Zustandsdiagnose mittels Signalverarbeitung und Neuronaler Klassifikation als Mikrosystem-Applikation

Olaf Enge-Rosenblatt, Holger Priwitzer, Peter Schneider
Fraunhofer-Institut Integrierte Schaltungen, Teilinstitut Entwurfsautomatisierung,
Zeunerstraße 38, 01069 Dresden, Olaf.Enge@eas.iis.fraunhofer.de

Eva Schönitz, Günther Wolf, Thomas Reichert, Claus Dittrich
Gesellschaft für Mikroelektronikanwendung Chemnitz mbH, Zwickauer Straße 227, 09116 Chemnitz

Kurzfassung

Der Überwachung von Maschinen und Anlagen mit Hilfe von Verhaltensanalyse- und Fehlerprädiktionssystemen kommt heutzutage eine immer größere Bedeutung zu. In diesem Beitrag wird eine Methode vorgestellt, die eine Online-Überwachung von Maschinen und Anlagen mittels eines Condition-Monitoring-Systems (CMS) auf µ-Controller-Basis gestattet. Die Methode stützt sich auf eine breite Palette von Signalverarbeitungsalgorithmen zur Merkmalsextraktion und auf neuronale Klassifikationswerkzeuge.

1 Einleitung

Produzierende Unternehmen stehen zunehmend vor dem Problem, ihre Wettbewerbsfähigkeit nur durch eine Produktion auf höchstem Niveau und mit maximaler Auslastung ihrer Produktionsmittel aufrecht erhalten zu können. Dazu müssen alle technischen Einrichtungen mit größtmöglicher Zuverlässigkeit arbeiten und bei Minimierung der Instandhaltungszeiten nahezu ständig verfügbar sein. Deshalb kommt der Überwachung von Maschinen und Anlagen mit Hilfe von Verhaltensanalyse- und Fehlerprädiktionssystemen eine immer größere Bedeutung zu. Eine vorausschauende und intelligente Instandhaltung ist heutzutage ein wesentlicher Wettbewerbsfaktor ([4]). Der zusätzliche Aufwand für solche Analysesysteme muss allerdings in einem ausgewogenen Verhältnis zum Nutzen stehen. Der Einsatz von kostengünstigen Mikrocontroller-Systemen ist dabei eine vielversprechende Möglichkeit.
Auf dem Gebiet der Zuverlässigkeit spielt der Verschleiß eine zentrale Rolle. Die Zustandsüberwachung und Vorhersage von verschleißbedingten Schäden ist ein viel beachtetes Wissenschaftsgebiet. In zahlreichen Publikationen werden Grundlagen und praktische Anwendungen zu diesem Thema behandelt (z.B. [1], [2], [3]). Aktuell werden auf dem Markt einige Condition- Monitoring-Systeme angeboten. Neuere Entwicklungen wurden z.B. in [4] vorgestellt. Allerdings beschränken sich diese Systeme sehr oft auf die Über-wachung von Schwellwerten für bestimmte physikalische Größen. Hauptsächlich wird aber in den meisten bekannten CMS der Beherrschung der Datenverwaltung und Logistik breite Aufmerksamkeit gewidmet. Bei der Festlegung der zu überwachenden Größen sowie ihrer Schwellwerte werden die Anwender gar nicht oder nur in sehr geringem Maße unterstützt.

Aber gerade diese Informationen, mit welchen Messsignalen und durch welche Signalverarbeitungsalgorithmen zuverlässige Erkenntnisse über den aktuellen Zustand der überwachten Maschine oder Anlage gewonnen werden können – sind dem Standard-Anwender a priori unbekannt. Hier ist eine weitreichende Unterstützung des Nutzers wünschenswert und letztlich für den durchschlagenden Erfolg eines CMS sogar notwendig.
Während des Einsatzes an der Maschine oder Anlage muss das CMS sehr einfach handhabbar sein. Die bereitgestellten Informationen sollten prägnant und unmissverständlich sein und den Bediener in kürzester Zeit zu einer Entscheidungsfindung befähigen.

