

FAKULTÄT MASCHINENBAU

Fachgebiet für IT in Produktion und Logistik

Prof. Dr.-Ing. Markus Rabe

Masterarbeit

Untersuchung von Validierungsmöglichkeiten für modulare, auf objektorientierten Petri-Netzen basierende Simulationsmodelle

von

Muzi Dong

Matrikel-Nr.: 174112

Studiengang: Wirtschaftsingenieurwesen

Prüfer: Prof. Dr-Ing. Markus Rabe

Betreuer: Dipl.-Geoinf. Maik Deininger

Ausgegeben am 07.06.2016

Eingereicht am 22.11.2016

Inhaltsverzeichnis

1.	Einleitung.....	1
2.	Ereignisdiskrete Simulation von Petri-Netzen	3
2.1	Ereignisdiskrete Simulation	4
2.2	Petri-Netze	6
2.2.1	Klassisches Petri-Netz.....	6
2.2.2	Systemnetze	10
2.2.3	Eigenschaften des Petri-Netzes	14
2.2.4	Objektorientierte Petri-Netze.....	15
2.2.5	Simulation von Workflows mittels Petri-Netzen	17
3.	THORNs.....	19
3.1	Bestandteile der THORNs	19
3.1.1	Individuelle Objekte	20
3.1.2	Stellen.....	20
3.1.3	Kanten	21
3.1.4	Transitionen.....	22
3.2	Zeitkonzepte	23
3.2.1	Schaltregel.....	24
3.3	Hierarchiekonzepte	25
3.3.1	Schaltregel.....	27
4.	Verifikation und Validierung	29
4.1	Definitionen und Ziele von Verifikation und Validierung	29
4.2	Techniken der Verifikation und Validierung (V&V)	30
4.2.1	Animation.....	31
4.2.2	Begutachtung	32
4.2.3	Ereignisvaliditätstest & Festwerttest.....	33
4.2.4	Monitoring	33
4.2.5	Schreibtischtest & Validierung im Dialog & Sensitivitätsanalyse	34
4.2.6	Statistische Techniken	34
4.2.7	Strukturiertes Durchgehen.....	35
4.2.8	Test von Teilmodellen.....	35
4.2.9	Trace-Analyse & Vergleich mit anderen Modellen	36
4.3	Bewertung der Technik bzgl. der Anwendbarkeit	37
5.	Beurteilung der Automatisierungsmöglichkeiten der V&V Techniken	42
5.1	Automatisierungsgründe und -ziele.....	42
5.2	Evaluation der Automatisierungsmöglichkeit bzgl. V&V Techniken	44
6.	Evaluation der V&V Techniken.....	46
6.1	Erstellen eines auf THORNs basierenden Simulationsmodells.....	48
6.2	Evaluieren der ausgewählten V&V-Techniken.....	54
6.2.1	Begutachtung	54
6.2.2	Ereignisvaliditätstest	56
6.2.3	Festwerttest & Vergleich mit anderen Modellen	57
6.2.4	Monitoring	58

6.2.5 Sensitivitätsanalyse	60
6.2.6 Strukturiertes Durchgehen, Schreibtischtest, Validierung im Dialog	61
6.2.7 Animation.....	63
6.2.8 Trace-Analyse	64
6.2.9 Statistische Techniken	64
6.2.10 Test von Teilmodellen.....	67
7. Zusammenfassung und Ausblick	70
Literaturverzeichnis.....	72
Abbildungsverzeichnis.....	76
Tabellenverzeichnis	77

1. Einleitung

Seit Jahren gewinnt die Simulation in der Produktion und Logistik immer mehr an Bedeutung. Simulation heißt, ein möglichst realitätsnahes Nachbild von Geschehen der Wirklichkeit. Mit Hilfe der Simulation lässt sich wie z. B. der gesamte Arbeitsablauf inklusive aller relevanten Wechselwirkungen zwischen den einzelnen Prozessen nachbilden und simulieren. Besonders bietet es bei Neu- oder Änderungsplanung die Möglichkeit, alle möglichen Alternativen mittels der Simulationsmodelle zu bewerten und anschließend eine richtige Entscheidung zu treffen. Deswegen ist es hilfreich und sinnvoll, dass durch die Abstraktion der Wirklichkeit ein reales System vollständig dargestellt und behandelt werden kann. Das bedeutet, die Richtigkeit und Genauigkeit eines Simulationsmodelles haben eine erhebliche Bedeutung. Ein fehlerfreies Simulationsmodell bringt den Unternehmen nicht nur die Senkung der Produktionskosten sondern auch die Steigerung der Produktionseffizienz. Dann wird eine große Herausforderung für alle Unternehmen gestellt, eine effiziente Prüfung der Korrektheit der Simulationsmodelle durchzuführen.

Ziel dieser Arbeit ist die Untersuchung von bestehenden Validierungsmöglichkeiten für modulare, auf objektorientierten Petri-Netzen basierende Simulationsmodelle. Die Untersuchungsgrundlage sind die Modelle, die mit zeitbehafteten hierarchischen Objekt-relationalen Netzen (THORNs) modelliert werden.

Am Anfang wird die ereignisdiskrete Simulation eingeführt, weil die meisten Systeme in den Bereichen Produktion und Logistik ereignisdiskret sind. Wegen der deutlichen Vorteile werden die Petri-Netze als Modellierungstechnik für ereignisdiskrete Systeme ausgewählt. Anschließend werden die grundlegenden Begriffe, Eigenschaften sowie ein Simulationsbeispiel von Petri-Netzen vorgestellt.

Im dritten Kapitel werden die speziellen höheren Petri-Netze - sogenannte Timed Hierarchical Object-Related Nets (THORNs) - beschrieben, inklusive ihrer Bestandteile und ihrer zwei charakteristischen Konzepte, Zeitkonzept und Hierarchiekonzept.

Bis jetzt sind alle benötigten Kenntnisse über die untersuchte Grundlage vorbereitet. Deshalb werden im nächsten Schritt die Begriffe und Ziele über die Verifikation und Validierung sowie

ihre Notwendigkeit in einer Simulationsstudie erläutert. Danach wird die ausgewählte Verifikation- und Validierungstechnik detailliert beschrieben. Am Ende des Kapitels wird die Anwendbarkeit der Verifikation- und Validierungstechniken nach den Phasenergebnissen ausgewertet.

Im fünften Kapitel werden zuerst die Wichtigkeit und Bedeutung der Automatisierung der Verifikation- und Validierungstechnik erklärt und danach werden die Verifikation- und Validierungstechnik in drei Kategorien wie manuell, teil-automatisch und automatisch eingeordnet und noch ihre Automatisierungsmöglichkeiten evaluiert.

Schließlich werden im sechsten Kapitel die Verifikation und Validierung für ein konkretes auf THORNs basierendes Simulationsmodell mit den ausgewählten Verifikation- und Validierungstechniken durchgeführt und noch die Anwendbarkeiten der Verifikation- und Validierungstechniken für dieses Modell detailliert erläutert.

2. Ereignisdiskrete Simulation von Petri-Netzen

Ereignisdiskrete Systeme sind eine Art von Systemen, die sehr üblich im Bereich Ingenieurtechnik, Wirtschaft, Militärwesen auftreten. Ereignisdiskret bedeutet in diesem Zusammenhang, dass sich die Zustände eines Systems nur in Abhängigkeit von eintretenden Ereignissen ändern, deren Eintrittszeitpunkte und deren Einflüsse stochastisch ist. Deshalb können die bestehenden mathematischen Methoden und Werkzeuge ereignisdiskrete Systeme nicht genau beurteilen und analysieren. Computer Simulation ist momentan eine der effektivsten Methoden.

Die Theorie der Petri-Netze ist eine Modellbildungstheorie, die ereignisdiskrete Systeme beschreiben und analysieren kann. Die kann einfach statische und dynamische Verhaltenseigenschaften des Systems beschreiben und analysieren. Wegen der speziellen Eigenschaften des ereignisdiskreten Systems sind die Petri-Netze besonders gut geeignet. Durch die Netzstruktur werden die statischen Eigenschaften des Systems beschrieben und durch die Schaltungsregel der Petri-Netze werden dynamische Eigenschaften des Systems erläutert. Wegen der starken Fähigkeit zur Beschreibung eines asynchronen, nebenläufigen und konfliktären Systems gewinnen die Petri-Netze immer mehr Aufmerksamkeit. Ereignisdiskretes System ist eine Art der komplexen Systeme, in den die Ereignisse einander beeinflussen. Die Forschung der Petri-Netze entwickelt sich schon mehrere Jahre, darunter sind die Petri-Netze als Werkzeug ausgezeichnet. Der Grund dafür ist die starke Beschreibungsfähigkeit für ereignisdiskrete Systeme.

Die Vorteile der Petri-Netze für eine Modellbildung eines ereignisdiskreten Systems liegen in folgenden Punkten (Tian 2006):

- Die Einzigartigkeit der Petri-Netze macht Modellbildung verständlich und übersichtlich.
- Die Petri-Netze haben standarder Kriterien zur Definition der Semantik, können die Situationen genau beschreiben.
- Das Systemverhalten kann mittels von Petri-Netzen analysiert werden.
- Die hohe Entwicklung der Informationstechnik bietet uns kräftige Werkzeuge für Problemanalyse und -lösung.

Vorerst beziehen sich die Forschung auf ereignisdiskrete Systeme auf Modellbildung, Modellanalyse und Modellsteuerung. Petri-Netze sind ähnlich wie eine Brücke für das Problemlösen. In der Tat können Petri-Netze einen Entwurf für ein Industriesystem aufbauen, das zur Entscheidung und Analyse auf verschiedener administrativer Ebene angewendet werden kann. Um die Beschreibungsfähigkeit und die Modellbildungsfähigkeit noch weiter zu verstärken, werden die Faktoren wie Zeit und Attribute in Petri-Netze addiert. Durch diese Erweiterung werden die Petri-Netze vollständig und werden die Anwendungsgebiete noch größer.

2.1 Ereignisdiskrete Simulation

In diesem Abschnitt werden das ereignisdiskrete System und die ereignisdiskrete Simulation noch detailliert erklärt. Ein Ziel der ereignisdiskreten Simulation lautet Beherrschung der Bewegungsregeln des Systems, nämlich Regeln für interne Zustandsänderung. Im Gegensatz zu dem kontinuierlichen System sind die enthaltenen Prozesse der Ereignisse im Sinne der Zeit und Raum diskret, die interne Zustandsänderung ist stochastisch, ein Zustand kann in verschiedenen Zuständen verändert werden. Deshalb können die ereignisdiskreten Systeme nicht von der Differentialgleichung bestimmt werden, sondern von den Wechselwirkungen zwischen den einzelnen diskreten Ereignissen. Das einzelne Ereignis schaltet sich in einer bestimmten Reihenfolge oder in einer bestimmten Bedingung ein, z.B. in Montage-Roboter bewegt ein Werkstück von Position A nach Position B, dann treten die folgenden Bewegungen nacheinander auf: Der Roboter erreicht Position A, greift das Werkstück, bewegt das Werkstück zu Position B, lässt das Werkstück los, und dann verlässt. Ihre Zustände können leider durch die traditionellen mathematischen Methoden nicht beschrieben werden. Normalerweise werden lediglich der Zeitpunkt der Zustandsänderung sowie die Ursache der Zustandsänderung berücksichtigt, die Änderungsvorgänge kann man vernachlässigen.

Für das kontinuierliche System erhält man die Systemeigenschaften von der Beziehung zwischen dem Eingang und dem Ausgang, die mit Hilfe der Übertragungsfunktion oder Zustandsgleichung modelliert werden können. Für das diskrete Ereignissystem ist ereignisdiskrete Simulation eine der effektivsten Methoden zur Forschung dieser Systeme. Die ereignisdiskrete Simulation bietet uns eine Möglichkeit an, dass vor der tatsächlichen Ausführung dieses Systems eine Auswertung über ihre Betriebsleistung gegeben werden kann, d.h. man kann zu einem angemessenen Zeitpunkt eine richtige Entscheidung treffen.

Ereignisdiskretes System ist ein dynamisches System, das abhängig von den Ereignissen ist, deren Zustandsänderungen sprunghaft sind. Ihre Zustandsänderung des Systems taucht nur in den diskreten Zeitpunkten auf, außerdem sind diese diskreten Zeitpunkte nicht deterministisch. Der Grund für die Zustandsänderung ist das Ereignis, das eins zu eins mit der Zustandsänderung ist.

Nachfolgend werden einige wichtige Begriffe von ereignisdiskreter Simulation genau definiert (Liu, Wang, Feng 2005).

– Entity (Objekt)

Das Objekt ist ein Bestandteil des Systems, in dem ereignisdiskreten System wird das Objekt in zwei Arten eingeteilt, sie sind vorläufiges Objekt und dauerhaftes Objekt. Vorläufige Objekte sind ein Bestandteil der Systemsaktivität, sie treten in einen bestimmten Zeitpunkt in das System ein, bleiben in dem System und funktionieren mit anderen Objekten, dann verlassen. Dauerhafte Objekte sind ein fester Bestandteil des Systems, bleiben immer in dem System. Zum Beispiel: in einem Friseursalon sind Kunden die vorläufigen Objekte, während die Friseurin dauerhafte Objekte sind. Die Wechselwirkung zwischen den beiden Objektarten löst die interne Zustandsänderung des Systems aus.

– Attribute (Attribut)

Das Attribut beschreibt das Objekt, spiegelt ihre Eigenschaften wider. Ein Objekt kann viele Attribute haben, in der Simulation soll man nur die Attribute, die im Zusammenhang mit dem Simulationsziel stehen, berücksichtigen.

– State (Zustand)

In einer bestimmten Zeit ist der Systemzustand die Attributmenge aller Objekte.

– Event (Ereignis)

Ereignis verursacht Zustandsänderung, es ist ein kurzzeitiges Verhalten. Ereignisdiskretes System wird von dem Ereignis verursacht.

– Activity (Aktivität)

Der kontinuierliche Prozess zwischen den beiden Ereignissen wird als Aktivität genannt, der Beginn und das Ende der Aktivität werden von den Ereignissen ausgelöst.

– Process (Prozess)

Der Prozess besteht aus den Objekten und den Aktivitäten. Ein Prozess beschreibt enthaltene Ereignisse, noch die logischen und zeitlichen Beziehungen zwischen den einzelnen Aktivitäten.

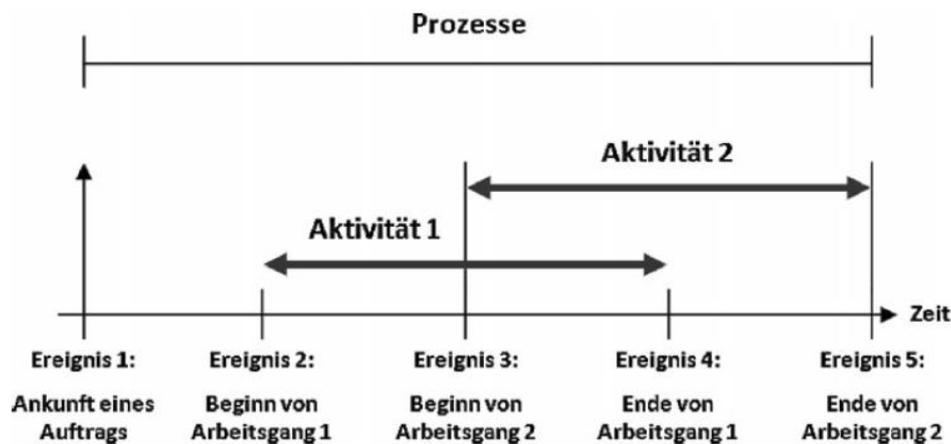


Abbildung 1: Prozesse, Aktivitäten und Ereignisse (Fishman 1973)

Abbildung 1 verdeutlicht den Zusammenhang zwischen den Ereignissen, Aktivitäten und Prozessen. Der ankommende Auftrag soll durch zwei Arbeitsgänge erledigt werden. Der Prozess wird von Ereignis 1 ausgelöst. Die beiden Aktivitäten beginnen und enden wegen der auftretenden Ereignisse und überlappen sich zeitweise.

2.2 Petri-Netze

Der Begriff Petri-Netz wurde im Jahre 1962 von einem deutschen Gelehrten Carl Adam Petri in seiner Dissertation „Kommunikation mit Automaten“ aufgestellt. Es ist ein systematisches mathematisches sowie graphisches Systembeschreibung- und Analyseninstrument, bietet uns die Möglichkeit an, dynamische Systeme mit nebenläufigen und nicht-deterministischen Vorgängen zu modellieren, zu analysieren und zu simulieren. Übersichtlich, anschaulich, verständlich sind die Vorteile von Petri-Netzen. Die Analysetechniken des Petri-Netzes können sowohl in der statischen Strukturanalyse als auch in der dynamischen Verhaltensanalyse verwendet werden, sind besonders geeignet für die Informationsverarbeitungssysteme, die nebenläufige, parallele, asynchrone, verteilte Eigenschaften besitzen. Nach Jahrzehnten der Entwicklung verbreiten sich die Anwendungsbereiche der Petri-Netze, nicht nur in dem Bereich Informatik, sondern auch in dem Geschäftsprozessmanagement, in der Produktion und Logistik sowie in vielen anderen Bereichen. In diesem Kapitel werden in Hinsicht auf den Grundsatz, die Definition sowie ihre Eigenschaften des Petri-Netzes detailliert erläutert.

2.2.1 Klassisches Petri-Netz

Ein klassisches Petri-Netz besteht aus drei grundlegenden Komponenten: Plätze (auch Stellen

genannt), Transitionen und Kanten (Reisig 2010).

– Plätze (auch Stellen genannt)

In Petri-Netzen existieren zwei Arten von Elementen, eine davon sind die Plätze, die in Petri-Netzen als Kreise oder als Ellipsen dargestellt werden. Ein Platz P/S ist eine passive Komponente, beschreibt einen bestimmten Zustand. Die Plätze können die Objekte lagern, speichern, sichtbar machen. Jeder Platz hat eine bestimmte Kapazität.

– Transitionen

Die anderen Arten von Elementen eines Petrinetzes sind die Transitionen, werden als Quadrate oder Rechtecke dargestellt. Eine Transition t modelliert immer eine aktive Komponente. Die Transition t kann die Objekte erzeugen, verbrauchen, transportieren und verändern. Eine Transition zeigt den Ressourcenverbrauch und die Zustandsveränderung des Systems. Die Feuerung der Transitionen wird von der Zustandsveränderung kontrolliert, das heißt, die Vorfeldbedingung der Transitionen muss erfüllt werden. Nach der Feuerung der Transitionen werden einige Vorbereichsbedingungen nicht mehr erfüllt, jedoch können einige Nachbereichsbedingungen erfüllt werden.

– Kanten

Plätze und Transitionen sind durch gerichtete Kanten verbunden. Die Kanten werden als Pfeile dargestellt und modellieren die Beziehungen zwischen Plätzen und Transitionen. Das heißt, eine Kante darf zwei Plätze oder zwei Transitionen nicht verbinden. Normalerweise führt die Kante von einem Platz zu einer Transition oder umgekehrt.

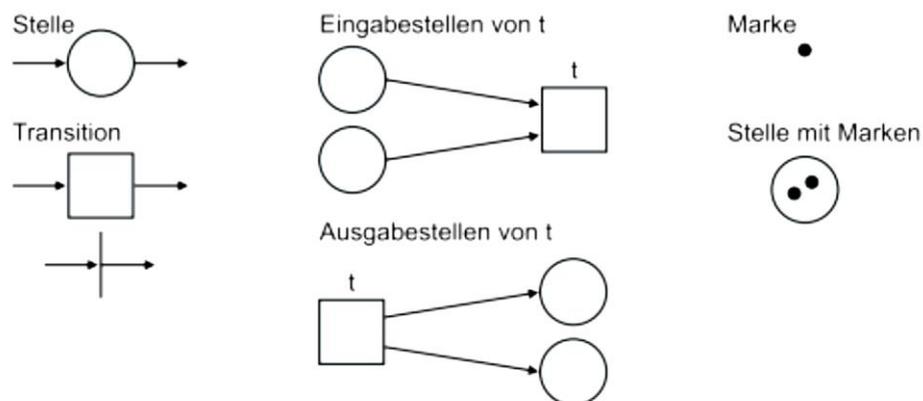
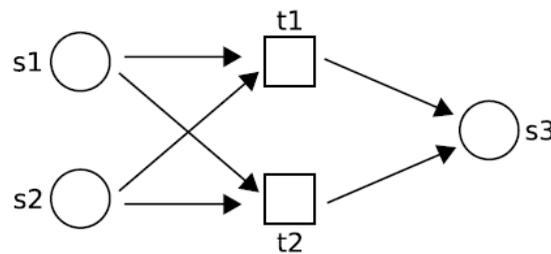


Abbildung 2: Elemente von Petri-Netzen (Balzert 2009, S. 303)

In Abbildung 2 werden die zuvor definierten Komponenten von Petri-Netzen und die Definition der Begriffe Eingabestelle und Ausgabestelle visualisiert (Balzert 2009). Der Begriff Marke wird im folgenden Abschnitt genauer erläutert.

Zusammenfassend ist klar, dass irgendein System in einer drei-Komponenten-Struktur mit dem Zustand, dem Ereignis und der Beziehung zwischen den Beiden abstrakt modelliert werden kann. Diese Struktur wird mathematisch als ein Tripel $N=(P,T,F)$ geschrieben (Mülle 2007):

1. $P = \{p_1, p_2, \dots, p_n\}$ ist eine beschränkte Menge von den Plätzen/Stellen.
2. $T = \{t_1, t_2, \dots, t_n\}$ ist eine beschränkte Menge von den Transitionen.
($P \cup T \neq \emptyset, P \cap T = \emptyset$)
3. $F = (P \times T) \cup (T \times P)$ sind die Menge von Input-Funktion und die Menge von Output-Funktion, wird als Flussrelation genannt.



$$S = \{s1, s2, s3\}$$

$$T = \{t1, t2\}$$

$$F = \{(s1, t1), (s1, t2), (s2, t1), (s2, t2), (t1, s3), (t2, s3)\}$$

Abbildung 3: Ein Beispiel von einem Petri-Netz (Frommholz 2007, S.16)

Das in Abbildung 3 skizzierte Petri-Netz besteht aus drei Stellen und zwei Transitionen. Input-Funktionen (s_1, t_1) , (s_2, t_1) , (s_1, t_2) , (s_2, t_2) und Output-Funktionen (t_1, s_3) , (t_2, s_3) werden in F eingetragen.

Um die Beschreibung zu vereinfachen, werden die Begriffe Vorbereich und Nachbereich eingeführt. Der Vorbereich und der Nachbereich von Element x werden definiert als (Mülle 2007):

$$\cdot x = \{y | \exists y: (y, x) \in F\}$$

$$x \cdot = \{y | \exists y: (x, y) \in F\}$$

Deshalb $\cdot x$ wird als der Vorbereich oder die Menge des Inputs genannt, $x \cdot$ wird als der Nachbereich oder die Menge des Outputs genannt. In Abbildung 3 ist $\{s_1, s_2\}$ der Vorbereich von t_1 , $\{t_1, t_2\}$ ist der Vorbereich von s_3 und der Nachbereich von s_1 und s_2 , $\{s_3\}$ ist der Nachbereich von t_1 und t_2 .

Bisher haben wir die Prozesse nur statisch modelliert. Um das dynamische Verhalten von Petri-Netzen zu untersuchen, wird der Begriff Systemnetze eingeführt. Bevor die Systemnetze vorgestellt wird, werden zuerst noch einige Begriffe erklärt.

– Markierungen

Eine Markierung zeigt die Verteilung von Marken/Token auf einem Platz. Es gibt natürlich verschiedene Arten von Marken/Token, deswegen bezeichnet man sie mit unterschiedlichen Symbolen, wie z.B. Nummer, Zylinder oder andere Bilder. Aber häufig werden die Marken nicht symbolisch sondern vereinfacht als schwarze Punkte bezeichnet. Der Zustand des Systemnetzes wird durch die Verteilung des Tokens bezeichnet. Der Anfangszustand des Systemnetzes wird durch Anfangsmarkierung dargestellt (Reisig 2010).

