
Wege aus der Ironie in Richtung ernsthafter Automatisierung

Andreas Lüdtkke

Einleitung

Dieser Text beschäftigt sich mit konzeptuellen Überlegungen und einem Vorgehensmodell zur systematischen Berücksichtigung des Faktors Mensch im Rahmen der Automatisierung von Aufgaben zur Überwachung und Steuerung sicherheitskritischer Prozesse. Gemeint sind Prozesse wie z. B. industrielle Fertigungsprozesse oder auch Fahrprozesse mit dem Auto oder Flugzeug. Der Mensch, um den es hier geht, ist (1) der menschliche Operateur, der unter Nutzung der Automatisierungssysteme den Fertigungsprozess überwacht, ggf. steuernd eingreift und Reparaturmaßnahmen in die Wege leitet, sowie (2) der menschliche Fahrer oder Pilot, der unterstützt durch Assistenzsysteme ein Fahrzeug oder Flugzeug steuert bzw. überwacht, wie dies von weitgehend autonome Autopiloten erledigt wird.¹ Die autonome Steuerung ist in Flugzeugen weit vorangeschritten, aber auch im Auto wird dies mit großer Wahrscheinlichkeit in absehbarer Zukunft Realität werden. Automatisierung wurde in der Vergangenheit und wird auch heute noch oft unter der Annahme betrieben, dass der Mensch das schwächste Glied der Kette ist und dass er deshalb nach und nach vollständig ersetzt werden muss. Dies hat zu einigen Ironien geführt, die eine neue Qualität von Fehlerarten hervorgebracht haben, die häufig lapidar als „Fehlbedienungen“ abgetan werden. Der Text zeigt auf, dass diese Ironien aus einer unzureichenden Auseinandersetzung mit dem Faktor Mensch resultieren. Es wird argumentiert, dass Automatisierung ohne Einbeziehung des Menschen nicht erfolgreich betrieben werden

¹Vereinfachend wird im Folgenden für beide Typen der Begriff Operateur verwendet. Gemeint ist damit also der menschliche Operateur oder spezifischer der Pilot oder Fahrer.

A. Lüdtkke (✉)

OFFIS, FuE Bereich Verkehr/R&D Division Transportation, Escherweg 2, 26121 Oldenburg, Deutschland

e-mail: andreas.luedtke@offis.de

kann. Die Rolle des Menschen muss explizit gestaltet werden und diese Rolle wegentwickeln zu wollen ist keine Lösung. Der Text argumentiert für die Auflösung einer starren Zuweisung von Aufgaben auf Mensch und Maschine. Eine optimale Aufgabenaufteilung kann nicht a-priori festgelegt werden, sondern muss zu jedem Zeitpunkt anhand festgelegter Verteilungsstrategien auf Basis situativer Erfordernisse neu bestimmt werden. Diese Flexibilität kann durch eine neue Perspektive auf die Mensch–Maschine Beziehung unterstützt werden: das Mensch–Maschine Team (MMT). In diesem Team wird interagiert, um die Aufgaben gemeinsam zu bewältigen und es wird kommuniziert, um ein gegenseitiges Verstehen herzustellen. Der Entwicklungsgegenstand ist nicht mehr das Automatisierungssystem sondern das MMT. Die Teamperspektive stellt die Entwickler vor neue Entwurfsherausforderungen. Der Text zeigt, dass diesen mit einem ingenieurmäßigen modellbasierten Vorgehen begegnet werden kann. Abschließend wird innerhalb dieses Vorgehens exemplarisch die Methodik des Ecological Interface Designs für den Entwurf der Mensch–Maschine Schnittstelle vorgestellt.

Das klassische Automatisierungsdilemma

Bainbridge hat 1983 in ihrem mittlerweile klassischen Beitrag die Ironien der Automatisierung von Überwachungs- und Steuerungsprozessen in der Industrie pointiert herausgestellt. Obwohl Bainbridge auf die Prozessindustrie fokussiert, führt sie bereits an, dass sich die Ironien auch in vielen anderen hochautomatisierten Bereichen wieder finden lassen. Das Dilemma der Automatisierung lässt sich beispielsweise sehr gut am Beispiel der Luftfahrt darstellen.

Die Automatisierung bestimmter Pilotenaufgaben erlebte in den 80er Jahren einen entscheidenden Auftrieb. Mit der Einführung der Boeing 767 im Jahre 1982 konnte die Arbeitslast im Cockpit so weit reduziert werden, dass statt der Drei-Mann-Besatzung nur noch zwei Piloten notwendig waren (Billings 1997). In diesen Anfängen der Cockpitautomatisierung wurde der Mensch als schwächstes Glied im Flugablauf betrachtet und es galt, seine Tätigkeiten so weit wie möglich durch Automatisierung zu ersetzen (Rüegger 1990). Nicht der Pilot, sondern das technisch Machbare stand im Vordergrund. Sicherheitssysteme, die riskante Manöver verhindern sollen, wurden eingebaut, z. B. Stallprotectionsysteme, um einen Strömungsabriss, und Speedprotectionsysteme, um zu hohe Geschwindigkeiten zu verhindern. Die menschliche Steuerung wurde mehr und mehr durch eine automatische Steuerung in Form von Autopiloten und Flight Management Systemen ersetzt, wobei sich die Rolle des Piloten vom *aktiv Handelnden* zum *passiv Überwachenden* wandelte. Die neue Rolle wird als *Supervisory Control* (Sheridan 1997) bezeichnet. Der Pilot programmiert und überwacht die Systeme, die für ihn die Steuerung des Flugprozesses übernehmen. Steuerungssysteme lesen den Zustand des zu steuernden Prozesses über Sensoren und manipulieren diesen über Aktuatoren.

Die Ironie beginnt damit, dass im Notfall von den Piloten verlangt wird, dass sie in die automatische Steuerung eingreifen und ggf. manuell übernehmen. Diese Sichtweise ist nicht konsistent mit der Vorstellung, dass der Mensch das schwächste Glied im Flugablauf

ist. Besonders kritisch ist, dass die Piloten vor allem in Flugphasen entlastet werden, in denen die Arbeitslast ohnehin bereits gering ist, z. B. im wenig anspruchsvollen Streckenflug durch völlige Reduzierung des körperlich-handwerklichen Aufwands. In kritischen Phasen mit hoher Arbeitslast (z. B. beim Start- oder Landeabbruch) hingegen wird die Belastung durch die oft kontraintuitive Bedienung der Automatik verstärkt (Sarter und Woods 1995).

Diese Situation lässt sich in ähnlicher Weise auf alle hochautomatisierten Bereiche beziehen, in denen die Aufgabe darin besteht, einen Prozess (z. B. Fertigungsprozess, Fahrprozess, Stromversorgung) zu überwachen und zu steuern. Bainbridge (1983) zeigt vier Ironien der Automatisierung auf:

- **Ironie 1:** Menschen werden von Entwicklern als wesentliche Fehlerquelle betrachtet und deshalb durch Automatisierung ersetzt. Allerdings sind auch die Entwickler Menschen und damit anfällig für Fehler, in ihrem Fall für Entwicklungsfehler. Dies führt dazu, dass eine Reihe operativer Fehler tatsächlich auf Entwicklungsfehler zurückgeführt werden können.
- **Ironie 2:** Aufgaben, die sich (derzeit) nicht automatisieren lassen, eventuell weil sie zu komplex sind und sich nicht vollständig a-priori spezifizieren lassen, werden auf den Menschen, also auf das schwächste Glied in der Prozesskontrolle übertragen.
- **Ironie 3:** Der Mensch wird durch Automatisierung ersetzt, weil die Systeme die Aufgaben besser durchführen können. Sie/er soll aber weiterhin die Systeme überwachen und prüfen, ob sie korrekt arbeiten. In Störfällen soll der Mensch dann eingreifen und ggf. manuell übernehmen.
- **Ironie 4:** Die zuverlässigsten Automatisierungssysteme erfordern den höchsten Aufwand an Trainingsmaßnahmen, weil sich im täglichen Betrieb keine Gelegenheit für aktive Kontrolle und Auseinandersetzung mit dem System bietet. Unverlässliche Systeme hingegen erfordern regelmäßiges aktives Eingreifen und Hineindenken in die funktionalen Zusammenhänge und erhalten damit die manuellen Kontrollfähigkeiten aufrecht, was den Trainingsaufwand reduziert.

Die Ironien zeigen, dass Entwickler dem Menschen implizit eine sehr wichtige Rolle auferlegen. Allerdings geschieht dies oft ohne sie/ihn mit geeigneten Mitteln auszustatten, die notwendig wären, um die Rolle zu erfüllen. Ein wichtiger Aspekt ist beispielsweise die Gestaltung der Mensch–Maschine Interaktion, um ein adäquates Situationsbewusstsein zu unterstützen. Hier müsste sichergestellt werden, dass der Operateur zu jedem Zeitpunkt die notwendige Information über den Zustand des Prozesses und die Kontrolleingriffe der Automatisierungssysteme in intuitiver Weise erhält. Der Trend geht eher dahin, dass der Mensch mehr und mehr aus der Kontrollschleife herausgenommen und eine zunehmende Distanz zum tatsächlichen Kontrollprozess geschaffen wird. Dadurch gehen praktische Kenntnisse darüber verloren, wie der Prozess zu steuern ist (Wiener und Curry 1980). Es findet eine Erosion manueller Fähigkeiten und intimer Kenntnisse über das Verhalten und die Steuerung der Prozesse statt. Diese Fähigkeiten und Kenntnisse sind aber essentiell, um in Störfällen effektiv eingreifen zu können.