2 Grundidee

Die Grundidee des hier vorgeschlagenen Mikrocontroller-basierten CMS ist in Bild 1 dargestellt. Ausgangspunkt sind die mittels geeigneter Sensoren gewonnenen Messsignale. Durch eine applikationsspezifische Verarbeitung dieser Messreihen werden Merkmale extrahiert, die Aussagen über den sich ändernden Zustand des Beobachtungsobjektes ermöglichen. Diese Merkmale werden mit einem zuvor entsprechend trainierten Neuronalen Netz klassifiziert. Durch Auswertung der Merkmalsklassen können dann Rückschlüsse auf den Gesamtzustand des Objektes gezogen werden.
Das mikrocontrollergestützte CMS ist als Stand- alone-Gerät konzipiert und kann leicht in das zu überwachende System implementiert werden.

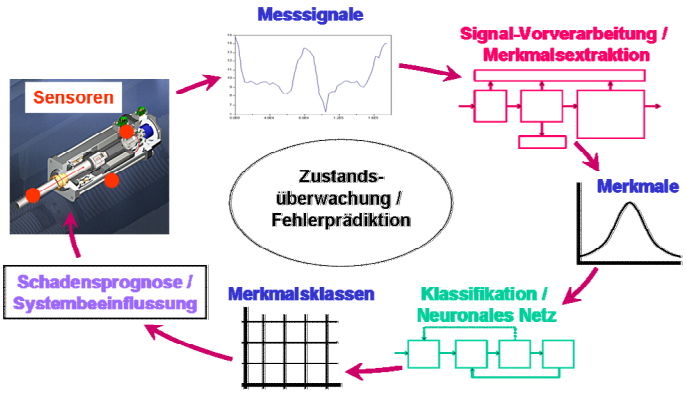

Bild 1: Prinzip der Zustandsüberwachung

Um eine bestimmte Überwachungsaufgabe realisieren zu können, sind im Vorfeld umfangreiche Untersuchungen hinsichtlich geeigneter Messsignale und merkmalsextrahierender Verarbeitungsalgorithmen sowie geeigneter Strukturen von Neuronalen Netzen notwendig. Die notwendigen Untersuchungsschritte zur Gewinnung dieser Erkenntnisse werden im Weiteren Experimentierphase genannt. Daran schließt sich die Entwurfs- und Trainingsphase an, in welcher mittels aus umfangreichen Messungen gewonnenen Merkmalsvektoren die Topologie des Neuronalen Netzes für den konkreten Anwendungsfall trainiert wird. Schließlich können die zielführenden Signalanalysepfade und das fertig trainierte Neuronale Netz auf der Zielhardware implementiert werden.

3 Signalanalyse

Die Signalanalyse dient der Ermittlung von Merkmalen, die einen Rückschluss auf den aktuellen Zustand der Maschine oder Anlage zulassen. Dazu werden Messsignalverläufe bestimmter, als geeignet erachteter physikalischer Größen mit verschiedenen mathematischen Algorithmen verarbeitet.
Die Funktionalität der Signalanalyseschritte muss für jede Applikation spezifisch festgelegt werden. Das beinhaltet – neben der Auswahl der zu messenden Größen – vor allem die Auswahl der Signalanalysefunktionen sowie deren sequenzielle Abfolge und Parametrisierung. Das ist eine schwierige und zeitraubende Prozedur, die wohl zumeist (jedenfalls aber bei nicht vorhandenem a-priori-Wissen) aus einem schrittweisen Ausprobieren verschiedener Signalanalysefunktionen besteht. Dabei ergeben sich naturgemäß sowohl zielführende (d.h. zu einem auswertbaren Merkmal führende) als auch nicht zielführende (und damit wertlose) Schrittfolgen. Deshalb kann diese Prozedur treffend als Experimentierphase bezeichnet werden. Sie kann als abgeschlossen betrachtet werden, wenn genügend zielführende Signalanalysepfade detektiert wurden.
Zur weitgehenden Unterstützung des Anwenders bei der Experimentierphase wird eine grafische Nutzeroberfläche (GUI-Graphic User Interface) bereitgestellt (siehe Bild 2). Innerhalb dieser GUI stehen die unterschiedlichsten Signalanalysefunktionen zur Verfügung. Diese können sowohl auf die Ausgangsdaten als auch auf Zwischenergebnisse angewandt werden. Dabei wird die Kompatibilität der entsprechenden Datenformate automatisch geprüft. Durch die mehrmalige (parallele und/oder sequenzielle) Anwendung von Algorithmen wird eine zunehmende Verzweigung erzeugt und es entsteht letztlich eine Baumstruktur von Zwischenergebnissen. Diese wird in der GUI visualisiert. Der Nutzer kann sich die entsprechenden Zwischenergebnisse anschauen und bestimmte Zweige positiv oder negativ bewerten. Schließlich kann er bestimmten Endknoten des Baumes die Eigenschaft „Merkmal“ zuweisen. Mit jeder solchen Zuweisung wird ein zielführender Signalanalysepfad spezifiziert.
Die innerhalb der GUI verfügbaren mathematischen Algorithmen können in die Kategorien Elementarverarbeitung, Zeitbereich/Bildbereich und statistische Analyse unterteilt werden. Alle Funktionen wurden für die Mathematikwerkzeuge Matlab, Octave und Scilab bereitgestellt. Damit ist bereits die Experimentierphase auch durch kostengünstige Werkzeuge abgedeckt.

