



# Entwicklung einer ereignisdiskreten Simulation auf Basis von BIM

Development of a discrete-event simulation  
based on BIM

Wissenschaftliche Arbeit zum Erlangen des akademischen Grades  
M. Sc.

an der Fakultät für Maschinenwesen der Technischen Universität München

**Themenstellender** Prof. Dr.-Ing. Johannes Fottner  
Lehrstuhl für Fördertechnik Materialfluss Logistik

**Betreuerin** Anne Fischer, M. Sc.

**Eingereicht von:** Shuo Ma, B. Sc.  
Enzianstraße 3  
85748 Garching

**Eingereicht am:** 01.06.2021 in Garching

**Inventarnr. fml:** 2020/104



## Kurzfassung

---

Building Information Modeling (BIM) kommt heutzutage aufgrund der Komplexität von Bauvorhaben zum Einsatz. Es wird BIM als Basis der Daten des digitalen Zwillings im Bauwesen angesehen. Hierfür wird in der vorliegenden Arbeit ein Modell der ereignisorientierten Simulation (DES) auf Basis von BIM mithilfe der Taktzeit entwickelt. Damit kann der digitale Zwilling im Bauwesen vereinfacht werden.

Die Informationsbasis von BIM als digitalem Zwilling im Bauwesen kann durch die Standardisierung des IFC-Formats aufgebaut werden. Hierbei wird die Modellierung des IFC-Formats anhand des UML-Diagramms mithilfe des xBIM-Toolkits und des Tools IfcConvert in Microsoft Visual Studio 2019 (MVS) implementiert. Darüber hinaus wird die Software Constructivity als Model Checker verwendet.

Weiterhin werden die Informationen aus dem IFC-Modell in der Software Plant Simulation von Siemens PLM abgespeichert. Es bewirkt, dass die Informationen als Eigenschaften sowohl dem BIM-Modell als auch der Informationsbasis des Simulationsmodells angehängt werden. Zusätzlich werden die Eigenschaften des IFC-Modells eindeutig klassifiziert. Des Weiteren kann das Simulationsmodell anhand des Flussdiagramms erstellt werden.

Die Simulationsstudie wird auf Basis des IFC-Modells durchgeführt. In der DES werden die Aussagen über die Bauzeit gewonnen. Des Weiteren werden dadurch die optimale Maschineneinsatzplanung sowie die Produktionsstrategie ermittelt und getestet.



## Abstract

---

Due to the complexity of building projects, Building Information Modelling (BIM) has been put into use at present and can be seen as the data base for the digital twin in construction industry. Therefore, in the current work, we have developed a BIM based event-oriented simulation (DES) using takt time. This simplifies the construction of digital twin.

It can establish a database of digital twin in the construction industry through standardizing the IFC format. Using xBIM-Toolkit and Tool IfcConvert in Microsoft Visual Studio 2019, the modelling of IFC with UML is implemented. Software Constructivity is also applied as a Model Checker.

In fact, the information of IFC model is stored in the Plant Simulation software of Siemens PLM. Its effect is to attach information as attributes to the information base of BIM model and Simulation model. Further more, the attributes of the IFC model are explicitly classified. IFC models can be built using UML diagrams, as well as simulation models can be created using flowcharts.

The simulation study based on IFC model can obtain the relevant construction time by means of DES. Moreover, the determination of relevant construction time can test the optimal machine deployment plan and production strategy.



## Vorwort

---

Die vorliegende Arbeit entstand unter der wissenschaftlichen und inhaltlichen Anleitung von Anne Fischer, M. Sc., wissenschaftlicher Mitarbeiterin am Lehrstuhl für Fördertechnik Materialfluss Logistik (fml) der Technischen Universität München.

### Vereinbarung zum Urheberrecht

Hiermit gestatte ich dem Lehrstuhl für Fördertechnik Materialfluss Logistik diese Studienarbeit bzw. Teile davon nach eigenem Ermessen an Dritte weiterzugeben, zu veröffentlichen oder anderweitig zu nutzen. Mein persönliches Urheberrecht ist über diese Regelung hinaus nicht beeinträchtigt. Eventuelle Geheimhaltungsvereinbarungen über den Inhalt der Arbeit zwischen mir bzw. dem Lehrstuhl für Fördertechnik Materialfluss Logistik und Dritten bleiben von dieser Vereinbarung unberührt.

Garching, 01.06.2021. Shuo Ma

Ort, Datum, Unterschrift



# Inhaltsverzeichnis

---

<b>Inhaltsverzeichnis</b>	<b>I</b>
<b>Abkürzungsverzeichnis</b>	<b>III</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Ausgangssituation und Problemstellung	1
1.2 Zielsetzung	2
1.3 Vorgehensweise	4
1.3.1 IFC-Schema	4
1.3.2 Simulationsmodell	5
1.4 Gliederung der Arbeit	6
<b>2 Stand der Wissenschaft und Technik</b>	<b>9</b>
2.1 Ereignisdiskrete Simulation im Bauwesen	9
2.1.1 Modellierungsansätze	9
2.1.2 Einordnung der Optimierungsansätze	15
2.1.3 Ereignisdiskrete Simulation als digitaler Zwilling im Bauwesen	16
2.2 Integration von Baufortschrittsdaten in die Simulation	17
2.2.1 Photogrammetrie und Videoanalyse	18
2.2.2 Sensordaten	18
2.3 Building Information Modeling (BIM)	19
2.3.1 Einordnung zu Vorgängerprojekten	19
2.3.2 Begriffserklärung	20
2.4 Industry Foundation Classes (IFC)	21
2.4.1 Historische Entwicklung	22
2.4.2 Datenformate	22
2.4.3 Die Vererbungshierarchie	23
<b>3 Lösungsansätze für den Umgang mit BIM-Modellen</b>	<b>31</b>
3.1 Programmiersprachen	31
3.2 Vergleich von STEP/IFC-Bibliotheken	31
3.3 Vergleich von Softwarelösungen für die BIM-Modellierung	35

<b>4 BIM-Modell</b>	<b>43</b>
4.1 Datenbeschaffung	43
4.2 Datenaufbereitung	48
4.2.1 IFC-Modell	48
4.2.2 Terminplan	54
4.3 XML-Import in Plant Simulation	57
<b>5 Simulationsstudie am Fallbeispiel „Westtangente Rosenheim (WTRO)“</b>	<b>59</b>
5.1 Datenbeschaffung	59
5.2 Datenaufbereitung	59
5.3 Modellierung	61
5.3.1 Modulares DES-Modell	61
5.3.2 Modellierung der Szenarien	66
5.4 Simulationsergebnisse und Auswertung	67
5.4.1 Vergleich der Bauzeit im ersten Szenario	67
5.4.2 Vergleich der Bauzeit im zweiten Szenario	73
5.4.3 Vergleich der Bauzeit in beiden Szenarien	76
<b>6 Zusammenfassung und Ausblick</b>	<b>79</b>
6.1 Zusammenfassung	79
6.2 Ausblick	81
<b>Literaturverzeichnis</b>	<b>83</b>
<b>Abbildungsverzeichnis</b>	<b>91</b>
<b>Tabellenverzeichnis</b>	<b>95</b>
<b>A Anhang - Darstellung der IFC-Eigenschaften in der Software Constructivity</b>	<b>A-1</b>
<b>B Anhang - IFC-Datei</b>	<b>B-1</b>
<b>C Anhang - Entitäten aus dem IFC-Datenmodell im Programm MVS</b>	<b>C-1</b>
<b>D Anhang - Visualisierung in der Software Plant Simulation</b>	<b>D-1</b>

# Abkürzungsverzeichnis

---

<b>Abkürzung</b>	<b>Bedeutung</b>
ABM	Agent-based model, dt. Agentenbasierte Modellierung
AEMP	Association of Equipment Management
AIM	Application Interpreted Model
ARM	Application Reference Model
API	Application Programming Interfaces Professionals
AGPL	Affero General Public License
BIM	Building Information Modeling
BSD	Berkeley Software Distribution
BPMN	Business Process Model and Notation
CC	Creative Commons
CDDL	Common Development and Distribution License
CPS	Cyber Physical System
CST	Construction Simulation Toolkit
DES	Discrete Event Simulation
DS	System Dynamics
DV	Datenverarbeitungsebene
fml	(Lehrstuhl für) Fördertechnik Materialfluss Logistik
ForBAU	Forschungsverbund „Virtuelle Baustelle“
FIFO	First-In-First-Out

GIS	Geographic Information System
GUID	Globally Unique Identifier
IFC	Industry Foundation Classes
JIT	Just in time
IoT	Internet der Dinge
LGPL	GNU Lesser General Public License
LOD	Level of Development
LOG	Level of Geometry
LOI	Level of Information
MiC 4.0	Machines in Construction – MiC 4.0
MIT	Massachusetts Institute of Technology
MVS	Microsoft Visual Studio
OOM	objektorientierte Modellierung
OOA	objektorientiertes Design
OOP	objektorientierte Programmierung
PDCA-Zyklus	Plan-Do-Check-Action-Zyklus
ProSIM	online Simulation-Schnittstelle
RFID	Radio-Frequency Identification
SCM	Supply Chain Management
SD	System Dynamics
SDAI	Standard Data Access Interface

SMC	Solibri Model Checker
STEP	Standard for the Exchange of Product Model Data
TUM	Technische Universität München
VDBUM	Verband der Baubranche, Umwelt- und Maschinentechnik e. V.
WebGL	Web Graphics Library
WTRO	Westtangente Rosenheim
xBim	eXtensible Building Information Modeling
XML	Extensible Markup Language, dt.: erweiterbare Auszeichnungssprache



# 1 Einleitung

---

## 1.1 Ausgangssituation und Problemstellung

Aufgrund der Komplexität von Bauvorhaben kommt BIM zum Einsatz. BIM kann im Bauwesen als Datenbasis des digitalen Zwillings betrachtet werden. Die größten Einschränkungen des aktuellen 4D-Modells im Bauvorhaben bestehen darin, dass das BIM-Modell hauptsächlich die Information des 3D-Modells und keine exakten Informationen für die Dynamik der Bauprozesse, um ein 4D-Modell abzubilden, zur Verfügung stellt. Das kann daher zu Verzögerungen und zu erhöhten Kosten führen [Abb-2020].

Ein probates Werkzeug, um bei Verzögerungen im Prozess Handlungsempfehlungen auf Basis von Variantenstudien geben zu können, ist die Ablaufsimulation oder DES. DES kann Prozesse mit verschiedenen Detaillierungsgraden abbilden und verschiedene Optimierungsansätze können damit modelliert werden. Ein grundsätzliches Problem bei der Anwendung von DES im Bauwesen ist, dass Vorhersagen wegen der Dynamik der Bauprozesse nicht präzise getroffen werden können. Weiterhin besteht das Problem, dass das Simulationsmodell sehr flexibel aufgebaut sein muss, um den bauspezifischen Randbedingungen zu genügen [Gal-2018].

Des Weiteren ist die Entwicklung von BIM auf Basis von DES im Bauwesen in Ismail et al. [Ism-2017] und Abbasi et al. [Abb-2020] beschrieben. Deren Auswertung diente als Basis der vorliegenden Arbeit. Von Ismail et al. [Ism-2017] wird ein integrierter prozessbasierter Simulationsbaukasten CST und die webbasierte Simulationsschnittstelle ProSIM für die Bauablaufplanung durch Simulation vorgestellt. Dazu basiert die Datenintegration zwischen den Simulationsmodellen und den Projektinformationen auf dem BIM und dem Multimodelansatz, der auch als BIM-Ansatz bezeichnet wird. Außerdem werden nicht nur die Bauprozesse, sondern auch Logistikprozesse mit der Business Process Model and Notation (BPMN) modelliert, die als Simulationsprozessvorlage in das Simulationsmodell möglicherweise automatisch transformiert werden kann. Es wird zudem von Ismail et al. [Ism-2017] erklärt, dass die Echtzeitinformationen auf der Baustelle erfasst und in der Simulationsstudie auf Basis von Cyber Physical Systems (CPS) und Radio-Frequency Identification (RFID) übertragen werden [Ism-2017]. Weiterhin entwickelte Abbasi et al. [Abb-2020] die

DES auf Basis von BIM mithilfe der Taktzeit, um das BIM-Modell besser mit dem Supply Chain Management (SCM) zu verknüpfen. Damit wird die Implementierung der Just-in-time-Produktion (JIT) gefördert. Hierbei geht es um den Plan-Do-Check-Action-Zyklus (PDCA-Zyklus). Das kann zu einer realistischen und optimalen Planung des Produktionsprozesses führen [Abb-2020]. Hierfür wird davon der erste Ansatz gezeigt, dass die Integration von BIM in die DES möglich ist, um das Simulationsmodell auf aktuellen Datengrundlagen zu modellieren [Abb-2020].

Darüber hinaus ist das Pull-System und Push-System beschrieben, wie in Hopp und Spearman [Hop-2008]. Mithilfe der Pull-Steuerung können der Umlaufbestand und die Zykluszeit reduziert werden. Außerdem kann die Anwendung der Pull-Steuerung im Gegensatz zur Push-Steuerung zu reibungslosem Produktionsfluss, besserer Qualität und geringeren Kosten führen [Hop-2008]. Hierfür werden die Optimierungsansätze zur Produktionssteuerung, wie Pull-Steuerung und Push-Steuerung, in Bezug auf den Maschineneinsatz in der vorliegenden Arbeit angewendet.

## 1.2 Zielsetzung

Die vorliegende Arbeit greift den Ansatz auf und hat zum Ziel, BIM in ein DES-Modell zur Abbildung des Spezialtiefbauprozesses effektiv zu integrieren. Dabei sollen folgende Fragestellungen geklärt werden:

- Wie ist der aktuelle Stand von BIM im Spezialtiefbau? Welche Eigenschaften können als Eingangparameter für ein DES-Modell dienen?
- Wie kann eine Datenintegration von BIM in ein DES-Modell erfolgen?
- Wie kann ein modulares DES-Modell für den Spezialtiefbau konzipiert werden, um eine Datenintegration zu ermöglichen?
- Welche Möglichkeiten ergeben sich durch das entwickelte DES-Modell hinsichtlich der Prozessoptimierung?

Hierfür wird ein BIM-Modell im IFC-Schema entwickelt und als Schnittstelle in das DES-Modell implementiert. Für die Erstellung werden unterschiedliche Quellen für die Beschreibung eines BIM-Modells im Spezialtiefbau herangezogen. Verschiedene Informationen, wie

Zeit, Geometrie und Kosten, sollen dabei berücksichtigt werden. Für qualitative Vorhersagen des Baufortschritts lassen sich die realen Daten prozessbegleitend in DES-Modelle einbinden [Fis-2020].

Um die ereignisdiskrete Simulation auf Basis von BIM im Bauwesen zu entwickeln, wird das IFC-Format durch das xBIM-Toolkit und das Tool IfcConvert von IfcOpenShell-Projekt anhand eines Beispielprozesses vom Bohrfahlfahrerfahren als die Schnittstelle für den Informationsaustausch zwischen dem BIM-Modell und dem DES-Modell entwickelt.