– Kantenbeschriftung

Bei konstanter Kantenbeschriftung sind einzelne Elemente an den Kanten rund um eine Transition t notiert. Normalerweise stehen an einer Kante mehr als ein Element. Man schreibt für jede Kante wie die untere Grafik (p,t) und (q,t) . Die Klammern $[]$ bezeichnen die Multimenge. Man kann die konstante Kantenbeschriftung auch durch eine Konstante notieren, wenn eine Stelle nur eine Art der Objekte enthält (Reisig 2010). Abbildung 4 veranschaulicht, wie man die konstante Kantenbeschriftung schreiben soll.

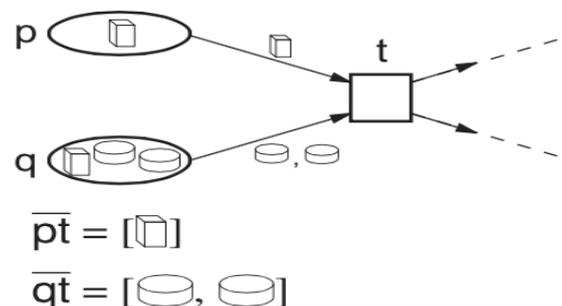


Abbildung 4: Konstante Kantenbeschriftung (Reisig 2010, S. 27)

Wenn $M(p) \geq \overline{pt}$, aktiviert die Markierung M die Transition t , gleichzeitig ist die neue Markierung $M'(p) = M(p) - \overline{pt} + \overline{tp}$ definiert (Reisig 2010).

Bei variabler Kantenbeschriftung werden die Kanten oder die Transition mit Variablen beschrieben. Wenn man Variable x durch eine Konstante ersetzt, wird die Variable x ausgerechnet, dann bekommt man eine Multimenge bei x an einer Kante, kriegt man eine Aussage „wahr“ oder „falsch“ bei x in einer Transition.

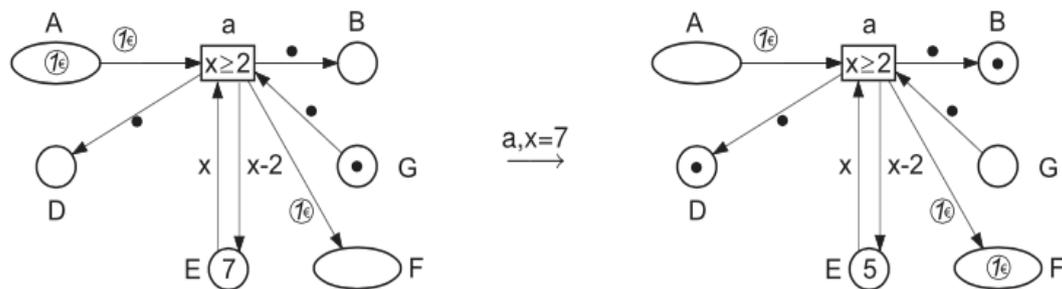


Abbildung 5: Variable Kantenbeschriftung (Reisig 2010, S. 29)

Wenn für jede Kante (p,t) gilt: $M(p) \geq \beta(p,t)$ und für die Anschrift i von t gilt: $\beta(i) = \text{true}$, aktiviert eine Markierung M eine Transition t in einem Modus β von t , gleichzeitig ist die neue Markierung $M'(p) = M(p) - \beta(p,t) + \beta(t,p)$ definiert (Reisig 2010). In Abbildung 5 wird die Transition a im Modus $x=7$ aktiviert.

2.2.2 Systemnetze

Ein Systemnetz wird von einer Netzstruktur mit einer Anfangsmarkierung, Transitionsbedingungen, Kantenanschriften und als kalt gekennzeichnete Transitionen zusammengebildet (Reisig 2010).

Für Petri-Netze ist die Markierung von großer Bedeutung. Markierung ist eine Verteilung von Marken/Token auf den Plätzen von N . Normalerweise schreibt man M_0 für die Anfangsmarkierung, M_0 stellt uns den Anfangszustand dar. Theoretisch sind unendlich viele Markierungen eines Systemnetzes erreichbar. Diese erreichbaren Markierungen und Schritte kann man in einem sogenannten Markierungsgraph von N zusammenstellen, deswegen wird es auch als Erreichbarkeitsgraph (Zustandsübergangsdiagramm) genannt. Die Beschriftungen von

Kanten und Transitionen lassen unterschiedliche Möglichkeiten passieren, die t aktivieren oder den Eintritt von t beeinflussen. Bei der Definition des Systemnetzes ist es auffällig, dass die Transitionen in heiß und kalt eingeteilt sind. Kalte Transition wird mit „ ε “ gekennzeichnet, es heißt, tritt nicht unbedingt ein, wenn sie aktiviert ist. Heiße Transition heißt, tritt ein, wenn sie aktiviert ist. Ebenso hat ein System auch einen Endzustand, in dem das System für immer verharren kann (Reisig 2010).

Die einfachsten Petri-Netze sind Bedingungs/Ereignis-Netze. In den Bedingungs/Ereignis-Netzen sind alle Marken gleich und jede Stelle darf maximale eine Marke enthalten. Die Transition t kann sich schalten, wenn jede Eingabestelle von t eine Marke enthält und wenn jede Ausgabestelle von t leer ist (Link 2016). Abbildung 6 zeigt die Markenbewegung beim Schalten von t in einem Bedingungs/Ereignis-Netz.

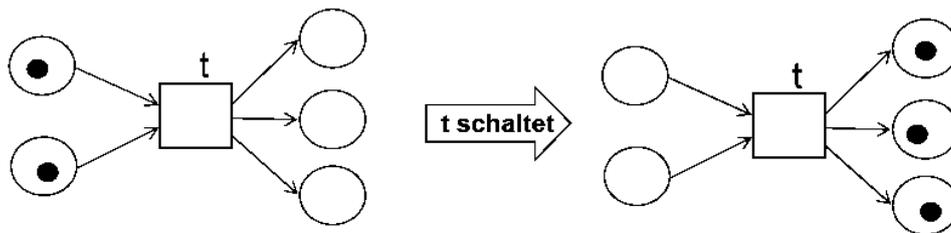


Abbildung 6: Bedingungs/Ereignis-Netz (Link 2016, S. 3)

Um die Einsatzgebiete zu erweitern, werden hier Stellen/Transitions-Netze eingeführt. Bei den Stellen/Transitions-Netzen darf jede Stelle mehrere gleiche Marken bis zur vordefinierten Kapazität K aufnehmen. Die Kanten werden mit Zahlen versehen. Bei dem Feuern einer Transition müssen die Eingabestellen so viele Marken entnehmen bzw. die Ausgabestellen hinzufügen, wie der jeweilige Gewichtswert der Kante angegeben wird. Abbildung 7 ist ein typisches Beispiel von Stellen/Transitions-Netz. Wenn die Transition "Fahrrad montieren" aktiviert ist, werden nach dem jeweiligen Gewichtswert von der Stelle Räder zwei Marken, von der Stelle Kette eine Marke und von der Stelle Rahmen eine Marke konsumiert, entsprechend wird auf der Stelle Fahrrad eine Marke produziert.

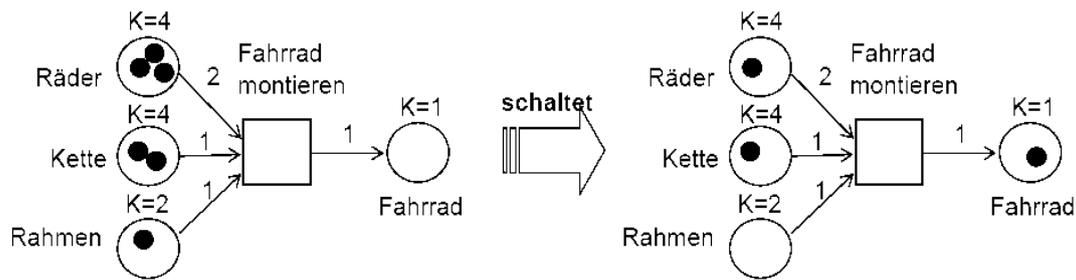


Abbildung 7: Stellen/Transitions-Netz (Link 2016, S. 10)

Mathematisch kann ein Stellen/Transitions-Netz als ein 6-Tupel $\Sigma = (S, T, F, K, W, M_0)$ definiert werden, S ist Stelle, T ist Transition, F ist Flussrelation, K ist Kapazitätsfunktion, W ist Gewichtsfunktion und M_0 ist die Anfangsmarkierung von Σ (Goldschmidt, Louen 2014). Wenn die Transition sich ständig ändert, ändert sich auch der Zustand des Systemnetzes. Dieser Prozess wird als die Ausführung des Systemnetzes genannt. Wir beschäftigen uns damit etwas genauer.

Wenn für alle $p \in P$, die Bedingung $p \in \cdot t \Rightarrow M(p) \geq W(p, t) \wedge p \in t \cdot \Rightarrow M(p) + W(t, p) \leq K(p)$ erfüllt wird, kann t unter der Markierung M aktiviert werden, schreibt man $M[t >$. Wenn eine Transition t unter der Markierung M aktiviert werden kann, wird die Markierung M zu M' werden, dann notiert man $M[t > M'$. Für alle $p \in P$ gibt es (He 2013; Frommholz 2007):

$$M'(p) = \begin{cases} M(p) - W(p, t) & \text{wenn } p \in \cdot t - t \cdot \\ M(p) + W(t, p) & \text{wenn } p \in t \cdot - \cdot t \\ M(p) - W(p, t) + W(t, p) & \text{wenn } p \in t \cdot \cap \cdot t \\ M(p) & \text{wenn } p \notin t \cup t \cdot \end{cases}$$

Die Aktivierungsbedingung und –regeln von Transition kann man beschreiben wie unten:

Eine Transition ist aktiviert, wenn die Tokenmenge jeder Vorbereichsstelle dieser Transition nicht weniger als das Gewicht der eingehenden Kanten ist. Gleichzeitig soll die neue Tokenmenge der Nachbereichsstelle (d.h. schon existierte Tokenmenge plus des Gewichtes der eingehenden Kante) kleiner als die Kapazität der Nachbereichsstelle sein. Abbildung 8 zeigt die Beispiele zum Schalten von Transitionen (León, Kiencke 2013).

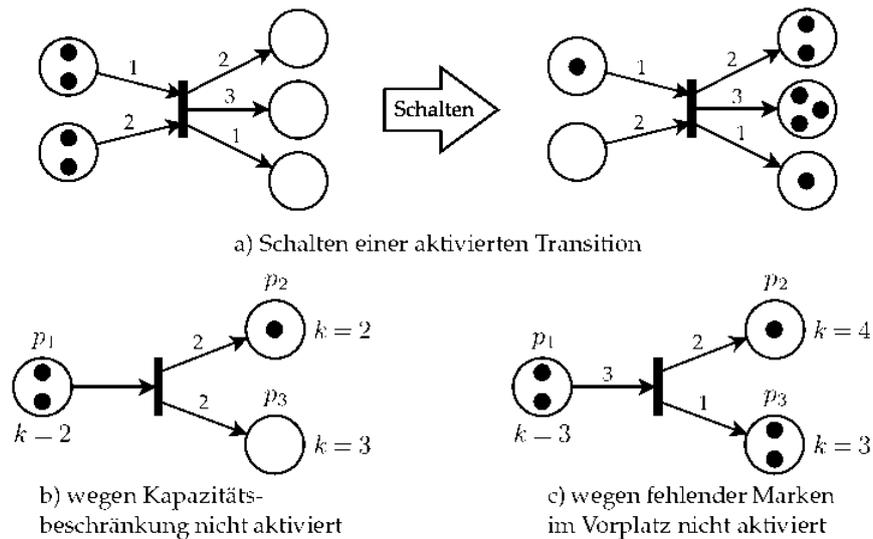


Abbildung 8: Aktivierung und Schalten (León, Kiencke 2013, S.340)

In einem Systemnetz kann man die möglichen Beziehungen zwischen den Transitionen definiert. Graphische Beschreibung ist in Abbildung 9 zu sehen. Nebenläufige Prozesse sind voneinander unabhängig, sie finden parallel statt. Bei dem Konflikt wird es beispielsweise passieren, dass zwei Maschinen für ein Werkstück zur Verfügung stehen. Bei der Vereinigung werden die Prozesse vereinigt, z.B. zwei Fließbänder, die zuerst unabhängig die gleichen Produkte transportieren, werden später in einem Depot vereinigt. Bei der Verklemmung können die Transitionen nicht aktiviert werden. Bei dem Falle wird entweder Transition t_1 oder Transition t_2 bevorzugt bearbeitet, je nachdem wer zuerst s_1 bzw. s_2 mit einer Marke belegt (Schlick 2012).

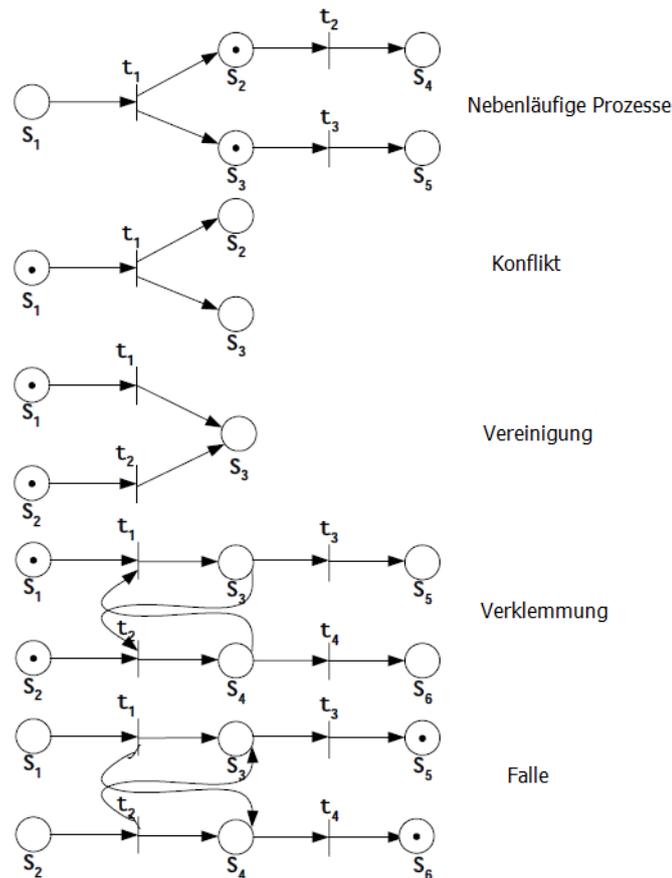


Abbildung 9: Die Beziehungen zwischen den Transitionen (Tian 2006, S.35)

2.2.3 Eigenschaften des Petri-Netzes

Auf Basis des Erreichbarkeitsgraphs werden hier einige dynamische Netzeigenschaften eingeführt (König 2011).

- Beschränktheit und Sicherheit
Es gibt eine Konstante $c \in \mathbb{N}_0$, für jede erreichbare Markierung m und jede Stelle s gilt, dass $m(s) \leq c$. Wenn $c = 1$, ist das Petri-Netz sicher.
- Verklemmung (Deadlock)
Es gibt eine erreichbare Markierung, unter der keine Transition aktiviert ist. Man kann mit Hilfe des Erreichbarkeitsgraphs die Möglichkeit der Verklemmung des Petri-Netzes untersuchen.
- Lebendigkeit (liveliness)
Lebendigkeit enthält die Lebendigkeit einer Transition und die Lebendigkeit des gesamten Petri-Netzes. Alle Transitionen des modellierten Systems können aus jedem möglichen Zustand heraus wieder ausgelöst werden, d.h. in dem modellierten System gibt es

keine totalen und partiellen Verklemmungen.

- Reversibilität

Das reversible System kann ausgehend von jedem möglichen Zustand wieder in den Anfangszustand überführt werden.

Es soll beachtet werden, dass Lebendigkeit und Deadlock nicht-strukturelle Eigenschaften sind. Die Beiden sind von der Anfangsmarkierung M_0 abhängig. In dem Prozess der Simulation ist Deadlock ein sehr wichtiges Problem, der besondere Aufmerksamkeit geschenkt werden soll.

2.2.4 Objektorientierte Petri-Netze

In dieser Arbeit beschäftigen wir uns mit den Petri-Netzen, die objektorientiert sind. In der Literatur gibt es vielfältige Ansätze, objektorientierte Techniken mit der Theorie der Petri-Netze zu kombinieren. In dem Artikel „Ansätze zur Integration von objektorientierten Konzepten und Petri-Netzen“ von Michael Zapf und Armin Heinzl werden die Ansätze nach der Integrationsrichtung in folgenden Kategorien zusammengefasst (Zapf, Heinzl 1998):

- Ansätze zur Einbettung von Objekten in Petri-Netze,
- Ansätze zur Einbettung von Petri-Netzen in Objekte und
- Ansätze zur beidseitigen Integration von Objekten und Petri-Netzen.

Am verbreitetsten sind die Object Petri Nets (kurz: OPN), die durch die Programmiersprache LOOPN++ (Language for Object-Oriented Petri Nets) beschrieben werden (Schnieder 1999), die zu dem ersten Ansatz gehört. Weil THORNs auch ein Ansatz zur Einbettung von Objekten in Petri-Netze ist, wird in meiner Arbeit lediglich dieser Ansatz betrachtet werden. In diesen objektorientierten Petri-Netzen werden die Marken mit objektorientierten Datentypen versehen und die Transitionen mit Hilfe einer objektorientierten Programmiersprache beschrieben. Abbildung 10 zeigt einen Überblick über die Einbettung von Objekten in Petri-Netze.

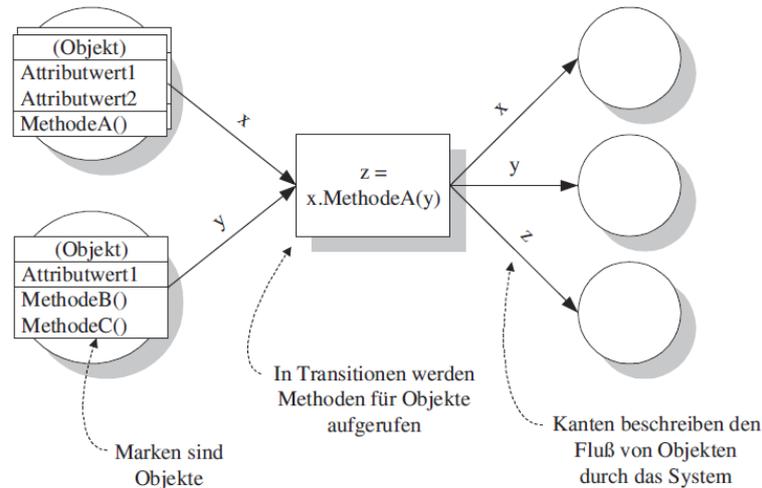


Abbildung 10: Einbettung von Objekten in Petri-Netze (Bastide 1995, S. 1)

Zur objektorientierten Beschreibung eines Systems sollen folgende Begriffe beachtet werden (Schnieder 1999).

- Objekte
sind wesentliche Bestandteile eines Systems, können mit verschiedenen Attributen (Eigenschaften) versehen werden.
- Klassen (Objektypen)
sind eine Reihe von ähnlichen Objekten, enthalten die Type der Objekte sowie die Bearbeitung der Objekte, während unter einer Instanz ein konkretes Objekt mit definierten Werten verstanden wird.
- Methoden (Verhaltensweisen)
beschreiben und implementieren das Verhalten der Objekte, mit denen das Element eines Objekts verändert werden kann.

Eigenschaften von objektorientierten Methoden (Schnieder 1999):

- Abstraktion
ermöglicht die Vereinfachung eines komplexen realen Zusammenhangs in einer einfachen abstrakten Form.
- Kapselung
bedeutet das Verbergen von Details eines Objekts vor anderen Teilen des Programms.
- Vererbung
bedeutet das Weitergeben von Variablen und Methoden an eine Subklasse, bzw. das Erben von Variablen und Methoden von einer Superklasse. Subklassen können die geerb-

ten Methoden und Variablen auch erweitern, sowie spezialisierte Methoden und Variablen implementieren.

- Polymorphie

tritt immer mit Vererbung und Schnittstellen auf. Wenn eine Methode in verschiedenen Klassen die gleiche Signatur hat, jedoch sie erneut implementiert ist, ist sie polymorph.

2.2.5 Simulation von Workflows mittels Petri-Netzen

Die Petri-Netze werden sehr häufig im Bereich Simulation von Workflow angewendet. Bevor das konkrete Simulationsbeispiel von Petri-Netzen eingeführt wird, werden Ihnen zuerst die Begriffe Workflow und Workflow-Net vorgestellt werden. Anschließend werden die Gründe für Petri-Netze als Werkzeug zur Modellierung des Workflow erläutert.

Der Begriff des Workflow stammt aus dem Bereich Produktionsorganisation und der Automatisierung im Büro. Es beschreibt die Aktivitäten in der alltäglichen Arbeit. Wegen der rasanten Entwicklung der Computer Technologie und der verteilten Datenbanktechnik wird das Anwendungsgebiet der Workflow-Net immer verbreitet. Bis heute wird die Workflow-Net Technik schon erfolgreich in dem Bereich Bibliothek, Krankenhaus, Projektmanagement usw. eingesetzt. Darüber hinaus unterstützt die Workflow(WF)-Net Technik auch die Software-Entwicklung.

Seit den letzten Jahren hat die Forschung über die Beschreibung und die Analyse-methode von WF-Net mehr Aufmerksamkeit gewonnen die Werkzeuge für die WF-Modellierung wird immer mehr. Wegen der nachfolgenden Vorteile werden die Petri-Netze ausgewählt, auf denen ein Workflow-Modell etabliert wird (Zhou, Wu, Gu, Chen 2005).

- die auf Petri-Netzen etablierte Workflow-Prozesse haben deutliche und präzise Bedeutung
- übersichtliche Grafikdarstellung, einfache Kommunikation mit den Endkunden
- starke Ausdrucksfähigkeit

Die Voraussetzungen der Richtigkeit von Workflow-Net lautet (hier i ist der Anfangszustand, o ist der Endzustand) (Zhou, Wu, Gu, Chen 2005):

- Zustand o ist der einzige Endzustand. Wenn ein Workflow endet, hat nur die Stelle o ein

Token, die andere Stelle sollen leer sein.

- Jede Transition ist ausführbar.

Nachfolgend wird in Abbildung 11 ein Workflow-Modell mit Hilfe von Petri-Netzen dargestellt.

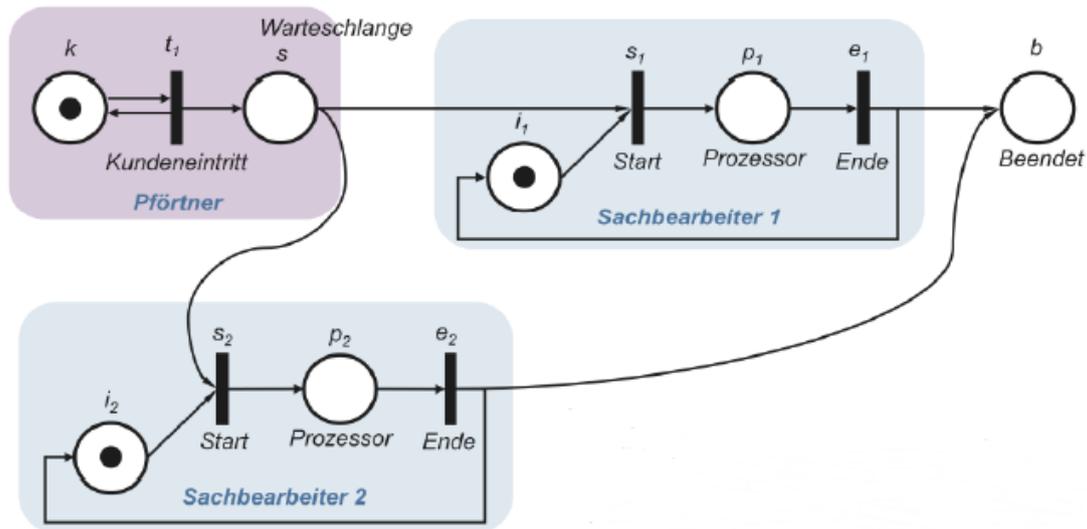


Abbildung 11: Zwei Sachbearbeiter bedienen die Bankkunden (Schlick 2012, S. 3)

Abbildung 11 bildet den Schaltbereich einer Bank mit einem Pförtner und zwei Sachbearbeitern vereinfacht ab. Es ist ein typisches Petri-Netz-Modell des Warteschlangensystems. Die wartenden Kunden sind auf der Stelle s , die mit einer definierten Kapazität k versehen werden kann. Stelle i_1 und Stelle i_2 bedeuten, dass der Sachbearbeiter 1 oder 2 bereit für den nächste Kunde ist. Danach wird der Kunde nach dem Status von Stelle i_1 und i_2 die Transition s_1 oder s_2 auswählen. Nach der Ausführung von den Transition e_1 oder e_2 wird ein Token auf der Stelle b hinzugefügt, zugleich wird auch ein Token auf der Stelle i_1 oder i_2 erzeugt.

