technische universität dortmund

IT in Produktion und Logistik

Masterarbeit

Entscheidungsbäume zur Prüfung der Simulationswürdigkeit von Produktionssystemmodellen

verfasst von: Wenbin Lei

Studiengang: Wirtschaftsingenieurwesen

Matrikel.-Nr.: 164369

Ausgegeben am: 16. 02. 2016
Eingereicht am: 27. 07. 2016

Erstprüfer: Univ.-Prof. Dr.-Ing. Markus Rabe

Zweitprüfer: Dipl.-Geoinf. Maik Deininger

Inhaltverzeichnis

Abkürzungsverzeichnis

V&V	Verifikation und Validierung
DES	Discrete Event Simulation
TDIDT	Top- Down Induction of Decision Trees
ID3	Iterative Dichotomiser 3
CART	Classification and Regression Tree
CHAID	Chi-square Automatic Interaction Detectors
QUEST	Quick, Unbiased, Efficient Statistical Tree

Abbildungsverzeichnis

Tabellenverzeichnis

Algorithmenverzeichnis

1 Einleitung

Das Thema der Arbeit umfasst die Analyse und Untersuchung der Simulationsdaten von Produktionssystemmodellen mit Hilfe von Entscheidungsbäumen. Die Vielzahl von zeit- und zufallsabhängigen Systemgrößen sowie deren komplexe Wirkungszusammenhänge führen zur Einschränkung der mathematisch-analytischen Methoden zur Untersuchung und Beurteilung der Produktionssysteme (VDI-Richtlinie 3633 Blatt 1). Aus diesem Grund kommt die simulationsbasierte Optimierung vermehrt zum Einsatz, um die Produktionssysteme zu optimieren. Tatsächlich ist die Simulationsstudie einerseits effizienter und effektiver, andererseits jedoch aufwendiger und komplexer als die mathematisch-analytischen Methoden (Wenzel et al. 2008). Insbesondere erfolgt die Optimierung der Simulation durch eine gezielte Veränderung der Simulationsparameter, da die Simulation selbst über keine Funktion der Optimierung verfügt (VDI-Richtlinie 3633 Blatt 1). Mit Entwicklung der Simulationstechnik und zunehmender Komplexität der Produktionssysteme sind immer mehr Parameter in der Simulation enthalten. Aufgrund der spezifischen Einstellung der Parameter dauern die Simulationsläufe immer länger und somit werden die Simulationsstudien immer aufwendiger. Deshalb liegt die Aufgabe der vorliegenden Arbeit in der Suche nach einem Lösungsansatz zur passenden Auswahl und Einstellung der Simulationsparameter. Auf diese Weise sollen nur noch die Simulationen mit vielversprechenden Ergebnissen durchgeführt werden.

Entscheidungsbäume sind eine weit verbreitete Methode im Bereich des Data Mining zur Analyse von großen Datenmengen. Um eine optimale Lösung für das Simulationsmodell zu erhalten, wird die Simulationsstudie wiederholt durchgeführt (VDI-Richtlinie 3633 Blatt 1), weshalb eine große Menge an Simulationsdaten gespeichert wird. Daher sollten die Daten zunächst für eine erfolgreiche Konstruktion der Entscheidungsbäume vorverarbeitet werden. Anschließend lassen sich die Entscheidungsbäume in Abhängigkeit der Trainingsdaten mit Hilfe der entsprechenden Algorithmen generieren. Nach der Analyse der Ergebnisse durch den Entscheidungsbaum kann ermittelt werden, wie sich die verschieden Simulationsparameter auf das Simulationsergebnis auswirken können. Im letzten Teil verfolgt die Arbeit das Ziel, einen Filter auf Basis des ausgewählten Entscheidungsbaums zu entwerfen. Die Hauptaufgabe des Filters besteht darin, die Eingabedaten für die Simulationsstudie auszuwählen und nur diese ausgewählten Eingabedaten für das Simulationsmodell freizugeben.

Nachdem im ersten Kapitel die Problemstellung, die Zielsetzung und der Aufbau dieser Arbeit erörtert wurden, werden im zweiten Kapitel die notwendigen Grundlagen erarbeitet. Dabei werden zuerst die Grundbegriffe im Bereich der Produktionssysteme erläutert, um ein grundlegendes Verständnis für das folgende Konzept zu gewährleisten. Anschließend werden die Grundlagen der ereignisdiskreten Simulation in Bezug auf Produktionssysteme vorgestellt. Zur Entwicklung des Simulationsmodells werden zunächst das grundsätzlichen Vorgehensweisen und Abläufe erläutert. Daneben werden die Grundzüge der Datenanalyse im Rahmen der Si-

mulationsstudien vorgestellt. Im Anschluss daran werden die grundlegenden Begriffe, die Entwicklung der simulationsbasierten Optimierung sowie die Kopplung von Simulation und Optimierung erarbeitet.

Zum Ende des zweiten Abschnitts werden erst einmal die notwendigen Grundbegriffe der Datenvorverarbeitung vorgestellt. Damit die Entscheidungsbäume mithilfe der Algorithmen erfolgreich generiert werden können, müssen nämlich zuerst die Trainingsdaten im Rahmen der Datenvorverarbeitung adäquat verarbeitet werden. Anschließend werden die notwendigen Grundlagen der Entscheidungsbäume dargestellt, wozu die Potentiale und Anwendungsbereiche genauer betrachtet werden. Auch werden die Konstruktionsverfahren der Entscheidungsbäume erläutert, wobei hier insbesondere die Kriterien der Attributauswahl und die relevanten Algorithmen vorgestellt werden. Weiterhin werden die grundlegenden Kenntnisse der Evaluation der Entscheidungsbäume erarbeitet, da die Performance der Entscheidungsbäume für die Weiteranwendung von großer Bedeutung ist. Abschließend werden die verschiedenen Typen der Entscheidungsbäume vorgestellt und miteinander verglichen, womit vor allem die Auswahl eines anwendbaren Entscheidungsbaums getroffen werden kann.

Nach Vorstellung der relevanten Grundlagen wird im dritten Kapital einerseits die Auswahl des geeignetsten Entscheidungsbaum durchgeführt, andererseits werden die Entscheidungen durch den Entscheidungsbaum ausgewertet. Dafür werden die gegebenen Datensätze im Rahmen der Datenvorverarbeitung behandelt, um sowohl die Effizienz, als auch die Korrektheit der Entscheidungsbäume zu gewährleisten. Im Anschluss daran gliedert sich die Auswahl des Entscheidungsbaums in zwei Schritte, nämlich die Vorauswahl nach dem Kriterium der Anwendbarkeit und die tatsächliche Auswahl durch Evaluation und Vergleich der Entscheidungsbäume. Danach wird der ausgewählte Entscheidungsbaum zur Analyse der Ausgabedaten eingesetzt. Im Rahmen der Analyse wird die Qualität der generierten Entscheidungen ausgewertet und eine Kosten-Nutzen-Analyse in Abhängigkeit der Konfusionsmatrix ausgeführt. Zum Schluss des dritten Kapitels werden die gewonnenen Erkenntnisse zum Entwurf des Konzepts zusammengefasst.

Im vierten Kapitel wird zunächst ein Konzept entworfen, welches anschließend anhand bestimmter Anforderungen validiert wird. Das Konzept besteht vor allem aus den beiden Komponenten „Datenvorverarbeitung“ und „Filter mittels C4.5 Entscheidungsbaum“. Nach dem Entwurf wird zuerst jede Komponente einzeln dahingehend überprüft, ob die entsprechenden Anforderungen erfüllt sind. Schließlich soll die Kopplung von Simulation und Konzept anhand der entsprechenden Anforderungen der simulationsbasierten Optimierung validiert werden.

2 Grundlagen

In diesem Kapitel werden die notwendigen Grundlagen von Produktionssystemen, Simulation und Optimierung sowie von Entscheidungsbäumen vorgestellt. Dabei werden zuerst die grundlegenden Begriffe des Produktionssystems erklärt, da das Simulationsmodell auf Basis des Produktionssystems aufgebaut ist. Im Bereich der ereignisdiskreten Simulation sollen ebenso die grundlegenden Aspekte vorgestellt werden, bevor eine detaillierte Einsicht in die Komponenten, in das Vorgehen einer Simulation und in die simulationsbasierte Optimierung erfolgen kann. Damit ein Gesamtkonzept erstellt werden kann, welches das Potential und die Anwendbarkeit der Entscheidungsbäume bestmöglich ausschöpft, werden die Themen nachfolgend detailliert vorgestellt. Dazu gehören im Bereich der Entscheidungsbäume neben den grundlegenden Definitionen und Anwendungsmöglichkeiten auch die Konstruktionsverfahren und die Typen nach Anwendbarkeit.

2.1 Grundlagen der Produktionssystemmodellen

Im diesen Abschnitt werden die Grundbegriffe von Produktionssystemen aufgearbeitet. Dabei werden zunächst die Grundbegriffe erläutert. Anschließend wird das Vorgehen der Modellierung der Produktionssysteme vorgestellt.

Der Begriff des Produktionssystems setzt sich aus den Begriffen Produktion und System zusammen. Unter dem Begriff der „*Produktion*“ kann man nach REFA alle Bereiche eines Unternehmens verstehen, die an der Herstellung von Erzeugnissen direkt oder indirekt beteilig sind. Dabei beinhaltet die Produktion die Prozesse wie Entwicklung, Beschaffung, Qualitätswesen und Fertigung inklusiv Instandhaltung, Teilfertigung, Montage und Logistik (REFA 1991). Laut DIN 25424 wird das System als “*die Zusammenfassung von technisch organisatorischen Mitteln zur autonomen Erfüllung eines Aufgabenkomplexes*“ definiert.

Basierend auf den vorangegangenen Erläuterungen kann das Produktionssystem als ein soziotechnisches System verstanden werden, in dem Menschen, Maschinen und Werkstoffe oder Dienstleistungen anhand einer Produktionsphilosophie und mit Hilfe standardisierter Methoden zur Erbringung von Gütern und Dienstleistungen entlang der Wertschöpfungsprozesse zusammenwirken, wie in der Abbildung 2-1 dargestellt (Günther und Tempelmeier 2012).

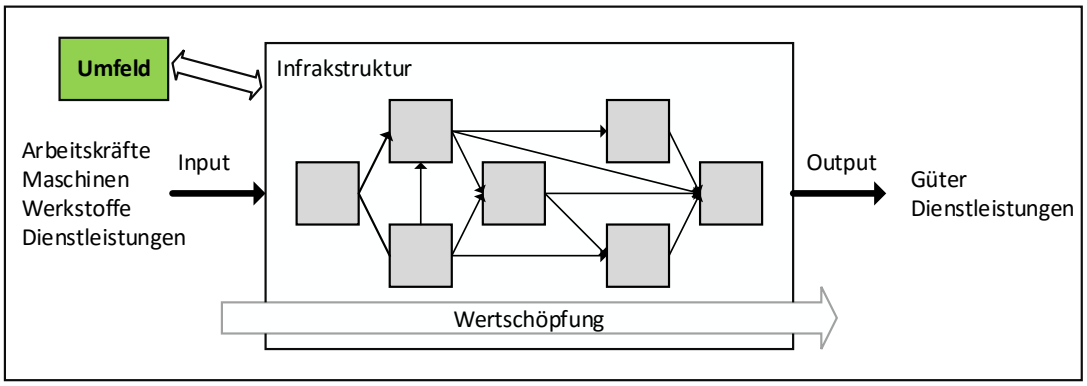


Abbildung 2-1: Das Produktionssystem in Anlehnung an Günther und Tempelmeier (Günther und Tempelmeier 2012)

Die Entwicklung der Produktionssysteme war und ist vor allem von Marktanforderungen und Technikentwicklungen abhängig (Günther und Tempelmeier 2012). Zurzeit führen die zunehmende Variantenvielfalt und neue Anforderungen an Flexibilität sowie innen- und außerbetrieblichen Wandlungen zu einer Weiterentwicklung der Produktionssysteme. Daher gewinnt der Begriff der Wandlungsfähigkeit immer mehr an Bedeutung. In diesem Kontext entstand das wandlungsfähige Produktionssystem und das modulare Produktionssystem (Dombrowski und Mielke 2015 ; Rauch 2013).

Damit ein Produktionssystem mit Simulationstechnik untersucht werden kann, sollte zunächst ein geeignetes Modell aufgebaut werden. Dazu sollte sowohl das Vorgehen der Modellbildung, als auch die relevanten Anforderungen des Produktionssystems vorgestellt werden.
Die Simulationsstudie setzt die Erstellung eines Modells des Produktionssystems voraus (VDI-Richtlinie 3633 Blatt 1). Unter dem Begriff Modell versteht man laut VDI-Richtlinie 3633 Blatt1 *„Die Vereinfachte Nachbildung eines geplanten oder existierenden Systems mit seinen Prozessen in einem anderen begrifflichen oder gegenständlichen System“.*

Die Modellbildung eines Produktionssystems gliedert sich in drei Schritte (Rabe et al. 2001). Zuerst sollte ein konzeptionelles Modell aufgebaut werden, das zur Zielformulierung in der Simulationsstudie und Beschreibung der Grundstruktur und der relevanten Daten und Elemente des Simulationsmodells eingesetzt wird. Anschließend ist ein programmiertes Modell auf Basis des konzeptionellen Modells mittels einer Programmiersprache zu bilden. Hierbei wird das Modell auf seine Richtigkeit getestet und validiert. Schließlich entsteht ein experimentierbares Modell mit der Einbindung der Szenarien, die im Versuchsplan durch die Vorgehensweise der Experimente festgelegt werden (Rabe et al. 2001). Bei der Modellbildung sollten Kriterien wie Richtigkeit, Relevanz, Klarheit, Vergleichbarkeit und systematischer Aufbau anhand der Grundsätze ordnungsmäßiger Modellierung in Betracht genommen werden, damit der Aufwand und die Effizienz der Simulationsstudie gewährleistet werden (Becker et al. 2012; Wenzel et al. 2008).

Um ein rationales Modell des Produktionssystems zu bilden, sollen sowohl die Charakter der Produktionssysteme als auch die Anforderungen der Produktionssysteme erläutert werden.

Hierbei lässt sich das Vorgehen zur Modellierung der modularen Produktionssysteme in diesem Abschnitt als ein Beispiel eines Produktionssystems erläutern. Die Modularisierung spielt eine wichtige Rolle für die Verbesserung der Wandlungsfähigkeit und für die Reduktion der Komplexität von Produktionssystemen (Schneider et al. 2010). Weiterhin stellen die modularen Produktionssysteme eine Robustheit zur Auslegung der Änderungstreiber der Produktionssysteme dar (Neuhausen 2001).

Modularität wird laut Heinen et al. definiert als: „Modularität beschreibt die Fähigkeit eines Produktionssystems, standardisierte, funktionsfähige Einheiten oder Elemente einfach auszutauschen." (Heinen et al. 2008; Rauch 2013). Der Vorteil der Modularität besteht vor allem in die Verwendbarkeit der standardisierten und funktionsfähigen Produktionseinheiten, die von den Produkten oder Prozessen unabhängig sind (Rauch 2013). Dadurch werden die prozessspezifischen Maschinenkosten erheblich reduziert, wie in Abbildung 2-2 dargestellt wird (Rauch 2013).

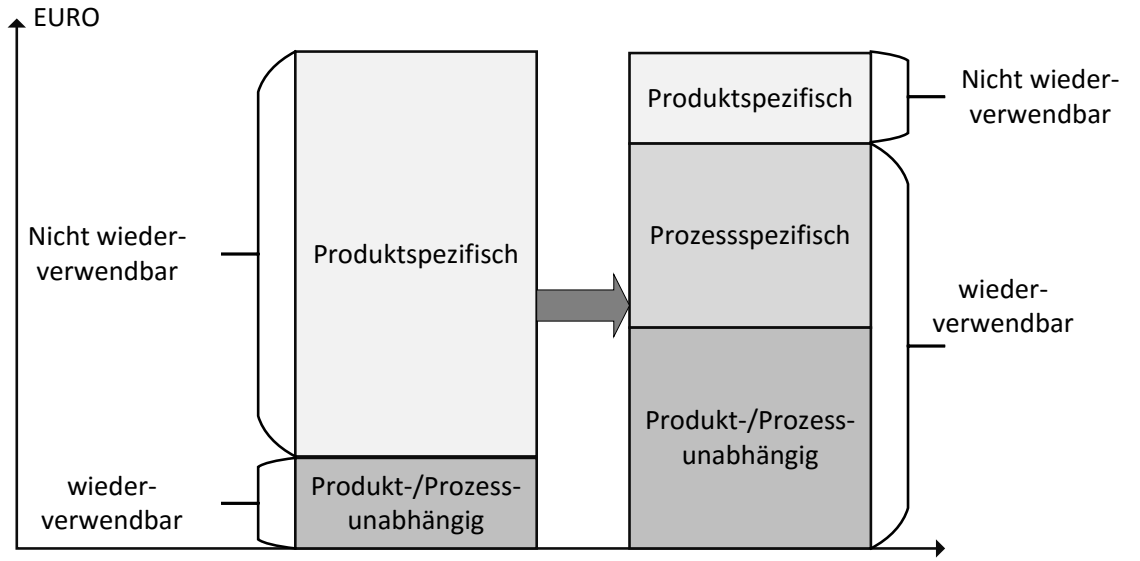


Abbildung 2-2: Vorteile modularer Fertigungseinrichtung (Rauch 2013)

Aufgrund der hohen Wiederverwendbarkeit durch die Modularität in der Produktion von Fertigungseinrichtungen und Arbeitskräften können die Produktionssysteme nach den Veränderungen schnell und angepasst mit den vorbestimmten Module rekonfiguriert werden, damit neue oder zusätzliche Aufträge abgewickelt werden können (Rauch 2013). Darüber hinaus kann die Modularität als grundlegende, zentrale und wesentliche Methode im Rahmen der Wandlungsfähigkeit zur Gestaltung von Produktionsmitteln betrachtet werden (Rauch 2013). Mit der Modularität wird der Vernetzungsgrad der Informationen und Ressourcen in der Produktionssysteme erheblich reduziert. Weiterhin, wie in Abbildung 2-3 gezeigt, wirken sich die Veränderungen der Produktekomponenten durch die Gestaltung der modularen Produktionssysteme nur auf bestimmte Elemente der Produktionssysteme aus (Neuhausen 2001). Das heißt, die zusätzlichen Aufträge oder Markteinflüsse beeinflussen nur bestimmte Produktionseinrichtungen oder -prozessschritte anstatt das ganze Produktionssysteme.

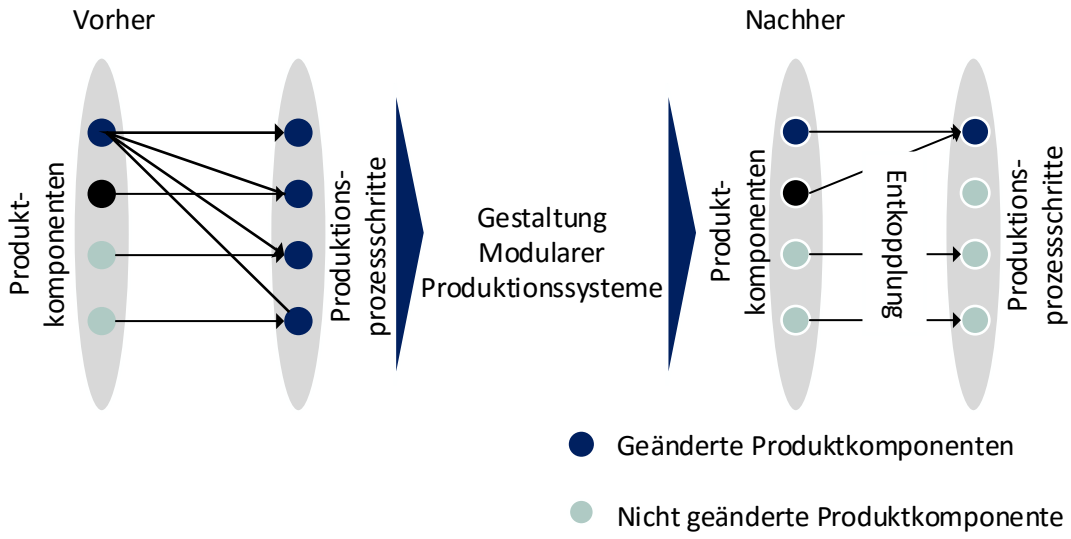


Abbildung 2-3: Genereller Ansatz zur Gestaltung modularer Produktionssysteme (Neuhausen 2001)

Die Anforderungen an das Modell zur Gestaltung modularer Produktionssysteme gliedern sich laut Neuhausen (2001) in allgemeine und spezielle Anforderungen (Neuhausen 2001). Während die allgemeinen Anforderungen ausgehend von der Anwendbarkeit abzuleiten sind, lassen sich die speziellen Anforderungen an die Methodik der Modellgestaltung orientieren (Neuhausen 2001). Die allgemeine Anforderung der Modellierung orientiert sich vor allem an der Anwendung der Modelle, Anpassbarkeit, Verständlichkeit, geringer Erstellungsaufwand und Erkenntnisgewinn aus dem Modell (Neuhausen 2001). Die allgemeinen Anforderungen können ebenfalls zur Gestaltung der Modelle anderer Produktionssystemen eingesetzt werden. Darüber hinaus werden die speziellen Anforderungen an das Modell der modularen Produktionssysteme von der Methodik der Modellgestaltung abgeleitet. Die Anforderungen beinhalten vor allem die Änderungen der Kundenanforderungen sowie die Senkung der Wechselwirkungen innerhalb des Produktionssystems (Neuhausen 2001).
In Kombination mit den grundlegenden Kenntnissen der Modellbildung und der modularen Produktionssysteme können die Simulationsmodelle von modularen Produktionssystemen aufgebaut werden. Nach der Modellbildung kann das Verhalten des Produktionssystems mit Hilfe der simulationsbasierten Methode analysiert, bewertet und optimiert werden. Dazu werden die relevanten Grundlagen der ereignisdiskreten Simulation in folgendem Abschnitt erläutert.

2.2 Grundlagen der ereignisdiskreten Simulation

Im diesen Abschnitt werden die grundlegenden Vorgehensweisen der Simulation und Optimierung aufgearbeitet. Zunächst werden die Klassifizierung der Simulation und die Anwendungsfelder der ereignisdiskreten Simulation (engl. Discrete Event Simulation: DES) im Industriebereich erläutert, wobei vor allem die Vorgehensweise der Simulationsstudie beschrieben wird. Insbesondere sind die Simulationsparameter und -daten für die Durchführung und Untersu-

chung der Simulationsstudie von großer Bedeutung, weshalb auch die Grundlagen der Datenanalyse erläutert werden. Zum Schluss werden die Grundlagen der Kopplung von Simulation und Optimierung vorgestellt.

2.2.1 Klassifizierung und Anwendungsfeld

Nach VDI-Richtlinie 3633 Blatt 1 wird die Simulation als *„Nachbilden eines Systems mit seinen dynamischen Prozessen in einem experimentierbaren Modell, um zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind; insbesondere werden die Prozesse über die Zeit entwickelt.“* definiert. Die Auswahl der Methode zur Analyse eines Systems hängt grundlegend davon ab, wie das System modelliert werden kann. Wenn die Untersuchung eines Systems nicht durch die mathematisch-analytische Methode mit Hilfe von mathematischen Modellen realisierbar ist, wird die Simulation auf Basis von Simulationsmodellen zur Analyse und Bewertung des Systems eingesetzt. Insbesondere wenn das System aufgrund einer Vielzahl von zeit- und zufallsabhängigen Systemgrößen sowie deren komplizierter Wirkungszusammenhänge sehr komplex ist, ist die Simulation geeigneter für die Analyse und Optimierung des Systems (VDI-Richtlinie 3633 Blatt 1).

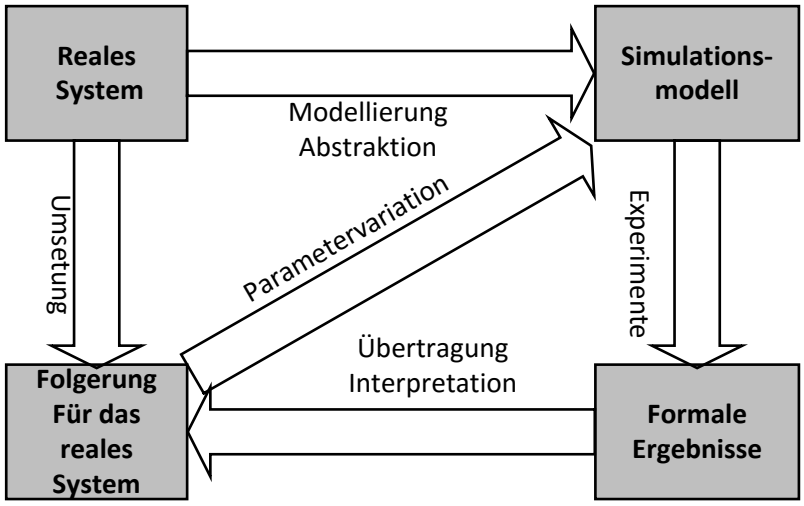


Abbildung 2-4: Vorgehen bei der Simulation nach Hrdliczka (1997)

Die Abbildung 2-4 stellt das Vorgehen bei der Simulation nach Hrdliczka (1997) dar. Zuerst wird ein Simulationsmodell durch Abstraktion und Reduktion anhand des realen Systems aufgebaut. Die Ausführung des Modells hängt von der Parametervariation ab. Dann erhält man die formalen Ergebnisse durch Experimente aus dem Simulationsmodell. Die Folgerung für das reale System erfolgt durch die Übertragung und Interpretation der formalen Ergebnisse. Da das Simulationsmodell eine vereinfachte Darstellung des realen Systems durch Abstraktion und Reduktion ist, sind nicht alle Kenntnisse auf das reale System übertragbar. Darüber hinaus gilt, dass ein vollständiges Wiedergeben des realen Systems durch das Simulationsmodell, die formalen Ergebnisse und die Folgerung für das reale System nicht realisierbar ist (Hrdliczka 1997; VDI-Richtlinie 3633 Blatt 1).

Eine Möglichkeit der Klassifikation der Simulation erfolgt in Abhängigkeit der Merkmale des Simulationsmodells. Anhand der Arten des Zufallsverhaltens und des zeitlichen Verhaltens der Simulationsmodelle gliedert sich die Simulation in eine stochastische vs. deterministische und dynamische vs. statische Simulation (Košturiak und Gregor 1995). Im Gegensatz zu der statischen Simulation, bei der die Experimente zu einen bestimmten Zeitpunkt ausgeführt werden, wie bei der Monte-Carlo Simulation, wird bei der dynamischen Simulation das zeitliche Verhalten des Systems repräsentiert (Fischer 2014). Die stochastische Simulation wird eingesetzt, wenn das System zufällige Ergebnisse beinhaltet, wie bspw. In Warteschlagensystemen. Im Gegensatz dazu wird mit der deterministischen Simulation ein System analysiert, welches keine zufälligen Komponenten enthält (Fischer 2014).

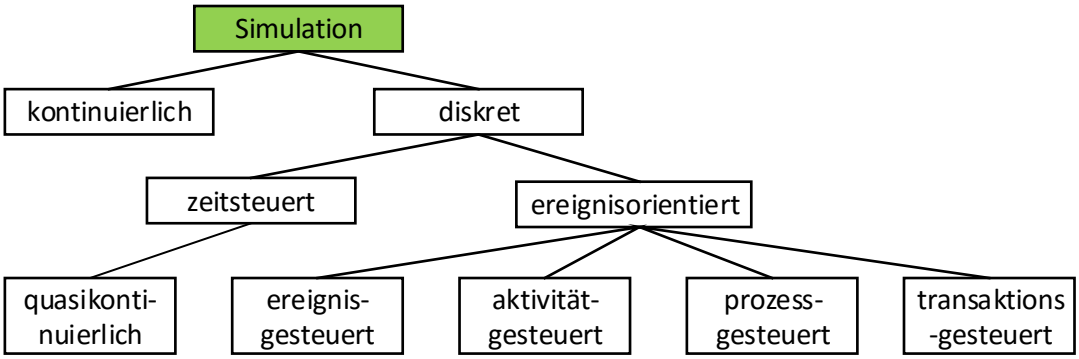


Abbildung 2-5: Klassifikation der Simulationsmethode (Mattern und Mehl 1989; Arnold et al. 2008)

Die Abbildung 2-5 stellt eine weitere Klassifikationsmöglichkeit einer Simulationsmethode dar, die in der Regel von der Art und Weise, wie beispielsweise dem Zeitverhalten und der Zustandsänderung der Simulationsmodelle abhängig ist (Arnold et al. 2008). Die Simulationsmethode gliedert sich anhand des Zeitverhaltens in kontinuierliche und diskrete Ansätze. Während die Zustandsänderungen bei der kontinuierlichen Simulation innerhalb der Simulation stetig erfolgen, wird das System bei der diskreten Simulation hingegen zu bestimmten Zeitpunkten ausgelöst (Košturiak und Gregor 1995; Mattern und Mehl 1989). Weiterhin unterscheidet sich die diskreten Simulation in eine zeitgesteuerte Simulation, wobei sich der Systemzustand nach der vorher festlegten Zeitinkrement verändert, und eine ereignisorientierte Simulation, bei der die Zustandsänderung durch das Eintreten neuer Ereignisse erfolgt (Mattern und Mehl 1989).

Aufgrund der zunehmenden Komplexität und Wandlungsfähigkeit der Produktionssysteme wie in Abschnitt 2.1 erläutert, ist die mathematisch-analytische Methode für die Untersuchung und Optimierung von komplexen Produktionssystemen an gewisse Grenzen gestoßen. Aus diesem Grund wird die Simulationstechnik vermehrt im Bereich der Produktion und Logistik eingesetzt (VDI-Richtlinie 3633 Blatt 1; Arnold et al. 2008). Daher kommt die ereignisdiskrete Simulation (Discrete Event Simulation) sowohl zur Planung, Steuerung und Verbesserung der Prozesse von Produktionssystemen, als auch im Bereich der Operations Research weit verbreitet zum Einsatz (Matter und Mehl 1989; Banks et al 2014.; Košturiak und Gregor 1995). Bei der ereignisdiskreten Simulation werden die Simulationsläufe durch ein Ereignis, wie z.B. ein Kundenauf-

trag, zu einem diskreten Eintrittszeitpunkt des Ereignisses ausgelöst. Danach lassen sich die Zustandsvariablen anhand der Ereignisprozedur und der Statistikvariablen aktualisieren und die Folgeereignisse erzeugen. Anschließend wird die Notwendigkeit einer Wiederholung geprüft. Ist die erwartete Zielsetzung erreicht, wird der Bericht erstellt, ansonsten beginnt eine neue Wiederholung (Rose und März 2011). Das Ablaufschema nach Rose und März (2011) wird in Abbildung 2-6 dargestellt.

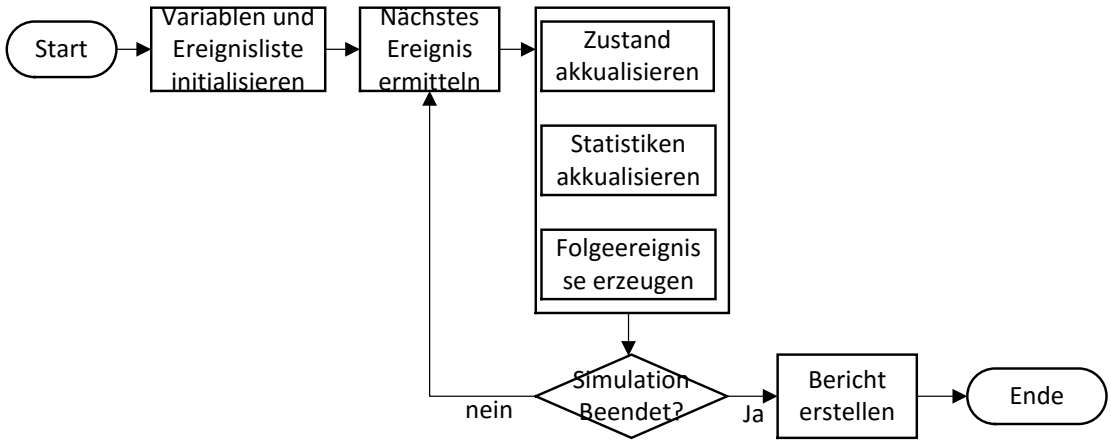


Abbildung 2-6: Ablaufschema einer ereignisdiskreten Simulation nach Rose und März (Rose und März 2011)

Die Anwendungsfelder der Simulationstechnik im Bereich der Produktion und der Logistiksysteme beschränken sich nicht mehr auf die Planungsphase. Die Simulation wird auch als Hilfsmittel zur Entscheidungsfindung in ganzen Lebenszyklus von Produktions- und Logistiksystemen, welcher aus Planungsphase, Realisierungsphase und Betriebsphase bestehen, angewandt (Rabe und Hellingrath, 2001; VDI-Richtlinie 3633 Blatt 1). Um die geeignetste Lösung zu finden, werden die Experimente des Simulationsmodells durch die wiederholten Einstellungen der Parameter durchgeführt. Die Simulation selbst verfügt über keine Funktion zur Optimierung (VDI-Richtlinie 3633 Blatt 1). Daher wird die Simulationstechnik vor allem als ein ergänzendes, aber nicht ersetzendes Hilfsmittel für die Entscheidungsfindung im Bereich der Produktion und Logistik betrachtet (VDI-Richtlinie 3633 Blatt 1).

2.2.2 Vorgehensweise bei der Simulationsstudie

Um eine Simulationsstudie erfolgreich durchzuführen, muss zunächst die grundsätzliche Vorgehensweise systematisch dargestellt werden. Die Vorgehensweise einer Simulationsstudie nach Rabe (2008) gliedert sich in 7 Phasen, die Abbildung 2-8 illustriert und nachfolgend genannt werden: Aufgabendefinition, Systemanalyse, Modellformulierung, Experimente und Analyse sowie die in diesen Phasen parallel verlaufende Datenbeschaffung und Datenaufbereitung (Rabe 2008). Dementsprechend werden die Ergebnisse jeder Phase in den rechten Spalten dargestellt. Alle Ergebnisse müssen durch Verifikation und Validierung (V&V) überprüft werden (Rabe 2008). Die Phasenergebnisse bestehen aus Aufgabenspezifikation, Konzeptmodelle, formales Modell, ausführbares Modell, Simulationsergebnisse sowie Rohdaten und auf-

bereitete Daten. Um ein erwartete Simulationsergebnis zu erhalten, werden alle Phasen in der Simulationsstudie wiederholt durchgeführt (Rabe 2008; VDI-Richtlinie 3633 Blatt 1).

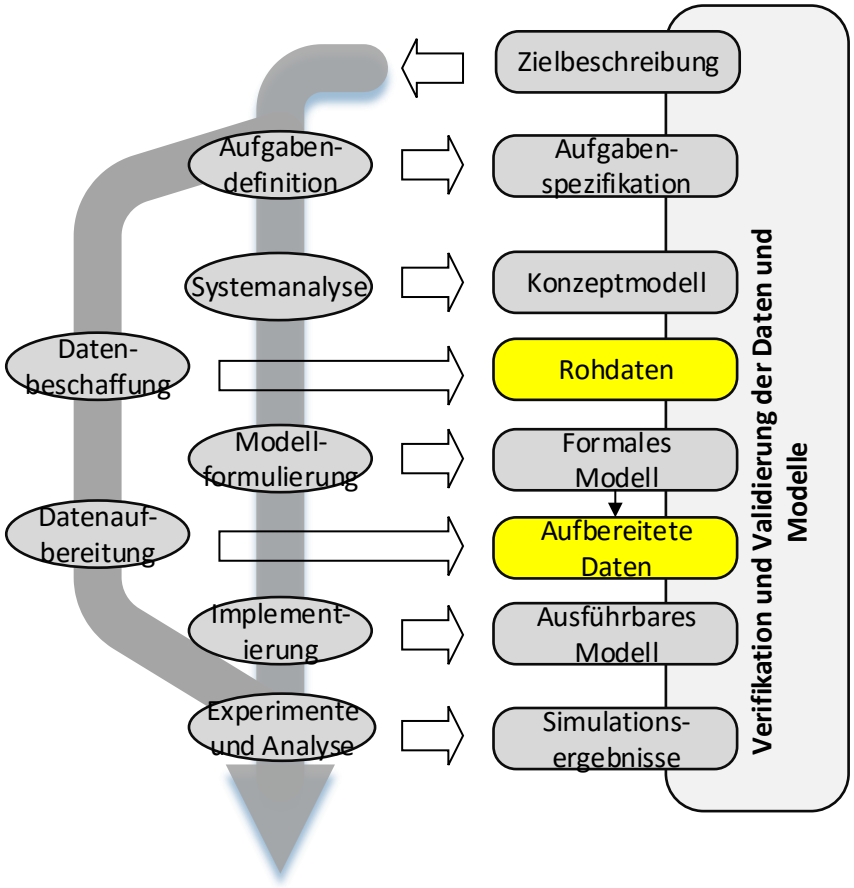


Abbildung 2-7: Vorgehensweise der Simulationsablauf nach Rabe (Rabe 2008; VDI-Richtlinie 3633 Blatt 1)

Eine wichtige Phase zur erfolgreichen Durchführung einer Simulation besteht in der Prüfung der Simulationswürdigkeit, welch Beginn der Simulationsstudie und nach der Problemformulierung ausgeführt werden sollte (Rabe et al. 2001). Daher sollte zuerst die Anwendbarkeit der Lösungstechniken in Abhängigkeit der Problemstellung untersucht werden. Da die Simulationsläufe im Rahmen der Optimierung oft sehr lange dauern, sollte in der Simulationsstudie mit Blick auf das Kosten-Nutzen-Verhältnis nicht nur der monetäre, sondern auch der zeitliche Aufwand überprüft werden. Außerdem sollten bei der Prüfung der Simulationswürdigkeit die folgende Aspekte beachtet werden: die Komplexität der Aufgabe, die Unsicherheit bezüglich der Daten, die Datenqualität in Abhängigkeit der erwarteten Ergebnisqualität sowie die Wiederverwendbarkeit des Simulationsmodells (VDI-Richtlinie 3633 Blatt 1). Insgesamt spielt die Prüfung der Simulationswürdigkeit immer eine wichtigere Rolle für die Durchführung der simulationsbasierten Optimierung.

