

# REKONSTRUKTION VON VERBORGENEM VERHALTEN BEI EINER PRODUKTIONSLINIE

Claudia Krull, Robert Buchholz, Thomas Strigl, Graham Horton

## 1 Einführung

Wir stellen einen neuen Modellierungsansatz vor, der aus Protokoll Daten einer Produktionsanlage verborgenes Verhalten rekonstruieren kann. Dieser Ansatz ermöglicht es, wahrscheinliche Ursachen für Systemzustände zu ermitteln, selbst wenn dafür keine Messdaten vorliegen. In modernen Fabrikanlagen werden an verschiedenen Stellen des Materialflusses Protokolle beispielsweise der Produktqualität oder der Maschinenparameter aufgenommen und in einer Datenbank gespeichert. Diese Daten werden u.a. dafür genutzt, um Abweichungen von vordefinierten Produktionsparametern zu erkennen und einen Alarm auszulösen. Typische Protokolle reichen jedoch nicht aus, um die Ursache solcher Störungen zu ermitteln, da die Daten nur punktuell erhoben werden. Wir haben ein neues Modellierungsparadigma entwickelt, mit dem man unserem Wissen nach erstmals nichtbeobachtbares

Systemverhalten von allgemeinen stochastischen Modellen aus deren stochastisch erzeugten Ausgaben errechnen kann. Das Paradigma basiert auf den so genannten Hidden non-Markovian Models, (HnMM). Diese Modelle können mit einer zustandsraumbasierten Methode – den so genannten Proxels – effektiv simuliert werden. Durch die Kombination von HnMM und Proxels wird es möglich, die Systemabläufe zu identifizieren, die ein gegebenes Protokoll erzeugt haben, und es ist ferner möglich, für jeden dieser möglichen Abläufe dessen Wahrscheinlichkeit zu berechnen.

Die vorgestellte Methode kann dazu genutzt werden, aus partiellen oder sogar verrauschten Produktionsdaten nicht beobachtetes Systemverhalten zu ermitteln. Dadurch kann der Nutzen von Daten gesteigert werden, die in vielen Produktionsumgebungen bereits erhoben werden. Es wäre beispielsweise möglich, den derzeitigen Zustand einer Maschine oder eines Bauteils zu schätzen und somit rechtzeitig Wartungen zu veranlassen oder die Ursachen für Fehlzustände in einer Produktionslinie zu ermitteln, die sich stromaufwärts befinden und nicht der Überwachung unterliegen. Durch kontinuierliche Auswertung der Produktionsprotokolle wäre es zudem möglich, im Betrieb ständig uneinsehbare Anlagenzustände zu kennen, und somit den Fortgang und die Qualität der Produktion zu sichern ohne zusätzliche Sensoren installieren zu müssen.

## 2 Anwendungsbeispiel: Protokollauswertung eines Qualitätstests in der Solarzellenfertigung

Wir demonstrieren unseren Ansatz an Hand eines einfachen Beispiels, das einen Ausschnitt aus der Solarzellenfertigung beschreibt. In diesem Modell treffen Silizium-Wafer von zwei unterschiedlichen Lieferanten (oder Maschinen im vorhergehenden Prozess) kontinuierlich ein. Die Wafer werden in einen Materialfluss zusammengeführt und durch ein Testgerät geleitet. Im Testprotokoll werden die Qualität und der Zeitpunkt jedes Wafers vermerkt, nicht jedoch dessen Lieferquelle.

Aus diesem Testprotokoll können wir rekonstruieren, welcher der beiden Lieferanten mit welcher Wahrscheinlichkeit den höheren Anteil an fehlerhaften Wafern geliefert hat. Das Modell wurde in Kooperation mit der Industrie (Dr. Thomas Strigl, iSILOG GmbH) spezifiziert und auch als Beispiel verwendet in Buchholz 10.