Es ist fraglos, dass dieses auf Petri-Netz basierende Modell eine anschauliche Darstellung für den alltäglichen Geschäftsprozess in der Bank gibt. Aber jetzt kommt die Frage, wie man mit Hilfe dieses Petri-Netz-Modells das Verständnis des bestehenden Realsystems verbessern kann. Es ist klar, dass das einfache Petri-Netz die Anforderung nicht mehr erfüllen kann, z.B. wie man die Wartezeit minimieren kann, wie man das Verhältnis zwischen Wartezeit und Bearbeitungszeit verbessern kann, wie man die Kundendaten gut verwalten kann usw. Im nächsten Kapitel wird ein hohes Petri-Netz (sogenannte THORN) vorgestellt, das stärkere Modellierungsfähigkeit hat, sodass die immer steigenden Anforderungen des Auftraggebers erfüllt werden können.

3. THORNs

THORNs sind Abkürzung für **T**imed **H**ierarchical **O**bject-**R**elated **N**ets. THORNs sind höhere Petrinetze, die für verteilte Systeme mit diskreter Ereignisstruktur eine kompakte und anschauliche Modellbildung erlauben.

Die Unterschiede zwischen THORNs und klassischen Petri-Netzen liegen in untenliegenden Punkten (Schöf, Sonnenschein 1997, S. 4):

- Komplexe Objekte wie z.B. objektorientierte Programmiersprache C++ statt attributloser Objekte können als Marken in THORNs angewendet werden.
- Plätze (Stellen) werden nach der Struktur Multimenge, Stack, Queue oder Priority Queue zugeordnet werden. Jeder Platz enthält Objekte eines Typs und kann durch eine Kapazität beschränkt werden.
- Kanten werden in vier Arten: Standardkanten, aktivierende Kanten, inhibitorische Kanten und konsumierende Kanten unterschieden.
- Transitionen erwerben eine Schaltbedingung und eine Schaltaktion, die in C++ ausgedrückt werden. Darüber hinaus haben Transitionen auch eine Schaltkapazität, die angibt, wie oft die Transition parallel zu sich selbst schalten kann.
- In THORNs haben die Transitionen noch zwei spezifische zeitliche Funktionen Verzögerungszeit und Schaltdauer. Die beiden Zeiten können sowohl deterministisch als auch stochastisch spezifiziert werden.
- Durch eine Transitionenverfeinerung oder den Unternetzaufruf einer Transition kann die Strukturierung der Netzmodelle durchgeführt werden.

3.1 Bestandteile der THORNs

THORNs sind Stellen/Transitions-Netze. Deswegen sind wesentliche Bestandteile Objekte (Marken), Stellen, Kanten und Transitionen.

3.1.1 Individuelle Objekte

Im Gegensatz zu den klassischen Petrinetzklassen können in THORNs die Objekte bzw. die Objekttypen mit verschiedenen Attributen definiert werden. Dazu werden die Objekte bzw. die entsprechenden Objekttypen mit Hilfe der weit verbreiteten objektorientierten Programmiersprache C++ spezifiziert. Weil THORNs objektorientiert sind, werden hier die Objekttypen in Form von C++-Klassen definiert. Marken sind Instanzen dieser Klassen. Das bedeutet, bei der Spezifikation der Objekttypen können alle objektorientierten Konzepte wie z.B. Datenabstraktion, Kapselung, Vererbung, Polymorphismus und dynamisches Binden verwendet werden.

3.1.2 Stellen

Stellen werden in THORNs nicht nur mit einem Namen, sondern auch mit einem Typ und einer Kapazität versehen. Der Objekttyp wird durch den Typ der Stelle festgelegt und kann ein elementarer oder ein selbstdefinierter komplexer Objekttyp sein. Die Kapazität der Stelle gibt an, wie viele Objekte maximal zu einem Zeitpunkt an dieser Stelle existieren dürfen. In THORNs darf die Kapazität auch unendlich sein, dann schreibt man den Wert Omega. Darüber hinaus hat die Stelle in THORNs noch eine Stellenstruktur. Die Stellenstrukturen werden nach der Reihenfolge der Objekte an einer Stelle in vier verschiedene Arten unterteilt, die Multiset, Queue, Stack und Priority Queue sind. Grafische Darstellungen sind in Abbildung 12 gezeigt.

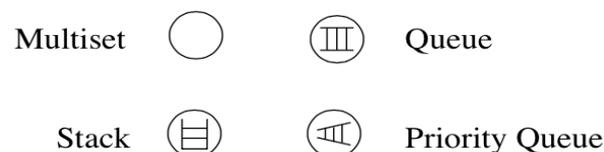


Abbildung 12: Darstellung von THOR-Netz-Stellen (Schöf, Sonnenschein 1997, S. 11)

- Multiset-Stellen
An den Multiset-Stellen können die Objekte beliebig hinzugefügt und abgezogen werden und werden die Objekte als Mengen verwaltet.
- Stack-Stellen
An den Stack-Stellen werden die Objekte nach dem LIFO-Prinzip (Last In First Out) ver-

waltet. Das Objekt, das zuletzt hinzugefügt wird, wird zuerst abgezogen.

- Queue-Stellen
An den Queue-Stellen werden die Objekte nach dem FIFO-Prinzip (First in First Out) verwaltet. Das Objekt, das sich am längsten an dieser Stelle befindet, wird zuerst abgezogen.
- Priority-Queue-Stellen
An den Priority-Queue-Stellen werden die Objekte nach einer vom Modellierer zur spezifizierenden Prioritätsfunktion zugeordnet. Das Objekt mit höchster Priorität wird zuerst abgezogen.

3.1.3 Kanten

Die Kanten in THORNs verfügen über einen Namen und einen Kantentyp. Im Vergleich mit klassischen Petri-Netzen gibt es neben den Standard-Kanten noch drei spezielle Kantentypen, sie sind Enabling-Kanten, Inhibitor-Kanten, Consuming-Kanten. Man muss aufpassen, dass die spezielle Kanten ausschließlich von Stellen zu Transitionen gehen dürfen. Dazu können einige Kantentypen mit einem Kantengewicht und einem Variablennamen versehen werden. Grafische Darstellungen sind in Abbildung 13 zu sehen.

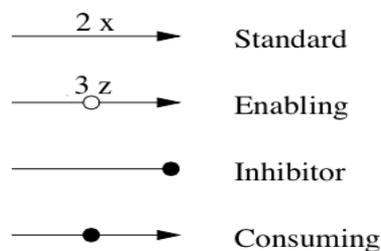


Abbildung 13: Darstellung von THOR-Netz-Kanten (Schöf, Sonnenschein 1997, S. 12)

- Standard-Kanten
Sie werden mit einem Gewicht und einem Variablennamen beschriftet. Das Gewicht der Kante von einer Stelle zu einer Transition gibt an, wie viele Objekte an der Stelle liegen müssen, um die Transition zu aktivieren bzw. wie viele Objekte von der Stelle abgezogen werden, wenn die Transition sich schaltet. Das Gewicht der Kante von einer Transition zu einer Stelle wird so definiert, wie viel Platz an der Stelle zur Verfügung stehen muss, um die Transition zu aktivieren bzw. wie viele Objekte an der Stelle beim Schalten

der Transition produziert werden.

- Enabling-Kanten (auch aktivierende Kanten genannt)
Die Beschriftung von Enabling-Kanten ist genauso gleich wie Standard-Kanten. Aber die Objekte dürfen beim Schalten der Transition nicht verändert werden. Deswegen wird Enabling-Kante auch als Lese- oder Testkante bezeichnet.
- Inhibitor-Kanten (inhibitorische Kanten)
Wenn die Objekte an der inzidenten Stelle stehen, deaktivieren Inhibitor-Kanten eine Transition, aber beeinflusst die adjazente Stelle nicht.
- Consuming-Kanten (konsumierende Kanten)
Beim Schalten dieser Transition werden sämtliche Objekte von der adjazenten Stelle gelöscht, haben aber diese Kanten keinen Einfluss auf die Aktiviertheit einer Transition.

3.1.4 Transitionen

Durch das Schalten der Transitionen werden die Objekte von Stellen konsumiert, modifiziert und produziert. Sie werden mit einer Schaltbedingung und einer Schaltaktion versehen. Die Spezifikation der Schaltbedingung und der Schaltaktion können mit C++-Code beschrieben werden. Der Bezug zu den Objekten auf den umliegenden Stellen wird durch die Variablennamen der inzidenten Kanten einer Transition hergestellt. Die Variablennamen sollen eindeutig sein, um die Konflikte zu vermeiden.

Dazu hat jede Transition auch ihre Transitionsnamen und Schaltkapazität. Mit der Schaltkapazität wird es definiert, wie oft eine Transition parallel zu sich selbst schalten kann, es gibt auch den Fall mit unbeschränkter Schaltkapazität Omega.

Die Schaltbedingung wird als booleschen C++-Ausdruck über den Variablen eingehender Kanten dargestellt. Wenn die Bedingung durch die Belegung der Variablen mit Objekten erfüllt ist, wird die Transition aktiviert. Die Zusammenstellung einer Transition mit einer Variablenbelegung wird in THORNs als Schaltereignis genannt. Die Schaltaktion einer Transition wird in einer Folge von den Anweisungen in Form von C++-Code angegeben und beim Schalten dieser Transition ausgelöst, um die Objekte zu erzeugen und zu modifizieren.

3.2 Zeitkonzepte

Für die Petri-Netze werden vielfältige Zeitkonzepte erhoben, das zeitliche Verhalten kann sowohl mit Stellen und Transitionen als auch mit Kanten und Marken in Verbindung gebracht werden. Heutzutage ist dieser Ansatz am verbreitetsten, dass die Transitionen mit dem Zeitkonzept modelliert werden, da es sehr notwendig in dem Bereich Produktion und Logistik ist, das zeitliche Verhalten jedes Prozesses wie z.B. Ausführungszeiten, Wartezeiten zu modellieren.

Hier werden zuerst zwei verschiedene Ansätze von den zeitbewerteten Transitionen präsentiert (Schöf, Sonnenschein 1997):

- Die Transition wird **mit einer festen Schaltdauer** versehen, mit der z.B. die Bearbeitungszeit eines Produktionsprozesses beschrieben werden kann. Am Beginn des Schaltens werden die Marken konsumiert, während des Schaltens liegen die konsumierten Marken im Inneren der Transition, und in diesem Zeitraum können sie nicht von den anderen Transitionen verwendet werden. Am Ende des Schaltens werden die Marken auf den Nachbereichstellen produziert.
- Die Transitionen werden **mit einem Intervall** $[x, y](x \leq y)$ beschriftet, wobei x der früheste Schaltzeitpunkt, y der späteste Schaltzeitpunkt nach einer ununterbrochenen Aktiviertheit festgestellt wird. Das Schalten einer Transition braucht keine Zeit. Die Marken, die zum Schalten benötigt werden, können während der Wartezeit von anderen Transitionen angewendet werden. Gleich wie der erste Ansatz werden die Transitionen auch mit Bearbeitungszeit versehen.

Das Zeitkonzept in THORNs ist eine Kombination von den bestehenden Ansätzen. Jede Transition kann man mit einer Verzögerungszeit und einer Schaltdauer beschreiben. Die Verzögerungszeit wird meistens zur Modellierung einer Vorbereitungs-, Warte- oder Rüstzeit benutzt, die einen Zeitraum definiert, über den eine Transition ununterbrochen aktiviert sein muss, bevor sie mit dem Schalten beginnen kann. Die Schaltdauer wird zur Modellierung einer Bearbeitungs- oder Ausführungszeit eines Prozesses angewendet und gibt einen Zeitraum an, der zwischen Schaltbeginn und Schaltende ist. Beide Zeiten dürfen keine negativen realen Werte annehmen. Die Spezifikation der Verzögerungszeit und der Schaltdauer können

mit Hilfe von C++-Code formuliert werden, und können die beiden Zeiten noch mittels der Zufallszahlengeneratoren verändert werden. Man soll beobachten, dass die Bewertung der Zeitbeschriftungen keinen Einfluss auf die gebundenen Vorbereichsobjekte hat.

3.2.1 Schaltregel

Für die flache zeitbeschriftete THORNs wird die Schaltregel abhängig von der Belegung der Variablen einer Transition, deswegen wird der Begriff Schaltereignis nochmals wiederholt, der eine Kombination von Transitionen und Variablenbelegungen ist.

Das Schaltereignis soll zunächst die folgenden Vorbereichsbedingungen erfüllen, um die Transition zu aktivieren (Schöf, Sonnenschein 1997).

- alle inhibitorisch wirkenden Stellen im Vorbereich leer sind.
- alle Vorbereichsstellen, die über Standard- oder aktivierende Kanten mit der Transition verbunden sind, nicht weniger Objekte enthalten als durch das Gewicht der verbindenden Kante gefordert wird.
- durch die Belegung der Variablen dieser Kanten mit Objekten der inzidenten Stellen die Schaltbedingung der Transition erfüllt wird.

Die Vorbereichsbedingungen sind ein Teil der Bedingungen, die erfüllt sein müssen, um eine Transition zu aktivieren. Wenn alle Schaltereignisse in einem Zustand die Vorbereichsbedingung erfüllen, wird die verzögert. Die Verzögerungszeit wird durch Bewertung des Transitionsausdrucks bestimmt.

Die Voraussetzungen der Aktivierung von einem Schaltereignis lauten:

- das Schaltereignis ist vollständig verzögert.
- die beteiligte Transition verfügt über freie Schaltkapazität.
- im Nachbereich der beteiligten Transition sind ausreichend Stellenkapazitäten vorhanden.

Der Schaltvorgang eines Schaltereignisses wird beschrieben in folgenden drei Schritten:

- Am Beginn werden die Vorbereichsobjekte, die an der Variablenbelegung gebunden werden, unter Berücksichtigung der Stellenstrukturen und der Kantentypen der Vorbereichsstellen abgezogen bzw. im Fall von Enabling-Kanten kopiert.

- Die von der Transition ausgehenden Objekte werden nach dem Typ der jeweiligen Nachbereichsstelle und dem jeweiligen Kantengewicht produziert.
- Zum Schluss wird die Schaltdauer bestimmt und die Schaltaktion ausgeführt.

Wenn die Schaltdauer größer als Null ist, wird die Schaltkapazität der Transition am Beginn des Schaltens reduziert und müssen ausreichende Plätze für ausgehende Objekte im Nachbereich dieser Transition nach dem Kantengewicht ausgehender Kanten vorbereitet werden. Die Schaltdauer kann Null sein, auf diesen Fall werden ausgehende Objekte direkt auf die Nachbereichsstellen platziert. Am Ende des Schaltens wird die Schaltkapazität der Transition wieder erhöht, die reservierten Plätze freigegeben und die ausgehenden Objekte werden auf die Stellen des Nachbereiches abgelegt.

3.3 Hierarchiekonzepte

Hierarchiekonzepte der THORNs ermöglicht, komplexe Systeme kompakt und übersichtlich zu modellieren. Durch das Hierarchiekonzept kann man die Modellierung eines Systems auf verschiedenen Abstraktionsebenen durchführen und sogar das bietet uns eine Möglichkeit an, die Wiederverwendbarkeit einzelner Modellteile an unterschiedlichen Stellen des Modells zu realisieren.

Bevor das Hierarchiekonzept der THORNs erläutert wird, werden hier zuerst zwei Ansätze der Hierarchiekonzepte präsentiert.

- Der erste Ansatz wurde erst von Genrich, Lautenbach und Thiagarajan diskutiert. Es bezieht sich auf eine einfache Ersetzung eines Knotens durch ein verfeinerndes Unternetz. Durch diese Ersetzung wird die Semantik eines Netzes nicht verändert, d.h. jedes hierarchische Netz kann in ein flaches Netz äquivalent transformiert werden. Die Vorteile des Hierarchiekonzepts liegen darin, solche bekannten Analysemethoden der fundamentalen Netzklasse auf die unbekannteren hierarchischen Netze übertragen zu können.
- Der zweite Ansatz wurde von Cherkasova und Kotov erhoben. Es handelt sich um einen Aufruf der Unternetze von den strukturierten Transitionen. Wenn eine strukturierte Transition sich schaltet, wird die Token im Vorbereich konsumiert und startet anschließend das Unternetz, das mit einer vordefinierten Initialmarkierung versehen wird.

Wenn eine Transition sich in einem Unternetz schalten kann, wird der Austausch der Token über die externe „Share“ Stellen mit der Umwelt ausgeführt. Wenn keine Transitionen sich in dem Unternetz schalten können, terminiert das Unternetz und erzeugt die aufrufende Transition ihre Token im Nachbereich.

Die oben erläuterten Ansätze werden üblich in einfachen Stellen/Transitionen-Netzen angewendet. Um die Übertragung und Erweiterung auf gefärbte Petrinetze zu ermöglichen, werden hier die Konzepte von Huber, Jensen und Shapiro vorgestellt. In diesem Konzept werden fünf Hierarchiekonzepte erläutert: Stellen- und Transitionen-Verfeinerung, Aufruf-Transitionen und Stellen- und Transitionen-Verschmelzung. Der Unterschied zwischen dem zweiten Ansatz und dem Konzept Aufruf-Transitionen liegt darin, dass die Rekursion erlaubt ist, d.h. die aufrufende Transition Token an das Unternetz übergibt und umgekehrt. Die detaillierte Beschreibung dieser fünf Konzepte ist in dem Abschlussbericht des Projekts DNS von Schöf und Sonnenschein 1997 auf der Seite 18 zu finden.

Nachfolgend wird das Hierarchiekonzept der THORN genauer erklärt. Das Hierarchiekonzept der THORN ist eine Kombination von der Aufruftransition und Stellenverschmelzung bzw. den Share-Stellen und klassischer Transitionenverfeinerung. Ein THORN besteht aus einer Menge von Unternetzen, darunter gibt es genau ein Hauptnetz, das keine Stop-Transition enthalten darf. Die Unterneetze können sich gegenseitig mittels von Aufruftransitionen erzeugen. Die Terminierung eines Unternetzes wird mittels Stop-Transitionen durchgeführt. Jedes Unternetz darf keine, eine oder mehrere Aufruf- und Stop-Transitionen haben. Die Kommunikation zwischen den Unternetzen erfolgt über die Share-Stellen. Im Vergleich mit normalen Transitionen werden die Aufruftransitionen anstelle einer Schaltaktion mit dem Namen des Unternetzes versehen, keine Schaltaktion haben die Stop-Transitionen.

Das aufrufende Netz kommuniziert mit dem Unternetz durch eine sogenannte Randstelle, die genau eine korrespondierende Stelle im Unternetz besitzen muss. Diese Stelle wird nach der Art der verbindenden Kante zwischen Eingabe-, Ausgabe- und Einbettungsstellen unterschieden. Die Eingabe- und Ausgabestellen werden durch die ein- und ausgehende Standardkante mit der Aufruftransition verbunden. Die Einbettungsstellen werden mittels eines speziellen Kantentyps Hyperkante mit der Aufruftransition in Beziehung gesetzt werden. Hyperkanten sind die Kante, die ungerichtet sind, aber die kann auch mit der Pfeilsrichtung bezeichnet werden, um die Richtung des Objektflusses zwischen dem aufrufenden Netz und

kapazität der aufrufenden Transition wieder erhöht und die Instanz im Unternetz gelöscht.

4. Verifikation und Validierung

Seit Jahren gewinnt Simulation immer mehr Bedeutung im Bereich Produktion und Logistik. Alle technischen Simulationsmodelle stellen eine starke Vereinfachung der Realität dar, erhöht die Transparenz der Realität, hilft uns komplexe Systeme, die Problemfindung sowie -lösung zu verstehen. Es ist zu beachten, dass „Die Richtigkeit und Übertragbarkeit der Simulationsergebnisse von erheblicher Bedeutung für das weitere Handeln sind“ (Rabe, Spieckermann, Wenzel 2008). Das stellt uns die Aufforderung, ein anforderungsgerechtes Simulationsmodell zu etablieren. Zu den Anforderungen zählen die übereinstimmte Widerspiegelung der Systemumgebung sowie die genaue Modellierung des gewünschten Verhaltens des abgebildeten Systems. Die Gefährdung des fehlerhafteten Simulationsmodells ist klar, z.B. es kann zu den Fehlentscheidungen führen, verursacht hohe Kosten sowie Zeitaufwand usw. Das heißt, das Simulationsmodell sollte regelmäßig kontrolliert werden, um eine sinnvolle Aussage mit ihnen zu treffen. Deswegen sind Verifikation und Validierung ein unentbehrlicher Bestandteil für eine erfolgreiche Simulationsstudie.

4.1 Definitionen und Ziele von Verifikation und Validierung

Verifikation (Wahrheitsbeweis) ist die Überprüfung, ob ein Modell von einer Bearbeitungsart in eine andere Beschreibungsart korrekt transformiert („Ist das Modell richtig?“) wird. Validierung (Gültigkeitsnachweis) ist die kontinuierliche Überprüfung, ob die Modelle das Verhalten des abgebildeten Systems hinreichende genau wiedergeben („Ist es das richtige Modell?“) (Rabe, Spieckermann, Wenzel 2008; Balci 2003; VDI3633 2008). Allerdings lassen sich Verifikation und Validierung in der Praxis schwer trennen (Balci 1998).

Die Zielsetzungen der Validierung und Verifikation fokussieren sich auf drei Aspekte:

- Sachliche Korrektheit
d.h. Das ausgebildete Simulationsmodell soll das untersuchte System hinreichend genau wiedergeben.
- Funktionale Korrektheit

d.h. Das ausgebildete Simulationsmodell enthält die erforderlichen Funktionen, die der Zielstellung entsprechen.

- Technische Korrektheit

d.h. Das ausgebildete Simulationsmodell erfüllt die für die Zielstellung erforderlichen Randbedingungen.

Die Ziele der Validierung und Verifikation werden im Buch „Verifikation und Validierung für die Simulation in Produktion und Logistik“ in folgenden Punkten zusammengefasst (Rabe, Spieckermann, Wenzel 2008):

- V&V soll die Fehler frühzeitig erkennen. Einerseits können die möglich verursachten Kosten und Zeit gespart werden, andererseits können die Fehler leichter erkannt werden.
- V&V soll fundierte und nachvollziehbare Grundlagen für die Entscheidung über die Glaubwürdigkeit des Modells schaffen.
- V&V soll festlegen, dass gewonnene Erkenntnisse vollständig und korrekt in die weitere Modellbildung einfließen.
- V&V soll die richtige Anwendung glaubwürdiger Modelle gewährleisten, um die fehlerfreie Schlussfolgerung daraus abzuleiten.

4.2 Techniken der Verifikation und Validierung (V&V)

In der Literatur gibt es zahlreiche V&V-Techniken. In dieser Arbeit werden nur die Techniken, die im Bereich Produktion und Logistik häufig angewendet werden, genauer beschrieben. Die meisten V&V-Techniken verdeutlichen die Abweichungen zwischen dem tatsächlichen Verhalten und dem erwarteten Verhalten eines Modells. Die Abweichungen können von einer fehlerhaften Übertragung (Verifikation) oder einer fehlerhaften Spezifikation des Modells (Validierung) verursacht werden. Aber die Quelle der Abweichungen ist von untergeordnetem Interesse. Deswegen lassen sich die V&V-Techniken nicht vollständig in Verifikationstechniken und Validierungstechniken unterscheiden (Wenzel et al. 2008).