Zur Aufgabendefinition und Zielformulierung zählt die Durchführung einer Simulationsstudie. In dieser Phase werden nicht nur die konkreten Simulationsziele oder der Umfang der Studie festgelegt, sondern auch die Randbedingungen wie bespielwiese die Durchlaufzeit definiert

(VDI-Richtlinie 3633 Blatt 1). Danach wird das System in der Phase der Systemanalyse untersucht. Dadurch wird ein nicht-gedanklich konzipiertes Modell entwickelt (VDI-Richtlinie 3633 Blatt 1; Rabe et al. 2001). Die Komplexität des Systems wird hinsichtlich der Untersuchungsziele und der relevanten Merkmalen mit Hilfe der Systemanalyse in seine Elemente und Grundstrukturen zerlegt. In dieser Phase werden für die zu entwickelnden Simulationsmodelle benötigten Eingaben, Ausgaben, Elemente, Zielsetzungen, Annahmen Vereinfachungen sowie auch die Aufbau- und Ablaufstruktur definiert. Schließlich wird ein Konzeptmodell mit Hilfe der Elemente und dieser grundsätzlichen Struktur aufgebaut (VDI-Richtlinie 3633 Blatt 1; Rabe 2008). Das Konzeptmodell, das die Lösung der Probleme beschreibt, wird als Grundlage für die formalen und ausführbaren Modelle herangezogen. Um das Systemmodell aufzubauen, sollte an dieser Stelle die Grundsätze ordnungsmäßiger Modellierung erläutert werden. Diese Grundsätze leiten sich aus den Kriterien für die Prozessmodellierung ab. Diese lauten Richtigkeit, Relevanz, Klarheit, Vergleichbarkeit und systematischer Aufbau (Becker et al. 2012; Wenzel et al. 2008).

In der Phase der Modellformulierung und Implementierung werden jeweils die formalen und ausführbaren Modelle auf Basis der Konzeptmodelle gebildet. In der Modellformulierung werden einerseits die formalen Entwürfe mit Hilfe der Elemente und der Konzeptmodellstruktur entwickelt, anderseits werden die Modelle auf ihre grundsätzliche Richtigkeit überprüft und validiert (Rabe 2008). Danach entstehen erst die ausführbaren oder experimentierbaren Modelle, wie Simulationsmodelle und Computermodelle im Rahmen der Implementierung. In dieser Phase werden die formalen Modelle in Abhängigkeit der ausgewählten Simulationswerkzeuge effizient implementiert. in der folgenden Phase wird das Experiment und die Analyse der Simulationsstudie genauer erarbeitet (Rabe 2008). Die Phase „Datenbeschaffung und Datenaufbereitung“ wird im nächsten Abschnitt detailliert vorgestellt.

Im Rahmen des Schritts „Experimente und Analyse“ werden die ausführbaren Modelle mit den aufbereiteten Daten ausgeführt. Dazu sind die geplanten Simulationsexperimente von Bedeutung, wobei die variierenden Parameterwerte und die Reihenfolge definiert und systematisiert werden, um die Simulationsziele mit möglichst wenigen Simulationsläufen zu erreichen (VDI-Richtlinie 3633 Blatt 1). Die Simulationsexperimente dauern oft sehr lange, weil im Rahmen der Optimierung viele Simulationsläufe durchgeführt werden, um die optimale Lösung zu finden. Um die Sicherheit der Simulation zu gewährleisten, muss ein Simulationslauf wiederholt durchgeführt werden, wozu die Parameter für die Experimente gleich aber die Startwerte für die verwendeten Zufallszahlen unterschiedlich sein müssen (Rabe 2008; Wenzel et al. 2008). Bei der Analyse der Experimentergebnisse ist insbesondere die Erfassung und Bewertung der Ausgabedaten von Bedeutung. Die quantitative Analyse und Bewertung der Ergebnisdaten und der Parameter können Hinweise für die Optimierung des Systems liefern (Rabe 2008).

Die Verifikation und Validierung der Daten und Modelle verlaufen synchron zu allen Phasen der Simulationsstudie, wodurch die Richtigkeit und die Eignung der Phasenergebnisse überprüft werden (VDI-Richtlinie 3633 Blatt 1). Während die Übereinstimmung des Verhaltens des

Modells mit dem abgebildeten System bei der Validierung geprüft wird, wird die Korrektheit des Modells anhand vorbestimmter Anforderungen überprüft (Rabe 2008; VDI-Richtlinie 3633 Blatt 1). Zusätzlich wird ein Test als ein Mittel zur Verifikation und Validierung betrachtet. Um die Gültigkeit eines Modells zu prüfen, sollten meistens mehrere Tests mit Hilfe der entsprechenden Technik durchgeführt werden (Rabe 2008; VDI-Richtlinie 3633 Blatt 1).

Die Eingabedaten sind in der Regel eine dominante Größe für die Performance der Simulationsstudie. Gleichzeitig spielen auch die Ausgangsergebnisse und -daten eine bedeutende Rolle sowohl für die Auswahl der Wiederholungsanzahl der Experimente, als auch für die Qualität der Simulation. Deshalb wird die Datenanalyse der Simulationsstudie im folgenden Kapital detailliert und systematisch erarbeitet.

2.2.3 Datenanalyse in Rahmen der Simulation

In der Simulationsstudie gliedern sich die Daten in Eingabe- und Ausgabedaten. Während die Eingabedaten für die Durchführung einer Simulation relevant sind, sind die Ausgabedaten für die Analyse der Simulationsergebnisse von essentieller Bedeutung. Daher werden zuerst die Phasen Datenbeschaffung und Datenaufbereitung von dem Ablauf der Simulationsstudie erläutert. Danach um die Qualität der Simulationsergebnisse zu bewerten und somit die Anzahl der Simulationsläufe zu reduzieren, werden die Ausgabedaten analysiert. Die Hauptaufgabe der Ausgabedatenanalyse von der Simulation ist sicherzustellen, dass genügend Ausgabedaten aus der Simulation erfasst werden, um die Performance der Simulation schätzen zu können (Robinson 2004). Die Zielsetzung der Datenanalyse im Rahmen der Simulationsstudie besteht in der Suche der optimalen Lösungen durch möglichst minimale Aufwände.

Nach der Vorgehensweise der Simulationsstudie gehört die Phase der Datenbeschaffung und Datenaufbereitung zur wichtigen Phase für Modellierung und Modellexperiment. In diesen Phasen werden jeweils die Rohdaten von dem betrachteten System sowie die für das ausführbare Modell und für die Experimente aufarbeiteten Daten erfasst und verarbeitet (Rabe 2008; VDI-Richtlinie 3633 Blatt 3). Die beiden Phasen werden häufig zeitlich parallel zu der Modellerstellung aber vor der Implementierung durchgeführt (VDI-Richtlinie 3633 Blatt 3).

Bei der Simulationsstudie gliedern sich Eingabedaten für die Simulation in Systemdaten, Organisationsdaten und technische Daten, wie in Tabelle 2-1 dargestellt (VDI-Richtlinie 3633 Blatt 1). In der Phase Datenbeschaffung werden die für die Simulationsstudie verwendeten Rohdaten erhoben. Aufgrund der Komplexität der Produktionssystems ist die Erhebung von Daten häufig aufwendig, und weiterhin wird die Datenbeschaffung nicht von den Simulationsfachleuten durchgeführt (Rabe 2008). Deswegen können diese Daten aufgrund mangelnder Konsistenz in der Regel nicht unmittelbar in der Simulationsstudie eingesetzt werden (VDI-Richtlinie 3633 Blatt 1). Deshalb werden die Rohdaten durch die Phase Datenaufbereitung, die vor allem von Fachleute durchgeführt wird, verarbeitet, damit die für das ausführbare Modell und die Phasen Experimente und Analyse angewandt werden können (Rabe 2008). In diese Phase können die für die Simulation relevanten Daten sowie Parameter durch entsprechenden Verfahren

wie Beispielsweise eine Filterung ausgewählt werden. Damit kann die Simulationsstudie erfolgreich durchgeführt werden (Rabe 2008). Weiterhin um die Qualität der Eingabedaten sicherzustellen, kommen häufig die statistischen Verfahren wie Anpassungstests mit Chi-Quadrat-Test für die Bewertung der Daten zum Einsatz (Wenzel et al. 2008).

Tabelle 2-1: Simulationsdaten nach VDI-Richtlinie 3633 Blatt 1

Art der Daten	Beschreibung
Systemlastdaten	**Auftragseinlastung** Produktions- und Transportaufträge, Menge, Termine
	Produktdaten Arbeitspläne/Stücklisten
Organisationsdaten	**Arbeitszeitorganisation** Pausenregelung, Schichtmodelle
	Ressourcenzuordnung Werker, Maschinen und Fördermittel
	Ablauforganisation Strategie, Restriktionen und Störfallmanagement
Technische Daten	**Fabrikstrukturdaten** Anlagetechnologie (Layout, Fertigungsmittel, Transportfunktionen, Verkehrswege, Flächen und Restriktionen)
	Fertigungsdaten Nutzungszeit, Leistungsdaten und Kapazität
	Materialflussdaten Topologie des Materialflusssystems, Fördermittel, Nutzungsart, Leistungsdaten und Kapazität
	Stördaten Funktionale Störungen, Verfügbarkeiten

Die Auswertung der Simulationsergebnissen spielt eine bedeutende Rolle sowohl für die Sicherstellung der Qualität der Simulationsstudie als auch für die Auswahl der Maßnahmen zur Verbesserung der Simulationsexperimente (VDI-Richtlinie 3633 Blatt 1). Um die erwarteten Simulationsergebnisse zu gewinnen, werden die Simulationsexperimente mit den Zufallsabhängigen Parametern wiederholt durchgeführt. Deswegen dauern die Simulationsläufe mit einer großen Anzahl der Simulationsläufe häufig sehr lange. Deshalb sind die Bestimmung der Länge der Simulationsdauer und die Anzahl der Simulationsläufe einer der wichtigen Forschungsschwerpunt bei der Simulationsstudie (Wenzel et al. 2008). Die Wiederholung einer Simulationsstudie erfolgt durch die Anwendung der spezifischen Zufallszahlenströme, die wiederum eine spezifische Reihe von Zufallsereignissen auslösen (Robinson 2004).

Um die Simulationsergebnisse zu analysieren und bewerten, soll zuerst die Klassifikation der Simulationsexperimente erläutert werden. Die Simulationsexperimente gliedern sich hinsichtlich des Zeithorizontes in terminierte (vorübergehende) und nichtterminierte (dauerhafte) Simulationen (Wenzel et al. 2008; Robinson 2004). Während die terminierten Simulationen unter einen natürlichen Endzeitpunkt mit festgelegter Länge des Durchlaufs durchgeführt werden, können die nichtterminierten Simulationen kontinuierlich oder über eine lange Zeit in Unabhängigkeit von Endzeitpunkt und Initialdaten durchgeführt werden (Robinson 2004; Wenzel et al. 2008). Die wiederholten Simulationsläufe in der nichtterminierten Simulationen werden bei Steady-state-Phase mit Hilfe der Batch-Methode oder Replicate/Delete Methode durchgeführt (Wenzel et al. 2008).

Die erfolgreiche Bestimmung der Länge und Anzahl der Simulationsläufe ist abhängig von der Analyse der Ausgabedaten. Hierbei werden die Methode wie graphische Methode und Konfidenzintervallmethode eingesetzt (Robinson 2004; Wenzel et al. 2008). Dazu gehört die graphische Methode zu einem einfachen und grafischen Ansatz, bei der die kumulierten Mittelwerte von der mittleren Zeit der Ausgabedaten aus einer Reihen von Wiederholungen dargestellt werden. Hierbei wird die Anzahl der erwarteten Wiederholungen durch den Punkt, an dem die Linie flach wird, festgelegt (Robinson 2004). Im Gegensatz dazu werden die Anzahl der Wiederholungen bei Konfidenzintervallmethode statistisch bestimmt. Die Formel der Konfidenzintervall wird dargestellt als $CI = \bar{X} \pm t_{n-1,a/2} \frac{S}{\sqrt{n}}$, wobei repräsentieren $\bar{X}$,S, n, $t_{n-1,a/2}$ jeweils den Mittelwert der Ausgabedaten der Wiederholungen, Standardabweichung der Outputdaten der Wiederholungen, Anzahl der Wiederholungen und Wert der Student- t-Verteilung mit $n-1$ Freiheitsgrad und einem Signifikanzniveau von $a/2$ (Robinson 2004). In der Praxis werden die Stichprobe und die Vertrauenswahrscheinlichkeit, z.B. 95%, für die Bestimmung der Anzahl der Wiederholungen vorher gegeben. Damit wird es bestimmt, mit welcher Wahrscheinlichkeit die statistischen Parameter sich in gegebenen Intervall befinden (Wenzel et al. 2008).

In der Simulationsstudie gewinnt die Optimierung der Simulationsparameter wegen der vernetzten Zusammenwirkung der Parameter immer mehr an Bedeutung. Die Parameter der Simulationsstudie gliedern sich in exogenen und endogenen Variablen (Košturiak und Gregor 1995). Während die endogenen Variablen die Beziehungen der Komponenten im Modell bezeichnen, stellen die exogenen Variablen die Beziehungen zwischen dem System und der Umfeld dar (Košturiak und Gregor 1995). In einer Simulationsstudie sind die Parameter häufig nicht gleichgewichtig für die Durchführung der Simulation, deswegen können die für die Qualität der Simulationsergebnisse irrelevante Parameter durch die Analyse und Bewertung der entsprechenden Ausgabedaten erkannt und sogar entfernt werden. Dadurch wird der Zeitaufwand für die Einstellung dieser Parameter verringert und somit die Anzahl und Länge der Simulationsstudie reduziert werden können. In der Simulationsstudie ist die Optimierung immer von Bedeutung, da die Simulation selbst über keine Optimierungsfunktionen verfügt (VDI-Richtlinie 3633 Blatt 1). Jedoch stellen die Ausgabedaten der Simulation in Produktion und

Logistik als stochastisch und diskrete dar, ist die Anwendbarkeit der klassischen mathematischen Methode vielmehr eingeschränkt. Zurzeit kommt vielmehr die simulationsbasierte Optimierung zur Untersuchung der Produktionssysteme zur Anwendung. Im Folgend werden die Grundbegriffe der simulationsbasierte Optimierung erläutert.

2.2.4 Kopplung der Simulation und Optimierung

Die Optimierung der Produktionssystemmodelle kann sowohl durch die mathematische Verfahren als auch simulationsbasierte Methode ausgeführt werden. Während mit Hilfe des mathematischen Verfahrens das globale Optimum für die Zielfunktion unter die Nebenbedingungen gefunden wird, dient die simulationsbasierte Optimierung zur Suche nach einer Kompromisslösung in Abhängigkeit von heuristischen und diskreten Methoden (Krug und Rose, 2011). Bei der simulationsbasierte Optimierung werden die Optimierungsverfahren mit den erstellten Simulationsmodellen kombiniert (Hong und Nelson 2009). Das Problem wird im Folgenden beschreibt als:

$$min\, g(x)\,, x \in \Theta,$$

wobei $g(x) = E[Y, \xi]$. Hier ist $g(x)$ das einzige Objektiv, das als die erwarteten Werte von einer Zufallsvariable, dazu stellt ξ die Zufälligkeit dar, beispielsweise die Zufallszahl in der Simulation. Die Verteilung der $Y(x, \xi)$ ist eine unbekannte Funktion des Vektors von den Entscheidungsvariablen x, jedoch ist $Y(x, \xi)$ durch die Simulationsexperimente realisierbar (Hong et al. 2015). Die diskreten und heuristischen Optimierungsansätze bei der Simulation und Optimierung in Produktion und Logistik gliedern sich in deterministische, stochastische, evolutionäre und genetische Verfahren sowie Schwellwert- und Permutationsverfahren. Um die beste Optimierungsverfahren hinsichtlich einer oder mehreren Zielfunktionen auszuwählen, kommt eine Lernstrategie oder –Prozess zu Einsatz (Krug und Rose, 2011).

Die Simulationsbasiere Optimierung basiert auf die erstellten Simulationsmodelle, die im Bereich der Produktion und Logistik als stochastisch ereignisorientierte Modelle betrachtet werden. Die Elemente in der Simulationsmodelle stellt in der Regel das statistische Verhalten dar. Weiterhin führt dieses Verhalten mit der vernetzten Wechselwirkungen zu stochastischer Schwankungen für den gesamten Prozess und somit für Simulationsergebnisse (VDI-Richtlinie 3633 Blatt 1). Deshalb ist die Optimierung von Simulationsmodellen ein schwieriges Problem. Das führt zur einen hohen Aufwand in der Simulation und fehlenden Strukturinformationen für die Optimierung. Sodass die Simulationsmodelle, wie die Abbildung 2-9 dargestellt, als eine „Black Box“ betrachtet werden. Dafür sind die klassischen mathematisch-analytischen Methoden für die Analyse und Optimierung der Simulationsmodelle nicht mehr geeignet. Daher kommen vielmehr die anderen zur Optimierung eingesetzten Methoden wie Heuristik und Metaheuristik zum Einsatz (April et al. 2003).

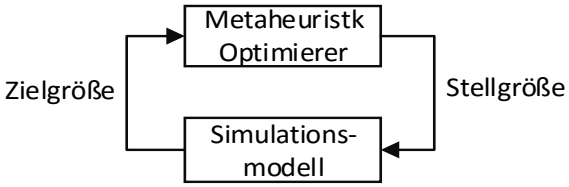


Abbildung 2-8: Black Box Ansatz zur Simulationsbasierten Simulation (April et al. 2003)

Die Bestimmung von Stellgrößen und Zielgrößen ist für die simulationsbasierte Optimierung von zentraler Bedeutung. Für eine Simulationsstudie stehen in der Regel viele Stellgrößen zur Verfügung (Weiger und Rose 2011). Bei der Optimierung der Simulationsergebnisse werden nicht alle Stellgrößen eingesetzt, da nicht alle Stellgrößen die gleiche Rolle für die Optimierung der Simulation spielen. Durch die unnötige Stellgröße erhöht sich nur der Rechenaufwand (Weiger und Rose 2011). Um die Effizienz der Simulationseffizienz zu steigern und die Anzahl oder Länge der Simulationsläufe zu reduzieren, werden nur die für die Optimierungsverfahren relevanten Stellgrößen systematisch analysiert und ausgewählt (Weiger und Rose 2011).

Im Rahmen der Simulationsbasierten Optimierung stellt April et al. (2003) eine Methode dar, die basierend auf eine Metamodell ist. Das Metamodell wird als Filter im der simulationsbasierte Optimierung integriert und somit die von den Simulationszielen nicht erwarteten Lösungen, Zielgrößen entfernt werden können. Wie die Abbildung 2-9 dargestellt, wird die von Metaheuristik erzeugte mögliche Lösungen durch das Metamodell überprüft, danach werden die erwartete Lösungen zurück zu dem Simulationsmodell als neue Inputparameter verwendet. Zugleich werden die schlechte Lösungen durch den Filter herausfiltert und nicht in dem Simulationsmodell freigegeben (April et al. 2003). Um eine bessere oder sogar optimale Lösung zu finden werden die Optimierungsprozesse mit Hilfe eines Filters vielmehr wiederholt ausgeführt (April et al. 2003)

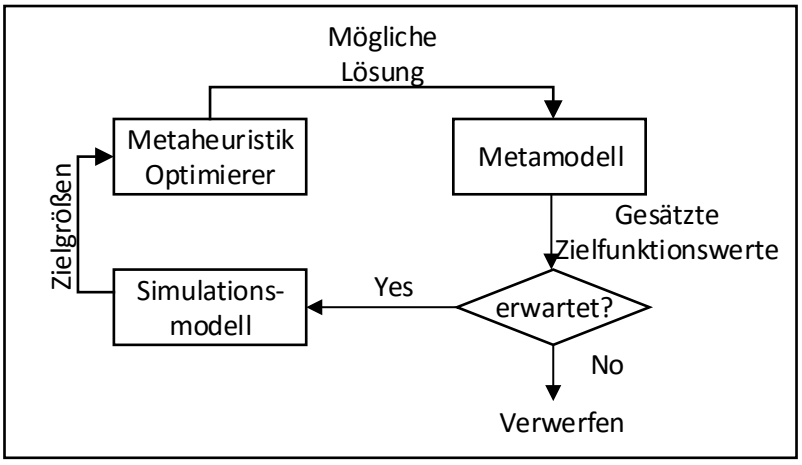
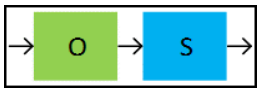
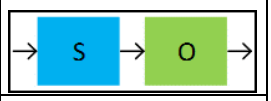
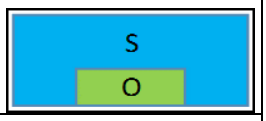
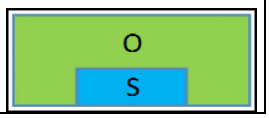


Abbildung 2-9: Metaheuristik Optimierung mit einem Metamodell in Anlehnung von April (April et al. 2003)

Die grundsätzliche Kopplung von Simulation und Optimierung lassen sich anhand der gegenseitige Anhängigkeit der Prozeduren in sequentielle und hierarchische Kopplung klassifiziert (Zis-

gen und Hanschke 2015). Während das Simulationsexperiment und die Optimierung bei der sequentiellen Kopplung als zwei Phasen, die eigenständig und getrennt voneinander ablaufend sind, betrachtet werden, lassen sich die Simulation und Optimierung bei der hierarchischen Kopplung aufeinander interpretieren (Zisgen und Hanschke 2015; März und Krug 2011).

Tabelle 2-2: Kopplungsarten der Simulation und Optimierung nach Krug (März und Krug 2011)

Kopplungsarten		Anwendungsfelder	Bezeichnung
Sequenzielle Kopplung	Simulation folgt der Optimierung	Bewertung und Überprüfung der Machbarkeit der vorgeschlagenen Lösung	
	Optimierung folgt der Simulation	Simulation zur Breitstellung der Startwerte für die Optimierung	
Hierarchische Kopplung	Integration der Optimierung in Simulation	Optimierung anhand des aktuelle Status der Simulation	
	Integration der Simulation in Optimierung	Simulation als Ersatz für die funktionale Bewertung der Zielfunktion in der Optimierung	

Die sequentielle Kopplung unterscheidet sich anhand der Startwerte, d.h. die Simulationsergebnisse als Startwert der Optimierung und umgekehrt die Optimierungsergebnisse zur Konfiguration der Simulation. Dementsprechend klassifiziert sich die hierarchische Kopplung in Integration der Optimierung in die Simulation und Simulation als Bewertungsfunktion der Optimierung (März und Krug 2011). Die vier grundsätzlichen Kopplungsarten von Simulation und Optimierung werden in Tabelle 2-2 detailliert und systematisch dargestellt, davon werden auch die Anwendungsfelder und die Bezeichnungen jeder Kopplungsart beschrieben. Während „Simulation folgt der Optimierung“ zur Bewertung und Überprüfung der Machbarkeit der vorgeschlagenen Lösung zum Einsatz kommt, lässt sich „Optimierung folgt der Simulation“ zur Breitstellung der Startwerte für die Optimierung anwenden (März und Krug 2011). Weiterhin wird die Optimierung im Rahmen der „Integration der Optimierung in Simulation“ anhand des aktuellen Status durchgeführt. Bei der „Integration der Simulation in Optimierung“ wird die Simulation als den Ersatz für die funktionale Bewertung der Zielfunktion in der Optimierung (März und Krug 2011). In der rechten Spalten der Tabelle werden die Bezeichnungen von jeden Kopplungsarten illustriert (März und Krug 2011). Damit kann die Beziehung der Simulation und Optimierung visualisiert bewertet werden. Mit der Kopplung der Simulation und Optimierung werden die jeweiligen Vorteile kombiniert. Dabei werden die Simulation in die Optimierungsverfahren die besten Parameter ausgewählt, damit die besseren oder optimalen Ziele erreichen werden (März und Krug 2011).

Um die simulationsbasierte Optimierung in der Praxis erfolgreich durchzuführen, sollten die ausgewählten Optimierungsmethoden den folgenden Anforderungen nach Fu (2002) erfüllen (Fu 2002):

- Allgemeingültigkeit (Generality): Die Optimierungsansätze mussten in der Lage sein, die verschiedenen Probleme zu behandeln
- Transparenz (Transparency): Bei der Anwendung der Optimierungsansätze sollte das mathematische Hemmnis für die Anwender vermieden werden
- Hohe Dimensionalität (High demensionality): Die implementierten Algorithmen sollten die Probleme mit den hohen Dimensionen effiziert lösen können
- Effizienz (Efficiency): Die Integration der Optimierung in der Simulationsstudie sollte in der Lage, die Rechenressourcen effizienter einzusetzen und zugleich die guten und einfachen Ergebnisse für die komplexen Probleme zu finden.

Hierbei ist es ersichtlich, dass nicht nur die Anwendbarkeit sondern auch die Effizienz der ausgewählten Algorithmen im Rahmen der simulationsbasierten Optimierung von wichtiger Bedeutung ist. Um die Effizienz der Simulation zu optimieren, sollten die unnötigen Simulationsläufe durch die Untersuchung der Simulationsdaten vermieden werden (Fu 2002). Mit Hilfe der Optimierungsverfahren besteht die Möglichkeit, die wichtigen Parameter von Simulationsmodellen zu erkennen.

In Rahmen der Simulation der Produktionssysteme steigert sich die Qualität der Entscheidungen durch die wiederholte Durchführung der Simulationsexperimente. Zugleich verursacht jedoch die zunehmende Anzahl der Simulationsläufe, die auf die Einstellung der vielmehr Parameter zurückzuführen sind, sowohl eine hohe Simulationsaufwand als auch eine Verzögerung der Entscheidungsfindung (Rabe et al. 2001). Eine mögliche Zielsetzung der Arbeit besteht darin, die relevanten Simulationsparameter mit Hilfe der Entscheidungsbäume zu erkennen und sich die Werte zielgerichtet zu verändern. Hierbei werden im folgenden Abschnitt die Grundbegriffe der Entscheidungsbäume erarbeitet.

2.3 Grundlagen der Entscheidungsbäume

In diesem Kapitel werden die notwendigen grundlegenden Begriffe, die für die Generierung und Auswahl der Entscheidungsbäume notwendig sind, erläutert. Dabei werden zunächst die Grundlagen der Datenvorbereitung und Datenvorverarbeitung für das Data Mining erarbeitet. Zudem werden die grundlegenden Begriffe zur Konstruktion der Entscheidungsbäume vorgestellt. Danach werden sowohl die Typen der gängigen Entscheidungsbäume als auch die Anforderungen an Entscheidungsbäume erarbeitet, damit ein geeigneter Entscheidungsbaum ausgewählt werden kann.

2.3.1 Datenvorbereitung und -vorverarbeitung

Die Datenvorbereitung gewinnt im Bereich des Data Mining an entscheidender Bedeutung, da die Qualität der Daten einen bedeutenden Einfluss auf sowohl die Prozesse des Data Mining als auch die Qualität der Ergebnisse hat. Daher befasst sich die Datenvorbereitung, insbesondere die Datenvorverarbeitung, mit der Verbesserung der Qualität er Daten, damit die ange-

gebenen Daten erfolgreich analysiert werden können (Lämmel und Cleve 2014). Weiterhin sollte die Datenvorverarbeitung in Unabhängigkeit der Generierung der Entscheidungsbäume durchgeführt werden, damit das Verfahren der Baumkonstruktion nicht beeinträchtigt wird. Darüber hinaus sollten die Daten vor der Vorverarbeitung mit statistischen Methode untersucht werden, um einen ersten Einblick dieser Daten zu erhalten. Damit die signifikanten Merkmale der Daten analysiert werden können. Dabei werden die Werte wie Median, Standardabweichung, Maximum und Minimum bei den numerischen Attributen angewandt. Weiterhin kommen in der Regel die Angaben wie mögliche Werte und Häufigkeit zur Analyse der nominalen Daten zum Einsatz (Lämmel und Cleve 2014).

Generell gliedert sich die Datenvorbereitung und Datenvorverarbeitung in 5 Phasen, wie Datenselektion und -integration, Datensäuberung, Datenreduktion und Datentransformation (Lämmel und Cleve 2014). Die Tabelle 2-3 stellt eine kurze Zusammenfassung der Phasen der Datenvorbereitung und -vorverarbeitung dar.

Tabelle 2-3: Arten der Datenvorbereitung und -vorverarbeitung (Lämmel und Cleve 2014).

Arten	Erklärung
Datenselektion und -Datenintegration	Auswahl der erforderlichen Daten anhand der Anforderungen und Zusammenfügung der ausgewählten Daten
Datenbereinigung	Bereinigung der vorliegenden Daten
Datenreduktion	Reduktion der Daten, z.B. bezüglich der Dimension
Datentransformation	Umwandlung der Daten zu adäquaten Darstellungsformen

Im Folgenden werden die klassische Probleme sowie die Vorgehensweise der Datenvorbereitung und der Datenvorverarbeitung detailliert erarbeitet, damit die Entscheidungs-bäume erfolgreich entwickelt werden können.

Datenselektion und -integration

In der Phase der Datenselektion werden die für die Untersuchung der Qualität der Ergebnisse erfordlichen Daten ausgewählt. Danach werden die ausgewählten Daten, die aus unterschiedlichen Quellen, wie bespielweise unterschiedlichen Tabelle, stammen, in der Phase der Datenintegration zu einer Datentabelle zusammengefügt (Lämmel und Cleve 2014). Mithilfe der Datenintegration werden die Redundanz und die Widersprüche der integrierten Datenmenge reduziert und somit kann die Genauigkeit und die Geschwindigkeit des Daten Mining erhöht werden (Han und Kamber 2006; Lämmel und Cleve 2014). Weiterhin treten die folgenden Probleme bei der Datenintegration auf, wie Entitätenidentifikationsproblem, Redundanz und Inkonsistenz und Datenwertkonflikte. Die Lösungen dieser Probleme werden in folgenden Abschnitte erläutert (Han und Kamber 2006; García et al. 2015). Für die detaillierte Erklärung

kann man auch die Literaturen von Han und Kamber (2006) und García et al. (2015) verweisen (Han und Kamber 2006; García et al. 2015).

Datenbereinigung (Data Cleaning)

In der Phase der Datenbereinigung geht es darum, die fehlenden, verrauschten, falschen und inkonsistenten Daten sowie Ausreißer zu erkennen und zu behandeln. Datenbereinigung wird in der Regel als den ersten Schritt der Datenvorverarbeitung betrachtet. Die vorliegenden Probleme und die entsprechenden Lösungsansätze werden in Tabelle 2-4 zusammengefasst (Han und Kamber 2006; Witten und Frank 2001; Lämmel und Cleve 2014).

Tabelle 2-4: Lösungsansätze der Datenbereinigung

Probleme	Lösungsansätze
Fehlenden Daten	• Ignorieren der Attribute mit fehlenden Werte • Manuelle Einfügung der fehlenden Daten und Werte • Ersetzung durch eine globale Konstante oder einen wahrscheinlichsten, häufigsten, durchschnittlichen Wert • Ersetzungswerte aus der Relationsanalyse der Attribute
Rauschen und Ausreißer	• Ersetzung der verrauschten Daten durch sowohl die Mittel- oder Grenzwerte mittels Binning als auch die berechneten Funktionswerte mittels Regression • Erkennung der Ausreißer durch Clustering und somit Eliminierung der Ausreißer mittels Glättungstechnik
Inkonsistente und falsche Daten	• Löschen der falschen Daten oder sogar auch deren Attribute • Suchen nach einem basierend auf die nicht fehlerbehafteten Datensätze plausiblen Wert

Nach der Datenbereinigung sind die Datensätze für die Analyse noch sehr umfangreich. Daher können die Data Mining Prozesse, wie Klassifikation, mit großen Datenmengen aufwendig oder sogar unmöglich durchgeführt werden. Aus diesem Grund sollen die vorliegenden Datensätze weiter verarbeitet werden (Lämmel und Cleve 2014).

Datenreduktion

Mit Hilfe der Datenreduktion werden die Datenmenge und die Komplexität der Datensätze ohne Zerstörung der Vollständigkeit der Datensätze reduziert (Han und Kamber 2006). Im Rahmen der Datenreduktion werden sowohl die Verringerung der Komplexität als auch die Auswahl einer geeigneten Teilmenge der Daten durchgeführt. Dabei gliedern sich die Strategien der Datenreduktion vor allem in dimensionale Reduktion, numerischen Datenreduktion und Datenkompression. Die Tabelle 2-5 fasst die wesentlichen Strategien und die Beschreibun-

gen sowie die Methode der Datenreduktion zusammen (Han und Kamber 2006; Witten und Frank 2001; Lämmel und Cleve 2014).

Tabelle 2-5: Strategie der Datenreduktion

Strategie	Beschreibung	Methode
Dimensions-reduktion	Verringerung der Datenmenge durch Elimination der irrelevanten Daten oder Attribute	DWT, PCA, Auswahl Attributteilmenge …
Numerische Datenreduktion	Auswahl einer geeigneten Stichprobe wie Teilmenge der gegebenen Datensätze	Regression, Windowing …
Daten-kompression	Reduktion der Datenmenge und der Komplexität durch Codierung oder Transformation der Daten	Aggression …

Für die Generierung eines Entscheidungsbaums mit ID3 oder C4.5 ist die Methode, wie bspw. Windowing, zur Durchführung der numerischen Datenreduktion von Bedeutung. Beim Windowing wird zunächst der Trainingsdatensatz nach Zufallsprinzip ausgewählt, anschließend wird die Generierung der Entscheidungsbaums durch das entsprechende Verfahren durchgeführt (Lämmel und Cleve 2014).

Wesentlich muss man darauf beachten, dass der Zeitaufwand der Datenreduktion nicht mehr als den bei der Datenanalyse mit den reduzierten Datenmenge eingesparten Zeitaufwand sein muss (Han und Kamber 2006). Bisher kann die Datenvorverarbeitung mithilfe der entsprechenden Methode in Unabhängigkeit der angewandten Verfahren der Data Mining durchgeführt werden. Im Folgenden muss die Datenvorverarbeitung anhand der eingesetzten Methode der Data Mining durchgeführt.

Datentransformation

Die Phase der Datentransformation handelt sich darum, dass das für den Data Mining nicht geeignete Datenformat in das erfordliche Datenformat umgewandelt wird. In dieser Phase können vielen Techniken wie z.B. Smoothing, Clustering, Standardisierung und Diskretisierung für die Datenverarbeitung angewandt werden (Han und Kamber 2006). Hierbei gewinnt die Diskretisierung der numerischen Attribute zur Verbesserung der Genauigkeit und der Effizient der Datenanalyse an wesentlicher Bedeutung. Insbesondere können einige Algorithmen zur Generierung der Entscheidungsbäume, wie bspw. ID3, nur mit nominalen Attributen durchgeführt (Witten und Frank 2001). Im Rahmen der Diskretisierung können die Attribute und Daten anhand des Intervalls und des Anwendungsfelder hierarchisch dargestellt werden (Han und Kamber 2006).

Wenn die Diskretisierung vor der Ausführung der Lernverfahren durchgeführt werden muss, können die numerischen Daten sowohl als nominale Variablen als auch als binären Attribute dekretiert mittels der Datentransformation behandelt werden (Witten und Frank 2001). D.h.

wenn das Lernverfahren die geordneten numerischen Daten behandeln kann, wird jedes Diskretisierungsintervall durch einen nominalen Wert dargestellt. Im Gegensatz dazu können die Attribute anhand der Anwendungsfelder in die binären Attribute umgewandelt werden. Bespielweise, die Zielvariable zur Generierung der Entscheidungsbäume können häufig mit den binären Attributen wie false und true oder ja und nein repräsentiert werden (Witten und Frank 2001). Weiterhin werden viele Verfahren zur Diskretisierung der numerischen Attribute in Bezug auf den Anwendungsfelder und die Anforderungen eingesetzt, wie (un)überwachte Diskretisierung, Entropie-basierte Diskretisierung, Fehler-basierte Diskretisierung (Witten und Frank 2001). Dabei kommen in der Regel die Entropie-basierte, 2-basierte Diskretisierung und die Binning Methode zum Einsatz (Han und Kamber 2006). Das Verfahren der Diskretisierung kann vereinfacht im Folgenden beschrieben werden: Vor der Diskretisierung sollen die gegebenen Beispieldaten nach Reihenfolge sortiert werden, anschließend wird der Punkt mit dem maximalem Informationsgewinn als den Aufteilungspunkt ausgewählt. Die Aufteilungsprozesse werden jeweils in oberen und unteren Teilen rekursiv fortgesetzt (Witten und Frank 2001).

Nach der Datenvorverarbeitung sollen die gegebenen Daten in die für die Generierung der Entscheidungsbäume erforderlichen Daten verarbeitet werden. Die Datenvorverarbeitung soll die Anforderungen wie Effizienz, Transparenz und Anwendbarkeit sowie Vollständigkeit erfüllen. Diese Anforderungen werden in der Validierungsphase detailliert erläutert und anhand des Ablaufs der Datenvorverarbeitung überprüft.

2.3.2 Konstruktion der Entscheidungsbäume

Um die Massendaten aus den Produktionssystemen zu analysieren, stößt die klassische Statische Methode häufig an der Grenzen. Daher kommen vielen neuen Techniken zur Datenanalyse zum Einsatz, eine der bekanntesten Methode ist Data Mining (Lämmel und Cleve 2014). Mit Hilfe des Data Mining können das Verhalten und die Entwicklung der Produktionssysteme untersucht und vorhersagt werden. Der Prozess des Data Mining gliedern sich vor allem in 6 Phasen, wie Datenselektion, Datenvorverarbeitung, Datentransformation, Data Mining und Evaluation und Interpretation (Lämmel und Cleve 2014; Han und Kamber 2006). Um die Massendaten aus den Produktionssystemen zu behandeln, kommen viele Methoden im Bereich der Data Mining, wie beispielsweise Cluster, zur Anwendung (Deuse 1998). In dieser Arbeit wird die Methode Entscheidungsbäume zur Optimierung des Simulationsmodells der Produktionssysteme eingesetzt.