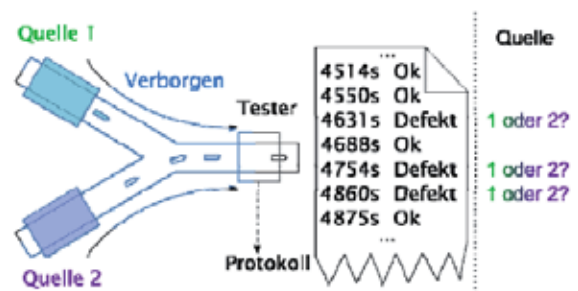


Abbildung 1: Systemaufbau des Beispielmodells Tester aus der Solarzellenfertigung

Abbildung 1 zeigt ein Schema des zu untersuchenden Systems und der resultierenden Fragestellung. Bei dem verborgenen Modell handelt es sich um ein diskretes stochastisches Modell. Die Wafer von Quelle 1 (Quelle 2) haben eine gewisse Wahrscheinlichkeit  $p_1$  ( $p_2$ ), fehlerhaft zu sein. Ausgehend vom besten Fall sind die Wafer beider Quellen von gleicher und hoher Qualität. Die Abstände zwischen den einzelnen eintreffenden Wafern sind beschrieben durch unterschiedliche nicht-Markovsche zeitkontinuierliche Verteilungsfunktionen, daher lässt sich die Quelle eines bestimmten Wafers nicht trivial aus dessen Testzeitpunkt ermitteln.

Ein Beispiel für ein Protokoll ist ebenfalls in Abbildung 1 zu sehen. Dabei ist nicht klar, aus welcher Quelle ein fehlerhafter Wafer stammt. Anhand eines solchen Protokolls können wir dennoch für jeden Wafer die Wahrscheinlichkeit feststellen, mit dem er von jeder der beiden Quellen erzeugt wurde. Daraus können wir

ermitteln, welche Quelle mehr defekte Wafer erzeugt hat, und daher welcher Lieferant für einen möglicherweise ungewöhnlich hohen Ausschussgrad verantwortlich ist. Daraus können dann Konsequenzen bezüglich der Lieferanten gezogen oder Anhaltspunkte für ein Problem in einem der beiden vorangegangenen Produktionsprozesse gefunden werden.

### 3 Stand der Forschung

#### 3.1 Verborgene nicht-Markovsche Modelle

Verborgene nicht-Markovsche Modelle wurden entwickelt, um die Einschränkungen von klassischen verborgenen Markov-Modellen (Rabiner 89, Fink 08) zu überwinden.

Verborgene Markov-Modelle enthalten ausschließlich eine zeitdiskrete Markovkette (Bolch 98) als verborgenes Modell, was die korrekte Abbildung realer zeitabhängiger Systeme stark erschwert. In (Krull 09) wurden diskrete stochastische Modelle als verborgene Komponente eingeführt. Dadurch ist es nun möglich zeitabhängiges Verhalten in verborgenen Systemen adäquat abzubilden. Der Fokus der Analyse verschiebt sich hierbei weg von den Systemzuständen und hin zu Zustandsübergängen. Ein verborgenes nicht-Markovsches Modell kann als 6-Tupel  $(S, C, V, A, B, \pi)$  beschrieben werden, mit der Menge der Zustände  $S$ , der Menge der Transitionen  $C$ , der Menge der Ausgabesymbole  $V$ , der zeitabhängigen Zustandsübergangsmatrix  $A(t)$ , der Emissionswahrscheinlichkeitsmatrix  $B$  und dem initialen Wahrscheinlichkeitsvektor  $\pi$ . Die zeitabhängigen Zustandsübergänge können durch beliebige kontinuierliche Verteilungsfunktionen beschrieben werden.

Die beobachtbaren Symbolausgaben eines solchen verborgenen Modells („Spur“) müssen nun auch mit Zeitstempeln versehen sein. Analog kann das - nicht beobachtbare - Systemverhalten eines verborgenen nicht-Markovschen Modells durch eine Sequenz von Zustandsübergängen mit dazugehörigen Zeitstempeln beschrieben werden. Dieser Pfad kann nun allerdings länger sein als die dadurch erzeugte Spur, da nicht jeder Zustandsübergang notwendigerweise eine Symbolausgabe erzeugen muss.

#### 3.2 Zustandsraumbasierte Analyseverfahren

Die zustandsraumbasierte Simulation von diskreten stochastischen Modellen stellt eine mittlerweile etablierte Alternative zur klassischen ereignisorientierten Simulation (DES) dar. Zustandsraumbasierte Simulation arbeitet deterministisch und eliminiert damit die Nachteile der inhärenten stochastischen Unsicherheit, wie sie bei DES auftritt. Insbesondere kleine, steife Simulationsprobleme können mit zustandsraumbasierten Verfahren sehr effizient gelöst werden.