Die Evaluierung erfolgt anhand der Durchführung einer Simulationsstudie. Es lassen sich zwei Szenarien mit unterschiedlichen Produktionssteuerungen, die Pull- und die Push-Steuerung, unterscheiden. Für beide Szenarien wird die Fahrzeit der Betonmischer und die Anzahl der Betonmischer durch die Parameterstudien in der Simulation auf aktuellen Datengrundlagen von BIM simuliert, um letztlich den optimalen Maschineneinsatz sowie die Produktionssteuerung computergestützt zu berechnen. Zur Verifizierung des DES-Modells werden Robustheitsprüfungen mithilfe des künstlich erzeugten Fahrzeugstaus für beide Szenarien durchgeführt. Daneben wird die Sequenzierung nach dem Push-Prinzip integriert und mithilfe der Parameterstudie durchgeführt. Hierfür werden die optimalen logistischen Faktoren, wie beispielsweise die Anzahl der Betonmischer und das Steuerungsprinzip, variiert, die zu einem besseren Produktionsverhalten führen.

Die vorliegende Arbeit zeigt somit drei Schwerpunkte auf:

1. Es wird ein Konzept zur Integration von BIM-Daten in ein DES-Modell anhand des Spezialtiefbaus aufgezeigt und implementiert.
2. Es wird ein modulares DES-Modell für den Spezialtiefbau entwickelt, was zwischen den Haupt- und Nebenprozessabläufen unterscheidet und insbesondere die Arbeitsschritte der Maschinen abbildet.
3. Es wird eine Simulationsstudie auf Datenbasis eines realen Anwendungsfalls durchgeführt, um eine optimale Maschineneinsatzplanung sowie verschiedene Produktionsstrategien zu testen, und um als Robustheitsprüfung das DES-Modell zu verifizieren.

## 1.3 Vorgehensweise

Der Fokus liegt in der vorliegenden Arbeit auf zwei grundsätzlichen Vorgehensweisen.

### 1.3.1 IFC-Schema

Die wesentliche Vorgehensweise für die Entwicklung von 4D- und 5D-Modellen wird in Abbildung 1-1 gezeigt. Diese Vorgehensweise basiert auf der objektorientierten Modellierung (OOM), die die drei aufeinander aufbauenden Prozesse, die objektorientierte Analyse (OOA), das objektorientierte Design (OOD) und die objektorientierte Programmierung (OOP), beinhaltet.

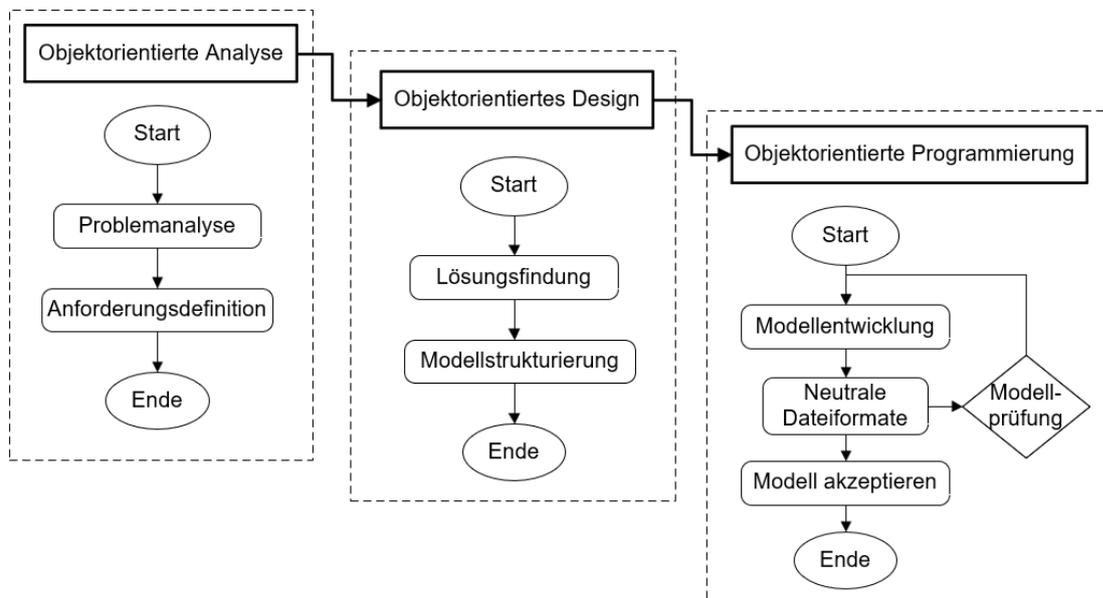


Abbildung 1-1: Vorgehensweise für die Entwicklung von 4D- und 5D-Modellen in Anlehnung an [Tus-2014, Val-2019]

In der OOA-Phase wird das Problem analysiert und anschließend werden die Anforderungen definiert. Ziel der OOD sind die Lösungsfindung und dabei die Modellstrukturierung [Tus-2014]. In der letzten OOP-Phase wird das 4D- oder 5D-Modell zunächst mithilfe einer Software entwickelt. Im nächsten Schritt wird die Validierung durch den Model Checker durchgeführt. Nach erfolgreicher Validierung wird es akzeptiert, sonst muss das Modell wieder modifiziert und validiert werden.

### 1.3.2 Simulationsmodell

Zur Durchführung einer Simulationsstudie eignen sich unterschiedliche Vorgehensmodelle. Ein mögliches ist in Abbildung 1-2 dargestellt, das in fünf Hauptphasen gegliedert wird: Problemdefinition, Systemanalyse, Modellierung, Simulationsexperimente und Auswertung. [Kud-2000].

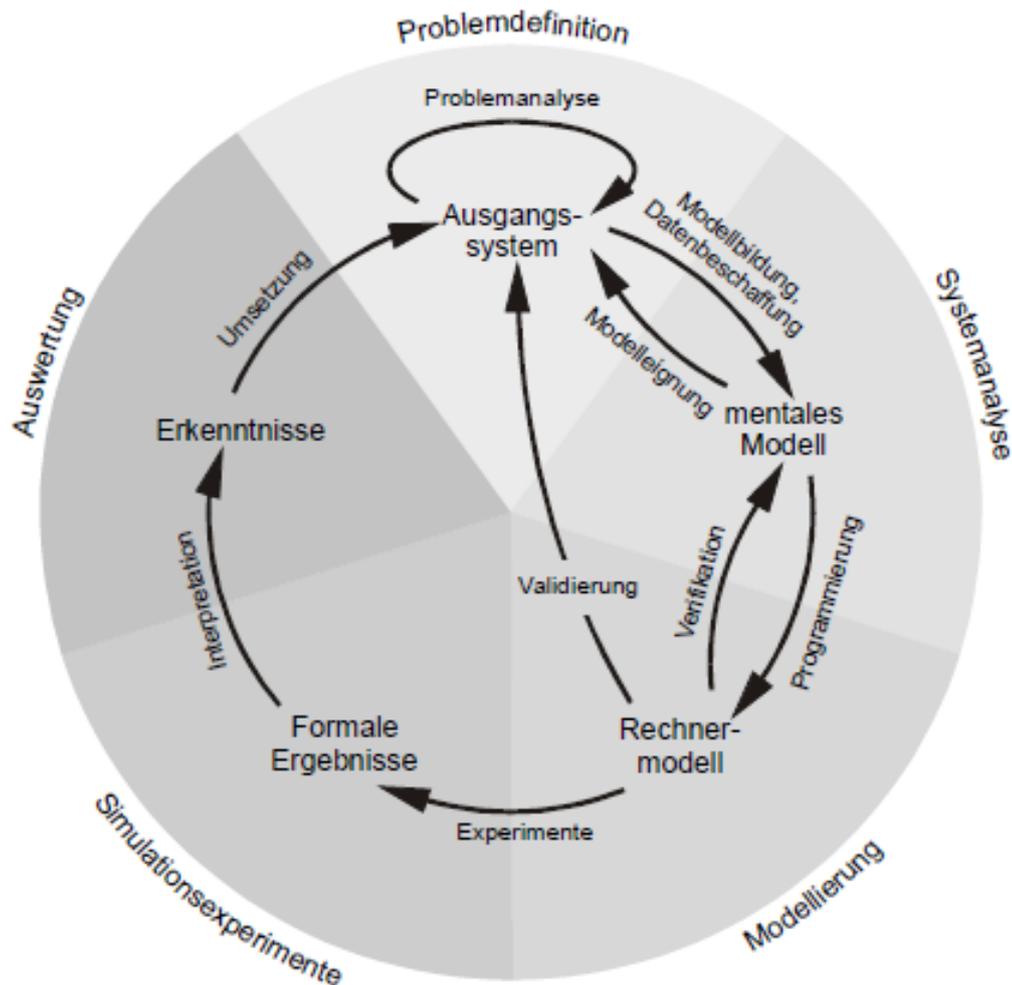


Abbildung 1-2: Simulationsvorgehensmodell [Kud-2000]

Aus der Problemanalyse entstehen eindeutige und quantifizierbare Ziele. Art und Umfang der erforderlichen Systemdaten und Elemente sowie deren Wechselwirkungen resultieren aus der Aufgabenspezifikation und dem Ausgangssystem [Kud-2000].

Um Manipulation und ungewollte Verfälschung von Daten zu vermeiden, müssen diese systematisch und möglichst objektiv erhoben, entsprechend aufbereitet und anschließend

in Modelle überführt werden [Gün-2011]. Für die parallel zur Datenbeschaffung erfolgende Modellbildung spielen die Reduktion und Abstraktion des Systems und der Daten eine entscheidende Rolle [Kud-2000]. Bei der Reduktion werden irrelevante Objekte oder spezifische Eigenschaften von Elementen vernachlässigt. Das mentale Modell entsteht und muss gleichzeitig auf seine Eignung überprüft werden [Gün-2011].

In der nächsten Phase wird das mentale Modell im Rechnermodell umgesetzt, welches nicht nur mit dem mentalen Modell zu vergleichen (Verifikation), sondern auch mit dem Ausgangssystem abzugleichen (Validierung) ist. Damit wird ein korrektes Modellverhalten gewährleistet [Gün-2011]. Die Variationsparameter müssen für spätere Experimente beachtet werden [Kud-2000].

Auf Basis des Rechnermodells erfolgen die Planung und die Durchführung der Versuche hinsichtlich der Zielparameter. Experimente werden als gezielte empirische Untersuchungen des Modellverhaltens durch wiederholte Simulationsläufe mit systematischer Parametervariation betrachtet [VDI-2000]. Besonders zu berücksichtigen ist, dass aussagekräftige Ergebnisse im Hinblick auf das Experimentdesign entstehen. Für die Zielgrößenoptimierung liegen verschiedene Methoden und Algorithmen vor [Gün-2011].

Im letzten Schritt werden die Ergebnisse ausgewertet. Die Maßnahmen für die reale Ausführung lassen sich anschließend ableiten. Hierzu können die Zielgrößen im Diagramm über statische Methoden abgebildet werden, um die Daten zu verdichten. Es ist weiterhin möglich, die Materialströme im Materialflussdiagramm darzustellen [Gün-2011].

In der vorliegenden Arbeit wird das Brückenbauprojekt „WTRO“ für die Verifikation und Validierung verwendet. Die Simulation wird am Beispiel des Kellybohrverfahrens entwickelt.

## **1.4 Gliederung der Arbeit**

Zu Beginn dieser Masterarbeit wird in Kapitel 2 auf den Stand der Technik eingegangen. Zunächst werden in Kapitel 2.1 die Grundlagen der Simulationsstudie erläutert. Dabei wird die DES im Bauwesen im Vergleich mit anderen Modellierungsansätzen ausführlich beschrieben und die Optimierungsansätze wird bezüglich des Maschineneinsatzs erläutert.

Anschließend wird in Kapitel 2.2 auf die Integration von Baufortschrittsdaten in die Simulation in Bezug auf die Photogrammetrie und die Videoanalyse sowie die Sensordaten eingegangen. Im nächsten Kapitel 2.3 wird BIM erörtert. Darin werden die Einordnung zu Vorgängerprojekten und die Begriffserklärung von BIM beschrieben. Weiterhin werden die Grundlagen von Industry Foundation Classes (IFC) in Kapitel 2.4 vorgestellt.

In Kapitel 3 werden die für die Modellierung des IFC-Formats benötigten Technologien erläutert. Dazu wird auf Programmiersprache, Bibliotheken und Software für BIM eingegangen und diese verglichen. Anschließend wird in Kapitel 4 die Implementierung der Modellierung des IFC-Formats anhand des UML-Diagramms mithilfe des xBIM-Toolkits und des Tools IfcConvert im Programm MVS eingehend beschrieben und erläutert. Im Kapitel 5 wird die Simulationsstudie im Rahmen des Brückenbauprojekts „WTRO“ auf Basis von BIM beschrieben. Die Vorgehensweisen für die Datenbeschaffung, Datenaufbereitung und Modellierung werden nacheinander erläutert. Die Ergebnisse der Simulationsstudie werden ebenfalls diskutiert und ausgewertet.



## 2 Stand der Wissenschaft und Technik

### 2.1 Ereignisdiskrete Simulation im Bauwesen

#### 2.1.1 Modellierungsansätze

Simulation wird als ein Nachbilden eines Systems mit seinen dynamischen Prozessen in einem experimentierfähigen Modell bezeichnet. Damit können die gewonnenen Erkenntnisse auf die Wirklichkeit übertragen werden [VDI-2000]. Abbildung 2-1 stellt eine Einteilung der Simulationsarten dar. In statischen Simulationen spielt die Zeit keine wichtige Rolle, während sich in dynamischen Simulationen verschiedene Zustände unter Berücksichtigung der Zeit ergeben [Gün-2011]. Die dynamische Simulation kann sich in kontinuierliche und diskrete Simulation gliedern. Bei der kontinuierlichen Simulation ändern sich die Modellzustände stetig während des Zeitverlaufs, wohingegen bei der diskreten Simulation die Änderungszustände sprunghaft zu bestimmten Zeitpunkten erfolgen [Fra-2011]. Für die diskrete Simulation ist weiterhin zwischen zeitgesteuerter und ereignisgesteuerter Simulation zu unterscheiden.

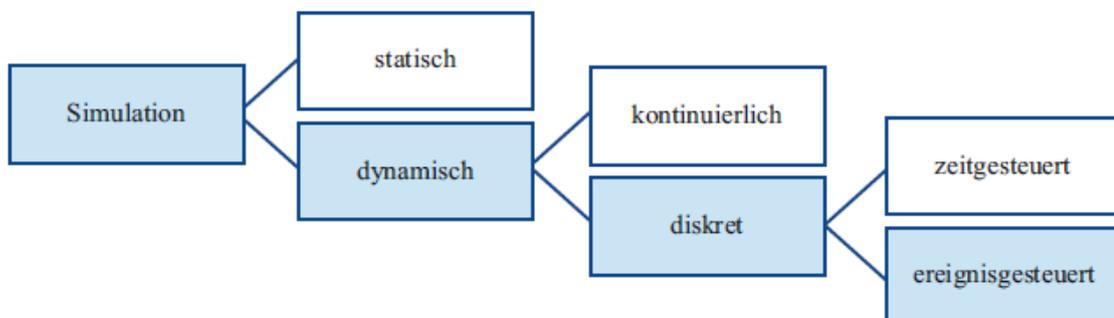


Abbildung 2-1: Klassifikation von Simulationsmodellen [Gün-2011]

Discrete Event Simulation (DES), agentenbasierte Modellierung (ABM) und System Dynamics (SD) werden im Bauwesen als Modellierungsansätze angewendet, um das Verhalten von komplexen und stochastischen Systemen zu simulieren. Ersteres wird in den Forschungsprojekten überwiegend angewendet [Abd-2020]. Außerdem wird neben den drei Methoden die hybride Simulation, wie beispielsweise ABM mit DES, diskutiert [Ban-2013]. Diese vier Modellierungsansätze werden im Folgenden betrachtet.