Entscheidungsbäume dienen zur Entscheidungsfindung durch eine baumartige Struktur, die aus einem Wurzelknoten, Kanten, interne Kanten und Blättern bestehen, wie die Abbildung 2-10 illustriert. Dabei kennzeichnet der Wurzelknoten den Test für ein Attribut, repräsentiert jede Kante das Ergebnis eines Tests und kennzeichnet jedes Blatt eine Klasse (Casjens 2013). Die Entscheidungsbäume sind wegen der baumartigen Struktur einfach zu verstehen und gut zu interpretieren, sodass diese sowohl für Regression als auch für Klassifikation verwendet

werden können (Han und Kamber 2006; Casjens 2013). Darüber hinaus werden die gesamten Trainingsdaten anhand der Regeln, die von der Bezeichnung der entsprechenden Algorithmen abhängig sind, rekursiv in einem Entscheidungsbaum partitioniert (Casjens 2013). Weiterhin kann ein Entscheidungsbaum als eine Menge von „*If-Then*" Regeln betrachtet werden (Runkler 2010; Han und Kamber 2006).

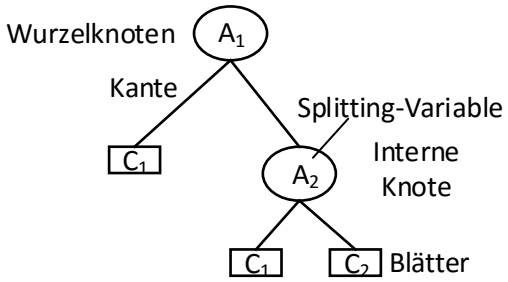


Abbildung 2-10: Schematische Darstellung eines Entscheidungsbaums in Anlehnung von Casjens (Casjens 2013)

Die Konstruktion der Entscheidungsbäume zählt zu dem überwachten Lernen, das von eine bestimmten Menge an Trainingsdaten abhängt (Han und Kamber 2006). Anschließend werden die gelernten Regeln durch die Validierungsdaten und Testdaten überprüft und danach auf Gesamtmenge der Daten angewandt (Krahl et al. 1998).

Um die Trainingsdaten zu separieren wird zuerst die Auswahl eines Attributes ausgeführt. Dazu werden im Folgen einige wichtige Auswahl- oder Splittingkriterien erläutert, wie Informationsgewinn (engl. Information Gain), Informationsgewinnverhältnis (engl. Gain Ratio) und Gini-Wichtigkeit (engl. Gini Index) usw. (Quinlan 1986; Quinlan 1993; Breiman 1984).

Information Gain und Gain Ratio basieren auf die Shannon-Entropie, und werden als das wichtige Auswahlkriterium jeweils für den ID3 und den C4.5/C5.0 Algorithmus angewandt (Lämmel und Cleve 2014; Quinlan 1986). Um den Informationsgewinn klar zu erläutern, werden zuerst die Grundbegriffe der Entropie erläutert. Im Bereich der Information Theorie repräsentiert Entropie die Unsicherheit der Variablen, und dieses Ergebnis ist die erwartete Anzahl an Bits (Li 2012). Wenn die Entropie der Wahrscheinlichkeitsverteilung der Daten abhängt, kann die Entropie hier 2 Arten unterscheiden: empirische und empirische bedingte Entropie (Li 2012).

Daher wird die empirische Entropie der Zufallsvariablen X in der Regel als $H(X) = -\sum_i^n p_i \log_2 p_i$ kennzeichnet, und zugleich wird die empirische bedingte Entropie $H(X|Y)$ als $H(Y|X) = \sum_i^n p_i H\ (Y|X = x_i)$ definiert (Li 2012). Davon wird die Verteilung der Zufallsvariablen X als $P(X = x_i) = p_i, i = 1,2, \dots, n$ und die bedingte Verteilung der Zufallsvariablen (X, Y) als $P(X = x_i, Y = y_i) = p_{ij}, i = 1,2, \dots, n; j = 1,2, \dots, m$ definiert (Li 2012).

Informationsgewinn (Information Gain)

Auf Basis der Theorie und Formeln der Entropie lässt sich der Algorithmus der Information Gain im Folgenden vorstellen. Hierbei werden die folgenden Begriffe vorher definiert, wie Trainingsdatensatz D, die Anzahl der Trainingsdaten $|D|$, Klasse der DatensatzC_k, die Anzahl

der Klasse $|C_k|$ und $\sum_{k=1}^{K}|C_k| = |D|$. Weiterhin verfügt das Attribut A über $\{a_1, a_2, \dots, a_n\}$. Dementsprechend unterteilte sich der Datensatz D in n Teilsätze $D_1, D_2, \dots, D_n$, $|D_i|$ ist die Anzahl der Teilmenge, zugleich repräsentiert D_{ik} als $D_{ik} = D_i \cap C_k$ und $|D_{ik}|$ ist die Anzahl des Datensatzes D_{ik} (Li 2012).

Algorithmus 1: Berechnung der Information Gain

Input: Trainingsdatensatz D und Attribute A;

Output: Information Gain $Gain(D, A)$ von Attribute A für den Trainingsdatensatz D

Methode:

Schritt 1, Messung der empirischen Entropie $H(D)$ vom Datensatz D

$$H(D) = -\sum_{k=1}^{k}\frac{|C_k|}{|D|}\log_2\frac{|C_k|}{|D|} \tag{2-1}$$

Schritt 2, Berechnung der empirischen bedingten Entropie $G(D|A)$

$$G(D|A) = -\sum_{i=1}^{n}\frac{|D_i|}{|D|}H(D_i) = -\sum_{i=1}^{n}\frac{|D_i|}{|D|}\sum_{k=1}^{k}\frac{|C_{ik}|}{|D_i|}\log_2\frac{|C_{ik}|}{|D_i|} \tag{2-2}$$

Schritt 3, Berechnung der Information Gain

$$Gain(D, A) = H(D) - G\,(D|A) \tag{2-3}$$

Mithilfe der Information Gain werden diese Attribute, die über die größte Informationsmenge bevorzugt verfügen, ausgewählt. Dies verursacht jedoch häufig die ungültigen Partitionen (Han und Kamber 2006; Witten und Frank 2001). Um die Hindernisse des Informationsgewinns zu vermeiden, wird das andere Kriterium wie Gain Ratio zur Baumgenerierung mit des C4.5 Algorithmus angewandt. Herbei werden die vorliegenden Formeln des Information Gain eingesetzt, um die Gain Ratio zu definieren.

Informationsgewinnverhältnis (Information Gain Ratio)

Information Gain Ratio von Attribute A in den Datensatz D definiert als

$$GR(D, A) = \frac{Gain\,(D,A)}{SplitInfo_A\,(D)} \tag{2-4}$$

Hierbei ist $GR\,(D, A)$ der Informationsgewinn und $SplitInfo_A\,(D)$ ist wie $H(D)$, die die Entropie des Trainingsdatensatzes (D) repräsentiert (Han und Kamber 2006).

Jedoch muss man darauf beachten, dass der Informationsgewinn der ausgewählten Attribute zumindest größer als den durchschnittlichen Gewinn der Testdaten ist, um die Unstabilität der Gain Ratio zu vermeiden (Han und Kamber 2006).

Gini Index

Beim Classification and Regression Tree (CART) Modell wird Gini Index angewandt (Han und Kamber 2006; Breiman 1984). Mit Hilfe der Gini index wird die Unreinigkeit des Trainingsdatensatzes oder die Partition der Daten gemessen, und Gini Index wird definiert als

$$Gini(D) = 1 - \sum_{i=1}^{n} {p_i}^2 \tag{2-5}$$

Herbei stellt die p_i die Wahrscheinlichkeit des Datensatzes zu den Klasse C_i in D dar, und p_i wird als $\left|C_{i,D}\right|/|C_D|$ definiert.

Aus einem binären Split der Beispielsmenge D in die Datenmenge D_1 und D_2 für das Attribut $A = \{a_1, a_2 \dots a_v\}$ ergibt sich der Gini Index:

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \tag{2-6}$$

Die Reduktion der Unreinigkeit des diskreten oder kontinuierlichen Attributes A wird definiert als:

$$\Delta Gini(A) = Gini(D) - Gini_A(D) \tag{2-7}$$

Das Attribut mit maximaler Reduktion der Unreinigkeit oder minimale Gini Index wird als das Splitattribut eingesetzt (Han und Kamber 2006).

Insgesamt werden in diesem Abschnitt die häufigsten angewandten Auswahlkriterien des Attributs vorgestellt, davon verhält sich das Gini Index ähnlich wie Information Gain. Bei anderen Algorithmen wie CHAID und QUEST kommt in der Regel die Statistischen Methode wie χ^2-Test und *F*-Test zur Attributauswahl zum Einsatz.

χ^2-Test

Außer der Anwendung der Informationstheorien basierte Kriterien und der Anwendung der Unreinigkeit basierte Methode als Splittingkriterien kommen häufig die statischen Methoden wie χ^2-*Test* und *F-Test* zur Baumgenerierung mittel der Algorithmen CHAID und QUSET zum Einsatz (Barros et al. 2015).

Der Person Chi-Square Test wird definiert als (IBM SPSS Modeler 14.2 Algorithm Guide):

$$\chi^2 = \sum_{k=1}^{K} \sum_{i=1}^{I} \frac{(n_{ij} - \hat{m}_{ik})^2}{\hat{m}_{ik}} \tag{2-8}$$

Hierbei bezeichnet der Wert $n_{ik} = \sum_n f_n I\ (x_n = i \vee y_n = k)$ als die beobachtete Häufigkeit und stellt der Wert $\hat{m}_{ik}$ die erwartete Häufigkeit für $(x_n = i\ y_n = k)$ aus einem Modell dar. Der *P*-Wert wird durch $p = \Pr\ (\chi_j^2 > \chi^2)$ berechnet, davon folgt χ_j^2 die Chi-Square Verteilung mit der Freiheit $h = \ (K - 1)\ (I - 1)$ (IBM SPSS Modeler 14.2 Algorithm Guide).

***F*-Test**

Beim QUEST Algorithmus wird *F*-Test zur Behandlung der kontinuierlichen Variablen eingesetzt (Barros et al. 2015). Der *F*-Test wird im Folgenden beschrieben, hierbei ist die C_k Klasse für die Knote *k* im Bereich der Zielattribute *Y* vorhanden, der *F* Test für die kontinuierlichen Attribute wird definiert als

$$F_X = \frac{\sum_{j=1}^{J_t} N_{f,c}(k)\left(\bar{x}^c(k) - \bar{x}(k)\right)^2 / \ (C_k - 1)}{\sum_{i \in t} f_i\ (x_i - \bar{x}^{(y_n)}(k))^2 / \ (N_f\ (k) - C_k)} \tag{2-9}$$

Hierbei repräsentieren $\bar{x}^c(k) = \frac{\sum_{i \in k} f_n x_n I(y_n = c)}{N_{f,c}(k)}$, $\bar{x}(k) = \frac{\sum_{i \in k} f_n x_n}{N_f(k)}$ jeweils die Mittel der Daten in und außer der Klasse c. Weiterhin sind C_k und $N_f\ (k)$ jeweils die Anzahl der Daten in Klasse c und die Gesamtanzahl der Daten. Weiterhin wird der *P* Wert als $pX = \Pr\left(F\left(C_k - 1, N_f(k) - C_k\right) > F_X\right)$ berechnet, hierbei folgt die $\left(C_k - 1, N_f(k) - C_k\right)$ eine *F* Verteilung mit dem Freiheitgrad $C_k - 1$ and $N_f(k) - C_k$ (IBM SPSS Modeler 14.2 Algorithm Guide).

Nach der Erörterung der Grundlagen der Splittingkriterien werden das Verfahren und die Methode zur Konstruktion der Entscheidungsbäume vorgestellt. Zuerst wird ein allgemeines Verfahren zur Generierung eines Entscheidungsbaums erläutert:

Schritt 1, Auswahl eines Attributs mittels Auswahlkriterien und Entfernung dieses Attributs aus der Menge an die Attribute;

Schritt 2, Unterteilung der Menge der Beispieldaten in Bezug auf die Klassifikation des ausgewählten Attributes in Teilmenge;

Schritt 3, Wiederholung der Schritte 1 und 2 für die anderen Attribute;

Die Generierung des Entscheidungsbaums stoppt, wenn alle Attribute aufgebraucht oder keine Trainingsdaten mehr oder die gegebenen Schwellenwerte ausgelöst sind.

Um die Beispieldatensätze besser zu untersuchen, werden die Algorithmen der gängigen Entscheidungsbäume, wie ID3, C4.5, C5.0, CART, CHAID und QUEST im Folgenden detailliert vorgestellt.

ID3 (Iterative Dichotomiser 3)

ID3 Algorithmus wurde von Quinlan entwickelt und basiert auf das Verfahren Top-Down Induction of Decision Trees (TDIDT) (Quinlan 1986). Beim ID3 Algorithmus werden die Attribute mit Hilfe des Informationsgewinn (Information Gain) auf Basis der Shannon-Entropie ausgewählt (Han und Kamber 2006; Lämmel und Cleve 2014; Runkler 2010). Anschließend werden alle Werte des gewählten Attributs rekursiv im Baum eingeordnet. Da der ID3 Algorithmus nur mit der diskreten Attribute generiert werden kann, müssen die Attribute mit den Kontinuierlichen Werten diskreditiert werden. Das Ziel der ID3 besteht darin, die Entscheidungsbäume so klein wie möglich zu halten (Li 2012). Nach Han (2006) und Li (2012) wird der Algorithmus ID3 im Folgenden dargestellt (Han und Kamber 2006; Li hang 2012).

Algorithmus 2: Algorithmus ID3 (D,A)

Input: Trainingsdatensatz D, Menge der Attribute A, Schwellenwert ε;

Output: Entscheidungsbaum T;

Methode:

(1) Erstellen eines Knotens N;

(2) ***IF*** alle Beispieldaten in D eine identische Klasse C haben oder die Anzahl der Daten kleiner als Schwellenwert ε ist ***THEN***

wird der einzige Knoten mit Klasse C zum Knoten N als Blatt;

(3) ***IF*** $A = \emptyset$, ***THEN***
Wird der einzige Knoten mit der häufigsten Klasse in D zum Konten N als Blatt zugeordnet;

(4) Berechnung des Information Gain von A in D und Bestimmung des besten Attributs A_g mit maximalem Informationsgewinn anhand $Gain(D, A) = H(D) - G\,(D|A)$;

(5) Bezeichnen den Konten N mit A_g;

(6) ***FOR*** alle Werte V_j von diesem A_g;

(7) Lässt die Bespielmenge des Datensatzes D_j aus dem Datensatz D mit den Werten V_j sein;

(8) ***IF*** $D_j = \emptyset$, ***THEN***
wird ein Blatt mit der häufigsten Klasse zum Knoten N angefügt;
Else

(9) ***Rekursion*** des Zweigs für den Wert $V_j = ID3\,(D_j, A)$;
Endfor

(10) ***Zurück*** zum N;

C4.5

Der C4.5 Algorithmus ist eine Weiterentwicklung des ID3 Algorithmus und sein Nachfolger ist der C5.0 Algorithmus. Im Vergleich mit ID3 kann der C4.5 Algorithmus mit nicht nur den diskreten sondern auch den kontinuierlichen Werten umgehen (Witten und Frank 2001). Dabei wird in der Regel die Entropiebasierte Diskretiesierung zur Verarbeitung der Kontinuierlichen Werte beim C4.5 eingesetzt (Witten und Frank 2001). Weiterhin kommt Gain Ratio anstatt Information Gain als Kriteriums der Attributauswahl zum Einsatz. Darüber hinaus werden beim C4.5 verschiedene Methoden des Pruning implementiert. Um die Hindernisse der Gain Ratio zu vermeiden, werden bei der Attributauswahl in der Regel zuerst die Attribute mit überdurchschnittlichen Informationsgewinn vorausgewählt, danach werden die Attribute mit größeren Gain Ratio ausgewählt (Quinlan 1993; Han und Kamber 2006; Li 2012). Nach Han (2006) und Li (2012) wird der Algorithmus C4.5 im Folgenden dargestellt (Han und Kamber 2006; Li hang 2012).

Algorithmus 3: Algorithmus C4.5 (D,A)

Input: Trainingsdatensatz D mit den Klassen C, Attribut A, Schwellenwert ε ;

Output: C4.5 Entscheidungsbaum T;

Methode:

(1) Erstellen eines Knotens N;

(2) ***IF*** alle Beispieldaten in D eine identische Klasse C haben oder die Anzahl der Daten kleiner als Schwellenwert ε ist ***THEN***

wird der einzige Knoten mit Klasse C zum Knoten N als Blatt zurück;

(3) ***IF*** $A = \emptyset$, ***THEN***
Wird der einzige Knoten mit der häufigsten Klasse in D zum Konten N als Blatt zugeordnet;

(4) ***IF*** die Attribute kontinuierlich sind, ***THEN***
werden die Werte mittels entropiebasierten Diskretisierung diskreditiert;

(5) Berechnung der Information Gain und Gain Ratio von A in D und Bestimmung des besten Attributs A_g mit überdurchschnittlichem Informationsgewinn $Gain(D,A)$ und maximaler Information Gain Ratio anhand $GR(D,A) = \frac{Gain\,(D,A)}{SplitInfo_A\,(D)}$;

(6) Bezeichnen den Konten N mit A_g;

(7) ***FOR*** alle Werte V_j von diesem A_g;

(8) Lässt die Bespielmenge des Datensatzes D_j aus dem Datensatz D mit den Werten V_j sein;

(9) ***IF*** $D_j = \emptyset$, ***THEN***
Fügen einen Blatt mit der häufigsten Klasse zum Knoten N an;
Else

(10) ***Rekursion*** der Zweige für Wert $V_j = C4.5\,(D_j, A)$;
ENDFOR

(11) ***Zurück*** zum N;

Während der Generierung der Entscheidungsbäume kann ein Schwellenwert zur Einschränkung der Anzahl der Instanzen in einem Blatt angegeben werden, womit die Baumgenerierung früher gestoppt und ein einfacher Baum entwickelt werden kann. Dadurch kann die Robustheit der Entscheidungsbäume sichergestellt werden (Liu et al. 2006).

Der C5.0 Algorithmus ist eine Weiterentwicklung von dem C4.5 und dem ID3. Die Beispieldaten kann mittels C5.0 Algorithmus in Abhängigkeit der Information Gain und Gain Ratio in zwei oder mehr Gruppen unterteilt werden, wozu kommt die Methode der Boosting beim C5.0 zum Einsatz (PANG und GONG 2009). Die Zielvariable des C5.0 Algorithmus sollten wie ID3 und C4.5 nominal und kategorial sein. Die detaillierten Vorstellungen des C5.0 sind wegen der kommerziellen Gründe nicht veröffentlicht (Polzin 2006). Im Vergleich zum C4.5 weist C5.0 nach dem Test von Witten (2011) einige Unterschiede auf, welche jedoch für die Verbesserungen nicht erheblich sind (Witten et al. 2011). Die Generierung der Regeln durch C5.0 zählt zu einer der wichtigeren Entwicklungen als die anderen Algorithmen (Witten et al. 2011).

CART (Classification and Regression Tree)

CART (Classification and Regression Tree) wurden von Breiman et.al (1984) entwickelt (Rokach und Maimon 2015; Breiman 1984). Beim CART kommt das Auswahlkriterium Gini Index zur Baumgenerierung zum Einsatz und der Cost Complexity Pruning Algorithmus wird zum Ab-

schneiden des Baums verwendet (Li 2012). Ein binärer Baum kann mittels CART generiert werden, d.h. jeder Knoten des Baums hat genau zwei Verzweigungen (Li 2012). Weiterhin kann der CART Algorithmus nicht nur Klassifikations- sondern auch Regressionsprobleme lösen (Rokach und Maimon 2015). In dieser Arbeit werden nur die Klassifikationsprobleme mittels der Entscheidungsbäume gelöst, wird hierbei vor allem der CART Algorithmus für die Klassifikation erläutert. Im Anschluss daran wird der CART Algorithmus schrittweise im Folgenden vorstellt. Wenn alle Attribute identisch oder die Anzahl der Beispieldaten oder die Gini Index weniger als den Schwellenwert seien, stoppt die Baumgenerierung mittels des CART Algorithmus (Li 2012). Nach Li (2012) wird der CART Algorithmus im Folgenden vereinfacht erläutertet (Li 2012).

Algorithmus 4: Algorithmus CART nach Li (2012)

(1) Input: Trainingsdatensatz D mit den Klassen C, Attribute A, Stoppkriterien;
(2) Output: CART Entscheidungsbaum T
(3) Methode:
(4) Berechnung des Gini Index der Attribute im Trainingsdatensatz D. Für alle Attribute A und deren Werte a, wird der Datensatz D in D_1 und D_2 in Abhängigkeit des Tests der $A = a$ (JA oder NEIN) aufgeteilt, dann wird der Gini Index von $A = a$ berechnet;
(5) Auswahl eines Attributs und seines Werts mit minimalem Gini Index anhand $Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$ jeweils als das optimale Attribut und den besten Aufteilungspunkt. Generierung von zwei Unterknoten anhand der optimalen Attribute und des besten Aufteilungspunkts und Verteilung der Trainingsdaten in zwei Knoten;
(6) Wiederholung der Schritte 1 und 2 in die zwei Unterkonten bis zum Stoppkriterium;
(7) Ende des Algorithmus und Generierung des Entscheidungsbaums.

CHAID (Chi-squared Automatic Interaction Detection)

Der CHAID (Chi-squared Automatic Interaction Detection) Algorithmus wurde von Kass (1980) vorgestellt und als eine automatische Erkennung der Interaktionen mittels Chi-Quadrat-Tests verstanden (Kass 1980). CHAID wird verbreitet im Bereiche der Marketing angewandt (Han und Kamber 2006). Der Entscheidungsbaum kann mittels des CHAID mit mehr als zwei Verzweigungen anhand der Auswahlkriterien des Attributes entwickelt werden (Ture et al. 2009). Beim CHAID werden die nominalen und geordneten Attribute und Werte anhand jeweils der Methode Signifikanzteste oder Chi-Quadrat-Test ausgewählt und aufgeteilt (Ture et al. 2009). Die Baumgenerierung wird gestoppt, wenn die Signifikanzniveau der Variablen beim χ^2- Test (wie 0.05) nicht mehr erreichen kann (Ture et al. 2009). Der Algorithmus von CHAID wird im Folgenden vereinfacht erläutert (Kass 1980).

Algorithmus 5: Algorithmus CHAID nach Kass (1980)

(1) zuerst wird eine Kreuztabelle in Abhängigkeit der Attribute und der Zielvariable erstellt, danach wird das beste Attribut in Schritt 2 und Schritt 3 ausgewählt;
  (2) hierbei werden die beiden Kategorien der unabhängigen Attribute gesucht, deren 2-d Kreuztabelle den geringsten Unterschied aufweisen. Wenn die Signifikanz den kritischen Wert (wie 0,05) nicht erreicht, werden die beiden Kategorien zusammengelegt und der Schritt 2 wiederholt;
  (3) es wird der signifikanteste binäre Split für die zusammengelegten Kategorien aus Schritt 2 gesucht. Wenn die Signifikanz größer als der kritischen Wert ist, dann wird der Split durchgeführt und zurück zum Schritt 2 gegangen;
(4) es wird die Signifikanz jedes zusammengelegten Attributs berechnet und die signifikantesten Attribute werden ausgewählt. Wenn die Signifikanz größer als den kritische Wert ist, dann werden die Daten anhand der zusammengelegten Kategorien mit dem ausgewählten Attribut dividiert;
(5) für jede Partition, die über nicht untersuchten Attributen und Daten verfügt, geht man zurück zum Schritt 1.

Ein Nachteil des CHAID Algorithmus besteht darin, dass kein Pruningverfahren nach der Konstruktion der Entscheidungsbäume angewandt werden kann (Polzin 2006).

QUEST (Quick, Unbiased, Efficient Statistical Tree)

Der QUEST Algorithmus (Quick, Unbiased, Efficient Statistical Tree) kann für die Entwicklung der binären Entscheidungsbäume mit sowohl univariaten als auch linear- kombinierten Attribute eingesetzt werden (Lämmel und Cleve 2014). Weiterhin erfordert QUEST die nominalen Zielattribute, damit die Beispieldaten klassifiziert werden können. Die Stoppkriterien von QUEST sind gleich wie die allgemeine Stoppkriterien von Entscheidungsbäume. Weiterhin wird die zehnfache Kreuzvalidierung zur Beschneidung der Bäume angewandt (Polzin 2006). Der vereinfachte Algorithmus QUEST wird nach Polzin (2006) im Folgenden vorgestellt (Polzin 2006):

Algorithmus 6: Algorithmus QUEST

(1) Auswahl eines Splittingattributs. Für jeden Split sollen die Attribute anhand eines Signifikanztests gemessen werden. Während die Überprüfung bei nominalen Attributen mittels χ^2-Test erfolgt, kommt hier in der Regel der *F-Test* zur Behandlung der numerischen oder kontinuierlichen Attribute zum Einsatz. Das Attribut, welches das gegebene Signifikanzniveau nicht überschreitet, wird als Splittingattribut ausgewählt;
(2) Unterteilung der Beispieldaten mit Hilfe des ausgewählten Attributs. Durch dieses Attribut wird der Knoten in zwei Unterknoten aufgeteilt, dadurch wird ein binärer Baum entwickelt.

```
Die Werte oder das Attribut mit der größten Menge der Zielva-
riablen werden als Splittingpunkt eingesetzt.
```

Ein Nachteil des QUSET Algorithmus besteht darin, dass der QUSET Algorithmus auf eine Annahme der Normalverteilung basiert (Polzin 2006). Wenn die Werte der Attribute nicht mit der Normalverteilung beschrieben werden können, verursacht dies häufig die Verzerrung der Ergebnisse (Polzin 2006).

Pruningverfahren

Mittels der Algorithmen können die Entscheidungsbäume rekursiv bis zum Ende generiert werden, womit die Trainingsdaten mit einer hohen Genauigkeit klassifiziert werden können. Jedoch weisen diese Entscheidungsbäume wegen der Überanpassung (engl. Overfitting) eine niedrige Genauigkeit zur Untersuchung der Testdaten auf (Li 2012). Aus diesem Grund soll das Problem der Überanpassung vermieden werden, womit die unnötigen Blätter und Knoten nicht generiert werden sollen. Dadurch können die entwickelten Entscheidungsbäume besser zur Untersuchung des Verhaltens des neuen Datensatzes angewandt werden (Lämmel und Cleve 2014; Petersohn 2005).

Das Pruning der Entscheidungsbäume gliedert sich in Prepruning und Postpruning (Han und Kamber 2006). Beim Prepruning wird der Abbruch der Baumentwicklung während der Baumbildungsverfahren durchgeführt, wobei der Unterbaum nach der Baumkonstruktion durch das Blatt mit maximale Häufigkeit innerhalb des Unterbaums ersetzt wird (Han und Kamber 2006; Witten und Frank 2001). Beim Prepruning kommen häufig die Abbruchkriterien wie statistische Signifikanz, Information Gain und Gini Index als ein Schwellwert zum Abbruch der Baumbildung zum Einsatz (Witten und Frank 2001). Wenn die Partitionierung des Datensatzes eines Knotens unter einen gewissen Schwellwert fällt, wird die Baumgenerierung gestoppt und der Knoten durch einen Blatt ersetzt. Allerdings ist die Bestimmung eines angemessenen Schwellwerts schwierig, da ein hoher Schwellwert zu einem zu vereinfachenden Baum führt, und zugleich ein niedriger Schwellwert hingegen eine hohe Verästelung verursacht (Han und Kamber 2006). Aus diesem Grund kommt das Postpruning vermehrt zum Einsatz. Hierbei werden einige Kriterien für das Postpruning im Folgend erläutert.

Ein Ansatz ist der Cost Complexity Pruning Algorithmus (Rokach und Maimon 2015). Dieser Algorithmus wird meistens beim Pruning der mittels CART generierten Bäume eingesetzt. Die Durchführung dieses Algorithmus muss zuerst für jeden Knoten des Baumes zwei Fehlerquoten berechnet werden. Eine Fehlerquote $C(t)$ ergibt sich aus dem Knoten, bei dem die ausgehenden Äste nicht entfernt werden. Andere Fehlerquoten $C\ (T_T)$ werden entstehen, wenn der Unterbaum aus diesem Knoten abgeschieden wird. Wenn die erste Fehlerquote größer als die zweite ist, wird dieser Knoten durch ein Blatt ersetzt, anderenfalls bleibt der Unterbaum. Dadurch wird eine Menge der stufenweise zurechtgestutzten Bäume entwickelt, dann wird in der Regel der Baum mit minimalsten Fehlerquote ausgewählt (Han und Kamber 2006; Rokach und Maimon 2015). Darüber hinaus wird die Complexity Cost pro Konten definiert als $\propto=$

$\frac{C(t)-C\,(T_t)}{|N(T_t)|-1}$, herbei sind $C\,(T_t)$ und $|N(T_t)|$ jeweils die Fehlerquote des Unterbaum und Anzahl seines Blattknotens (Lämmel und Cleve 2014; Petersohn 2005).

Der Pessimistic Pruning Algorithmus gehört zu anderem verbreitet angewandten Ansatz in ID3 und C4.5 Algorithmus (Quinlan 1993). Das Pessimistic Pruning ist ähnlich wie das Cost Complexity Pruning, da die Fehlerquote für das Pruning der Unterbäume mittels dieses Ansatzes berechnet wird. Jedoch ergibt sich die Fehlerquote beim Pessimistic Pruning aus dem Trainingsdatensatz anstatt der Menge der Unterbäume (Han und Kamber 2006). Auf diese Weise erzeugt diese Korrelation noch eine optimistische Fehlerquote. Weiterhin, das Fehlerreduktion Pruning (Error-Based Pruning) ist einer Weiterentwicklung von dem Pessimistic Pruning und kommt in der Regel zum Beschneiden der mittels C4.5 generierten Bäume zum Einsatz (Rokach und Maimon 2015). Dazu werden die in Bezug auf die Trainingsdaten generierten Entscheidungsbäume mit Hilfe der Test- oder Validierungsdaten beschnitten (Mitchell 1997). Werden durch das Abschneiden des Unterbaums die Klassifikationsfehler stärker reduziert, dann wird der Unterbaum durch den Blattkonten ersetzt. Falls keine solchen Unterbäume existieren, ist das Pruningverfahren fertig (Mitchell 1997).

Es besteht auch die Möglichkeit, die entwickelten Entscheidungsbäume in eine Regelmenge umzuwandeln. Hierbei lässt sich der Regel durch einen Blattknoten mit den Bedingungen erzeugen, die die von der Wurzel über die Kante zu diesem Blatt entwickelte Bedingung beinhalten (Witten und Frank 2001). Die Abbildung 2-11 illustriert die Umwandlung der Entscheidungsbäume in Regeln. Mit der umgewandelten Regeln werden die Entscheidungen vereinfacht eingesetzt.

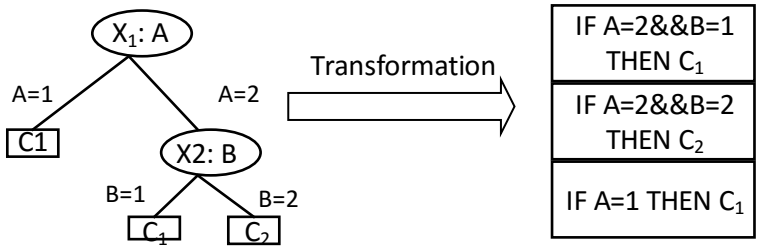


Abbildung 2-11: Umwandlung der Entscheidungsbäume in Regeln (Witten und Frank 2001)

2.3.3 Evaluation der Entscheidungsbäume

Nach der Generierung des Entscheidungsbaums kann dementsprechend ein Modell entwickelt werden. Hierbei muss das Modell in Bezug auf den relevanten Kriterien untersucht und evaluiert werden. Im Folgenden werden die relevanten Kriterien zur Evaluation und zur Verbesserung der Korrektheit des entwickelten Modells erläutert.

Zur Untersuchung des Modells werden zunächst die Ergebnisse der Klassifikation untersucht, wozu eine zweidimensionale Konfusionsmatrix von Bedeutung ist, wie die Tabelle 2-6 darstellt. Davon sind True Positives und True Negatives die korrekte Klassifizierungen. Im Gegensatz

dazu, während bei False Positives die falschen Ergebnisse als richtig klassifiziert werden, sind die richtige Ergebnisse bei False Negatives umkehrt als negativ zu klassifizieren. Jedes Matrixelement weist die Anzahl der Testdaten auf. Wenn die Hauptdiagonalen einen hohen Wert und die Nebendiagonalen einen niedrigen Wert zeigen, treten die positiven Ergebnisse mittels des Modells auf (Witten und Frank 2001). Außerdem, werden die Kosten und Gewinn des Modells durch Kennzahlen in der Konfusionsmatrix bewertet (Han et al. 2012). Während die Kosten durch die beiden Fehlerarten erzeugt werden, erzielt die korrekte Klassifikation die entsprechenden Gewinne (Witten und Frank 2001).

Tabelle 2-6: Konfusionsmatrix (Han et al. 2012)

		Vorhersagte Klasse	
		C_1	C_2
Tatsächliche Klasse	C_1	True Positives (TP)	False Negatives (FN)
	C_2	False Positives (FP)	True Negatives (TN)

Anschließend werden die relevanten Kennzahlen und die entsprechenden Formeln in Bezug auf die Klassifikationsmatrix in Tabelle 2-7 aufgelistet. Die Korrektheit (*Accuracy*) zeigt den Anteil der richtig klassifizierten Testdaten. Weiterhin werden die Kennzahlen wie *Sensitivity* (Recall, True Positives Rate) und *Specificity* (True Negatives Rate) eingesetzt, um die Fähigkeit des Modells, die die Instanzen als *True Positives* und *True Negatives* zu erkennen (Han und Kamber 2006). Die Korrektheit des Klassifizier kann auch durch die Kennzahlen *Sensitivity* und *Specificity* berechnet werden und die Berechnungsformel wird als $Korrektheit = sensitivity\frac{TP+FN}{TP+TN+FP+FN} + Specificity\frac{FP+TN}{TP+TN+FP+FN}$ beschrieben (Han et al. 2012). Anhand dieser Formel wird festgelegt, wenn die Instanzen jeder Klasse eine gleichgewichtige Verteilung darstellen, dann ist die Korrektheit zur Evaluation von wesentlicherer Bedeutung (Han et al. 2012). Ansonsten lassen sich die anderen Kriterien wie *Precision* und *Recall* zur jeweils Evaluation der Präzision und Vollkommenheit anwenden. Weiterhin kommt die Kennzahl *F-Maß* häufig als eine Kombination von *Precision* und *Recall* zur Evaluation der Modelle zum Einsatz, womit der Fehler der beiden Kennzahl ausglichen werden kann (Han et al. 2012).

Tabelle 2-7: Evaluationskennzahlen (Han et al. 2012)

Kennzahlen	Formel
Korrektheit (Accuracy)	$\frac{TP+TN}{TP+TN+FP+FN}$
Fehlerrate	$\frac{FP+FN}{TP+TN+FP+FN}$
True Positives Rate (TPR)/Recall/Sensitivity	$\frac{TP}{TP+FN}$

True Negatives Rate (TNR)/Specificity	$\frac{TN}{FP + TN}$
Präzision (Precision)	$\frac{TP}{TP + FP}$
F-Maß	$\frac{2 * precision * recall}{presion * recall}$

Um die Performance der Entscheidungsbäume weiter zu untersuchen, kommt das andere Kriterium „ROC-Kurve“ vermehrt zum Einsatz. ROC-Kurve steht für Receiver Operating Characteristic Kurve und hängt in der Regel von der Konfusionsmatrix ab (Han und Kamber 2006). Die ROC Kurve stellt die Abwägung zwischen True Positiver Rate (*TPR*) und False Negativer Rate (*FPR*) dar. Hierbei stellen die horizontalen und vertikalen Achsen jeweils False Negativer Rate (*FPR*) und True Positiver Rate (*TPR*) dar. Mit der ROC-Kurve kann auch der genieriete Klassifikator bewertet werden. Wenn das Modell eine hohe Korrektheit zeigt, nähren sich die Flächen unter der ROC-Kurve an 1, die als ein Kennzahl „Arae under ROC“ verwendet werden kann (Han und Kamber 2006).

Damit die Evaluationsqualität des entwickelten Modells durch den Entscheidungsbaum verbessert werden kann, kommen in der Regel die Methoden wie Holdout, *k*-fache Kreuzvalidierung (*k*-fold cross-validation) zum Einsatz. Dabei wird der Datensatz mittels Holdout Methode durch das Zufallsprinzip in den Trainings- und Testdatensatz aufgeteilt. Die Korrektheit und die Leistung des durch die Trainingsdaten entwickelten Modells werden durch die Testdaten überprüft (Han und Kamber 2006). Die Holdout Methode wird in Abbildung 2-12 illustriert (Han und Kamber 2006). Durch den Vergleich der Korrektheit anhand der Trainingsdaten und anhand der Testdaten kann die Performance der Entscheidungsbäume bewertet werden. Beispielsweise, wenn die Korrektheit durch die Testdaten kleiner ist als die Korrektheit durch die Trainingsdaten, stellt der generierte Entscheidungsbaum normalerweise das Problem der Überanpassung dar. Hierbei kommt in der Regel die Methode Pruning zur Lösung dieser Probleme zum Einsatz (Han und Kamber 2006).