Dabei wird der gesamte Zustandsraum des Modells untersucht. Dies kann im einfachsten Fall durch das Erstellen und Lösen einer Markov-Kette geschehen, was allerdings wieder erhebliche Restriktionen im Modell mit sich bringt. Das vor wenigen Jahren entwickelte Proxelbasierte Simulationsverfahren stellt eine wesentlich flexiblere Ausprägung zustandsraumbasierter Verfahren dar (Horton 02, Lazarova-Molnar 05). Das Verfahren ist in der Lage allgemeine diskrete stochastische Modelle ohne Restriktionen in der Modellstruktur zu analysieren. Lediglich die Größe der Modelle ist beschränkt, was aber in der Natur zustandsraumbasierter Verfahren liegt. Das entwickelte Verfahren wurde bereits erfolgreich auf Warteschlangenanalyse angepasst (Krull 10). HMM Analyseverfahren arbeiten auf dem Zustandsraum eines realen Modells in Form einer zeitdiskreten Markov-Kette. Durch diese gemeinsame Grundlage sehen wir hier eine Ansatzmöglichkeit unsere Kenntnisse auf dem Gebiet der zustandsraumbasierten Analyse diskreter stochastischer Modelle zur Erweiterung von HMM auf allgemeinere Modelle zu nutzen.

#### 3.3 Analyse verborgener nicht-Markovscher Modelle

Zu verborgenen Markov-Modelle existieren drei unterschiedliche Fragestellungen: Das Evaluierungsproblem fragt nach der absoluten Wahrscheinlichkeit, mit der eine bestimmte Spur von einem bestimmten System erzeugt wurde. Der Forward Algorithmus löst dieses Problem effizient für verborgene Markovsche Modelle. Das Dekodierungsproblem fragt nach dem wahrscheinlichsten Pfad, der zur Erzeugung einer bestimmten Spur von einem bestimmten System genommen wurde. Der Viterbi-Algorithmus löst diese Fragestellung für verborgene Markovsche Modelle. Das Trainingsproblem versucht aus einer gegebenen Spur die wahrscheinlichste Systemkonfiguration zu ermitteln, die diese Spur erzeugt hat. Die Verfahren zur Lösung dieser Aufgabe sind Optimierungsalgorithmen wie der Baum-Welch Algorithmus. (Viterbi 69, Rabiner 89, Fink 08) Das zeitdiskrete Proxel-basierte Verfahren wurde in vorangegangenen Experimenten so modifiziert, dass das Evaluierungs- und das Dekodierungsproblem auch für nicht-Markovsche Modelle gelöst werden können, bei denen die Ausgabe der Symbole nicht mehr von den Zuständen abhängig ist, sondern bei Zustandswechseln erfolgt (Krull 08). Problematisch ist bisher insbesondere die Zustandsraumexplosion bei der Verwendung von Zusatzvariablen (Lazarova-Molnar 05). Gegenwärtig ist dieser Ansatz daher nur für sehr kleine Modelle praktikabel. Andererseits kann bei der Bestimmung der wahrscheinlichsten erzeugenden Zustandssequenz ohne signifikanten zusätzlichen Rechenaufwand eine Rangliste der Pfade ermittelt werden, wohingegen der originale Viterbi-Algorithmus (Fink 08) nur einen einzelnen Pfad

ermitteln kann. Hier wird ein neuerer Ansatz benutzt, der die Proxelbasierte Simulation zur Bestimmung des wahrscheinlichsten Systemverhaltens aufgrund aufgezeichneter Pfade (Buchholz 10) einsetzt.

#### **4 Rekonstruktion von Systemverhalten mithilfe von verborgenen Markov-Modellen und Proxels**

Das Proxel-basierte Simulationsverfahren verfolgt alle möglichen Systementwicklungen in diskreten Zeitschritten.