## Ereignisdiskrete Simulation

Für die Abbildung und Planung von Prozessen im Bauwesen kommt die dynamische diskrete ereignisgesteuerte Simulation, auch als ereignisdiskrete Simulation oder „Discrete Event Simulation“ (DES) bezeichnet, zum Einsatz [Gün-2011]. Bei der ereignisdiskreten Modellbildung handelt es sich um die Ereignisse, Aktivitäten und Prozesse, wie in Abbildung 2-2.

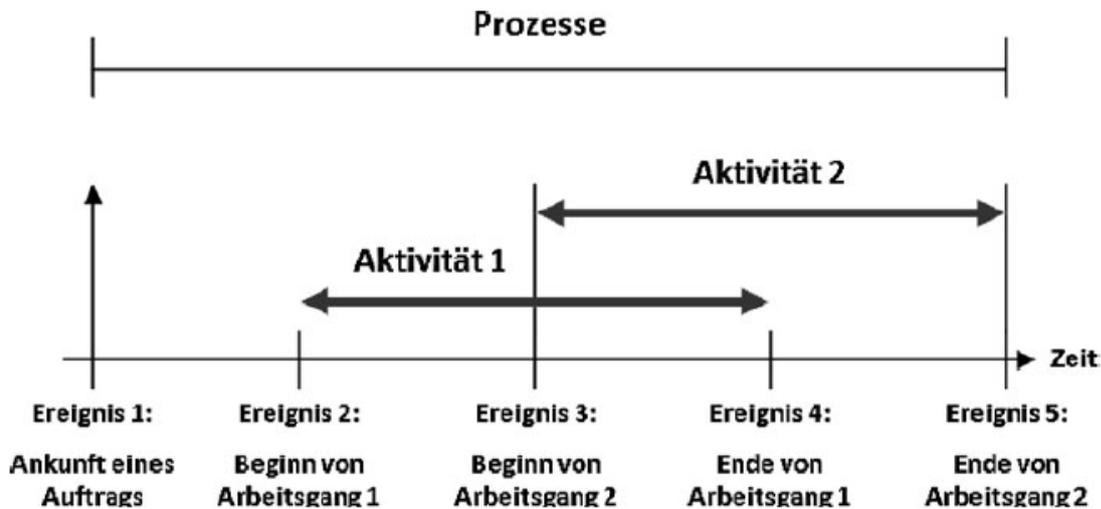


Abbildung 2-2: Prozess, Aktivität und Ergebnisse [Fis-1973]

Des Weiteren wird die DES als die Standard-Simulationsmethode im Bauwesen betrachtet und ist somit Bestandteil der Digitalen Fabrik [Alk-2019, Kuh-2006]. Dazu wird DES als flexibles Werkzeug mit umfangreichen Anwendungen angesehen. Sie ist in der Lage, dass die komplexesten Systeme mit verschiedenem Detaillierungsgrad mithilfe dieses Werkzeugs dargestellt werden; dabei werden die stochastischen Elemente zur Verfügung gestellt. Im Vergleich zum ABS und SD wird DES bevorzugt eingesetzt, wenn das System über einen hohen Unsicherheitsgrad oder viele stochastische Prozesse verfügt. Außerdem kann das reale System nachgebildet werden, indem die Datensammlung von Prozessabläufen, Prozesszeiten und Nachfragemustern durchgeführt wird. Dadurch kann eine nützliche Schätzung der realen Systemleistung im Rahmen der verschiedenen Szenarien geliefert werden [Gal-2018]. Hierfür wurde am Lehrstuhl fml eine Bausteinbibliothek für den Tiefbau in der Simulationsumgebung Plant Simulation von Siemens PLM entwickelt.

Die Abbildung 2-3 zeigt exemplarisch den Prozessbaustein (engl. process building bricks) unter Berücksichtigung des Baggers während des Erdabtragens [Wim-2014]. Dabei han-

delt es sich um die drei Bereiche des Zustandsautomaten, des Ressourcenhandlings und der Visualisierung. Im Zustandsautomaten durchläuft die sämtliche Informationen des Auftrags enthaltende Prozessmarke nacheinander unterschiedliche Elementartätigkeiten. Dabei wird ein neuer Zustand durch jeweils das Eintreten und das Vollenden aktiviert. Einzelne Zustände lassen sich sowohl mehrfach durchlaufen als auch überspringen. Das ist von der Parametrierung und den aktuellen Randbedingungen des Prozessbausteins im Gesamtumfeld abhängig. Für das Ressourcenhandling kann die angeforderte Ressource, wie Material, Personal, Geräte und Flächen, jeweils für einen bestimmten Tätigkeitstyp reserviert und wieder freigegeben werden. Des Weiteren wird zusätzlich die Funktionalität für die Visualisierung durchgeführt [Gün-2011].

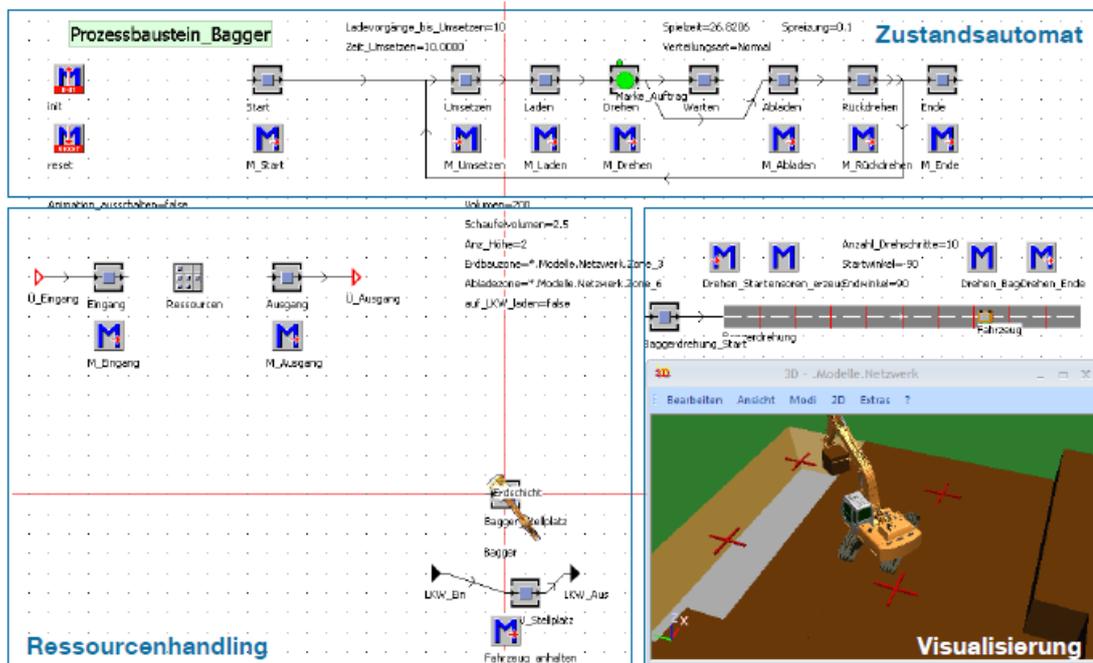


Abbildung 2-3: Aufbau eines Prozessbausteins [Gün-2011]

In der vorliegenden Arbeit wird die Simulationsstudie mithilfe der DES im Rahmen des Brückenbauprojekts „WTRO“ durchgeführt. Der Prozessbaustein unter Berücksichtigung der Betonmischer während des Betonierens wird in Tecnomatix Plant Simulation mithilfe eines Flussdiagramms erstellt.

## Agentenbasierte Modellierung

Agentenbasierte Modellierung (ABM) ist aus einer Vielzahl von Agenten zusammengesetzt, die miteinander und zwischen einzelnen Agenten sowie mit ihrer Umgebung unter Befolgung von bestimmten, möglichst soziologisch plausiblen Verhaltensannahmen interagieren, siehe Abbildung 2-4. Demnach wird das emergente Verhalten der Individuen (Agenten) und des Systems analysiert [Nos-2015, Gal-2018].

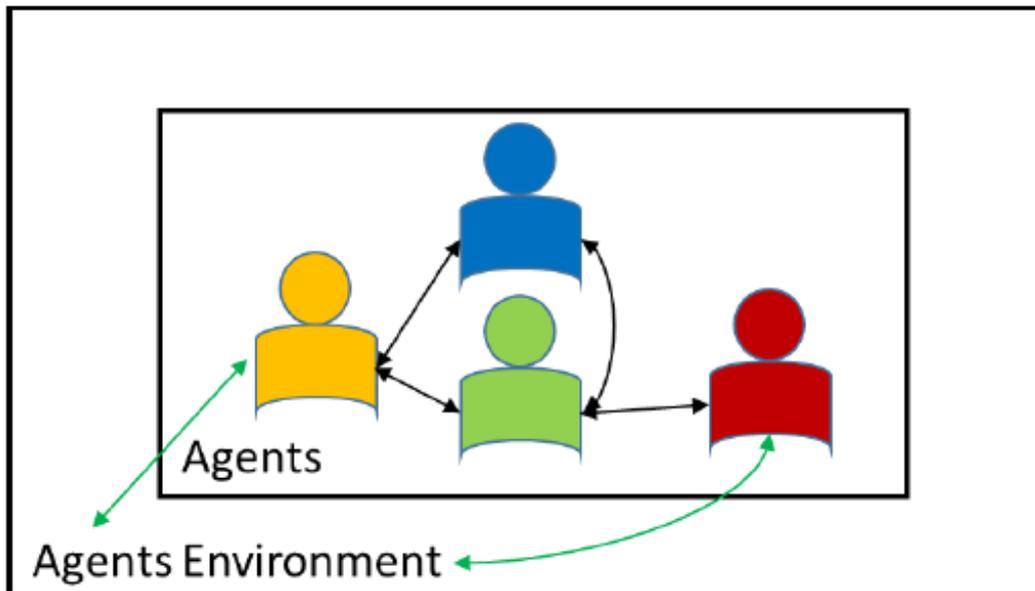


Abbildung 2-4: Interaktion heterogener Agenten [Mit-2016]

Im Gegensatz zu DES und SD wird ABM als ein Bottom-up-Modellierungsansatz angesehen, in dem die Modellelemente nach der Untersuchung des Prozesses als Ganzes aufgebaut werden. Hierfür lässt sich das System aus der Perspektive von individuellen Agenten beschreiben, nachdem ein gewisser Einblick in individuelles Verhalten aus Sicht der beteiligten Agenten gewonnen wird. Die Verhaltensmuster werden dadurch nachgebildet, dass die individuellen Agenten, die in einem dynamischen adaptiven System miteinander interagieren, dargestellt und manipuliert werden. Es wird damit untersucht, wie das Verhalten der Individuen auf der Mikroebene zu einem Systemverhalten auf der Makroebene führen kann, dadurch kann es zu Emergenzen kommen [Gal-2018, Ben-2015, Nos-2015]. Agenten lassen sich strukturell als Tupel aus Verhaltensweisen, statischen Attributen, dynamischen Attributen, einer Nachbarschaft zu anderen und Mechanismen zum Update des internen Zustands betrachten. Mithilfe von ABM wird jede Form von Heterogenität in der Simulation, beispielsweise die unterschiedlichen Attribute oder Verhaltensweisen, relativ leicht

modelliert, die die Interaktion mit der Umgebung und mit anderen Agenten steuern können [Jac-2017, Jab-2017]. Ein Emergenz-Phänomen kann verschiedene Eigenschaften zur Verfügung stellen und damit können ein hinreichendes Verständnis und akzeptable Prognosen für das Systemverhalten durch Modelle geliefert werden [Nos-2015]. Des Weiteren kann die ABM als die diskrete Methode, die kontinuierliche Methode oder eine Kombi-Methode der beiden Methoden betrachtet werden [Jab-2017].

## **System Dynamics**

System Dynamics (SD) wird als ein wissenschaftlicher Top-Down-Modellierungsansatz zur Analyse des Verhaltens von komplexen Systemen im Zeitablauf beschrieben [Nos-2015]. Das Ziel des SD-Ansatzes ist es, das System mithilfe qualitativer und quantitativer Modelle zu beschreiben und zu verstehen, wie Rückkopplungsstrukturen das Systemverhalten bestimmen. Aus dem durch Simulationen Erlernten lassen sich die Rückkopplungsstrukturen aktiv schaffen und Entscheidungsregeln ableiten [Kap-2016]. Darüber hinaus werden die qualitativen und quantitativen Modelle nicht nur mithilfe der Computersimulation ausgewertet, sondern auch das Experimentieren virtuell erlaubt [Wil-2012].

SD-Modelle stellen ihre Flexibilität und ihre Interdisziplinarität als ihre größten Stärken zur Verfügung. Die Systemzusammenhänge werden durch Lager (engl. stocks), Raten (engl. flows) und Hilfsgrößen beschrieben, mit denen sich mit Differentialgleichungen Wirkungsketten im System simulieren lassen [Mar-2012]. Die Prinzipien der Rückkopplungen (engl. feedback), der Lager und Raten spielen eine wesentliche Rolle für den SD-Ansatz, siehe Abbildung 2-5. Bei den Rückkopplungen können die negativen oder positiven Auswirkungen auf die den Einfluss nehmende Variable zurückkommen und das führt zur Dämpfung oder Verstärkung des Effekts. Lager werden als unabhängige Effektquellen angesehen, die mit einer bestimmten Rate pro Zeiteinheit eine oder mehrere Modellelemente beeinflussen. Außerdem ist es möglich, verschiedene Szenarien auf ihre Entwicklung hin mithilfe der SD zu testen [Nas-2018, Nos-2015].

Des Weiteren handelt es sich mithilfe der SD um eine kontinuierliche Systembildung und Simulation. Diese wird realisiert, indem der Systemzustand in kurzen gleichmäßigen Zeitabständen aktualisiert wird. Dadurch kommt es zu einer nahezu parallelen Aktualisierung aller

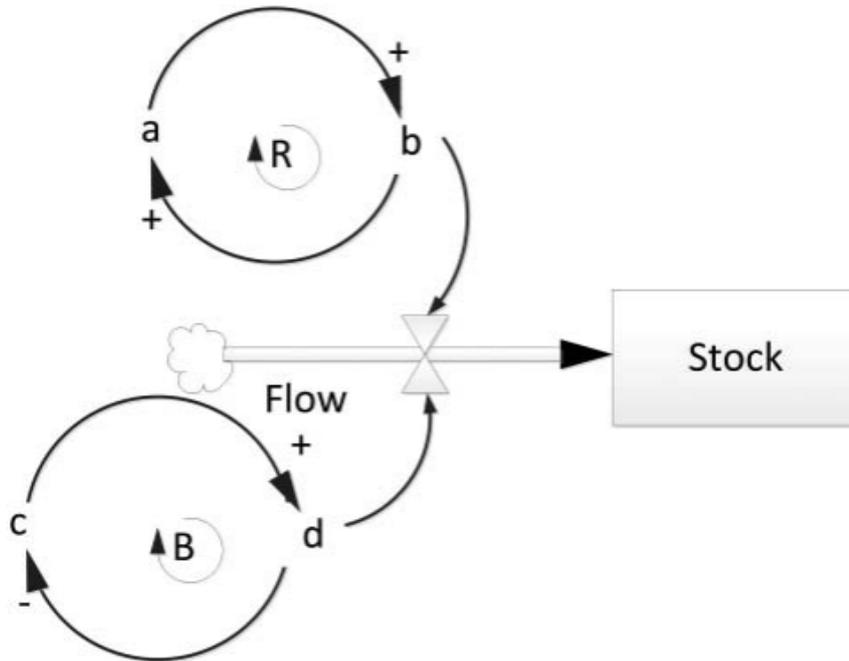


Abbildung 2-5: Lager, Rate und Rückkopplungen in SD [Nas-2018]

Parameter des Systems. Hierfür wird SD insbesondere für die Simulation von den Systemen angewendet, die durch die ständigen, harmonischen Änderungen und die wechselseitigen Abhängigkeiten charakterisiert werden [Hol-2007].

