	

Fig 4. Alarm performance during controlled ventilation in 5 dogs. The faults were created as shown in the rows of the matrix. The columns show the number of times a message was reported by the alarm system. Column 14 (specific alarm indeterminate) indicates faults for which the system reported multiple alarm messages.

vectors for each dog had been used once for testing and four times for training.

Operating Room Protocol

We measured the performance of the alarm system by using it to monitor 20 patients during general anesthesia. The University of Utah Institutional Review Board approved the study and each patient gave informed consent. Inhalation anesthesia was delivered following standard clinical practice using a Narkomed 2B anesthesia machine (Drager, Telford, PA). The study included periods of controlled ventilation, spontaneous breathing, and manual ventilation. Before going into the operating room, we trained the alarm system neural networks using all of the data from the animal study.

Following intubation, we placed the alarm system sensors between the endotracheal tube and the breathing circuit Y-piece or between an elbow connected to the endotracheal tube and the Y-piece. The computer ran continuously during the operation, storing 30 features from each breath in sequential data files. In most cases, the sensors remained in the breathing circuit and data collection continued until extubation.

An independent observer, a graduate student not involved in providing patient care, watched each procedure and the computer display, and kept a record of all the breathing circuit events listed in Table 4 (e.g., disconnects, leaks, occlusions). He recorded the breath number (k) when each event occurred. Most observed events were not "critical" faults, but rather events that normally occur in the course of clinical care (e.g., endotracheal tube disconnect to install a heat-moisture exchanger into the circuit).

Upon completion of the operating room study, we used the data from 10 of the patients and the data from the animal study to train the neural networks. We added 20 hidden-layer nodes to networks 4 and 5 because of the increased training data. We used the data from the remaining 10 patients (data not used in training) to measure the accuracy of the trained alarm system by presenting these feature vectors to the system in the same order as they were collected in the operating room.

RESULTS

Animal Testing

The alarm system's four general alarm messages were correct for 99.7% of the 616 faults created during controlled ventilation in 5 dogs. General alarm messages were correct for 96.2% of the 130 faults created during spontaneous breathing (5 errors). The 13 specific alarm messages were correct for 95.0% of the 616 faults during controlled ventilation (30 errors), and for 86.9% of the 130 faults during spontaneous breathing (17 errors). Figures 4 and 5 show the confusion matrices, giving the number of times each fault was created (rows) and the number of times each message was reported by the alarm system (columns). For example, the second row in Figure 4 shows that an "endotracheal tube leak" was created 49 times. The alarm system reported the correct message, "endotracheal tube leak," 48 times. The alarm system reported an incorrect message once, "endotracheal tube disconnect." If the alarm system's performance were perfect, all of the numbers would appear on the matrix "diagonal."

Neural network number 1 correctly classified breaths as spontaneous or controlled for 98.9% of 26,733 total breaths. This total included breaths in which faults were present, as well as normal breaths. During the animal testing, the alarm system reported 26 alarm messages when no faults were present, giving a false-positive rate of 0.097%.

Operating Room Testing

Table 6 lists the 57 critical events that the observer saw during 43.6 hr in the operating room. The true-positive

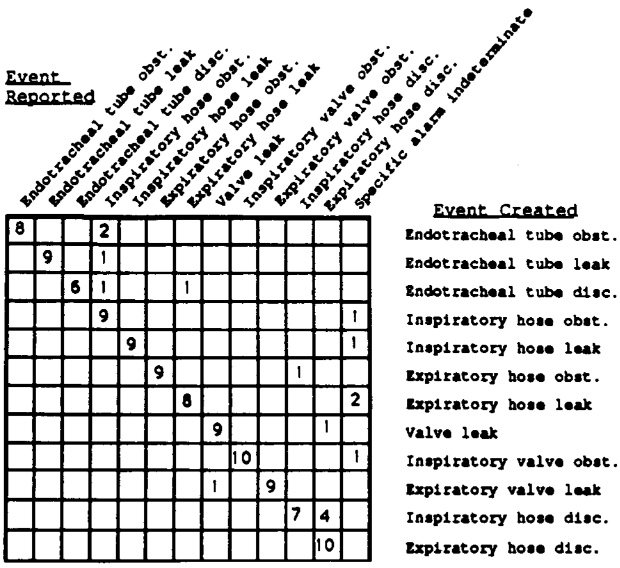

Fig 5. Alarm performance during spontaneous ventilation in 5 dogs. The rows of the matrix show the faults that were created. The columns show the number of times a message was reported by the alarm system. Column 14 (specific alarm indeterminate) indicates faults for which the system reported multiple alarm messages.

column shows that 54 of the 57 events were correctly identified by the alarm system. For 44 of these events, the alarm system gave specific alarm messages. For 10 events, it gave general messages. The false-negative column shows that two events were not detected by the alarm system: an inspiratory hose leak and an endotracheal tube obstruction. The incorrect message column shows that on one occasion the wrong alarm message was given in response to an endotracheal tube disconnection.