Dabei enthält ein Proxel (Probability Element) Informationen zum diskreten Systemzustand, das Alter aller derzeit aktiven oder anderweitig relevanten Zustandsübergänge und die Wahrscheinlichkeit dieser Kombination. Für jeden dieser Proxel kann nun bestimmt werden, mit welcher Wahrscheinlichkeit innerhalb des nächsten Zeitschritts in jeden möglichen Folgezustand gewechselt wird, und wie wahrscheinlich es ist, im selben diskreten Zustand zu verharren. Entsprechend werden die Folgeproxel für den nächsten Zeitschritt erzeugt und iterativ abgearbeitet. (Horton 02, Lazarova-Molnar 05)

Die Rekonstruktion von nicht beobachtbarem Systemverhalten verlangt die Erweiterung der Information eines Proxels. Es muss erkennbar sein, welche Ketten von möglichem Systemverhalten zum derzeitigen Zustand geführt haben. Dieses Systemverhalten kann die komplette Sequenz der bisher aufgetretenen Zustandsübergänge sein. Dadurch steigt aber der Speicheraufwand extrem, weil es nun nicht mehr möglich ist, verschiedenen Proxel zusammenzufassen. Dies ist aber essentiell für die praktische Anwendbarkeit des Verfahrens. Je nach Anwendungsfall kann die Information über vergangenes Systemverhalten aber, wie in Abschnitt 4.1 beschrieben, reduziert werden.

Ein Vorteil der Anwendung von zustandsraumbasierten Verfahren auf die Analyse von verborgenen nicht-Markovschen Modellen ist, dass nicht alle möglichen Entwicklungspfade verfolgt werden müssen. Pfade, die ein Systemverhalten repräsentieren, welches die zu analysierende Sequenz nicht erzeugt haben kann, müssen nicht weiter verfolgt werden.

Die Proxel zum Ende der Simulation repräsentieren zu diesem Zeitpunkt möglicherweise erreichbare Systemzustände und die zugehörige Wahrscheinlichkeit. Allerdings sind dies nur noch die Endpunkte der Pfade, die eine bestimmte beobachtete Sequenz erzeugt haben können. Je nach Anwendungsgebiet enthalten diese Proxel darüber hinaus auch Informationen über das Systemverhalten, das zu dem jeweiligen Zustand geführt hat. Wahrscheinliches Systemverhalten, das eine gegebene Beobachtung erzeugt hat, kann nun aus dieser Menge von Pfaden ermittelt werden, um dann Konsequenzen zu ziehen oder mögliche Problemursachen

gezielt zu untersuchen.

Mithilfe eines so modifizierten Proxel-Algorithmus ist es also möglich, wahrscheinliches Systemverhalten zu ermitteln, das eine bestimmte gegebene Beobachtung erzeugt hat.

#### **4.1 Anpassung auf Anwendungsbeispiel: Qualitätskontrolle in der Solarzellenfertigung**

Um das Analyseverfahren für eine bestimmte Anwendung praktikabel zu machen, müssen jeweils spezifische Anpassungen vorgenommen werden. Meist ist beispielsweise nicht die komplette Systementwicklung, die zu einem gegebenen Protokoll geführt, hat von Interesse.

Dann kann die mitzuführende Information auf das relevante Verhalten reduziert werden, was enorm Speicherplatz und Rechenzeit einsparen kann.

Im gegebenen Beispiel des Solarzellentesters ist es nicht von Interesse, welcher Wafer von welcher Quelle produziert wurde. Dies würde bei  $n$  Wafers zu  $2n$  möglichen Systementwicklungen führen, was schon bei Protokollängen ab 30 Einträgen nicht mehr praktikabel ist (In unserem Beispiel werden Protokolle bis zu einer Längenvon 3000 Einträgen analysiert). Allerdings ist für dieses Beispiel tatsächlich auch nur die Anzahl der defekten Wafer interessant, die jede Quelle produziert hat, nicht die Quelle jedes einzelnen Defektes.

Diese Information reicht aus, um den Verursacher einer erhöhten Defektanzahl zu ermitteln. Dadurch reduziert sich die Information über das bisherige Systemverhalten auf eine Zahl, die angibt, wie viele defekte Wafer von Quelle 1 produziert wurden (Da die Gesamtanzahl der Defekte aus dem Protokoll bekannt ist, muss die Anzahl Defekte von Quelle 2 nicht mitgeführt werden (Buchholz 10)). Das führt zu maximal  $n$  möglichen Systementwicklungen, eher weniger, da nur eine geringe Anzahl von Wafers tatsächlich defekt ist.