### Hybride Simulation

Unter hybrider Simulation wird die Kombination von unterschiedlichen Modellierungsansätzen verstanden. Auf Basis der hybriden Simulation wird es ermöglicht, dass das System in der Simulationsstudie nicht nur auf der Makroebene, sondern auch auf der Mikroebene betrachtet wird [Xu-2018]. Es kann mithilfe der hybriden Simulation zum umfassenden Verständnis des komplexen Systems führen, indem das Problem aus verschiedenen Dimensionen und Perspektiven untersucht wird [Eld-2016]. Es ist möglich, ein genaueres und umfassenderes Modell mithilfe der hybriden Simulation zu erstellen [MAT-2018]. Außerdem lässt sich der grundlegende Modellierungsansatz mit anderen Methoden, wie beispielsweise Fuzzy-Logik, kombinieren [Aza-2011, Sha-2005, Sad-2015]. Dazu kann die Fuzzy-Logik mit der DES kombiniert und damit ein Fuzzy-DES Modell erstellt werden, für das die Fuzzy-Nummer als Eingangsparameter dienen wird. Wenn keine historische Daten für die Anpassung der Wahrscheinlichkeitsverteilung in der Fuzzy-DES-Simulation zur Verfügung

stehen, kann die Fuzzy-Nummer die Dauer des Teilprozesses im Bauwesen repräsentieren [Sad-2015, Rao-2016]. Darüber hinaus ist es möglich, dass die Fuzzy-Logik in die ABM integriert wird, damit die subjektive Unbestimmtheit von Begriffen in das Modell eingebunden wird. Weiterhin können die Unsicherheiten durch die Integration der Fuzzy-Logik in die SD erfasst werden [Rao-2016].

## **2.1.2 Einordnung der Optimierungsansätze**

### **Push-Steuerung**

Das Push-System wird als ein Bestellsystem definiert, bei dem Bestellungen zur Fertigstellung zu bestimmten Fälligkeitsterminen auf der Grundlage der geschätzten Vorlaufzeiten ausgegeben werden [BS-1993].

Bei der Push- oder Bring-Steuerung werden sowohl der zu erwartende Ressourcenverbrauch als auch die jeweiligen Bedarfszeitpunkte durch das Logistikmanagement prognostiziert. Auf Basis dieser Pläne werden nicht nur sämtliche Ressourcenflüsse durch das Logistiksystem geplant, sondern auch die Ressourcen von einer Aufgabenerfüllung zur nächsten „gedrückt“, damit diese fixen Ablaufpläne miteinander verzahnt werden [Kla-2008].

Darüber hinaus produziert eine vorausgehende Maschine Teile in einem Push-System, ohne auf eine Anfrage der nachfolgenden Maschine zu warten [Bon-1999]. Das führt in der Regel zu hohen Lagerbeständen oder Prozessblockaden, wie beispielsweise aufgrund der übersteigenden Arbeitszeiten die Erwartungen oder Erfüllung späterer Prozessanforderungen, zu langen Arbeitszeiten, zu Schwierigkeiten der Umstellung auf andere Produkte sowie hohen Planungskosten [KIM-2007]. Zudem führt das zu Störungen im Produktionsprozess bei jeder unvorhergesehenen Abweichung [Kla-2008].

### **Pull-Steuerung**

Weiterhin wird das Pull-System als ein Bestellsystem betrachtet, bei dem von jedem Artikel ein fester Lagerbestand gehalten wird und Bestellungen für den sofortigen Ersatz von Artikeln erteilt werden, die aus dem Lager genommen werden [BS-1993].

Im Gegensatz zur Push-Steuerung wird der Produktions- bzw. Transportauftrag bei der Pull- oder Hol-Steuerung aufgrund eines konkreten Bedarfs durch den Nachfolgeprozess ausgelöst [Gün-2011]. Das bedeutet, die vorausgehende Maschine produziert in einem Pull-System lediglich dann Teile, wenn sie eine Anfrage von der nachfolgenden Maschine erhält [Bon-1999]. Die Produktion erfolgt entsprechend dem tatsächlichen Bedarf der Folgearbeiten und die Ware wird innerhalb kurzer Zeit nach Auftragseingang produziert. Dies erfordert eine effiziente Logistik, einen schnellen Informationsfluss und eine schnelle Informationsübertragung. Gleichzeitig bietet es die Vorteile geringer Lagerbestände, kurzer Lieferzeit, reduziertem Planungsaufwand und Bedarfsorientierung [KIM-2007].

In der vorliegenden Arbeit wird die Simulationsstudie im Szenario mit Push- und Pull-Steuerung durchgeführt.

### **2.1.3 Ereignisdiskrete Simulation als digitaler Zwilling im Bauwesen**

Der digitale Zwilling (engl. digital twin) kann als das visuelle Abbild eines real existierenden Gegenstands angesehen werden [Klo-2020]. Mit dem digitalen Zwilling werden mehrere unterschiedliche Simulationsmodelle mit unterschiedlichem Detaillierungsgrad über die gesamte Lebensdauer durchgespielt. Dabei wird es ermöglicht, dass der Systemzustand genau dargestellt und zuverlässige Prognosen getroffen werden können, indem die Sensor-, System- und Wartungsdaten und die aufgezeichneten historischen Daten kontinuierlich gesammelt und in Echtzeit an den digitalen Zwilling des Systems übertragen werden [Sch-2019].

Darüber hinaus lässt der digitale Zwilling sich auf Basis der ereignisdiskreten Simulation im Bauwesen realisieren. Er ist für das Konzept der Digitalen Fabrik von entscheidender Bedeutung. Digitaler Zwilling eines Bauwerks wird sowohl dessen statische als auch dynamische Abbildung, um einen effizienten und ressourcenschonenden Bau mit hoher Genauigkeit, stärkerer Kontrolle und verbesserter Vorhersehbarkeit bestimmter Ereignisse zu erstellen [Hub-2019].

Als Ausgangslage für den digitalen Zwilling sind Daten des realen Objekts zu betrachten. Dabei wird zwischen geometrischer und nichtgeometrischer Information unterschieden. Als

geometrische Information gilt eine Ist-Aufnahme eines realen Objektes und deren Darstellung als Punktwolke oder photogrammetrische Aufnahme. Demnach hat die dreidimensionale, objektbasierte Darstellung für den digitalen Zwilling besondere Bedeutung. Neben der geometrischen Information kommen nichtgeometrische Informationen zum Einsatz, wie beispielsweise in Form von Eigenschaften. In Bezug auf das zu erstellende Bauwerk wird dadurch ermöglicht, planerischen Vorhersagen zu machen. Hierbei handelt es sich um die Visualisierung, die Simulation von Zuständen oder die Kontrolle von geplanten Abläufen. Ein digitaler Zwilling mit den Echtzeitdaten des Bauwerkes, wie beispielsweise dem Internet der Dinge (engl. Internet of Things, kurz IoT), erzielt weitere Verbesserungen. Hierbei lässt sich das reale Bauwerk bei Erstellung und Betrieb echtzeitlich abbilden. Während der Erstellung ist es möglich, den Baufortschritt automatisch mit dem digitalen Zwilling abzugleichen und damit Abweichungen zu erkennen. Das führt zur Darlegung der Folgen und Darstellung der möglichen Problemlösungen [Hub-2019].

Im Fokus der vorliegenden Arbeit liegt die DES. Sie wird als ein leistungsfähiges Werkzeug beschrieben, um den Bauprozess zu bewerten. Dabei kann zwischen Mikro- und Makrosimulation unterschieden werden. Über die Makrosimulation können die Bauprozessabläufe unter den vorliegenden Einschränkungen und Abhängigkeiten auf einer Baustelle angeordnet werden. Dadurch wird es ermöglicht, die Bauzeitterminierung und die realen Bauzeiten vorherzusagen. In der Mikrosimulation geht es um einzelne begleitende Bauprozesse. Dabei können Empfehlungen für den nächsten Schritt gegeben werden [Ji-2020]. Mithilfe der DES werden die Operationen oder die Vorhersagen auf der Mikroebene optimiert. Dagegen fokussiert SD das Strategiemangement auf der Makroebene für die Modellierung und die Simulation; damit werden eine Gesamtbetrachtung und die Analyse des gesamten Bauprojekts angestrebt. Daneben handelt es sich bei der ABM um eine Untersuchung der Individuen auf der Mikroebene zu einem Systemverhalten auf der Makroebene. Außerdem wird das System in der hybriden Simulation auf beiden Ebenen betrachtet [Bra-2014].

## **2.2 Integration von Baufortschrittsdaten in die Simulation**

Um die Terminplanung zu unterstützen, gibt es eine weitere Möglichkeit zur detaillierten Modellierung der einzelnen Prozesse. Im Rahmen des Forschungsprojekts FAUST des Lehr-

stuhls fml werden die automatisierte Erfassung, Verarbeitung und Integration von Daten aus der Bauausführung als Grundlagen betrachtet. Damit können Vorhersagen des weiteren Baufortschritts gemacht und die Simulationsergebnisse visualisiert werden. Darüber hinaus wurde untersucht, welche Daten von der Baustelle online zur Verfügung stehen müssen. Damit lassen sich im Störfall mögliche Lösungen simulieren. Bei den Baufortschrittsdaten handelt es sich um sowohl Produktionsdaten (Leistungsbeschreibung, Geometrie, Menge) als auch Prozessdaten (Zeit, Kosten, Ressourcen). In diesem Unterkapitel wird die Umsetzung der Integration der verschiedenen Bauschrittsdaten ausführlich beschrieben, wie beispielsweise Photogrammetrie und Videoanalyse sowie Sensordaten [Gün-2015, Sch-2020, Kes-2020].

### **2.2.1 Photogrammetrie und Videoanalyse**

Im Rahmen von FAUST entwickelte Bügler et al. [Büg-2017] den Ansatz aus der Kombination von photogrammetrischem Verfahren und Videoanalyse und dadurch wurde der Baufortschritt der Erdarbeiten bewertet. Dabei wurde die Punktwolke erstellt, damit das Volumen des ausgehobenen Bodens gemessen wurde, und LKW-Kipper auf Bildern wurden durch Foreground Detection erkannt. Im Rahmen der automatisierten Baufortschrittskontrolle werden Methoden und Techniken am Lehrstuhl für Computergestützte Modellierung und Simulation der TUM entwickelt. Diese basieren auf der Integration der Punktwolkeninterpretation und der 4D-Bauwerkmodellierung. Im Forschungsvorhaben progressTrack wird die automatisierte Baufortschrittsdatenerfassung auf der Baustelle gefördert [Bor-2021].

### **2.2.2 Sensordaten**

Über den Verband Umwelt- und Maschinentechnik e. V. (VDBUM) der Baubranche und den amerikanischen Anwenderverband „Association of Equipment Management Professionals“ (AEMP) mit japanischer Unterstützung ließ sich die ISO 15143-3 [ISO-2020] als Standard für die Baumaschinentelematikdaten von Erdbewegungs- und Straßenbaumaschinen schaffen. Diese beruhen auf dem Forschungsprojekt BauFlott. Um eine Erweiterung dieser Norm auf weitere Baumaschinen anzustreben, wurde zusätzlich die Arbeitsgemeinschaft „Machines in Construction – MiC 4.0“ (MiC 4.0) gegründet. Hier werden nicht nur die Zu-

standsdaten generiert und erfasst, sondern auch die Prozess- und Qualitätsdaten zugeordnet [VDM-2021]. Daneben sollen die Produktionsdaten der Maschinen erfasst werden. Hierfür ist es notwendig, dass eine geeignete Sensorik zum geeigneten Einsatz kommt [Fis-2020].

Ziel des Forschungsprojekts AutoBauLog ist die intelligente Vernetzung sämtlicher Baumaschinen auf einer Baustelle, um die Leistungserfüllung durch Wahrnehmung, Bewertung und Optimierung der übertragenen Aufgaben zu verbessern. Hierfür werden Baumaschinen mit softwarebasierter Intelligenz und Sensorik eingesetzt. Dabei war es möglich, mit den weiteren Maschinen zu kooperieren und damit die Maschinenteam abzubilden [Jac-2011, Sie-2011]. Darüber hinaus werden Parameter wie Messung von Bohrparametern und Temperatur mithilfe der Sensortechnologie im Spezialtiefbau erfasst [Sch-2016]. Heutzutage kommen moderne, mit Sensoren ausgestattete Drehbohrgeräte zum Einsatz, um die Bohrparameter zu erfassen, wie beispielsweise Vorschubkraft, Vorschubgeschwindigkeit, Drehzahl und Druck am Bohrantrieb in Abhängigkeit der Bohrtiefe [Sch-2009, Bau-2021]. Mit dem dynamischen Positionierungssystem wird die Position über dem Bohrloch gewährleistet. Dabei kommt es zum Abgleich mit dem Bohrplan [Fis-2020]. Die historischen Prozessdaten werden von Kargul et al. [Kar-2015] ausgewertet. Daraus ergibt sich die Dauer für die herstellenspezifischen Prozesse und die repräsentativen Wahrscheinlichkeitsdichtfunktionen werden abgeleitet. Diese Dauer wird als Eingangsgröße für die Simulation betrachtet.

## **2.3 Building Information Modeling (BIM)**

Building Information Modeling (BIM) wird nicht nur als computerbasiertes Planen und Bauen angesehen, sondern damit wird auch das einzelne Modell und die Modellierung des gesamten mehrdimensionalen Informationsraums beschrieben. In der vorliegenden Arbeit wird die Integration von BIM in die DES durchgeführt [Sch-2014].

### **2.3.1 Einordnung zu Vorgängerprojekten**

Im Forschungsprojekt ForBAU wurden die Daten aus den ERP-Systemen (Enterprise Resource Planning System) der Bauunternehmen für den Erdbau bereits eingebunden und

diese Einbindung wird aufgegriffen [Gün-2010]. Im Projekt Mefisto wurden die mit der prozessbasierten Arbeitsweise kombinierte Multimodellmethode und eine Vielzahl von Datenspezifikationen, Methoden und Softwaresystemen entwickelt, damit die Fachmodelle in Multimodellen zusammengeführt, ausgetauscht, geprüft, visualisiert und transformiert sowie in Simulations- und Analysemodellen weiterführend verwendet wurden. In den letzten Jahren hat sich im Bauwesen BIM als standardisiertes Format für die modellbasierte Abbildung von Bauprojekten etabliert. Für die Integration der Produktionsinformationen spielt das BIM eine wichtige Rolle [Sch-2014].

### **2.3.2 Begriffserklärung**

BIM bezeichnet „ein umfassendes digitales Abbild eines Bauwerks mit großer Informationstiefe“ [Bor-2015]. Dabei handelt es sich um die durchgängige Verwendung des digitalen Modells über alle Lebenszyklusphasen eines Bauwerks (siehe Abbildung 2-6) [Bor-2015]. Im Mittelpunkt liegt ein neben geometrischen und topologischen auch semantischen Informationen umfassendes BIM, womit das digitale Bauwerksmodell gemeint ist [Sin-2016]. Die digitalen Fachmodelle werden als disziplinspezifische digitale Bauwerksmodelle über alle Bauphasen hinweg beschrieben. Damit wird ausgedrückt, dass BIM als eine objektorientierte Methode im Bauwesen betrachtet werden kann. Deren Ziel ist es, durch die stärkere Kooperation mit den Projektbeteiligten eine optimale Planung und Ausführung sowie die Auswahl des bestmöglichen Betriebs zu gewährleisten [Tus-2014].