Ein allgemeines Auswahlvorgehen der geeigneten V&V-Techniken existiert nicht. Um ein Modell richtig zu verifizieren und zu validieren, sollten das Verhalten des Modells, das Ziel des Modells und noch die Verfügbarkeit von Daten unbedingt berücksichtigt werden. In die-

sem Abschnitt wird die Beschreibung der ausgewählten V&V-Techniken zuerst präsentiert. Tabelle 1 gibt Ihnen eine Übersicht der üblich verwendeten V&V Techniken.

Tabelle 1: Deutsche und englische Bezeichnungen von V&V-Techniken (Rabe, Spieckermann, Wenzel 2008, S. 96)

Deutsche Bezeichnung	Englische Bezeichnung
Animation	Animation
Begutachtung	Review
Dimensionstest	Dimensional Consistency Test
Ereignisvaliditätstest	Event Validity Test
Festwerttest	Fixed Value Test
Grenzwerttest	Extreme-Condition Test
Monitoring	Monitoring, Operational Graphics
Schreibtischtest	Desk Checking
Sensitivitätsanalyse	Sensitivity Analysis
Statistische Techniken	Statistical Techniques
Strukturiertes Durchgehen	Structured Walkthrough
Test der internen Validität	Internal Validity Test
Test von Teilmodellen	Submodel Testing
Trace-Analyse	Trace Analysis
Turing-Test	Turing Test
Ursache-Wirkungs-Graph	Cause-Effect Graph
Validierung im Dialog	Face Validity
Validierung von Vorhersagen	Predictive Validation
Vergleich mit anderen Modellen	Comparison to other Models
Vergleich mit aufgezeichneten Daten	Historical Data Validation

4.2.1 Animation

Die hauptsächliche Stärke der Animation liegt in der Überprüfung des Modellverhaltens in ausgewählten Modellabschnitten über kurze Zeiträume (Carson 2002). Wenn mit anderen Techniken unerwartete Ergebnisse in einem bestimmten Zeitraum festgestellt werden, können mittels Animation schnell die Fehler im Detail lokalisiert werden. Dazu bietet Animation die visuelle Möglichkeit an, die unerwarteten Ergebnisse durch Beobachtung des Modellverlaufs zu verstehen.

Allerdings hat Animation auch einige Nachteile, wie z.B. Vortäuschung von Detailtreue, Grenzen visueller Erfassbarkeit und Überbewertung untypischer Modellzustände. Dann kommt die Frage, wie man die Wirksamkeit der Animation verbessern kann? Hier werden einige geeignete Maßnahmen vorgestellt (Swider et al. 1994):

- bewusster Entwurf der Animationsdarstellung für die Validierung

- (d.h. mit dem Ziel, das Systemverhalten möglichst transparent darzustellen)
- in den Teilbereichen des Modells die Animation begrenzt verwenden
 - die Animation mit angemessener Geschwindigkeit durchführen, damit die Beobachter die Kernvorgänge verfolgen können

4.2.2 Begutachtung

Im Rahmen der Begutachtung wird das Simulationsmodell gemäß den vereinbarten Zielen und Randbedingungen ausgewertet, ob das die Realität richtig nachbildet oder ob das richtig verläuft. Die Begutachtung schließt das Management auf Auftraggeber- und Auftragnehmerseite ein (Balci 1998). Dazu kann die Begutachtung mit dem Schreibtischtest oder mit dem strukturierten Durchgehen kombiniert eingesetzt werden (Rabe, Spieckermann, Wenzel 2008).

Um die Begutachtung zielorientiert anzuwenden, werden einige Kriterien festgelegt, in Tabelle 2 sind die Kriterien für den Erfolg einer Begutachtung mit Bezug zu den V&V-Kriterien zu sehen.

Tabelle 2: Kriterien für den Erfolg einer Begutachtung mit Bezug zu den V&V-Kriterien (Rabe, Spieckermann, Wenzel 2008, S. 97)

Kriterium für Erfolg der Begutachtung	V&V-Kriterium
Eignung der Ziel- und Systembeschreibungen	Eignung
Angemessenheit der getroffenen Annahmen	Genauigkeit, Machbarkeit
Einhaltung der vereinbarten Richtlinien und Standards	Verständlichkeit, Konsistenz
Angemessenheit und Effizienz der verwendeten Modellierungstechniken	Eignung, Verständlichkeit, Machbarkeit
Grad der Übereinstimmung des Modells mit der Realität	Vollständigkeit, Genauigkeit, Aktualität
Strukturiertheit des Modells	Verständlichkeit, Vollständigkeit, Konsistenz
Vollständigkeit des Modells	Vollständigkeit, Machbarkeit
Qualität der Dokumentation	– alle Kriterien –

4.2.3 Ereignisvaliditätstest & Festwerttest

Bei dem Ereignisvaliditätstest wird das Auftreten von Ereignissen im Simulationsmodell mit der realen Welt verglichen. Es handelt sich um sowohl zusammenhängende Ereignisse als auch einzelne Ereignisse in ihrer zeitlichen Reihenfolge (Hermann 1967; Balci 1990). Um den Ereignisvaliditätstest objektiv durchzuführen, sollten zuerst Hypothesen aufgestellt werden, z.B. wie oft ein bestimmtes Ereignis pro Tag eintreten wird, oder in welcher Abfolge bestimmte Ereignisse auftreten werden. Danach werden die Hypothesen im Modell überprüft, ob die unerwarteten Abweichungen existieren.

Festwerttest kann nur in einem deterministischen Modell benutzt werden, z.B. Um den Festwert auszuführen, müssen statt einer Verteilung für die Montagezeit konstante Werte ohne Schwankungen gegeben werden. Analog zum Ereignisvaliditätstest sollte beim Festwerttest zuerst eine Hypothese formuliert werden, wie sich das deterministische Modell verhalten wird. Danach wird diese formulierte Hypothese bewertet, wenn die Hypothese zutrifft, wird die Glaubwürdigkeit dieser Technik erhöht, sonst ist eine Fehlerüberprüfung notwendig. Festwerttest ist besonders geeignet für einen Arbeitsablaufplan. Mittels des Festwerttests kann man festlegen, ob ein Arbeitsablauf nach dem Arbeitsplan glatt läuft. Die Beschränktheit des Festwerttests ist offensichtlich, dass unter Berücksichtigung der stochastischen Einflüsse diese Technik nur teilweise in einem Modell benutzt werden kann.

4.2.4 Monitoring

Unter Monitoring versteht man „die Werte von Zustandsgrößen und Variablen (z.B. Auslastung, Durchlaufzeit, Pufferbelegung) während des Simulationslaufes graphisch angezeigt und überprüft“ (Sargent 2005). Es wird geprüft, ob die in der Graphik dargestellten Werte und die aktuelle Situation des Simulationsmodells konsistent sind. Es gibt zwei Verfahren zur Darstellung, ein ist „Werte zum Zeitpunkt“, ein anderes ist „Werte im Zeitverlauf“. Bei dem ersten Verfahren werden die aktuellen Werte einer Größe zu einem bestimmten Zeitpunkt graphisch dargestellt. Unterschiedliche Arten der Darstellung wie z.B. Kreisdiagramm, Balkendiagramm stehen uns zur Verfügung. Mit dem zweiten Verfahren werden die beobachteten Werte für einen bestimmten Zeitraum über der Zeit mittels Kurven oder Balkengraphiken dargestellt.

Monitoring hat auch gewisse Begrenztheiten, es kann nur in den räumlichen und logischen Teilmodellen eingeführt werden. Ihre Stärke ist auch ersichtlich, z.B. mit Hilfe einer Kurve kann man das fehlerhafteste Kurventeil wie unerwartete Ausschläge schneller auffinden.

4.2.5 Schreibtischtest & Validierung im Dialog & Sensitivitätsanalyse

Schreibtischtest bezieht sich auf eine sorgfältige Überprüfung der eigenen Arbeit nach den Kriterien Vollständigkeit, Korrektheit, Konsistenz und Eindeutigkeit (Balci 1998). Weil diese Überprüfung von selbst durchgeführt wird, kann es unvermeidbar auftreten, dass die eigenen Fehler trotz nochmaligen Durchgehens schwierig aufgefunden werden können. Deshalb benutzt man in der Praxis diese Technik häufig mit Validierung im Dialog zusammen.

Bei der Validierung im Dialog wird das Modell mit den Fachexperten zusammen untersucht. Diese Technik gilt nicht nur für das ausführbare Modell, sondern auch für das Konzeptmodell. Aufgrund ihrer fachlichen Erfahrungen können die Fehler schneller aufgefunden werden, z.B. ungewöhnliche Daten, ein abnormaler Verlauf usw. Ein Dialog ähnlich wie ein Brainstorming zwingt den Experten, ihren eigenen Verantwortungsbereich des Simulationsmodells nochmals zu überdenken. Möglicherweise führt es zur Auffindung von versteckten Fehlern. Jedoch hat Validierung im Dialog auch ihre Begrenztheit. Wenn die Experten mit dem relevanten Modell oder den Modellteilen nicht vertraut sind, können die Experten ihre auf dem vorher schon bekannten Modell basierten Kenntnisse falsch auf das neue Modell übertragen.

Die Sensitivitätsanalyse untersucht, wie deutlich die Ausgabegrößen von Veränderungen der Eingabegrößen abhängig sind. Es soll beachtet werden, dass die Richtung der Wirkung mit den Beobachtungen in der Realität übereinstimmen muss.

4.2.6 Statistische Techniken

Durch statistische Techniken wird das Modell ausgewertet, mit welcher Sicherheit die Ausgabegrößen des Modells das Verhalten des realen Systems darstellen. Darüber hinaus kann mit Hilfe der statistischen Techniken die Gültigkeit der im Simulationsmodell verwendeten Verteil-

lung der Eingabegrößen geprüft werden (Law 2007), z.B. man kann Chi-Quadrat-Test als Anpassungstest anwenden, um die Gültigkeit der Verteilung der Eingabegrößen zu prüfen. Das konkrete Anwendungsbeispiel von Chi-Quadrat-Test wird in Kapitel 6 genauer erläutert. Um die Sicherheit der Ausgabegrößen zu kontrollieren, wird die Bestimmung der Konfidenzintervalle empfohlen.

4.2.7 Strukturiertes Durchgehen

Das strukturierte Durchgehen kann man in dem Konzeptmodell, formalem Modell und ausführbarem Modell benutzt. Der Vorgang wird von den Personen durchgeführt, die sich an der Formulierung der Phasenergebnisse nicht beteiligt. Die Simulationsfachleute prüfen die einzelnen Dokumente mit den Fachverantwortlichen zusammen, um die möglichen Fehler, Unklarheiten oder Probleme zu finden. Einige Teile der Dokumenten enthalten die Annahmen, die sowohl über die Daten und als auch über logische Zusammenhänge und Abfolgen handeln kann. Das Ziel dieser Technik ist auch eine Übereinstimmung mit den Sachverhalten. Dank der gemeinsamen Abstimmung wird die Glaubwürdigkeit der Phasenergebnisse wesentlich erhöht.

4.2.8 Test von Teilmodellen

Dieser Technik wird angewendet bei einem Modell, das hierarchisch in Teilmodelle angeordnet ist. Es gilt für die Phasenergebnisse des Modellierungsprozesses wie Konzeptmodell, formales Modell und ausführbares Modell. Es wird geprüft, „ob die korrespondierenden Komponenten von realem System und Modell ausreichend gut übereinstimmen“ (Berchtold et al. 2002). Bei den nicht-ausführbaren Modellen wird der Test von Teilmodellen mit anderen Techniken kombiniert, z.B. bei einem hierarchischen Petri-Netz kann ein strukturiertes Durchgehen mit dem Test von Teilmodellen untersucht werden, die Arbeitsweise wird einfacher und effizienter. Für das ausführbare Modell werden die Eingabe- und die Ausgabedaten der Teilmodelle während der Modellausführung notiert, um das Verhalten des Teilmodells zu analysieren und zu validieren. Hier kann diese Technik auch mit anderen Techniken kombiniert werden, wie z.B. Im Vergleich mit anderen Modellen. Wichtig ist, dass die Verifikation und Validierung sämtlicher Teilmodelle nicht die Verifikation und Validierung des gesamten

Modells ersetzen können.

4.2.9 Trace-Analyse & Vergleich mit anderen Modellen

Bei der Trace-Analyse wird das Verhalten einzelner Objekte im ausführbaren Modell verfolgt und dabei das logische Verhalten und die Plausibilität überprüft (Rabe, Spieckermann, Wenzel 2008). Alle Ereignisse werden in einer „Trace-Datei“ zusammengefasst, wenn man ein bestimmtes Objekt untersuchen will, können die bezüglichen Ereignisse gefiltert und bewertet werden. Ein Trace beinhaltet die zusammenhängenden Informationen der Objekte aus dem Modell wie z.B. den Zustand des Objektes, den Ort des Objektes, den Bewegungsweg des Objektes, den Zustand der dem Objekt zugeordneten Ressourcen usw. Um die Trace-Analyse objektiv durchzuführen, werden zunächst die Hypothesen aufgestellt und anschließend mit der Trace-Analyse überprüft, ob man die Hypothesen ablehnt oder beibehält.

Bei der Technik Vergleich mit anderen Modellen werden die Ergebnisse des ausführbaren Modells unter den gleichen Eingabegrößen mit den Ergebnissen eines einfacheren Modells verglichen. Diese Technik kann nur für ein ausführbares Modell angewendet werden (Robinson 2004). Die beiden Modelle werden unabhängig voneinander verifiziert und validiert. Es ist unvermeidbar, dass die zum hohen Aufwand führt. Durch den Vergleich können die entstehenden Abweichungen analysiert und ausgewertet werden. In der Praxis wird die Technik Vergleich mit anderen Modellen mit anderen Techniken wie z.B. Festwerttest, Grenzwerttest kombiniert eingesetzt, um das Verhalten des Simulationsmodells zu vereinfachen.

In Tabelle 3 werden alle häufig im Bereich Produktion und Logistik angewendeten V&V-Techniken sowie ihre kurzen Beschreibungen zusammengefasst.

Tabelle 3: Kurzbeschreibungen der V&V-Techniken (Eley 2012, S. 16)

Technik	Beschreibung
Animation	Zeitliche Abläufe in dem Modell werden zwei- oder dreidimensional graphisch dargestellt. Dabei kann nur beobachtet werden, ob die Abläufe im Modell in dem tatsächlich betrachteten Modell und Zeitabschnitt plausibel sind oder ob es Unterschiede zum realen System gibt.
Begutachtung	Die Begutachtung durch das Management von Auftraggeber und -nehmer soll klären, ob die Simulationsstudie in Übereinstimmung mit den vereinbarten Zielen und Randbedingungen verläuft.

Dimensionstest	Aufdecken von konzeptionellen Fehlern oder Fehlern bei der Entwicklung von Formeln durch die Nachrechnung der Dimension der Formel.
Ereignisvaliditätstest	Vergleich des Auftretens von Ereignissen im Simulationsmodell mit der Realität, z. B. Anzahl an Aufträgen je Schicht.
Festwerttest	Ersetzen aller stochastischen Größen durch feste Werte, z. B. Ersetzen der Bearbeitungszeiten in einer Montagelinie durch den Mittelwert. Somit kann der Durchsatz der Linie leicht berechnet werden.
Grenzwerttest	Eingabewerte werden so gesetzt, dass das Verhalten des Modells besser vorhersagbar wird, z. B. werden alle Losgrößen in einem Produktionssystem auf den kleinsten vorgesehenen Wert gesetzt.
Monitoring	Die Werte von Zustandsgrößen und Variablen (z. B. Auslastung, Durchlaufzeit, Pufferbelegung) werden während des Simulationslaufes graphisch angezeigt und überprüft.
Schreibtischtest	Sorgfältige Überprüfung der eigenen Arbeit in Bezug auf Vollständigkeit, Korrektheit, Konsistenz und Eindeutigkeit durch die Erstellung einer Dokumentation.
Sensitivitätsanalyse	Eingabewerte des Modells werden verändert und die Auswirkungen auf Ausgabewerte bestimmt. Die Richtung der Auswirkung muss mit Beobachtungen in der Realität übereinstimmen
strukturiertes Durchgehen	Die Projektbeteiligten gehen jede Anweisung des Programms gemeinsam durch, bis alle von der Richtigkeit der Anweisungen überzeugt sind.
Test der internen Validität	Mit unterschiedlichen Startwerten der Zufallszahlengeneratoren werden bei gleichbleibenden Eingabewerten Simulationsläufe durchgeführt.
Trace-Analyse	Es wird das Verhalten einzelner Objekte im ausführbaren Modell verfolgt und dabei das logische Verhalten und die Plausibilität überprüft.
Turing-Test	Experte, die das reale System kennen, legen eine Reihe von Ausgabewerten aus der Simulation und den Beobachtungen des realen Systems vor.
Ursache-Wirkungs-Graph	Die verschiedenen Ursachen und Wirkungen werden in einem Graphen dargestellt, der die logischen Beziehungen verdeutlicht.
Validierung im Dialog	Das Modell wird gemeinsam mit Fachexperten diskutiert, die Kenntnisse über das reale System besitzen.
Validierung von Vorhersagen	Das Modell wird zunächst genutzt, um eine Vorhersage zu treffen. Anschließend wird die Richtigkeit dieser Vorhersage am realen System überprüft.
Vergleich mit anderen Modellen	Die Ergebnisse des Modells für bestimmte Eingabewerte werden mit den Ergebnissen eines anderen, i. d. R. einfacheren Modells bei gleichen Eingabewerten verglichen.
Vergleich mit aufgezeichneten Daten	Ein- und Ausgabewert eines real existierenden Systems, die in der Vergangenheit aufgezeichnet wurden, werden zum Test verwendet.

4.3 Bewertung der Technik bzgl. der Anwendbarkeit

Kapitel 4.2 bietet eine reiche Auswahl an den V&V-Techniken an. Um die richtigen und effektiven Techniken auszuwählen, wird ihre Anwendbarkeit nach Zielbeschreibung und fünf ver-

schiedenen Phasenergebnissen bewertet. Mittels der Tabelle wird es empfohlen, welche Techniken zu welchen Phasenergebnissen passen.

Abbildung 15 verdeutlicht das Simulationsvorgehensmodell mit V&V. Das gesamte Vorgehen wird in fünf Phasen Aufgabedefinition, Systemanalyse, Modellformalisierung, Implementierung und Experimente und Analyse eingeteilt. Jede Phase wird ihrem entsprechenden Phasenergebnis zugeordnet, wie Aufgabenspezifikation, Konzeptmodell, Formales Modell, Ausführbares Modell und Simulationsergebnisse. Das Phasenergebnis kann Modelle und Dokumente oder die Kombination von beiden sein. Es ist zu beachten, dass Zielbeschreibung kein Phasenergebnis ist, sondern die Ausgangsbasis einer Simulationsstudie ist (Rabe, Spieckermann, Wenzel 2008).

Die Phasen „Datenbeschaffung“ und „Datenaufbereitung“ werden nicht dem Prozess der Modellierung zugeordnet. Die Position von den Rohdaten in Abbildung 15 bedeutet nicht, dass die Rohdaten erst nach dem Konzeptmodell gesammelt werden, sondern es unnötig ist, vor der Entstehung des formalen Modells, vollständige Rohdaten zu versammeln. Das Vorgehensmodell gibt die Informationen, dass die Datenbeschaffung die Ergebnisse der Aufgabedefinition benutzt und die Datenaufbereitung die Datenbeschaffung voraussetzt.

Es ist in Abbildung 15 auffällig, dass V&V der Daten und Modelle während der ganzen Simulationsstudie phasenbegleitend durchgeführt werden, weil alle Phasenergebnisse und die Zielbeschreibung überprüft werden müssen. Aus dem Vorgehensmodell kann man deutlich sehen, dass die Zielbeschreibung sowie alle Phasenergebnisse die tatsächlichen Untersuchungsobjekte der V&V sind, nicht die Phasen selbst. Deswegen werden im Folgenden die Zielbeschreibung sowie die von der Modellierung abhängigen Phasenergebnisse kurz vorgestellt.

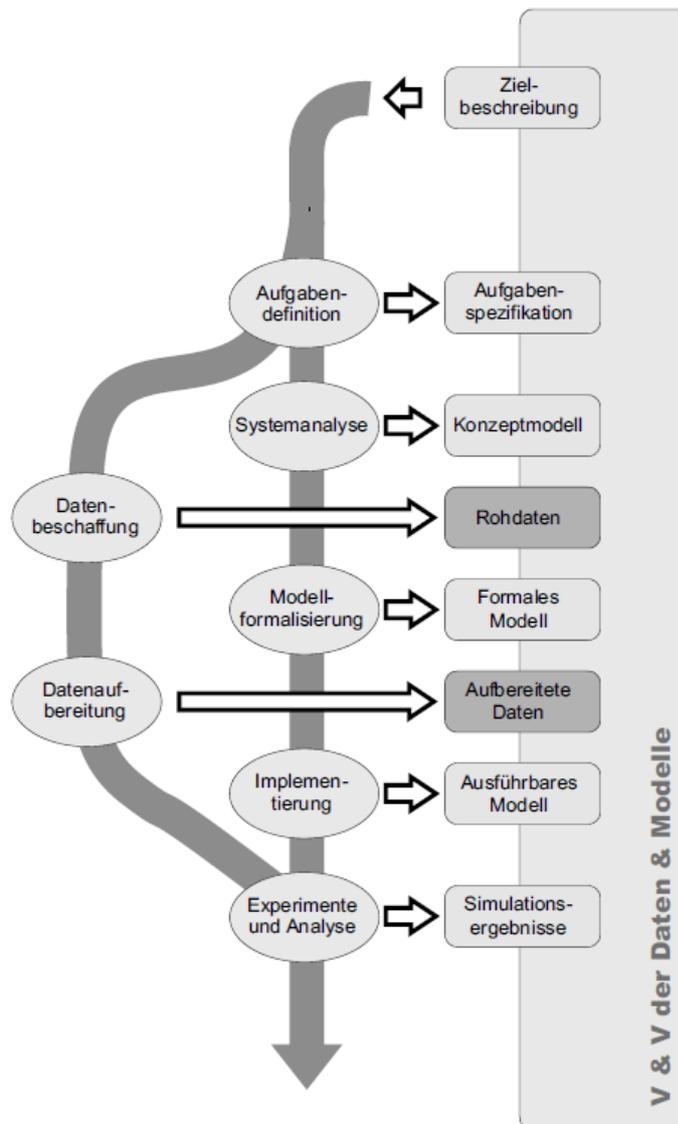


Abbildung 15: Vorgehensmodell bei der Simulation mit V&V
(Rabe, Spieckermann, Wenzel 2008, S. 5)

1. Zielbeschreibung

Die Dokumente der Zielbeschreibung beinhaltet eine Beschreibung der Ausgangssituation von dem untersuchten Projekt, eine Erklärung des inhaltlichen Projektumfanges sowie die Darstellung der unternehmensspezifischen Randbedingungen wie z.B. in Bezug auf die Zeit, Kosten, Organisation usw.

2. Aufgabenspezifikation

Aufgabenspezifikation ist das erste Dokument, das im Rahmen der Simulationsstudie entsteht. Aufgabenspezifikation lässt die oben erläuterte Zielbeschreibung konkret werden. Die Doku-

mente von der Aufgabenspezifikation werden durch alle Beteiligten des Projekts festgestellt, wie z.B. die konkrete Darstellung des Problems. Allerdings soll man beachten, dass die abgestimmte Aufgabenspezifikation unter vorgegebenen Ressourcen durchführbar ist. Dazu bildet Aufgabenspezifikation die Basis für alle nachfolgenden Projektschritte.

3. Konzeptmodell

Bei der Systemanalyse werden die Dokumente bezüglich der Zielsetzungen, Eingaben, Ausgaben, Elemente und Beziehungen, Annahmen und Vereinfachungen des Modells erarbeitet. Konzeptmodell ermöglicht einen Übergang von der Beschreibung, welche Aufgabe gelöst werden soll, zu der von aller Beteiligten abgestimmten Beschreibung, z.B. wie die Aufgabe erledigt werden kann. Außerdem werden der Umfang des Modells sowie die erforderliche Detaillierung im Konzeptmodell dargestellt.