Das Ergebnis eines Simulationslaufes ist die Wahrscheinlichkeit jeder möglichen Defektanzahl für jede der beiden Quellen. Daraus lassen sich die Verteilung und eine mittlere Defektanzahl berechnen. Diese Defektanzahl kann in Verbindung mit dem Durchsatz der beiden Quellen zur Defektwahrscheinlichkeit umgerechnet werden. Ein solches beispielhaftes Ergebnis und weitere Experimente zum Verfahren sind im nächsten Abschnitt beschrieben.

### **5 Experimente**

Die Experimente in diesem Abschnitt basieren auf dem in Abschnitt 2 beschriebenen Anwendungsbeispiel. Ein Qualitätstester in der Solarzellenfertigung untersucht einen Materialfluss, der tatsächlich eine Zusammenführung zweier Quellen ist. In Abbildung 2 ist das System als verborgenes nicht-Markovsches Modell dargestellt. Dabei ist das verborgene

Modell ein stochastisches Petri Netz (Bobbio 98) mit nur einem Zustand und 2 Transitionen, die unabhängig voneinander in zufällig verteilten Abständen feuern. Dabei stellt jedes Feuern einer Transition den Test eines Wafers aus der jeweiligen Quelle dar. Die Abstände der Wafer aus den Quellen unterliegen einer Normalverteilung. Quelle 1 hat den Mittelwert 120 Sekunden (Standardabweichung 20 Sekunden), Quelle 2 hat Mittelwert 150 Sekunden (Standardabweichung 25 Sekunden). Jeder produzierte Wafer hat eine bestimmte Defektwahrscheinlichkeit, dargestellt durch die Pfeile über die Grenze des nichtbeobachtbaren Systems (gestrichelte Linie) hinausreichen. Zur Erzeugung der Testprotokolle wurde die Defektwahrscheinlichkeit von Quelle 1 auf 0.05 gesetzt und die von Quelle 2 auf 0.1. Die Testprotokolle wurden mithilfe einer diskreten ereignisgesteuerten Simulation mit AnyLogic 6.4 (Borshchev 07) erzeugt. Ziel der Analyse ist es nun, Quelle 2 als Verursacher der zu hohen Defektwahrscheinlichkeit zu identifizieren.

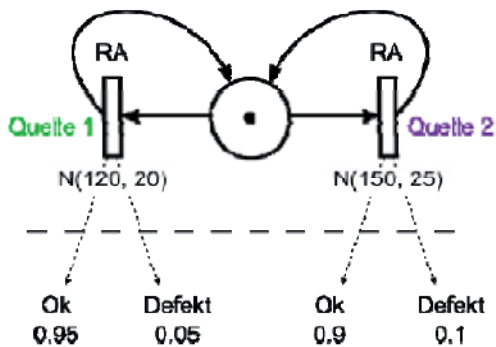


Abbildung 2: Verborgenes nicht-Markovsches Modell des Solarzellentesters zur Erzeugung der Testprotokolle

### 5.1 Funktionstest des Analyseverfahrens

Im ersten Experiment wurde ein Protokoll mit der Länge von 30 Stunden (etwa 1600 Einträge) analysiert. Abbildung 3 zeigt eine Zusammenfassung des Simulationsergebnisses. Für jede der beiden Quellen konnte eine Verteilung über die möglichen Defektwahrscheinlichkeiten ermittelt werden (dargestellt durch ein Histogramm der möglichen Ausprägungen). Die mittlere Defektwahrscheinlichkeit für Quelle 2 liegt bei etwa dem doppelten von Quelle 1 und die beiden Verteilungen überlappen sich fast nicht.

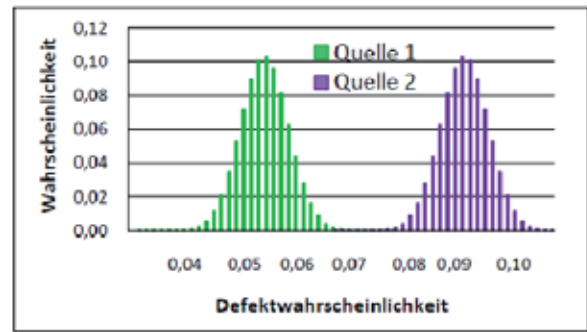


Abbildung 3: Verteilung der Ermittelten Defektwahrscheinlichkeiten für die beiden Quellen