Mit der BIM-Methode werden alle wesentlichen Gebäudedaten digital erfasst und vernetzt. Dabei entsteht in den überwiegenden Fällen ein dreidimensionales digitales Gebäudemodell. Aus diesem lassen sich eine präzise Mengenermittlung und Pläne oder Kostenkalkulationen ermitteln [Tus-2014]. Dennoch ist BIM mehr als eine 3D-Konstruktion und dabei entstehen die Bezeichnungen 4D, 5D und XN BIM. Bei dem 4D-Modell handelt es sich um ein 3D-Modell, das mit den Terminplänen verknüpft wird, um den Bauablauf darzustellen. Die Abhängigkeit der Prozesse von den Ressourcen wie Menschen, Maschinen und Material wird in die Simulation integriert [Bor-2015]. Somit lässt sich die zeitliche Planung der Baustelle nicht nur kontrollieren, sondern auch optimieren. Außerdem entsteht ein 5D-Modell durch die Verbindung des 4D-Modells mit den Kosten. Das 5D-Modell ermöglicht

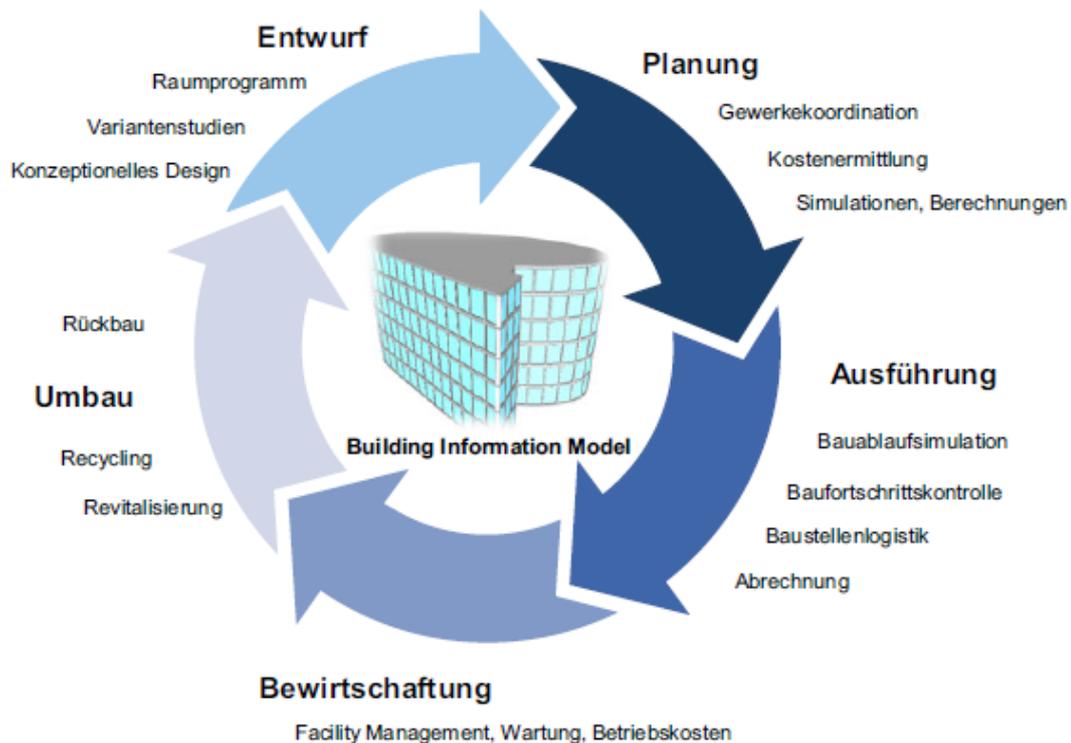


Abbildung 2-6: Building Information Modelling über den gesamten Lebenszyklus eines Bauwerks [Bor-2015]

eine Kostenverlaufssimulation und eine Simulation der Material- und Personalganglinien [VDI-2018].

Der Ausarbeitungsgrad des BIM-Modells wird als „Level of Development“ (LOD) bezeichnet, der mit LOG (Level of Geometry) die geometrische und mit LOI (Level of Information) die alphanumerischen Modellinformationen beinhaltet und entlang der Planungs- und Ausführungsphasen mit dem steigenden Informationsgehalt in den BIM-Stufen 100 bis 500 erfolgt [Wis-2018]. Auf dem Positionspapier Spezialtiefbau vom Hauptverband der Deutschen Bauindustrie wird LOI am Beispiel von Bohrpfählen beschrieben [HDB-2019]. Mit laufendem Projektfortschritt steigen die Granularität und Genauigkeit [VDI-2018].

## 2.4 Industry Foundation Classes (IFC)

Um die Realisierung von weitreichender und nachhaltiger Interoperabilität im Bauwesen gewährleisten zu können, kommt nun ein herstellernerutrales, offenes Datenformat zum Einsatz, das in buildingSMART IFC getauft wurde. Hierfür kann das IFC als ein herstellerunabhängiges, offenes, standardisiertes und objektorientiertes Datenaustauschformat zur

Beschreibung von digitalen Gebäudemodellen im gesamten Lebenszyklus eines Bauwerks angesehen werden. Die Umsetzung von Big Open BIM basiert damit stark auf dem IFC [Bor-2015].

### 2.4.1 Historische Entwicklung

Seit der ersten Veröffentlichung von Version IFC 1.0 im Jahr 1997 wurden in rascher Folge Revisionen und Erweiterungen des Modells von der buildingSMART entwickelt (siehe Abbildung 2-7). Derzeit verbreitet sich die Version IFC 2X3 am weitesten und wurde im Jahr 2014 durch die IFC4 abgelöst [Bor-2015].

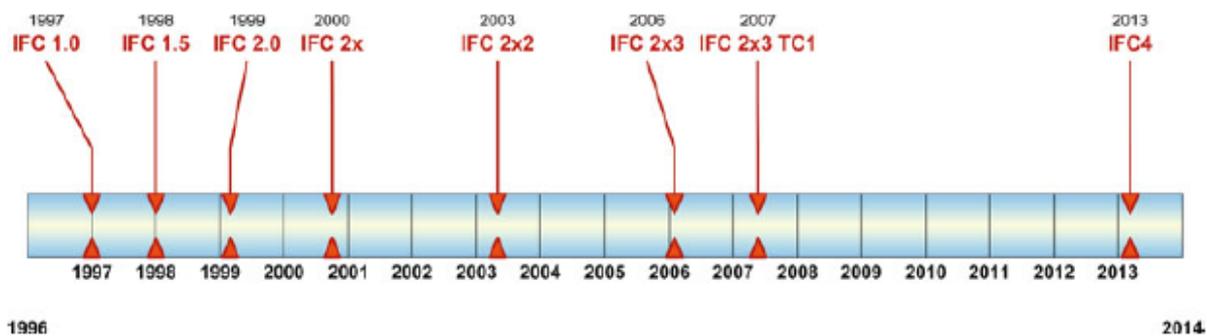


Abbildung 2-7: Versionsgeschichte des IFC-Modells [Bor-2015]

### 2.4.2 Datenformate

Das IFC-Format ist als Standardformat der IFC auf Basis der ISO-Standardisierung STEP (Standard for the Exchange of Product Model Data) beschrieben. EXPRESS ist im STEP-Standard definiert und wird vor allem als die Datenmodellierungssprache betrachtet [ISO-2020]. Das IFC-Format in der vorliegenden Arbeit wurde durch das xBIM-Toolkit entwickelt. Generell wird eine IFC-Datei in zwei Abschnitte unterteilt, den HEADER-Abschnitt für Dateiinformationen und den DATA-Abschnitt für Produktinformationen. Es handelt sich um die dateiinternen Objektidentifikatoren (#-Zeichen) und das Globally Unique Identifier im STEP P21 Dateiformat. Ein Globally Unique Identifier oder kurz GUID wird nicht nur als ein global eindeutiger 128-Bit-Wert bezeichnet, sondern auch bevorzugt bei der Datenbankprogrammierung eingesetzt. Außerdem enthält das resultierende IFC-GUID 22 Zeichen [Eil-2010]. Weiterhin beinhaltet der Header die Datenbeschreibung (FILE\_DESCRIPTION),

die Angaben zu Dateinamen, Datum und Zeit, Ersteller sowie verwendeter Software(FILE\_NAME) und das IFC-Schema (FILM\_SCHEMA). Für die vorliegende Arbeit wird die Version IFC4 angewendet [Bor-2018].

Im Rahmen der IFC4-Entwicklung entwickelte sich das neue XML-basierte Datenaustauschformat ifcXML. Darüber hinaus sind ifcXML-Dateien aufgrund der XML-Syntax ca. 2-3-mal größer als die IFC-Dateien bei der ifcXML-Konvertierung in IFC4, jedoch gewinnt das Format ifcXML zukünftig deutlich an Bedeutung [Bor-2015].

### 2.4.3 Die Vererbungshierarchie

Die Vererbungshierarchie spielt im objektorientierten IFC-Datenmodell eine übergeordnete Rolle. Es handelt sich um Spezialisierungs- bzw. Generalisierungsbeziehungen. Sie werden festgelegt und somit auch, welche Attribute von welchen Klassen vererbt werden.

#### Bauteilklassen

In der folgenden Abbildung 2-8 wird ein Ausschnitt aus der Vererbungshierarchie mit Fokus auf die wichtigen Bauteilklassen dargestellt, wie beispielsweise IfcWindow, IfcWall, IfcBeam und IfcColumn. Abstrakte Basisklassen stehen zur Verfügung, wie IfcRoot, IfcObject, IfcActor, IfcProcess, IfcProduct, IfcProject, IfcRelationship. Dabei wird IfcProduct als die wesentliche Klasse des IFC-Datenmodells angesehen und es geht um die Subklassen, wie beispielsweise IfcBuilding, IfcBuildingStorey, IfcSpace, IfcElement, IfcBuildingElement, IfcOpeningElement und die Beziehungsklassen IfcRelAssociatesMaterial, IfcRelFillsElement und IfcRelVoidsElement. In der vorliegenden Arbeit handelt es sich um die Bauteilklass IfcPile unter der Subklasse IfcBuildingElement [Bor-2015]. Das IfcPile-Objekt wird sowohl als Pfahl als auch als Tiefgründung bezeichnet [Bui-2021].

Aus der Abbildung 2-8 ist zu erkennen, dass die Klasse **IfcRoot** nicht nur der Ausgangspunkt und die Wurzel des Vererbungsbaums und von IfcRelationship ist, sondern auch eine Superklasse für die drei Klassen IfcObjectDefinition, IfcPropertyDefinition und IfcRelationship bildet. IfcRoot beinhaltet grundlegende Funktionalitäten, um mithilfe eines GUIDs zu identifizieren, um die Eigentümerschaft und Herkunft eines Objekts zu beschreiben sowie

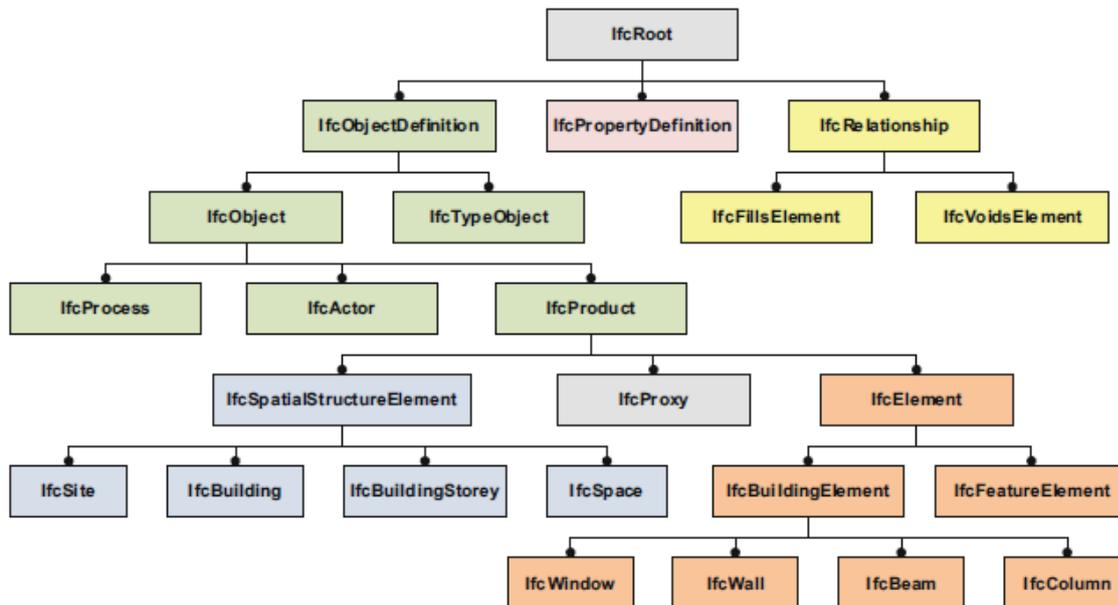


Abbildung 2-8: Ausschnitt aus dem IFC-Datenmodell mit den wichtigsten Entitäten (engl. Entity) der obersten Ebenen der Vererbungshierarchie [Bor-2015]

um die Modifikationshistorie abzubilden, wie beispielsweise die Identität des Erstellers und weiterer Bearbeiter sowie die Versionsgeschichte. Darüber hinaus lassen sich jedem Objekt ein Name und eine Beschreibung zuordnen. Diese drei Klassen werden als die nächste Stufe in der Vererbungshierarchie bezeichnet und von `IfcRoot` direkt abgeleitet [Bor-2015].

Die Klasse **IfcObjectDefinition** wird als die abstrakte Superklasse aller Klassen unter Berücksichtigung der Definitionen für die Beschreibung der Beteiligten am Bauvorhaben beschrieben, die physische Objekte (Bauteile usw.), Raum-Objekte (Ausparungen, Spaces) oder konzeptionelle Elemente (Prozesse, Kosten usw.) repräsentativ darstellen. Es handelt sich um die drei Subklassen der Superklasse `IfcObjectDefinition`, wie `IfcObject` (einzelne Objekte des Bauvorhabens), `IfcTypeObject` (Objekttypen) und `IfcContext` (allgemeine Projektinformation). Darüber hinaus wird von einem `IfcObject` ein individuelles Objekt als Teil eines Bauprojekts repräsentiert. `IfcObject` bildet die Superklasse für sechs wichtige Klassen des IFC-Datenmodells: (1) `IfcProduct`, (2) `IfcControl`, (3) `IfcProcess`, (4) `IfcResource`, (5) `IfcActor` und (6) `IfcGroup`. Hierbei werden die sogenannten Subklassen entsprechend dem prinzipiellen Herangehen an die Modellierung von Geschäftsprozessen in Bereiche, wie Produkt, Prozess, Kontrollelement und Ressource, aufgeteilt [Bor-2015].

In der vorliegenden Arbeit steht **IfcProduct** im Fokus, das sowohl als ein physisches Objekt

als auch als ein Raumobjekt bezeichnet wird. Darüber hinaus lässt sich IfcProdukt-Objekten eine geometrische Repräsentation zuordnen. Daneben befinden sie sich im Raum des Koordinatensystems. Außerdem wird IfcProduct als die abstrakte Repräsentation aller Objekte angesehen. Es bildet die Superklasse für die Subklassen IfcElement, IfcProxy und IfcSpatialStructureElement. Die Subklasse IfcElement bildet die Superklasse einer ganzen Reihe von wichtigen Basisklassen und die Bauteilklassen wie IfcWall, IfcColumn, IfcBeam und IfcWindow. Unter der Superklasse IfcSpatialStructureElement liegen die Subklassen IfcSite (Grundstück), IfcBuilding (Gebäude), IfcBuildingStorey (Geschoss) und IfcSpace (Raum) [Bor-2015, Bui-2021].