The false-positive column shows that the system reported 74 alarms when the observer did not see an event. This gives an average false-positive alarm rate of 1.7 alarms per hour. The majority of these alarms (52) occurred when the surgeon leaned on the patient's chest. The abrupt change in pulmonary compliance triggered the "endotracheal tube obstruction," the "breathing circuit obstruction," or the "inspiratory hose obstruction" alarms. The "endotracheal tube leak," the second most common false-alarm message, occurred when the surgeon removed his weight from the patient's chest, abruptly altering the difference between inspired and expired tidal volumes.

Twice during the clinical study, the alarm system helped diagnose unknown faults. During reconstructive

Table 6. Results of Alarm System Testing in the Operating Room

Alarm Type	True Positive	False Positive	False Negative	Incorrect Message
SPECIFIC ALARMS				
Expiratory hose disconnect	15			
Inspiratory hose disconnect	4			
Expiratory valve leak		1		
Inspiratory valve leak		2		
Expiratory hose leak		1		
Expiratory hose obstruction	2	1		
Inspiratory hose leak		1	1	
Inspiratory hose obstruction	1	6		
Endotracheal tube disconnect	17			1
Endotracheal tube cuff leak	5	12		
Endotracheal tube obstruction		13	1	
GENERAL ALARMS				
Hose disconnect	8	2		
Breathing circuit obstruction		13		
Valve leak		2		
Endotracheal tube problem	2	20		
TOTAL	54	74	2	1

A total of 57 events were picked up by the observer. The true-positive column shows that the system correctly reported 54 of the 57 events. The false-positive column shows that 74 alarm messages were given when no event was observed. The false-negative column shows that no message was given when two events were present. Column 4 shows that one event was incorrectly reported.

surgery of the nasal septum, the ventilator low-pressure alarm sounded, but the attendee did not find the disconnected endotracheal tube hidden under the drapes until after the neural network alarm system reported the fault. In another case, following rotation of a patient from the supine to the prone position, for lumbar decompression, the alarm system correctly reported an endotracheal cuff leak. Because neither the conventional low minute volume nor the low airway pressure alarm sounded, the event might otherwise have gone unnoticed.

DISCUSSION

Animal test results show that the neural network based alarm system correctly identified 99.7% of the breathing circuit faults during the best conditions (controlled ventilation, general fault messages) and 86.9% of the faults in the worst conditions (spontaneous breathing, specific fault messages). When used in the operating room, the alarm system correctly identified 54 of the 57 events seen by the observer. Given this true-positive rate and the very specific nature of the alarm messages, the system should be of significant advantage in helping the anesthesiologist quickly find breathing circuit faults.

Our results for the neural network based system are similar to those obtained using knowledge-based reasoning, thresholds, and "if–then" type rules. Brunner, Loeb, and Jiang used knowledge-based systems to generate specific alarm messages from features extracted from the pressure, flow, and CO_2 signals [10–13]; van der AA used a true expert system (Simplexys) [4]. Animal test results are very similar for all systems; but, a comparison of true-positive and false-positive rates is not meaningful because of dissimilar test conditions. Results from the clinical testing of these rule-based systems have not been published.

Although neural network system performance seems similar to the performance achieved with knowledge-based systems, there are advantages to both approaches and the ideal intelligent alarm system probably will use both. Knowledge-based systems generally use the simplest and most apparent relationships between the sensor signals and faults; neural networks embody the complete, and often extremely complex, relationships between sensor signals and faults (as they appear in the training data). The author of a rule-based system knows which signals are truly important and writes rules accordingly. Most if–then rules are based on one or two variables and, even though the variables selected may be those that change most when a critical event occurs, subtle changes in other variables may also contain useful information [10–14]. The author of the production rules must understand the behavior of all features during each fault, for various breathing circuit configurations and for all ventilator settings. The author must remove personal bias he or she has gained from past experience.