Unbestritten ist sicherlich, dass die Einführung moderner Automatisierungssysteme zur Erhöhung der Sicherheit beigetragen hat. Dennoch muss festhalten werden, dass eine neue Qualität von Fehlern hervorgebracht wurde, die weitgehend ein Resultat der aufgezeigten Ironien sind. Stichwörter sind hier Automation Surprises (Sarter et al. 1997), Lack of Understanding (Endsley 1996), Mode Confusion (Norman 1981; Degani et al. 1999), Complacency (Parasuraman und Manzey 2010), Skill Degradation (Sherman 1997), Out-of-the-loop Effects (Endsley und Kiris 1995), Automation Misuse, Disuse, Abuse (Parasuraman und Riley 1997). In Unfall- und Vorfallberichten wird dann oft von „menschlichem Fehlverhalten“ oder „Fehlbedienungen“ gesprochen. Allerdings sind diese Fehler häufig nicht ursächlich auf den menschlichen Operateur zurückzuführen, sondern vielmehr auf ein Missverständnis menschlicher Fähigkeiten durch die Entwickler von Automatisierungssystemen bzw. auf eine unzureichenden Entwicklungsphilosophie kombiniert mit unvollständigen Entwicklungsprozessen und -werkzeugen, die den Faktor Mensch ignorieren oder nur unzulänglich berücksichtigen.

Automatisierung ohne explizite und systematische Berücksichtigung des Menschen kann nicht funktionieren. Ziel sollte es sein, ein Gesamtsystem bestehend aus Automatisierungssystemen und menschlichen Operateuren zu entwickeln, welches im *Zusammenspiel* seine Aufgaben, z. B. die Überwachung und Steuerung eines industriellen Fertigungsprozesses, verlässlich und sicher erfüllt. Dieses Gesamtsystem wird weiter unten als Mensch-Maschine Team weiter ausgeführt. Die Rolle der involvierten Menschen muss explizit unter Berücksichtigung eines tiefen Verständnisses der menschlichen Kognition entworfen werden. Das steigende Interesse an Methoden zur Berücksichtigung des Faktors Mensch in der Automatisierung zeigt, dass das Problem generell als relevant wahrgenommen wird. In der Forschung zeichnen sich vielversprechende Lösungsansätze ab, die allerdings noch Eingang in die industrielle Praxis finden müssen. Eine Voraussetzung dafür ist, dass, auf der einen Seite, Entwickler etwas über den Faktor Mensch lernen und dass, auf der anderen Seite, Human Factors Experten die Sprache der Entwickler sprechen. Die Werkzeuglandschaften der beiden Gruppen müssen integriert werden, sodass es für Entwickler einfacher wird, bereits in den frühesten Phasen der Entwicklung Anforderungen des Menschen systematisch zu berücksichtigen und in den folgenden Schritten konsequent umzusetzen. Zunächst gilt es jedoch, einige grundsätzliche Fragen bzgl. der Rolle des Menschen zu klären.

Die Rolle des Menschen in der Automatisierung

Wenn man akzeptiert, dass der Mensch eine entscheidende Rolle bei der Überwachung und Steuerung industrieller Prozesse spielt, dann muss man sich fragen, unter welchen Voraussetzungen er die Rolle bestmöglich ausfüllen kann. Bereits bei der Aufgabenaufteilung, also der Entscheidung, welche Funktionalität die Maschine haben soll, müssen (1) Fähigkeiten und Schwächen der menschlichen Operateure, (2) Möglichkeiten und Grenzen der Automatisierung sowie (3) das sinnvolle Zusammenspiel von Mensch und Maschine berücksichtigt werden.

Eine einfache Philosophie für die Automatisierung wäre, die Stärken sowohl der Menschen (what can men do better than machines) als auch der (zu entwickelnden) Maschinen (what can machines do better than men) zu analysieren und dann entsprechend die Aufgaben zuzuweisen (Fitts 1951). Fitts (1951) hat diese Stärken in einer Liste, die auch als MABA–MABA (Men Are Better At – Machines Are Better At) bezeichnet wird, zusammengefasst. Vereinfacht dargestellt sind demnach Menschen besser in vielen Aspekten der Entdeckung (kleinster Veränderungen), der Wahrnehmung, der Beurteilung, der Induktion, der Improvisation und des langfristigen Speicherns und Erinnerns von Information. Maschinen sind besser in vielen Aspekten der Geschwindigkeit, der Anwendung starker Kräfte, der Durchführung sich fortwährend wiederholender Aufgaben, der komplexen Berechnung/Deduktion, des Multitasking und des kurzfristigen Speicherns und Abrufens von Information. Beispielsweise zeigen zahlreiche Studien (siehe z. B. Warm et al. 1996), dass es für Menschen unmöglich ist, über lange Zeit aufmerksam einen Prozess zu überwachen, bei dem wenig passiert. Einige Studien weisen auf eine maximale Aufmerksamkeitsspanne von einer halben Stunde (Mackworth 1950). Menschen sind hingegen gut darin kreative Lösungen zu „erfinden“ in Situationen für die kein vorgefertigter Plan vorliegt. Diese Fähigkeit kann aber mit zunehmendem Zeitdruck abnehmen. Zeitdruck führt dazu, dass Menschen auf bewährte Lösungen bzw. einfache Heuristiken zurückgreifen (Simon 1955; Zsombok et al. 1992; Gigerenzer et al. 1999). Diese Heuristiken sind optimal angepasst an wiederkehrende Bedingungen. In völlig neuen Situationen können sie jedoch zu Fehlurteilen und gefährlichen Fehlhandlungen führen (Frey und Schulz-Hardt 1997; Javaux 1998; Lüdtke und Möbus 2004).

Nimmt man Fitts' Liste wörtlich, dann würde man lediglich die Fähigkeiten automatisieren, bei denen Maschinen den Menschen überlegen sind und der Mensch würde weiterhin seine Stärken einbringen können. Diese Aufteilung würde akzeptieren, dass Menschen eine entscheidende Rolle spielen und in einigen Aspekten den Maschinen überlegen sind. Dennoch betrachtet Fitts diese Zuweisung als problematisch, weil weitere Faktoren eine entscheidende Rolle spielen, wie beispielsweise die Zufriedenheit, Motivation und auch das Ansehen sowie Selbstverständnis der involvierten Menschen. Weiterhin berücksichtigt diese starre Aufgabenaufteilung, immer noch nicht, dass Fähigkeiten sowohl der Menschen als auch Maschinen von der konkreten Situation abhängig sind, z. B. von den situativen Herausforderungen und der induzierten Aufgabenschwierigkeit bzw. der induzierten Arbeitslast. MABA–MABA berücksichtigt nicht, dass in Abhängigkeit von der Situation Maschinenaufgaben dynamisch vom Mensch übernommen werden müssen (oder umgekehrt) und dass hierfür ein adäquates Situationsbewusstsein geschaffen sowie adäquate manuelle Fertigkeiten erhalten werden müssen. Fitts' Sichtweise behebt somit nicht die eingangs erläuterten Ironien der Automatisierung (Wiener und Curry 1980; Rouse 1981).

Eine möglicher Beitrag zur Lösung des Problems sind beispielsweise unterschiedliche Automatisierungsgrade (Parasuraman et al. 2000). Man spricht auch von adaptiver Automatisierung (Opperman 1994; Byrne und Parasuraman 1996). Die Grade in einem Steuerungssystem können sich von vollständig manuell bis vollständig autonom erstrecken. Dazwischen werden bestimmte Aufgaben vom Menschen und andere von der Maschine

übernommen. Wesentlich dabei ist, dass die Aufteilung nicht fix sondern dynamisch ist. Abhängig von der Situation können Übergänge entweder automatisch durch die Maschine (z. B. Übergabe an den Menschen, wenn Performanzgrenzen der Automatik erreicht werden) oder durch den Menschen initiiert werden. Diese Aufteilung ermöglicht z. B. den Menschen bestimmte Aufgaben selbst zu übernehmen, um manuelle Fähigkeiten aufrecht zu erhalten.

Die Einführung gestufter Automatisierungsgrade schafft eine wesentliche Voraussetzung für den Ausweg aus dem Automatisierungsdilemma. Das Konzept sieht explizit vor, dass die Kontrolle zwischen Mensch und Maschine wechseln kann. Damit rückt neben der grundsätzlichen *Aufgabenaufteilung* die Gestaltung der *Aufgabenübergabe* in den Vordergrund. Wenn der Mensch in bestimmten Situationen übernehmen soll, dann muss er dazu auch in die Lage versetzt werden, z. B. indem er bei bestimmten Aufgaben ausreichend in der Kontrollschleife gehalten oder rechtzeitig in die Schleife zurückgeholt wird. Die Frage ist also, wieweit der Mensch in die Aufgabe eingebunden werden bzw. über den Aufgabenverlauf informiert werden muss, sodass er ausreichend Situationsbewusstsein hat, um ggf. einzugreifen. Es ist zu berücksichtigen, dass die reine Überwachung einer Aufgabe weniger Situationsbewusstsein schafft als die aktive Durchführung.