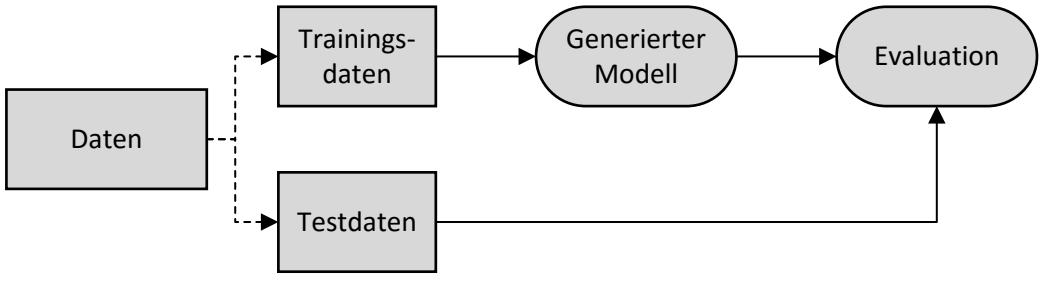


Abbildung 2-12: Evaluation mit der Holdout Methode (Han und Kamber 2006)

Ist die Datenmenge für das Training und den Test nicht ausreichend, kommt häufig die k-fache Kreuzvalidierung zur Evaluation des Modells zum Einsatz. Im Rahmen der Kreuzvalidierung gliedern sich die Daten durch das Zufallsprinzip in k Partitionen, wie D_1, D_2…D_k. Diese Partitio-

nen beinhalten eine erfordlichen Anzahl der Trainingsdaten (Han und Kamber 2006). Bei jeder Iteration wird die Partition D_j als Testdaten und die anderen als die Trainingsdaten eingesetzt. Die Korrektheit des Modells wird als das Verhältnis zwischen der Anzahl der richtig klassifizierten Daten aus *k*-fachen Iteration und Anzahl der initialen Daten. In der Praxis kommt vielmehr die 10-fache Kreuzvalidierung zum Einsatz, da die 10 fache Kreuzvalidierung eine relativ niedrige Bias und Variante darstellt (Han und Kamber 2006; Breiman 1984).

Damit die Korrektheit des Modells verbessert werden kann, kommt die Ensemble-Methode vermehrt zum Einsatz, die aus den mehreren entwickelten Modellen bestehen. Das Modell durch die Ensemble-Methode stellt in der Regel eine höhere Korrektheit als das einzelne Modell dar (Han und Kamber 2006). Davon sind Bagging, Boosting und Radom Forest verbreitet die populäre Ensemble-Methode (Han und Kamber 2006).

2.3.4 Vergleich der Entscheidungsbäume

Um einen geeigneten Entscheidungsbaum zur Untersuchung der Simulationsdaten auszuwählen, werden zunächst die verschiedenen Typen der Entscheidungsbäume vorgestellt. Anschließend werden die gängigen Entscheidungsbäume in Bezug auf die Anwendbarkeit verglichen und beurteilt. Schließlich werden die relevanten Kriterien zur Evaluation der Entscheidungsbäume zusammengefasst und erläutert.

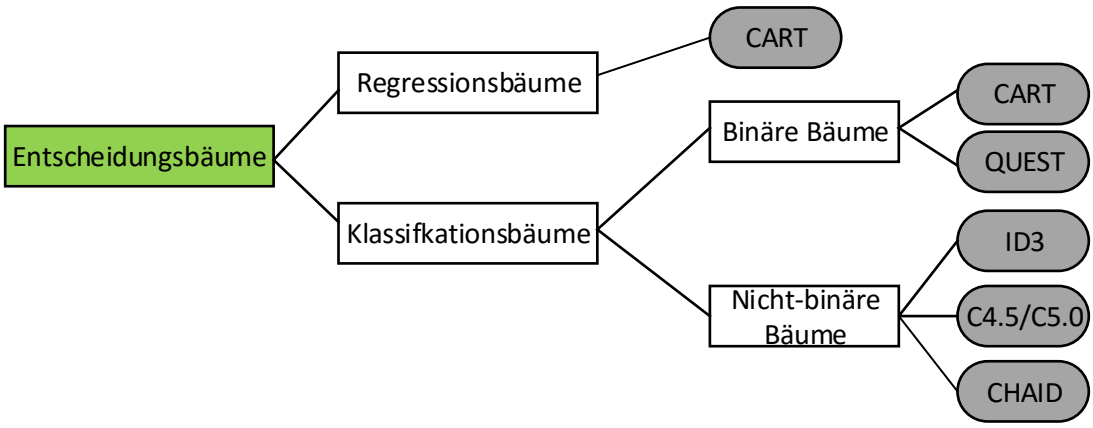


Abbildung 2-13: Klassifikation der Entscheidungsbäume

Die Klassifikationsmöglichkeiten der Entscheidungsbäume werden in unterschiedlichen Literaturen nach unterschiedlicher Kriterien ausgeführt. Beispielsweise gliedern die Entscheidungsbäume sich anhand des Anwendungsbereiches in die Klassifikationsbäume und die Regressionsbäume, wie in Abbildung 2-13 dargestellt (Ture et al. 2009; Rokach und Maimon 2015; Díaz-Pérez und Bethencourt-Cejas 2016). Die Klassifikationsbäume können zur Untersuchung des Modells angewandt, das Modell mittels der diskreten oder kontinuierlichen Attribute entwickelten wird. Hingegen kommen die Regressionsbäume in der Regel zur Untersuchung des Verhaltens des Modells, das nur mittels der kontinuierlichen Attribute generiert wird, zum Einsatz kommen (Han und Kamber 2006; Breiman 1984; Lämmel und Cleve 2014). Weiterhin, in Abhängigkeit der Anzahl der Verzweigungen an den Knoten unterscheiden sich die Entschei-

dungsbäume in binären und nicht-binären Bäume. Während jeder Knoten bei einem binären Baum gemäß dem Ausgang eines Tests genau zwei Verzweigungen besitzt, verfügt ein nicht-binärer Baum hingegen über zwei oder mehrere Verzweigungen an einem Knoten (Ture et al. 2009).

Tabelle 2-8: Vergleich der Entscheidungsbäume in Bezug der Algorithmen

Kategorie	ID3	C4.5/C5.0	CART	CHAID	QUEST
Zielvariable	kategorial	kategorial	alle	alle	kategorial
Attribute	nominal	alle	alle	alle	alle
Splitting-kriterium	Information Gain	Gain Ratio	Gini Index	χ^2-Test	χ^2-Test oder F-Test
Abbruch-kiterium	χ^2-test	Schwellen-werte	Schwellen-werte	χ^2- Test	Signifikanztest
Verzweigungen	nicht-binär	nicht-binär	binär	nicht-binär	binär
Pruning Algo-rithmus	Pessimistic Error Pruning	Error Based Pruning	Cost Com-plexity Pruning	keine	Cost Complexi-ty Pruning

Die Tabelle 2-8 stellt einen Vergleich der gängigen Entscheidungsbäume dar, wobei die am häufigsten angewandten Algorithmen ID3, C4.5/C5.0, CART, QUEST und CHAID miteinander in Bezug auf den verschiedenen Kriterien verglichen werden (Quinlan 1986; Quinlan 1993; Breiman 1984). Dabei sind die Kriterien „Format der Attribute und Zielvariablen“, „Splitting- und Abbruchkriterium“, „Anzahl der Verzeigungen“ und „Pruning Algorithmus“ von wesentlicher Bedeutung.

Während die Informationstheorie wie Informationsgewinn und Gain Ratio zur Entwicklung der Entscheidungsbäume als Splittingkriterien bei der Algorithmen wie ID3, C4.5, und C5.0 zum Einsatz kommen, werden die Unreinigkeit-basierten Methode Gini Index und die statischen Theorien des Chi-Quadrat-Tests und F-Tests bei der Algorithmen CART sowie CHAID und QUEST angewandt(Barros et al. 2015). Zudem gliedern sich die Entscheidungsbäume bezgl. der Anzahl der Attribute an einem Konto in univariaten, multivariant-linearen und nicht-linearen Entscheidungsbäumen (Ittner und Schlosser 1996). Während nur ein Attribut an jedem Knoten bei univariaten Entscheidungsbäumen bewertet wird, sind mehrere Attribute an jeden Konten bei den multivariant-linearen und nicht-linearen Entscheidungsbäumen linear oder nicht-linear zu untersuchen. Die Algorithmen ID3, C4.5/C5.0 können nur zur Entwicklung der univariaten Entscheidungsbäume angewandt werden (Ittner und Schlosser 1996). Im Gegensatz dazu können die Algorithmen CART und QUEST in Abhängigkeit der linearen Kombination der Attribute den multivarianten Entscheidungsbäume generiert werden (Han und Kamber 2006).

Weiterhin werden die Attribute bei den Entscheidungsbäumen zwischen geordneten und ungeordneten unterschieden. Während die geordneten Attribute die kontinuierlich numerischen Werte besitzen, verfügen die ungeordneten Attribute über die diskreten oder nominalen Werte (Witten und Frank 2001). Wenn der Entscheidungsbaum mittels der nominalen Werte generiert wird, ist die Anzahl der Verzweigungen gleich die Anzahl der nominalen Werte des Attributs. Hingegen wird die Anzahl der Verzweigungen mit Hilfe der Teilung der kontinuierlich numerischen Werte des Attribute gemessen, wenn der Entscheidungsbaum mittels den kontinuierlich numerischen Wert entwickelt wird (Witten und Frank 2001). Jedoch kann der ID3 Algorithmus nur mit ungeordneten Werten ausgeführt werden. Die anderen Algorithmen C4.5/C5.0, CART, CHAID und QUEST können nicht nur mit den ungeordneten Werten sondern auch mit den geordneten Werten umgehen (Polzin 2006). Dabei muss man beachten, dass die Entscheidungsbäume nur mit der kategorialen Zielattribute mittels der Algorithmen wie ID3, C4.5 und C5.0 entwickelt werden können (Polzin 2006). Aufgrund der unterschiedlichen Merkmale der Algorithmen müssen die Attribute und die Werte vor der Baumgenerierung in Abhängigkeit der Methode der Datenvorverarbeitung behandelt werden.

Damit der Vergleich der Stärke und der Schwäche der gängigen Entscheidungsbäume verdeutlicht wird, werden die relevanten Vor- und Nachteile dieser Algorithmen in Tabelle 2-9 aufgelistet.

Tabelle 2-9: Vergleich der Vor- und Nachteile der Algorithmen der Entscheidungsbäume

Typen	Vorteile	Nachteile
ID3	Einfaches Verstehen, Kleinere Baumgröße, kurze Laufzeit	Lokales Optimum, Überanpassungsproblem, Kategorische Zielattribute
C4.5	Behandlung mit numerischen, nominalen und Fehlenden Werten, kurze Laufzeit, Pruningverfahren	Kategorische Zielattribute, Nicht geeignet für große Datenbank
C5.0	Wie C4.5, Generierung der Regelmenge, Anwendbar für große Datenbank, Einsatz von Boosting	Keine detaillierte Vorstellung wegen Kommerzieller Gründe, Kategorische Zielattribute
CART	Behandlung mit den numerischen, nominalen und fehlenden Werten, Regressionsbäume, Pruningverfahren	Binäre Baumbildung, Lokales aber nicht globales Optimum, Keine Kombination der Variable, Lange Laufzeit
CHAID	Alle Format der Variablen, Nicht binäre Bäume	kein Pruningverfahren Annahme der statischen Modelle
QUEST	Univariate und linear splits	Basierend auf Normalverteilung

	Pruningverfahren	Binäre Baumbildung, Nominale Zielattribute

Nach dem Vergleich sind die Algorithmen C4.5 und C5.0 erheblich besser als ID3. Weiterhin nach der Untersuchung von Witten (2011) weist die Korrektheit der Entscheidungsbäume mittels C4.5 und mittels C5.0 sehr geringe Unterschiede auf (Witten et al. 2011). Ein wichtiger Fortschritt des C5.0 Algorithmus besteht in die Generierung der Regelmenge, jedoch ist die detaillierte Vorstellung noch nicht veröffentlicht (Witten et al. 2011). Nach Vergleich der Effizienz dauert die Konstruktion der Entscheidungsbäume mittels des CART Algorithmus erheblich länger als mittels des C4.5 Algorithmus (Sivasankari et. al 2014).

Damit ein geeigneter Entscheidungsbaum zur Untersuchung der Simulationsdaten ausgewählt werden kann, sollen nicht nur die Anwendungsbereiche sondern auch die anderen Kriterien wie Korrektheit und Effizienz der Entscheidungsbäume untersucht werden. Daher wird die Auswahl der Entscheidungsbäume im folgenden Abschnitt in zwei Schritte unterteilt. Zunächst wird die Phase der Vorauswahl anhand der Anwendbarkeit der Entscheidungsbäume in Bezug auf die gegebenen Datensätze durchgeführt. Anschließend wird die genaue Auswahl anhand der Evaluation der Entscheidungsbäume mit Hilfe der relevanten Kriterien durchgeführt. Die Kriterien der Evaluation der Entscheidungsbäume werden in der Tabelle 2-10 zusammengefasst, wobei die Berechnung der Korrektheit von der Konfusionsmatrix abhängt (Witten et al. 2011).

Im Rahmen der Auswahl der Entscheidungsbäume werden diese Kriterien aus den unterschiedlichen Modellen miteinander verglichen. In dieser Arbeit zählt die Korrektheit zu dem wichtigsten Evaluationskriterium. Weiterhin kennzeichnet das Kriterium der „Laufzeit“ die Dauer der Generierung der Entscheidungsbäume. Die „Laufzeit“ kann zur Auswertung der Effizienz der Entscheidungsbäume in Bezug auf die Arten oder Menge der Trainingsdaten eingesetzt werden. Darüber hinaus werden die Kriterien „Baumgröße“ und „Blätter“ der Entscheidungsbäume untersucht, womit die Analyse der Komplexität der Entscheidungsbäume durchgeführt werden kann (Su und Zhang 2006). Dabei repräsentiert die Kennzahl „Blätter“ die Anzahl der Blätter eines generierten Entscheidungsbaums. Die Kennzahl „Baumgröße“ kennzeichnet die Anzahl der Knoten eines Entscheidungsbaums (Witten und Frank 2001).

Tabelle 2-10: Kennzahlen der Evaluation der Modelle (Witten und Frank 2001)

Kennzahlen	Korrektheit	TP Rate/ Recall	Precsion	TN Rate	F-Maß	Area under ROC	Laufzeit
Wert	*	*	*	*	*	*	*

Zusammenfassend sollten die Entscheidungsbäume nicht nur die Anforderung der Korrektheit, sondern auch die weiteren Anforderungen erfüllen, damit die Anwendbarkeit der Entschei-

dungsbäume gewährleistet werden kann. Im Folgenden werden die Relevanten Anforderungen an die Entscheidungsbäume nach Han (2012) erläutert (Han et al. 2012):

Korrektheit (Accurancy): Der generierte Entscheidungsbaum soll eine höhere Korrektheit aufweisen.

Effizienz (Efficiency): Die Laufzeit der Konstruktion des Entscheidungsbaums soll so kurz wie möglich sein, gleichzeitig soll ebenfalls die Effizienz der Anwendung des Entscheidungsbaums effizient sichergestellt werden.

Robustheit (Robustness): Der Entscheidungsbaum soll in der Lage sein, die fehlenden und verrutschten Werte zu behandeln.

Interpretierbarkeit (Interpretability): Die Ergebnisse durch den Entscheidungsbaum sollten klar von dem Anwender verstanden werden.

Im Abschnitt 4.3 werden die vorliegenden Anforderungen in Kombination mit den Anforderungen an die simulationsbasierte Optimierung zur Validierung des Konzepts eingesetzt. D.h. das entwickelte Konzept soll nicht nur die Anforderungen an die Entscheidungsbäume sondern auch die Anforderungen an die simulationsbasierte Optimierung erfüllen.

3 Auswahl eines Entscheidungsbaumes und Analyse der Entscheidungen durch den Entscheidungsbaum

Nach der Erläuterung der relevanten Grundlagen zu der Simulationsstudien und der simulationsbasierten Optimierung sowie zu den Entscheidungsbäumen werden in diesem Kapital sowohl die Auswahl eines geeigneten Entscheidungsbaumes, als auch die Analyse der Entscheidungen durch den Entscheidungsbaum erläutert. Hierbei kommen zwei Datensätze aus dem Simulationsmodell zum Einsatz, die jeweils als Datensatz 1 und Datensatz 2 definiert werden. Zuerst werden die gegebenen Ausgabedaten im Rahmen der Datenvorverarbeitung behandelt. Danach wird der geeignetste Entscheidungsbaum durch die Evaluation der generierten Entscheidungsbäume in Abhängigkeit des Datensatzes 1 ausgewählt. Anschließend wird die Anwendbarkeit dieses Entscheidungsbaums mit Hilfe des Datensatzes 2 überprüft. Im Anschluss daran wird eine Auswertung der Entscheidungen durch den ausgewählten Entscheidungsbaum in Abhängigkeit des jeweiligen Datensatzes 1 und des Datensatzes 2 durchgeführt. Danach kommt die Kosten-Nutzen-Analyse zur weiteren Untersuchung der Entscheidungen auf Grundlage der Konfusionsmatrix zur Anwendung. Zum Schluss werden die Ergebnisse aus der Analyse der beiden Datensätze zusammengefasst. Die gewonnenen Erkenntnisse können als Voraussetzung für den Entwurf des Filters eingesetzt werden.

Um die Entscheidungsbäume mittels verschiedener Algorithmen in Abhängigkeit der Beispieldatensätze zu generieren, kommt die Software Weka und IBM SPSS Modeler 14.2 zum Einsatz. Die Entscheidungsbäume mit Hilfe der C4.5 und CART Algorithmen werden durch die Software Weka untersucht. Darüber hinaus lassen sich die Entscheidungsbäume mittels C5.0, CHAID und QUEST innerhalb der Software IBM SPSS Modeler 14.2 generieren und auswerten.

3.1 Beschreibung und Vorverarbeitung der Simulationsdaten

In diesem Abschnitt werden die Ausgabedaten (Datensatz 1 und Datenatz2) für die Generierung der Entscheidungsbäume vorverarbeitet. Dazu werden die entsprechenden Methoden der Datenvorverarbeitung zur Behandlung der Daten eingesetzt. Nach der Datenverarbeitung werden diese Daten als Eingabedaten für den Aufbau der Entscheidungsbäume angewandt.

3.1.1 Vorverarbeitung des Datensatzes 1

Der gegebene Datensatz 1 und der gegebene Datensatz 2 sind aus den Simulationen des Modularen Produktionssystemmodells. Hierbei liegen insgesamt 18 Attribute im Datensatz 1 vor, nämlich „OrderId“, „OrderName“, „CustomerID“, „Count“, „Green“, „Red“, „Yellow“, „Transition: splcl-DelayTime“, „splcl-BaseFiringTime“, „splbat-DelayTime“, „splbat-BaseFiringTime“, „initSub-DelayTime“, „initSub-BaseFiringTime“, „storet-DelayTime“, „storet-BaseFiringTime“, „shipt-DelayTime“, „shipt-BaseFiringTime“ und „Result“, wie in Tablle 6-2 (Anhang) dargestellt.

Da die Attribute, deren Bezeichnung „Delay-Time“ und „BaseFiringTime“ beinhaltet, mit 20 und 10 ZE jeweils über die gleichen Werte verfügen, werden diese Paramater vereinigt und jeweils durch „Delay-Time“ und „BaseFiring-Time“ ersetzt. Danach wird dieser Datensatz 1, der nur 10 Attribute beinhaltet (Tabelle 3-1), weiter verarbeitet. Vor der Generierung der Entscheidungsbäume können zunächst die Ausgabedaten der Simulationsstudie mit Hilfe von statistischen Methoden analysiert werden, um einen ersten Einblick in die Daten zu gewinnen.

Tabelle 3-1: Beispieldaten aus dem Datensatz 1 mit 10 Attributen

Order-Id	Order-Name	Customer-ID	Count	Green	Red	Yellow	Delay-Time	BaseFiringTime	Result
1	Order1	9005	2055	530	592	933	20	10	41230
2	Order2	4129	1975	814	724	437	20	10	39630
3	Order3	9853	593	54	524	15	20	10	11990
4	Order4	5813	947	337	610	0	20	10	19060
5	Order5	7521	1254	109	478	667	20	10	25210
6	Order6	8473	703	187	224	292	20	10	14190
7	Order7	5942	956	37	208	711	20	10	19250
8	Order8	8868	1735	491	406	838	20	10	34830
9	Order9	3847	1553	473	829	251	20	10	31190
10	Order10	5032	1537	221	716	600	20	10	30870

In der Simulationsstudie werden die Simulationsexperimente wiederholt durchgeführt, wozu eine große Menge an Ausgabedaten gegeben ist. Wenn die Simulationsdaten in unterschiedlichen Tabellen gespeichert werden, müssen diese Daten anhand der Methode der Datenintegration in eine Tabelle integriert werden (vgl. 2.3.1). Danach werden die Attribute und Werte für die Generierung der Entscheidungsbäume ausgewählt, um die Effizienz der Konstruktion zu erhöhen. Da die Information Gain und Gain Ratio von den Attributen „Delay-Time“ und „BaseFiringTime“ (anhand der Formel 2-3 und 2-4) gleich „0“ ist, werden diese Attribute für den Baumaufbau als ungültig betrachtet und entfernt. Darüber hinaus ist das Attribut „Order-Name“ ausschließlich die Bezeichnung der Aufträge und gleicht dem Attribut „OderID“. Daher werden die Attribute und Werte, die für die Konstruktion der Entscheidungsbäume unnötig sind, in der Phase der Datenvorverarbeitung entfernt. Auf diese Weise kann die Effizienz der Generierung der Entscheidungsbäume erheblich erhöht und die Komplexität der Bäume reduziert werden. Nach der Datenvorverarbeitung des Datensatzes 1 können die Attribute „CustomerID“, „Count“, „Green“, „Red“, „Yellow“ und „Result“ für die weitere Untersuchung einge-

setzt werden. Dabei repräsentieren die Attribute „Green“, „Red“, „Yellow“ Produkte. Das Attribut „Result“ repräsentiert die Länge der Simulationsläufe.

Weiterhin können diese Beispieldaten aus dem Datensatz 1 mit statistischen Verfahren bewertet werden. Damit kann man einen ersten Einblick in die Daten, z.B. mit Blick auf die Verteilung und die Zusammenhänge, erhalten. Dadurch wird der Zusammenhang der Attribute „Count“, „Green“, „Red“, „Yellow“ als *„Count“ = „Green“+ „Red“+„Yellow“* deutlich. Darüber hinaus wird der Zusammenhang der Attribute „Count“ und „Result“ in der Abbildung 3-1 dargestellt. Damit wird ersichtlich, dass die Werte der Attribute „Count“ und „Result“ eine Normalverteilung darstellen. Weiterhin, kann die Beziehung zwischen „Count“ und „Result“ visuell verdeutlichen, dass sich die Simulationslänge mit der Gesamtmenge der hergestellten Produkte durch eine positive Korrelation auszeichnet.

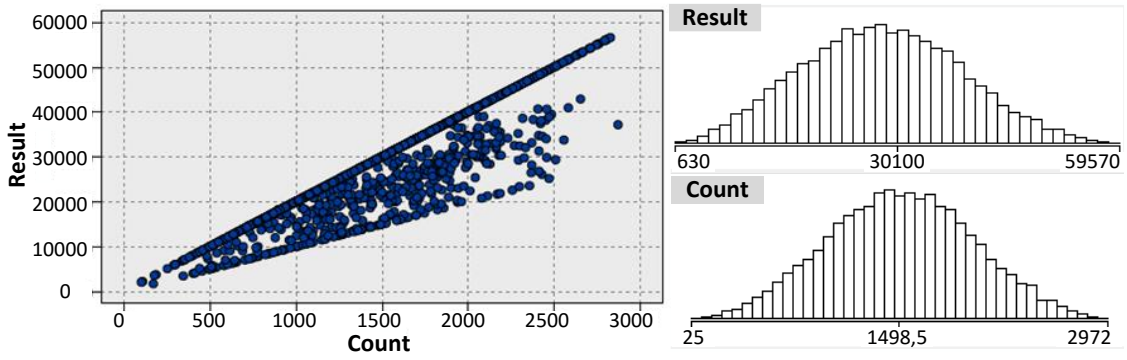


Abbildung 3-1: Analyse des Zusammenhangs zwischen den Attributen „Count“ und „Result“

Um die Klassifikationsbäume zu generieren, sollten die Zielattribute nominal und kategorial sein, insbesondere mittels ID3, C4.5 und C5.0. Darüber hinaus wird im Rahmen des Datensatzes 1 die Dauer der Simulationsläufe mit mehr als 30000 ZE als fehlerhaft definiert. Aus diesen Gründen werden die Ausgabedaten anhand der Simulationslänge, nämlich dem Wert von „Result“, in zwei Gruppen mit mehr als 30000 ZE und weniger als 30000 ZE aufgeteilt. Weiterhin wird das Attribut „Result“ nominiert und somit als Zielattribut zur Konstruktion der Entscheidungsbäume angewandt. Dieser Nominierungsprozess wird als eine Ja-Nein-Entscheidung in Regel 3-1 beschrieben. D.h., wenn der Wert von „Result“ mehr als 30000 ZE erreicht, dann wird „Result“ als „Nein” definiert, ansonsten als „Ja“.

Regel 3-1

If “Result“>= 30000 Then “Result“= “Nein“;

Else “Result“=“Ja“;

Endif

Nach der Datenvorverarbeitung kann der vorliegende Datensatz 1 zur Generierung der Entscheidungsbäume angewandt werden. Ein Beispiel des behandelten Datensatzes 1 wird in der Tabelle 3-2 gezeigt.

Tabelle 3-2: Beispieldaten nach Datenvorverarbeitung

Count	Green	Red	Yellow	Result
2055	530	592	933	Nein
1975	814	724	437	Nein
593	54	524	15	Ja
947	337	610	0	Ja
1254	109	478	667	Nein
703	187	224	292	Ja
956	37	208	711	Ja
1735	491	406	838	Nein
1553	473	829	251	Ja
1537	221	716	600	Ja

Die Zusammenhänge zwischen den Attributen und der nominierte Zielvariable, die aus Ja-Nein Entscheidung besteht, werden in der Abbildung 3-2 auf Grundlage des Datensatzes 1 illustriert. Die Anzahl der hergestellten Produkte „Green“, „Red“, „Yellow“ und die Gesamtmenge „Count“ werden mit Hilfe der Histogramme dargestellt. Dabei zeigt die Anzahl der hergestellten Produkte „Green“, „Red“ und „Yellow“ eine positive Korrelation mit der Simulationslänge, d.h. mit wachsender Anzahl der hergestellten Produkte der Simulationsexperimente nimmt die Simulationsdauer zu. Weiterhin liegen die Experimente mit einer längeren Simulationsdauer in dem Bereich, der über der durchschnittlichen Anzahl der Gesamtmenge der hergestellten Produkte liegt. Dabei repräsentiert der dunkle Bereich die Experimente mit kürzeren Simulationsläufen, nämlich mit dem Wert „Result“ weniger als 30000 ZE. Hingegen dauern die Experimente im hellen Bereich mehr als 30000 ZE. Nach der Untersuchung des Datensatzes 1 kann man 57,40% der Simulationsexperimente als simulierbar und 42,60% als fehlerhaft betrachten.

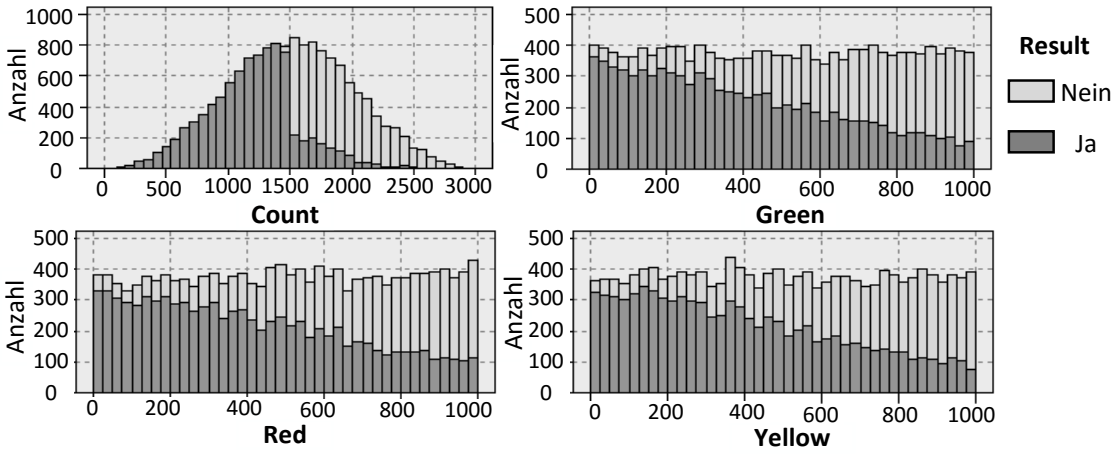


Abbildung 3-2: Illustration der Verteilung des Datensatzes 1 anhand der Ja-Nein-Entscheidung

3.1.2 Vorverarbeitung des Datensatzes 2

Um die Durchführbarkeit der Untersuchung durch die Entscheidungsbäume zu verbessern, sollte der Datensatz 2 des Simulationsmodells in dieser Arbeit zur Anwendung kommen. Die Ausgabedaten des Datensatzes 2 werden analog zum Datensatz 1 im Rahmen der Datenvorverarbeitung behandelt. Zuerst werden die Attribute des Datensatzes 2 in der Tabelle 3-3 beschrieben. Dabei laufen 3 Aufträge, die jeweils als Auftrag 0, 1 und 2 bezeichnet werden, durch das Simulationssystem. Zudem beschreibt das Attribut „#AvgDelay“ die mittlere Verspätung der Simulationsläufe von 3 Aufträgen. Das Ziel der Untersuchung des Datensatzes 2 besteht darin, die Simulationsläufe mit „#AvgDelay“ < 0 zu klassifizieren. Anschließend werden die Daten für die Generierung der Entscheidungsbäume vorverarbeitet. Die Attribute “Stop0, Stop1, Stop2” und “Delay0, Delay1, Delay2” werden in dieser Phase entfernt, da diese die Simulationsergebnisse sind. Diese Attribute können die Analysen durch die Entscheidungsbäume verfälscht werden. Das Attribut „Seed“ ist, wie „OderID“ im Datensatz 1, ebenfalls zu entfernen. Außerdem müssen die Attribute „#Orders“ und „#OrdersFinished“, welche die Anzahl der Aufträge repräsentieren, entfernt werden. Das Attribut „#AvgDelay” wird, wie “Result” im Datensatz 1, als Zielattribut verarbeitet. Das Ergebnis wird im Folgenden als Regel 3-2 beschrieben.

Regel 3-2

If “#AvgDelay“>0 Then “#AvgDelay“= “Nein“;

Else “ #AvgDelay“=“Ja“;

Endif

Tabelle 3-3: Beschreibung der Attribute des Datensatzes 2

Attribute	Beschreibung
Seed	Initialisierungswert des Zufallsgenerators für die Simulation
#Material	Anzahl des verfügbaren Materials zu Beginn der Simulation
#Orders	Anzahl der Aufträge, die das System durchlaufen sollen
#OrdersFinished	Anzahl der Aufträge, die das System bei Simulationsende durchlaufen hat
Count0	Anzahl der herzustellenden Produkte für Auftrag 0
Start0	Start der Bearbeitung des Auftrags 0
Stop0	Ende der Bearbeitung des Auftrags 0
TargetDuration0	Angestrebte Bearbeitungsdauer des Auftrags 0
Due0	Angestrebtes Bearbeitungsende des Auftrags 0 (=Start0+ TargetDuration0)
Delay0	Verspätung (= Stop0 - Due0)
#AvgDelay	Mittlere Verspätung ((= Delay0+ Delay1+ Delay2)/3)
Count1; Start1; Stop1; TargetDuration1; Due1; Delay1 und Count2; Start2; Stop2; TargetDuration2; Due2; Delay2 – jeweils analog zu dem Auftrag 0	

Weiterhin werden die Werte des Attributs „#Material“ in dieser Phase nominiert, da das Attribut „#Material“ nur über 4 Werte verfügt. Im Rahmen der Nominalisierung werden die Werte 50, 100, 150 und 200 jeweils durch „M50“, „M100“, „M150“ und „M200“ ersetzt. Durch die Nominierung des Attributs kann die Entwicklung der Entscheidungsbäume vereinfacht werden, was die Effizienz der Baumgenerierung erhöht. Die Tabelle 3-4 stellt eine Beispielmenge der für den Baumaufbau eingesetzten Daten aus dem Datensatz 2 dar, die 14 Attribute beinhaltet. Die Information Gain und Information Gain Ratio werden anhand der Formel (2-3 und 2-5) berechnet. Die Ergebnisse sind in der Tabelle 3-5 dargestellt.

Nach der statistischen Analyse des Datensatzes 2 kann die Beziehung zwischen dem Attribut „TargetDuration“ und „Count“ als “TargetDuration = 10*Count/3“ beschrieben werden. Weiterhin sind „Information Gain“ und „Gain Ratio“ von beiden Attributen identisch, wie Tabelle 3-5 zeigt. Für die Generierung der Entscheidungsbäume spielen die beiden Attribute folglich eine identische Rolle. Daher kann das Attribut „TargetDuration“ vor der Baumgenerierung entfernt werden, damit sich die Effizienz der Entwicklung der Entscheidungsbäume erhöht. Jedoch wird der Einfluss der Anzahl der Attribute auf die Effizienz der Baumgenerierung im folgenden Abschnitt detailliert überprüft.

Im Rahmen der Datenvorverarbeitung werden einerseits die Werte der Attribute in entsprechende Formate umgewandelt, anderseits werden die unnötigen Attribute herausgefiltert. Danach werden die behandelten Daten zur Generierung und Beurteilung der Entscheidungsbäume verwendet. Aus diesen Gründen sollte im folgenden Abschnitt ein Filterprozess anhand der Datenvorverarbeitung verwendet werden, um die Korrektheit und Effizienz der Entscheidungsbäume zu verbessern.

Tabelle 3-4: Beispieldaten von dem Datensatz 2 nach der Vorverarbeitung

#Material	Count0	Start0	Target-Duration0	Due0	Count1	Start1	Target-Duration1	Due1	Count2	Start2	Target-Duration2	Due2	#Avg-Delay
M150	96	456.331	320	776.331	40	790.302	133	923.302	167	1.166.700	556	1.722.700	Ja
M100	150	436.085	500	936.085	108	738.112	360	1.098.110	168	1.147.490	560	1.707.490	Ja
M150	183	468.735	610	1.078.730	97	857.684	323	1.180.680	108	1.177.190	360	1.537.190	Ja
M100	123	449.554	410	859.554	173	856.023	576	1.432.020	99	1.298.680	330	1.628.680	Ja
M50	196	308.545	653	961.545	221	640.948	736	1.376.950	139	1.028.650	463	1.491.650	Ja
M50	122	364.608	406	770.608	44	854.639	146	1.000.640	237	1.253.850	790	2.043.850	Ja
M50	217	486.772	723	1.209.770	242	797.160	806	1.603.160	112	1.284.330	373	1.657.330	Nein
M50	214	328.871	713	1.041.870	243	706.561	810	1.516.560	148	1.194.070	493	1.687.070	Nein
M200	88	362.956	293	655.956	97	806.253	323	1.129.250	233	1.283.500	776	2.059.500	Ja
M200	158	404.250	526	930.250	154	744.401	513	1.257.400	127	1.242.190	423	1.665.190	Ja

Tabelle 3-5 : Information Gain und Information Gain Raio von Datensatz 2

Attribut	Count0	Target-Duration0	Due0	Count1	Target-Duration1	Due2	Start2	Due1	Count2	Target-Duration2	Start1	Start0	#Material
Gain	0,06494	0,06494	0,05379	0,04535	0,04535	0.04037	0.02505	0.02284	0.01936	0.01936	0,00901	0	0
Gain Ratio	0,03556	0,03556	0,03887	0,02388	0,02388	0,01959	0,01308	0,01126	0,01218	0,01218	0,00629	0	0

3.2 Auswahl des Entscheidungsbaumes

In diesen Abschnitt wird der für die Analyse der Simulationsdaten geeignetste Entscheidungsbaum ausgewählt. Dazu werden zuerst die relevanten Anforderungen an die Entscheidungsbäume erläutert. Anschließend wird die Vorauswahl in Abhängigkeit dieser Anforderungen durchgeführt. Die vorgewählten Entscheidungsbäume sollten anhand der Anforderungen und unter Berücksichtigung der gegebenen Datensätze weiter untersucht werden. Dazu kommen Datensatz 1 und Datensatz 2 sowohl zur Generierung und Evaluation als auch zur Überprüfung des ausgewählten Entscheidungsbaums zum Einsatz.

3.2.1 Anforderungen an Entscheidungsbäume

Im Abschnitt 2.3.2 wurden bereits die relevanten Erkenntnisse bezüglich der verschiedenen Entscheidungsbäume vorgestellt. Die vorgestellten Entscheidungsbäume wurden im Abschnitt 2.3.4 schließlich mit Blick auf die Anwendbarkeit miteinander verglichen. Damit der geeignetste Entscheidungsbaum zur Optimierung der Simulationsstudie für Produktionssysteme ausgewählt werden kann, sollten zuerst die relevanten Anforderungen an die Entscheidungsbäume untersucht werden. In Kombination mit der Zielsetzung dieser Arbeit und den vorliegenden Datensätzen lassen sich die folgenden Anforderungen nennen:

- die Entscheidungsbäume sollten mittels der entsprechenden Algorithmen in Abhängigkeit der Trainingsdaten erfolgreich generiert werden. Die generierten Entscheidungsbäume dienen in dieser Arbeit vor allem zur Lösung der Klassifikationsprobleme
- die Algorithmen der Entscheidungsbäume sollten klar und einfach für die Anwender zu verstehen sein, damit weitere Entwicklungsmöglichkeiten gewährleistet sind
- die generierten Entscheidungsbäume sollten nicht nur zu einer größeren Korrektheit, sondern auch zu einer geringeren Komplexität führen. Damit kann die Anwendbarkeit und Verständlichkeit der Ergebnisse sichergestellt werden
- die Entscheidungsbäume sollten mittels der Algorithmen in Abhängigkeit der gegebenen Datensätze mit einer höheren Effizienz entwickelt werden, damit die Kosten des Verfahrens der Baumgenerierung minimiert werden können

Die Anwendbarkeit und Verständlichkeit der Entscheidungsbäume lassen sich in der Vorauswahlphase anhand der Erkenntnisse, die schon im Abschnitt 2.3.4 erläutert wurden, auswerten (vgl. 2.3.4). Um weitere Vergleiche der in der Vorauswahl gewählten Entscheidungsbäume vorzunehmen, werden die Entscheidungsbäume zunächst mit Hilfe der Trainingsdaten generiert. Anschließend lassen sich die Entscheidungsbäume anhand der Evaluationskriterien, wie z.B. Korrektheit und Effizienz, miteinander vergleichen. Nach der Auswertung der Ergebnisse wird der geeignetste Entscheidungsbaum zur Analyse der Daten des Simulationsmodells festgelegt.