4. Formales Modell

Das Konzeptmodell wird in dieser Phase ganz oder teilweise formalisiert. Die Elemente und die Wechselwirkungen zwischen den einzelnen Elementen werden auf Basis der Ergebnisse des Konzeptmodells weiter entwickelt. Das Ziel des formalen Modells liegt darin, dass das erworbene formale Modell ohne weitere Analyse und ohne ergänzende Abstimmung mit den Fachexperten von den Simulationsfachleuten und Softwareexperten ausgeführt werden kann (Rabe, Spieckermann, Wenzel 2008).

5. Ausführbares Modell

In der Phase Implementierung wird das formale Modell unter Beachtung der Eigenschaften des verwendeten Simulationswerkzeugs von den Simulationsfachleuten oder Softwareexperten durchgeführt. Die Dokumente des ausführbaren Modells basiert auf den Dokumenten der vorherigen Modellierungsstufen und die Aspekte der simulationswerkzeugspezifischen Umsetzung werden den Dokumente dieser Phase hinzugefügt.

6. Simulationsergebnisse

Bei der Phase „Experimente und Analyse“ werden das ausführbare Modell und die aufbereiteten Daten zusammengefasst. Die Inhalte dieser Phase beinhalten die Beschreibung der Annahmen und Vereinfachungen, die Einfluss auf die Ergebnisse haben können, und der Experimentplanung sowie Darstellung der erzielten Ergebnisse. Unter Verwendung von Experimentplänen werden die zuvor aufgestellten Hypothesen angenommen oder widerlegt,

anschließend ergeben sich aus den quantitativen Ergebnissen die nützlichen Schlussfolgerungen für das reale System.

Phasenergebnisse des Modellierungsprozesses

V&V-Techniken	Zielbe-schreibung	Aufgaben-spezifikation	Konzept-modell	Formales Modell	Ausführbares Modell	Simulations-ergebnisse	Roh-daten	Aufbereitete Daten
Animation					◆	◆		
Begutachtung	◆	◆	◆	◆	◆	◆	◆	◆
Dimensionstest				◆	◆	◆	◆	◆
Ereignisvaliditätstest					◆			
Festwerttest				◆	◆	◆		
Grenzwerttest				◆	◆	◆		
Monitoring					◆	◆		◆
Schreibtischtest	◆	◆	◆	◆	◆	◆	◆	◆
Sensitivitätsanalyse					◆	◆		◆
Statistische Techniken					◆	◆	◆	◆
Strukturiertes Durchgehen	◆	◆	◆	◆	◆	◆	◆	◆
Test der internen Validität					◆	◆		
Test von Teilmodellen			◆	◆	◆			
Trace-Analyse					◆			
Turing-Test					◆			◆
Ursache-Wirkungs-Graph			◆	◆	◆			
Validierung im Dialog	◆	◆	◆	◆	◆	◆	◆	◆
Validierung von Vorhersagen					◆			
Vergleich mit anderen Modellen					◆	◆		
Vergleich mit aufgezeichneten Daten					◆			

Abbildung 16: Verwendbarkeit von V&V-Techniken im Verlauf der Simulationsstudie (Rabe, Spieckermann, Wenzel 2008, S. 113)

Abbildung 16 veranschaulicht die Verwendbarkeit von V&V-Techniken für verschiedene Phasenergebnisse, für welche Phasenergebnisse welche Techniken angewendet werden können. Es ist auffällig, dass die Techniken wie z.B. Validierung im Dialog, strukturiertes Durchgehen, Ursache-Wirkungs-Graph, Begutachtung, die mit den Fachexperten zusammen ausgeführt werden, für jedes Phasenergebnis eingesetzt werden können. Die Techniken wie Animation, Ereignisvaliditätstest, Monitoring, Trace-Analyse sind auf Basis des ausführbaren Modells anwendbar. Diese Abbildung gibt uns nur einen Vorschlag. Die verwendeten Techniken für ein Modell sind abhängig von der tatsächlichen Situation und werden von den beteiligten Simulationsfachleuten zusammen entschieden.

5. Beurteilung der Automatisierungsmöglichkeiten der V&V Techniken

Im letzten Kapitel wurde die V&V Technik schon detailliert erklärt. Aufgrund der hohen Anforderung wie z.B. Zeitersparnis, Personalsparnis wird untersucht werden, welche V&V Techniken automatisiert oder teil-automatisiert arbeiten können. Unter „Automatisierung“ versteht man die Übertragung von Funktionen des Produktionsprozesses, insbesondere Prozesssteuerungs- und -regelungsaufgaben vom Menschen auf künstliche Systeme (Voigt 2016). Je nach Umfang der Übernahme von Steuerungs- und Regelungsaufgaben durch die Maschine wird von Teil- oder Vollautomatisierung gesprochen (Voigt 2016). Es bedeutet, die Maschinen arbeiten selbstständig, dadurch wird die menschliche Kommunikation oder Kontrolle vermindert.

5.1 Automatisierungsgründe und -ziele

Für eine Automatisierung gibt es vielerlei Gründe. An oberster Stelle steht dabei grundsätzlich der wirtschaftliche Aspekt. Die Automatisierung der V&V lohnt sich besonders bei den komplexen Simulationsmodellen. Wenn ein komplexes Simulationsmodell vorhanden ist und die Automatisierung der V&V technisch machbar ist, kann die Automatisierung zu einer erheblichen Einsparung von den Personalkosten sowie Zeitersparnis führen. Dazu werden die Mitarbeiter durch die Automatisierung nicht nur körperlich sondern auch geistig entlastet.

Andererseits ist oft in der Realität der einzelne Mitarbeiter sogar der einzelne Fachexperte nicht in der Lage, die bestehenden Sachzusammenhänge oder die logischen Beziehungen in einem komplexen Modell zu überblicken. In dieser Situation ist der Einsatz der Automatisierung sinnvoll, um das komplexe Modell zu beherrschen.

Es ist bekannt, dass die Automatisierung der Produktionsprozesse im Allgemeinen zu einer Verbesserung der Zuverlässigkeit führt. Beispielsweise treten weniger Produktionsausfälle auf, da menschliche Fehler in der Bedienung ausgeschlossen werden. Solche Vorteile erhält man auch bei der Automatisierung der V&V Techniken. Aufgrund der Standardisierung der Automatisierung der V&V bleibt die Qualität der V&V auf einem konstanten Niveau.

Zusammenfassend wird an dieser Stelle eine Auflistung von manueller und automatischer V&V geleitet (Feja, Witt, Speck 2013).

Manuelle V&V

- zeitaufwendig, fehleranfällig und bei komplexen Modellen nicht praktikabel
Trotz des hohen Detaillierungsgrads und Flexibilitätsgrads der manuellen V&V ist es unvermeidbar, dass die manuelle V&V deutlich länger dauert als automatische V&V. Deswegen ist manuelle V&V nicht geeignet für komplexe Modelle. Wegen der Einbeziehung von der menschlichen Arbeit ist die manuelle V&V fehleranfällig.
- bei einigen speziellen Fällen besser als automatische V&V
In einigen Situationen (z.B. hoher Detaillierungsgrad oder neue menschliche Ideen werden in dem V&V-Prozess gefordert) ist manuelle V&V besser geeignet als automatische V&V.

Automatische V&V

- geringere Fehleranfälligkeit, Zeitersparnis, Kostenersparnis
Automatische V&V arbeitet ordentlich nach dem zuvor schon festgelegten Plan, deshalb ist automatisch V&V im Vergleich mit manueller V&V geringer fehleranfällig, zeitsparend und kostensparend.
- Wiederholbarkeit
Die automatische V&V kann beliebig oft wiederholt werden.
- aber kein vollständiger Ersatz für manuelle V&V
Wie in manueller V&V schon erwähnt, gibt es einige bestimmte Fälle, in denen manuelle V&V besser geeignet ist.
- eindeutige Kommunikationsgrundlage, Softwarelesbarkeit
Die automatische V&V hat eine eindeutige Kommunikationsgrundlage. Dazu soll das Modell formalisiert werden, um das Modell von der verwendeten Software lesen zu können.

Die Ziele der Automatisierung der V&V Techniken werden durch folgende zwei Eigenschaften ausgedrückt werden (Schnieder 1999).

- Wirtschaftlichkeit
Man wird durch die vorher festgelegten Soll-Werte die Wirtschaftlichkeit (z.B. Produktivität, Kosten, Zuverlässigkeit, Material- und Energieeinsatz usw.) eines Systems bewerten, die durch die Automatisierung steigen kann. Zum Beispiel: durch die Animati-

on werden die bestehenden Fehler schneller aufgefunden als die anderen manuellen V&V-Techniken. Dieser Prozess ist kostengünstig, wenn man die Fehlerstellen schon gefunden hat, können die anderen kostenaufwändigen Techniken eingeführt werden, um weitere V&V-Prozesse durchzuführen.

– Produktqualität

Um die V&V automatisch durchzuführen, wird zunächst der gesamte V&V Prozess standardisiert, beispielsweise mittels Software-Code, d.h. die möglichen von den Menschen verursachten V&V-Lücken können vermieden werden. Der Grad der Standardisierung bietet die Grundlage der hohen Prozessergebnisse d.h. der Produkte an.

5.2 Evaluation der Automatisierungsmöglichkeit bzgl. V&V Techniken

Tabelle 4: Automatisierungsgrad von den V&V-Techniken

Techniken \ Automatisierungsgrad	automatisch	teil-automatisch	manuell
Animation	X		
Begutachtung			X
Ereignisvaliditätstest		X	
Festwerttest		X	
Grenzwerttest		X	
Monitoring	X		
Schreibtischtest			X
Statische Techniken		X	
Strukturiertes Durchgehen			X
Test der internen Validität	X		
Test von Teilmodellen		X	
Trace-Analyse	X		
Turing-Test			X
Ursache-Wirkungs-Graph		X	
Validierung im Dialog			X
Validierung von Vorhersagen			X
Vergleich mit aufgezeichneten Daten			X

Aus Tabelle 4 kann man sehen, dass insgesamt vier V&V Techniken automatisch sind. Sie sind Animation, Monitoring, Test der internen Validität und Trace-Analyse. Durch die Animation kann man eine realitätsnahe Darstellung erhalten und dank der Computer-Technik kann die Animation automatisch ausgeführt werden und die Fehlerstellen sofortig erkannt werden. Monitoring macht die Werte von Zustandsgrößen und Variablen graphisch übersehbar und überprüft sogar diese Werte während der Simulationszeit, ihre Durchführung kann mit Hilfe einer Simulationssoftware realisiert werden. Ähnlich wie Animation und Monitoring kann der Test der internen Validität und Trace-Analyse mittels der Software, der Zufallszahlengeneratoren oder der simulierten Anlagen automatisch durchlaufen.

Zu teil-automatischen V&V Techniken zählen Ereignisvaliditätstest, Festwerttest, Grenzwerttest, Statische Techniken, Test von Teilmodellen sowie Ursache-Wirkungs-Graph. Beispielsweise ist die Grapherstellung der Beziehung zwischen Ursachen und ihren Wirkungen durch Fachexperten manuell. Anschließend kann aus dem Ursache-Wirkungs-Graph eine Entscheidungstabelle abgeleitet werden. Mit dieser Entscheidungstabelle können die Verifikation und die Validierung für das ausführbare Modell automatisch durchgeführt werden. Deswegen wird hier Ursache-Wirkungs-Graph den teil-automatischen V&V Techniken zugeordnet.

Die bleibenden Management-Techniken wie Begutachtung, Schreibtischtest, Strukturiertes Durchgehen, Turing-Test, Validierung im Dialog, Validierung von Vorhersagen und Vergleich mit aufgezeichneten Daten gehören zu den manuellen V&V Techniken. Diese Technik kann nicht ohne menschliche Einflüsse ausgeführt werden.

6. Evaluation der V&V Techniken

In diesem Kapitel wird zunächst die generelle Evaluation der Anwendung der V&V Techniken auf THORNs erläutert. Dann wird in Abschnitt 6.1 das entworfene Modell eingeführt werden. Anhand dieses Modells werden die ausgewählten V&V Techniken ausgewertet. Abschließend werden die vernünftigen Vorschläge zur Auswahl der geeigneten V&V Techniken für THORNs abgeleitet.

Wie im Kapitel 4 aufgezeigt, ist die Begutachtung eine sehr hilfreiche Technik für die V&V auf THORNs. Die Mitarbeiter sollen zuerst unbedingt eine geeignete Systembeschreibung geben. Anschließend sollen die Ziele und die Randbedingungen des Modells festgelegt werden. Wie z.B. die Frage, ob das Modell die Realität richtig und vollständig widerspiegelt, ob das Modell ausführbar ist, ob das ganze Modell in der vorgegebenen Zeit erledigt werden kann, ob die komplexe hierarchische Struktur des Modells richtig vereinfacht wird usw.

Um eine richtige V&V für THORNs durchzusetzen, soll man auf jeden Fall die Eigenschaften von THORNs (z.B. Lebendigkeit, Beschränktheit, Erreichbarkeit, Sicherheit) als erstes analysieren und noch weiter überprüfen. Dank der Vielfältigkeit der Analysealternativen von THORNs werden uns viele Methoden angeboten. Für das dynamische Verhalten des THORNs wird die V&V mit Hilfe der Erreichbarkeitsgraph durchgeführt. Um die statischen strukturellen Eigenschaften in THORNs zu verifizieren und zu validieren, kann man mit der Invarianten-Methode verwenden. Genauer gesagt, gehören solche Überprüfungen zur Verifikation, um die Spezifikationsverletzungen aufzufinden.

Eine Kombination zweier oder mehrerer V&V Techniken tauchen sehr oft bei der Anwendung auf. Zum Beispiel: eine Kombination von der Begutachtung mit anderen Management-Techniken wie der Schreibtischtest, Validierung im Dialog und Ursache-Wirkung-Graph ist auch sehr empfehlenswert für das THORNs.

Für den zeitlichen Aspekt des THORNs kann man den Festwerttest und noch den Grenzwerttest in einer deterministischen THORNs benutzen, um das Verhalten des formalen Modells oder des ausführbaren Modells gemäß den Daten quantitativ zu analysieren und auszuwerten.

Für den hierarchischen Aspekt des THORNs wird hier die Technik Test von Teilmodellen empfohlen. Man kann jeden einzelnen Teil des THORNs detailliert verifizieren und validieren. Hier soll mehr Aufmerksamkeit geschenkt werden, dass die V&V sämtlicher Teilmodelle die V&V des Gesamtmodells nicht ersetzt kann. Um die V&V für ein komplexes hierarchisches THORNs-Modell einfacher und effizienter durchführen zu können, kann man auch eine äquivalente Vereinfachung durchsetzen. Auf diese Weise wird das Verständnis über das Systemmodell verbessert und die Effizienz der V&V gesteigert. Für ein nicht ausführbares Modell muss ein strukturiertes Durchgehen mit dieser Technik kombiniert angewendet werden.

Darüber hinaus kann ein THORNs-Modell mittels ihrer eigenen speziellen Simulationssoftware als Animation oder Monitoring verifiziert und validiert werden. Dieser V&V Prozess arbeitet selbst automatisch.

6.1 Erstellen eines auf THORNs basierenden Simulationsmodells

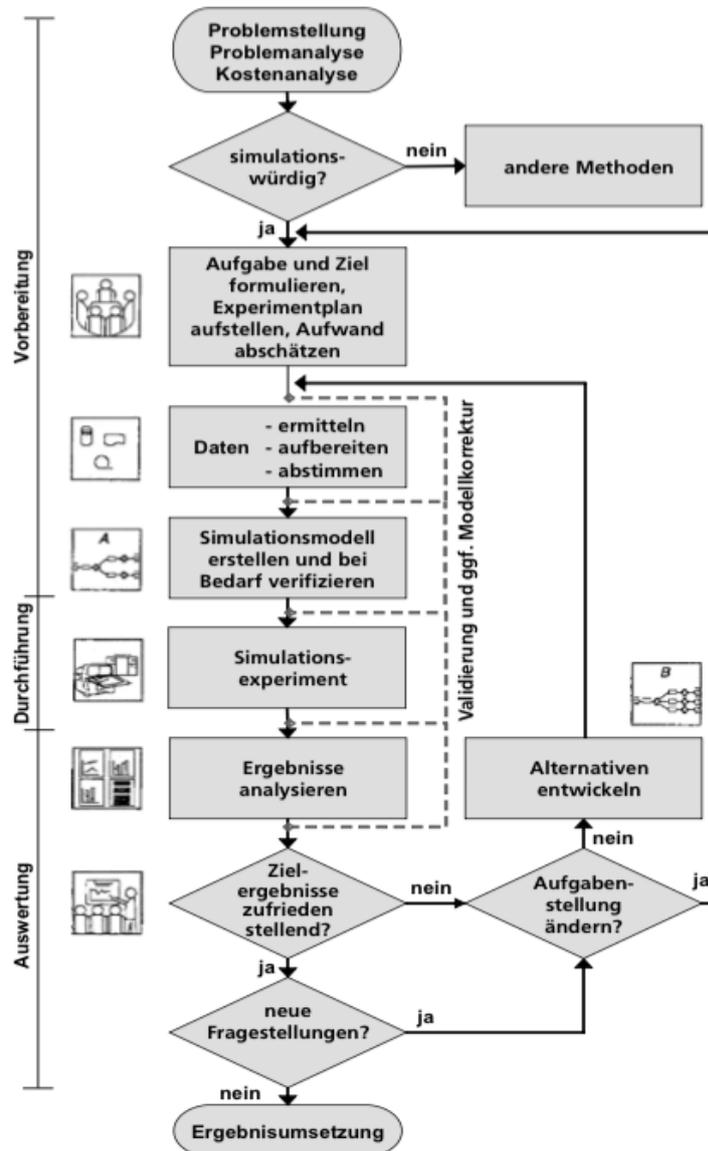


Abbildung 17: Vorgehensmodell bei der Erstellung von Simulationsmodellen nach VDI-Richtlinie 3363, Blatt 1 (Rabe, Spieckermann, Wenzel 2008, S. 32)

In der Literatur gibt es zahlreiche unterschiedliche Vorgehensmodelle für die Erstellung von Simulationsmodellen. Abbildung 17 ist das Vorgehensmodell nach VDI-Richtlinie. Die Aktivitäten, die zu einer Simulationsstudie gehören, kann man in sechs Aufgaben zusammenfassen, Aufgaben und Ziele festlegen, Datenerhebung, Implementierung des Simulationsmodells, Verifikation und Validation des Modells, Durchführung und Auswertung von Experimenten und Übertragung der Ergebnisse auf die Realität (Eley 2012). In Abbildung 17 wird

der gesamte Prozess in drei Phasen Vorbereitung, Durchführung und Auswertung aufgeteilt. Im ersten Schritt werden die Probleme sowie das Ziel analysiert und formuliert. Danach folgt der Aufbau der Datenbasis. Im nächsten Schritt wird ein Simulationsmodell erstellt und verifiziert. Anschließend wird anhand des Simulationsmodells ein Simulationsexperiment durchgeführt und die erhaltenden Ergebnisse bewertet. Bei Bedarf soll die V&V während des Prozesses mehrmals ausgeführt werden.

Wie zuvor schon erwähnt, ermöglichen objektorientierte Petri-Netze die Modellierung der realen Prozesse (wie z.B. Produktionsprozesse, Logistikprozess usw.) und die Beschreibung des dynamischen Verhaltens des Systems. Die Besonderheiten der THORNs liegen in den speziellen Stellenstrukturen, den zusätzlichen Kantentypen sowie der kompakten Struktur. Im Bereich wie Produktion und Logistik ist es sinnvoll, dass nach der Stellenstruktur Queue die benötigten Objekte in dem Lager nach dem Prinzip FIFO entnommen werden, die Materialflüsse abgebildet werden. Es ist auch sehr üblich im Alltag, dass in einem Geschäft die Kunden in einer Schlange nach dem Prinzip FIFO bedient werden. In dieser Situation kann die Warteschlange als eine Queue-Stelle modelliert werden. Es ist auch schon bekannt, dass in THORNs uns die zeitliche Darstellung des realen Systems zur Verfügung steht. In einem Prozess kann die ganze Prozesszeit in einige Teile unterteilt, wie z.B. die Vorbereitungszeit, Wartezeit, Bearbeitungszeit usw. Diese Anforderungen können THORNs mit Hilfe der Verzögerungszeit sowie der Schaltdauer sehr gut befriedigen. Das Hierarchiekonzept hilft bei der Modellierung zuerst ein anschauliches Hauptnetz darzustellen. Es lässt die ganze Modellierung einfacher und fehlerfrei. Wenn nach der Überlegung der Tatsachen nötig ist, ein Prozess zu verfeinern, kann man die detaillierte Darstellung eines Prozesses in ihrem entsprechenden Unternetz kennzeichnen. In diesem Abschnitt wird ein auf THORNs basierendes Workflow-Modell für eine Logistikstation erstellt.

Das Bestellzentrum empfängt die Bestellung von den Kunden, dann entsteht ein Bestellprotokoll und die Details der Bestellung sollten an das Rationierzentrum und Warenlagerzentrum verschickt werden. Das Rationierzentrum wird nach den Details der Bestellung und der Ressourcenbelegung einen vernünftigen Transportplan ausarbeiten und schickt dem Warenlagerzentrum eine Lieferungsbenachrichtigung, übrigens bietet das Rationierzentrum dem Transportzentrum eine Auswahl der optimalen Routen an. Das administrative Personal in dem Lager überprüft die Bestellinformationen und die Lieferungsbenachrichtigungen. Wenn alles in Ordnung ist, wird die Bestellung vorbereitet und die Lieferungsbestäti-

gung erstellt. Anschließend wird die Bestellung von dem Transportzentrum transportiert und den Kunden abgegeben, dann ist der Prozess erfolgreich vollendet (Zhao 2010).

Nach der sorgfältigen Analyse des obenstehenden Prozesses sind uns vier Objekte auffällig, sie sind das Bestellzentrum, das Rationierzentrum, das Warenlagerzentrum und das Transportzentrum. Ihre dazugehörigen Aktivitäten werden im Folgenden aufgelistet:

- Bestellzentrum: Bestellung annehmen, Details der Bestellung weiterleiten
- Rationierzentrum: Transportplan erstellen, Lieferungsbenachrichtigung geben und optimale Routenauswahl anbieten
- Warenlagerzentrum: Bestellung vorbereiten, Lieferungsbestätigung erstellen
- Transportzentrum: Paket von dem Lager abholen, aufladen und den Kunden abgeben

Um den hauptsächlichsten Nachrichtenaustausch zwischen dem einzelnen Objekt anschaulich zu machen, wird der Austausch mit der untenstehenden Grafik erklärt.

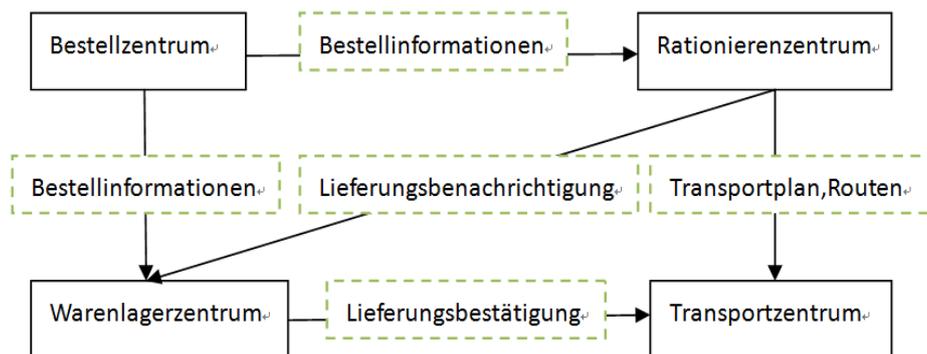


Abbildung 18: Nachrichtenaustausch

Jetzt kann man nach der Realität ein objektorientiertes Petri-Netz Modell (Hauptnetz) aufstellen. Vier Zentren werden hier lieber als vier Aufruftransitionen gekennzeichnet (in Abbildung 19 als schwarze Kästchen), weil im nächsten Schritt jede Aufruftransition weiter verfeinert werden muss, um den ganzen Prozess so genau wie möglich zu behandeln. Die Pfeile auf den Kanten zeigt die Richtung des Nachrichtaustausches. S_1 wird als Queue-Stelle betrachtet.