Was muss der Mensch über die Maschine und den aktuellen Zustand des Kontrollprozesses wissen? Wie kann die Maschine dieses Wissen an den Menschen geeignet kommunizieren? Diese Fragen verschieben den Fokus der Automatisierung von der reinen Betrachtung der technischen Funktionalität hin zur Betrachtung des Zusammenspiels zwischen Mensch und Maschine. Diese Fokusverschiebung erfordert ein neues Verständnis des Entwicklungsgegenstandes. Der Entwicklungsgegenstand ist nicht länger das isolierte Automatisierungssystem. Die im Folgenden beschriebene Teamperspektive nimmt das Konzept der dynamischen *Aufgabenaufteilung* auf und bietet eine vielversprechende wenn nicht sogar notwendige Voraussetzung für die Auflösung des Automatisierungsdilemmas.

Teamperspektive für die Automatisierung

Eine sinnvolle Perspektive auf die Automatisierung komplexer Aufgaben ist die Betrachtung von Mensch und Maschine als Team (Mensch–Maschine Team, MMT). Diese Perspektive wurde bereits von Christoffersen und Woods (2002) sowie von Klein et al. (2004) vertreten. Dabei werden Mensch und Maschine als Gesamtsystem verstanden. Gesamtsystem bedeutet, dass Mensch und Maschine gemeinsam eine Menge von Aufgaben bearbeiten, um ein gemeinsames Ziel zu erreichen. Unteraufgaben werden je nach den situativen Erfordernissen dynamisch auf Mensch und Maschine verteilt.

In einem guten Team liegt ein wesentlicher Schwerpunkt auf der Kommunikation zwischen den Teammitgliedern. Kenntnisse über Fähigkeiten, Aktivitäten, Rollen und Pläne der anderen sind essentielle Bestandteile eines guten Situationsbewusstseins innerhalb eines Teams. In einem Team unterrichten sich Teammitglieder über ihre Pläne und stimmen sich gegenseitig ab. Man weiß, was der andere vorhat, welche Rolle er im Gesamtsystem

gerade spielt und welche er generell spielen kann. Es muss eine entsprechende Kommunikation stattfinden, um gegenseitiges Verstehen herzustellen.

Klein et al. (2004) postulieren zehn Herausforderungen für die Entwicklung von Automatisierungssystemen, die als Teamplayer agieren können. Diese Herausforderungen basieren auf der Annahme, dass Teammitglieder einen sogenannten Common Ground, also ein *gemeinsames Verständnis* der Arbeitssituation haben müssen. Dies beinhaltet beispielsweise, dass Maschinen die Operateure rechtzeitig informieren, wenn der Normalbetrieb des Prozesses ungewöhnliche Kontrolleingriffe erfordert und sich somit eine Störung abzeichnet. In diesen Situationen kann die Maschine den Menschen in die Kontrollschleife zurückholen und so ein Bewusstsein für die Kontrollsituation herstellen, welches durch reinen Informationsaustausch z. B. über Displays nicht vermittelbar ist. Auf diese Weise kann eine gemeinsame Problemlösung betrieben und ggf. ein Übernehmen durch den menschlichen Operateur vorbereitet werden.

Ein weiterer Aspekt der konsequenten Umsetzung des Teamkonzeptes beinhaltet, dass das Verhalten der Maschinen für den Menschen nachvollziehbar und vorhersagbar sein muss. Das bedeutet, dass die Automatik Methoden und Kriterien anwenden muss, die für die Operateure nachvollziehbar sind. Auch die Geschwindigkeit muss in bestimmten Situationen an die Verarbeitungsgeschwindigkeit der Menschen angepasst werden. Die Teamperspektive erfordert also in manchen Situationen Vorgehensweisen, die nicht maximal effizient sind, die aber wesentlich die Sicherheit des Gesamtsystems erhöhen (Bainbridge 1983). In der Gesamtbetrachtung führt dies zu einer stabileren Funktionalität des Gesamtsystems mit weniger Ausfällen und damit weniger Stillstand, sodass sich dieser Kompromiss letztendlich auszahlt.

Die Realisierung von MMTs erfordert eine neue Herangehensweise an die Entwicklung von Automatisierungssystemen. Nicht die einzelne Maschine sollte im Zentrum der Entwicklung stehen, sondern das Gesamtsystem. Das bedeutet, dass zunächst überlegt werden muss, welche Aufgaben das Gesamtsystem bearbeiten soll. Diese Aufgaben müssen in Teilaufgaben zerlegt und weiter spezifiziert werden. Strategien zur dynamischen Aufgabenallokation auf die einzelnen Akteure des Gesamtsystems müssen definiert und systematisch im Hinblick auf Lastverteilung und Sicherheit untersucht werden. Strategien zur Kommunikation zwischen Menschen und Maschinen müssen definiert und im Hinblick auf ein adäquates verteiltes Situationsbewusstsein analysiert werden.

Vorgehensmodell zur Entwicklung von Mensch–Maschine Teams

Die Teamperspektive inklusive der dynamischen Aufgabenallokation stellt besondere Herausforderungen an die Entwickler. Die Dynamik der Systemkonfiguration (definiert beispielsweise durch: wer macht zurzeit welche Aufgabe, wer kann welche Aufgaben übernehmen, wer hat welche Information) bewirkt eine hohe Systemvarianz. Für die Evaluation des Systems bedeutet dies, dass eine sehr große Menge von Testszenarien bewältigt werden muss. Klassischerweise werden Human Factors Aspekte, also z. B. das Situationsbewusstsein der Operateure, durch Probandentest empirisch in Simulationen oder mittels

Wizard-of-Oz Techniken (Dahlback et al. 1993) getestet. Die hohe Systemvarianz verlangt nach neuen Methoden, die ein effizienteres Vorgehen erlauben.

Die Entwicklung von Software für die technische Funktionalität wird in der Industrie im Falle sicherheitskritischer Anwendungen meist mit systematischen Ingenieurmethoden durchgeführt; das bedeutet: rigorose Modellierung von Anforderungen, formale Spezifikation der Funktionen, weitgehend formaler Test, ob die Spezifikation die Anforderungen erfüllt sowie Tests der Implementierung in bereits früh definierten Testszenarien und schließlich formale Sicherheitsanalysen (z. B. Fehlerbaumanalyse, Ereignisbaumanalyse). Die Mensch–Maschine Interaktion wird hingegen oft ad-hoc zum Schluss „draufgesetzt“ ohne eine auch nur im Ansatz vergleichbare systematische Vorgehensweise. Oft sind bereits die initialen Anforderungen unklar bzw. nur sehr abstrakt und unspezifisch formuliert. Das bedingt in der Praxis zum Teil ein ineffizientes Trial&Error-Vorgehen, bei dem Prototypen der Mensch–Maschine Interaktion iterativ ohne klaren Entwurfsprozess gebaut und dann getestet werden.

Dabei können Human Factors Aspekte von Mensch–Maschine Systemen, oder besser MMTs, ähnlich konsequent und systematisch entwickelt werden wie die technische Funktionalität. Hier bietet der Bereich des Cognitive Engineering adäquate Methoden und Werkzeuge. Cognitive Engineering ist eine wissenschaftliche Disziplin mit dem Ziel, Wissen und Techniken zur Unterstützung der Entwicklung von Mensch–Maschine Systemen nach kognitiven Prinzipien zur Verfügung zu stellen (Woods und Roth 1988). Das zugehörige Forschungsspektrum reicht von Studien zur empirischen Untersuchung kognitiver Prinzipien bis hin zur Entwicklung von Methoden und Werkzeugen zur Anwendung der Prinzipien. Es handelt sich um eine angewandte Disziplin, die auf Theorien und Methoden sowohl aus der Psychologie als auch aus der Informatik zurückgreift (Card et al. 1983). Cognitive Engineering legt Wert auf eine formale und weitgehend objektive Vorgehensweise. In der Praxis hängt die Anwendbarkeit der Methoden des Cognitive Engineering wesentlich davon ab, inwieweit sie zu den gängigen Methoden der technischen Entwicklung konsistent sind und sich in oder mit diesen integrieren lassen.

Im Folgenden wird ein Vorgehensmodell (vgl. Tab. 1, stark verkürzte Darstellung) skizziert, welches als Rahmen zur holistischen Entwicklung von MMTs und damit zur Integration der Entwicklung der Automatisierungssysteme und der Human Factors Aspekte verwendet werden kann.

Auf der obersten Ebene besteht das Vorgehensmodell aus den vier Bestandteilen MMT Komposition, MMT Kooperation, MMT Interaktion und MMT Schnittstelle. Jeder Bestandteil bezeichnet einen Entwicklungsgegenstand. Die *Komposition* bestimmt die generellen Aufgaben des MMT, die Anzahl und Typen der notwendigen Akteure (Operateure und Maschinen) und die Anzahl und Typen der notwendigen Ressourcen. Die *Kooperation* bestimmt, wer mit wem gemeinsam Aufgaben bearbeitet und wer ggf. Aufgaben von anderen Akteuren übernehmen soll. Die *Interaktion* bestimmt, welche Akteure miteinander über welche Inhalte mittels welcher Modalitäten kommunizieren. Die *Schnittstelle* bestimmt die konkrete Ausgestaltung der Interaktionsschnittstellen zwischen Mensch und Maschine (z. B. die Gestaltung grafischer Anzeigen), zwischen Mensch und Mensch (z. B.

Tab. 1 Vorgehensmodell für die Entwicklung von Mensch–Maschine Teams (MMT)

MTT Komposition

Anforderungsdefinition

- In welcher Umgebung soll das MMT arbeiten?
- Welche Aufgaben soll das MMT durchführen? ...