Weiterhin liegen die mittleren Werte in Tabelle 1 nah an den zur Erzeugung des Protokolls genutzten Defektwahrscheinlichkeiten. Die Abweichung ist hier nicht auf eine Ungenauigkeit im Analysealgorithmus zurückzuführen, sondern hat zweierlei Ursachen: Erstens liegt die Defektwahrscheinlichkeit in einem tatsächlichen Protokoll nie genau beim vorgegebenen theoretischen Wert. Zweitens gibt es im Modell inhärente Ungewissheiten, dadurch dass beide Quellen mit stochastischen Zwischenankunftszeiten arbeiten. Durch den höheren Durchsatz von Quelle 1, wird dort die Defektwahrscheinlichkeit grundsätzlich etwas überschätzt, und bei der langsameren Quelle 2 etwas unterschätzt.

	Zielwert	Ergebnis
Quelle 1	0.05	0.050865672
Quelle 2	0.1	0.090584577

Tabelle 1: Erwartete und ermittelte mittlere Defektwahrscheinlichkeit

Im ersten Experiment konnte Quelle 2 also zuverlässig und eindeutig als Verursacher der zu hohen Defektwahrscheinlichkeit ermittelt werden.

### 5.2 Einfluss der Eingabelänge auf Analysequalität

Im zweiten Experiment wurde untersucht, wie sich die Protokolllänge auf die Ergebnisqualität auswirkt. Abbildung 4 zeigt die ermittelte mittlere Defektwahrscheinlichkeit der beiden Quellen bei Protokollängen von 150 bis 3000 Einträgen (ca. 55h Produktionszeit). Die durchgezogenen Linien markieren die tatsächliche Defektwahrscheinlichkeit der beiden Quellen im erzeugenden Modell. Es ist ersichtlich, dass mit steigender Protokolllänge die ermittelten Defektwahrscheinlichkeiten sich den erwarteten annähern. Die absoluten Abweichungen sind wiederum mit inhärenten Ungewissheiten im Modell erklärbar. Weiterhin ist zu erkennen, dass schon bei geringen Protokollängen Quelle 2 eindeutig als Verursacher der zu hohen Defektwahrscheinlichkeit identifiziert werden kann.

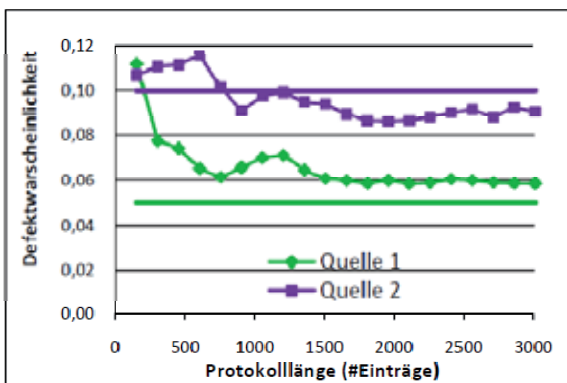


Abbildung 4: Entwicklung der ermittelten Defektwahrscheinlichkeit mit steigender Protokolllänge

### 5.3 Einfluss von Rauschen auf Analysequalität

Im dritten Experiment sollte untersucht werden, wie sich Rauschen in den Daten auf die Ergebnisqualität, auswirkt. Dazu wurden in einem Nachbearbeitungsschritt die Zeitstempel der Protokolleinträge verändert, durch die Addition gleichverteilter Zufallswerte mit gegebener maximaler Abweichung. Das Ergebnis in Abbildung 5 zeigt die ermittelte Defektwahrscheinlichkeit für beide Quellen (ca. 1600 Einträge im Protokoll) bei zunehmendem Rauschen. Die durchgezogenen Linien stellen wieder den tatsächlichen Wert im untersuchten Protokoll dar (Quelle 1 = 0.037, Quelle 2 = 0.094). Selbst bei nicht vorhandenem Rauschen zeigt sich die bereits erwähnte Abweichung vom erwarteten Ergebnis, verursacht durch inhärente Ungenauigkeiten. Es zeigt sich, dass erst bei maximalen Abweichungen von +/- 20 Sekunden für jeden Zeitstempel die Ergebnisqualität merklich abnimmt. Dabei muss beachtet werden, dass Abweichungen von bis zu +/- 20 Sekunden schon erheblich sind, da die Mittelwerte der Wahrscheinlichkeitsverteilungen nur um 30 Sekunden voneinander abweichen und die Standardabweichungen die gleiche Größenordnung haben. Erst ab einer maximalen Abweichung +/- von 40 Sekunden nimmt die Ergebnisqualität stark ab und das Analyseergebnis wird unbrauchbar. Bis dahin wurde immer noch Quelle 2 als Verursacher der erhöhten Defektanzahl identifiziert.