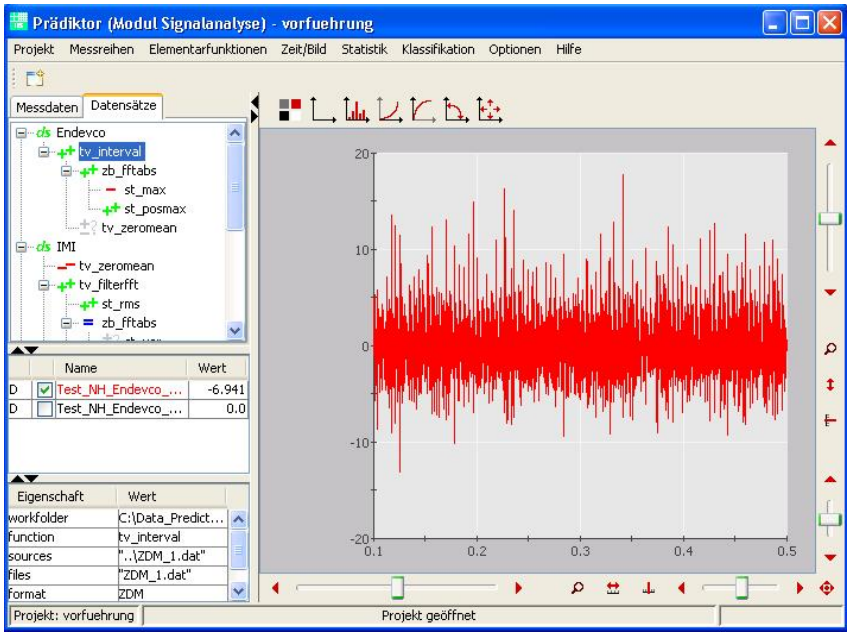

Bild 2: Grafische Oberfläche für die Signalanalyse

Die aktuelle Struktur des Signalanalyse-Baumes (und somit auch die evtl. bereits spezifizierten Signalanalysepfade) wird in einer Projektverwaltung innerhalb der GUI abgespeichert. Diese Eigenschaft ermöglicht nach erfolgreicher Beendigung der Experimentierphase eine automatische Abarbeitung der sogenannten Trainingsphase. Hierbei werden die gefundenen zielführenden Signalanalysepfade quasi auf „Knopfdruck“ für alle zum Training vorgesehenen Messreihen abgearbeitet. Damit ist die Bestimmung der Merkmalsvektoren für das Training des Neuronalen Netzes aus den dafür vorgesehenen Trainingsmessreihen sehr einfach durchführbar und auch jederzeit wiederholbar (z.B. bei Verfügbarkeit von weiteren, bisher nicht verwendeten Trainingsmessreihen). Hauptmotivationen für die GUI sind also einerseits die weitgehende Unterstützung der Experimentierphase und andererseits die Automatisierung der Trainingsphase.