Spezifikation

- Welche Ressourcen werden für das MMT benötigt?
- Aus welchen Operateuren (Rollen, Skills, ...) und Maschinen soll sich das MMT zusammensetzen? ...

Implementation

- Implementierung der Maschinenfunktionalität,
- Auswahlkriterien und -verfahren sowie Trainingsprogramme für die Operateure festlegen, ...

Evaluation

- Abschätzung, ob die Operateure und Maschinen ausreichen, um die Aufgaben zu bewältigen, ...

MMT Kooperation

Anforderungsdefinition

- Welche Akteure sollen zusammenarbeiten?
- Welche Akteure sollen welche Aufgaben übernehmen? ...

Spezifikation

- Definition der Aufgabenallokation, Übergabestrategien, Kooperationsformen, ...

Implementation

- Implementierung der Aufgabenübernahme und -übergabe durch die Maschinen,
- Definition von Prozeduren für die Aufgabenübernahme und -übergabe durch die Operateure, ...

Evaluation

- Abschätzung, ob die Aufgabenübernahme und -übergabe sicher und effizient funktioniert, ...

MMI Interaktion

Anforderungsdefinition

- Welche Informationen sind zu welchem Zeitpunkt für wen relevant? Wieviel muss der Mensch in welcher Situation über die Maschine wissen? Wieviel muss die Maschine in welcher Situation über den Menschen wissen? ...
- Welche Bedienaktionen sind notwendig? ...

Spezifikation

- Definition der zu kommunizierenden Informationen, der Interaktionsmodalitäten (z. B. visuell, akustisch, taktil), der Informationsverteilung, der Bedienstrategien, ...
- Ggf. Definition der Methoden zur Messung des Zustands der Operateure (z. B. Müdigkeits-, Arbeitslastmessung), ...

Implementation

- Implementierung der Informationsbereitstellung und -verteilung,
- Realisierung der spezifizierten Interaktionsmodalitäten, der Zustandsmessung, ...
- Ausarbeitung und Dokumentation der Interaktionsprozeduren, ...

Evaluation

- Abschätzung, ob jeder Akteur die notwendigen Informationen zur richtigen Zeit zur Verfügung hat, ...
-

Tab. 1 (Fortsetzung)

MMT Schnittstelle
<i>Anforderungsdefinition</i> Ergonomische Anforderungen laut EN ISO 9241, Berücksichtigung der menschlichen Informationsverarbeitung, ...
<i>Spezifikation</i> Gestaltung der (z. B. grafischen) Informationsdarstellung, der Bedienelemente, ...
<i>Implementation</i> Implementation der Ausgabe- (z. B. Displays) und Bedienelemente (z. B.) auf Seiten der Maschinen, ...
<i>Evaluation</i> Abschätzung, ob die Schnittstelle eine intuitive, fehlerfreie Mensch–Maschine Interaktion erlaubt, ...

Verwendung normierter sprachlicher Formulierungen) sowie zwischen Maschine und Maschine (z. B. Verwendung von Netzwerkprotokollen). Die vier Entwicklungsgegenstände lassen sich unterteilen in die klassischen Entwicklungsphasen Anforderungsdefinition, Spezifikation, Implementation und Evaluation. Die folgende Tabelle zeigt typische Fragestellungen bzw. Entwicklungsschritte der einzelnen Phasen. Dabei sei darauf hingewiesen, dass die Phasen iterativ durchgeführt werden, wobei in jeder Iteration der Entwicklungsgegenstand (Komposition, Kooperation, Interaktion, Schnittstelle) weiter konkretisiert wird. Es handelt sich somit um ein spiralförmiges Vorgehensmodell.

Ein wichtiger Punkt ist die Einbindung der Operateure, z. B. durch Participatory Design (Yamauchi 2012) und Probandentests, in alle Phasen der Entwicklung. Aber im Falle sicherheitskritischer Anwendungen reicht dies nicht aus. Menschen können beispielsweise ihre Anforderungen und Vorlieben nennen, auf Basis von Erfahrungen Entwurfsideen einbringen und schließlich die Auswirkungen und Akzeptanz geplanter Entwicklungen abschätzen. Aufgrund der oben skizzierten Systemvarianz ist es jedoch höchst unwahrscheinlich, dass dabei alle Anforderungen und Auswirkungen aufgedeckt und bewertet werden können. Des Weiteren ist dieses Vorgehen sehr zeitaufwendig und kann nur wenige Male im Entwicklungsprozess praktiziert werden. Darüber hinaus gibt es bei der Überwachung und Steuerung industrieller Prozesse neben dem Operateur weitere wichtige Quellen zur Ermittlung von Anforderungen und Auswirkungen, beispielsweise den Arbeitskontext bzw. die Arbeitsdomäne, die Arbeitsaufgaben, die menschliche Informationsverarbeitung (Kognition), die menschliche Anthropometrie (physikalische und biologische Charakteristika) sowie relevante Human Factors Standards und Leitfäden. Deshalb soll im Folgenden in Ergänzung zur Einbindung der Operateure eine komplementäre modellbasierte Vorgehensweise vorgeschlagen werden.

Die Phase der MMT Anforderungsdefinition, Spezifikation, Implementation und Evaluation lassen sich durch modellbasierte Vorgehensweisen systematisch unterstützen. Mittlerweile gibt es erprobte und bewährte Techniken und Vorgehensweisen für die Modellierung von Mensch–Maschine Systemen, die sich kombiniert mit Participatory Design und

Probandentests in der Praxis anwenden lassen. Ein modellbasiertes Vorgehen hilft dabei, die notwendigen Anforderungsarten systematisch, nachvollziehbar und eindeutig zu erfassen und auf Vollständigkeit, Widerspruchsfreiheit etc. zu analysieren. Bei der Spezifikation helfen Modelle, Entwurfsentscheidungen zu formalisieren und bereits in frühen Entwicklungsphasen gegeneinander anhand bestimmter (Human Factors) Metriken abzuwägen. Einige Werkzeuge bieten die Möglichkeit, Entwurfsspezifikationen (semi-)automatisch in Implementierungsbausteine zu übersetzen. Schließlich können Modelle angewendet werden, um implementierte Prototypen innerhalb einer Simulationsplattform zu testen. Im Rahmen des oben skizzierten Vorgehensmodells lassen sich unterschiedliche Modelltypen anwenden. In vielen Studien erprobt ist die Modellierung der Aufgaben, der Arbeitsdomäne, der menschlichen Informationsverarbeitung sowie der Funktionalität der Maschinen.

Modellierung der Aufgaben Hierbei werden Aufgaben auf höheren Abstraktionsebenen (z. B. Landen des Flugzeugs) iterativ in Teilaufgaben (z. B. Geschwindigkeit drosseln, Höhe abbauen, auf Kurs fliegen) zerlegt bis auf der untersten Ebene konkrete Aktionen (z. B. Schubhebel positionieren, Landeklappen ausfahren, Fahrwerk ausfahren) erreicht sind. Auf diese Weise entsteht eine Aufgabenhierarchie. Auf jeder Hierarchieebene werden die zeitlichen Abhängigkeiten der jeweiligen Teilaufgaben modelliert. Die Teilzeile und Aktionen innerhalb der Hierarchie können mit Prioritäten (Wichtigkeit/Dringlichkeit des Ziels), Häufigkeiten (Wie oft wird dieses Ziel im Rahmen dieser Aufgabe bearbeitet?) und Fehlerwahrscheinlichkeiten (Wahrscheinlichkeit, dass das Ziel nicht korrekt bearbeitet wird) annotiert werden. Ein Aufgabenmodell ist vergleichbar mit einem Kochrezept, das festlegt, welche Schritte in welcher Reihenfolge zu tun sind, um das Oberziel, das fertige Gericht, zu erreichen. An einigen Stellen im Rezept gibt es alternative Vorgehensweisen, d.h. dasselbe Unterziel kann mit unterschiedlichen Mitteln erfüllt werden. Die Auswahl einer geeigneten Vorgehensweise ist oft an situative Voraussetzungen gebunden. Um diese Entscheidungspunkte zu modellieren, lassen sich in Aufgabenmodelle Wenn-Dann-Regeln verwenden: *Wenn* Situation S_i *Dann* wähle Vorgehensweise V_j . Aufgabenmodelle beinhalten alle Daten bzw. Informationen, die notwendig sind, um die Aufgaben erfolgreich durchzuführen, z. B. Parameter zur Überwachung des Prozesszustands und Informationen, um an Entscheidungspunkten zwischen Handlungsalternativen zu wählen. Diese Informationen sowie die Annotationen der Ziele liefern essentielle Anforderungen an die Gestaltung der MMT Interaktion und der MMT Schnittstellen, z. B. von grafischen Displays, die zu jedem Zeitpunkt der Aufgabendurchführung die notwendigen Informationen in kognitiv adäquater Darstellungsform präsentieren müssen. Darüber hinaus bilden die Aufgabenmodelle eine wichtige Grundlage um Strategien zur dynamischen Aufgabenallokation innerhalb der Gestaltung der MMT Kooperation zu definieren und zu testen. Zur Aufgabenmodellierung gibt es Techniken und Werkzeuge wie beispielsweise GOMS (Card et al. 1983), HTA (Annett 2004; Stanton 2006), CTA (May und Barnard 2004) und PED (Lenk et al. 2012).