Da der ausgewählte Entscheidungsbaum als einen zentralen Bestandteil des Filters zur Optimierung des Simulationsmodells eingesetzt wird, sollte die Auswahl des Entscheidungsbaums unter Einbezug der erläuterten Anforderungen sorgfältig durchgeführt werden. Die Qualität des ausgewählten Entscheidungsbaums spielt eine wesentliche Rolle im Hinblick auf die Qualität des Konzeptentwurfs und die Optimierung der Simulationsstudie.

3.2.2 Vorauswahl der Entscheidungsbäume

In Abhängigkeit des Datensatzes 1 lässt sich eine Vorauswahl der Entscheidungsbäume anhand der Anwendbarkeit der Entscheidungsbäume treffen. Das Format der vorliegenden Attribute des Datensatzes 1 wird in Tabelle 3-6 dargestellt. Dabei ist das Format des Zielattributs „Result“ nach der Vorverarbeitung als kategorial einzuordnen.

Tabelle 3-6: Formate der Attribute des Datensatzes 1 nach Datenvorverarbeitung

Attribut	Count	Green	Yellow	Red	Result
Format	numerisch	numerisch	numerisch	numerisch	kategorial

Da beim ID3 Algorithmus nur die nominalen Attribute mit der kategorischen Zielvariable zum Aufbau der Entscheidungsbäume eingesetzt werden können, ist der ID3 Algorithmus zur Generierung der Entscheidungsbäume in dieser Arbeit nicht anwendbar (vgl. 2.3.2). Darüber hinaus fokussiert sich diese Arbeit insbesondere auf die Untersuchung des Zusammenhangs zwischen den Parametern des Simulationsmodells und dem Zielattribut „Simulationslänge“, welches in der Phase der Datenvorverarbeitung in ein kategorisches Attribut umgewandelt wird. Dieses Problem wird als ein Klassifikationsproblem betrachtet, weshalb die für die Regressionsprobleme eigesetzten Entscheidungsbäume nicht berücksichtigt werden.

Der Algorithmus C5.0 kann als eine Weiterentwicklung des Algorithmus C4.5 gesehen werden. Jedoch wird C5.0 wegen kommerzieller Gründe nicht veröffentlicht (vgl.2.3.2). Die beiden Algorithmen werden im Folgenden anhand des Kriteriums der Anwendbarkeit mit Hilfe der Simulationsdaten bewertet. Danach wird die Performance von C4.5 und C5.0 verglichen, um eine Auswahl zwischen C4.5 oder C5.0 zu treffen. Wenn C5.0 eine wesentlich größere Korrektheit als C4.5 aufweist, wird C5.0 in dieser Arbeit zum Einsatz kommen, ansonsten C4.5.

Weiterhin kann der CART Algorithmus die geordneten numerischen Attribute behandeln, weshalb der CART Algorithmus zur Lösung der Klassifikationsprobleme im Folgenden untersucht wird. Darüber hinaus werden vor allem die statistischen Methoden *F*-Test und Chi-Quadrat-Test bei der QUEST und CHAID Algorithmen als Kriterien der Attributauswahl zur Behandlung der numerischen Werte eingesetzt.

Die Auswahl des geeignetsten Entscheidungsbaums lässt sich anhand der Anwendungsbereiche sowie der Vor- und Nachteile der vorliegenden Algorithmen nur schwer durchführen. Daher sollte die Performance der Entscheidungsbäume anhand der gegebenen Datensätze vergli-

chen und untersucht werden, damit der adäquate Algorithmus zur Generierung der Entscheidungsbäume ausgewählt werden kann.

Zusammenfassend kann festgehalten werden, dass in dieser Phase in Bezug auf die Algorithmen wie C4.5, C5.0, CART, QUEST und CHAID für die Generierung der Entscheidungsbäume eine Vorauswahl getroffen wird. Um die genaue Auswahl eines Entscheidungsbaums durchzuführen, werden die generierten Entscheidungsbäume anhand der Evaluationskriterien bewertet und verglichen. Dabei sind Kriterien wie Korrektheit und Effizienz von großer Bedeutung.

3.2.3 Auswahl des geeignetsten Entscheidungsbaums

In diesen Abschnitt werden zuerst die Entscheidungsbäume mittels verschiedener Algorithmen unter Einbezug des Datensatzes 1 generiert. Anschließend lassen sich die entwickelten Bäume anhand der Evaluationskriterien miteinander vergleichen. Nach der Gegenüberstellung der Entscheidungsbäume sollte der geeignetste Entscheidungsbaum für die Untersuchung der Simulationsdaten ausgewählt werden.

Zuerst werden die Entscheidungsbäume auf Grundlage des vorverarbeiteten Datensatzes 1, wie in Tabelle 3-3 gezeigt, mittels der vorliegenden Algorithmen generiert. Dabei werden 70% der Instanzen als Trainingsdaten und 30% der Instanzen als Testdaten angewandt. In diesem Fall werden die generierten Modelle in Abhängigkeit der Testdaten evaluiert. Die Analyse der Ergebnisse ermöglicht einen Vergleich der Entscheidungsbäume. Hierbei zeigen die Algorithmen C4.5, C5.0 und CART eine höhere Korrektheit als QUEST und CHAID (vgl. Tabelle 3-10). Jedoch sind die Abweichungen zwischen den verschiedenen Algorithmen sehr gering. Die Konfusionsmatrix der angewandten Algorithmen wird in der folgenden Tabelle dargestellt. Die Tabelle 3-7 zeigt die Konfusionsmatrix von C4.5, C5.0 und CART, da die Ergebnisse aus diesen Algorithmen identisch sind. Zudem wird die Konfusionsmatrix von QUEST und CHAID in Tabelle 3-8 bzw. 3-9 gezeigt. Die Analyse der Ergebnisse zeigt die sehr geringe Abweichung von QUEST und CHAID.

Tabelle 3-7: Konfusionsmatrix von C4.5, C5.0 und CART anhand des Datensatzes 1

		Vorhersagte Klasse	
		Ja	Nein
Tatsächliche Klasse	Ja	2190	342
	Nein	0	1868

Tabelle 3-8: Konfusionsmatrix von QUSET anhand des Datensatzes 1

		Vorhersagte Klasse	
		Ja	Nein
Tatsächliche	Ja	2194	388

Klasse	Nein	23	1845

Tabelle 3-9: Konfusionsmatrix von CHAID anhand des Datensatzes 1

		Vorhersagte Klasse	
		Ja	Nein
Tatsächliche Klasse	Ja	2200	332
	Nein	42	1826

Auf Grundlage der vorliegenden Konfusionsmatrix wird ein Vergleich der entwickelten Entscheidungsbäume anhand der im Abschnitt 2.3.4 vorgestellten Kriterien durchgeführt. Der detaillierte Vergleich wird in Tabelle 3-10 dargestellt. Die Entscheidungsbäume von C4.5, C5.0 und CART weisen mit 91,58% eine höhere Korrektheit auf als die von QUEST und CHAID mit jeweils 91.23% bzw. 91.04%. Jedoch sind die Abweichungen bei diesen Algorithmen sehr gering. Darüber hinaus ist die Laufzeit der Baumgenerierung mittels C4.5 kürzer als bei den anderen Algorithmen. Auch wird nach der Analyse der generierten Entscheidungsbäume deutlich, dass das Attribut „Count“ erheblich bedeutender für die Generierung der Entscheidungsbäume ist als die anderen Attribute, wie Abbildung 3-3 zeigt.

Tabelle 3-10: Evaluation der Entscheidungsbäume anhand der Beispieldaten mit „Count“

	Korrektheit (%)	Recall/TP Rate	FP Rate	Precision	F-Maß	Area under AOC	Laufzeit
C4.5	92,23	0,856	0	1	0,922	0,93	0,85
C5.0	92,23	0,856	0	1	0,922	0,93	1,00
CART	92,23	0,856	0	1	0,922	0,93	4,42
QUEST	91,73	0,850	0,012	0,990	0,915	0,93	1,00
CHAID	91,50	0,870	0,022	0,981	0,922	0,95	1,00

Im folgenden Abschnitt wird das Attribut „Count“ für den Aufbau der Entscheidungsbäume herausgefiltert, womit der Zusammenhang zwischen den hergestellten Produkten und der Simulationslänge untersucht wird. Zudem wird eine weitere Auswahl der Entscheidungsbäume anhand des Datensatzes 1 ohne das Attribut „Count“ durchgeführt. Auf diese Weise kann der geeignetste Entscheidungsbaum für die Untersuchung der Ausgabedaten des gegebenen Simulationsmodells ausgewählt werden.

Der Vergleich der Entscheidungsbäume wird nur anhand der wichtigsten Kriterien Korrektheit, Area under ROC sowie Laufzeit durchgeführt. Die anderen Kriterien werden in dieser Phase

nicht berücksichtigt. Die Korrektheit von C4.5, C5.0 und CART ist mit über 85% höher als QUEST und CHAID. Bezogen auf das Kriterium Area under ROC ist C4.5 und C5.0 genauer als die anderen Algorithmen, wie in Tabelle 3-11 dargestellt. Weiterhin dauert die Baumgenerierung mittels C4.5 und C5.0 weniger lang als bei den anderen Algorithmen, d.h. C4.5 und C5.0 Algorithmen zeichnen sich durch eine höhere Effizienz aus. Daher wird festgestellt, dass die Algorithmen C4.5 und C5.0 für den Datensatz 1 aus dem Simulationsmodell geeigneter sind. Jedoch wird das Pruningverfahren in der Auswahlphase in Bezug auf die generierten Entscheidungsbäume noch nicht angewandt.

Tabelle 3-11: Vergleich der Ergebnisse ohne das Attribut „Count“

	C4.5	C5.0	CART	QUEST	CHAID
Korrektheit (%)	88.15	87.90	85.77	83.00	78.55
Area under ROC	0,923	0,926	0,889	0,87	0,888
Laufzeit (s)	0,55	< 1	4,12	1,00	1,00

Um das Verhalten der Algorithmen angesichts der unterschiedlichen Datenmengen zu untersuchen, wird der Entscheidungsbaum mit Hilfe der Trainingsdaten von jeweils 5%, 10%, 20%, 40% und 60% der gesamten Datenmenge (Datensatz 1) entwickelt. Die Menge von den Trainingsdaten erhöht sich von 750 über 1500, 3000, 6000 auf 9000 Instanzen. Nach dem Zufallsprinzip werden 25% der Gesamtdaten als Testdatensatz (3750 Instanzen) ausgewählt, dieser Datensatz wird als Datensatz3 definiert. Um die Performance der Entscheidungsbäume zu untersuchen, werden die entwickelten Modelle anhand des Testdatensatzes evaluiert. Dazu wird vor allem die Korrektheit der Modelle verglichen und bewertet. In dieser Phase werden C4.5, C5.0 und CART zur Entwicklung der Entscheidungsbäume eingesetzt, womit die Unterschiede und Tendenzen der Entwicklungen vergleichbar dargestellt werden können. Es wird vor allem die Performance von C5.0 und C4.5 bewertet. Außerdem wird „Error Based Pruning“ bei C4.5 eingesetzt, um sowohl die Überanpassungsprobleme zu lösen als auch die kleineren Entscheidungsbäume zu generieren. Um die Qualität bei der Auswertung der Entscheidungsbäume zu gewährleisten, werden die Beispieldaten nach dem Zufallsprinzip 5-mal vom Datensatz 1 ausgewählt. Danach wird die durchschnittliche Korrektheit in Abhängigkeit der ausgewählten Bespieldaten berechnet. Die durchschnittliche Korrektheit von C4.5, C5.0 und CART wird in Tabelle 6-2 (Anhang) angezeigt.

Die Abbildung 3-4 illustriert die Tendenz der Veränderung der durchschnittlichen Korrektheit von C4.5, C5.0 und CART in Abhängigkeit der unterschiedlichen Mengen der Trainingsdaten. Hierbei repräsentiert die Abszisse die Anzahl der Trainingsdaten und die Ordinate die durchschnittliche Korrektheit des Modells. Dabei kann festgestellt werden, dass die Korrektheit der Modelle von allen Algorithmen mit zunehmender Anzahl der Trainingsdaten steigt. Weiterhin sind jedoch die Abweichungen der Korrektheit von C4.5 und C5.0 sehr gering. Daher könnte

entweder C4.5 oder C5.0 für die Untersuchung des Verhaltens der Simulationsdaten eingesetzt werden.

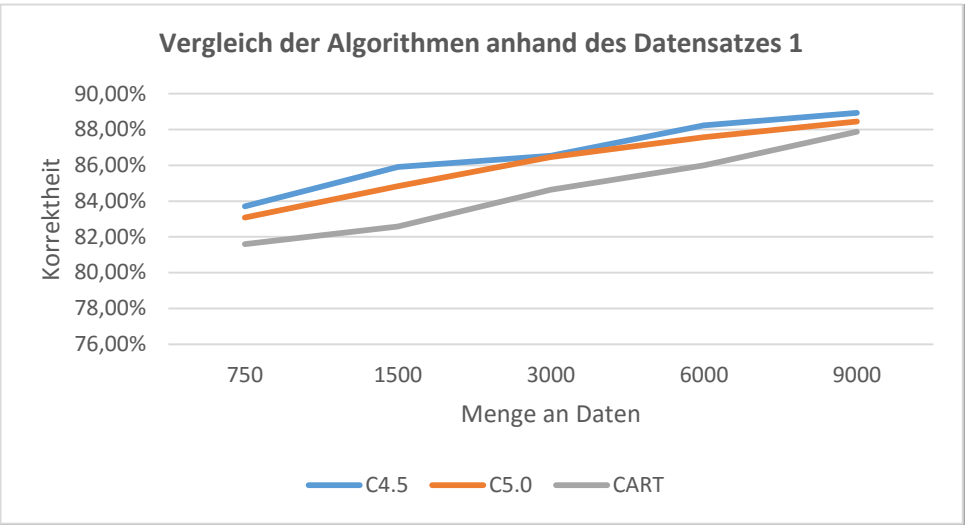


Abbildung 3-3: Vergleich der Algorithmen C4.5, C5.0 und CART nach Zufallsprinzip

Nach der Untersuchung der Anwendbarkeit in der Vorauswahlphase und dem Vergleich der Korrektheit mit anderen Algorithmen in der Auswahlphase wird der Algorithmus C4.5 zur Generierung der Entscheidungsbäume und zur weiteren Analyse der Ausgabedaten in dieser Arbeit verwendet. Einerseits weist C4.5 eine hohe Korrektheit mit Blick auf die gegebenen Simulationsdaten auf, andererseits ist der Inhalt von C4.5 vollständig veröffentlicht. Die Performance der Entscheidungsbäume durch C4.5 und C5.0 ist für die Beispieldaten fast gleich. Weitere Anwendungen von C5.0 sind jedoch wegen kommerzieller Gründe eingeschränkt, weshalb C4.5 insgesamt geeigneter erscheint.

3.3 Anwendungen des ausgewählten Entscheidungsbaumes

In diesem Abschnitt wird der mittels C4.5 Algorithmus ausgewählte Entscheidungsbaum eingesetzt. Zuerst werden die anhand der unterschiedlichen Beispieldaten generierten Entscheidungen ausgewertet, damit die Anwendbarkeit des ausgewählten Entscheidungsbaums überprüft werden kann. Anschließend lässt sich die Qualität der Entscheidungen in Abhängigkeit zur Art und Menge der Trainingsdaten untersuchen. Danach werden die durch die Analyse der Entscheidungen erworbenen Kenntnisse zusammengefasst, womit das Konzept der simulationsbasierten Optimierung entwickelt werden kann.

3.3.1 Auswertung der Entscheidungen

Abhängig vom Datensatz 1 konstruieren C4.5, C5.0 und CART eine fast identische Baumstruktur. Wenn der Entscheidungsbaum durch C4.5 aufgebaut wird, werden zuerst die Gain Ratio und Information Gain der Attribute berechnet (vgl. 2.3.2). Die Ergebnisse der Information Gain und Gain Ratio der Attribute „Count“, „Green“, „Yellow“ und „Red“ werden in Tabelle 3-12 dargestellt. Dabei ist „Count“ mit den Werten 0,6702 und 0,3724 erheblich größer als die an-

deren Attribute „Green“, „Yellow“ und „Red“ mit jeweils 0,1495 und 0,0510, 0,1367 und 0,0465 sowie 0,1275 und 0,0429. Von diesen spielt das Attribut „Count“, mit Hilfe von C4.5 entwickelt, eine bedeutende Rolle für den Aufbau der Entscheidungsbäume. Die anderen Attribute wie „Green“, „Yellow“ und „Red“ sind für die Baumgenerierung als irrelevant zu betrachten. Um die Korrektheit und Effizienz des ausgewählten Entscheidungsbaums C4.5 in Abhängigkeit der Ausgabedaten des Simulationsmodells zu untersuchen, wird die Phase der Generierung und Auswertung der Entscheidungsbäume in zwei Schritte aufgeteilt. Zuerst werden alle Attribute inklusive „Count“ „Green“, „Yellow“ und „Red“ eingesetzt, danach wird das Attribut „Count“ aus dem Datensatz 1 entfernt. In jedem Schritt werden die generierten Entscheidungsbäume detailliert bewertet.

Tabelle 3-12: Information Gain und Gain Ratio der Attribute

Attribute	Count	Green	Yellow	Red
Information Gain	0,6702	0,1495	0,1367	0,1275
Information Gain Ratio	0,3724	0,0510	0,0465	0,0429

Die Konfusionsmatrix des entwickelten Entscheidungsbaums ist vergleichbar mit Tabelle 3-7. Dabei sind 342 tatsächlich simulierbare Instanzen in den Testdaten durch die Entwicklung des Entscheidungsbaums mit Bezug auf Datensatz 1 als nicht simulierbar klassifiziert. Das bedeutet, 51,33% der Simulationsläufe sind von dem Simulationsmodell ausgeschlossen. Darüber hinaus werden überhaupt keine tatsächlich nicht simulierbaren Instanzen als simulierbar vorhergesagt. Weiterhin werden die Evaluationskriterien dieses generierten Entscheidungsbaums in Tabelle 3-13 dargestellt. Hier beträgt die Korrektheit und Area under ROC 92.23% bzw. 0,93. Daran kann man erkennen, dass der Entscheidungsbaum eine hohe Korrektheit besitzt.

Tabelle 3-13: Evaluation des Entscheidungsbaums mittels C4.5, Attribut „Count“

	Korrektheit (%)	Recall/TP Rate	TN Rate	Precision	F-Maß	Area under ROC
C4.5	92,23	0,856	1	1	0,922	0,93

Eine Entscheidung durch die Analyse der generierten Entscheidungsbäume findet statt, wenn der Wert von „Count“ kleiner oder gleich 1493 ist, denn dann laufen alle Simulationsexperimente weniger als 30000 ZE. Wenn der Wert von „Count“ im Gegensatz dazu größer als 1493 ist, laufen fast 83,5% der Simulationsexperimente länger oder gleich 30000 ZE. Die durch C5.0 entwickelte Baumstruktur wird, abhängig von den Trainingsdaten nach dem Zufallsprinzip, in der Abbildung 3-4 skizziert.

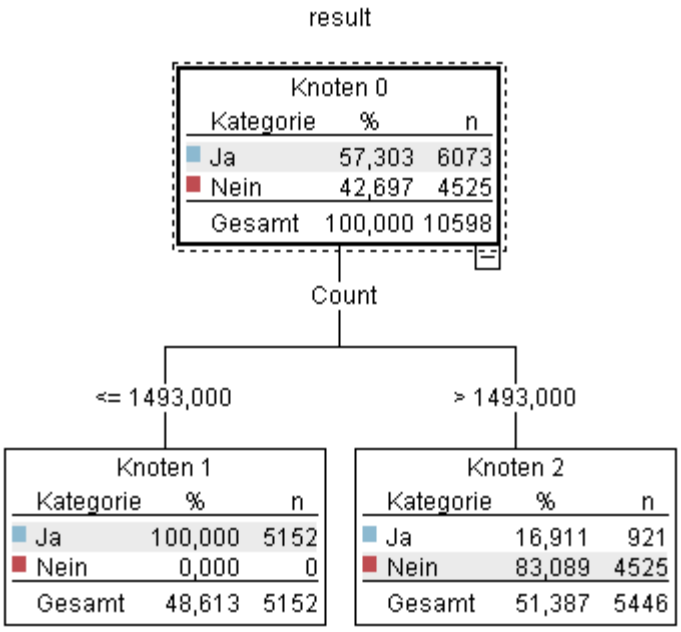


Abbildung 3-4: Entscheidungsbaum mit C5.0 anhand der Beispieldaten mit „ Count“

Nach der Untersuchung des generierten Entscheidungsbaums kann die Regel entwickelt werden, die im Folgenden als Regel 3-3 definiert wird.

Regel 3-3

(1) IF “Count“<= 1493 THEN

“Resulte“= “Ja”;

(2) IF “Count“> 1493 THEN

“Resulte“= “Nein“;

Wenn die Gesamtmenge aus den Eingangsdaten nicht erkennbar ist, sollte der Einfluss der hergestellten Produkte auf die Simulationslänge untersucht werden. Davor werden vor allem die Zusammenhänge zwischen der Gesamtmenge der hergestellten Produkte und der Simulationslänge untersucht. Um die Beziehungen zwischen den hergestellten Produkten „Green“, „Yellow“ „Red“ und die Dauer der Simulation zu untersuchen, wird im Folgenden der Entscheidungsbaum mittels C4.5 in der Software Weka aufgebaut. Um sowohl die Robustheit des Klassifikators als auch eine klare Baumgenerierung zu gewährleisten, wird die Methode „Error Based Pruning“ eingesetzt. Zudem wird ein Stellenwert zum Abbruch der Baumkonstruktion eingestellt. Dabei wird die Anzahl der Instanzen von jedem Blatt mit nicht weniger als 40 und der Konfidenzfaktor mit 0,25 angegeben. Auf diese Weise kann die Robustheit der Entscheidungsbäume innerhalb von C4.5 verbessert werden. Zugleich können die Entscheidungen aus einem kleinen Entscheidungsbaum besser untersucht werden.

Tabelle 3-14: Konfusionsmatrix von C4.5 anhand des Datensatzes 1 mit den Attributen „Green, Yellow, Red“

		Vorhersagte Klasse	
		Ja	Nein
Tatsächliche Klasse	Ja	2165	411
	Nein	191	1732

Nach der Baumgenerierung werden die Entscheidungen analysiert. Zuerst wird die Konfusionsmatrix in Tabelle 3-14 dargestellt. Es sind 4397 Instanzen (*TP+TN*) in die richtigen Klassen zugeordnet und die Korrektheit beträgt 86,62%. Darüber hinaus sind 2143 Instanzen (47,63%) der Simulation im Simulationsmodell nicht freigegeben. Für die falsch klassifizierten Instanzen werden 191 der tatsächlich nicht simulierbaren Experimente als simulierbar klassifiziert. Dagegen werden 411 der tatsächlichen simulierbaren Experimente als fehlerhaft klassifiziert. Der Anteil der falsch klassifizierten Instanzen beträgt 13,34%, davon liegen die Anteile von *FP* (Falsch Positiv) und *FN* (Falsch Negativ) jeweils bei 4,25% und 9,14%. Die detaillierten Evaluationskriterien des generierten Entscheidungsbaums werden in Tabelle 3-15 aufgelistet. Im Vergleich zum Entscheidungsbaum anhand der Beispieldaten ohne das Attribut „Count“ ist die Korrektheit dieses Entscheidungsbaums mit dem Attribut „Count“ niedriger geworden.

Tabelle 3-15: Evaluation des Entscheidungsbaums mit C4.5 ohne Attribut „Count“

	Korrektheit (%)	Recall/TP Rate	TN Rate	Precision	F-Maß	Area Under ROC
C4.5	86,62	0,84	0,901	0,919	0,878	0,915

Der entwickelte Entscheidungsbaum mit C4.5 beinhaltet 36 Blätter, wodurch 36 Regeln generiert werden können, da die Anzahl der Blätter der Anzahl der Regeln entspricht. Da die entwickelte Baumstruktur zu groß ist, wird ein Teil des Baumes in Abbildung 3-5 dargestellt. Der komplette Entscheidungsbaum ist im Anhang 6-1 darzustellen.

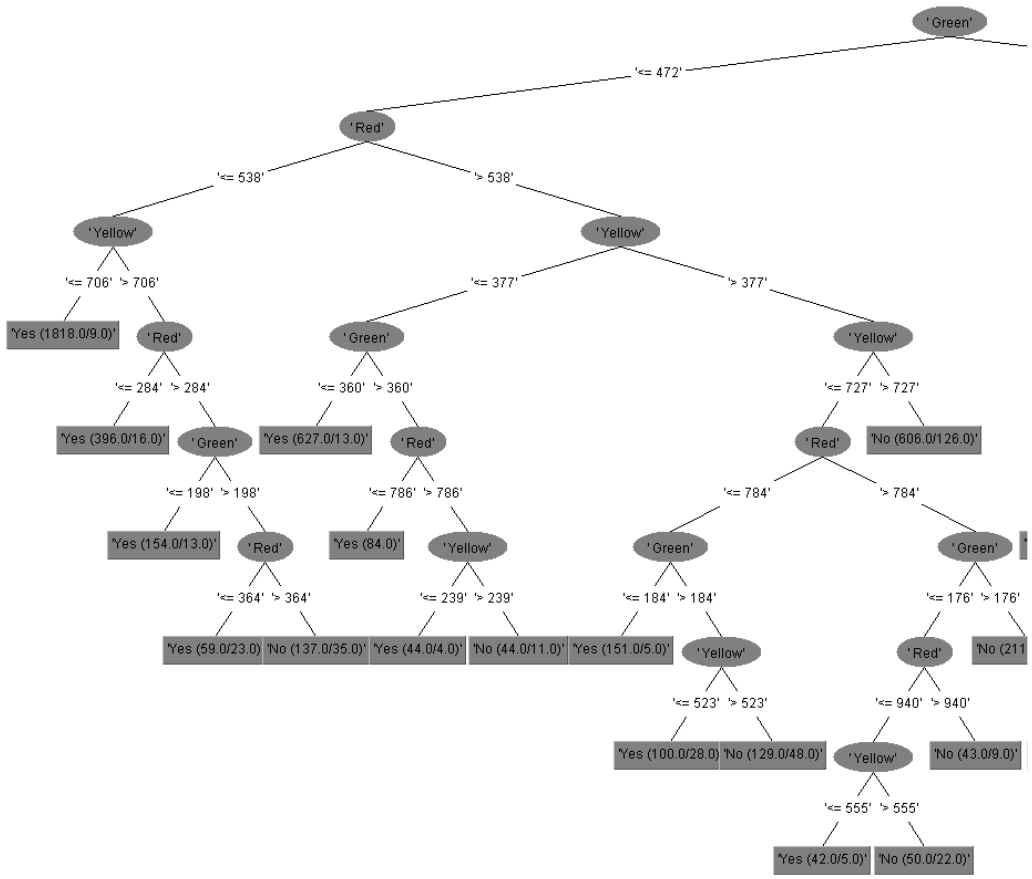


Abbildung 3-5: Entscheidungsbaum mittels C4.5 in Bezug auf den Datensatz 1 ohne das Attribut „Count“ (Teil)

Um die Ergebnisse des Entscheidungsbaums deutlich zu interpretieren, kann dieser Entscheidungsbaum in eine Menge von Regeln umgewandelt werden. Die kompletten Regeln, die aus dem Entscheidungsbaum in Abbildung 3-2 erstellt werden können, werden im Anhang dargestellt. Da die Menge der Regeln zu groß ist, wird nur ein Teil von ihnen, der auf eine große Anzahl der Instanzen zutrifft, vorgestellt. Im Folgenden wird ein Beispiel der Regeln als Regel 3-3 beschrieben. Dabei sind jeweils 2 Regeln als „Ja“- und „Nein“-Entscheidungen aufgelistet. Die Zahlen hinter jeder Regel repräsentieren sowohl die Anzahl der Beispieldaten als auch die Anzahl der falsch klassifizierten Instanzen. Beispielsweise zeigt die Regel „IF “Green” <= 472 and “Red” <= 538 and “Yellow”<= 706 THEN “Result” = “Ja”; (1818.0/9.0)”, dass die Simulationen als simulierbar klassifiziert werden, wenn die Werte der Attribute die vorliegende Regel erfüllen. Danach sind 1818 Instanzen als richtig und 9 als falsch klassifiziert. Mit Hilfe der Regeln aus den Entscheidungsbäumen wird vereinfacht überprüft, ob die Simulationsexperimente mit neuen Eingabedaten länger oder kürzer als 30000 ZE dauern. Im Vergleich zur Baumstruktur sind die Regeln für die Beschreibung der Klassifikations-möglichkeiten in diesem Abschnitt geeigneter.

Regel 3-4

(1) IF “Green” <= 472 and “Red” <= 538 and “Yellow”<= 706 THEN

"Result" = "Ja"; (1818.0/9.0)

(2) IF "Green" > 472 and "Red" <= 405 and "Yellow" <=386 THEN

"Result" = "Ja"; (800.0/24.0)

(3) IF "Green" > 472 and "Red" > 278 and "Yellow" > 613 THEN

"Result" = "Nein"; (1528.0/130.0)

(4) IF "Green" > 472 and "Red" > 456 and 396 < "Yellow" <= 613 THEN

"Result" = "Nein"; (654.0/92.0)

Im Anschluss lassen sich die Entscheidungen des durch C4.5 entwickelten Entscheidungsbaum auf Grundlage des Datensatzes 2 analysieren. Dazu werden 15000 Beispieldaten aus dem Datensatz 2, analog zum Verfahren bzgl. des Datensatzes 1, zur Untersuchung der Entscheidungsbäume eingesetzt. Bezogen auf den Datensatz 2 werden die folgenden Ziele verfolgt: Zuerst lässt sich die Anwendbarkeit des C4.5 Algorithmus anhand des Datensatzes 2 noch einmal überprüfen, danach werden die Entscheidungen durch den C4.5 Entscheidungsbaum untersucht.

Die Analyse der Entscheidungsbäume mit Hilfe des Datensatzes 2 wird ähnlich wie das Verfahren beim Datensatz 1 durgeführt. Hierbei werden 15000 behandelte Daten für die Generierung und Evaluation des Entscheidungsbaums eingesetzt. Weiterhin wird die „Error Reduced Pruning Methode" zur Vereinfachung des Entscheidungsbaums angewandt. Zudem sollte die minimale Anzahl der Instanzen pro Blatt 40 sein. Die Gain Ratio und Information Gain der Attribute des Datensatzes 2 sind in Tabelle 3-5 dargestellt. Dieser Einstellung folgend lässt sich der Entscheidungsbaum mittels C4.5 in der Weka Software generieren.

Tabelle 3-16: Konfusionsmatrix von C4.5 anhand des Datensatzes 2

		Vorhersagte Klasse	
		Ja	Nein
Tatsächliche Klasse	Ja	2927	204
	Nein	815	554

Anhand der erstellten Konfusionsmatrix kann die Evaluation des Entscheidungsbaums durchgeführt werden. Es werden 3481 Instanzen (*TP+TN*) in die richtigen Klassen zugeordnet, die Korrektheit liegt bei 77,36%. Weiterhin lassen sich 758 Simulationsläufe (16,84%) in dem Simulationsmodell nicht ausführen. Für die falsch klassifizierten Instanzen werden 815 der tatsächlich nicht simulierbaren Experimente als simulierbar klassifiziert. Dagegen werden 204 der tatsächlich simulierbaren Experimente als fehlerhaft vorhergesagt. Die detaillierten Evaluationskriterien des generierten Entscheidungsbaums werden in Tabelle 3-17 aufgelistet. Nach der Auswertung der Ergebnisse der Tabelle 3-16 kann das Verhalten des C4.5 Algorithmus für den

Datensatz 2 ausgewertet werden. Hier bietet der durch C4.5 entwickelte Entscheidungsbaum die Möglichkeit, die tatsächlich simulierbaren Instanzen genauer erkennen zu können. Zugleich besitzt der C4.5 Entscheidungsbaum zur Erkennung der nicht simulierbaren Instanzen mit 40,5% eine niedrigere Korrektheit. Zusammengefasst ist festzustellen, dass der C4.5 Entscheidungsbaum zur Untersuchung des Datensatzes 2 eine akzeptable Performance bietet.

Tabelle 3-17: Evaluation des Entscheidungsbaums mit C4.5 in Abhängigkeit des Datensatzes 2

	Korrektheit (%)	Recall/TP Rate	TN Rate	Precision	F-Maß	Area Under ROC
C4.5	77,36	0,935	0,405	0,782	0,852	0,682

Nach der Untersuchung der Ergebnisse, insbesondere im Hinblick auf die Korrektheit, durch C4.5 in Abhängigkeit des Datensatzes 2 kann man feststellen, dass C4.5 zur Klassifikation der Eingabedaten des Simulationsmodells anwendbar ist. Anschließend werden die generierten Entscheidungen untersucht. Die Abbildung 3-3 illustriert einen Entscheidungsbaum anhand C4.5 in Bezug auf den Datensatz 2. Dieser Entscheidungsbaum beinhaltet 21 Blätter, woraus 21 Regeln für die Klassifikation des Datensatzes 2 abgeleitet werden können. Weil die entwickelte Baumstruktur zu groß ist, wird ein Teil des Baumes in Abbildung 3-6 dargestellt. Der komplette Entscheidungsbaum ist im Anhang 6-2 darzustellen

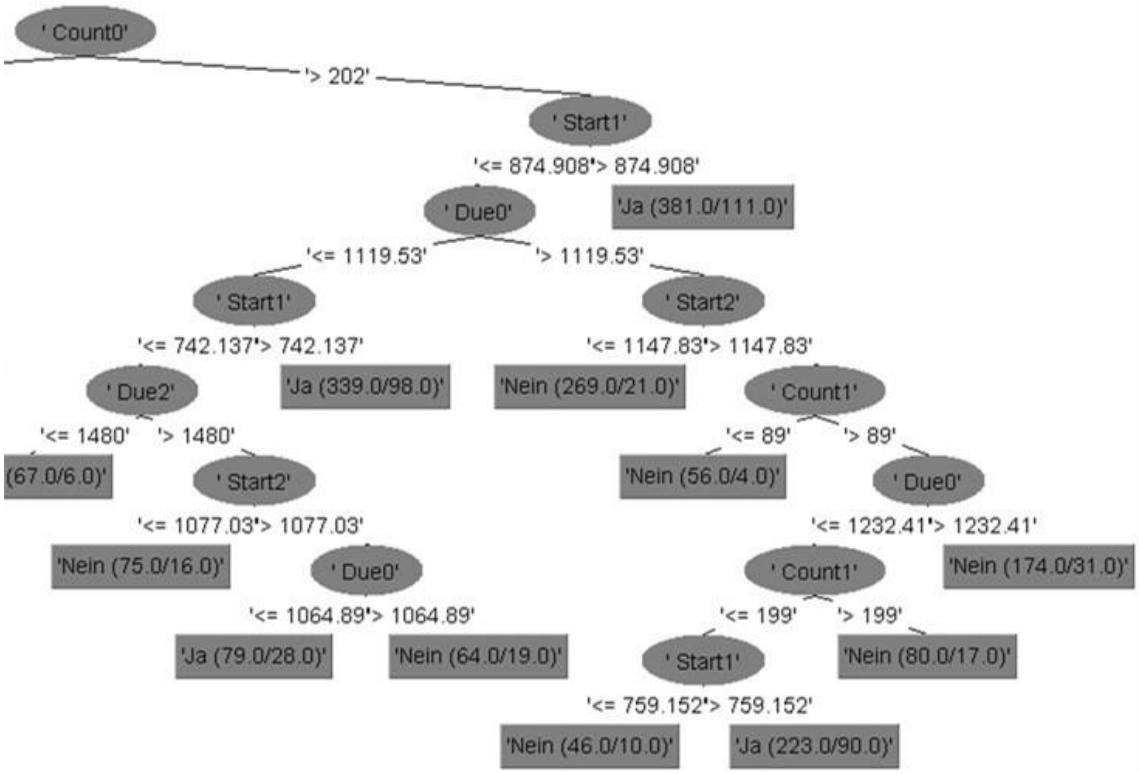


Abbildung 3-6: Entscheidungsbaum mit C4.5 in Abhängigkeit des Datensatzes 2 (Teil)

Durch die Analyse des Entscheidungsbaums ist ersichtlich, dass die Anzahl der herzustellenden Produkte des Auftrags (insbesondere Count0 und Count1) eine bedeutende Rolle für die Klassifikation der Simulationsläufe spielt. Beispielsweise kann eine der generierten Regeln als *„If*

Count0 <= 202 und Count1<=202, Then #AveDelay = "Ja"; (6671.0/1244.0)" beschrieben werden. D.h. 6617 Instanzen aus dem Trainingsdatensatz wurden mit einer Korrektheit von 84,28% als simulierbar klassifiziert. Weiterhin können fast 63% der Trainingsdaten (10500) mit dieser Regel untersucht werden. Die anderen Attribute, wie beispielsweise *„#Material",* werden zur Konstruktion des Entscheidungsbaums nicht eingesetzt. Im nächsten Abschnitt werden die Beziehungen zwischen Korrektheit, Effizienz und den Attributen des Datensatzes 2 detailliert untersucht.