Durch die Klasse **IfcPropertyDefiniton** werden die Eigenschaften eines Objektes definiert. Das führt zu keiner festen Verankerung mit den Eigenschaften im IFC-Modell und bildet die Superklasse für die Subklasse IfcProperty. Neben den statisch definierten Attributen werden zusätzlich die dynamisch erzeugbaren Eigenschaften (engl. properties) im IFC-Modell definiert. Solche Eigenschaften sind mithilfe der Subklassen der IfcProperty (meist IfcPropertySingleValue) frei zu definieren und in beliebiger Anzahl im Instanzmodell hinzuzufügen [Bor-2015].

**IfcProperty** bildet eine Superklasse für die Klassen IfcComplexProperty und IfcSimpleProperty. Prinzipiell werden die komplexen Eigenschaften von IfcComplexProperty definiert, die innerhalb eines Eigenschaftssatzes verwaltet werden können. Dagegen repräsentiert ein IfcSimpleProperty ein einzelnes Objekt, dem die generalisierten Eigenschaften zugewiesen sind. Es bildet die abstrakte Superklasse für sechs wichtige Klassen des IFC-Datenmodells [Bui-2021]:

- **IfcPropertyListValue:** Einer Eigenschaft wird ein einziger Wert zugeordnet. Er wird aus den Elementen Name, NominalValue, Type und Unit zusammengefasst.
- **IfcPropertyEnumeratedValue:** Einer Eigenschaft wird ein Wert aus einer Nummerierung zugeordnet. Er besteht aus den Elementen Name, Value (EnumerationValue), Type und IfcPropertyEnumeratio .
- **IfcPropertyBoundedValue:** Die zwei Grenzwerte sind einer Eigenschaft zugewiesen. Er besteht aus Name, UpperBoundValue, LowerBoundValue, Type und Unit.
- **IfcPropertyTableValue:** Die zwei Wertelisten werden einer Eigenschaft zugeordnet. Er wird aus DefiningValues, DefiningValueType und DefingUnit zusammengefasst.

- **IfcPropertyReferenceValue** : Ein Wert wird der Eigenschaft durch die Verbindung mit den komplexen Eigenschaften zugeordnet.
- **IfcPropertyListValue**: Mehrere Werte, die durch eine genaue und geordnete Liste angegeben werden, sind einer Eigenschaft zugewiesen. Es gibt eindeutige und keine doppelten Werte. Er besteht aus Name, ListValue, Type und Unit .

In dieser Arbeit werden die wichtigsten Eigenschaften am Beispiel des Kellybohrverfahrens im Rahmen des Brückenbauprojekts „WTRO“ durch das xBIM-Toolkit im IFC-Modell definiert. Das wird in Kapitel 4 ausführlich beschrieben.

## Objektbeziehungen

Des Weiteren beziehen sich die Klasse IfcRelationship und ihre Subklassen auf die Abbildung objektifizierter Beziehungen (engl. objectified relationship). Das führt zur Entkopplung der Semantik der Beziehung von den Objektattributen und dadurch wird ermöglicht, beziehungspezifische Eigenschaften direkt am Beziehungsobjekt zu speichern. Das wird im Folgenden ausführlich erläutert [Bor-2015].

Objektbeziehungen sind für das IFC-Datenmodell von großer Bedeutung. In der Tat wird ein Alleinstellungsmerkmal des IFC-Standards durch die mächtigen Funktionen zur detaillierten Beschreibung von Beziehungen gebildet. Nach dem Prinzip objektifizierter Beziehungen kommt es zu aus semantischer Sicht relevanten Beziehungen zwischen Objekten nicht über eine direkte Assoziation im IFC-Datenmodell, sondern mithilfe eines gesonderten, dazwischengeschalteten Objekts, von dem die Beziehung selbst repräsentiert wird. Daneben handelt es sich um ein wichtiges Prinzip der in den IFC umgesetzten Datenmodellierung. Nach dem Prinzip beziehen sich die Vorwärtsbeziehungen (die definierten Attribute) auf das Beziehungsobjekt und zeigt auf die Objekte, die in Beziehung gesetzt werden. Beziehungsobjekte werden immer als Instanzen einer Subklasse von IfcRelationship bezeichnet. In Abbildung 2-9 wird der Vererbungsbaum der Objektbeziehungen gezeigt [Bor-2015].

Im Folgenden werden die grundlegenden Funktionen im IFC-Datenmodell erklärt [Bui-2021, Bor-2015, Sch-2014]:

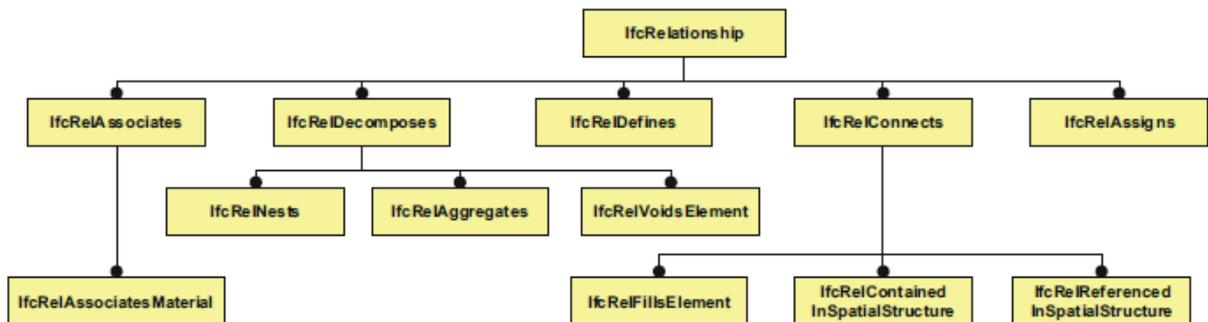


Abbildung 2-9: Die Vererbungshierarchie der Beziehungsklassen des IFC-Datenmodells [Bor-2015]

- **IfcRelAssociates:** bezieht sich auf externe Informationsquellen (wie Klassifikationen, Bibliotheken oder Dokumente). Die assoziierten Informationen können sich innerhalb oder außerhalb der Projektdaten befinden. Es gibt keine Abhängigkeit, die durch die Assoziation impliziert wird.
- **IfcRelDecomposes:** definiert das generelle Konzept von Objekten, die sich zusammensetzen oder zerlegen lassen. Die Dekompositionsbeziehung bezeichnet eine Teil-/Ganzes-Hierarchie mit der Möglichkeit, vom Ganzen zu den Teilen zu navigieren und andersherum. Sie bildet die Superklasse für drei wichtige Klassen des IFC-Modells: IfcRelNests (Teile haben eine Reihenfolge), IfcRelAggregates (Teile haben keine Reihenfolge) sowie IfcRelVoidsElement (Aussparungsbeziehung).
- **IfcRelDefines:** verknüpft eine Objekt-Instanz durch die Anwendung einer Property-Set-Definition oder einer Typ-Definition, die die IFC-Konzepte anbieten.
- **IfcRelConnects :** beschreibt eine Konnektivitätsbeziehung, die Objekte nach bestimmten Kriterien miteinander verbindet.
- **IfcRelDeclares:** behandelt die Deklaration von Objekten (Subtype von IfcObject) oder Eigenschaften (Subtype von IfcPropertyDefinition) zu einem Projekt oder einer Projektbibliothek (repräsentiert durch IfcProject, oder IfcProjectLibrary).
- **IfcRelAssigns:** beschreibt eine generalisierte Verknüpfungsbeziehung zwischen den Instanzen von IfcObject. Subklassen sind beispielsweise IfcRelAssignsToProduct (Zuordnung von Objekten zu einem Produkt), IfcRelAssignsToControl (Zuordnung eines Controls zu anderen Objekten).

## Terminplan

Das BIM-Modell kann als initiale Datenquelle für die Einrichtung eines Baustellenmanagementsystems betrachtet werden. Darüber hinaus wird es ermöglicht, den Baufortschritt auf der Baustelle selbst im Modell zu visualisieren. Das heißt, dass ein sogenanntes As-Built-Modell zu jedem Zeitpunkt der Bauphase erzeugt werden kann. Hierfür ist es erforderlich, dass die einzelnen Arbeitsgänge mit den entsprechenden BIM-Objekten auf Datenverarbeitungsebene (DV) verbunden werden können, um den Baufortschritt im BIM-Modell zu visualisieren. Auf der DV kommt es zur Informationsspeicherung in Instanzen sogenannter Unterklassen [Dal-2018]. Die Unterklassen `IfcTask` und `IfcProcess` spielen eine wichtige Rolle für die die Bauausführung betreffenden Informationen. `IfcProcess` bildet die Superklasse für die Subklasse `IfcTask` und als ein im Rahmen eines Bauvorhabens (Planung, Bau, Betrieb) auftretender Vorgang. Für die Prozesse steht eine zeitliche Ausdehnung zur Verfügung [Bor-2015].

Dazu wird das **IfcTask**-Objekt als eine identifizierbare Arbeitseinheit im Bauprojekt bezeichnet und wurde in der vorliegenden Arbeit im BIM-Modell mithilfe vom XBIM-Toolkit eingeführt [Bui-2021]. Außerdem ist es möglich, dass das IFC-Objekt über das Beziehungsobjekt mit einem `IfcTask`-Objekt verknüpft wird, siehe Abbildung 2-10. Dadurch kann ein Arbeitsschritt dem `IfcTask`-Objekt zugeordnet werden [Bra-2013]. Um den zeitgerechten Baufortschritt kontrollieren zu können, wird ein Terminplan aufgestellt [Har-2014]. Für das 4D-Modell mit den Terminplandaten wurde die Software Constructivity als Modell Checker oder Viewer angewendet. Das wird in Unterkapitel 3.3 dargestellt. Darüber hinaus bezieht sich das 4D-Modell mit den Terminplandaten auf die Objekte zur Abbildung eines Terminplans (`IfcWorkPlan`, `IfcWorkSchedule`), einzelner Vorgänge und Meilensteine (`IfcTask`) sowie deren hierarchischen Aufbau (`IfcRelNests`). Daneben werden im 4D-Modell die Objekte für die Abbildung unterschiedlicher Typen von Vorgänger- und Nachfolgerbeziehungen (`IfcRelSequence`) abgebildet [Tul-2010].

Anhand der prozessrelevanten Unterklassen wird es ermöglicht, dass den Objekten innerhalb des BIM-Modells Vorgangsdauern zugeordnet werden. Sie sind in der Lage, den Objekten in der Version IFC4 explizite Start- und Enddaten zuzuweisen. Diese hinzugekommenen Funktionalitäten werden in Abbildung 2-10 dargestellt. Dadurch können die `IfcTaskTime`-Objekte in den `IfcTask`-Objekten neu eingeführt und ihnen zugeordnet werden.

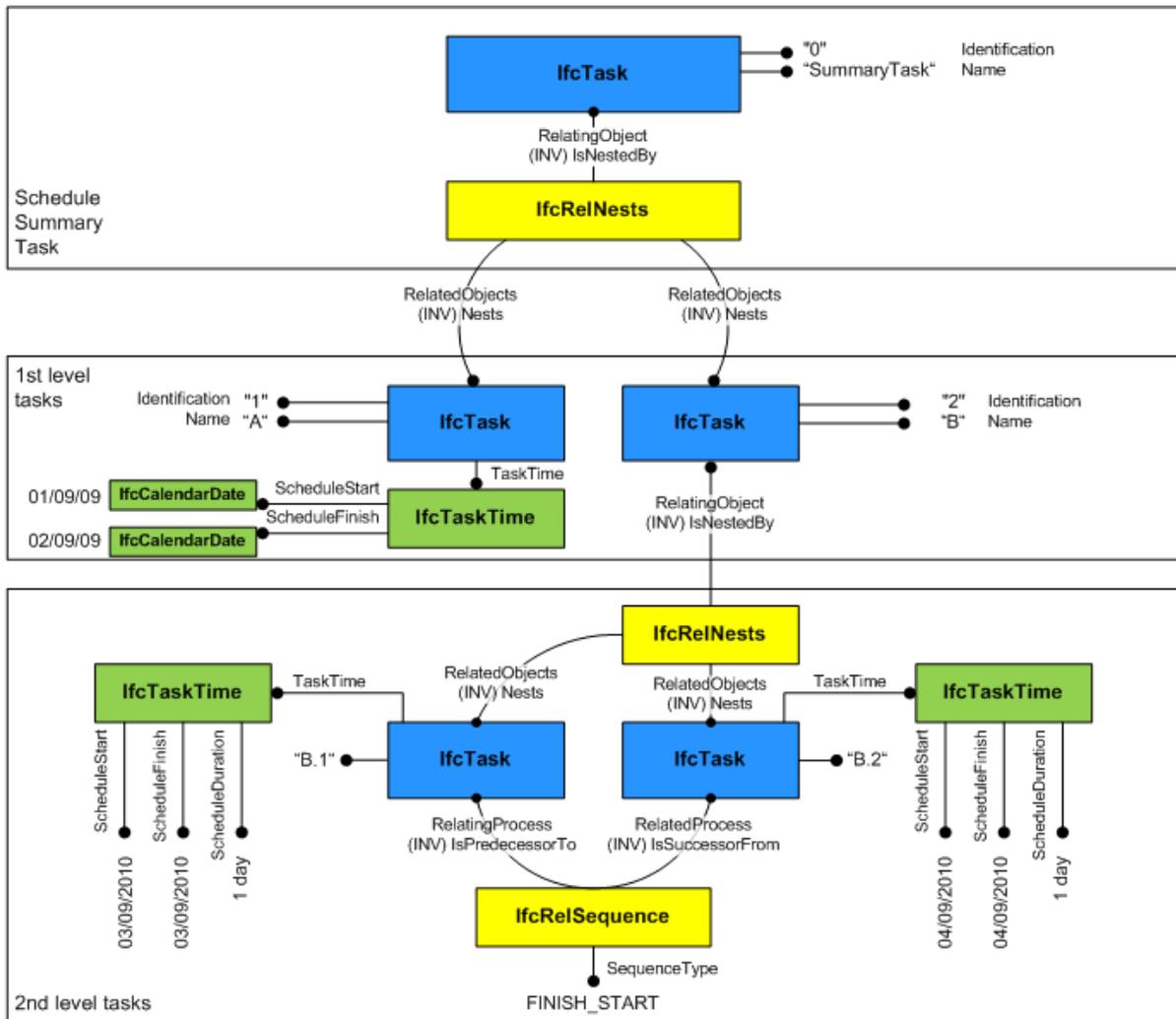


Abbildung 2-10: IFC Task Instantiation Diagram [Bui-2021]

Die **IfcTaskTime**-Objekte beinhalten die Eigenschaften, wie beispielsweise **ScheduleStart** und **ScheduleFinish**. Außerdem wird im Instanzierungsdiagramm deutlich dargestellt, wie die Arbeitsgänge mittels eines Relationsobjekts des Typs **IfcRelNests** gegliedert werden können. Im Beispiel kommt es zu zwei Ebenen (1st und 2nd level tasks) in der Abbildung 2-10 [Dal-2018].