Modellierung der Arbeitsdomäne Da es nicht immer für alle nicht alle Situationen möglich ist, Aufgaben klar zu definieren und manchmal insbesondere im Fehlerfall kreative Vorgehensweisen notwendig sind, muss die Arbeitsdomäne klar verstanden sein und an die menschlichen Operateure zielorientiert über die MMT Schnittstellen kommuniziert werden. Die Arbeitsdomäne umfasst den Prozess, der überwacht und gesteuert werden soll, sowie das gesamte MMT, welches die Überwachung und Steuerung durchführt. Modelliert werden hierbei die Prinzipien und Rahmenbedingungen auf denen die Arbeitsdomäne aufgebaut ist, z. B. die physikalischen Gesetze, die ausgenutzt werden, um das Oberziel (den Zweck, z. B. Fliegen von A nach B) des MMTs zu realisieren. Im nächsten Abschnitt wird die Modellierung der Arbeitsdomäne etwas ausführlicher dargestellt.

Modellierung der Menschen Die Anforderungen der Operateure an eine adäquate MMT Kooperation, Interaktion und Schnittstellengestaltung müssen ermittelt, umgesetzt und evaluiert werden. Zu diesem Zweck müssen insbesondere relevante Aspekte der menschlichen Kognition verstanden werden. Hierzu gehören Fähigkeiten und Grenzen der Wahrnehmung, der Entscheidungsfindung, des Multitasking, sowie der mentalen, visuellen und motorischen Arbeitslast. Um diese Aspekte explizit und sogar ablauffähig zu modellieren, eignen sich kognitive Architekturen (Forsythe et al. 2005) wie beispielsweise ACT-R (Anderson et al. 2004), SOAR (Lewis 2001), CASCaS (Lüdtkke et al. 2012) und MIDAS (Corker 2000). Diese Modelle können angewendet werden, um das Verhalten eines Operateurs innerhalb eines MMT, z. B. die Interaktion eines Operateurs mit einer Maschine über eine Mensch–Maschine Schnittstelle, zu simulieren. In der Simulation können unterschiedliche Varianten der Schnittstelle miteinander unter Verwendung geeigneter Human Factors Metriken verglichen werden. Mit heutigen Methoden (und sehr wahrscheinlich auch mit zukünftigen Methoden) ist es (selbstverständlich) nicht möglich, menschliches Verhalten vollständig zu modellieren. Deshalb ist es bei der Auswahl einer kognitiven Architektur wichtig, den Modellierungsfokus und die im Modell vorgenommenen Abstraktionen des menschlichen Verhaltens zu verstehen, um zu entscheiden, welche Architektur für welche Entwicklungsfragestellungen geeignet ist. Um die Simulation zu ermöglichen, muss das Modell in eine Simulationsplattform integriert werden. Zur Anwendung von CASCaS existieren derzeit Simulationsplattformen für Fragestellungen der Entwicklung von Flugzeug- (Lüdtkke et al. 2012), Automobil- (Wortelen et al. 2013) und Schiffsbrückensystemen (Sobiech et al. 2014).

Modellierung der Maschinen Weiter oben wurde beschrieben, dass die Entwicklung der Maschinenfunktionalität meist mit systematischen Ingenieurmethoden durchgeführt. Hierzu gehört in vielen Fällen die Modellierung funktionaler Anforderungen, der Funktionsspezifikation, der Systemarchitektur und der Testfälle. Angewendet werden Modellierungssprachen wie UML oder SysML unter Verwendung von Werkzeugen wie MatLab, RTmaps, SCADE oder IBM Rational. In Kombination mit Menschmodellen lassen sich Maschinenmodelle innerhalb von Simulationsplattformen anwenden, um die Mensch–Maschinen Interaktion in einer geschlossen Input–Output Schleife (closed loop) zu simulieren und zu evaluieren. Auf Basis der Modelle kann diese Evaluation bereits in

frühen Entwicklungsphasen, wenn noch kein implementierter Prototyp vorliegt, durchgeführt werden.

Die Modellierung kann auf unterschiedlichen Ebenen der Formalisierung betrieben werden. Modellierung kann beginnen, indem Entwickler textuell beschreiben, auf welchen Annahmen ihre Entwürfe basieren. Damit werden diese explizit und lassen sich z. B. in moderierten Workshops hinterfragen und plausibilisieren. Ein weitergehender Schritt ist die Verwendung einer formalen Notation, um bestimmte Annahmen klar und widerspruchsfrei zu beschreiben. Auf Basis der formalen Beschreibung lassen sich Berechnungen anstellen und Vorhersagen analytisch ableiten. Schließlich können ausführbare Modelle erstellt werden, die eine Simulation und damit eine detaillierte Analyse in realistischen Testszenarien erlauben. Die Wahl eines geeigneten Formalisierungsgrads sowie der Umfang als auch die Tiefe der Modellierung hängen u. a. von der Kritikalität der Entwicklungsfragestellung, von den Zertifizierungsanforderungen und den zur Verfügung stehenden Ressourcen ab. Bzgl. der Ressourcen ist unbedingt zu beachten, dass der in frühen Entwicklungsphasen investierte Modellierungsaufwand ein Vielfaches an Test- und Änderungsaufwand in späteren Phasen einspart.

Wie oben bereits erörtert, lassen sich die modellbasierten Verfahren komplementär zu anderen Verfahren wie z. B. Probandentests anwenden, um zwischen Entwurfsalternativen zu entscheiden. Es empfiehlt sich, die modellbasierte Analyse vor den Probandentests durchzuführen. Auf diese Weise können bestimmte Entwurfsschwächen bereits vorher eliminiert werden und der Aufwand für die Tests mit Probanden wird wesentlich verringert. Die modellbasierte Simulation ermöglicht darüber hinaus eine weitgehend automatische Analyse einer großen Menge von Testszenarien, die wegen des Aufwands mit Probanden gar nicht möglich wäre. Mittels der Simulation lassen sich die besonders kritische Szenarien identifizieren, die dann im nächsten Schritte mit Operateuren weitergehend untersucht werden müssen. Die Kombination von modellbasierten und Probanden-basierten Verfahren ist unbedingt notwendig, da Modelle immer nur eine Abstraktion der Realität abbilden.

Im Folgenden soll exemplarisch die Modellierung der Arbeitsdomäne vertieft dargestellt werden. Hier liefert die Vorgehensweise des Ecological Interface Design eine mächtige Technik, die in der Praxis noch wenig Verbreitung gefunden hat, obwohl praktische Anwendungen bereits 1983 von den Vätern dieses Verfahrens Vicente und Rasmussen eindrucksvoll aufgezeigt wurden.

Ecological Interface Design

Ecological Interface Design (EID) ist eine Methode zur Modellierung von Arbeitsdomänen (Vicente und Rasmussen 1992; Vicente 2002; Burns und Hajdukiewicz 2004). Bei der Entwicklung von MMTs zur Überwachung und Steuerung industrieller Prozesse ist zunächst der Prozess ein wesentliches Element der Arbeitsdomäne, darüber hinaus gehören aus Sicht der Operateure auch die Automatisierungssysteme dazu. Während mit

Aufgabenmodellen das Verhalten in wohldefinierten Situationen festgelegt wird, werden Arbeitsdomänenmodelle zur Unterstützung der Prozessüberwachung- und Steuerung in unerwarteten Situationen, wie z. B. bei unerwarteten Prozessstörungen, erstellt. In so einem Fall müssen die Operateure ein klares Verständnis des Prozesses haben, um den oder die Fehler zu detektieren, zu diagnostizieren und Reparaturmaßnahmen einzuleiten. Innerhalb des Vorgehensmodells (Tab. 1) eignet sich EID insbesondere, um Anforderungen für die MMT Schnittstellen abzuleiten. Dabei wird folgende Frage adressiert: Welche Informationen über die Arbeitsdomäne brauchen die Operateure und in welcher strukturellen Form sollten diese dargestellt werden? Hierbei wird dem Automatisierungsdilemma durch explizite Unterstützung eines adäquaten Situationsbewusstseins entgegengewirkt. Das Arbeitsdomänenmodell kann darüber hinaus im Rahmen der MMT Komposition als gemeinsame Grundlage für die Entwicklung sowohl der Automatisierungssysteme als auch des Trainingsmaterials für die Operateure verwendet werden. Damit wird ein weiterer wesentlicher Aspekt des Automatisierungsdilemmas adressiert: wenn Operateure die Systeme überwachen sollen (oder besser, innerhalb eines MMTs mit ihnen kooperieren sollen), dann müssen Mensch und Maschine auf demselben Verständnis des Prozesses aufbauen. Die Operateure werden dann die Vorgehensweise der Maschinen besser nachvollziehen können, was eine wesentliche Voraussetzung für die Realisierung eines funktionierenden MMTs ist.