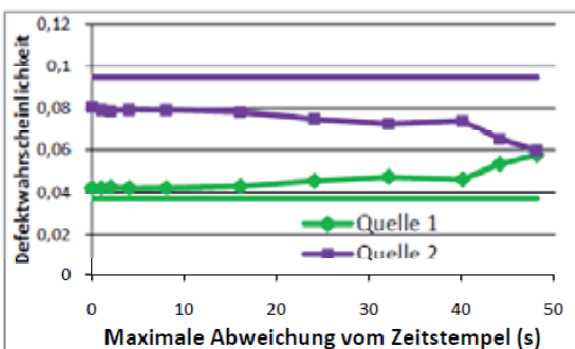


Abbildung 5: Entwicklung der ermittelten Defektwahrscheinlichkeit mit zunehmendem Rauschen

Das zustandsraumbasierte Analyseverfahren zeigt sich in

diesem Experiment erstaunlich robust gegenüber verrauschten Daten. Das stellt eine wichtige Eigenschaft in der Praxis dar, da dort immer mit ungenauen Messungen, ungenauen Abbildungen der Systemparameter oder anderen Abweichungen vom idealisierten Systemverhalten zu rechnen ist.

### 5.4 Einfluss der Eingabelänge auf Laufzeitverhalten

Im letzten Experiment wurde noch der Berechnungsaufwand des Verfahrens untersucht, da der praktische Einsatz sehr stark von den Laufzeiten der Analyse und der Größe der praktikabel zu untersuchenden Modelle abhängt.

Abbildung 6 zeigt die Entwicklung der Rechenzeit in Sekunden der Analyse mit steigender Protokolllänge. Dabei wurden die Berechnungen auf einer Core2 Duo CPU mit 2 GHz und 2 GB RAM ausgeführt, wobei das Programm nur einen Rechnerkern nutzte. Die Abbildung zeigt einen exponentiellen Anstieg der Rechenzeit, wobei zu beachten ist, dass selbst Protokolle mit bis zu 3000 Einträgen (55h Produktionszeit) in unter 3 Minuten analysiert werden konnten. Daher ist das Verfahren für das hier untersuchte kleine Beispielmmodell praktikabel. Dieses gute Laufzeitverhalten des Verfahrens ist jedoch wichtig, da in der Realität die Modelle von größerer Komplexität sein werden.

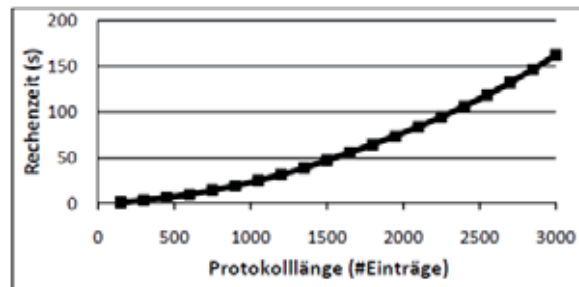


Abbildung 6: Entwicklung der Rechenzeit mit steigender Protokolllänge

### 5.5 Fazit der Experimente

Die Experimente zeigen, dass die Analyse des verborgenen nicht-Markovschen Modells des Qualitätstesters verlässlich den Verursacher einer erhöhten Defektanzahl liefert. Das Verfahren zeigt sich robust gegenüber Rauschen und benötigt moderate Rechenzeit. Die Genauigkeit der Ergebnisse scheint nur durch die inhärente Unsicherheit im Modell begrenzt zu werden. Dies ist jedoch noch nachzuweisen.