4 Neuronale Klassifikation

An Bauteilen und Baugruppen von Maschinen und Anlagen ist Verschleiß ein Prozeß, der zeitfortschreitend negative Eigenschaften durch verschiedenste physikalische Vorgänge hervorbringt, die letztendlich zur Funktionsbeeinträchtigung und zum Ausfall dieser Bauteile oder Baugruppen führen. Dieser Prozeß vollzieht sich im Allgemeinen in größeren Zeitabschnitten und nur bei fehlerhafter Dimensionierung oder unzulässiger Belastung der Bauteile kurzfristig. Verschleiß messtechnisch zu erfassen, setzt Langzeitmessungen voraus, aber auch Expertenwissen über Verschleiß, das von dem Betreiber der jeweils betrachteten technischen Objekte gesammelt wurde.

Der Neuronale Klassifikator des CMS dient der automatischen Ermittlung des momentanen Verschleißzustandes der Maschine oder Anlage und ersetzt damit Wartungspersonal, das sonst die Verfügbarkeit des technischen Objektes überwachen müsste. Der Neuronale Klassifikator ist ein adaptives, lernfähiges System auf der Basis Künstlicher Neuronaler Netze. Dieser Klassifikator wird abhängig von den in der Experimentierphase der Signalanalyse extrahierten Merkmalen in einer spezifischen Entwurfsphase über eine unter der Software-Oberfläche der GUI aufrufbaren Entwurfs- und Simulations-Software für Neuronale Netze entworfen und validiert. Dieser Entwurfsprozess gliedert sich in zwei Teile, in die Aufbereitung der bereits in der Signalanalyse ermittelten Merkmale als Merkmalsvektoren und den Entwurf einer Netztopologie für eine verfügbare Netzstruktur.

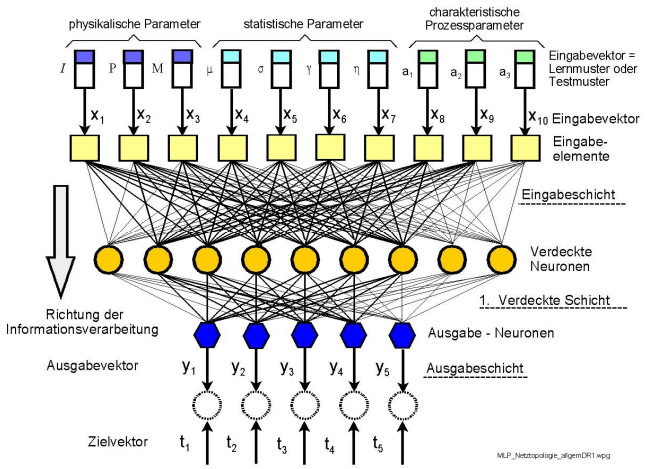

Bild 3: Neuronaler Klassifikator zur Verarbeitung von Datenreihen

Die Merkmalsvektoren werden als Spaltenvektoren (sogenannte Datenreihen, z.B. aus statistischen Daten) oder als Zeilenvektoren, die einem zeit- oder frequenzabhängigen Vorgang (z.B. Schwingungen) entstammen als Trainings- oder Testdaten für die Klassifikation formiert. Bild 3 zeigt die Topologie eines Neuronalen Klassifikators für die Klassifikation von Datenreihen. Für die numerische parametrische Klassifikation dieses CMS werden Feedforward- oder Partiell-Rekurrente-Netzstrukturen eingesetzt. Eine weitere Aufgabe im Entwurfsprozess ist die sinnvolle Festlegung der Klassen zur Charakterisierung des Verschleißprozesses am betrachteten Objekt. Für jeden spezifischen Einsatzfall sind Trainingsvorgänge mit Musterdatensätzen notwendig.
In der Adaptionsphase, der sogenannten Trainingsphase wird das Netz mit einem Satz von Mustern, der aus P Merkmalsvektoren m_P, p = 1 ... P, mit jeweils n Komponenten besteht, trainiert. Bei Anwendung des Lernprinzips „Überwachtes Lernen“ müssen dabei auch die Zielvektoren bekannt sein. Die Ausgabegrößen Y des Netzes, die für diesen Anwendungsfall einen Verschleißzustand repräsentieren sollen, werden vorteilhaft als binäre Größen (0,1) dargestellt, können aber auch als Festkommazahl bei entsprechender Formierung der Ausgabeneuronen ausgegeben werden. Durch Simulation in der Trainingsphase werden die Neuronalen Netze mit Testdaten des Objektes validiert. Durch Umsetzung der Topologie des Netzes in den Code eines Mikrocontroller-Programmes erfolgt die Implementierung in die vorgesehene Zielhardware.