Zur Modellierung der Arbeitsdomäne werden die Rahmenbedingungen und Prinzipien, auf denen der Prozess beruht, identifiziert und unter Verwendung spezifischer Strukturen beschrieben. Rahmenbedingungen und Prinzipien sind beispielsweise Naturgesetze, auf denen das System beruht, funktionale Zusammenhänge und organisatorische oder rechtliche Vorschriften. Das entstehende Modell beschreibt den Prozess aus einer zielorientierten Perspektive und wird von Anfang an so gedacht, dass es für Operateure nachvollziehbar ist und für das Ziel der Detektion und Analyse unerwarteter Ereignisse verwendet werden kann. EID basiert auf der Annahme, dass Operateure ein mentales Modell des Prozesses bilden. Aus der Analyse einer Vielzahl von Problemlöseprotokollen erfahrener Operateure wurde abgeleitet, dass die Struktur des mentalen Modells oft mehrere Abstraktionsebenen umfasst (Rasmussen 1985). Dabei werden auf höheren Abstraktionsebenen generische funktionale Zusammenhänge und auf unteren Ebenen deren konkrete Umsetzung durch physikalische Komponenten abgebildet. Mittels EID kann ein derart hierarchisches mentales Modell explizit modelliert und mittels geeigneter grafischer Darstellungsformen kommuniziert werden.

Im Folgenden wird zunächst die Abstraktionshierarchie detaillierter vorgestellt und anschließend aufgezeigt, wie sie sich nutzen lässt, um den Entwurf der Mensch–Maschine Schnittstelle anzuleiten.

Abstraktionshierarchie als mentales Modell der Arbeitsdomäne Als durchgängiges Beispiel zur Vorstellung der Abstraktionshierarchie wird die grobe Strukturierung eines Flugzeugs verwendet. Der zu überwachende und zu steuernde Prozess ist das Fliegen von Flughafen A zum Flughafen B. Dieser Prozess wird von Maschinenagenten, z. B. Autopilot, Flight Management Systeme, und den beiden Piloten durchgeführt. Im Folgenden

ist mit dem Begriff System das Flugzeug mit den Automatisierungssubsystemen gemeint. Die Abstraktionshierarchie umfasst fünf Stufen:

1. **Funktionaler Zweck:** Dies ist der Zweck, für den das System entwickelt wurde. Der Zweck eines Flugzeugs ist, Passagiere von A nach B zu transportieren. Zusätzlich werden Kriterien definiert, um einzuschätzen, ob das System korrekt arbeitet: Die Passagiere sollen sicher und ohne Zeitverlust transportiert werden.
2. **Abstrakte Funktion:** Dies sind die grundlegenden Gesetze, auf denen das System beruht. In den meisten Fällen handelt es sich um physikalische Gesetze zur Berechnung von Masse, Energie und Informationen. Weitere Gesetze zur Beschreibung der abstrakten Funktion sind organisatorische Prinzipien und rechtliche Rahmenbedingungen. Für das Flugzeug gelten u. a. die physikalischen Gesetze der Bewegung von Masse, der Erzeugung von Kraft und der Energieerhaltung. Beispielsweise geht Energie grundsätzlich nicht verloren, sondern wird lediglich umgewandelt, z. B. von potentieller in kinetische Energie (und umgekehrt).
3. **Generalisierte Funktion:** Dies sind die Grundfunktionen, welche die physikalischen Gesetze der abstrakten Funktion realisieren. Beim Flugzeug sind dies die Grundfunktionen, die genutzt werden, um beim Landen kinetische Energie und Höhe abzubauen, um schließlich sicher auf der Landebahn aufzusetzen. Während die abstrakte Funktion lediglich die Gesetze nennt, wird hier beschrieben, welche Funktionen in welcher Abfolge zur Anwendung kommen. Hier wird also über die gesetzmäßigen Zusammenhänge hinaus der Fluss von Masse, Energie und/oder Informationen berücksichtigt. Das Flugzeug besitzt potentielle Energie, die beim Landeanflug abgebaut werden muss. Das Flugzeug hat Vorrichtungen zur Verringerung der Höhe, zur Verringerung der Geschwindigkeit und zur Aufrechterhaltung des Auftriebs. Diese Vorrichtungen müssen zum richtigen Zeitpunkt adäquat koordiniert werden, sodass potentielle Energie abgebaut wird, ohne dass der Auftrieb verloren geht. Auf der Ebene der generalisierten Funktion wird die zeitliche Abfolge mit den herrschenden Input–Output-Beziehungen beschrieben: Potentielle Energie wird durch Sinken des Flugzeugs abgebaut, dabei steigt die Geschwindigkeit. Der Geschwindigkeitsanstieg muss durch Vorrichtungen zur Drosselung der Treibstoffzufuhr und Veränderung der Luftströmung kompensiert werden; Geschwindigkeitsverlust muss durch Vorrichtungen zur Aufrechterhaltung des Auftriebs kompensiert werden; etc. In einem industriellen Fertigungsprozess wird auf dieser Ebene der Fluss und die Umformung der Rohlinge durch generische Funktionen wie erhitzen, kühlen, formen, schleifen, pressen und walzen beschrieben.
4. **Physikalische Funktion:** Hier werden die physikalischen Systemkomponenten eingeführt und hinsichtlich ihrer Charakteristika beschrieben. Während im Rahmen der generischen Funktion lediglich von Vorrichtungen zur Verringerung der Höhe gesprochen wird, wird hier beschrieben, wie diese Vorrichtungen konkret realisiert sind. Die Verringerung der Höhe wird beispielsweise durch Bewegungen des Höhenruders erreicht. Die physikalische Funktion des Höhenruders besteht in der Rotation nach oben und unten.

5. **Physikalische Form:** Hier werden die physikalischen Systemkomponenten bzgl. ihrer physikalischen Erscheinung beschrieben: Form, Größe, Farbe, Position und Material. Z.B. befindet sich das Höheruder am Heckleitwerk. Welche weiteren Informationen modelliert werden sollen, hängt davon ab, ob sie für das Modellierungsziel – Unterstützung der Detektion und Analyse unerwarteter Ereignisse – relevant sind.

Anders als bei einer klassischen Teil-Ganzes-Hierarchie sind die Ebenen durch eine Ziel-Mittel Relation aufeinander bezogen: der funktionale Zweck wird unter Ausnutzung der physikalischen Gesetze (abstrakte Funktion) realisiert; hierzu müssen diese Gesetze durch bestimmte generalisierte Funktionen in einer bestimmten Reihenfolge zur Anwendung gebracht werden; diese Funktionen werden realisiert durch physikalischen Systemkomponenten, die zunächst bzgl. ihrer physikalischen Funktionen und dann hinsichtlich ihrer physikalischen Eigenschaften, welche die Funktionen hervorbringen, beschrieben werden. Diese Art der Strukturierung soll die Operateure bei der Problemlösung unterstützen. Wenn ein System korrekt funktioniert, sind die Prinzipien und Rahmenbedingungen auf allen Ebenen und die Beziehungen zwischen den Ebenen erfüllt. Im Fehlerfall sind einige verletzt. Eine wesentliche Komponente der Fehlerdetektion besteht in der Identifikation solcher Verletzungen. Die Abstraktionshierarchie erlaubt auf Basis der Ziel-Mittel-Relation ein systematisches „hineinzoomen“ in das System. Ein Fehler zeigt sich meist an der Oberfläche darin, dass ein bestimmter Systemzweck nicht mehr vollständig erfüllt ist. Von dort können auf der nächsten Ebene die physikalischen Gesetze ermittelt werden, welche verletzt sind, weil bestimmte Grundfunktionen nicht korrekt erfüllt werden, was z. B. auf den Ausfall einer physikalischen Funktion und damit auf eine oder mehrere Komponenten zurückzuführen ist. Aufgrund des „hineinzoomens“ von abstrakten zu konkreten Ebenen, müssen lediglich die physikalischen Komponenten betrachtet werden, die zu den nicht mehr erfüllten Grundfunktionen beitragen. Auf diese Weise ermöglicht das Modell ein zielorientiertes und effizientes Problemlösen und damit eine schnellere Einleitung von Reparaturmaßnahmen.

Informationsdarstellung unter Berücksichtigung der menschlichen Informationsverarbeitung Wie lassen sich die innerhalb der Abstraktionshierarchie repräsentierten Zusammenhänge über die MMT Schnittstelle kommunizieren? Diese Frage betrifft die Präsentationsform. Zur Herleitung der Form stützt sich EID auf die drei von Rasmussen definierten Ebenen der Informationsverarbeitung (Rasmussen 1983): Fähigkeits- (Skill), Regel- (Rule) und Wissens- (Knowledge)basierte Informationsverarbeitung. Es handelt sich um eine Taxonomie, welche Informationsverarbeitung anhand der verarbeiteten Informationstypen, der involvierten Wissensstrukturen und der involvierten mentalen Ressourcen klassifiziert.

Die Fähigkeits-basierte Ebene wird aktiviert, wenn die aus der Umgebung wahrgenommenen Informationen als Signale (Signals) innerhalb eines Raum-Zeit Kontinuums interpretiert werden. Dann werden unbewusst motorische Muster angewendet, die in der Vergangenheit gelernt wurden und in „Fleisch und Blut“ übergegangen sind. Dabei handelt es

sich meist um Regulierungsprozesse, bei denen ein räumliches Ziel über die Zeit erreicht und beibehalten werden muss, z. B. das Lenken eines Autos innerhalb der Fahrbahnmarkierungen. Räumliche Abweichungen werden wahrgenommen und es wird regelnd gesteuert. Die verwendeten Wissensstrukturen stellen eine enge Kopplung zwischen Wahrnehmung und Motorik her. Die Regel-basierte Ebene wird aktiviert, wenn die wahrgenommenen Informationen als Zeichen (Signs) interpretiert werden. Die Zeichen weisen darauf hin, dass die aktuelle Situation bereits in der Vergangenheit (mehrmals) erfolgreich bewältigt wurde. Die erfolgreiche Bearbeitung wurde als rezeptartige Handlungsanleitung mental gespeichert und kann nun abgerufen und erneut Schritt für Schritt angewendet werden. Die Abarbeitung der Handlungsanleitung wird zumindest teilweise bewusst durchgeführt, weil es Entscheidungspunkte geben kann, an denen auf Basis der aktuellen Situationsvarianz zwischen mehreren Möglichkeiten gewählt werden muss. Die Wissensbasierte-Ebene wird aktiviert, wenn die aktuelle Situation neu ist und die wahrgenommenen Informationen als Symbole (Symbols) interpretiert werden. Es sind keine fertigen Lösungen mental gespeichert und deshalb muss die Situation analysiert und Lösungen z. B. mittels mentaler Simulation oder durch Analogie erfunden werden. Hierbei wird bewusstes, analytisches Problemlösen auf Basis einer mentalen Repräsentation der Arbeitsumgebung betrieben.