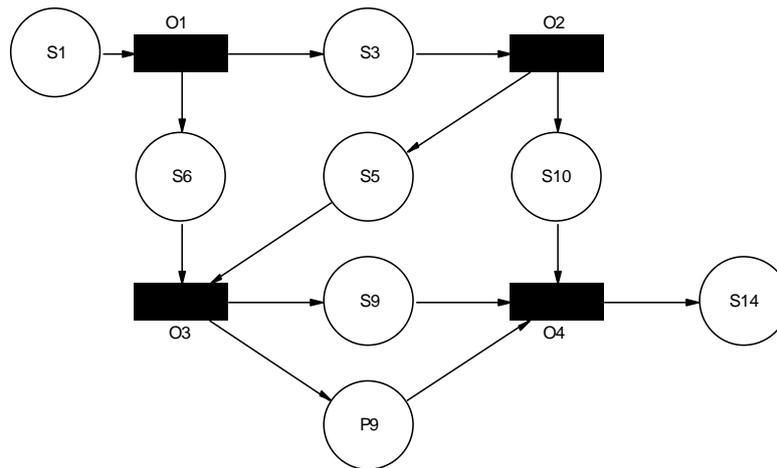


Abbildung 19: Hauptnetz der Logistikstation

Legende:

O₁: Bestellzentrum, O₂: Rationierenzentrum, O₃: Warenlagerzentrum, O₄: Transportzentrum

S₁: wartende Bestellungen, S₃: Details der Bestellung geschickt,

S₅: Lieferungsbenachrichtigung verschickt, S₆: Details der Bestellung geschickt,

S₉: Lieferungsbestätigung geschickt, P₉: Bestellung in dem Warenausgang gelagert

S₁₀: Transportplan verschickt, S₁₄: Bestellung erfolgreich zugestellt

Jetzt werden Ihnen vier Unternetze detailliert erklärt. In Abbildung 20 wird der Arbeitsablauf des Bestellzentrums mittels THORNs dargestellt. Die zum Hauptnetz korrespondierende Stelle S₁, S₃ und S₆ werden hier Gelb gekennzeichnet. S₁ ist im Unternetz die Input-Stelle, während S₃ und S₆ die Output-Stellen sind. Entsprechend sind im Hauptnetz S₁, S₃ und S₆ die Randstellen von der Aufruftransition Bestellzentrum, S₁ ist die Eingabestelle, S₃ und S₆ sind Ausgabestellen. Die zuvor definierte Initialmarkierung lautet (0,0,1,0,0).

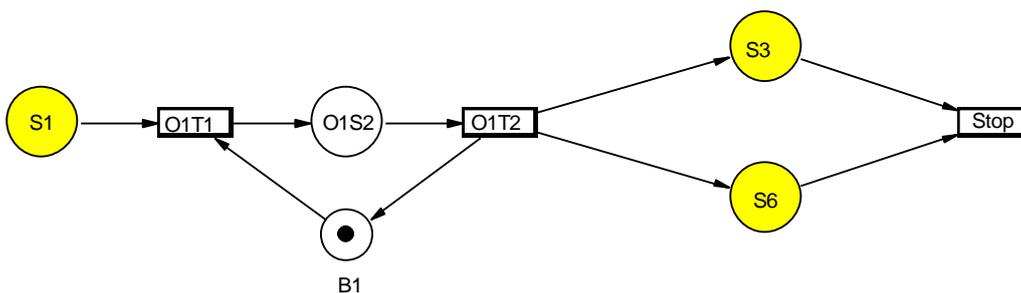


Abbildung 20: Unternetz_Bestellzentrum

Legende:

O₁T₁: Bestellung bearbeiten, O₁T₂:Details der Bestellung schicken

S₁: wartende Bestellungen, O₁S₂: bearbeitete Bestellung, B₁: das Bestellzentrum ist frei,

S₃, S₆:Details der Bestellung geschickt

In Abbildung 21 wird der Arbeitsablauf des Rationierenzentrums mittels THORNs präsentiert. S_3 ist die Input-Stelle, S_5 und S_{10} sind die Output-Stellen in diesem Unternetz. Die Initialmarkierung lautet $(0,0,1,0,0)$.

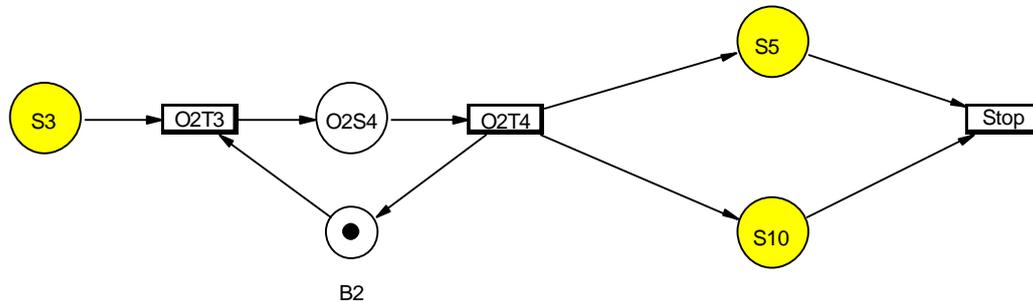


Abbildung 21: Unternetz_Rationierenzentrum

Legende:

O_{2T3} :Transportplan und Lieferungsbenachrichtigung erstellen,

O_{2T4} :Transportplan und Lieferungsbenachrichtigung verschicken

S_3 : Details der Bestellung geschickt,

O_{2S4} : Transportplan und Lieferungsbenachrichtigung erstellt,

B_2 : Rationierenzentrum ist frei, S_5 : Lieferungsbenachrichtigung verschickt,

S_{10} :Transportplan verschickt

In den zuvor dargestellten zwei Unternetzen existiert lediglich der Austausch der Informationen, während in den verbleibenden Unternetzen Warenlagerzentrum sowie Transportzentrum sowohl der Informationsaustausch, als auch der reale Produktaustausch vorhanden sind. Der Informationsfluss und der Produktfluss sind nebenläufig. In Abbildung 22 wird die Kapazität der Stelle P 200 eingestellt, d.h. das Lager kann maximal 200 Stück Produkte annehmen. Das Kantengewicht von der Transition B zu der Stelle P bedeutet die Bestelllosgrößen, die abhängig von den Lieferanten sind. Die Initialmarkierung lautet $(0,0,200,0,0,1,0,0,0,0)$.

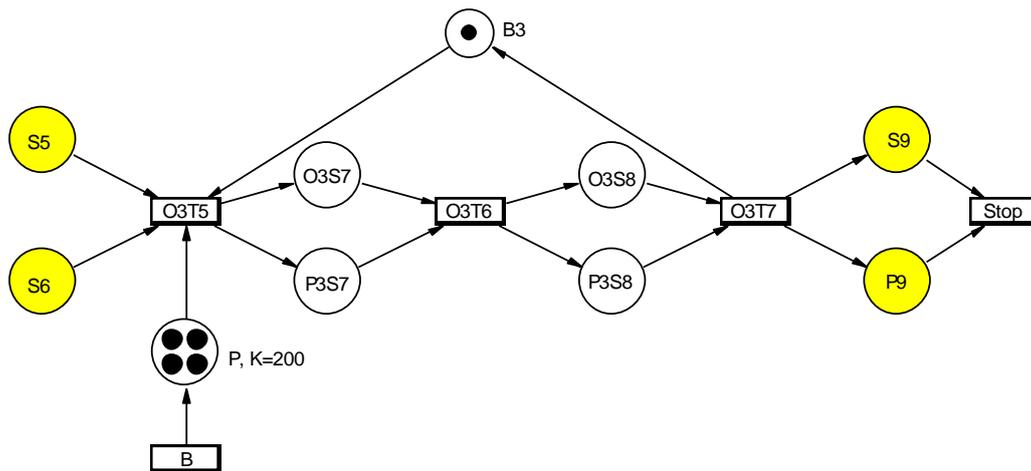


Abbildung 22: Unternetz_Warenlagerzentrum

Legende:

O_3T_5 : Bestellinformationen mit der Lieferungsbenachrichtigung übereinstimmen, die entnommenen Produkte mit der Lieferungsbenachrichtigung übereinstimmen

B: Bestellen von den Lieferanten, O_3T_6 : Bestellung vorbereiten,

O_3T_7 : Lieferungsbestätigung erstellen und schicken, die vorbereitete Bestellung zu dem Warenausgang liefern

S_5 : Lieferungsbenachrichtigung verschickt, S_6 : Details der Bestellung geschickt, P: Lager,

O_3S_7 : Informationen übereinstimmt, P_3S_7 : die benötigten Produkte von dem Lager entnehmen, B_3 : Warenlagerzentrum ist frei, O_3S_8 : der Status der Bestellung „eingepackt“,

P_3S_8 : Bestellung eingepackt, S_9 : Lieferungsbestätigung aus dem Lagerzentrum geschickt,

P_9 : Bestellung in dem Warenausgang gelagert

In Abbildung 23 wird der Arbeitsablauf des Transportzentrums mittels THORNs vorgezeigt. S_9 , P_9 und S_{10} sind die Input-Stellen, S_{14} ist die Output-Stelle in diesem Unternetz. Das Transportzentrum empfängt von dem Rationierzentrum den Transportplan und vom dem Warenlagerzentrum die Lieferungsbestätigung sowie vorbereitete Produkte. Der Transportplan enthält optimale Routen und Transportressourcenzuweisung. Zu den Transportressourcen gehören die Transportmitarbeiter und die Lastkraftwagen. Die Transportressourcenzuweisung gibt an, welche und wie viele Transportmitarbeiter mit welchen Lastkraftwagen diese Bestellung transportieren. Es gibt auch die Möglichkeit, dass eine große Bestellung mehrere Lastkraftwagen und jeder Lastkraftwagen mehrere Mitarbeiter benötigt. Die Kapazität k von der Stelle LKW_2 definiert die maximale Anzahl der Lastkraftwagen zu einem gleichen Zeitpunkt in dem Parkplatz. Die Initialmarkierung lautet $(0,0,0,0,0,k,1,0,0,0,0,0)$.

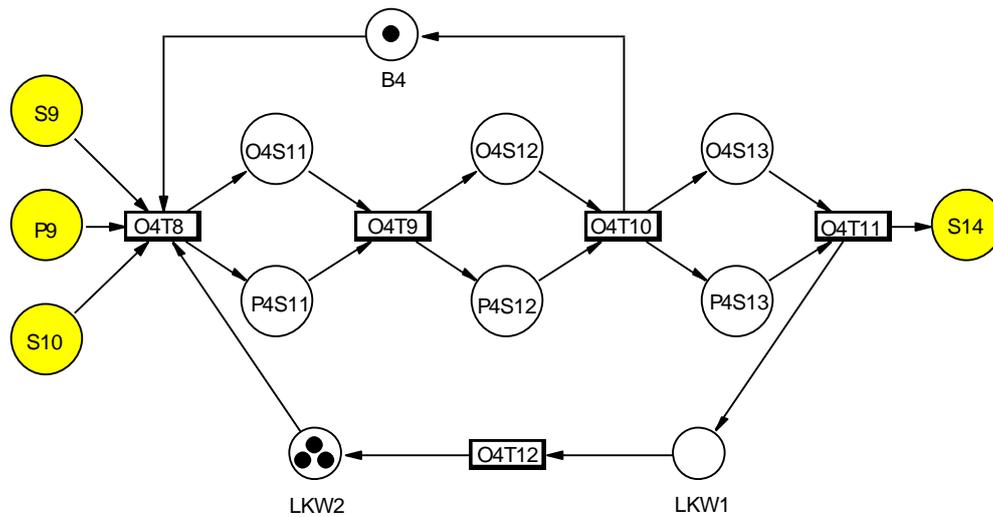


Abbildung 23: Unternetz_Transportzentrum

Legende:

O_4T_8 : Transportressourcen zuweisen, O_4T_9 : Transportpersonal zu dem Lager fahren,

O_4T_{10} : Aufladen der Bestellung, Warenausgangsbestätigung erstellen,

O_4T_{11} : Transportieren der Bestellung, O_4T_{12} : Rückfahren der LKW

S_9 : Lieferungsbestätigung aus dem Lagerzentrum geschickt,

P_9 : Bestellung in dem Warenausgang gelagert, S_{10} : Transportplan verschickt,

O_4S_{11} : Transportbestätigung erstellt, P_4S_{11} : LKW und Personal zugewiesen,

LKW_2 : Ende der Rückfahrt, B_4 : Transportzentrum ist frei,

O_4S_{12} : in dem Warenausgang angekommen P_4S_{12} : wartende Transportressourcen,

O_4S_{13} : Warenausgangsbestätigung erstellt P_4S_{13} : Bestellung aufgeladen,

LKW_1 : Beginn der Rückfahrt, S_{14} : Bestellung erfolgreich zugestellt

6.2 Evaluieren der ausgewählten V&V-Techniken

In diesem Abschnitt werden anhand des in 6.1 erstellten Simulationsmodells die bestehenden V&V-Techniken evaluiert.

6.2.1 Begutachtung

Am Anfang des Evaluierens berücksichtigt man zuerst, ob das Simulationsmodell in Übereinstimmung mit den zuvor vereinbarten Zielen verläuft oder ob das Simulationsmodell die

Realität richtig und vollständig widerspiegelt. An dieser Stelle wird die Technik Begutachtung empfohlen. Eine gute Vorbereitung spielt bei dieser Technik eine große Rolle, d.h. die relevanten Kriterien für den Erfolg dieser Begutachtung festlegen. Für dieses Modell werden insgesamt vier nachfolgende Kriterien ausgewählt.

- die Eignung der Ziel- und Systembeschreibungen

Unsere Ziele liegen in folgenden Punkten:

1. Kundenzufriedenheit steigern

Kundenzufriedenheit ist die kognitive und emotionale Bewertung der gesamten Erfahrungen mit einem bestimmten Anbieter und dessen Produkten bzw. Dienstleistungen (Homburg et al. 1999). Für eine Logistikstation wird Kundenzufriedenheit unter den Aspekten wie z.B. Lieferungstermintreue, Bestellungssicherung (ob die Bestellung in richtiger Menge, ob schadenlos transportiert werden) und noch der günstigere Service betrachtet. Durch das Modell wird der gesamte Prozess transparent dargestellt. Es ist hilfreich, Kundenzufriedenheit zu verbessern.

2. die Beziehungen zwischen den Lieferanten verstärken

Dieses Ziel kann leider durch das auf THORNs basierende Modell nicht erreicht werden.

3. Zusammenarbeit zwischen den internen Zentren verbessern

Durch die Kanten verbindet sich jedes einzelne Zentrum miteinander. Aber in dem Modell existiert keine Interaktion.

Über die Systembeschreibung wird hier die Begutachtung durch die einzelnen Teammitglieder (z.B. die Teammitglieder bestehen aus den jeweiligen Zentrumleiter) aus verschiedenen Zentren mit Hilfe vom Schreibtischtest durchgeführt. Jedes Teammitglied kennt sich seinen eigenen verantwortlichen Bereich gut aus. Wenn etwas ihnen auffällig ist, können sie existierende Fehler oder Redundanzen schnell aufdecken. Durch Aufbau dieses Simulationsmodells wird der reale Arbeitsablauf anschaulich beschrieben und das Systemverhalten transparent dargestellt.

- Angemessenheit und Effizienz der verwendeten Modellierungstechniken

Um das Modell nach diesem Kriterium zu bewerten, kann die Begutachtung mit der Technik Validierung mit Dialog kombiniert angewendet werden. Die Fachexperten kennen das reale System und die verwendete Modellierungstechnik THORNs gut. Das Modell in 6.1 beschreibt den Arbeitsablauf. Jede Transition kann mit einer Verzögerungszeit und einer Schaltdauer versehen, es ist sehr vernünftig für eine Logistikstation. Mittels

des Tortendiagramms können wir die Prozentzahl von der Verzögerungszeit und der Schaltdauer anschaulich sehen. Wenn bei einer Transition eine sehr hohe Prozentzahl an der Verzögerungszeit entsteht, soll man beachten und analysieren, welche Verbesserungspotentiale existieren. Das Hierarchiekonzept der THORNs lässt uns schneller die Fehler auffinden.

– Grad der Übereinstimmung des Modells mit der Realität

In dem Simulationsmodell wird die Stelle S_1 als Queue-Stelle gekennzeichnet. Aber in der Realität wird in einigen Notfällen S_1 als Priority Queue-Stelle betrachtet. Darüber hinaus gibt es in jedem Zentrum eine Stelle wie B_1, B_2, B_3, B_4 , die den Zustand des Zentrums bezeichnet, d.h. wenn eine Bestellung in das Zentrum reinkommt, muss zuerst diese Bestellung fertig behandelt werden. In diesem Zeitraum darf die nächste Bestellung nicht hineinkommen. Zum Beispiel: in dem Warenlagerzentrum wird die eingehende Bestellung durch drei Transitionen O_3T_5, O_3T_6 und O_3T_7 nacheinander bearbeitet. Hier stimmt das Modell mit der Realität nicht überein. Die Mitarbeiter, die für die Transition O_3T_5 verantwortlich sind, wollen nicht warten, wenn die nächste Bestellung schon in den vorgelagerten Stellen steht.

– Vollständigkeit des Modells

Im Vergleich mit dem realen System ist das Simulationsmodell vollständig und machbar.

6.2.2 Ereignisvaliditätstest

Jetzt wird das Verhalten der eingehenden Bestellungen analysiert. Die Bestellungen von den Kunden werden in zwei Arten eingeteilt, eine ist die Bestellung von den bestehenden Kunden, eine andere ist die Bestellung von den neuen Kunden. Der Unterschied zwischen den beiden liegt in der Verhaltensweise. Die Bestellungen von den bestehenden Kunden kommen normalerweise regelmäßig an, beispielsweise kommen die Bestellungen von der Firma A immer am Mittwochvormittag an. Aber es gibt keine Verhaltensregel bei den Bestellungen von den neuen Kunden. Um das stochastische Verhalten zu modellieren, kann der Zufallszahlengenerator oder eine statistische Verteilung aus der bereitgestellten Bibliothek eingesetzt werden. Aber manchmal stimmt das modellierte Auftreten der Bestellungen mit der Realität nicht überein. Deswegen wird die V&V-Technik Ereignisvaliditätstest angewendet. Sowohl zusammenhängende Ereignisse als auch einzelne Ereignisse in ihrer zeitlichen Abfolge kön-

nen betrachtet werden (Hermann 1967; Balci 1990).

In diesem Modell gehört die Stelle S_1 zu dem Fall „einzelne Ereignisse in zeitlicher Abfolge“. Einfach kann die gesamte Anzahl an den eingehenden Bestellungen von den neuen Kunden pro Woche mit der Anzahl in der Realität verglichen werden. Wenn es einen erheblichen Unterschied zwischen den Beiden gibt, ist das Modell nicht gültig. Wie oft soll die Vergleichen stattfinden, pro Tag oder pro Woche, je nachdem. Dieser Test kann ebenso bei dem Lastkraftwagenausfall auf die Transitionen O_4T_9 , O_4T_{11} sowie O_4T_{12} und noch dem Werker ausfall auf fast jeder Transition ausgeführt werden. Während des Transports gibt es vielfältige stochastische Faktoren, die die Transportzeit beeinflussen können, wie z.B. Verkehrsstau, Unwetter, Reifenpanne, Bauarbeit usw., Um diese drei Transitionen O_4T_9 , O_4T_{11} und O_4T_{12} noch detailliert zu untersuchen, können noch die Eintrittshäufigkeiten der jeweiligen möglichen Einflussfaktoren in einem bestimmten Zeitraum mit der Realität verglichen werden. Wenn große Unterschiede zwischen den Beiden bestehen, ist das Modell nicht gültig.

Aber um manchmal ihre Gültigkeit zu bestimmen, soll nicht nur die Anzahl des Auftretens berücksichtigt werden, sondern auch die zeitliche Abfolge in einem zuvor bestimmten Muster. Man kann mithilfe der Animation oder Trace-Analyse zweier oder mehrerer verschiedener Bestellungen untersuchen, ob sie in der gleichen Reihenfolge wie in der Realität auf der Stelle S_1 auftauchen. Es ist auch sehr wichtig, da die verschiedenen Kunden ihre eigene gewünschte Zustellzeit haben. Wenn das Modell die richtige Reihenfolge modelliert, wird es möglich, dass die ankommenden Bestellungen rechtzeitig behandelt werden können.

6.2.3 Festwerttest & Vergleich mit anderen Modellen

Um eine maximale Anzahl der transportierten Produkte pro Tag in der Logistikstation festzulegen, wird die Technik Festwerttest empfohlen. Diese Technik kann lediglich in dem Modellteil, indem nur konstante Werte angewendet werden, eingesetzt werden. Diese Technik ist für die Transition O_3T_6 „Bestellung vorbereiten“ im Warenlagerzentrum besonders geeignet. In dieser Transition wird die Aktivität „die Bestellung einpacken“ ausgeführt. Das ist ein teilautomatischer Prozess. Die von den Lieferanten bestellten Produkte werden zuerst durch verschiedene Maschinen nacheinander in dem Fließband eingepackt und befestigt, die gesamte Bearbeitungszeit in dem Fließband für jedes Produkt ist ein deterministischer Wert.

Danach werden jede 20 Stück Produkte auf einer Palette gelegt und die Palette befestigt. Dieser Prozess wird manuell durchgeführt, aber kann die Bearbeitungszeit als Konstante betrachtet werden. Die Mitarbeiter besitzen unterschiedliche Fähigkeiten, wie z.B. die Verpackungsgeschwindigkeiten, deshalb werden die unterschiedlichen manuellen Bearbeitungszeiten durch die deterministischen Mittelwerte der empirischen Daten ersetzt. Aus der von den Maschinen bearbeiteten Zeit und der von den Mitarbeitern behandelten Zeit ergibt sich der gesamte Zeitaufwand für die Transition O_3T_6 . Dann kann die maximale Anzahl an der Palette pro Tag ohne Simulation ausgerechnet werden. Normalerweise bei dem Festwerttest werden zunächst die Hypothesen formuliert, beispielsweise bedeutet es, dass pro Tag 500 Paletten zu dem Warenausgang transportiert werden. Anschließend wird das Simulationsmodell ausgeführt. Wenn die Anzahl der Paletten größer als die ausgerechnete Anzahl ist, ist das Simulationsmodell nicht gültig, ansonsten steigt die Glaubwürdigkeit durch die Technik.

In der Praxis wird diese Technik häufig kombiniert im Vergleich mit anderen Modellen verwendet. Das Vergleichsmodell wird am bestens von den damaligen Lieferanten der Verpackungslinien, die sich in der Transition O_3T_6 befinden, angeboten und ist normalerweise einfacher als das Simulationsmodell. Der Grund dafür ist, dass die Lieferanten ihre selbst hergestellten Verpackungslinien besser kennen, d.h. das von den Lieferanten angebotene Vergleichsmodell ist zuverlässig. Unter den gleichen Eingabewerten werden die Ergebnisse des Simulationsmodells mit den durch die Tabellenkalkulation berechneten Ergebnissen von dem Vergleichsmodell verglichen. Wenn große Abweichungen auftreten, ist unbedingt festzustellen, wo die Fehlerstellen sind, welche Gründe dafür sind und wie die Fehler beseitigt werden können.

6.2.4 Monitoring

Für dieses logistische Simulationsmodell ist Monitoring als V&V- Technik von großer Bedeutung. In dem Buch „Verifikation und Validierung für die Simulation in Produktion und Logistik“ von Rabe, Spieckermann und Wenzel wird die Darstellung von Monitoring in zwei Verfahren dargestellt:

- Werte zum Zeitpunkt

Bei diesem Simulationsmodell wird es empfohlen, dass auf der Stelle S_1 wartende Bestellungen die aktuelle Wartezeit von Bestellungen vor der Bearbeitung durch ein Kreis-

diagramm sichtbar machen. Das Kreisdiagramm wird hier als eine ablaufende Uhr betrachtet. Die ähnliche Situation ist die Transition O_4T_{12} Rückfahren des LKWs, mittels eines Kreisdiagramms die Restrücklaufzeit des LKWs veranschaulicht wird. Bei Monitoring werden die Darstellungen nach verschiedenen Anforderungen ausgewählt. Um die Effektivität des Transportplans zu kontrollieren und zu verbessern, werden mit Hilfe des Balkendiagramms die Transportauslastungen (in Prozent) von jedem einzelnen LKW gezeichnet. Darüber hinaus sind die Daten von Kunden für die Managementebene sehr wichtig, deshalb können die Mitarbeiter die Kunden in verschiedene Kundensegmente einteilen (z.B. Großkunden, Normale Kunden usw.). Die unterschiedlichen Kundensegmente werden als farbige Balken dargestellt, die aufeinander platziert werden können, dessen Werte sich zu 100% addieren.