Zusammenfassend lässt sich festhalten, dass der C4.5 Entscheidungsbaum eine gute Möglichkeit zur Klassifikation der Daten aus den gegebenen Datensätzen darstellt. Daher kann C4.5 zur Untersuchung und Klassifikation der Eingabe- und Ausgabedaten des Simulationsmodells eingesetzt werden. Nunmehr wird die Qualität der Entscheidungen durch C4.5 in Bezug auf die Menge und die Arten der Simulationsdaten untersucht.

3.3.2 Analyse der Qualität der Entscheidungen durch den C4.5 Entscheidungsbaum

Nach der Auswertung der Entscheidungen durch den ausgewählten Entscheidungsbaum mittels C4.5 wird in diesem Abschnitt untersucht, wie sich die Qualität der Entscheidungen auf Grundlage der Menge der Trainingsdaten darstellt. Im Abschnitt 3.2.2 ist die Performance der unterschiedlichen Entscheidungsbäume in Bezug auf die Menge der Trainingsdaten schon grob untersucht worden. Der beste Entscheidungsbaum zur Auswertung der Beispieldaten konnte identifiziert werden. Daher wird nun die Performance des C4.5 Entscheidungsbaums detailliert untersucht.

Um die Qualität der Entscheidungen in Bezug auf die Menge der Trainingsdaten zu untersuchen, werden der Datensatz 1 und der Datensatz 2 nach Zufallsprinzip in unterschiedliche Teilmengen gegliedert. Dabei werden 9 Teildatensätze aus dem Datensatz 1 und Datensatz 2 mit jeweils 1%, 5%, 10%, 20%, 30%, 40%, 50%, 60% und 70% untersucht. Hierbei wird der eingesetzte Datensatz 1 und der Datensatz 2 mit der Methode der Datenvorverarbeitung, wie in Abschnitt 3.1.1, bearbeitet. Die entsprechende Anzahl der jeweiligen Teildatensätze wird in den Tabellen 3-12, 3-13 und 3-14 dargestellt. Um die Genauigkeit der Bewertung zu erhöhen, wird dieser Prozess zufällig 5-mal anhand unterschiedlicher Datensätze durchgeführt. Weiterhin wird vor allem die Korrektheit der mittels C4.5 entwickelten Entscheidungsbäume bewertet und verglichen. Um die Qualität der Entscheidungen durch den Entscheidungsbaum anhand unterschiedlicher Attribute aus dem Datensatz 1 zu bewerten, wird die durchschnittliche Korrektheit der Entscheidungsbäume in Bezug auf Datensatz 1 jeweils mit und ohne das Attribut „Count" berechnet. Darüber hinaus werden die Kriterien wie z.B. Korrektheit und Baumgröße anhand der unterschiedlichen Anzahl der Attribute in Bezug auf Datensatz 2 untersucht. Hierbei wird das Pruningverfahren „Error Based Pruning" eingesetzt. Weiterhin sind die Schwellenwerte für den Abbruch der Baum-generierung nicht eingestellt, da sich die Menge der Trainingsdaten aufgrund einer hohen Schwankung verändert. Die Effizienz der Generierung der Entscheidungsbäume wird in Bezug auf die Menge der Trainingsdaten untersucht, da die Effizi-

enz stark von der Datenmenge beeinflusst wird. Besonders wichtig hierbei ist, dass die Effizienz des entscheidungsbaum-basierten Optimierungsverfahrens eng von der Effizienz der Entscheidungsbäume abhängt. Daher sollte ein Schwellenwert in Bezug auf die Korrektheit und die Effizienz der Entscheidungsbäume gesucht werden, um die Menge der Trainingsdaten einzuschränken. Dazu werden die Veränderungen und die Beziehungen der Korrektheit und Effizienz der Datensätze analysiert. Außerdem werden die Beziehungen zwischen der Anzahl der Attribute und der Baumgröße sowie der Effizienz anhand der vorliegenden Datensätze untersucht. Nach der Untersuchung werden diese Erkenntnisse zusammengefasst und als Grundkenntnisse zum Entwurf eines Filters eingesetzt.

Die durchschnittliche Korrektheit und die Laufzeit der generierten Entscheidungsbäume mit dem Attribut „Count“ in Bezug auf Datensatz 1 wird in Tabelle 3-18 aufgelistet. Die Laufzeit wird zur Messung der Effizienz der Baumgenerierung verwendet. Je größer der Wert der Laufzeit ist, desto geringer ist die Effizienz (vgl. 2.3.5).

Tabelle 3-18: Durchschnittliche Korrektheit und Laufzeit der Entscheidungsbäume mit „Count“

C4.5	150	750	1500	3000	4500	6000	7500	9000	10500
Korrektheit (%)	91,05	91,50	91,64	91,74	91,73	91,72	91,73	91,74	91,74
Laufzeit (s)	0,002	0,02	0,026	0,050	0,054	0,068	0,106	0,124	0,180

Durch die Untersuchung der durchschnittlichen Korrektheit und Laufzeit aus der Tabelle 3-18 kann man feststellen, dass 3000 Trainingsdaten für die Generierung der Entscheidungsbäume in Abhängigkeit zum Datensatz 1 mit dem Attribut „Count“ am besten sind. Dies erscheint sinnvoll, da die durchschnittliche Korrektheit der Entscheidungsbäume mit einer zunehmenden Menge der Trainingsdaten stabil bleibt, während die Effizienz erheblich abnimmt.

Im Folgenden sollen die durchschnittliche Korrektheit und Effizienz der Entscheidungsbäume ohne das Attribut „Count“ des Datensatzes 1 bewertet werden, wozu 3 Attribute zum Einsatz kommen.

Tabelle 3-19: Durchschnittliche Korrektheit und Laufzeit der Entscheidungsbäume ohne „Count“

C4.5	150	750	1500	3000	4500	6000	7500	9000	10500
Korrektheit (%)	76,48	83,32	84,42	86,46	87,51	87,92	88,53	89,08	89,01
Laufzeit (s)	0,012	0,022	0,036	0,074	0,11	0,134	0,154	0,162	0,20

Nach der Analyse der Ergebnisse der Tabelle 3-19 ist es ersichtlich, dass der Entscheidungsbaum mit Hilfe von 9000 Trainingsdaten eine höhere Korrektheit aufweist. Jedoch ist die entsprechende Laufzeit zugleich länger als die Laufzeit der Entscheidungsbäume mit einer gerin-

geren Anzahl an Trainingsdaten. Daher zeigen die Ergebnisse mit 3000 Beispieldaten eine hohe Korrektheit (über 85%) bei gleichzeitig größerer Effizienz. Daher kann der Schwellenwert zur Analyse dieses Datensatzes bei 3000 angesetzt werden. Weiterhin lässt sich, nach dem Vergleich der Ergebnisse von Tabelle 3-19 mit den Ergebnissen von Tabelle 3-20, die Beziehung zwischen der Laufzeit und der Anzahl der Attribute untersuchen. D.h. die Baumgenerierung dauert mit einer zunehmenden Anzahl an eingesetzten Attributen immer länger, was die Effizienz dementsprechend reduziert.

Daher können die Erkenntnisse der Analyse im Hinblick auf die Qualität der Entscheidungen mit Hilfe des Datensatz 1 zusammengefasst werden: Während die Korrektheit der Entscheidungsbäume mit zunehmender Menge der Trainingsdaten gestiegen ist, reduziert sich die Effizienz mit zunehmenden Menge der Trainingsdaten und der eingesetzten Attribute. Im Folgenden werden diese gewonnenen Erkenntnisse mit Hilfe des Datensatzes 2 überprüft.

Nach Untersuchung des Datensatzes 1 kommen die Attribute „Count“, „Green“, „Yellow“ und „Green“ sowie das Zielattribut „Result“ zur Generierung der Entscheidungsbäume zum Einsatz. Jedoch spielen nicht alle Attribute eine identische Rolle im Rahmen der Baumgenerierung. Daher wird nunmehr die Wichtigkeit der Attribute für die Entscheidungsbäume in Abhängigkeit des Datensatzes 2 analysiert. Durch den Vergleich der Kriterien, wie Korrektheit, Effizienz und Baumgröße der Entscheidungsbäume anhand unterschiedlicher Attribute, wird beschrieben, welche Attribute für die Generierung der besseren Entscheidungsbäume von Bedeutung sind. Danach werden die durchschnittliche Korrektheit und die Laufzeit in Abhängigkeit der unterschiedlichen Menge der Trainingsdaten des Datensatzes 2 untersucht, um die Erkenntnisse aus dem Datensatz 1 zu überprüfen.

Im Folgenden wird die Qualität der Entscheidungen auf Grundlage des Datensatzes 2 untersucht. Es gibt im Rahmen des Datensatz 2 Attribute, bei denen Information Gain kleiner als der durchschnittliche Wert ist. Daher wird ein Vergleich zwischen den Entscheidungsbäumen mit solchen Attributen und ohne solche Attribute gezogen. Weiterhin werden in dieser Phase auch Attribute mit identischen Gain Ratio Werten analysiert. Danach lässt sich eine Analyse der Korrektheit und Effizienz der Entscheidungsbäume in Bezug auf die Menge der Trainingsdaten durchführen.

Zuerst werden die 13 Attribute mit dem Zielattribut aus Datensatz 2 (vgl. Tabelle 3-4) zur Entwicklung der Entscheidungsbäume verwendet. Anschließend werden die Attribute, deren Gain Ratio und Information Gain zu klein sind, entfernt. Auf diese Weise werden die Attribute „#Material“, „Start0“ und „Start1“ entfernt. Nach der Auswertung der Entscheidungsbäume anhand des vollkommenden Datensatzes 2 ist festzustellen, dass die Attribute „TargetDuration0“, „TargetDuration1“ und „TargetDuration2“ keine Rolle spielen (vgl. 3.2.1). Daher werden auch diese Attribute entfernt. Daran anschließend wird die Effizienz der Baumgenerierung untersucht. Hier kommen 7 Attribute zum Aufbau der Entscheidungsbäume zum Einsatz. Jedoch spielen auch diese 7 Attribute keine gleichwertige Rolle, im Hinblick auf ihren Einfluss auf die Effizienz können auch die Attribute „Due1“, „Count2“ und „Start2“ zur Baumgenerierung

entfernt werden. Folglich werden die Entscheidungsbäume mit Hilfe der 4 Attribute „Due0“ „Count0“, „Count1“ und „Due2“ sowie dem Zielattribut „AveDelay“ entwickelt. Die Untersuchung kann weiter ausgeführt werden, indem die Attribute mit einer vergleichsweise niedrigen Information Gain entfernt werden. Entsprechend der Größe der Gain Ratio und Information Gain kann auch das Attribut „Due“ entfernt werden, sodass die Entscheidungsbäume nur mit den 3 Attributen „Due0“ „Count0“ und „Count1“ sowie dem Zielattribut „AveDelay“ aufgebaut werden. Diese unterschiedlich generierten Entscheidungsbäume kann man miteinander vergleichen. Die Ergebnisse davon bzw. der vorher erläuterten Untersuchungsphasen werden in die Tabelle 3-20 dargestellt.

Zur Generierung der Entscheidungsbäume werden in dieser Phase 15000 Daten aus dem Datensatz 2 eingesetzt. Hierbei wird kein Schwellenwert zum Abbruch der Baumgenerierung eingestellt. Die Kennzahl „Baumgröße“ und „Blätter“ werden zur Bewertung der Komplexität der generierten Entscheidungsbäume in Abhängigkeit der unterschiedlichen Anzahl der Attribute eingesetzt. Je mehr Blätter ein Entscheidungsbaum enthält, desto komplexer sind die Entscheidungen durch diesen Entscheidungsbaum. Die Kennzahl „Baumgröße“ kennzeichnet das Ausmaß der generierten Entscheidungsbäume.

Tabelle 3-20: Gegenüberstellung der Entscheidungsbäume mit unterschiedlichen Attribute

C4.5	Korrektheit (%)	Laufzeit (s)	Baumgröße	Blätter
Datensatz 2 mit 13 Attributen	76,77	1,72	299	167
Datensatz 2 mit 7 Attributen	76,71	0,81	109	55
Datensatz 2 mit 4 Attributen	76,08	0,37	39	20
Datensatz 2 mit 3 Attributen	73,64	0,23	19	10

Nach der Untersuchung der Ergebnisse der Tabelle 3-20 ist ersichtlich, dass die Korrektheit der Entscheidungsbäume variiert. Entscheidungsbäume mit nur 3 Attributen, welche nicht das Attribut „Due“ beinhalten, sind mit 73,64% erheblich weniger korrekt als die anderen Entscheidungsbäume mit jeweils 76,77%, 76,71% und 76,08%. Daher ist festzustellen, dass das Attribut „Due2“ für die Generierung eines Entscheidungsbaums mit hoher Korrektheit nicht entfernt werden darf. Zudem sind die Unterschiede bzgl. der Korrektheit der Entscheidungsbäume in den unterschiedlichen Fällen von jeweils 13, 7 und 4 Attributen und dem Zielattribut „AveDelay“ nicht erheblich. Daher kann man festhalten, dass nur die Attribute, welche sowohl eine höheren Information Gain als auch eine höheren Gain Ratio aufweisen, eine zentrale Rolle für die Baumgenerierung spielen. Jedoch ist die Laufzeit der Generierung des Entscheidungsbaums mit 4 Attributen deutlich kürzer als die Baumkonstruktion mit 7 oder 13 Attributen. Weiterhin ist die Komplexität der Entscheidungsbäume mit abnehmender Anzahl an Attributen stark reduziert, d.h. der generierte Entscheidungsbaum mit 4 Attributen beinhaltet nur 20 Blätter. Durch die Analyse des Verhaltens der Entscheidungsbäume in Abhängigkeit der unter-

schiedlichen Attribute aus dem Datensatz 2 wird deutlich, dass nur die wichtigsten Attribute für die Generierung der Entscheidungsbäume ausgewählt werden sollten. Auf diese Weise kann sowohl die Effizienz der Baumgenerierung erhöht als auch eine Reduktion der Komplexität der Entscheidungsbäume ohne Beeinträchtigung der Korrektheit der Entscheidungsbäume realisiert werden.

Um die Qualität der Entscheidungen in Abhängigkeit der Menge der Trainingsdaten des Datensatzes 2 weiter zu untersuchen, werden analoge Analyseverfahren zum Datensatz 1 durchgeführt. Dabei kommen nur die Attribute „Due0“ „Count0“, „Count1“ und „Due2“ sowie das Zielattribut „AveDelay“ zur Generierung der Entscheidungsbäume zum Einsatz. Um die Qualität der Untersuchung zu gewährleisten, sollen die Trainingsdaten 5-mal zufällig aus dem Datensatz 2 ausgewählt werden und die Testdaten (3000 Daten) zur Evaluation aller entwickelten Modelle identisch sein. Die durchschnittliche Korrektheit und Laufzeit der Entscheidungsbäume wird in Tabelle 3-22 dargestellt.

Tabelle 3-21: Durchschnittliche Korrektheit und Laufzeit in Abhängigkeit des Datensatz 2

C4.5	150	750	1500	3000	4500	6000	7500	9000	10500
Korrektheit (%)	71,93	74,93	76,55	77,14	77,532	78,03	78,56	78,47	79,11
Laufzeit (s)	0	0,018	0,1	0,102	0,13	0,166	0,196	0,24	0,332

Nach der Analyse der Ergebnisse aus Tabelle 3-21 ist ersichtlich, dass die Korrektheit und die Laufzeit des Klassifikators mit steigender Anzahl der Trainingsdaten aus dem Datensatz 2 zunehmen. Jedoch hängt die Bestimmung der Menge der eingesetzten Trainingsdaten ebenfalls von der erwarteten Korrektheit und Effizienz ab. Wenn bespielweise 3000 Trainingsdaten zur Generierung der Entscheidungsbäume zum Einsatz kommen, können über 2/3 der Testdaten innerhalb von fast 0,1s als richtig klassifiziert werden.

Nach dem Vergleich der Ergebnisse der Tabellen 3-19, 3-20 und 3-21, jeweils in Abhängigkeit des Datensatzes 1 und des Datensatzes 2, können die Erkenntnisse zusammengefasst werden. Die Korrektheit und Effizienz der Entscheidungsbäume sind sowohl von der Menge der Trainingsdaten als auch von der Anzahl der Attribute abhängig. Daher sollte eine geeignete Menge an Trainingsdaten in Abhängigkeit der erforderlichen Korrektheit bestimmt werden, um sowohl die Korrektheit als auch die Effizienz der Entscheidungsbäume zu gewährleisten.

3.3.3 Kosten-Nutzen-Analyse der Entscheidungsbäume

Die Hauptzielsetzung der Simulation und Optimierung von Produktionssystemen besteht darin, die Produktionsplanung in der Praxis zu verbessern. In diese Arbeit kann anhand der Analyse der Parameter des Simulationsmodells untersucht werden, ob bestimmte Aufträge vom Produktionssystem angenommen werden sollten oder nicht. Um diese Fragestellung zu beantwor-

ten, kann die Methode der Kosten-Nutzen-Analyse in Abhängigkeit der entsprechenden Konfusionsmatrix durch die Entscheidungsbäume eingesetzt werden.

Im ersten Teil einer Kosten-Nutzen-Betrachtung wird in Bezug auf die Kosten und Nutzen folgendes angenommen: die Kosten sind auf die Ablehnung der tatsächlichen simulierbaren Instanzen zurückzuführen. Der Nutzen wird durch die Herausfilterung der vorhergesagten nicht simulierbaren Instanzen gewonnen. In der Praxis könnten die Kosten als Verlustkosten der abgelehnten, aber herstellbaren Aufträge betrachtet werden. In Kombination mit der Konfusionsmatrix lassen sich Kosten und Nutzen jeweils als Anteil von FN und FN+TN in den Testdaten berechnen. Außerdem wird angenommen, dass die falsche Klassifikation der tatsächlich nicht simulierbaren Instanzen (FP) keine Kosten verursacht. In diesem Fall werden keine Aufträge abgelehnt, was im Rahmen der Nutzenkalkulation ermittelt wird.

Um den Prozess der Kosten-Nutzen-Analyse zu verdeutlichen, werden nunmehr die folgenden Annahmen eingeführt:

1. Wenn keine Optimierungsverfahren in das System eingesetzt werden, werden sowohl die Kosten als auch der Nutzen mit null beziffert.
2. Wenn die Korrektheit der Klassifikation 100% erreicht, werden die Kosten C_1 gleich null und zugleich der Nutzen als B_1 definiert. Dieser Nutzen wird hierbei als der optimale Nutzen des Systems betrachtet. Der optimale Nutzen B_1 wird in Kombination der Konfusionsmatrix als $\frac{FP+TN}{TF+FP+FN+TN}$ beschrieben. Weiterhin lässt sich der optimale Gewinn G_1 als $G_1=B_1$ definieren, da C_1 gleich null ist.
3. Wenn die Korrektheit weniger als 100% beträgt, werden die Kosten als C_2 und der Nutzen als B_2 definiert. Analog zur Annahme 2 lassen sich Ist-Kosten C_2 und der Ist- Nutzen B_2 jeweils als $\frac{FN}{TF+FP+FN+TN}$ und $\frac{FN+TN}{TF+FP+FN+TN}$ ausdrücken. Weiterhin kann der Ist-Gewinn G_2 des Systems mit $G_2 = B_2 - wC_2$ beschrieben werden. Hierbei wird eine Gewichtung *w* für die Ist-Kosten C_2 eingeführt und als $w>1$ definiert. Letzteres ist plausibel, da die Ablehnung der Aufträge in der Regel zur höheren Kosten für das Gesamtsystem führt. Wenn $G_2 > 0$ ist, ist die Optimierung durch Entscheidungsbäume gültig. Ansonsten ist diese Optimierung ungültig.
4. Um die Performance der Optimierungsverfahren zu bewerten, wird das Verhältnis zwischen dem Ist-Gewinn und dem optimalen Gewinn berechnet. Je mehr sich das Verhältnis an 1 nähert, desto höher ist der Ist-Gewinn und desto besser ist die Performance der Optimierungsverfahren. Wenn der optimale Gewinn als 1 definiert ist, dann könnte das Verhältnis genauso als der Ist-Gewinn betrachtet werden.

In Kombination mit der Konfusionsmatrix durch den Entscheidungsbaum werden die vorliegenden Annahmen in die Tabelle 3-22 detailliert beschrieben.

Tabelle 3-22 : Annahme der Kosten und des Nutzens des Systems

Kennzahl	Formel	Anmerkung
Optimale Nutzen B_1	$\frac{FP+TN}{TF+FP+FN+TN}$	Kosten für B_1 ist gleich null, C_1=0, dann G_1= B_1
Ist-Nutzen B_2	$\frac{FN+TN}{TF+FP+FN+TN}$	
Ist-Kosten C_2	$\frac{FN}{TF+FP+FN+TN}$	
Gewicht w	w	$w > 1$
Ist-Gewinn G_2	$B_2 - wC_2$	
Verhältnis	C_2/C_1	Je mehr sich das Verhältnis an 1 nähert, desto besser ist G_2

Im Folgenden wird die Kosten-Nutzen-Analyse in Kombination mit den vorliegenden Annahmen und in Abhängigkeit der Konfusionsmatrix aus den Tabellen 3-9, 3-15 und 3-17 durchgeführt, wobei die Gewichtung w mit 2 angenommen wird. Die Ergebnisse werden in der Tabelle 3-23 dargestellt. Die Unterschiede des optimalen Nutzens zwischen Datensatz 1 mit und ohne „Count“ sind auf die Zufälligkeit des Samplings zurückzuführen.

Tabelle 3-23: Kosten-Nutzen-Analyse anhand des Datensatzes 1 und des Datensatzes 2

	B_1	B_2	C_2	w	G	Verhältnis	Korrektheit
Datensatz 1 mit „Count“	0,415	0,491	0,076	2	0,339	0,817	92,23%
Datensatz 1 ohne „Count“	0,427	0,476	0,091	2	0,294	0,689	86,62%
Datensatz 2	0,304	0,168	0,045	2	0,078	0,257	77,36%

Nach den vorliegenden Untersuchungen ist ersichtlich, dass der Gewinn des Systems von der TP-Rate und TN-Rate abhängig ist. Im Abschnitt 2.3.3 wurde nach Han (2012) untersucht, dass das Verhalten der Kennzahlen TP-Rate und TN-Rate durch die Korrektheit dargestellt werden kann (vgl. 2.3.3), wenn die Instanzen jeder Klasse gleichmäßig verteilt sind. In Kombination mit den vorliegenden Datensätzen des Simulationsmodells ist zu folgern, dass die Verteilung der Instanzen der Klassen “Ja” und “Nein” relativ gleichmäßig ist. Daher wird in dieser Arbeit vor allem der Zusammenhang zwischen dem Gewinn und der Korrektheit untersucht.

Zuerst wurden die Kriterien des Datensatzes 1 in Tabelle 3-23 untersucht: Nach dem Vergleich der letzten beiden Spalten, nämlich „Verhältnis“ und „Korrektheit“, ist deutlich geworden, dass der Gewinn und die Korrektheit des Modells eine positive Korrelation aufweist. Anschließend wird das Ergebnis durch den Vergleich zwischen dem Datensatz 1 und dem Datensatz 2 über-

prüft. Der Vergleich zeigt hier ein identisches Ergebnis. Nach der Auswertung konnte folglich festgestellt werden, dass die Korrektheit eine wesentliche Rolle für den Gewinn des Systems spielt.

Im Abschnitt 3.3.2 wurde untersucht, dass sich die Korrektheit der Entscheidungsbäume mit zunehmender Menge der Trainingsdaten erhöht. In Kombination mit der Kosten-Nutzen-Analyse in diesem Abschnitt lässt sich festhalten, dass die Menge der Trainingsdaten zur Optimierung des Gewinns des Systems von wesentlicher Bedeutung ist. Weiterhin kann die Kosten-Nutzen Analyse im Folgenden ebenso zur Untersuchung der Effizienz, der Korrektheit und des Gewinn des Gesamtsystems eingesetzt werden.

3.3.4 Zusammenfassung der Erkenntnisse durch den C4.5-Entscheidungsbaum

Nach der Untersuchung der Entscheidungsbäume mittels des C4.5 Algorithmus in Abhängigkeit der gegebenen Datensätze aus dem Simulationsmodell sind einige wichtige Erkenntnisse abgeleitet worden. Im diesen Abschnitt werden diese gewonnenen Erkenntnisse zusammengefasst, die zum Entwurf eines Filters verwendet werden können.

Zuerst konnte gezeigt werden, dass der C4.5 Entscheidungsbaum zur Untersuchung der Ausgabedaten und zur Klassifikation der Eingabedaten des Simulationsmodells am effektivsten ist. Denn die Entscheidungsbäume mit Hilfe von C4.5 weisen eine bessere Performance als die anderen Algorithmen (vgl.3.2.1) auf. Weiterhin zeichnen sich die durch C4.5 Entscheidungsbäume generierten Entscheidungen durch eine gute Anwendbarkeit zur Klassifikation der Simulationsdaten aus. Daher kann der C4.5 Algorithmus einerseits zur Generierung und Auswertung der Entscheidungsbäume in Abhängigkeit der Ausgabedaten eingesetzt werden. Andererseits lässt sich der C4.5 Algorithmus als zentraler Bestandteil des Filters zur Optimierung des Simulationsmodells anwenden.

Zudem ist die Phase der Datenvorverarbeitung zur Generierung der Entscheidungsbäume mittels des C4.5 Algorithmus von wesentlicher Bedeutung. Im Rahmen der Datenvorverarbeitung lassen sich einerseits die irrelevanten Attribute entfernen, andererseits kann das numerische Zielattribut in ein Kategorisches umgewandelt werden. Auf diese Weise können die Entscheidungsbäume genauer und effektiver entwickelt werden.

Danach wird durch die Analyse der Qualität der Entscheidungen deutlich, dass die Korrektheit der Entscheidungsbäume mit zunehmender Datenmenge steigt. Weiterhin ist nach der Kosten-Nutzen-Analyse ersichtlich, dass der Gewinn des Systems ebenfalls von der Menge der Trainingsdaten abhängt. Jedoch kann eine große Menge an Trainingsdaten zu längeren Laufzeiten der Baumkonstruktion führen, sodass die Effizienz der Entwicklung der Entscheidungsbäume beeinträchtigt wird. Dies ist besonders hervorzuheben, da das Ziel der Arbeit in der Verminderung der Anzahl der länger laufenden Simulationen besteht. Daher sollten auch die Optimierungsverfahren nicht zu viel Zeit in Anspruch nehmen. Aus diesem Grund sollte folglich ein Schwellenwert für die Baumgenerierung festgelegt werden. Auf diese Weise können nicht nur

die Korrektheit, sondern auch die Effizienz und der Nutzen des Klassifikators sichergestellt werden.

Darüber hinaus sind, anhand der Analyse der Ergebnisse aus Tabelle 3-19, die Anzahl und die Qualität der Attribute zur Generierung der Entscheidungsbäume von großer Bedeutung. Zu viele Attribute führen zu einer zunehmenden Komplexität und zu längeren Laufzeiten der Entscheidungsbäume, obwohl sich die Korrektheit zugleich nicht signifikant erhöht. Daher sollten die Attribute vor der Baumgenerierung sorgfältig überprüft und ausgewählt werden. Die Auswahl der Attribute für die C4.5 Entscheidungsbäume sollte hierbei in Abhängigkeit der Werte Information Gain und Gain Ratio durchgeführt werden. Die Attribute sollten sowohl über ein höheren Wert von Gain Ratio als auch von Information Gain verfügen. Auf diese Weise bleibt die Konstruktion der Entscheidungsbäume effizient und zugleich einfach.

Alles in allem besteht die Möglichkeit, dass ein allgemeingültiger Filter zur Optimierung der Simulationsstudie für das Produktionssystemmodell entwickelt wird. Mit Hilfe des Filters wird das Ziel verfolgt, die Eingabedaten des Simulationsmodells effizienter und korrekter zu klassifizieren. Dadurch können die nicht erwarteten Eingabedaten von der Simulationsstudie ausgeschlossen werden, womit sich die Anzahl der Simulationsläufe reduzieren lässt. Im folgenden Abschnitt lässt sich ein Konzept für einen Filter anhand der vorliegenden Erkenntnisse erarbeiten und validieren.

4 Entwurf eines Filters anhand des Entscheidungsbaumes

Im diesen Abschnitt wird zuerst ein Filter mit Hilfe der gewonnenen Erkenntnisse auf Grundlage der Datensätzen und der C4.5 Entscheidungsbäume entwickelt. Anschließend muss der entwickelte Filter validiert werden, um seine Anwendbarkeit zu gewährleisten. Die Zielsetzung dieses Konzepts besteht darin, dass die Entscheidungsbaum-basierten Optimierungsverfahren mit dem Simulationsmodell des Produktionssystems verknüpft werden können. Damit kann die Anzahl der Simulationsläufe minimiert werden ohne die Simulationsqualität zu beeinträchtigen.

4.1 Verknüpfung von Filter und Simulationsmodell

Um das Produktionssystemen zu optimieren, kommt derzeit vermehrt die simulationsbasierte Optimierung zur Anwendung. In diesem Abschnitt werden die Optimierungsmethoden mit dem Simulationsmodell verknüpft. Durch die Optimierungsmethode werden die Parameter der Simulation gezielt verändert, um die besten Lösungen zu finden. Der zentrale Bestandteil der Optimierung ist ein Filter, der auf Entscheidungsbäumen basiert. Durch den Filter werden die von der Optimierung vorgeschlagenen Parameter geprüft, sodass nur Simulationen mit vielversprechenden Ergebnissen durchgeführt werden. Dadurch wird das Ziel verfolgt, die Anzahl der Simulationsläufe zu reduzieren ohne dabei zugleich die Qualität der Simulationsergebnisse zu reduzieren. Der Entwurf des Filters bzw. seine Optimierung hängt wiederum vom „Learning“ auf Grundlage der Ausgabedaten der Simulationsexperimente ab.

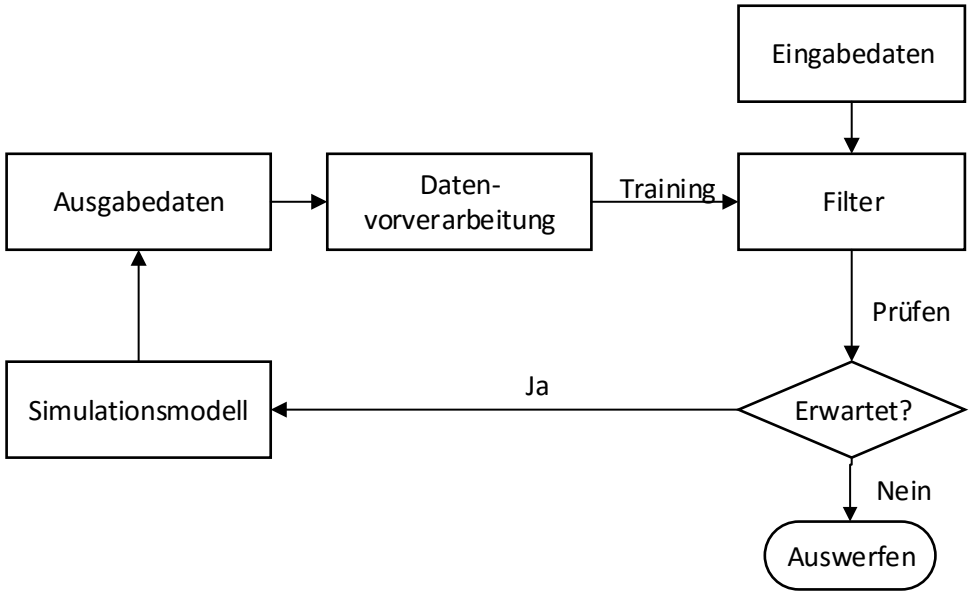


Abbildung 4-1: Verknüpfung der Filter und Simulationsmodell

Die Abbildung 4-1 illustriert die Verknüpfung des Filters mit dem Simulationsmodell. Nach Untersuchung der Beispieldaten aus dem Simulationsmodell im Abschnitt 3.1 ist ersichtlich, dass die Phase der Datenvorverarbeitung für die Entwicklung eines Entscheidungsbaums mit guter

Performance von großer Bedeutung ist. Daher wird die Phase der Datenvorverarbeitung in das Optimierungsverfahren integriert. Der Filter wird auf Basis der gewonnenen Erkenntnisse durch die Entscheidungsbäume in Abhängigkeit der behandelten Ausgabedaten entwickelt. Nach der erfolgreichen Kopplung des Simulationsmodells und des Filter werden die Eingabedaten der Simulationsexperimente zuerst durch den Filter überprüft. Werden die Eingabedaten durch den Filter als „erwartet“ eingeschätzt, werden diese Daten zur Simulation freigegeben, womit die Simulationsexperimente durchgeführt werden können. Andernfalls werden die Eingabedaten ausgeschlossen und die Simulation findet nicht statt. Auf diese Weise können Teile der Eingabedaten herausgefiltert und die nicht erwarteten Simulationsläufe vermieden werden, was die Anzahl der Simulationsläufe insgesamt minimiert.

Weiterhin kann das Optimierungsverfahren und das Simulationsmodell als ein Gesamtsystem betrachtet werden. Um die Performance des Gesamtsystems zu untersuchen, kann wiederum die Kosten-Nutzen-Analyse eingesetzt werden.

4.2 Entwurf eines Konzepts

In diesem Abschnitt wird zuerst die Vorbereitungsphase wie die Datenvorverarbeitung für den Entwurf des Filters erläutern. Im Anschluss daran wird ein Verfahren zur Entwicklung des Filters erarbeitet. Dabei wird insbesondere die Attributauswahl für die Generierung der Entscheidungsbäume erklärt. Weiterhin ist die Untersuchung der Anzahl an Trainingsdaten von Bedeutung, da die Korrektheit und Effizienz der Entscheidungsbäume vor allem von der Menge der Trainingsdaten abhängt. Abschließend soll der Filter bewertet werden, um sowohl Stärken und Schwächen als auch die weiteren Entwicklungsmöglichkeit des Filters zu untersuchen.

4.2.1 Vorbereitung vor dem Filterentwurf

Die Ausgabedaten des Simulationsmodells beinhalten Parameter, die für die Generierung der Entscheidungsbäume irrelevant sind oder die Qualität der Entscheidungsbäume sogar negativ beeinflussen können. Daher lassen sich die Ausgabedaten mit den Methoden der Datenvorverarbeitung vor Generierung der Entscheidungsbäume überprüfen und verarbeiten. Das Ziel der Datenvorverarbeitung besteht darin, einerseits die besten Attribute für den Baumaufbau auszuwählen. Andererseits können die Daten in ein passendes Format umgewandelt werden. Der Ablauf der Datenvorverarbeitung wird in Abbildung 4-2 dargestellt, dabei werden die Methoden „Datenintegration“, „Datenbereinigung“, „Datenreduktion“ sowie „Datentransformation“, insbesondere „Nominalisierung“, verwendet (vgl. 2.3.1). Nach der Vorverarbeitung werden die Daten zur Generierung und Evaluation der Entscheidungsbäume genutzt.

Im Folgenden werden die Verfahren der Datenvorverarbeitung vor dem Hintergrund der gewonnenen Erkenntnisse aus den gegebenen Datensätzen detailliert erläutert.

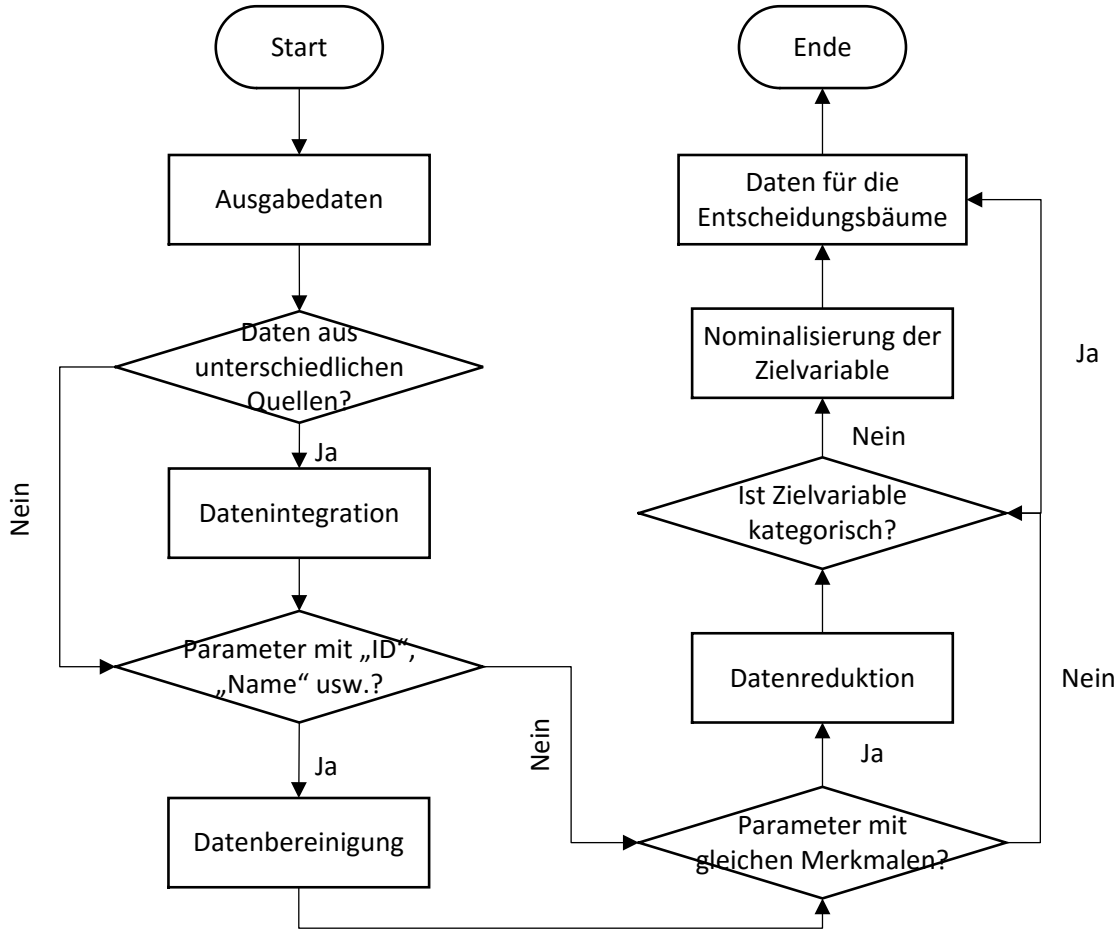


Abbildung 4-2: Ablauf der Datenvorverarbeitung

Zunächst sollen die Ausgabedaten in einer Tabelle integriert werden, falls die Simulationsdaten in unterschiedlichen Tabellen gesammelt und gespeichert wurden. Damit kann die Redundanz und Widersprüchlichkeit der Datensätze reduziert werden (vgl. 2.3.1). Mit Hilfe der Datenintegration kann sowohl die Korrektheit der Entscheidungsbäume als auch die Effizienz der Baumgenerierung verbessert werden.