Die Fähigkeits-basierte Ebene erfordert wenig kognitive Ressourcen. Während auf der Wissens-basierten Ebene auf einen reichhaltigen Wissensbackground zurückgegriffen wird. Welche Ebene in einer konkreten Situation aktiviert wird, hängt von den Eigenschaften der durchzuführenden Aufgabe, der Erfahrung des Operateurs und von der Form der Informationen ab. Die Mensch-Maschine Schnittstelle sollte es dem Operateur erlauben, Aufgaben auf einer unteren Informationsverarbeitungsebene zu bearbeiten. Andererseits müssen auch höhere Ebenen unterstützt werden, da die mentale Verarbeitungsebene eben nicht nur von der Informationspräsentation abhängt. Folglich sollte die Schnittstelle so gestaltet werden, dass beim Operateur jeweils die mentale Ebene angesprochen wird, die für die aktuelle Aufgabe unter den gegebenen Kenntnissen und Fähigkeiten der handelnden Person angemessen ist. Folgende Leitlinien für den Schnittstellenentwurf (im Folgenden Displayentwurf) lassen sich aus der Rasmussen Taxonomie ableiten (vgl. Vicente und Rasmussen 1992).

Unterstützung Fähigkeits-basierter Verarbeitung Der Operateur muss die Möglichkeit haben, Objekte auf dem Display zu manipulieren, als würde sie/er direkt mit dem realen Objekt arbeiten. Dabei werden reale Objekte durch das Bewegen angezeigter Icons räumlich und zeitlich bewegt. Die dargestellte Information muss die räumliche und zeitliche Anordnung realer Objekte widerspiegeln, sodass motorische Verhaltensmuster auf der Fähigkeits-basierten Ebene aktiviert werden können. Damit mit zunehmender Übung, Einzelbewegungen zu komplexen Bewegungsabfolgen integriert werden können, muss die angezeigte Struktur isomorph zu Elementarbewegungen und deren Zusammenfassung zu komplexen Handlungsmustern sein. Dies kann durch einen hierarchischen Aufbau des Displays erreicht werden, bei dem angezeigt wird, wie sich Informationen auf höheren Aggregationsebenen durch Einzelinformationen auf niedrigeren Aggregationsebenen zusammensetzen (Flach und Vicente 1989). Auf diese Weise sind unterschiedlich Aggrega-

tionsstufen zur selben Zeit sichtbar und der Operateur kann seine Aufmerksamkeit auf die Ebene richten, die seinem Erfahrungsgrad entspricht.

Unterstützung Regel-basierten Verhaltens Das Display sollte ein konsistentes ein-zu-eins Mapping zwischen angezeigten Zeichen und den Prinzipien und Rahmenbedingungen der Arbeitsdomäne zeigen. D.h. die angezeigten Zeichen müssen zu jedem Zeitpunkt eindeutig den Systemzustand reflektieren. Hier eignet sich in vielen Fällen eine Darstellung des Zustands auf der Ebene der abstrakten oder generalisierten Funktion (siehe z. B. Amelink et al. 2005). Überwachungs- und Steuerungsregeln werden in der Praxis jedoch oft auf einer physikalischen Ebene definiert. Das kann dazu führen, dass die Operateure nicht verstehen, wie diese Regeln mit den Systemfunktionen bzw. mit dem Systemzweck verbunden sind. Deshalb sollten Regeln auf unterschiedlichen Ebenen sowie deren Zusammenhang entlang der Abstraktionshierarchie unterstützt werden. Wichtig ist, dass in Abhängigkeit der gewählten Abstraktionsebenen die Einflussfaktoren vollständig dargestellt werden. Wenn hier relevante Faktoren nicht angezeigt werden, ist es dem Operateur nicht möglich, ein vollständiges und unverfälschtes Bild der Arbeitsdomäne zu gewinnen.

Unterstützung Wissens-basierten Verhaltens Das Display sollte die Zusammenhänge der Arbeitsdomäne so anzeigen, dass sie direkt zur Problemlösung angewendet werden können. Das Display sollte die Abstraktionshierarchie vollständig zur Verfügung stellen (siehe z. B. Burns 2000). Die Schnittstelle sollte das „Hineinzoomen“ in die Arbeitsdomäne durch geeignete Verlinkungsstrukturen zwischen den Abstraktionsebenen unterstützen. Auf diese Weise kann der Operateur zielorientiert die komplexitätsreduzierende Eigenschaft der Hierarchie nutzen, um Fehlerhypothesen zu explorieren.

Diese Richtlinien liefern Grundlagen zur Gestaltung des Inhalts und der Struktur der MMT Schnittstelle auf Basis eines Verständnisses der menschlichen Informationsverarbeitung. EID bietet keine Unterstützung zur grafischen Gestaltung wie der Auswahl der Farben, der Größe der Formen oder grafischer Effekte. Wichtige Anforderungen an diese grafische Gestaltung z. B. im Hinblick auf die Saliénz der angezeigten Informationen lassen sich aus Aufgabenmodellen ableiten, da diese etwas zur Wichtigkeit der Informationen aussagen. Zusammen mit einem grundsätzlichen Verständnis der menschlichen Wahrnehmung lässt sich die Wichtigkeit übersetzen in eine grafische Saliénzgestaltung.

Die Ausführungen in den beiden vorangehenden Abschnitten sollten grundsätzlich aufzeigen, dass modellbasierte Verfahren angewendet werden können, um den Faktor Mensch systematisch bei der Entwicklung von Mensch–Maschine Systemen, insbesondere MMTs, zu berücksichtigen und entsprechende Anforderungen in das System „hinein zu entwickeln“ statt erst zum Schluss „draufzusetzen“.

Zusammenfassung

Der Text stellt das Konzept des Mensch–Maschine Teams als Perspektive für die Automatisierung und möglichen Ausweg aus dem Automatisierungsdilemma vor. Hierbei ist weder das Automatisierungssystem noch der Mensch im Zentrum der Entwicklung, son-

dern das dynamische Zusammenspiel unterstützt durch eine adäquate Mensch–Maschine Kommunikation. Ein modellbasiertes Vorgehen kann helfen, die damit verbundenen Entwicklungs herausforderungen mit Ingenieurmethoden zu bewältigen. Die klassischen Ironien der Automatisierung werden dabei auf folgende Weise adressiert:

- **Ironie 1:** Es wird anerkannt, dass Menschen den Maschinen in bestimmten Aspekten überlegen sind. Aufgaben werden je nach Situation dynamisch auf Mensch und Maschine verteilt. Das Gesamtsystem und insbesondere das Zusammenspiel zwischen Mensch und Maschine wird eingehend durch modellbasierte Verfahren evaluiert, um Entwicklungsfehler früh im Entwicklungsprozess zu finden und zu beheben.
- **Ironie 2:** Aufgaben werden nicht nach Machbarkeit verteilt, sondern unter Berücksichtigung der situativen Stärken und Schwächen der Mitglieder des Menschen–Maschine Teams.
- **Ironie 3:** Die Überwachung der Maschinen wird unterstützt durch Kommunikationsstrategien und entsprechende Mensch–Maschine Schnittstellen, deren Gestaltung sich an der menschlichen Informationsverarbeitung orientiert. Es wird erkannt, dass die dynamische Übergabe von Aufgaben an den Menschen (z. B. in Störfällen) durch explizit entworfene intuitive Übergabestrategien vorbereitet werden muss.
- **Ironie 4:** Den Operateuren wird in definierten Fällen die Möglichkeit eingeräumt, Aufgaben selbst durchzuführen, um einer Erosion manueller Fähigkeiten entgegenzuwirken.

An der Umsetzung eines modellbasierten Vorgehensmodells zur Entwicklung von Mensch–Maschine Teams arbeiten aktuell 31 Industrie- und Forschungspartner aus sieben Ländern in dem durch ARTEMIS (www.artemis-ju.eu) geförderten Projekt HoliDes (Holistic Human Factors and System Design of Adaptive Cooperative Human–Maschine Systems). Adressiert werden die Anwendungsbereiche Luftfahrt, Straßenverkehr, Leitstände und Medizin. Weiterführende Informationen zu dem Thema dieses Textes lassen sich auf der Webseite des Projektes www.holides.eu finden.