5 Realisierung auf µ-Controller

Das CMS besteht grundsätzlich aus der µ-Controller-basierenden Hardware und einer spezifischen, den Einsatzbedingungen entsprechenden Software. Die Implementierung der Software, die aus den benötigten Signalanalysealgorithmen und dem trainiertem Neuronalen Netz besteht, erfolgt auf dem µ-Controller, der das Kernstück der Hardware darstellt. Für die vorgesehene Applikation wurde ein leistungsfähiger ARM9-Controller ausgewählt, der neben der hohen Rechenleistung noch weitere für das System günstige Parameter aufweist.
Hier war wichtig, dass das System in der Lage ist, sowohl die Signalanalysealgorithmen für jedes auszuwertende Messsignal als auch die Realisierung der Klassifikation mittels Neuronalem Netzes ohne zeitliche Engpässe zu bewältigen.
Das Gerät verfügt über geeignete Schnittstellen für die Einspeisung von Messsignalen (wahlweise analog und/oder digital), die analoge Aufbereitung der Messsignale, die digitale Messwertverarbeitung und die benötigten Ein- und Ausgangsschnittstellen für das Gesamtsystem.
Zur Auswertung bzw. Anzeige des aktuellen Zustandes der überwachten Maschine oder Anlage existieren mehrere Möglichkeiten, z.B. eine einfache „Ampel“ (grün: in Ordnung; gelb: umfangreiche Diagnose empfohlen; rot: Alarm).
Weiterhin besteht die Möglichkeit, besondere Ereignisse auf einer SD Card aufzuzeichnen. Das CMS befindet sich in einem Alu-Druckguss-Gehäuse (IP65) und ist damit für den Einsatz unter industriellen Bedingungen geeignet.

6 Zusammenfassung

Der Beitrag stellt eine Methode vor, die die Entwicklung von Mikrocontroller-basierten Condition-Monitoring-Systemen für die Online-Überwachung von Maschinen und Anlagen in sehr wesentlichen Teilen unterstützt. Damit wird der Entwurf von kostengünstigen CMS entscheidend effektiviert.
Das fertige System zeichnet sich durch eine leicht verständliche Bedienung aus und kommt vorzugsweise als „Stand-Alone“-Gerät zum Einsatz.
Das Vorhaben wurde mit Mitteln des BMBF unter dem Förderkennzeichen 02PG1280 gefördert.

Literatur

[1] Dongfeng Shi; Gindy, N.N.: Industrial Applications of Online Machining Process Monitoring System. IEEE/ASME Transactions on Mechatronics, Volume 12, Issue 5, Oct. 2007, pp. 561-564

[2] Hayashi, S.; Asakura, T.; Sheng Zhang: Study of machine fault diagnosis system using neural networks. Proc. International Joint Conference on Neural Networks (IJCNN '02), 12-17 May 2002, Page(s): 956-961

[3] Kyusung Kim; Parlos, A.G.: Induction motor fault diagnosis based on neuropredictors and wavelet signal processing. IEEE/ASME Transactions on Mechatronics, Volume 7, Issue 2, June 2002, pp. 201-219

[4] VDMA-Informationsveranstaltung „Intelligenter Produzieren“, Frankfurt, 7. Nov. 2008

[5] Wan Shuting; Li Heming; Xu Zhaofeng: A new method of turbine-generator vibration fault diagnosis based on correlation dimension and ANN. International Conference on Power System Technology (PowerCon 2002), 13-17 Oct 2002, Proc., Vol.3, Page(s): 1655-1659