Anschließend sind die Methoden der „*Datenbereinigung*“ für das Entfernen der irrelevanten Attribute anzuwenden (vgl. 2.2.1). Wenn die Attribute die Bezeichnungen wie bspw. „ID“, „Name“ usw. beinhalten, sollten diese Attribute im Rahmen der Datenbereinigung entfernt werden, da diese Attribute sich für die Baumgenerierung nicht anwenden lassen. Weiterhin müssen diese Attribute entfernt werden, wenn die Attribute zugleich die Simulationsergebnisse wie bspw. „Stop“ und „Delay“ im Datensatz 2 sind. Diese Attribute können die Analysen durch die Entscheidungsbäume verfälschen. Durch die Datenvorverarbeitung und die die entsprechende Bereinigung der Daten kann die Effizienz und Korrektheit der Entscheidungsbäume erhöht werden.

Anschließend werden die Daten mit Hilfe der „*Datenreduktion*“ verarbeitet, womit die Datenmenge und die Komplexität der Datensätze reduziert werden können, ohne die Vollständigkeit der Datensätze zu beeinträchtigen (vgl. 2.3.1). Beispielsweise verfügen alle Attribute mit der (Teil-)Bezeichnung „DelayTime“ und „BaseFiringTime“ im Datensatz 1 über die gleichen Werte, weshalb diese Attribute durch die Attribute „DelayTime“ und „BaseFiringTime“ ersetzt wurden. Der behandelte Datensatz kann somit auch die Merkmale des originalen Datensatzes widerspiegeln. Auch auf diese Weise lässt sich einerseits die Effizienz der Baumgenerierung verbessern und andererseits die Komplexität der Entscheidungsbäume reduzieren.

Weiterhin sollten die Attribute mittels der Methode „*Datentransformation*“ in das für die Konstruktion der Entscheidungsbäume erforderliche Format umgewandelt werden. Hierbei sollte vor allem die numerische Zielvariable im Rahmen der Datentransformation in eine kategorische Zielvariable umgewandelt werden, da die Generierung der Entscheidungsbäume mittels C4.5 Algorithmus nur mit Hilfe der kategorischen Zielvariable erfolgen kann (vgl. 2.3.2). Somit können die Klassifikationsprobleme mit Hilfe der kategorischen Zielvariable klarer formuliert werden und effizienter gelöst werden.

Im Rahmen der Datenvorverarbeitung werden die Ausgabedaten zur Konstruktion der Entscheidungsbäume durch C4.5 mit Hilfe der erläuterten Methoden behandelt. Die Datenvorverarbeitung wird als eine Phase der Optimierung betrachtet, da diese für die Effizienz und Komplexität der Entscheidungsbäume von wesentlicher Bedeutung ist.

4.2.2 Entwurf des Filters

Nach der Datenvorverarbeitung wird in diesem Abschnitt ein Konzept des Filterentwurfs vorgestellt. Die behandelten Ausgabedaten des Simulationsmodells werden als Eingabedaten sowohl zur Konstruktion als auch zur Evaluation der Entscheidungsbäume eingesetzt. Der zentrale Bestandteil des Filters besteht aus dem C4.5 Entscheidungsbaum und dem damit generierten Klassifikator. Der Arbeitsprozess des Filters wird in Abbildung 4-3 illustriert. Hierbei wird der Ablauf vor allem in die 3 Phasen Konstruktion, Evaluation und Anwendung der Entscheidungsbäume unterteilt. Zum Abbruch der Suche nach einem weiteren, genaueren Entscheidungsbaum werden zwei Regeln angegeben, nämlich „Auslösen des Schwellen-werts“ und „Erreichen der Korrektheit“. Dadurch kann nicht nur die Korrektheit, sondern auch die Effizienz des Filters sichergestellt werden. Im Folgenden wird der Arbeitsprozess des Filters systematisch erläutert.

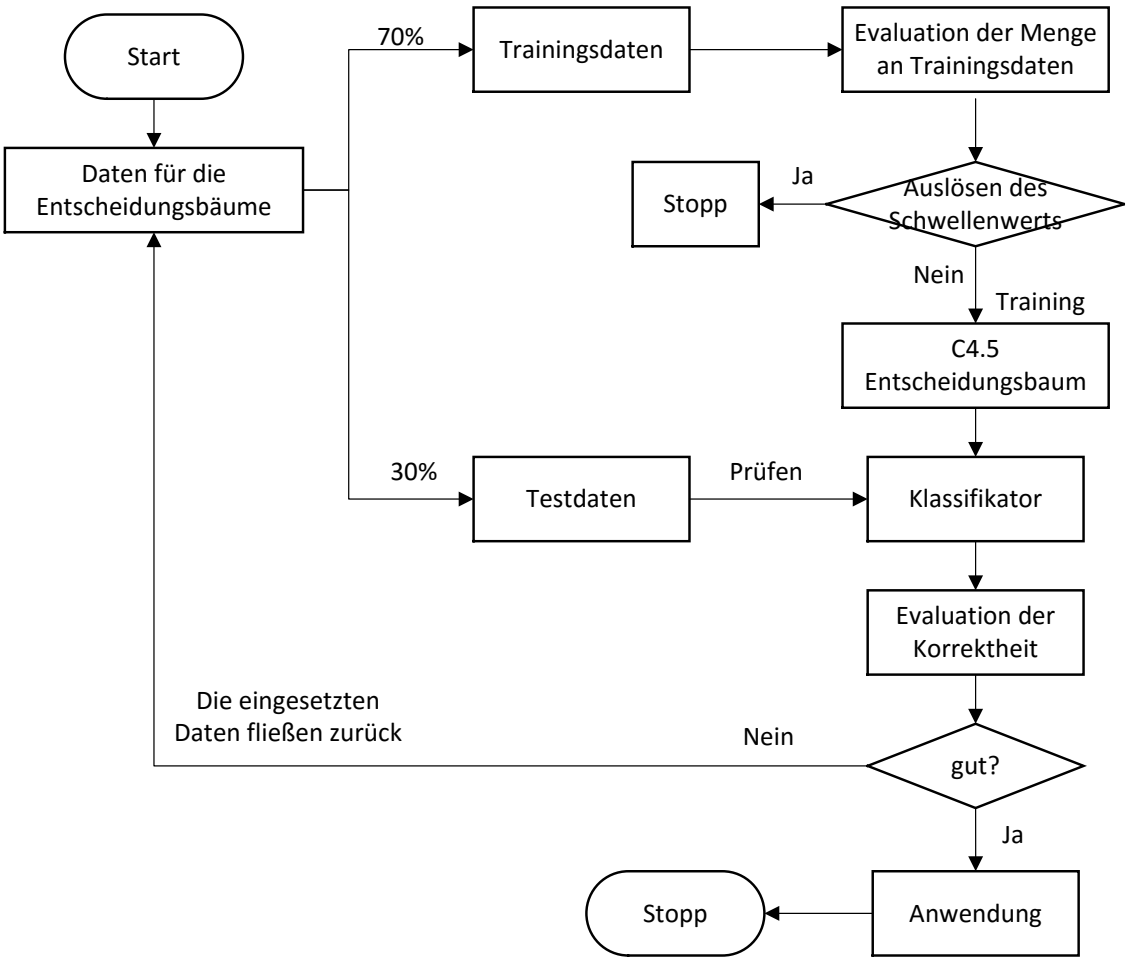


Abbildung 4-3: Entwurf des Filters durch den C4.5 Entscheidungsbaum

Die im Rahmen der Datenvorverarbeitung behandelten Ausgabedaten werden mit jeweils 70% und 30% in Trainings- und Testdaten unterteilt. Dabei kommen die Trainingsdaten zur Generierung der Entscheidungsbäume mittels C4.5 zum Einsatz. Die Testdaten werden zur Überprüfung der Qualität des entwickelten Klassifikators verwendet. Mit Hilfe des Filters versucht man die Eingabedaten des Simulationsmodells sowohl genau, als auch effizient zu verwerten. Die Analyse der Ausgabedaten des Simulationsmodells sollte generell als ein dynamischer Prozess betrachtet werden, da sich der Klassifikator und die Regeln durch den Entscheidungsbaum sowohl in Abhängigkeit der Art, als auch der Menge der Trainingsdaten verändern. Darüber hinaus wird die Effizienz der Generierung der Entscheidungsbäume mit der zunehmenden Datenmenge reduziert (vgl. 3.3.2). Daher sollte ein Schwellenwert zum Abbruch der Suche nach einem weiteren, genaueren Entscheidungsbaum angegeben werden. Einerseits wird die Phase der Datenvorverarbeitung und Baumgenerierung gestoppt, wenn die erwartete Korrektheit des Modells innerhalb des Schellenwerts erreicht werden kann. Andererseits wird die Phase der Datenvorverarbeitung und der Baumgenerierung abgebrochen, wenn der in Bezug auf die Menge der Trainingsdaten angegebene Schwellenwert ausgelöst wird. Der Schwellenwert wird nicht überschritten, wenn die erwartete Korrektheit innerhalb der Evaluation nicht erreicht werden kann. Dies kann anhand einer Kosten-Nutzen-Analyse des Gesamtsystems geschehen.

Nach der Generierung des Entscheidungsbaums mittels C4.5 kann ein entsprechender Klassifikator entwickelt werden. Jedoch sollte dieser Klassifikator durch die Testdaten überprüft werden, bevor er zur Klassifizierung der Eingabedaten für das Simulationsmodell eingesetzt wird. Hierbei kommen in der Regel die Kriterien Korrektheit und Effizienz zur Anwendung. Jedoch sollte man auch die Situation, in der die erwartete Korrektheit nicht erreicht werden kann, in Betracht ziehen. Nach der Untersuchung der Entscheidungsbäume in Bezug auf die Beispieldaten im Abschnitt 2.3.2 lässt sich herausstellen, dass die Korrektheit und die Menge der Trainingsdaten eine positive Korrelation aufweisen. Um die erwartete Korrektheit zu erreichen, wird versucht, dass das Simulationsmodell und das Optimierungs-verfahren unaufhörlich durchlaufen werden. Einerseits sollten die Simulationsexperimente zur Generierung der Ausgabedaten wiederholt ausgeführt werden andererseits sollte auch die Datenvorverarbeitung und die Konstruktion der Entscheidungsbäume wiederholt durchgeführt werden. Dies verursacht in der Regel sehr hohe Kosten für das Gesamtsystem. Daher sollte man geeignete Maßnahmen treffen, um diese Situation zu vermeiden.

Allerdings kann der Wert der Korrektheit vom Anwender neu angegeben werden, um den Ablauf des Konstruktionsverfahrens zu stoppen. In diesem Fall ist eine niedrigere Korrektheit des Entscheidungsbaums zu erwarten. Insgesamt ist es schwierig, eine geeignete Korrektheit für den Entscheidungsbaum zu finden, da sich die Korrektheit in Abhängigkeit der Art und Menge der Ausgabedaten verändert. Daher wird ein Schwellenwert, der als die Grenze der Menge an Trainingsdaten innerhalb der Evaluation betrachtet wird, zur Einschränkung der Suche der Entscheidungsbaumkonstruktion definiert.

Wenn der durch den Entscheidungsbaum generierte Klassifikator innerhalb des gegebenen Schwellenwerts die erwartete Korrektheit und Effizienz erreicht, wird dieser Klassifikator als Filter zur Optimierung der Eingabedaten eingesetzt. Ansonsten fließen die eingesetzten Trainings- und Testdaten zurück zum Ausgangspunkt „Daten für die Entscheidungsbäume“ und können ohne Datenvorverarbeitung unmittelbar zur Baumgenerierung angewandt werden. Auf diese Weise kann die Effizienz des Gesamtsystems verbessert werden.

Anderenfalls könnte ein Schwellenwert zum Abbruch der Suche nach einem passenden Entscheidungsbaum mit der erwarteten Korrektheit angegeben werden, wenn die gegebene Korrektheit innerhalb der bestimmten Menge von Trainingsdaten nicht erreicht werden kann. Die zunehmende Menge der Trainingsdaten führt nicht nur zu einer Verbesserung der Korrektheit des Modells, sondern auch zu einer Steigerung der Kosten des Gesamtsystems. Daher kann die Kosten-Nutzen-Analyse für das Gesamtsystem eingeführt werden, um die Bedeutsamkeit des Schwellenwerts der Menge an Trainingsdaten zu verdeutlichen. Bei der Kosten-Nutzen-Analyse wird angenommen, dass nur das Attribut „Menge der Trainingsdaten“ als unabhängiges Attribut untersucht wird, die Kosten und Nutzen hängen von der Menge der Trainingsdaten ab. Weiterhin gelten die anderen Attribute als unveränderlich.

Zunächst stammen alle Daten aus dem Simulationsmodell. Um eine Menge an Ausgabedaten zu sammeln, müssen die Simulationsexperimente mit einer entsprechenden Häufigkeit durch-

geführt werden. Im Rahmen der Simulation weist jedes Experiment einen bestimmten Kostenaufwand auf. Wenn die anderen Attribute unveränderlich sind, führt die zunehmende Anzahl der Simulationsläufe in der Regel zu einer Steigerung der Kosten für das Simulations-system, d.h. je mehr Simulationsexperimente ausgeführt werden, desto höher sind die Kosten. Hierbei wird die Kosten der Simulation als K_1 definiert.

Anschließend werden die Ausgabedaten im Rahmen der Datenvorverarbeitung behandelt. Je mehr die Daten vorverarbeitet werden sollen, desto länger dauert die Vorverarbeitungszeit und desto höher sind die Kosten der Datenvorverarbeitung. Diese Kosten werden als K_2 definiert. Weiterhin steigt die Laufzeit der Baumgenerierung mittels C4.5 mit zunehmender Menge an Trainingsdaten an. Bei längerer Laufzeit wird die Effizienz der Baumgenerierung reduziert, was zu höheren Kosten des Optimierungsverfahrens führt. Diese Kosten werden als K_3 definiert.

Außerdem verursacht die falsche Klassifikation der simulierbaren Instanzen ebenfalls einzurechnende Kosten für das Gesamtsystem. Diese Kosten wurden schon im Abschnitt 3.3.3 untersucht und als Verlustkosten eingeordnet. Sie werden hier mit K_4 bezeichnet.

Die möglichen Kosten des Gesamtsystems „K“, die von der Menge an Trainingsdaten abhängig sind, lassen sich somit vereinfacht darstellen. Die Gesamtkosten stellen sich als $K = K_1 + K_2 + K_3 + K_4$ dar.

Im Anschluss daran kann der Nutzen des Gesamtsystems erläutert werden. Der Nutzen des Gesamtsystems resultiert aus der erfolgreichen Reduktion der Anzahl der Simulationsläufe. Die Reduktion ist dabei wiederum von der Klassifikation der Eingabedaten abhängig. Der Nutzen ist identisch wie der Nutzen, der schon im Abschnitt 3.3.3 beschrieben wurde. Im Unterschied zum genannten Abschnitt wird nun der Nutzen des Gesamtsystems betrachtet. Der Nutzen wird hierbei als *N* definiert.

Durch die Analyse des Simulations- und Optimierungssystems und die Annahme der relevanten Attribute kann der Gewinn (G) des Gesamtsystems als Differenz zwischen Nutzen (N) und Kosten (K) definiert werden. Es soll noch einmal die Annahme betont werden, dass der Nutzen und die Kosten des Gesamtsystems in dieser Darstellung nur von der Menge der Trainingsdaten abhängig sind.

Im Sinne einer Optimierung kann die Zielfunktion hierbei als $Max\ G = Max\ (N - K)$ beschrieben werden. Hierbei sollten $N > 0, C > 0$ sein. Anhand dieser Zielfunktion lässt sich das Ziel verfolgen, einen maximalen Gewinn des Gesamtsystems durch die Integration des Optimierungssystems zu erreichen. Wenn der Gewinn des Systems negativ ist, wird das Optimierungsverfahren als ungültig betrachtet.

Beispielsweise, könnte nach der Untersuchung der Beispieldaten von dem Simulationsmodell in Abschnitt 3.3.2 angenommen werden, dass der Schwellenwert der Menge an Trainingsdaten mit 3000 angegeben ist. Mit Hilfe der 3000 Trainingsdaten weisen die generierten Entschei-

dungsbäume sowohl eine höhere Korrektheit als auch eine höhere Effizienz auf als andere Trainingsdatenmengen.

Wenn die erwartete Korrektheit der Entscheidungsbäume innerhalb der Schwellenwertmenge nicht erreicht werden kann, gibt es auch eine andere Möglichkeit einen Entscheidungsbaum in Abhängigkeit der angegebenen Schwellenwertmenge zu generieren. Jedoch muss der generierte Entscheidungsbaum dabei vom Anwender bewertet werden. Wenn die Korrektheit nach der Auswertung vom Anwender als akzeptierbar eingestuft wird, kann der Entscheidungsbaum ebenfalls zur Klassifikation der Eingabedaten für das Simulationsmodell eingesetzt werden. Ansonsten wird das Optimierungsverfahren nicht angewandt, da die Ungenauigkeit des Modells möglichweise höhere Kosten der Simulation und Optimierung verursacht.

Die Generierung der Entscheidungsbäume wird durch den C4.5 Algorithmus (vgl. 2.3.2, Algorithmus 3) mit Bezug auf die Trainingsdaten durchgeführt. Um die Entscheidungsbäume effizient und einfach zu generieren, sollten die Attribute anhand der erläuterten Kriterien ausgewählt werden. Mit Blick auf die Untersuchung der Ergebnisse aus den vorgegebenen Datensätzen und unter Berücksichtigung der relevanten Forschungsergebnisse lässt sich das Verfahren der Attributauswahl in Abbildung 4-4 darstellen (vgl.2.3.2 und 3.3.2).

Zunächst werden Information Gain und Gain Ratio der Trainingsdaten mittels der Formel 2-3 und 2-4 berechnet. Danach kann die Bedeutsamkeit der Attribute bestimmt werden. Da Gain Ratio als Auswahlkriterium bei C4.5 angewandt wird, sollten in der Regel nur die Attribute mit höheren Werten von Gain Ratio für die Baumgenerierung eingesetzt werden. Jedoch können auch die für die Baumgenerierung irrelevanten Attribute manchmal einen höheren Gain Ratio Wert besitzen. Daher sollte man auch den Wert Information Gain der Attribute bei der Auswahl in Betracht ziehen (vgl. 2.3.2). Nach der Untersuchung des Datensatzes 2 in Abschnitt 3.3.2 wurde festgehalten, dass die Attribute mit überdurchschnittlichen Werten der Information Gain für die Baumkonstruktion ausgewählt werden sollten. Anschließend lassen sich die Attribute anhand des Wertes Gain Ratio vergleichen und zur Generierung der Entscheidungsbäume verwenden. Auf diese Weise kann nicht nur die Korrektheit des Klassifikators, sondern auch die Effizienz der Baumgenerierung sichergestellt werden. Weiterhin reduziert sich zugleich die Komplexität der Entscheidungsbäume mit einer geringeren Anzahl an Attributen, da auch die Anzahl der Blätter und Knoten der generierten Entscheidungsbäume geringer wird (vgl. 3.3.2). Durch Beachtung der geschilderten Umstände kann ein einfacher und effizienter Entscheidungsbaum entwickelt werden.

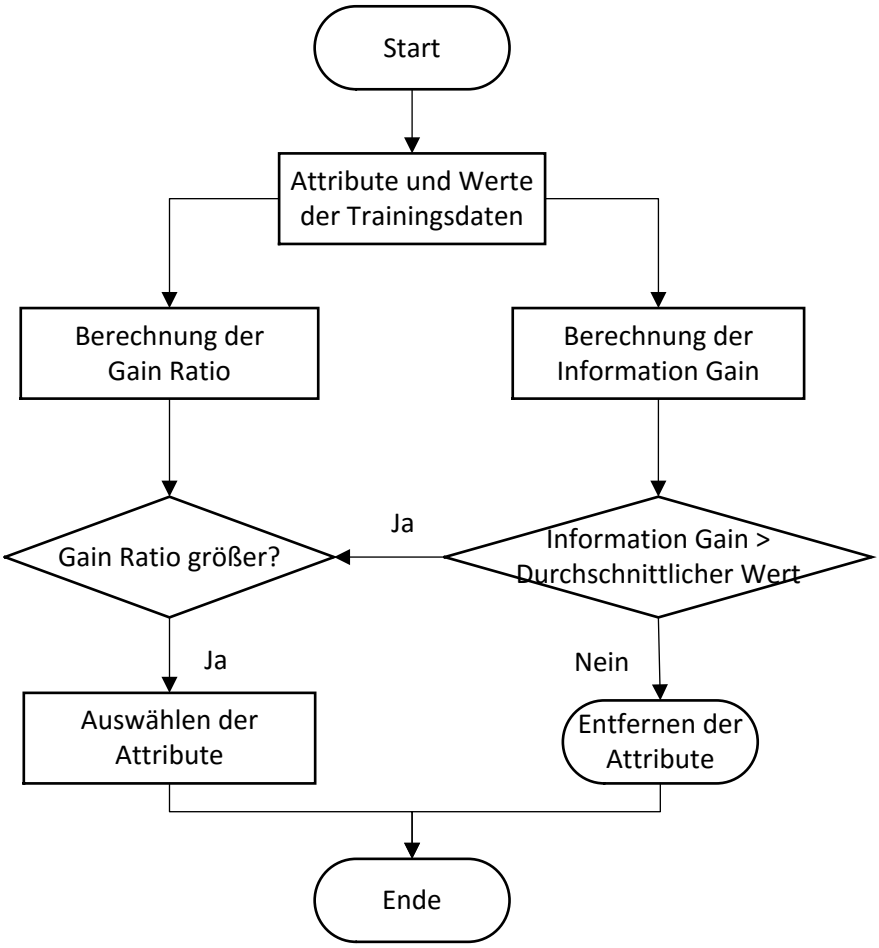


Abbildung 4-4: Auswahl der Attribute für die Entscheidungsbäume

Die für die Optimierung des Simulationsmodells relevanten Attribute könnten innerhalb der Attributauswahl identifiziert werden, da die ausgewählten Attribute in der Regel ein engen Zusammenhang mit den Simulationsergebnissen aufweisen. Beispielsweise spielt im Datensatz 1 das Attribut „Count“ eine bedeutende Rolle für die Generierung eines korrekten und einfachen Entscheidungsbaums. Hier wird nur das Attribut „Count“ analysiert, um das Simulationsmodell zu untersuchen. Außer der Identifikation der einflussreichen Attribute mittels Information Gain und Gain Ratio gibt es auch die Möglichkeit, diese Attribute durch den generierten Entscheidungsbaum zu identifizieren, da sich die einflussreichen Attribute in der Regel in der Spitze des Entscheidungsbaums befinden.

Insgesamt hat die Untersuchung des Arbeitsablaufs des Filters gezeigt, dass sowohl die Anwendbarkeit des Konzepts als auch ein Gewinn des Gesamtsystems ermöglicht werden sollte. Hierbei sollte eine Endlosschleife des Ablaufs entweder durch die Einstellung eines Schwellenwerts oder durch die manuelle Eingabe einer erwarteten Korrektheit vermieden werden. Drüber hinaus sollten die Kosten des Simulationsmodell und des Optimierungsverfahren, die jeweils von der Menge der Trainingsdaten abhängig sind, durch die Angabe eines Schwellenwerts minimiert werden. Auf diese Weise ist die Gültigkeit der simulationsbasierten Optimierung anhand des C4.5 Algorithmus sichergestellt.

Nach der Entwicklung des Konzepts sollte das Konzept verifiziert werden, damit die Korrektheit des Konzepts gewährleistet ist. Da das Konzept auf den gewonnenen Erkenntnissen durch die Untersuchung der Beispieldatensätzen basiert, kann festgehalten werden, dass dieses Konzept zur Analyse der Ausgabedaten als korrekt und anwendbar beurteilt wird. Außerdem wird eine Endlosschleife des Ablaufs durch die Angabe eines Schwellenwerts innerhalb der Konstruktion und Evaluation der Entscheidungsbäume vermieden.

4.3 Validierung des entwickelten Konzepts

In diesem Abschnitt wird das entwickelte Konzept validiert, um das Konzept zur Optimierung der Simulationsstudie erfolgreich anwenden zu können. Da das Konzept aus den Komponenten „Datenvorverarbeitung“ und „Filter mittels C4.5 Entscheidungsbaum“ besteht, sollten im Rahmen der Validierung die Anforderungen an jede Komponente untersucht werden. Hierbei sollte die Validierung die kombinierten Anforderungen an die simulationsbasierte Optimierung, die Datenvorverarbeitung sowie an die Entscheidungsbäume berücksichtigen. Anschließend lässt sich die Kopplung der Simulation und der Optimierung dahingehend überprüfen, ob die entsprechenden Anforderungen erfüllt sind. Schließlich sollen die Stärken und Schwächen des Konzepts systematisch untersucht werden.

Vor der Validierung sollen zuerst die relevanten Anforderungen aus den vorliegenden Untersuchungen an die einzelnen Komponenten zusammengefasst werden. Dadurch kann die Validierung erfolgreich durchgeführt werden. Dabei sollte der Filter nicht nur die Anforderungen an die Entscheidungsbäume nach Han (2012), sondern auch die Anforderungen an die simulationsbasierte Optimierung nach Fu (2002) erfüllen (vgl.2.2.4 und 2.3.4). Darüber hinaus soll die Komponente der Datenvorverarbeitung vor allem hinsichtlich der Anforderungen wie Effizienz, Transparenz, Anwendbarkeit sowie Vollständigkeit untersucht werden.

Nach der Analyse der Anforderungen an die einzelnen Komponenten sollen auch die Anforderungen an die Integration des Simulationsmodells untersucht werden. Die Integration soll sowohl die Anforderungen der Kopplung von Simulation und Optimierung nach Fu (2002) als auch die Anforderungen mit Blick auf die Kosten-Nutzen-Analyse des Gesamtsystems erfüllen.

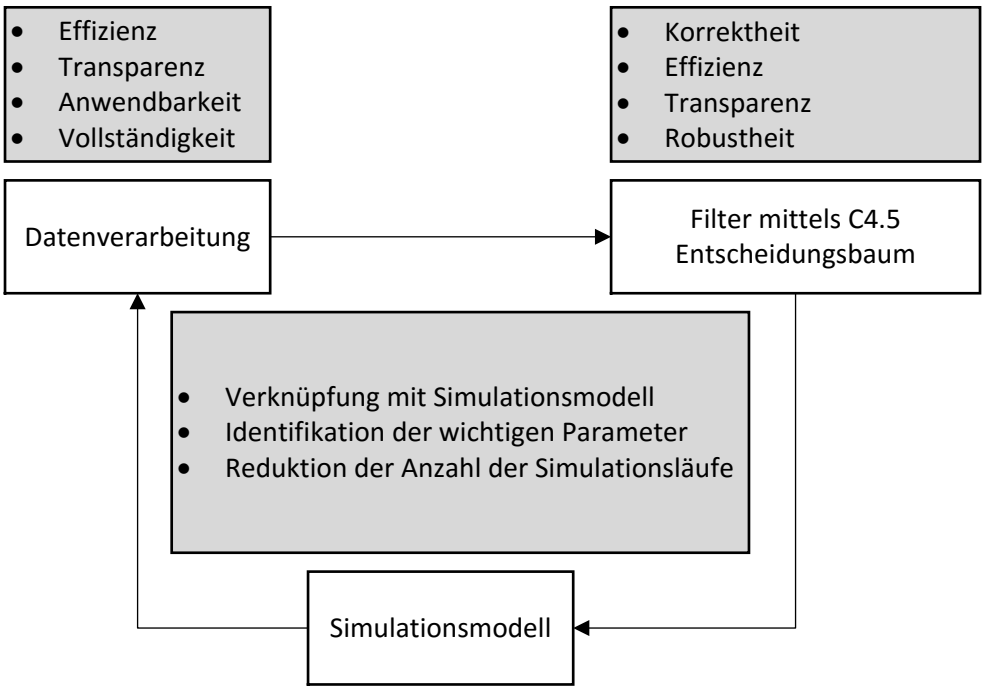


Abbildung 4-5: Anforderungen an das Konzept

Die Abbildung 4-5 illustriert die Zusammenfassung der Anforderungen an das Gesamtkonzept. Im Folgenden werden die entsprechenden Anforderungen an die einzelnen Komponenten sowie die Kopplung der Simulation und Optimierung detailliert erläutert.

4.3.1 Anforderungen an die Komponenten

Zuerst werden die Anforderungen an die Datenvorverarbeitung für die Konstruktion der Entscheidungsbäume erläutert. Die Datenvorverarbeitung spielt eine bedeutende Rolle für die Generierung der Entscheidungsbäume mit hoher Korrektheit und Effizienz sowie niedriger Komplexität. Die Komponente der Datenvorverarbeitung soll daher anhand der Anforderungen Effizienz, Transparenz und Anwendbarkeit im Folgenden untersucht werden.

4.3.1.1 Anforderungen an Datenvorverarbeitung

Effizienz

Im Rahmen der Datenvorverarbeitung werden die Ausgabedaten zur Generierung der Entscheidungsbäume mittels verschiedener Methoden verarbeitet (vgl. 4.2.1). Im Abschnitt 4.2.1 wurde der Datenvorverarbeitungsablauf unter Berücksichtigung der gewonnenen Erkenntnisse vorgestellt. Anhand dieses Ablaufs können die Daten für die Entscheidungsbäume schrittweise verarbeitet werden. Somit können Verschwendungen durch wiederholte Operationen vermieden werden. Der schrittweise Prozess gewährleistet sowohl die Effizienz, als auch die Gültigkeit der Datenvorverarbeitung.

Bespielweise lässt sich mit Hilfe der Methode „Daten Bereinigung“ und „Reduktion“ die Menge der relevanten Attribute erheblich reduzieren. Dadurch kann die Effizienz der Konstruktion der Entscheidungsbäume wesentlich verbessert werden.

Transparenz

Mit Hilfe der Methode der Datenvorverarbeitung wird dem Anwender ersichtlich, welche Daten, auf welche Weise und aus welchem Grund verarbeitet werden sollen. Beispielsweise muss das Attribut „OderID“ aus dem Datensatz 1 nach der Analyse entfernt werden, denn es verfälscht die Konstruktion der Entscheidungsbäume. Darüber hinaus werden keine komplizierten mathematischen Verfahren im Rahmen der Datenvorverarbeitung eingesetzt, womit sich mathematische Hemmnisse für die Anwender vermeiden lassen.

Anwendbarkeit und Vollständigkeit

Die Anforderung an die Anwendbarkeit fokussiert sich vor allem auf die erfolgreiche Generierung und Evaluation der Entscheidungsbäume in Abhängigkeit der verarbeiteten Daten. Insbesondere können die Entscheidungsbäume mittels C4.5 Algorithmus nur in Abhängigkeit der kategorischen Zielvariable entwickelt werden. Daher muss die Zielvariable mit Hilfe der Methoden „Diskretisierung“ und „Nominierung“ entwickelt werden. Neben der Anwendbarkeit sollte die Datenvorverarbeitung auch die Anforderung der Vollständigkeit erfüllen. Nach der Datenvorverarbeitung müssen die Daten alle relevanten Informationen der originalen Daten umfassen, d.h. im Rahmen der Datenvorverarbeitung sollen die relevanten Informationen der Daten nicht verändert oder eliminiert werden.

4.3.1.2 Anforderungen an den Filter mittels C4.5 Entscheidungsbaum

Im Rahmen der Validierung des Filters mittels C4.5 Entscheidungsbaum sollten die folgenden Anforderungen, die sowohl aus den Entscheidungsbäumen als auch der simulationsbasierten Optimierung abzuleiten sind, erfüllt sein.

Korrektheit

Als eines der wichtigsten Kriterien ist die Korrektheit zur Auswertung und Weiteranwendung der Entscheidungsbäume zu nennen. In der Regel erfordert die Simulationsstudie eine hohe Korrektheit zur Klassifikation der Eingabedaten. Die falsche Klassifikation der akzeptablen Eingabedaten verursacht höhere Kosten für das System. Die falsche Klassifikation der nicht akzeptierbaren Eingabedaten führt zugleich zu einer Reduktion des Nutzens des Systems. Nach der Untersuchung im Abschnitt 3.3.3 spielte die Korrektheit zudem eine wichtige Rolle im Sinne einer Erhöhung des Gewinns.

Nach dem Vergleich der Algorithmen in Abhängigkeit der Ausgabedaten weist der gewählte C4.5 Algorithmus eine höhere Korrektheit als die anderen Algorithmen auf (vgl. 3.2.3). Daher wurde der C4.5 Algorithmus zu weiterer Untersuchung angewandt, sodass der C4.5 Algorithmus auch für den Entwurf des Filters zum Einsatz kam.

Effizienz

Zur Überprüfung der Effizienz des C4.5 Algorithmus sollten sowohl die Anforderungen an die Entscheidungsbäume, als auch die Anforderungen an die Optimierungsmethoden in Betracht gezogen werden. Im Vergleich zu anderen Algorithmen dauert die Generierung der Entschei-

dungsbäume mittels C4.5 Algorithmus kürzer (vgl. 3.2.3). Die Entwicklung der Entscheidungsbäume mittels C4.5 weist somit eine höhere Effizienz als die anderen Algorithmen auf. Weiterhin verfügt der C4.5 Algorithmus über das Pruningverfahren, womit die irrelevanten Knoten und Blätter nach der Baumgenerierung abgeschnitten werden können. Dadurch ist es möglich, einen fein entwickelten Baum zur Analyse der Entscheidungen zu generieren. Darüber hinaus werden die Attribute nach ihrer Bedeutsamkeit anhand der Werte Information Gain und Gain Ratio für die Baumgenerierung ausgewählt. So können die unnötigen Attribute herausgefiltert und zugleich kann die Komplexität der Entscheidungsbäume reduziert werden.

Die Konstruktion eines kleinen Entscheidungsbaums kann nicht nur die Laufzeit der Baumgenerierung sondern auch die Laufzeit des Evaluationsprozesses verkürzen. Dadurch kann die Effizienz des Gesamtsystems verbessert werden. Jedoch ist der Zusammenhang zwischen Effizienz und Korrektheit des Gesamtsystems sehr komplex. Wenn beide Kennzahlen nur von der Menge der Trainingsdaten, wie bisher unterstellt, abhängig sind, weisen beide eine negative Korrelation auf. Daher wurde in diesem Konzept die Angabe eines Schwellenwerts zum Abbruch der Suche nach Entscheidungsbäumen mit höherer Korrektheit vorgeschlagen. Auf diese Weise lässt sich die Effizienz des Systems sicherstellen, was zugleich einen Gewinn des Gesamtsystems ermöglicht. Außerdem kann die Angabe eines Schwellen-werts die beschriebenen Endlosschleifen des Systems vermeiden.

Transparenz

Zunächst ist der C4.5 Algorithmus vom Anwender einfach zu verstehen und stellt keine hohen mathematischen Ansprüche an den Anwender. Zudem kann der Anwender auch nach eigenen Vorstellungen die Attribute der Baumgenerierung, beispielsweise die minimale Anzahl der Instanzen pro Blatt, verändern. Nach Han (2012) ist auch die Interpretierbarkeit eine wichtige Anforderung an die Entscheidungsbäume. Unter Interpretierbarkeit versteht man in diesem Zusammenhang, dass die Ergebnisse aus dem Modell für den Anwender einfach zu verstehen sind. Die Anforderung der Interpretierbarkeit wird in dieser Arbeit vor allem durch die Transparenzanforderung in Bezug auf den Anwender repräsentiert. Mit Hilfe der generierten Entscheidungsbäume können die erforderlichen Regeln entlang der Baumstruktur von den Blättern zu den Wurzeln einfach nachvollzogen werden. Weiterhin werden in den Knoten alle eingesetzten Attribute dargestellt, womit die Ergebnisse einfach erkannt und dem-entsprechend untersucht werden können.

Robustheit

Unter der Anforderung der Robustheit versteht man, dass der Algorithmus trotz fehlender oder verrutschter Daten erfolgreich ausgeführt werden kann (vgl. 2.3.4). Mit Blick auf die Erläuterung der Vorteile des C4.5 Algorithmus im Abschnitt 2.3.4 kann man festhalten, dass innerhalb des C4.5 Algorithmus fehlende oder verrutschte Daten behandelt werden.

In der Regel ist die Anzahl der fehlenden oder verrutschten Daten eines Datensatzes sehr gering. Diese Daten können in der Regel nach der Baumgenerierung durch das Pruningverfahren

oder mit Erhöhung des Schwellenwerts zum Abbruch der Baumkonstruktion entfernt werden. Nach Beschreibung des C4.5 Algorithmus im Abschnitt 2.3.2 ist ersichtlich, dass der C4.5 Algorithmus über das Pruningverfahren verfügt. Zugleich kann ein Schwellenwert zum Abbruch der Baumgenerierung vom Anwender angegeben werden. Daher wird abschließend festgestellt, dass der C4.5 Algorithmus die Anforderung der Robustheit erfüllen kann.

Neben der Untersuchung der wichtigsten Anforderungen an die C4.5 Entscheidungsbäume wurden auch die Anforderungen der Allgemeingültigkeit und einer hohen Dimensionalität vereinfacht erläutert. Beide Anforderungen sind für die Auswertung der C4.5 Entscheidungsbäume in dieser Arbeit jedoch nicht relevant. Da mit Hilfe des Entscheidungsbaums in diese Arbeit vor allem Klassifikationsprobleme gelöst werden, wird die Anforderung der Allgemeingültigkeit nicht weiter untersucht. Die Daten mit hoher Dimensionalität werden im Rahmen der Datenvorverarbeitung durch die Methode der Datenreduktion behandelt. Die Generierung der Entscheidungsbäume mittels C4.5 fokussiert sich jedoch vor allem auf Daten mit niedriger Dimensionalität, womit die Effizienz und Korrektheit der Entscheidungsbäume erhöht werden kann.