Open Access This chapter is distributed under the terms of the Creative Commons Attribution Noncommercial License, which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

Literaturverzeichnis

- Amelink, M. H. J., Mulder, M., van Paassen, M. M., & Flach, J. (2005). Theoretical foundations for a total energy-based perspective flight-path display. *International Journal of Aviation Psychology*, *15*(3), 205–231.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036–1060.
- Annett, J. (2004). Hierarchical task analysis. In D. Diaper & N. Stanton (Hrsg.), *The handbook of task analysis for human–computer interaction* (S. 67–82). Mahwah: Lawrence Erlbaum.

- Bainbridge, L. (1983). Ironies of automation. *Automatica*, 19(6), 775–779.
- Billings, C. E. (1997). *Aviation automation: the search for a human-centered approach*. Mahwah: Lawrence Erlbaum Associates.
- Burns, C. M. (2000). Putting it all together: improving display integration in ecological displays. *Human Factors*, 42, 226–241.
- Burns, C. M., & Hajdukiewicz, J. R. (2004). *Ecological interface design*. Boca Raton: CRC Press.
- Byrne, E. A., & Parasuraman, R. (1996). Psychophysiology and adaptive automation. *Biological Psychology*, 42(3), 249–268.
- Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human–computer interaction*. Hillsdale: Lawrence Erlbaum Associates.
- Christoffersen, K., & Woods, D. D. (2002). How to make automated systems team players. *Advances in Human Performance and Cognitive Engineering Research*, 2, 1–12.
- Corker, K. M. (2000). Cognitive models and control. In N. B. Sarter & R. Amalberti (Hrsg.), *Cognitive engineering in the aviation domain*. Mahwah: LEA.
- Dahlback, N., Jonsson, A., & Ahrenberg, L. (1993). Wizard of Oz studies – why and how. *Knowledge Based Systems*, 6(4), 258–266.
- Degani, A., Shafto, M., & Kirlik, A. (1999). Modes in human–machine systems: review, classification, and application. *International Journal of Aviation Psychology*, 9(2), 125–138.
- Endsley, M. R. (1996). *Automation and situation awareness. Automation and human performance: theory and applications*, S. 163–181.
- Endsley, M. R., & Kiris, E. O. (1995). The out-of-the-loop performance problem and level of control in automation. *Human Factors*, 37, 381–394.
- Fitts, P. M. (Hrsg.) (1951). *Human engineering for an effective air navigation and traffic control system*. Washington: National Research Council.
- Flach, J. M., & Vicente, K. J. (1989). *Complexity, difficulty, direct manipulation and direct perception*. Technical Report EPRL-89-03. Engineering Psychol. Res. Lab., Univ. of Illinois, Urbana-Champaign, IL.
- Forsythe, C., Bernard, M. L., & Goldsmith, T. E. (2005). *Human cognitive models in systems design*. Mahwah: Lawrence Erlbaum Associates.
- Frey, D., & Schulz-Hardt, S. (1997). Eine Theorie der gelernten Sorglosigkeit. In H. Mandl (Hrsg.), *Bericht über den 40. Kongress der Deutschen Gesellschaft für Psychologie* (S. 604–611). Göttingen: Hogrefe Verlag für Psychologie.
- Gigerenzer, G., Todd, P. M., & ABC Res. Group (1999). *Simple heuristics that make us smart*. New York: Oxford Univ. Press.
- Javaux, D. (1998). Explaining Sarter & Woods’ classical results. The cognitive complexity of pilot–autopilot interaction on the Boeing 737-EFIS. In N. Leveson & C. Johnson (Hrsg.), *Proceedings of the 2nd workshop on human error, safety and systems development*.
- Klein, G., Woods, D. D., Bradshaw, J. M., Hoffman, R. R., & Feltovich, P. J. (2004). Ten challenges for making automation a “Team player” in joint human-agent activity. *IEEE Intelligent Systems archive*, 19(6), 91–95.
- Lenk, J. C., Droste, R., Sobiech, C., Lüdtkke, A., & Hahn, A. (2012). In T. Bossomaier & S. Nolfi (Hrsg.), *Proceedings of the fourth international conference on advanced cognitive technologies and applications (COGNITIVE)*. ThinkMind (S. 67–70).
- Lewis, R. L. (2001). Cognitive theory, soar. In *International encyclopedia of the social and behavioral sciences*. Amsterdam: Pergamon.
- Lüdtkke, A., & Möbus, C. (2004). A cognitive pilot model to predict learned carelessness for system design. In A. Pritchett & A. Jackson (Hrsg.), *Proceedings of the international conference on human–computer interaction in aeronautics (HCI-aero)*. Cépaduès-Éditions: France. CD-ROM

- Lüdtke, A., Osterloh, J.-P., & Frische, F. (2012). Multi-criteria evaluation of aircraft cockpit systems by model-based simulation of pilot performance. In *Proceedings international conference on embedded real-time systems and software (ERTS²)*, 1.–3. February 2012, Toulouse, France.
- Mackworth, N. H. (1950). Researches on the measurement of human performance. In H. W. Sinaiko (Hrsg.), *Selected papers on human factors in the design and use of control systems (1961)* (S. 174–331). New York: Dover.
- May, J., & Barnard, P. J. (2004). Cognitive task analysis in interacting cognitive subsystems. In D. Diaper & N. Stanton (Hrsg.), *The handbook of task analysis for human–computer interaction* (S. 291–325). Mahwah: Lawrence Erlbaum.
- Norman, D. A. (1981). Categorization of action slips. *Psychological Review*, 88(1), 1–15.
- Opperman, R. (1994). *Adaptive user support*. Hillsdale: Erlbaum.
- Parasuraman, R., & Manzey, D. H. (2010). Complacency and bias in human use of automation: an attentional integration. *Journal of the Human Factors and Ergonomics Society*, 52, 381–410.
- Parasuraman, R., & Riley, V. (1997). Human and automation: use, misuse, disuse, abuse. *Human Factors*, 39(2), 230–253.
- Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 30(3), 286–297.
- Rasmussen, J. (1983). Skills, rules, knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man and Cybernetics*, 13(3), 257–266.
- Rasmussen, J. (1985). The role of hierarchical knowledge representation in decision making and system management. *IEEE Transactions on Systems, Man and Cybernetics*, 15(2), 234–243.
- Rouse, W. B. (1981). Human-computer interaction in the control of dynamic systems. *ACM Computing Surveys*, 13, 71–99.
- Rüegger, B. (1990). *Menschliches Fehlverhalten im Cockpit*. Schweizerische Rückversicherungs-Gesellschaft, Abteilung Luftfahrt, Mythenquai 50/60, Postfach, CH-8022 Zürich, Schweiz.
- Sarter, N. B., & Woods, D. D. (1995). *Strong, silent and out of the loop: properties of advanced (cockpit) automation and their impact on human–automation interaction*. Cognitive Systems Engineering Laboratory Report, CSEL 95-TR-01. The Ohio State University, Columbus OH, March 1995. Prepared for NASA Ames Research Center.
- Sarter, N. B., Woods, D. D., & Billings, C. (1997). Automation surprises. In G. Salvendy (Hrsg.), *Handbook of human factors/ergonomics* (2. Aufl., S. 1926–1943). New York: Wiley.
- Sheridan, T. B. (1997). Supervisory control. In *Handbook of human factors and ergonomics*. New York: Wiley.
- Sherman, P. J. (1997). *Aircrews' evaluations of flight deck automation training and use: measuring and ameliorating threats to safety*. The University of Texas Aerospace Crew Research Project. Technical Report 97-2, http://www.bluecoat.org/reports/Sherman_97_Aircrews.pdf (Stand 23.04.2014).
- Simon, H. A. (1955). A behavioral model of rational choice. *Journal of Economics*, 69, 99–118.
- Sobiech, C., Eilers, M., Denker, C., Lüdtke, A., Allen, P., Randall, G., & Javaux, D. (2014). Simulation of socio-technical systems for human-centre ship bridge design. In *Proceedings of the international conference on human factors in ship design & operation* (S. 115–122). London: The Royal Institution of Naval Architects.
- Stanton, N. A. (2006). Hierarchical task analysis: development applications and extensions. *Applied Ergonomics*, 37, 55–79.
- Vicente, K. J., & Rasmussen, J. (1992). Ecological interface design: theoretical foundations. *IEEE Transactions on Systems, Man and Cybernetics*, 22, 589–606.
- Vicente, K. J. (2002). Ecological interface design: progress and challenges. *Human Factors*, 44, 62–78.

- Warm, J. S., Dember, W. N., & Hancock, P. A. (1996). Vigilance and workload in automated systems. In R. Parasuraman & M. Mouloua (Hrsg.), *Automation and human performance: theory and applications* (S. 183–200). Mahwah: Lawrence Erlbaum.
- Wiener, E. L., & Curry, R. E. (1980). Flight-deck automation: promises and problems. *Ergonomics*, 23(10), 995–1011.
- Woods, D. D., & Roth, E. M. (1988). Cognitive engineering: human problem solving with tools. *Human Factors*, 30(4), 415–430.
- Wortelen, B., Baumann, M., & Lüdtké, A. (2013). Dynamic simulation and prediction of drivers' attention distribution. *Journal of Transportation Research Part F: Traffic Psychology and Behaviour*, 21, 278–294.
- Yamauchi, Y. (2012). Participatory design. In T. Ishida (Hrsg.), *Field informatics* (S. 123–138). Welwyn: Springer.
- Zsombok, C. E., Beach, L. R., & Klein, G. (1992). A literature review of analytical and naturalistic decision making. In *Contract N66001-90-C-6023 for the Naval command, control and ocean surveillance center*. San Diego: Klein Associates Inc.