A neural network, on the other hand, uses information from all variables simultaneously, in an unbiased way [8]. The more complete and more representative the training data, the better the neural network performs. All information from all sensors is used by the neural network. Although the same can be done with rule-based systems, the rules become extremely complex. Our neural network used 30 input variables for each decision, a formidible task for the author of AI rules. If rule-based systems have a weakness, generally it is a lack of medical knowledge. If neural networks have a weakness, generally it comes because of an incomplete training database. For our application, we believe it is easier to create a complete database (via simulation) than it is to understand the complex relationship between the sensors and the breathing circuit faults.

Our neural networks have an advantage over existing rule-based systems in that they detect abnormal states directly, rather than just as changes from normal. The van der Aa [4] and Jiang [12,13] systems need a learning period to collect data from several normal breaths from each patient, a period of normal breathing, before they can identify events. Their rules are triggered when measured variables cross thresholds, the thresholds being based on information from past normal breaths in that patient. Because their system detects any deviation from the normal steady state, they alarm falsely when the tidal volume, ventilation rate, or fresh gas flow change. Our neural network bases part of its decision on absolute features; thus, it does not require a learning period or a period of normal respiration before it can detect faults when used in the operating room. It detects abnormal states upon start-up, because it was trained offline, long before the case began. Our neural network also uses differential features, which give it increased sensitivity to detect changes, thus incorporating some of the advantages of the van der Aa and Jiang systems.

Equivalent performance seems to have been achieved with neural networks, without relying on an expert's judgment, and with simpler options for updating the system. In many cases, it is easier to find good data than it is to find an expert with sufficient time to write rules. This may be particularly true for breathing circuit alarms. When equipment is upgraded, it would seem that collecting new data and retraining is preferable to reassembling the experts for a rewrite of AI rules. Updating the system requires that the expert revisit all rules, and carefully evaluate how changes may affect the old rules. With neural networks, a nonexpert uses simulation to create a database, and training algorithms establish the equivalent of thousands of optimum rules [15,16]. Updating the networks or expanding the area of application requires the creation of new training data, and perhaps increasing the size of the network to handle the increased complexity of the data; but, the network training and implementation algorithms remain unchanged. If a weakness is found in the system, new examples are added to the training data, the neural network is retrained and performance improves. For example, in our work, we found it necessary to include in the training base examples where ventilation and fresh gas flow were changed, to reduce false alarm in the clinical study.

We found it best to train one network to find faults upon start-up (using absolute features) and to train a second network to find more specific faults during operation (using differential features based on breath-to-breath changes in the absolute features). Because differential features are more sensitive to change, and because changes are often rapid when faults occur, the network that uses differential features can recognize all 13 specific faults. However, the differential network cannot be used alone, because it cannot detect faults that are present when the system is first turned on, nor can it detect faults that develop slowly over the course of many breaths. We needed a second network, which uses absolute features, to detect abnormal states upon start-up. Because this second network is less sensitive, it only identifies general faults and does not give their specific location.

The varying event detection threshold is a subtle, but significant, feature of our system (eq 5). Because the threshold (δ) is a function of the breath-to-breath change in a feature, it increases when the feature is unstable—causing the system to be less sensitive during spontaneous ventilation—and falls during times of stability—causing the alarm system to become more sensitive during controlled ventilation. This approach reduces false alarms during spontaneous ventilation, while maintaining sensitivity during controlled ventilation. During the clinical trial, even spontaneous overbreathing during controlled ventilation did not cause false alarms.

Methods for training and optimizing neural network architectures are developing rapidly. We used a simple, fully interconnected multilayer perceptron network trained with backward error propagation. Undoubtedly, there are other architectures and training methods that would reduce training time and improve performance.

The system's actual false-positive rate is probably lower than reported. One weakness in our clinical evaluation is that we could not intentionally create breathing circuit events, nor could we be certain that an event actually existed when the neural network alarm system sounded. Our observer could not look for a temporary leak around the endotracheal tube cuff or a momentary obstruction of a hose without disturbing surgery. The observer recorded only those events that were obvious to him, or obvious to the anesthesiologist.