– Werte im Zeitverlauf

Lagermanagement spielt immer eine sehr wichtige Rolle in einer Logistikstation. angemessene Bestandslevels zu bestimmen, ist eine der wichtigsten und schwierigsten Aufgaben in dem Lagermanagement. Wenn zu wenig Bestand im Lager existiert, gerät die Logistikstation wahrscheinlich in Gefahr, dass der Bestand die Bestellung von Kunden nicht rechtzeitig abdecken kann. Aber zu viel Bestand kann eine hohe Kapitalbindung verursachen. Deswegen ist es notwendig, einen Sicherheitsbestand festzulegen, um die gewünschte Produktverfügbarkeit zu erhalten (Eazystock 2015). Durch den Einsatz des Monitoring wird eine ähnliche Kurve wie in Abbildung 24 über das Bestandsniveau im Lager für die Stelle P angezeigt. Die blaue Linie zeigt die normale Zustandsänderung, während die rote Linie die Situation Umsatz > Prognose und die lila Linie die Situation späte Lieferung gezeichnet. In den Situationen Umsatz > Prognose und späte Lieferung wird der Sicherheitsbestand angewendet, um den Kunden die Bestellungen rechtzeitig zuzustellen. Wenn in dem Simulationsmodell die beiden Fälle auftreten, soll man nachdenken, welche Fehler existieren, wie die unerwarteten Linien vermieden werden können. Beispielsweise kommen bei der roten Linie mehrere Bestellungen als erwartet, d.h. die Mitarbeiter sollen unbedingt immer die veränderte Marktumwelt beachten und rechtzeitig darauf reagieren. Bei der späten Lieferung verursacht direkt die längere Beschaffungszeit, an dieser Stelle sollen die Mitarbeiter dieser Logistikstation überlegen, welche Gründe dafür sind, ein Problem in dieser Branche z.B. wegen der knappen Rohmaterialien aus ihrem Herkunftsland oder ein spezielles Problem, das nur bei einem Lieferant auftaucht, z.B. die internen Ausfälle der Produktionsanlagen. Deswegen sollte die

Kommunikation sich zwischen den Lieferanten noch verstärken, um die negativen Einflüsse so weit wie möglich zu senken.

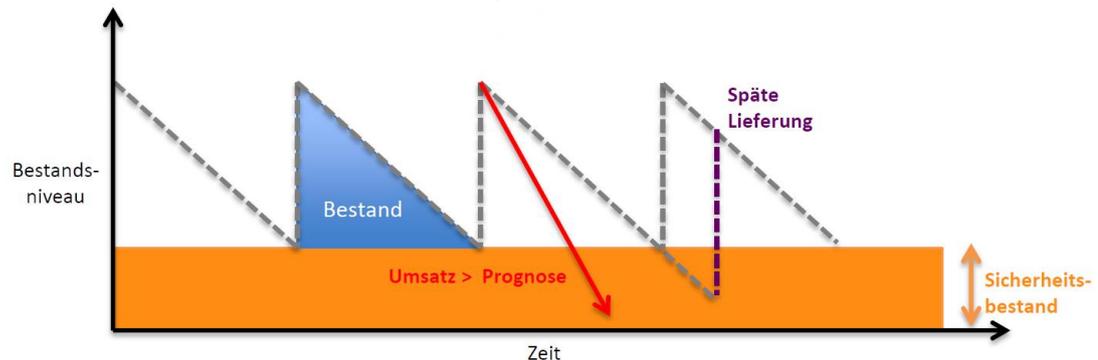


Abbildung 24: Bestandsniveau über die Zeit (Eazystock 2015, Seite 2)

6.2.5 Sensitivitätsanalyse

Die V&V-Technik Sensitivitätsanalyse wird vorgeschlagen, dass bei der Transition B Bestellen von den Lieferanten Sensitivitätsanalyse eingesetzt werden soll. Der Grund dafür ist, dass es notwendig ist, die Abhängigkeit dieser Logistikstation von ihren Lieferanten zu untersuchen. Für die sensitiven Parameter werden hier die Bestellzeiten festgelegt, weil die Schwankungen der Bestellzeiten die nachfolgenden Prozesse sogar die Kundenzufriedenheit beeinflussen können. In diesem Modell werden von verschiedenen Lieferanten unterschiedliche Bestellzeiten angekündigt. Bei einem Lieferanten kann es auch auftreten, dass in verschiedenen Situationen unterschiedliche Bestellzeiten vorgegeben werden. Die Richtung der Auswirkung auf das Simulationsmodell wird beobachtet, wie z.B. ob die verlängerte Bestellzeit die Kundenzufriedenheit schwer beeinträchtigt oder die noch in dem Toleranzbereich liegt, danach wird ein Vergleich mit der Realität durchgeführt. Wenn die Beiden miteinander übereinstimmen, ist das Simulationsmodell glaubwürdig.

Die Sensitivitätsanalyse kann noch für die Transitionen O_4T_9 , O_4T_{11} und O_4T_{12} angewendet werden. Es ist bekannt, dass die Schaltdauer von solchen Transitionen abhängig von der Transportdistanz ist. Aber während des Transports gibt es vielfältige stochastische Faktoren, die die Transportzeit beeinflussen können, wie z.B. Verkehrsstau, Unwetter, Reifenpanne, Bauarbeit usw. Man kann alle aufgelisteten Einflussfaktoren zusammengefasst und gemäß den Erfahrungen jedem Einflussfaktor eine Auftretenswahrscheinlichkeit und ein Schweregrad der Auswirkung geben. Das heißt, dass für jeden Einflussfaktor eine sogenannte Sensiti-

vitätszahl aus der Auftretenswahrscheinlichkeit und dem Schweregrad gegeben wird. Eine große Sensitivitätszahl bedeutet, dass dieser Einflussfaktor die Transportzeit stark beeinflusst. Hier soll es berücksichtigt werden, wenn beide oder mehrere Einflussfaktoren zugleich auftreten, wird der Schweregrad der Auswirkung verstärkt? Die Zusammenwirkungen zwischen jedem Einflussfaktor können durch eine Matrix aufgeklärt werden. Zum Beispiel: Unwetter und Bauarbeit könnten den Verkehrsstau auslösen und auf diesen Fall kann die Wahrscheinlichkeit von Verkehrsstau auch gesteigert werden. Abschließend werden die ausgerechneten Sensitivitätszahlen der Einflussfaktoren mit ihren tatsächlichen Einflüssen auf die Transportzeiten verglichen. Wenn sie übereinstimmen, ist das Simulationsmodell glaubwürdig.

6.2.6 Strukturiertes Durchgehen, Schreibtischtest, Validierung im Dialog

In diesem Abschnitt werden die Techniken strukturiertes Durchgehen, Schreibtischtest und Validierung im Dialog anhand des Simulationsmodells zusammen erläutert, weil diese drei Techniken für alle Phasenergebnisse des Modellierungsprozesses verwendet werden können. Nach der detaillierten Erklärung werden die Beziehungen zwischen einander gut kennengelernt.

Es wird empfohlen, dass der Schreibtischtest von jedem Teilnehmer, der sich an der Modellbildung beteiligt, benutzt werden soll, um das Modell in Bezug auf Vollständigkeit, Korrektheit, Konsistenz und Eindeutigkeit zu überprüfen. Nachfolgend werden einige praktische Beispiele angezeigt. An der Transition O_3T_5 von dem Warenlagerzentrum werden zuerst zwei Aktivitäten durchgeführt. Zum Glück haben die Simulationsfachleute durch den Schreibtischtest gefunden, dass die modellierte Transition nicht korrekt und unvollständig ist, die Aktivität „die Produkte aus dem Lager entnommen“ fehlt. Die Transition verläuft in der Realität in drei Schritten. Zuerst wird die Bestellinformation mit der Lieferungsbenachrichtigung von den zuständigen Mitarbeitern verglichen. Wenn die beiden Informationen miteinander übereinstimmen, wird die zweite Aktivität „die Produkte aus dem Lager entnommen“ ausgelöst. Danach wird die dritte Aktivität „die entnommenen Produkte mit der Lieferungsbenachrichtigung übereinstimmen“ von den Mitarbeitern ausgeführt. Deswegen soll die Aktivität „die Produkte aus dem Lager entnommen“ in der Beschreibung von der Transition O_3T_5 addiert

werden. Natürlich kann die Transition O_3T_5 auch als Aufruftransition modelliert werden. In dem Unternetz von O_3T_5 gibt es drei Transitionen, sie sind die obengenannten drei Aktivitäten.

Aber manchmal ist es schwer, die Fehler von selbst aufzufinden. Deswegen wird der Schreibtischtest sehr häufig mit der Technik Validierung im Dialog kombiniert verwendet. Zum Beispiel: wegen des Mangels an den Kenntnissen über das reale System hatten die Simulationssachleute die Bestelllosgrößen von der Transition B falsch festgelegt. Wenn der Fachexperte die ungewöhnlichen Werte sieht, kann er schnell reagieren und wird ein mit der Realität übereinstimmender Wert dann in der Transition B erneut eingegeben.

Mit der Technik Strukturiertes Durchgehen können die Phasenergebnisse von den Experten im Einzelnen durchgegangen werden. Dazu müssen die ausgewählten Experten am Aufbau des Modells sowie Festlegung der Dokumente nicht beteiligt sein. Bei dem Phasenergebnis Konzeptmodell, formales Modell und ausführbares Modell gliedern sich ihre zugehörige Dokumente in fünf Kapitel, die sind Aufgabenspezifikation und Systembeschreibung, Modellierung der Systemstruktur, Modellierung der Teilsysteme, systematische Zusammenstellung der erforderlichen Modelldaten und Wiederverwendbare Komponenten (Rabe, Spieckermann, Wenzel 2008). Im Verlauf von Strukturiertem Durchgehen werden die Simulationsleute zusammen mit den Fachexperten über jedes Kapitel der Dokumente diskutieren und verbessern. Zum Beispiel: bei dem Phasenergebnis ausführbares Modell wird festgelegt und dokumentiert, welche Software für das auf THORNs basierende Simulationsmodell endlich ausgewählt wird, sogar die verwendeten Bibliotheken, die verwendete Version und die verwendete Lizenz. Es ist auch wichtig, alle logischen Beziehungen zwischen jedem einzelnen Element sowie ihre Reihenfolge nochmals zu prüfen. Für THORNs ist es unbedingt zu beachten, ob die Verzögerungszeit und die Schaltdauer von jeder Transition richtig in C++-Code transformiert werden, ob die Attributwerte von den Marken auf der Stelle richtig in C++-Code formuliert werden, ob die Methoden von den Marken, die Schaltbedingungen sowie die Schaltaktionen von den Transitionen in C++-Code genau präsentiert werden wie in der Realität.

6.2.7 Animation



Abbildung 25: Animation von einer Fabrik (Tarakos, 2016)

In Abbildung 25 wird ein alltägliches Arbeitsszenario in einer Fabrik durch eine Animationssoftware dargestellt. Hier ist nur ein Snapshot von Animation. Das kann ein Teilbereich von dem Warenlagerzentrum sein. Die von dem Lager entnommenen Produkte werden in dem Fließband transportiert. Zuerst werden die Produkte von den Maschinen grob eingepackt, anschließend werden die Produkte manuell bearbeitet. Durch die Animation wird ein dynamischer manueller Bearbeitungsprozess im Warenlagerzentrum visuell angezeigt. Die grob eingepackten Produkte werden nach dem Arbeitsplan eine Station über eine Station fein bearbeitet, wie z.B. Füllmaterialien in Karton verpacken, Gebrauchsanweisung in Karton verpacken, die Etiketten kleben usw., schließlich zum Warenausgang des Fließbandes transportiert. In jeder Station werden ein oder mehrere Mitarbeiter belegt, je nachdem. Durch die Animation haben die Simulationsfachleute gefunden, dass die simulierte Zeit deutlich weniger als die Zeit in der Realität ist. Dann vergleichen die Simulationsfachleute das animierte Szenario mit dem realen Szenario Schritt für Schritt. Die Bearbeitungsgeschwindigkeit der animierten Arbeiter ist deutlich höher als in der Realität, fast alle animierten Arbeiter werden als qualifizierte Arbeiter betrachtet, aber tatsächlich sind ein Drittel der gesamten Arbeiter in diesem Bereich die Neulinge. Darüber hinaus gibt es den Arbeiterausfall wegen der Verletzung und der Erkrankung. Solche Einflussfaktoren sollen auch in dem Simulationsmodell berücksichtigt werden. Es gibt auch die Möglichkeit, dass durch die Animation die in dem Simulationsmodell fehlenden Arbeitsschritte schnell aufgefunden und lokalisiert werden können.

6.2.8 Trace-Analyse

In diesem Simulationsmodell wird die Technik Trace-Analyse mit dem zuvor erläuterten Festwerttest kombiniert durchgeführt. Hier wird ein Produkt von der Transition O_3T_6 verfolgt, von dem in dem Fließband eingelegt wird, bis es das Fließband verlässt. In der Transition O_3T_6 werden alle ungepackten Produkte von verschiedenen Maschinen am Band und den am Fließband arbeitenden Mitarbeitern bearbeitet und verpackt. Die alle über dieses Produkt betreffenden Daten werden in einer „Trace-Datei“ gespeichert. Die Orte, die benötigten Zeiten für jede Verpackungsphase dieses Produktes können verfolgt, analysiert und jede Zeit aus der Datei nachvollgezogen werden.

Um dieses Technik objektiv anzuwenden, wird zuerst eine Hypothese formuliert, dass das Produkt innerhalb von 9 Minuten zum Warenausgang des Fließbandes transportiert werden kann. Alle stochastischen Einflussfaktoren werden erfahrungsgemäß durch deterministische Werte ersetzt, z.B. der Zustand der Mitarbeiter und der Maschinen ist normalerweise instabil, sogar können manchmal Ausfälle entstehen, aber durch den Festwerttest wird der Zustand als konstant angenommen. Während der Trace-Analyse können alle aufgezeichneten Messdaten von diesem Produkt (z.B. die einzelnen Bearbeitungszeiten von verschiedenen Maschinen und von verschiedenen Mitarbeitern) durch eine ausgewählte Trace-Software mittels der erstellten sogenannten Traceschrittvorlagen, in denen alle Spezifikationen als auswertbare formale oder semiformale Analysevorschrift gekapselt werden, analysiert und ausgewertet (Wikia, 2016). Darüber hinaus können die in „Trace-Datei“ gespeicherten Daten auch von Hand nach der zuvor festgelegten Regel nachgerechnet. Wenn das Simulationsergebnis innerhalb von 9 Minuten ist, stimmt die Hypothese.

6.2.9 Statistische Techniken

In diesem Simulationsmodell wird der Chi-Quadrat Goodness-of-Fit-Test als ein Anpassungstest für die verwendete Verteilung von der Stelle S_1 ausgewählt. Einige häufig angewendete statistische Verteilungen sowie ihre kurzen Beschreibungen sind in Tabelle 2 zusammengefasst. Unter Zuhilfenahme von Chi-Quadrat-Test kann man untersuchen, wie gut eine beobachtete Häufigkeitsverteilung einer nominalen Variablen einer erwarteten Häufigkeitsverteilung entspricht (matheguru, 2016). Der Test beginnt hier mit den Hypothesen, auf der

Stelle S_1 wird die Verteilung Gleichverteilung ausgewählt, deswegen lautet H_0 : Die Verteilung der ankommenden Bestellungen sind in vier Zeitabschnitten gleichhäufig verteilt, symbolisch schreibt man $F(\text{Bestellung_Zeitabschnitte}) = F_{\text{GLEICH}}(x)$. Der Test wird auf einem Signifikanzniveau α von 5% festgelegt. Die Definition ist in Abbildung 22 zu sehen.

Tabelle 5: Einige Verteilungsfunktion (Eley 2012, Seite 20)

Verteilung	Parameter	Beschreibung
Gleich	a, b	Alle Werte zwischen einem minimalen Wert a und einem maximalen Wert b sind gleich wahrscheinlich.
Dreieck	a, b, m	Neben dem minimalen Wert a und dem maximalen Wert b ist auch noch der am ehesten wahrscheinliche Wert m bekannt. Die Dichtefunktion setzt sich aus zwei Geradenstücken zusammen, die die Punkte $(a, 0)$ und $(m, f(m))$ bzw. $(m, f(m))$ und $(b, 0)$ verbinden.
Normal	μ, σ	Die Normalverteilung beschreibt Phänomene, die symmetrisch um einen vorgegebenen Mittelwert μ schwanken. Die Schwankungsbreite wird durch die Standardabweichung σ beschrieben. Die Normalverteilung ist auch deshalb von Interesse, da sie die Summe von unabhängigen Zufallsvariablen approximiert.
Poisson	λ	Die Poissonverteilung beschreibt die Anzahl an Ereignissen, die in einem vorgegebenen Intervall eintreten. Dies kann z.B. die Anzahl an Kunden sein, die in einem Zeitraum eintreffen. λ wird dabei als Ankunftsrate bezeichnet und stellt die durchschnittliche Anzahl an Ereignissen dar.
Negativ Exponential	λ	Die negative Exponentialverteilung wird insbesondere dazu verwendet, um die Zeit zwischen zwei unabhängigen Ereignissen, z.B. der Ankunft von zwei aufeinanderfolgenden Kunden, zu modellieren. Der Parameter λ stellt den Kehrwert des Erwartungswerts dar, d.h. $E(X) = 1 / \lambda$. Diese Verteilung ist gedächtnislos. Daher ist die zufällige Lebenszeit eines Systems, das in einem Zeitintervall fester Länge unabhängig von seiner bisherigen Lebensdauer mit einer konstanten Wahrscheinlichkeit ausfällt, negativ exponentiell verteilt.
Weibull	α, β	Die Weibull-Verteilung stellt eine Verallgemeinerung der Exponentialverteilung dar. Sie wird häufig zur Bestimmung der Zeit, die bis zur Beendigung einer Aufgabe benötigt wird, verwendet oder um die Ausfallzeit einer Maschine zu modellieren.
Erlang	k	Die Erlang-Verteilung ist die Summe von k unabhängigen negativ exponentialverteilten Zufallszahlen mit dem gleichen Parameter λ .
Bernoulli	p	Diese Verteilung verfügt nur über zwei Realisierungen: „Erfolg“ mit Wahrscheinlichkeit p oder „kein Erfolg“ mit einer Wahrscheinlichkeit von $(1 - p)$.
Binomial	n, p	Wird ein Bernoulliversuch n -mal hintereinander wiederholt, dann gibt die Realisation einer binomialverteilten Zufallsvariablen die Anzahl an, mit der das Ereignis „Erfolg“ eingetreten ist. Die Wahrscheinlichkeit für „Erfolg“ wird dabei wiederum durch den Parameter p vorgegeben.

Die Prüfgröße χ^2 wird wie folgt berechnet:

$$\chi^2 = \sum \frac{(\text{beobachtete Häufigkeit} - \text{erwartete Häufigkeit})^2}{\text{erwartete Häufigkeit}}$$

Mit den Hypothesen:

- H_0 : Die Zufallsvariable besitzt die angegebene Verteilung
- H_1 : Die Zufallsvariable besitzt nicht die angegebene Verteilung

Freiheitsgrade: Anzahl der möglichen Ausprägungen der Variable – 1.

Abbildung 26: Definition von dem Chi-Quadrat-Test (Matheguru, 2016)

Die Arbeitszeit wird in vier Teile 6:00—12:00, 12:00—18:00, 18:00—00:00, 00:00—6:00 unterteilt. Das bedeutet, dass unser Modell vier Ausprägungen (Arbeitszeit) hat, daher gibt es drei Freiheitsgrade q . Anschließend werden die Daten von der ankommenden Bestellung durch die Beobachtung aus der Realität gesammelt und in den entsprechenden Bereichen als beobachtete Häufigkeit eingetragen. Mit der nachfolgenden Formel wird die Prüfgröße χ^2 ausgerechnet, wobei $\text{erwartete Häufigkeit} = \frac{\text{gesamte Anzahl der ankommende Bestellung}}{4}$ ist.

Im nächsten Schritt wird der kritische Wert gemäß der Tabelle 3 $\chi^2_{(q)}$ -Verteilung durchgesucht. Wenn die zuvor berechneten Prüfgröße kleiner als der kritische Wert ist, werden die aufgestellten Null-Hypothesen nicht verworfen, d.h. die auf der Stelle S_1 verwendeten Verteilungen—Gleichverteilung geben die Realität hinreichend genau wieder.

Tabelle 6: $\chi^2_{(q)}$ -Verteilung (Universität Innsbruck, 2016)

q	$\alpha = 0.99$	$\alpha = 0.95$	$\alpha = 0.9$	$\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.01$	$\alpha = 0.005$
1	0.000	0.004	0.016	2.706	3.841	6.635	7.879
2	0.020	0.103	0.211	4.605	5.991	9.210	10.597
3	0.115	0.352	0.584	6.251	7.815	11.345	12.838
4	0.297	0.711	1.064	7.779	9.488	13.277	14.860
5	0.554	1.145	1.610	9.236	11.070	15.086	16.750
6	0.872	1.635	2.204	10.645	12.592	16.812	18.548
7	1.239	2.167	2.833	12.017	14.067	18.475	20.278
8	1.646	2.733	3.490	13.362	15.507	20.090	21.955
9	2.088	3.325	4.168	14.684	16.919	21.666	23.589
10	2.558	3.940	4.865	15.987	18.307	23.209	25.188
11	3.053	4.575	5.578	17.275	19.675	24.725	26.757
12	3.571	5.226	6.304	18.549	21.026	26.217	28.300
13	4.107	5.892	7.042	19.812	22.362	27.688	29.819
14	4.660	6.571	7.790	21.064	23.685	29.141	31.319
15	5.229	7.261	8.547	22.307	24.996	30.578	32.801
16	5.812	7.962	9.312	23.542	26.296	32.000	34.267
17	6.408	8.672	10.085	24.769	27.587	33.409	35.718
18	7.015	9.390	10.865	25.989	28.869	34.805	37.156
19	7.633	10.117	11.651	27.204	30.144	36.191	38.582
20	8.260	10.851	12.443	28.412	31.410	37.566	39.997

6.2.10 Test von Teilmodellen

Der Test von Teilmodellen ist besonders geeignet für unsere THORNs, weil sie hierarchisch strukturiert sind. In diesem Simulationsmodell werden zunächst die Verifikation und Validierung für jedes Unternetz über strukturell und zeitlich durchgeführt. Danach wird das Hauptnetz verifiziert und validiert.

Für dieses THORNs-Modell liegt der Schwerpunkt der V&V in dem Zeitmanagement. Es gibt zwei Arten von Zeitzwang im Workflow-THORNs-Modell. Die beiden Zeitzwänge beeinflussen und schränken einander ein.

1. struktureller Zeitzwang

wird von der Kontrollstruktur des Workflows hergeleitet, d.h. in der Phase der Durchführung soll der strukturelle Zeitzwang von dem Workflow befriedigt werden.

2. expliziter Zeitzwang

ist für die Organisationsordnung relevant, wird von dem Prozess-Konstrukteur festgestellt, wie z.B. die Beginnzeit, die Endzeit.

Um das Modell zu verifizieren und zu validieren, werden einige Begriffe eingeführt.