Weiterhin können **IfcProcedure** (Prozess) und **IfcEvent** (Ereignis), die durch **IfcRelSequence** verbunden sind, durch **IfcRelNests** mit einem **IfcTask** verknüpft werden, siehe Abbildung 2-11. **IfcProcedure** wird als eine logische Reihenfolge von Tätigkeiten beschrieben, die durch das Ereignis ausgelöst werden. Daneben beschreibt **IfcEvent** ein Ereignis, das durch eine Nachricht oder eine bestimmte Information ausgelöst wird [Bui-2021].

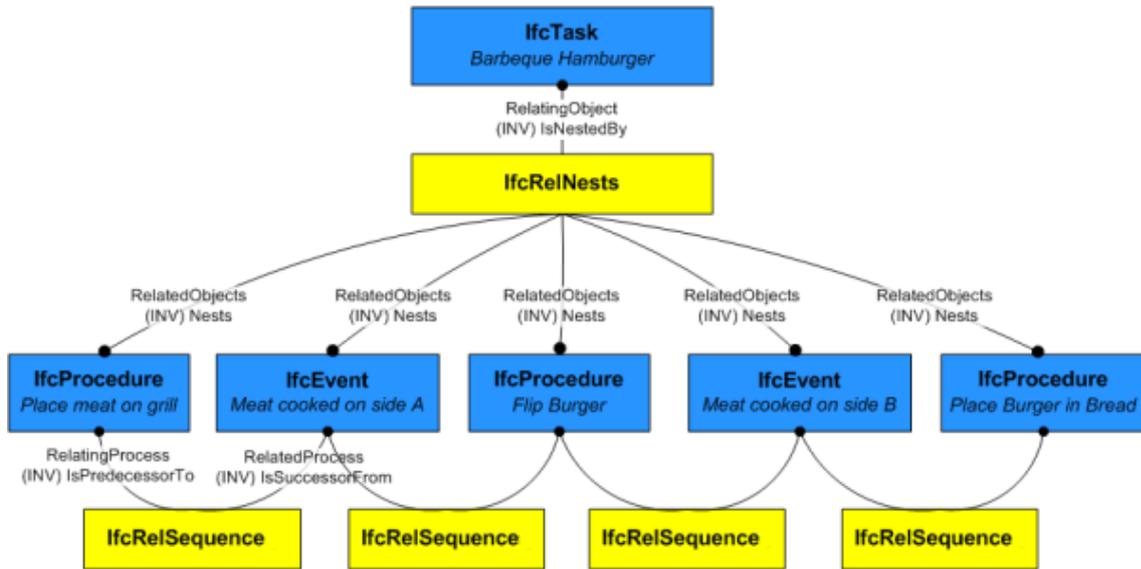


Abbildung 2-11: Prozessbeziehungen [Bui-2021]

## 3 Lösungsansätze für den Umgang mit BIM-Modellen

---

### 3.1 Programmiersprachen

Für die Entwicklung des BIM-Modells im IFC-Schema standen vier Programmiersprachen zur Auswahl: C++, C#, Python sowie Java. Diese sogenannten höheren Programmiersprachen sind aufgrund der großen Verbreitung sowohl an den Wissenschaftsbereich als auch den OpenSource-Entwicklerkreis anpassbar. Weiterhin ist es erforderlich, dass die geeignete Bibliothek bei der Anwendung von Programmiersprachen mit umfangreichen Funktionen verwendet wird. Daneben ist die Programmiersprache gut lesbar [Bor-2018, Bra-2013].

In der vorliegenden Arbeit kommt die Programmiersprache C# mithilfe des xBIM-Toolkits zum Einsatz. Die Gründe dafür werden in Kapitel 3.2 dargestellt. Dieses Modellierungskonzept wird in Kapitel 4 ausführlich erklärt.

### 3.2 Vergleich von STEP/IFC-Bibliotheken

Der Begriff Bibliothek wird als eine Sammlung nützlicher Funktionen und Klassen in einem Programm verwendet [Lou-2011]. Viele Open-Source-Bibliotheken stehen zur Verfügung. In der folgenden Tabelle 3-1 werden die verschiedenen IFC-Bibliotheken dargestellt. Hiermit wird es ermöglicht, STEP- bzw. IFC-Dateien zu lesen und zu schreiben.

Für diese vorliegende Arbeit ist eine geeignete stabile Bibliothek auszuwählen, die das neue IFC4-Format unterstützen wird, das die Abbildung des 4D-Modells mit den Terminplandaten unterstützen kann. Darüber hinaus kommen Open-Source-Projekte hinsichtlich wirtschaftlicher Aspekte zum Einsatz. Die Open-Source-Lizenz als juristische Grundlage ist von großer Bedeutung wie in der proprietären Softwareentwicklung [Sal-2005]. Es geht darum, welche Tätigkeiten dem Nutzer erlaubt sowie verboten sind, wie beispielsweise Kopieren und Verändern [Bra-2013]. Außerdem wird die Visualisierung als ein BIM-Ziel betrachtet. Dabei werden mehrere Mehrwerte erzielt, damit die Darstellung leichter verstanden und die Kommunikation vereinfacht wird [Pil-2017]. Es spielt aufgrund der Modellierung von IFC mithilfe von EXPRESS für das Bauwesen eine wichtige Rolle, die STEP-Datei zu lesen bzw.

Tabelle 3-1: Übersicht über verschiedene STEP/IFC-Bibliotheken [Bor-2018]

Name	Sprache	Lizenz	STEP	IFC	Visualisierung	Webseite
IfcPlusPlus	C++	MIT	Ja	Ja	Ja	<a href="https://github.com/ifcquery/ifcplusplus">https://github.com/ifcquery/ifcplusplus</a>
IfcOpenShell	C++/Python	LGPL	Nein	Ja	Ja	<a href="http://ifcopenshell.org/">http://ifcopenshell.org/</a>
JSDAI	Java	AGPL V3	Ja	Nein	Ja	<a href="http://www.jsdai.net/">http://www.jsdai.net/</a>
xBIM-Toolkit	C#	CDDL	Nein	Ja	Ja	<a href="https://docs.xbim.net/">https://docs.xbim.net/</a>
IFC Tools Project	Java/C#	CC	Ja	Ja	Ja	<a href="http://www.openifctools.org/Open_IFC_Tools/Home.html">http://www.openifctools.org/Open_IFC_Tools/Home.html</a>
IFC-Engine	C++/C#	Proprietary	Ja	Ja	Ja	<a href="http://rdf.bg/product-list/ifc-engine/">http://rdf.bg/product-list/ifc-engine/</a>
STEPcode	C++/Python	BSD	Ja	Ja	Nein	<a href="https://stepcode.github.io/">https://stepcode.github.io/</a>
Ifc-dotnet	C#	BSD	Nein	Ja	Nein	<a href="https://code.google.com/p/ifc-dotnet/">https://code.google.com/p/ifc-dotnet/</a>

zu schreiben [Gün-2011]. Darüber hinaus werden die STEP-Dateien in zwei Kategorien, das Early-Binding-Verfahren (frühe Bindung) und das Late-Binding-Verfahren (späte Bindung), aufgeteilt. Beim Early Binding kann eine STEP-Datei der Hostentitäten abgebildet, gelesen und geschrieben werden, d.h. in eine STEP-Datei überführt werden. Im Vergleich mit dem Early Binding wird eine STEP-Datei beim Late Binding mithilfe vom Standard Data Access Interface (SDAI) gelesen und geschrieben. Das SDAI ist als eine Programmierschnittstelle (engl. Application Programming Interface, kurz API) beschrieben.

### IfcPlusPlus

IfcPlusPlus (IFC++) basiert auf C++ und lässt sich unter MIT-Lizenz als Open Source für kommerzielle Projekte verwenden. Es ist mit IfcPlusPlus möglich, die IFC-Modelle zu visualisieren. Dazu können die STEP-Datei und die IFC-Datei lesen und schreiben. Daneben

kann die Version IFC4 mit IfcPlusPlus unterstützen [Ifc-2021].

### **IfcOpenShell**

Auf Basis von C++/Python wird IfcOpenShell als quelleoffene Bibliothek unter LGPL-Lizenz angewendet, um das IFC-Format in der Version IFC4 bearbeiten zu können. Allerdings gibt es keine Möglichkeit, die STEP-Datei, die XML-basierte ifcXML-Datei und die ifcZIP-Datei zu bearbeiten. IfcOpenShell verwendet Open-CASCADE um zu visualisieren. Allerdings können die geometrischen IFC-Eigenschaften mit IfcOpenShell nicht vollständig verstanden werden [Ifc-2021].

Weiterhin wird das Tool IfcConvert aus der IfcOpenShell-Bibliothek verwendet, um das IFC-Format in das XML-Format zu überführen. Gegenüber dem Tool IfcConvert kann das XML-basierte ifcXML-Format mithilfe des xBIM-Toolkits vollständiger konvertiert werden. Es beinhaltet zusätzlich beispielsweise die Prozessparameter (IfcTask). Allerdings führt es zur komplexeren Dateistruktur als mithilfe des Tools IfcConvert. Hierfür wird das Tool IfcConvert für die vorliegende Arbeit angewendet. Damit wird das XML-Format in die Software Plant Simulation von Siemens PLM importiert, um die Informationen in der Simulationsumgebung anzuwenden.

### **JSDAI**

MIT JSDAI wird es auf Basis von C++ unter AGPL-Lizenz ermöglicht, die STEP-Datei zu bearbeiten. Die Visualisierung und die Verarbeitung von IFC-Dateien sind mit JSDAI unmöglich. Darüber hinaus werden hiermit vier verschiedene APIs parallel unterstützt, wie Early Binding, SDAI-Dictionary, ARM-AIM-Mapping sowie Late Binding [JSD-2021].

### **xBIM-Toolkit**

Bei dem xBIM-Toolkit (eXtensible Building Information Modeling) geht es um eine .NET Open-Source-Software unter CDDL-Lizenz auf Codeplex. Mit dem xBIM-Toolkit kann die Version IFC4 unterstützt werden. Die geometrischen, topologischen, operativen und visualisierenden IFC Eigenschaften werden mithilfe des xBIM-Toolkits vollständig implementiert,

dadurch dass das BIM-Modell im IFC-Schema erstellt und bearbeitet werden kann. Im Unterschied zu IfcOpenShell können IFC-Datei, ifcXML-Datei und ifcZIP-Datei mithilfe des xBIM-Toolkits gelesen und beschrieben werden [xBI-2021].

Darüber hinaus basiert das xBIM-Toolkit auf C#. Hierfür wird das IFC-Modell im Programm MVS mit C# erstellt. Dazu muss eine Vielzahl an .NET-Bibliotheken, wie beispielsweise xBIM Essentials und xBIM Geometry, über NuGet im Programm MVS integriert werden. Deswegen wird das BIM-Modell im IFC-Schema in der vorliegenden Arbeit mithilfe visueller Programmierung in dieser Umgebung entwickelt [xBI-2021].

Zusätzlich kann das IFC-Format im WexBIM-Format durch WebGL-Plug-in von xBIM überführt werden. Dadurch kann eine Visualisierung der IFC-Datei in einem Webbrowser erfolgen [xBI-2021].

### **IFC Tools Project**

Das aus den Open IFC Tools hervorgegangene IFC Tools Project basiert auf Java und wird unter CC-Lizenz nicht-kommerziell angewendet. Dieses Projekt wird als eine eigenständig entwickelte Abspaltung (engl. fork) angesehen. Außerdem hat es die Fähigkeit, die STEP-Datei und die IFC-Datei zu bearbeiten sowie das BIM-Modell im IFC-Schema zu visualisieren. Zusätzlich gibt es C#-Unterstützung. Daneben kann die Version IFC4 bei IFC Tool Project unterstützt werden. Allerdings werden die Open IFC Tools seit 2020 nicht mehr weiter aktualisiert [IFC-2021].

### **IFC-Engine**

Die IFC-Engine wird als eine STEP-Toolbox unter Closed-Source-Lizenz mit Grafikunterstützung betrachtet. Sie basiert sowohl auf C++ als auch C#. Sie kommt für ein wissenschaftliches Projekt begrenzt zum Einsatz. Daneben wird es mit IFC-Engine ermöglicht, die STEP-Datei und die IFC-Datei zu bearbeiten, die Version IFC4 zu unterstützen sowie das BIM-Modell im IFC-Schema zu visualisieren [IFC-2021].

### **STEPcode**

STEPcode wird als Open Source angesehen und unter BSD-Lizenz für kommerzielle Projekte verwendet. Mit STEPcode gibt es keine Möglichkeit, das BIM-Modell im IFC-Schema zu visualisieren. Sie basiert auf C++ und Python. Außerdem wird die STEP-Datei und die IFC-Datei mit STEPcode bearbeitet [STE-2021].

### **Ifc-dotnet**

Ifc-dotnet ist eine Open-Source-Bibliothek unter BSD-Lizenz. Mit IFC-dotnet werden der Export und Import von STEP-Datei und ifcXML-Datei in der Version IFC2x3 unterstützt. Darüber hinaus gibt es mit IFC-dotnet keine Möglichkeit, das BIM-Modell im IFC-Schema zu visualisieren. Die Version IFC4 wird von Ifc-dotnet unterstützt und die IFC-Datei kann damit bearbeitet werden [Ifc-2021].

### **Zusammenfassung**

In diesem Abschnitt wurden die umfangreichen IFC-Bibliotheken übersichtlich dargestellt. Im Gegensatz zu den anderen Bibliotheken verfügt das xBIM-Toolkit über umfangreiche Funktionen und Unterstützungen. Darüber hinaus wird die gut lesbare Programmiersprache C# ausgewählt.

Für die Umsetzung werden das xBIM-Toolkit und das Tool IfcConvert in der vorliegenden Arbeit verwendet. Einerseits wird das BIM-Modell im IFC-Schema im Programm MVS mit C# mithilfe des xBIM-Toolkits erstellt. Andererseits wird das IFC-Format in der Version IFC4 mit Hilfe des Tools IfcConvert in das XML-Format überführt.

## **3.3 Vergleich von Softwarelösungen für die BIM-Modellierung**

Für die Entwicklung von 4D-Modellen ist die Validierung des IFC-Modells von großer Bedeutung. Das Ziel der Validierung bezieht sich darauf, sowohl in der Modellierung erstellte Daten durch Validierung für die weitere Nutzung aufzubereiten als auch an nachfolgende Softwaresysteme oder Prozessbeteiligte weiterzuleiten [Bor-2015]. Hierbei werden die

Softwarelösungen für die Validierung des BIM-Modells vorgestellt, siehe folgende Tabelle 3-2.

Tabelle 3-2: Übersicht über verschiedene Softwarelösungen für die Validierung des BIM-Modells

Name	3D Modell	4D Modell	Kosten	Webseite
Notepad++	Nein	Nein	Nein	<a href="https://notepad-plus-plus.org/">https://notepad-plus-plus.org/</a>
IFC Quick Browser	Nein	Nein	Nein	<a href="http://www.team-solutions.de/">http://www.team-solutions.de/</a>
Constructivity	Ja	Ja	Teilweise	<a href="http://www.constructivity.com/">http://www.constructivity.com/</a>
Solibri Model Checker	Ja	Ja	Teilweise	<a href="https://www.solibri.com/">https://www.solibri.com/</a>
FZK Viewer	Ja	Nein	Nein	<a href="https://www.iai.kit.edu/1302.php">https://www.iai.kit.edu/1302.php</a>
BIMcollab ZOOM	Ja	Nein	Teilweise	<a href="https://www.bimcollab.com/en">https://www.bimcollab.com/en</a>
xBIM Explorer	Ja	Nein	Teilweise	<a href="https://docs.xbim.net/downloads/xbimexplorer.html">https://docs.xbim.net/downloads/xbimexplorer.html</a>
Simplebim	Ja	Ja	Ja	<a href="http://simplebim.com/">http://simplebim.com/</a>

Hierbei gibt es die Möglichkeit, 3D- oder 4D-Modelle darzustellen. Daneben spielt der wirtschaftliche Aspekt für die Auswahl der Software eine wichtige Rolle. Hinsichtlich der verschiedenen Aspekte wird das geeignete Validierungswerkzeug ausgewählt. In der vorliegenden Arbeit wird die Software Constructivity als Model Checker ausgewählt.