## 6 Fazit

Die Arbeit stellt einen neuen Modellierungsansatz vor, der aus Protokolldaten einer Produktionsanlage verborgenes Verhalten rekonstruieren kann. Hier wird ein Beispiel aus der Solarzellenfertigung genommen, wo anhand eines Testprotokolls auf den Verursacher einer erhöhten Defektanzahl geschlossen wird. Die Experimente zeigen, dass das Verfahren funktioniert, und lassen vermuten, dass es auch effizient reale Probleme lösen kann. Eine Einschränkung des vorgestellten Verfahrens ist, dass es nur auf kleine Systeme effizient anwendbar ist, und dass ein tatsächlicher Praxistest noch aussteht. Allerdings existieren weitere Ansätze zur Analyse spezieller Klassen verborgener nicht-Markovscher Modelle, die wesentlich geringere Laufzeiten versprechen. Mit diesen Verfahren ist auch eine Analyse größerer Systeme möglich. Zukünftig ermöglichen verborgene nicht-Markovsche Modelle die Rekonstruktion nicht beobachteten Systemverhaltens anhand von partiellen oder verrauschten Produktionsdaten. Dadurch kann man eine Wertsteigerung bereits erhobener Daten erreichen, und bisher verborgene dynamische Prozesse einer Analyse unterziehen.

Praktikable Analyseverfahren für verborgene nicht-Markovsche Modelle werden die Lösung von neuen Fragestellungen ermöglichen.

Potentielle Anwendungsszenarien in der Produktion sind die folgenden: Die Analyse von Produktionslinien basierend auf Beobachtungen durch externe Sensoren ist eine Alternative zu teurer interner Sensortechnik, und damit eine praktikable Lösung für kleinere Betriebe mit einfachen Fragestellungen. Außerdem können im laufenden Betrieb größerer Anlagen Produktionsprotokolle analysiert werden, und zur Überwachung aktueller Maschinenparameter und somit zur Qualitätssicherung genutzt werden. Effiziente Verfahren zur Analyse verborgener Modelle würden eine kontinuierliche Auswertung existierender Information ohne Unterbrechung der Produktion ermöglichen.

## 7 Literatur

Bobbio, Andrea, Puliafito, Antonio, Telek, Miklós, Trivedi, Kishor S., 1998. Recent Developments in Non-Markovian Stochastic Petri Nets. *Journal of Systems Circuits and Computers*, 8(1):119-158.

Bolch, Gunter, Greiner, Stefan, de Meer, Hermann, Trivedi, Kishor S., 1998, *Queueing Networks and Markov Chains*. John Wiley & Sons, New York.

Borshchev, Andrei, 2007, Multi-method Simulation Modeling using AnyLogic, INFORMS Roundtable Fall Meeting, Seattle, [www.anylogic.com](http://www.anylogic.com) XJ Technologies.

Buchholz, Robert, Krull, Claudia, Strigl, Thomas, Horton, Graham, 2010, Using Hidden non-Markovian Models to Reconstruct System Behavior in Partially-Observable Systems, in 3rd International ICST Conference on Simulation Tools and Techniques.

Fink, Gernot A., 2008, *Markov Models for Pattern Recognition*, Springer Berlin Heidelberg.

Horton, Graham, 2002, A New Paradigm for the Numerical Simulation of Stochastic Petri Nets with General FiringTimes. In *European Simulation Symposium*, Dresden, Germany. SCS European Publishing House.

Krull, Claudia, Horton, Graham, 2008, The Effect of Rare Events on the Evaluation and Decoding of Hidden non-Markovian Models. In *Proceedings of the 7th International Workshop on Rare Event Simulation (RESIM 2008)*, Rennes, France.

Krull, Claudia, Horton, Graham, 2009, Hidden non-Markovian Models: Formalization and Solution Approaches. In *Proceedings of 6th Vienna International Conference on Mathematical Modelling*, Vienna, Austria.

Krull, Claudia, Xu, Wenjing, Horton, Graham, 2010, Proxelbased simulation of queuing systems with attributed customers, *Int. J. Simulation and Process Modelling*, Vol. 6, No. 1.

Lazarova-Molnar, Sanja, 2005, *The Proxel-Based Method: Formalisation, Analysis and Applications*. PhD thesis, Otto-von-Guericke Universität Magdeburg.

Rabiner, Lawrence R., 1989, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257-286.

Viterbi, Andrew J., 1967, Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory*, IT-13:260-269