- obere Schranke (upper bound constraint): Der Zeitabstand zwischen Transition i und Transition j soll kleiner oder gleich dieser Wert sein. Es wird als $\{\mathbf{upb}(i, j)\}$ bezeichnet.
- untere Schranke (lower bound constraint): Der Zeitabstand zwischen Transition i und Transition j soll größer als dieser Wert sein. Es wird als $\{\mathbf{lob}(i, j)\}$ bezeichnet.
- kleinste Verzögerungszeit zwischen beiden Transitionen: Der kürzeste Zeitabstand zwischen Transition i und Transition j , wird als $\{\mathbf{d}(i, j)\}$ bezeichnet.
- größte Verzögerungszeit zwischen beiden Transitionen: Der größte Zeitabstand zwischen Transition i und Transition j , wird als $\{\mathbf{D}(i, j)\}$ bezeichnet.
- Verzögerungszeit der Transition i : Die Menge wird als $\mathbf{C} = \{\mathbf{C}_{\min}(i), \mathbf{C}_{\max}(i)\}$ bezeichnet, wobei C_{\min} kleinste Verzögerungszeit der Transition i ist, C_{\max} größte Verzögerungszeit der Transition i ist.
- Schaltbeginn der Transition i : unter Schaltbeginn versteht man hier den Zeitpunkt des Bestellungsankommens, die Menge wird als $\mathbf{S} = \{\mathbf{s}(i)\}$ bezeichnet.
- Schaltend der Transition i : Die Menge wird als $\mathbf{E} = \{\mathbf{e}(i)\}$ bezeichnet.
- Frist der Transition i : Die Transition i muss vor der Frist erledigt werden und die Frist wird als $\{\mathbf{F}(i)\}$ bezeichnet.

Die Voraussetzungen $D(i,j) \leq \text{upb}(i,j)$, $d(i,j) \geq \text{lob}(i,j)$ und die Frist $F(i) \geq \max\{D(1, i) + s(1)\}$ sollen erfüllt werden.

Nachfolgend werden vier wichtige Zeitparameter formelhaft angegeben:

- früheste Beginnzeit $\text{FBZ} = C_{\min}(i) + s(i)$
- späteste Beginnzeit $\text{SBZ} = C_{\min}(i) + e(i)$
- früheste Endzeit $\text{FEZ} = C_{\max}(i) + s(i)$
- späteste Endzeit $\text{SEZ} = C_{\max}(i) + e(i)$

In diesem Modell werden die Simulationsfachleute zunächst die Zeitbeziehung des Unternetzes untersuchen und überprüfen. Zum Beispiel, im Unternetz Bestellzentrum werden die Zeiten der Transitionen $O_1T_1\{(0,2), 8,9,11\}$ und $O_1T_2\{(2,5), 11,12,14\}$ unter der Form $\{C, S, E, F\}$ angegeben. Aus den obenstehenden Formeln werden die vier Zeitparameter für die Transition O_1T_1 (8,9,10,11) und O_1T_2 (13,14,16,17) abgeleitet. Die Transition O_1T_1 und O_1T_2 erfüllt die notwendige Bedingung $s(i) < e(i) \leq s(i + 1)$ und $s(i) \geq \text{FEZ}(i - 1)$. Dazu gibt es noch zwei Prüfungsregeln über den Zeitzwang für die Transitionen (Li, Chen, Liu 2007). Regel 1 ist für die einzelne Transition, wenn $\text{FBZ}(i) \geq F(i - 1)$, erfüllt die Transition i den Zeitzwang. Regel 2 ist für zwei nacheinander auftretende Transitionen wie O_1T_1 und

O_1, T_2 , sollen $FBZ \leq SBZ \leq SEZ \leq F$ und $FBZ \leq FEZ \leq SEZ \leq F$ erfüllt werden, gleichzeitig existiert eine Beziehung zwischen beiden Transitionen $SEZ(O_1 T_1) \leq FBZ(O_1 T_2)$. Aber offensichtlich ist in diesem Modell $SEZ \not\leq F$ ($17 \not\leq 14$), d.h. die Simulationsfachleute sollen die entstehenden Fehler nochmals kontrollieren und das Modell mit der Realität vergleichen, um die Fehler zu beseitigen. Nach den kompletten Überprüfungen der Unternetze kann man diese Prüfung weiter im Hauptnetz einsetzen. Es ist zu beachten, dass die Transition O_3 zwei vorgelagerte Transitionen O_1 und O_2 haben und O_1 die vorgelagerte Transition von O_2 ist. Zum Beispiel: $O_1 \{(0,2), 8,14,16\}$, $O_2 \{(2,3), 14,18,23\}$ und $O_3 \{(1,3), 18,32,36\}$ sind gegeben. Daraus ergeben sich vier Zeitparameter für die Transition O_1 (8,14,10,16), O_2 (16,20,17,21) und O_3 (19,33,21,35). Nach der Nachrechnung erfüllen O_1 und O_2 die notwendige Bedingung sowie zwei Regeln für sequentielle Beziehungen zwischen O_1 und O_2 . Für O_3 soll ihr Schaltbeginn 18 größer als oder gleich $\min\{10,17\}$ sein. Darüber hinaus muss die $FBZ(O_3) \geq \max\{F(O_1), F(O_2)\}$ sein, aber im Modell $19 \not\geq 23$. Für die Transition O_2 und O_3 haben die Simulationsfachleute noch gefunden, dass die Regel $SEZ(O_2) \leq FBZ(O_3)$ auch nicht erfüllt wird. Deshalb ist das Simulationsmodell fehlerhaft, es soll noch weiter korrigiert werden. Aus der oben erläuterten Vorgehensweise kann man sagen, dass eine strukturelle V&V-Technik wie Test von Teilmodellen, besonders effizient für die auf THORNs basierenden Simulationsmodelle ist.

7. Zusammenfassung und Ausblick

In der vorliegenden Arbeit werden die Anwendungsmöglichkeiten der Verifikation- und Validierungstechniken für ein auf THORNs basierendes Simulationsmodell untersucht.

Ausgehend von den Grundlagen aus Kapiteln zwei und drei werden die Grundkenntnisse vom ereignisdiskreten System, Petri-Netzen und THORNs vorgestellt. In dieser Arbeit werden ausschließlich die objektorientierten Petri-Netze betrachtet. Deswegen soll man an dieser Stelle die Eigenschaften von einem objektorientierten Ansatz kennen, wie z.B. Abstraktion, Kapselung, Vererbung und Polymorphie. Durch die Simulation von Petri-Netzen können die alltäglichen Arbeitsabläufe vereinfacht dargestellt werden. Aber es ist zu beachten, dass klassische Petri-Netze die Darstellungsbegrenztheit für komplexe Systeme haben. Allerdings verstärken THORNs die Modellierungsfähigkeit von Petri-Netzen, z.B. spezifische Stellenstruktur, spezifische Kantentypen, mögliche Zeitangabe, Unternetze usw. THORNs können auch mehrere Informationen als klassische Petri-Netze enthalten und lassen die Formulierung über komplexe Systeme vereinfachen. Es ist zweifellos, dass mit Hilfe des auf THORNs basierenden Simulationsmodells das reale System realitätsnaher wiedergegeben werden kann. Wenn ein reales System zu groß ist, kann man es zuerst in einige Teile unterteilen und danach das einzelne Teilsystem analysieren.

Aufgrund der Modellierungsaffinität für technische Aufgaben wurde in den neunziger Jahren an den deutschen Universitäten eine Reihe von Petrinetz-basierten/-orientierten Werkzeugen entwickelt (Schnieder 1999). Man kann nach ihren eigenen Bedürfnissen das geeignete Werkzeug zur Modellierung des realen Systems auswählen, z.B. die von TU Dortmund entwickelte Software HiQPN-Tool wird normalerweise zur Spezifikation und Analyse ereignisdiskreter Systeme eingesetzt.

Der Schwerpunkt dieser Arbeit liegt darin, welche V&V-Techniken geeignet für auf THORNs basierendes Simulationsmodell sind, darunter welche V&V-Techniken automatisch arbeiten können. Nach der sorgfältigen Untersuchung des erstellten Simulationsmodells von einer Logistikstation ist es ersichtlich, dass fast alle Management V&V-Techniken (Begutachtung, Strukturiertes Durchgehen, Schreibtischtest usw.) nicht automatisch arbeiten können. Da der V&V-Prozess mit solchen Techniken hauptsächlich durch die Kommunikationen zwischen

Menschen durchgeführt wird, d.h. menschliche Intelligenzen werden dem V&V-Prozess hinzugefügt. Deswegen kann die menschliche Arbeit in diesem Fall durch die Software nicht ersetzt werden.

Des Weiteren ist es auffällig, dass die automatischen Techniken Monitoring, Animation, Trace-Analyse in dem Modell angewendet werden können. Um solche automatische V&V-Techniken in dem Modell reibungslos anzuwenden, wird den Fachexperten die Anforderung gestellt, praktische und anwenderfreundliche Software für die automatische V&V-Techniken zu entwickeln. Am bestens kann eine integrierte Software für V&V-Techniken angeboten werden, in der mehrere V&V-Techniken einbezogen werden.

Verifikation und Validierung für Simulationsmodelle ist ein beliebtes Forschungsthema in den Bereiche Produktion und Logistik. Das auf Petri-Netzen basierende Simulationsmodell ist vernünftig, aber die mit Hilfe vom Petrinetz-Werkzeug modellierten Prozesse werden schließlich von den Menschen benutzt. In der Zukunft wird es untersucht, wie man das Gleichgewicht zwischen den Menschen und Software finden kann. Das ist ein sinnvolles Thema.

Literaturverzeichnis

Balci O (1990): Guidelines for successful simulation studies. In: Balci O, Sadowski RP, Nance RE (Hrsg) Proceedings of the 1990 Winter Simulation Conference, New Orleans (USA). IEEE, Piscataway, S. 25-32

Balci O (1998): Verification, Validation and Testing. In: Banks J (Hrsg) Handbook of Simulation, John Wiley & Sons, New York

Balci O (2003): Validation, verification and certification of modeling and simulation applications. In: Chick S, Sanchez PJ, Ferrin E, Morrice DJ (Hrsg.) Proceedings of the 2003 Winter Simulation Conference, S. 150-158

Balzert H (2009): Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering, 3. Aufl. Heidelberg: Spektrum Akademischer Verlag

Bastide R (1995): Approaches in unifying Petri Nets and the Object-Oriented Approach. In: Agha G, de Cindio F: Proceedings of the 1st Workshop on Object-Oriented Programming and Models of Concurrency

Bresser (2004), Validierung und Verifikation:
http://www2.cs.uni-paderborn.de/cs/ag-schaefer/Lehre/Lehrveranstaltungen/Seminare/AEizS/Abgaben/Folien/11_Validierung_TBresser.pdf, geprüft am 10.10.2016

Carson JS II (2002): Model verification and validation. In: Yücesan E, Chen C-H, Snowdon JL, Charnes JM (Hrsg) Proceedings of the 2002 Winter Simulation Conference, San Diego (USA). IEEE, Piscataway, S. 52-58

Eazystock(2015): White Paper: Der Sicherheitsbestand im Lager-Berechnungen, Methoden, Tipps, <http://www.eazystock.com/wp-content/uploads/2015/06/Whitepaper-Sicherheitsbestand-im-Lager.pdf>, geprüft am 10.10.2016

Feja S, Witt S, Speck A (2013): Automatische Validierung von Prozessmodellen, http://www.fh-kiel.de/fileadmin/data/wirtschaft/ProzessManagementForum2012/PMF_2013/Feja_Witt_automatische_Validierung.pdf, geprüft am 26.10.2016

Fishman G.S.(1973): Concepts and methods in discrete event digital simulation. Wiley, New York

- Frommholz I (2007): Prozessmodellierung mit Petri-Netzen, [http://www.is.informatik.uni-
duisburg.de/courses/ie_ss07/folien/folien-prozesse.pdf](http://www.is.informatik.uni-duisburg.de/courses/ie_ss07/folien/folien-prozesse.pdf), geprüft am 19.10.2016
- Goldschmidt H, Louen C (2014): Prozessautomatisierung Petri-Netze, [http://aks.uni-
du-
e.de/htm/index.php?option=com_remository&Itemid=30&func=fileinfo&id=344&lang=en](http://aks.uni-
du-
e.de/htm/index.php?option=com_remository&Itemid=30&func=fileinfo&id=344&lang=en),
geprüft am 19.10.2016
- He Z (2013): Petri-Nets,
[http://wenku.baidu.com/link?url=9ZIC6fOJ5Xfeu6n4I3uqdYtQcsZ14JKMQ0w35CVzZ3FAaHUu
at49U0SOA3GkZDP8WHf5xfcR9txrvcC8UZdtaZB9JzePgRBFzmY5X6B5Ra](http://wenku.baidu.com/link?url=9ZIC6fOJ5Xfeu6n4I3uqdYtQcsZ14JKMQ0w35CVzZ3FAaHUu
at49U0SOA3GkZDP8WHf5xfcR9txrvcC8UZdtaZB9JzePgRBFzmY5X6B5Ra), geprüft am
19.10.2016
- Hermann CF (1967): Validation problems in games and simulations with special reference to
model of international politics. Behavioral Science 12 (1967), S. 216-231
- Homburg C, Giering A, Hentschel F (1999): Der Zusammenhang zwischen Kundenzufrieden-
heit und Kundenbindung. In: Die Betriebswirtschaft, 59, S. 174-195.
- IRT RWTH Aachen Universität (2016): Ereignisdiskrete Systeme,
[http://www.irt.rwth-
aachen.de/fileadmin/IRT/Download/Lehre/MRT/De/RT_Aufl_33_Kap14_S443_bis_470.pdf](http://www.irt.rwth-
aachen.de/fileadmin/IRT/Download/Lehre/MRT/De/RT_Aufl_33_Kap14_S443_bis_470.pdf),
geprüft am 10.10.2016
- König B (2011): Vorlesung "Modellierung", "Modellierungsmethoden der Informatik",
<http://www.ti.inf.uni-due.de/fileadmin/public/teaching/mod/slides/ws201112/petri-2x2.pdf>,
geprüft am 19.10.2016
- Law AM (2007): Simulation Modeling and Analysis, 4. Aufl. Boston: McGraw-Hill
- León F, Kiencke U (2013): Ereignisdiskrete Systeme: Modellierung und Steuerung verteilter
Systeme, 3. Aufl. München: Oldenbourg Verlag München
- Li D; Chen Q; Liu Q (2007): Research and Verification of Petri-net Based Time Model in Work-
flow. In: Computer Engineering Vol.33 No.7, Seite 78-80
- Link N (2016): Petri-Netze, [http://www.iwi.hs-
karlsruhe.de/~lino0001/skripte/Automatisierungsprojekte/FolienPDF/VorlesungAuto2_5.pdf](http://www.iwi.hs-
karlsruhe.de/~lino0001/skripte/Automatisierungsprojekte/FolienPDF/VorlesungAuto2_5.pdf),
geprüft am 19.10.2016
- Liu Y, Wang D, Feng Z (2005): Simulation Modeling and Simulation Diplomacy of Discrete
Event System. In: Journal of Southwest China Normal University (Natural Science) Vol. 30
No.6, Seite 1019-1025
- Matheguru (2016): Chi-Quadrat-Test, <http://matheguru.com/stochastik/248-chi-quadrat->

test.html, geprüft am 28.10.2016

Mülle J (2007): Petrinetze und Workflow Modellierung,
<https://www.ipd.kit.edu/~ipd/institut/workflow/kap03.pdf>, geprüft am 19.10.2016

Rabe M, Spieckermann S, Wenzel S (2008): Verifikation und Validierung für die Simulation in Produktion und Logistik, Berlin, Heidelberg: Springer Verlag

Riesig W (2010): Petrinetze: Modellierungstechnik, Analysemethoden, Fallstudien, Wiesbaden: Vieweg+Teubner Verlag

Robinson S (2004): Simulation: The Practice of model development and use, Chichester: John Wiley & Sons

Sargent RG (2005): Verification and validation of simulation models. In: KuhlME, Steiger NM, Armstrong FB, Joines JA (Hrsg) Proceedings of the 2005 Winter Simulation Conference, Orlando (USA). SCS International, San Diego, S. 130-143

Schlick C (2012): Dynamische Unternehmensmodellierung und -simulation: Lehrinheit 09,
http://www.iaw.rwth-aachen.de/files/aw4_le09_wise12_folien+notizen.pdf, geprüft am 19.10.2016

Schlick C (2012): Dynamische Unternehmensmodellierung und -simulation: Lehrinheit 10,
http://www.iaw.rwth-aachen.de/files/Vorlesungen/WS2013/AW4/AW4_LE10_WiSe13_Folien+Notizen.pdf, geprüft am 19.10.2016

Schnieder E (1999): Methoden der Automatisierung, Braunschweig/ Wiesbaden: Verlag Vieweg

Schöf S, Wieting R, Sonnenschein M (1997): Abschlussbericht des Projekts DNS (Distributed Net Simulation), Oldenburg

Swider CL, Bauer KW Jr, Schuppe TF (1994): The Effective Use of Animation in Simulation Model Validation. In: Tew JD, Manivannan S, Sadowski DA, Seila AF (Hrsg) Proceedings of the 1994 Winter Simulation Conference, Lake Buena Vista (USA). IEEE, Piscataway, S. 633-640

Tarakos (2016): Farikplanungssoftware für die Industrie & Logistik,
<http://www.tarakos.de/fabrikplanungssoftware.html>, geprüft am 27.10.2016

Tian W (2006): The Simulation model of Discrete Event System based on Petri-Nets and its Application Research,
http://xueshu.baidu.com/s?wd=paperuri%3A%2866a8b6e03a04bb71899f656c0ae8aca0%29&filter=sc_long_sign&tn=SE_xueshusource_2kduw22v&sc_vurl=http%3A%2F%2Fwww.doc8

8.com%2Fp-776474731991.html&ie=utf-8&sc_us=14005124683594264835, geprüft am 18.10.2016

Universität Innsbruck (2016): Tabelle: $\chi^2_{(q)}$ -Verteilung,

https://www.uibk.ac.at/econometrics/einf/tab_chisq.pdf, geprüft am 28.10.2016

VDI Richtlinie 3633 Blatt 1 (2008): Simulation von Logistik-, Materialfluss- und Produktionssystemen - Grundlagen. VDI-Handbuch Materialfluss und Fördertechnik 8, Beuth, Berlin

Voigt K (2016): Automatisierung,

<http://wirtschaftslexikon.gabler.de/Definition/automatisierung.html#definition>, geprüft am 25.10.2016

Wenzel S, Weiß M, Collisi-Böhmer S, Pitsch H, Rose O (2008): Qualitätskriterien für die Simulation in Produktion und Logistik, Berlin, Heidelberg: Springer Verlag

Wikia (2016): Trace-Check, <http://de.software.wikia.com/wiki/TRACE-CHECK>, geprüft am 27.10.2016

Zapf M, Heinzl A (1998): Ansätze zur Integration von objektorientierten Konzepten und Petri-Netzen, Bayreuth: Universität Bayreuth

Zhao D (2010): Logistics Enterprise Business Process Optimization and Simulation based on Object-oriented Time Petri Nets,

http://xueshu.baidu.com/s?wd=paperuri%3A%28376c2541dce96aba5a2e38117a22d4f6%29&filter=sc_long_sign&tn=SE_xueshusource_2kduw22v&sc_vurl=http%3A%2F%2Fwww.doc88.com%2Fp-7718886100714.html&ie=utf-8&sc_us=7302333286814196871, geprüft am 26.10.2016

Zhou F, Wu B, Gu Q, Chen D (2005): Modeling and Correctness Checking of Petri-Net-Based Workflow. In: Computer Science Vol.32 No.2, S. 121-124

Abbildungsverzeichnis

ABBILDUNG 1: PROZESSE, AKTIVITÄTEN UND EREIGNISSE (FISHMAN 1973)	6
ABBILDUNG 2: ELEMENTE VON PETRI-NETZEN (BALZERT 2009, S. 303)	7
ABBILDUNG 3: EIN BEISPIEL VON EINEM PETRI-NETZ (FROMMHOLZ 2007, S.16)	8
ABBILDUNG 4: KONSTANTE KANTENBESCHRIFTUNG (REISIG 2010, S. 27).....	9
ABBILDUNG 5: VARIABLE KANTENBESCHRIFTUNG (REISIG 2010, S. 29).....	10
ABBILDUNG 6: BEDINGUNGS/EREIGNIS-NETZ (LINK 2016, S. 3).....	11
ABBILDUNG 7: STELLEN/TRANSITIONS-NETZ (LINK 2016, S. 10)	12
ABBILDUNG 8: AKTIVIERUNG UND SCHALTEN (LEÓN, KIENCKE 2013, S.340)	13
ABBILDUNG 9: DIE BEZIEHUNGEN ZWISCHEN DEN TRANSITIONEN (TIAN 2006, S.35)	14
ABBILDUNG 10: EINBETTUNG VON OBJEKTEN IN PETRI-NETZE (BASTIDE 1995, S. 1).....	16
ABBILDUNG 11: ZWEI SACHBEARBEITER BEDIENEN DIE BANKKUNDEN (SCHLICK 2012, S. 3).....	18
ABBILDUNG 12: DARSTELLUNG VON THOR-NETZ-STELLEN (SCHÖF, SONNENSCHNEIN 1997, S. 11).....	20
ABBILDUNG 13: DARSTELLUNG VON THOR-NETZ-KANTEN (SCHÖF, SONNENSCHNEIN 1997, S. 12).....	21
ABBILDUNG 14: DARSTELLUNG DER HIERARCHISCHEN NETZELEMENTE (SCHÖF, SONNENSCHNEIN 1997, S.19).....	27
ABBILDUNG 15: VORGEHENSMODELL BEI DER SIMULATION MIT V&V.....	39
ABBILDUNG 16: VERWENDBARKEIT VON V&V-TECHNIKEN IM VERLAUF DER SIMULATIONSSTUDIE	41
ABBILDUNG 17: VORGEHENSMODELL BEI DER ERSTELLUNG VON SIMULATIONSMODELLEN NACH VDI-RICHTLINIE 3363, BLATT 1 (RABE, SPIECKERMANN, WENZEL 2008, S. 32)	48
ABBILDUNG 18: NACHRICHTENAUSTAUSCH	50
ABBILDUNG 19: HAUPTNETZ DER LOGISTIKSTATION.....	51
ABBILDUNG 20: UNTERNETZ_ BESTELLZENTRUM	51
ABBILDUNG 21: UNTERNETZ_ RATIONIERENZENTRUM	52
ABBILDUNG 22: UNTERNETZ_ WARENLAGERZENTRUM	53
ABBILDUNG 23: UNTERNETZ_ TRANSPORTZENTRUM	54
ABBILDUNG 24: BESTANDNIVEAU ÜBER DIE ZEIT (EAZYSTOCK 2015, SEITE 2)	60
ABBILDUNG 25: ANIMATION VON EINER FABRIK (TARAKOS, 2016).....	63
ABBILDUNG 26: DEFINITION VON DEM CHI-QUADRAT-TEST (MATHEGURU, 2016).....	66

Tabellenverzeichnis

TABELLE 1: DEUTSCHE UND ENGLISCHE BEZEICHNUNGEN VON V&V-TECHNIKEN (RABE, SPIECKERMANN, WENZEL 2008, S. 96).....	31
TABELLE 2: KRITERIEN FÜR DEN ERFOLG EINER BEGUTACHTUNG MIT BEZUG ZU DEN V&V-KRITERIEN (RABE, SPIECKERMANN, WENZEL 2008, S. 97)	32
TABELLE 3: KURZBESCHREIBUNGEN DER V&V-TECHNIKEN (ELEY 2012, S. 16)	36
TABELLE 4: AUTOMATISIERUNGSGRAD VON DEN V&V-TECHNIKEN.....	44
TABELLE 5: EINIGE VERTEILUNGSFUNKTION (ELEY 2012, SEITE 20)	65
TABELLE 6: χ^2 -VERTEILUNG (UNIVERSITÄT INNSBRUCK, 2016).....	67

Eidesstattliche Versicherung

Dong, Muzi

174112

Name, Vorname

Matr.-Nr.

Ich versichere hiermit an Eides statt, dass ich die vorliegende Bachelorarbeit/Masterarbeit* mit dem Titel

Untersuchung von Validierungsmöglichkeiten für modulare, auf objektorientierten Petri-Netzen

basierende Simulationsmodelle

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Dortmund, den 22.11.2016

Ort, Datum

Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50.000,00 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden. (§ 63 Abs. 5 Hochschulgesetz - HG -)

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird gfls. elektronische Vergleichswerkzeuge (wie z.B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die oben stehende Belehrung habe ich zur Kenntnis genommen:

Dortmund, den 22.11.2016

Ort, Datum

Unterschrift