Before the system is ready for use in the operating room, a larger and more diverse data set is needed for training. We expect performance to degrade when the system is used on patients who are different from those used for training. We expect performance degradation will be graceful because the system will continue to operate and will make its best guess. The system will need extensive training on a patient simulator to recognize all types of events in all types of patients and under all conditions. If, after expanded training, a weakness is found, new examples should be added to the training data and the neural networks should be retrained. A patient simulator probably is the best way to obtain these additional data because of its ability to create many faults over a wide range of conditions. We expect that the final system's true-positive rate in clinical use will be better than the true-positive rate measured in our animal study (99.7% to 86.7%). False-positive alarms arose from events the system was not trained to recognize. Training the system to recognize a more diverse set of events should reduce false-positive alarms. Performance should improve for spontaneous breathing as new training data are collected during spontaneous breathing, to supplement the sparse data we now have. The false-positive rate should fall well below 1.7/hr as the system is trained to deal with surgical manipulations, suctioning, and manual ventilation.

Although our study emphasizes neural networks, some alarm tasks are better handled using knowledge-based methods. The best alarm system will be one that combines many decision-making tools. We anticipate, however, that neural networks will be considered in the design of all future alarm systems [17].

Ohmeda, Diatek Corporation, and the Anesthesia Patient Safety Foundation supported this work in part. This work was presented in part at the ASA Annual Meeting, October 1991, San Francisco, CA.

The authors wish to thank Lue-Ellen Merchant for her assistance in the preparation of this article.

REFERENCES

1. Davis DA. An analysis of anesthetic mishaps from medical liability claims. Int Anesthesiol Clin 1984;22:31–42

2. Tinker JH, Dull DL, Caplan RA, et al. Role of monitoring devices in prevention of anesthetic mishaps: A closed claims analysis. Anesthesiology 1989;71:541–546
3. Epstein RA. The elusive disconnect alarm examined. APSF Newsletter, Dec 1988:39
4. van der Aa JJ. Dissertation: Intelligent alarms in anesthesia. Eindhoven, The Netherlands: Eindhoven Technical University, 1990
5. Cooper MG, Vouden J, Rigg D. Circuit leaks. Anaesth Intens Care 1987;15:359–360
6. Solazzi RW, Ward RJ. The spectrum of medical liability cases. Int Anesthesiol Clin 1984;22(2):43–59
7. Pyles ST, Berman LS, Modell JH. Expiratory valve dysfunction in a semi-closed circle anesthesia circuit-verification of carbon dioxide waveform. Anesth Analg 1984;63:536–537
8. Lippman RP. An introduction to computing with neural nets. IEEE ASSP Mag, Apr 1987:4–22
9. Rumelhart DE, Hinton GE, Williams RJ. Learning internal representations by error propagation. In: Rumelhart DE and McClelland JL, eds. Parallel Distributed Processing. Cambridge, MA: MIT Press, 1986:318–362
10. Brunner JX, Westenskow DR, Zelenkov P. Prototype ventilator and alarms algorithm for the NASA space station. J Clin Monit 1989;5:90–99
11. Loeb RG, Brunner JX, Westenskow DR, et al. The Utah anesthesia workstation. Anesthesiology 1989;70:999–1007
12. Jiang A, Kambam KJ, Smith BE. Real-time respiratory system monitoring. Int J Clin Monit 1991;8:139
13. Jiang A. PhD dissertation: The design and development of a knowledge-based ventilatory and respiratory monitoring system. Nashville, TN: Vanderbilt University, Dec 1991
14. Rennels GD, Miller PL. Artificial intelligence in anesthesia and intensive care. J Clin Monit 1988;4:274–289
15. Orr JA, Westenskow DR. A breathing circuit alarm system based on neural networks. Anesthesiology 1989; 71:A339
16. Wasserman PD, Schwartz T. Neural networks, Part 2: What are they and why is everyone so interested in them now? IEEE Expert, Spring 1988:10
17. Stubbs DF. Neurocomputers. MD Comput 1988;5:3

NOTICE REGARDING MANUSCRIPT SUBMISSION

As of January 1, 1994, the *Journal of Clinical Monitoring* officially moved its Editorial Office from California to Florida. Please note the following new address for submitting manuscripts:

J. S. Gravenstein, MD, Editor
Journal of Clinical Monitoring
Department of Anesthesiology, Box 100254
University of Florida
College of Medicine
Gainesville, FL 32610-0254