Nach Validierung der einzelnen Komponente des Konzeptes wird nunmehr die Kopplung des Konzept und des Simulationsmodells untersucht. Durch die Integration des Simulationsmodells kann die Durchführbarkeit des Konzepts sichergestellt werden.

4.3.2 Anforderungen an die Kopplung von Simulation und Konzept

In diesem Abschnitt wird die Kopplung von Konzept und Simulationsmodell validiert. Dazu wird untersucht, wie die erläuterten Anforderungen durch die Kopplung erfüllt werden können. Im Folgenden wird die Validierung vor allem ausgehend von den Aspekten „Verknüpfung mit dem Simulationsmodell“, „Identifikation der wichtigen Parameter“ und „Reduktion der Anzahl der Simulationsläufe“ durchgeführt.

Verknüpfung mit dem Simulationsmodell

Wie vorher mehrmals erläutert, funktioniert das Gesamtkonzept durch die Zusammenarbeit der beiden Komponenten „Datenvorverarbeitung“ und „Filter mittels C4.5 Entscheidungsbaum“. In diesem Zusammenhang wurde die Durchführbarkeit der Verknüpfung des entwickelten Konzepts mit dem Simulationsmodell insbesondere in Abschnitt 4.2.2 untersucht. Das Ziel der Untersuchung bestand vor allem darin, weitere Anwendungsmöglichkeiten des Konzepts sicherzustellen.

Die Ausgabedaten des Simulationsmodells können unmittelbar durch die Methode der Datenvorverarbeitung behandelt werden. Anschließend können die verarbeiteten Daten als die Eingabedaten im Rahmen der Untersuchung der Entscheidungsbäume angewandt werden. Daher kann der Schritt der Datenvorverarbeitung als eine Anschlusskomponente zwischen dem Simulationsmodell und dem Filter betrachtet werden. Hierdurch ist ersichtlich, dass die Phase der Datenvorverarbeitung innerhalb des Konzepts nicht ausgeschlossen werden kann.

Nach der Konstruktion des Entscheidungsbaums kann ein Klassifikator entwickelt werden, womit sich die Eingabedaten des Simulationsmodells überprüfen lassen. Dabei werden nur die Daten, mit denen die Simulation vielversprechende Resultate ermöglicht, in dem Simulationsmodell freigegeben. Alle anderen Daten werden durch den Filter ausgeschlossen. H Somit spielt der durch den Entscheidungsbaum entwickelte Klassifikator eine wesentliche Rolle zur Verknüpfung der Eingabedaten mit dem Simulationsmodell.

Es kann festgestellt werden, dass das Simulationsmodell mit dem entwickelten Konzept unmittelbar verknüpft werden kann. Die Optimierung des Simulationsmodells erfolgt durch das Zusammenwirken der beiden Komponenten „Datenvorverarbeitung“ und „Filter mittels C4.5 Entscheidungsbaum“.

Identifikation der wichtigen Parameter

Mit Hilfe des Konzepts soll auch das Ziel erreicht werden, die einflussreichen Parameter für die Optimierung der Simulation zu erkennen. Durch die Untersuchung der wichtigsten Parameter kann die Optimierung zielgerecht durchgeführt, womit sich die Effizienz der simulationsbasierten Optimierung verbessern lässt.

Im Rahmen der Datenvorverarbeitung wird ein Teil der unnötigen Parameter mittels der entsprechenden Methoden, wie bspw. Datenbereinigung und Datenreduktion, entfernt. Weiterhin können die Parameter auch auf Grundlage des Vergleichs der Werte Information Gain und Gain Ratio ausgewählt werden (vgl. 4.2.2). Hierbei werden nur die Parameter mit größeren Werten zur Konstruktion der Entscheidungsbäume angewandt. Auf diese Weise kann die Effizienz der Baumgenerierung verbessert und die Komplexität des Baums reduziert werden. Anderenfalls können die wichtigen Parameter durch Beobachtung der generierten Entscheidungsbäume erkannt werden, da die wichtigen Parameter in der Regel in der Spitze des Baums liegen. Zugleich kann die Klassifikation der Werte dieser Parameter anhand der Kanten des Baums erfolgen.

Nach Untersuchung der gegebenen Datensätze in Kapitel 3 wurde auch festgehalten, dass die durch den C4.5 Entscheidungsbaum erkannten, einflussreichen Parameter wichtige Hinweise für die effiziente Optimierung der Simulationsstudie bieten können.

Reduktion der Anzahl der Simulationsläufe

Eine weitere Anforderung an die simulationsbasierte Optimierung besteht darin, unnötige Simulationsläufe zu vermeiden und somit die Anzahl der Simulationsläufe zu reduzieren, ohne dabei die Qualität der Simulationsergebnisse zu mindern. Dieses Ziel wird vor allem durch die Komponente „Filter mittels C4.5 Entscheidungsbaum“ erreicht, weshalb diese Anforderung genauer untersucht wird.

Mit Hilfe des C4.5 Entscheidungsbaums ist es möglich, die Parameter, die zu längeren Simulationsläufen führen könnten, zu erkennen. Diese Parameter und die entsprechenden Werte können durch bestimmte Regeln repräsentiert werden. Die generierten Regeln können als ein

Filter betrachtet werden, wodurch die für die Simulationsstudie nicht erwarteten Eingabedaten erkannt und ausgeschlossen werden. Auf diese Weise kann die Anzahl der Simulationsläufe reduziert werden.

Allerdings spielt die Korrektheit und die Effizienz des Filters in dieser Arbeit für die Sicherstellung der Gewinn des Simulations- und Optimierungssystems eine wesentliche Rolle. Denn der Nutzen des Gesamtsystems entsteht vor allem aus der Reduktion der Simulationsläufe. Zugleich sind wesentliche Bestandteile der Kosten auf die falsche Klassifikation der produzierbaren Aufträge zurückzuführen (vgl. 3.3.3). In dieser Arbeit kommt daher ein Schwellenwert zum Abbruch der Suche nach einem korrekteren Entscheidungsbaum mit Berücksichtigung der Kosten und Nutzen des Gesamtsystems zum Einsatz.

4.3.3 Stärken und Schwächen des Konzepts

Nach der Validierung des Konzepts lässt sich herausstellen, dass das generierte Konzept sowohl den Anforderungen der simulationsbasierten Optimierung als auch den speziellen Anforderungen der angewandten Methoden gerecht wird. Somit kann das Konzept zur Optimierung der Simulationsstudie eingesetzt werden. Im Folgenden werden die relevanten Stärken und Schwächen des Konzepts erläutert.

Die Hauptstärken des Gesamtkonzepts bestehen darin, dass

- es auf Basis der gewonnenen Erkenntnisse durch die Untersuchung der Simulationsdaten entwickelt wurde. Daher kann das Konzept vom Anwender unmittelbar zur Analyse der Simulationsmodelle angewandt werden.
- der durch den C4.5 Entscheidungsbaum entwickelte Klassifikator für Anwender einfach zu verstehen ist. Zugleich weist dieser Klassifikator eine akzeptable Korrektheit und Effizienz zur Analyse der Ein- und Ausgabedaten auf.
- die einflussreichen Parameter für die Optimierung der Simulationsexperimente identifiziert werden können. Auf diese Weise kann die Simulation nach den Vorstellungen des Anwenders durchgeführt werden.
- die Anzahl der Simulationsläufe durch eine zielgerichtete Veränderung der Parameter innerhalb der Entscheidungsbäume reduziert werden kann.

Zwar kann das entwickelte Konzept zur Klassifikation der Eingabedaten des Simulationsmodells angewandt werden, jedoch müssen auch einige Schwachstellen erläutert werden. Diese beziehen sich vor allem auf die Korrektheit und den angegebenen Schwellenwert des Konzepts.

Da sich die Korrektheit in Abhängigkeit der Menge an Ausgabedaten verändert, ist eine adäquate Korrektheit für die Evaluation der Entscheidungsbäume vom Anwender nicht einfach abzustimmen. Eine zu hohe Korrektheit kann möglichweise zwar zu einem hohen Nutzen, aber auch zu noch höheren Kosten des Systems führen.

- ein passender Schwellenwert zum Abbruch der Suche nach einem besseren Entscheidungsbaum vom Anwender ist nur schwer festzulegen. Um sowohl die Effizienz, als

auch den Gewinn des Systems sicherzustellen, wird in dem Konzept ein Schwellenwert angegeben. Solch ein Schwellenwert wurde in dieser Arbeit nach Untersuchung der Bespieldaten für die vorliegende Optimierung nur vorgeschlagen. Eine allgemeine Methode zur Suche eines passenden Schwellenwerts für die anderen Datensätzen ist nicht bekannt.

- in dieser Arbeit wurde die Kosten-Nutzen-Analyse zur Untersuchung der Bedeutung des Schwellenwerts eingesetzt. Jedoch wurde einfach angenommen, dass die Dauer der Simulation und der Datenvorverarbeitung nur von der Menge der Trainingsdaten abhängt. Eine umfassendere Analyse mit Berücksichtigung weiterer Faktoren wurde in dieser Arbeit nicht durchgeführt.
- die Zielsetzung dieser Arbeit war es, einen Filter zu entwerfen, der vorher mittels eines Entscheidungsbaums generiert wurde. Die Phase der Datenvorverarbeitung ist prinzipiell zu diesem Zwecke nicht zwingend erforderlich. Jedoch ist die Datenvorverarbeitung für die Generierung des Entscheidungsbaums mittels C4.5 Algorithmus von wesentlicher Bedeutung. Daher könnte als ein zukünftiger Forschungsschwerpunkt ein Verfahren entwickelt werden, welches die Anwendung der Datenvorverarbeitung nicht erfordert. Damit könnte die Auswahl der Eingabe-daten effizierter und einfacher durchgeführt werden.

5 Zusammenfassung

Im Rahmen dieser Arbeit konnte ein Konzept entwickelt werden, welches im Sinne einer Optimierung die wichtigen Modellparameter erkennt und prüft. Somit werden nur solche Simulationen durchgeführt, welche ein vielversprechendes Ergebnis erwarten lassen. Dabei werden nur die Daten ausgewählt, die mittels eines generierten Filters als Eingabedaten zum Simulationsmodell freigegeben werden. Der Filter wird in einem vorgelagerten Schritt durch einen C4.5 Entscheidungsbaum generiert. Dadurch kann das Ziel erreicht werden, die Anzahl der Simulationsläufe zu reduzieren, ohne dabei gleichzeitig die Qualität der Simulations-ergebnisse zu mindern.

Zur erfolgreichen Entwicklung dieses Konzepts wurde zunächst eine Aufarbeitung der theoretischen Grundlagen durchgeführt. Durch die Untersuchung der Beispieldatensätze wurden anschließend weitere relevante Erkenntnisse für das Konzept gewonnen.

Aufgrund der zunehmenden Komplexität der Produktionssysteme gewinnt das wandlungsfähige und modulare Produktionssystem an Bedeutung. Zugleich kommen vermehrt Simulationsstudien zur Untersuchung und zur Optimierung des Produktionssystems zum Einsatz, um die Engpässe der analytisch-mathematischen Methoden zu überwinden. Daher wurden in einem weiteren Abschnitt sowohl die Vorgehensweise der ereignisdiskreten Simulation, als auch die Grundlagen der simulationsbasierten Optimierung erläutert.

Danach wurden die gängigen Entscheidungsbäume vorgestellt und miteinander verglichen, um den geeignetsten Entscheidungsbaum zur Analyse der Ausgabedaten und zur Optimierung des Produktionssystemmodells auszuwählen. Dabei wurden die Vor- und Nachteile der dargestellten Entscheidungsbäume anhand verschiedener Kriterien zusammenfasst und bewertet. Damit die Performance der Entscheidungsbäume gewährleistet werden konnte, war die Vorstellung der theoretischen Grundlagen der Datenvorverarbeitung ebenfalls von großer Bedeutung.

Nach Betrachtung der Grundlagen wurden zunächst die Ausgabedaten des Simulationsmodells sowohl mit Hilfe statistischer Methoden, als auch durch die Methode der Datenvorverarbeitung behandelt. Folgenden Ziele sollten damit erreicht werden: 1. Gewinn eines ersten Einblicks in die Ausgabedaten, 2. Sicherstellung der Effizienz und der Korrektheit der Entscheidungsbäume durch die Entfernung irrelevanter Daten, 3. Umwandlung der Daten in ein für die Baumgenerierung anwendbares Format.

Anschließend wurde die Phase der Auswahl der Entscheidungsbäume in eine Vorauswahl und in die tatsächliche Auswahl gegliedert. In der Phase der Vorauswahl wurden die Entscheidungsbäume vor allem mit Blick auf die Formate der Daten ausgewählt. Weiterhin wurden die ausgewählten Entscheidungsbäume auf Grundlage der Beispieldaten generiert und anhand von ausgewählten Kriterien bewertet. Dadurch konnte der Entscheidungsbaum mit der besten Performance ausgewählt werden, wobei hierbei und in den folgenden Abschnitten der C4.5

Algorithmus als geeignetes Verfahren eingesetzt wurde. Darüber hinaus wurde der Zusammenhang zwischen der Korrektheit und der Menge der Trainingsdaten analysiert, wozu die Kosten-Nutzen-Analyse zur Anwendung kam. Im Anschluss daran wurden die durch die Analyse der Beispieldaten gewonnenen Erkenntnisse zusammengefasst.

Im Kapitel 4 wurden diese Erkenntnisse zum Entwurf des Konzepts eingesetzt. Das Konzept bestand vor allem aus den beiden Komponenten „Datenvorverarbeitung“ und „Filter mittels C4.5 Entscheidungsbäume“. Im Rahmen der Datenvorverarbeitung wurden einerseits die unnötigen Attribute entfernt, andererseits wurden die Daten in entsprechende Formate umgewandelt. Die verarbeiteten Daten konnten nunmehr zur Konstruktion der Entscheidungsbäume mittels des C4.5 Algorithmus eingesetzt werden.

Im Kontext der Entwicklung der Entscheidungsbäume wurden die Attribute durch die Berechnung und den Vergleich der Werte Information Gain und Gain Ratio ausgewählt. Um die Effizienz der Konstruktion der Entscheidungsbäume zu gewährleisten, wurden nur die Attribute mit höheren Werten Information Gain und Gain Ratio angewandt. Allerdings muss man die Möglichkeit berücksichtigen, dass die Suche nach dem besten Entscheidungsbaum aufwendig oder sogar unmöglich ist. Daher wurde ein Schwellenwert zur Einschränkung der Suche des Entscheidungsbaums angegeben. Hierbei diente vor allem die Kosten-Nutzen-Analyse zur einfachen Erklärung der Schwellenwertangabe. Schließlich wurde das Konzept sowohl hinsichtlich der Anforderungen der simulationsbasierte Optimierung nach Fu (2002), als auch hinsichtlich der Anforderungen der Datenvorverarbeitung und Entscheidungsbäume nach Han (2012) validiert. Durch die Validierung wurde die Anwendbarkeit und die Korrektheit des Konzepts überprüft und sichergestellt. Abschließend wurden die Stärken und Schwächen des Gesamtkonzepts auf Grundlage der bisherigen Analyse benannt und zusammenfassend erläutert.

Literaturverzeichnis

DIN 25424, September 1981: Fehlerbaumanalyse - Methode und Bildzeichen

VDI-Richtlinie 3633, Dezember 2013: Simulation von Logistik-, Materialfluss- und Produktionssystemen - Begriffe.

VDI-Richtlinie 3633 Blatt 1, Dezember 2014: Simulation von Logistik-, Materialfluss und Produktionssystemen - Grundlagen.

VDI-Richtlinie 3633 Blatt 3, Dezember 1997: Simulation von Logistik-, Materialflussund Produktionssystemen - Experimentplanung und -auswertung.

IBM SPSS Modeler 14.2 Algorithm Guide (2011)

April, Jay; Glover, Fred; Kelly, James P.; Laguna, Manuel (2003): Practical introduction to simulation optimization. In: Ferrin, D.; Sanchez, P.; Chick, S.; (Hrsg.): Proceedings of the 2003 Winter Simulation Conference, New Orleans, USA, S. 71-78.

Arnold, Dieter; Furmans, Kai; Isermann, Heinz; Kuhn, Axel; Tempelmeier, Horst (2008): Handbuch Logistik (VDI-Buch) (German Edition). Dordrecht: Springer (VDI-Buch).

Banks, Jerry; Carson, John S.; Nelson, Barry L.; Nicol, David M (2014): Discrete-event system simulation. Harlow: Pearson.

Barros, Rodrigo C.; Carvalho, André Carlos Ponce de Leon Ferreira; Freitas, Alex A. (2015): Automatic Design of Decision-Tree Induction Algorithms. Cham [Switzerland]: Springer.

Becker, Jörg; Probandt, Wolfgang; Vering, Oliver (2012): Grundsätze ordnungsmäßiger Modellierung: Konzeption und Praxisbeispiel für ein effizientes Prozessmanagement. Berlin, Heidelberg: Springer Berlin Heidelberg (BPM kompetent).

Breiman, Leo (1984): Classification and regression trees. Belmont, Calif.: Wadsworth (Wadsworth statistics / probability series).

Casjens, Sw*aa*ntje Wiarda (2013): Adaption und Vergleich evolutionärer mehrkriterieller Algorithmen mit Hilfe von Variablenwichtigkeitsmaßen – Am Beispiel der kostensensitiven Klassifikation von Lungenkrebssubtypen. Dissertation. Technische Universität Dortmund. Dortmund: o. V.

Deuse, Jochen (1998): Fertigungsfamilienbildung mit feature-basierten Produktmodelldaten. Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, Aachen: Shaker.

Díaz-Pérez, Flora Ma; Bethencourt-Cejas, Ma (2016): CHAID algorithm as an appropriate analytical method for tourism market segmentation. In: Journal of Destination Marketing & Management, S. 1-8.

Dombrowski, Uwe; Mielke, Tim (2015): Ganzheitliche Produktionssysteme: Aktueller Stand und zukünftige Entwicklungen. Berlin, Heidelberg: Springer (VDI-Buch).

Fischer, Christian (2014): Planung und Steuerung von Montageanlagen: Grundlagen für die Optimierung in der Montage. In: Klaus Feldmann, Volker Schöppner und Günter Spur (Hrsg.): Handbuch Fügen, Handhaben und Montieren. München: Carl Hanser Fach-buchverlag, S. 602–603.

García, Salvador; Luengo, Julián; Herrera, Francisco (2015): Data preprocessing in data mining. Cham: Springer.

Günther, Hans-Otto; Tempelmeier, Horst (2012): Produktion und Logistik. 9., aktualisierte und erw. Aufl. Berlin, Heidelberg: Springer (Springer-Lehrbuch).

Han, Jiawei; Kamber, Micheline (2006): Data mining: Concepts and techniques. 2nd ed. Amsterdam, London: Elsevier.

Han, Jiawei; Kamber, Micheline; Pei Jian; Fan Ming; Meng Xiaofeng (2012): Data Mining. 2nd ed. Beijing: China Machine Press.

Hanschke, Thomas und Zisgen, Horst (2015): Verknüpfung von Simulation und Optimierung: Kategorien und Beispiel – Ein Bericht über die VDI Richtlinie 3633 Blatte12. In: Rabe, Markus und Clausen, Uwe (Hrsg.): Simulation in Production and Logistics 2015. Stuttgart: Fraunhofer IRB Verlag, Stuttgart, S. 111-118.

Heinen, T.; Rimpau, C.; Wörn, A. (2008): Wandlungsfähigkeit als Ziel der Produktionssystemgestaltung. In: Nyhuis, P.; Reinhart, G.; Abele, E. (Hrsg.): Wandlungsfähige Produktionssysteme. Hannover: Garbsen, S. 19-32.

Hirsch-Kreinsen, Hartmut; Weyer, Johannes (2014): Wandel von Produktionsarbeit – „Industrie 4.0“, Arbeitspapier Nr. 38.

Hrdliczka, Veronika (1997): Leitfaden für Simulationsbenutzer in Produktion und Logistik. 2. Aufl. ASIM-Fachgruppe 4.5.6. Simulation in Produktion und Logistik (ASIM-Mitteilung, 58).

Hong, L.Jeff; Nelson, Barry, L. (2008): A brief introduction to optimization via simulation. In: Winter Simulation Conference 2009, S. 75-85.

Hong, L. Jeff; Nelson, Barry L.; Xu, Jie (2015): Discrete Optimization via Simulation. In: Fu, Micheal (Hrsg.): Handbook of Simulation Optimization. Aufl. New York, Heidelberg, Dordrecht, London: Springer, S. 9-44.

Ittner, Andreas, & Schlosser, Michael (1996): Non-linear decision trees-NDT. In: Proceedings of the 13th International Conference on Machine Learning (ICML), S. 252-257.

Kass, G. V. (1980): An Exploratory Technique for Investigating Large Quantities of Categorical Data. In: Applied Statistics 29 (2), S. 119.

Košturiak, Jan; Gregor, Milan (1995): Simulation von Produktionssystemen. Wien, New York: Springer.

Krahl, Daniela; Windheuser, Ulrich; Zick, Friedrich-Karl (1998): Data Mining. Einsatz in der Praxis. 1. Aufl. Bonn: Addison-Wesley-Longman.

Krug, Wilfried; Rose, Oliver; (2011): Optimierung. In: März, L; Krug, W.; Rose, O.; Weigert, G. (Hrsg.): Simulation und Optimierung in Produktion und Logistik: Praxisorientierte Leitfaden mit Fallbeispielen. Berlin: Springer, S. 21-28.

Kuhn, Axel; Wenzel, Sigrid (2008): Simulation logistischer Systeme. In: Dieter Arnold, Axel Kuhn, Kai Furmans, Heinz Isermann und Horst Tempelmeier (Hrsg.): Handbuch Logistik. Berlin, Heidelberg: Springer (VDI-Buch), S. 73–94.

Lämmel, Uwe; Cleve, Jürgen (2014): Data Mining. München: De Gruyter Oldenbourg.

Li, hang (2012): Statistical Learning Methods. Beijing: Tsinghua University Press.

Liu, Peng; Yao, Zheng; Yin, Junjie (2006): Improved Decision Tree of C4.5. Journal of Tsinghua University (Sci&Tech) 46 (1), S. 996-1001.

März, Lothar; Krug, Wilfried (2011): Kopplung von Simulation und Optimierung. In: März, L; Krug, W.; Rose, O.; Weigert, G. (Hrsg.): Simulation und Optimierung in Produktion und Logistik: Praxisorientierte Leitfaden mit Fallbeispielen. Berlin: Springer, S. 41-47.

März, Lothar; Weigert, Gerald (2011): Simulationsgestützte Optimierung. In: Lothar März, Wilfried Krug, Oliver Rose und Gerald Weigert (Hrsg.): Simulation und Optimierung in Produktion und Logistik. Praxisorientierter Leitfaden mit Fallbeispielen. Berlin: Springer (VDI-Buch), S. 3–12.

Mattern, Friedenmann und Mehl, Horst (1989): Diskrete Simulation - Prinzipien und Probleme der Effizienzsteigerung durch Parallelisierung. In: Informatik-Spektrum 12 (4), S. 198-210.

Mitchell, Tom M. (1997): Machine Learning. New York: McGraw-Hill (McGraw-Hill series in computer science).

Neuhausen, Jörn (2001): Methodik zur Gestaltung modularer Produktionssysteme für Unternehmen der Serienproduktion. Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, Aachen: o.V.

Pang, Sulin; Gong, Jizhang (2009): C5.0 Classification Algorithm and Application on Individual Credit Evaluation of Banks. In: Systems Engineering - Theory & Practice 29 (12), S. 94–104.

Petersohn, Helge (2005): Data Mining. Verfahren, Prozesse, Anwendungsarchitektur. München, Wien: Oldenbourg.

Polzin, Brigitte (2006): Die Reduktion von Daten: Ein Vergleich datenreduzierender Verfahren auf ihre Leistungsfähigkeit im sozialwissenschaftlich-statistischen Umfeld am Beispiel einer

Untersuchung zu den Faktoren des subjektiven Sicherheitsgefühls von Frauen und Männern in der Stadt Bochum, Dissertation Universität Duisburg-Essen, Essen: o.V.

Quinlan, J. Ross (1986): Induction of decision trees. In: Machine Learning 1 (1), S. 81–106.

Quinlan, J. Ross (1993): C4.5. Programs for machine learning. In: J. Ross Quinlan. San Mateo, Calif.: Morgan Kaufmann (The Morgan Kaufmann series in machine learning).

Quinlan, J. Ross (1996): Improved use of continuous attributes in C4. 5. Journal of artificial intelligence research 4, S. 77-90.

Rabe, Markus (2001): Handlungsanleitung Simulation in Produktion und Logistik. Ein Leitfaden mit Beispielen für kleinere und mittlere Unternehmen. San Diego: SCS International.

Rabe, M.; Dietel, U.; Kreppenhofer, D. (2001): Handlungsanleitung Simulation. In: Rabe und Hellingrath (Hrsg.): Handlungsanleitung Simulation in Produktion und Logistik. San Diego: SCS International.

Rabe, Markus (2008): Verifikation und Validierung für die Simulation in Produktion und Logistik. Vorgehensmodelle und Techniken. Berlin, Heidelberg: Springer (VDI-Buch).

Rauch, Ewin (2013): Konzept eines wandlungsfähigen und modularen Produktionssystems für Franchising-Modelle. Dissertation, Universität Stuttgart, Stuttgart: o.V

REFA-Verband für Arbeitsstudien und Betriebsorganisation e. V. (1991): Arbeitsgestaltung in der Produktion. In: REFA-Verband für Arbeitsstudien und Betriebsorganisation e. V. (Hrsg.): Methodenlehre der Betriebsorganisation: München: Hanser.

Robinson, Stewart (2004): Simulation. The practice of model development and use. Chichester: Wiley.

Rokach, Lior; Maimon, Oded (2015): Data mining with decision trees: Theory and applications. Second edition. Hackensack New Jersey: World Scientific.

Rose, Oliver; März, Lothar (2011): Simulation. In: Lothar März, Wilfried Krug, Oliver Rose und Gerald Weigert (Hrsg.): Simulation und Optimierung in Produktion und Logistik. Praxisorientierter Leitfaden mit Fallbeispielen. Berlin: Springer (VDI-Buch), S. 13–19.

Runkler, Thomas A. (2010): Data-Mining. Methoden und Algorithmen intelligenter Datenanalyse. 1. Aufl. Wiesbaden: Vieweg + Teubner (Studium).

Schneider, Christina; Bunse, Katharina; Schönsleben, Paul (2010): Bewertungskriterien für die Modularisierung in der Automobilproduktion. In: Nyhuis Peter (Hrsg.): Wandlungsfähige Produktionssysteme. Berlin: Gito, S. 121-136.

Shannon, Claude. E. (2001): A mathematical theory of communication. In: ACM SIGMOBILE Mobile Computing and Communications Review 5 (1), S. 3-55.

Sivasankari, A.; Sudarvizh, S.; Radhika Amirtha Bai, S. (2014): Comprative study of different clustering and decision tree for data mining algrorithm. In: international Journal of Computer Science and Information Technology Research 2 (3), S. 221-232.

Su, Jiang und Zhang, Harry (2006): A fast decision tree learning algorithm. In: Association for the Advancement of Artificial Intelligence (AAAI), S. 500-505.

Ture, Mevlut; Tokatli, Fusun; Kurt, Imran (2009): Using Kaplan–Meier analysis together with decision tree methods (C&RT, CHAID, QUEST, C4.5 and ID3) in determining recurrence-free survival of breast cancer patients. In: Expert Systems with Applications. S. 2017–2026.

Weiger, Gerald; Rose, Oliver (2011): Stell- und Zielgrößen. In: März, L; Krug, W.; Rose, O.; Weigert, G. (Hrsg.): Simulation und Optimierung in Produktion und Logistik: Praxisorientierte Leitfaden mit Fallbeispielen. Berlin: Springer 2011, S. 29-39.

Wenzel, Sigrid; Collisi-Böhmer, Simone; Pitsch, Holger; Röse, Oliver; Weiß, Matthias (2008): Qualitätskriterien für die Simulation in Produktion und Logistik. Planung und Durchführung von Simulationsstudien. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg (VDI-Buch).

Witten, Iran; Frank, Eibe; Hall, Mark (2011): Data mining. Practical machine learning tools and techniques. San Francisco, Calif., London: Morgan Kaufmann.

Witten, Ian und Frank, Eibe (2001): Data mining: Praktische Werkzeuge und Techniken für das maschinelle Lernen. München, Wien: Hanser.

6 Anhang

Tabelle 6-1: Die Korrektheit der 3 Algorithmen anhand der unterschiedlichen Datenmenge

Trainingsdaten	Testdaten	C4.5	C5.0	CART
5% des Datensatz 1 (750)	Datensatz3	83,7%	83,07%	81,6%
10% des Datensatz 1 (1500)	Datensatz3	85,9%	84,84%	82,57%
20% des Datensatz 1 (3000)	Datensatz3	86.53%	86,46%	84,63%
40% des Datensatz 1 (6000)	Datensatz3	88,23%	87,57%	86,0%
60% des Datensatz 1 (9000)	Datensatz3	88,93%	88,45%	87,88%

* Die Trainingsdaten werden nach dem Zufallsprinzip 5-mal ausgewählt.

Tabelle 6-2: Beispieldaten aus dem Datensatz 1 mit 18 Parameter

Order-Id	Order-Name	Custo-merID	Count	Green	Red	Yellow	splcl – BFT	splbat - DF	splbat - BFT	initSub - DT	initSub - BFT	storet - DT	Storet - BFT	shipt - DT	Shipt - BFT	Result
1	Order1	9005	2055	530	592	933	10	20	10	20	10	20	10	20	10	41230
2	Order2	4129	1975	814	724	437	10	20	10	20	10	20	10	20	10	39630
3	Order3	9853	593	54	524	15	10	20	10	20	10	20	10	20	10	11990
4	Order4	5813	947	337	610	0	10	20	10	20	10	20	10	20	10	19060
5	Order5	7521	1254	109	478	667	10	20	10	20	10	20	10	20	10	25210
6	Order6	8473	703	187	224	292	10	20	10	20	10	20	10	20	10	14190
7	Order7	5942	956	37	208	711	10	20	10	20	10	20	10	20	10	19250
8	Order8	8868	1735	491	406	838	10	20	10	20	10	20	10	20	10	34830
9	Order9	3847	1553	473	829	251	10	20	10	20	10	20	10	20	10	31190
10	Order10	5032	1537	221	716	600	10	20	10	20	10	20	10	20	10	30870

* BFT: BaseFiringTime; DT: DelayTime

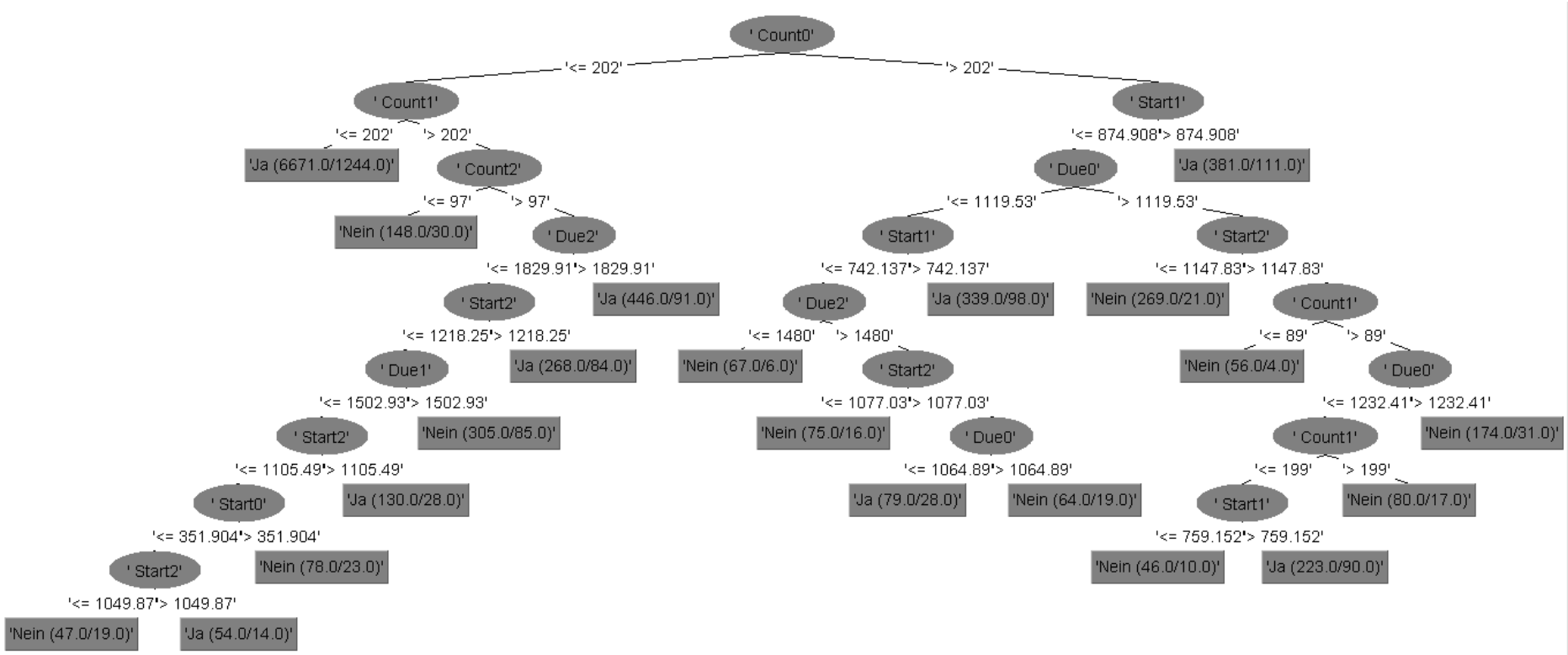


Abbildung 6-1: Entscheidungsbaum mit C4.5 in Abhängigkeit des Datensatzes 2 (Komplett)

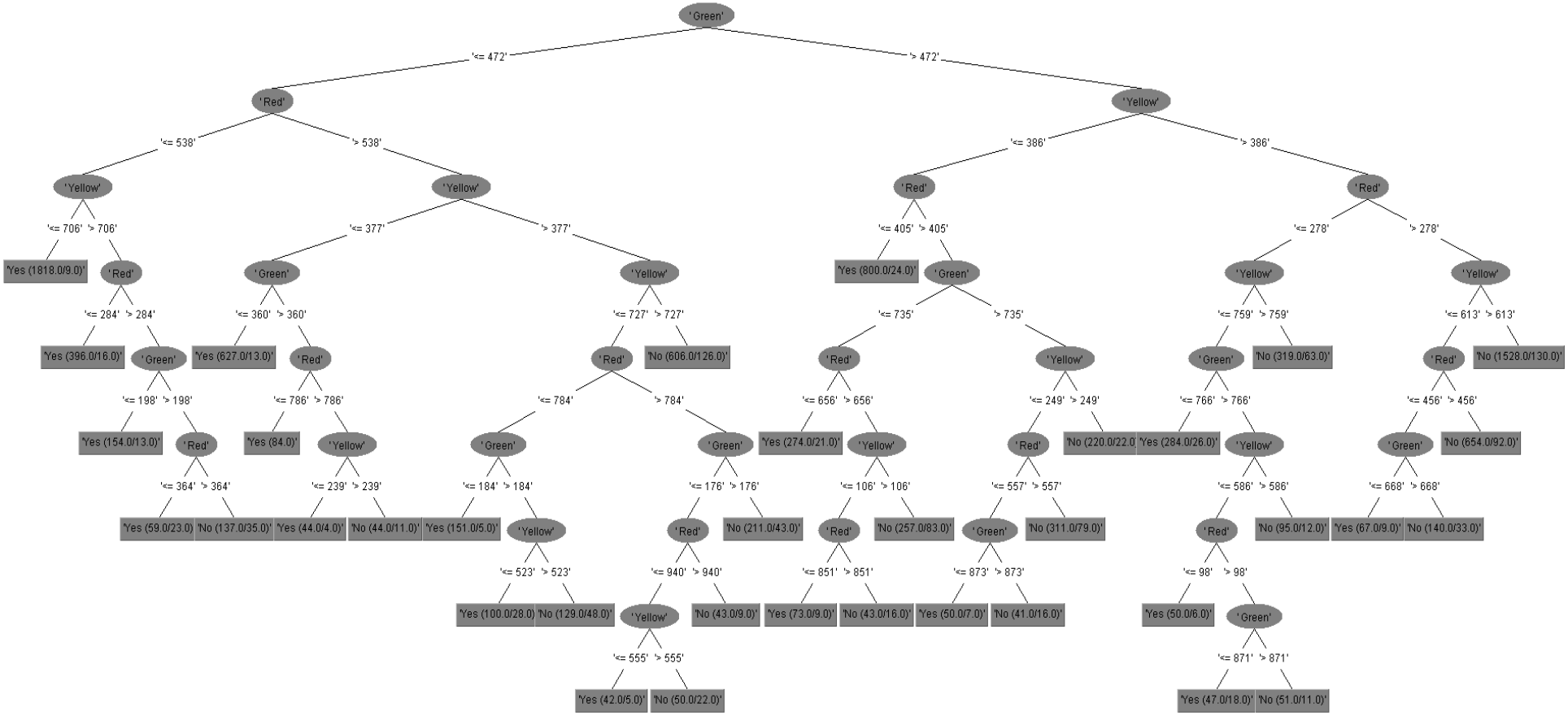


Abbildung 6-2: Entscheidungsbaum mit C4.5 in Abhängigkeit des Datensatzes 2 (Komplett)

Eidesstattliche Versicherung

_____ _____

Name, Vorname Matr.-Nr.

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

_____

_____

_____

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.

_____ _____

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:

_____ _____

Ort, Datum Unterschrift