### Notepad++

Die Software Notepad++ ist ein kostenloser Texteditor, um die STEP-Datei zu lesen und zu schreiben. Außerdem ist es möglich, die ifcXML-Datei und die XML-Datei zu bearbeiten, siehe Abbildung 3-1. Mit der Software Notepad++ werden das 3D-Modell und das 4D-Modell nicht gezeigt.

```

1 ISO-10303-21;
2 HEADER;
3 FILE_DESCRIPTION (('), '2;1');
4 FILE_NAME ('', '2021-05-15T10:12:54', ('), ('), 'Processor version 5.1.0.0');
5 FILE_SCHEMA (('IFC4'));
6 ENDSEC;
7 DATA;
8 #1=IFCPROJECT('2uR08VrznAd83$Pi$x_JCz',#2,'Pile',$, $, 'B15, Westtangente Roser...');
9 #2=IFCOWNERHISTORY(#5, #6, $, .ADDED, 1621073575, $, $, 0);

```

Abbildung 3-1: Aufbau der IFC-Datei mit Notepad++ in der vorliegenden Arbeit

## IFC Quick Browser

Mit der Software IFC Quick Brower kann das textuelle Durchsuchen von großen IFC-Dateien für durchgeführt werden. Hierbei wird die IFC-Datei in Textform dargestellt. Es wird mit der Software IFC Quick Browser ermöglicht, die relevanten Zeilen durch die Auswahl in einer Baumstruktur zu zeigen, siehe Abbildung 3-2.

```

ISO-10303-21;HEADER;FILE_DESCRIPTION (('), '2;1');FILE_NAME ('', '2021-05-15T10:12:54', ('), ('), 'Processor version 5.1.0.0', ' ^
#1=IFCPROJECT('2uR08VrznAd83$Pi$x_JCz',#2,'Pile', $, $, 'B15, Westtangente Rosenheim', 'Projektstatus', (#19,#22),#7);
#2=IFCOWNERHISTORY(#5,#6,$,.ADDED,1621073575,$,$,0);
#3=IFCPERSON($,'Team', 'xBIM', $, $, $, $);
#4=IFCORGANIZATION($,'fml', $, $, $);
#5=IFCPERSONANDORGANIZATION(#3,#4,$);
#6=IFCAPPLICATION(#4,'1.0', 'Pile Application', 'Pile.exe');
#41=IFCRELAGGREGATES('2YKWPwxqLDBvQebbak2Nth',#2,$,$,#1,(#36));
#201=IFCRELDECLARES('387PzuUNf3WfofHAOBRb1',#2,'CONTROL', $,#1,(#186));
#200=IFCRELDECLARES('227rj9czr0WuboYSRa8mnl',#2,'PROCESS', $,#1,(#188));

```

Abbildung 3-2: Aufbau der IFC-Datei mit IFC Quick Browser in der vorliegenden Arbeit

## Constructivity

Die Software Constructivity wird als ein Model Checker für das 3D-Modell bzw. das 4D-Modell angesehen. Es gibt die Möglichkeit zur Visualisierung von 3D- und 4D-Modellen.

Allerdings sind die Animation und die Bearbeitung des IFC-Modells nicht kostenlos. Innerhalb der Software Constructivity werden die Daten eines IFC-Modells automatisch auf ein proprietäres, internes Datenmodell vollständig abgebildet, siehe Abbildung 3-3. Das Objekt `IfcPile` verfügt über die Attribute, wie beispielsweise „Name: ‚Bohrpfahl: Durchmesser: 1.2cm‘; PredefinedType: ‚BORED‘“. Dazu werden zeitliche Daten des IFC-Modells in einem Gantt-Diagramm dargestellt, siehe Abbildung 3-4. Das erste Objekt `IfcTask` (1) ist mit dem Attribut „Name: ‚Pile(Standard) A140-Gründungspfähle‘ für den gesamten Prozess. Das `IfcTaskTime`-Objekt mit den Attributen, wie beispielsweise „ScheduleStart: ‚2019-08-13T10:24:00‘; ScheduleFinish: ‚2019-08-22T08:24:00‘“, wird in das `IfcTask`-Objekt (1) eingeführt. Das zweite Objekt `IfcTask` (2) besitzt das Attribut „Name: ‚A140.1 Gründungspfähle‘“. Es ist lediglich ein Vorgang im gesamten Prozess.

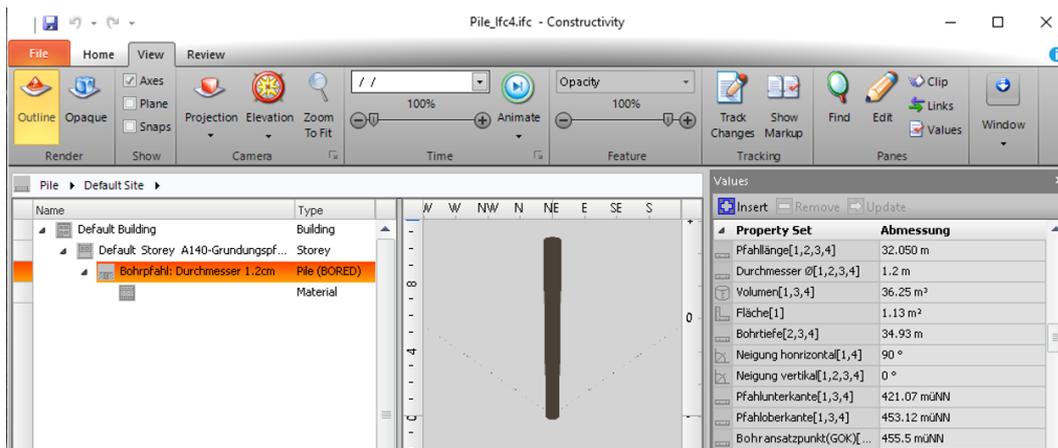


Abbildung 3-3: 4D-Modell in der Software Constructivity in der vorliegenden Arbeit

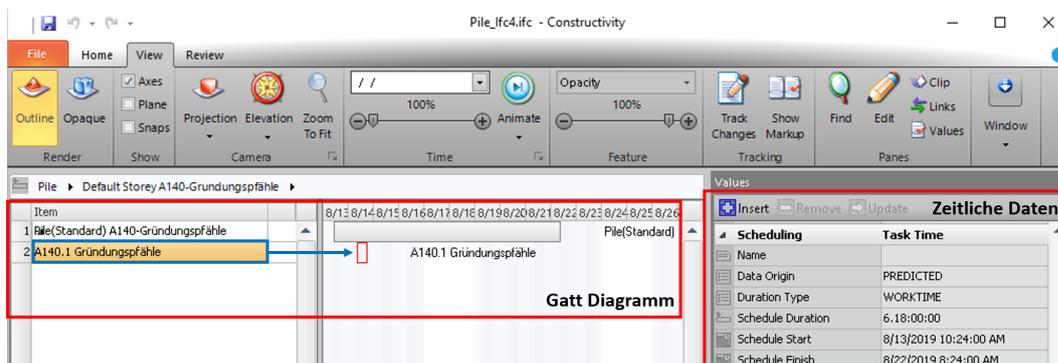


Abbildung 3-4: Darstellung des Gantt-Diagramms in der vorliegenden Arbeit

## Solibri Model Checker

Mit der Software Solibri Model Checker (SMC) wird es ermöglicht, IFC-Modelle zu visualisieren und nutzbar visuell zu kontrollieren. Die Software wird als ein Model Checker für das 3D-Modell betrachtet. Für die vorliegende Arbeit ist es unmöglich, die zeitlichen Daten zu überprüfen, siehe Abbildung 3-5. Bei dem SMS handelt es sich um die automatische Abbildung der Daten vom IFC-Modell auf ein proprietäres, internes Datenmodell. Anschließend ist es möglich, die Daten weiterzuverarbeiten und die Gestaltungsplanung automatisiert zu überprüfen. Die Funktionen sind kostenpflichtig [Bor-2015].

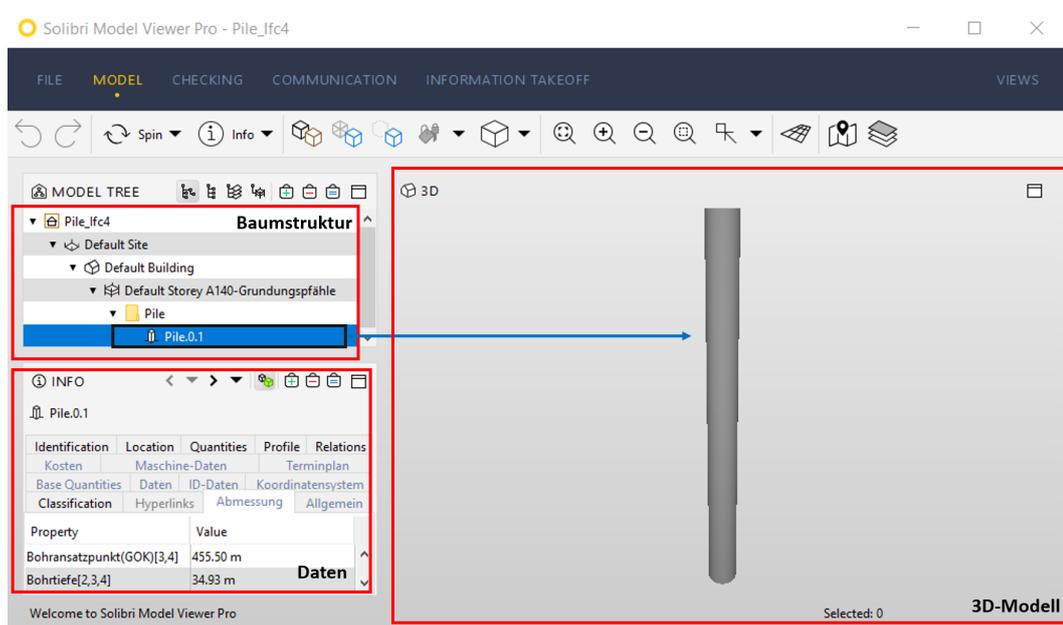


Abbildung 3-5: 4D-Modell in Solibri in der vorliegenden Arbeit

## FZK Viewer

FZK Viewer wurde am Karlsruher Institut für Technologie entwickelt. Er wird als ein Softwarewerkzeug angesehen, um standardisierte semantische Datenmodelle aus den Bereichen BIM und GIS zu visualisieren, siehe Abbildung 3-6. Durch die Funktion Anfragen (engl. query) werden die Daten übersichtlich gezeigt, wie beispielsweise Eigenschafts-Sets, siehe Abbildung 3-7. Der FZK Viewer beinhaltet die vollständige Darstellung der IFC-Objekte, wie beispielsweise IfcPile, IfcTask und IfcWorkPlan. Allerdings werden die IFC-Eigenschaften nicht vollständig dargestellt. Es handelt sich um keinen Wert, siehe Abbildung 3-7. Die Nummer [1] bedeutet die Anzahl des Objekts IfcPile im IFC-Modell.

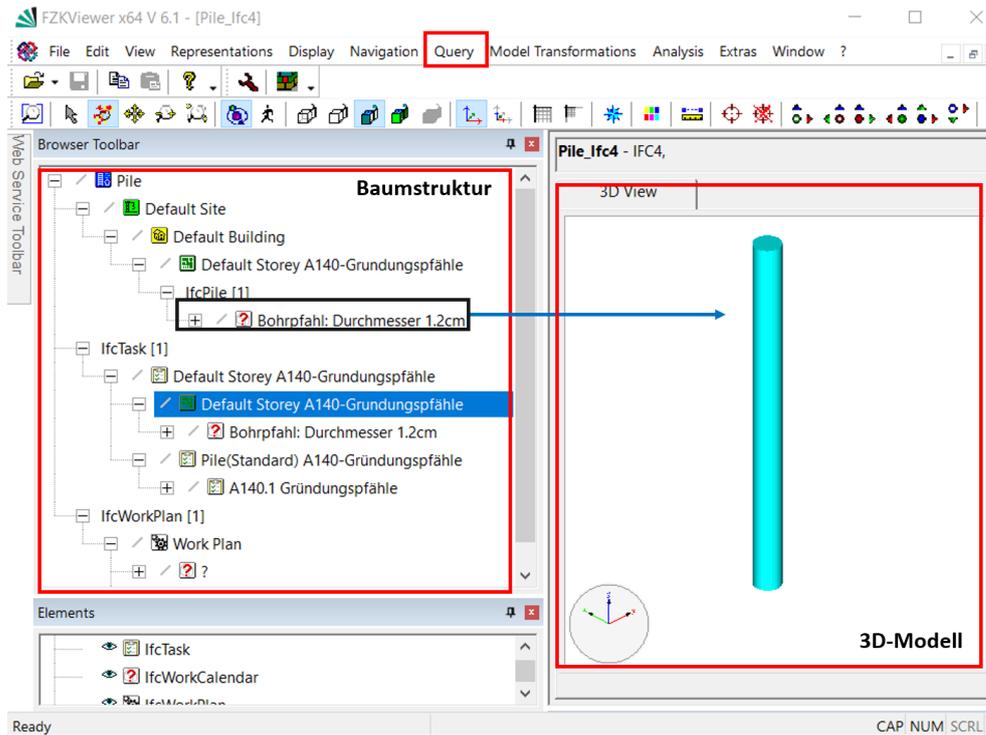


Abbildung 3-6: 4D-Modell in FZK Viewer in der vorliegenden Arbeit

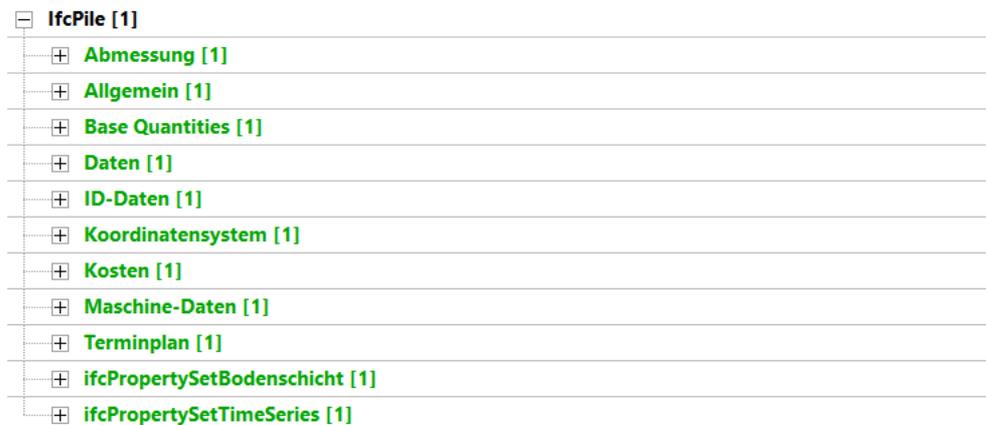


Abbildung 3-7: Funktion Abfragen in FZK Viewer

## BIMcollab ZOOM

BIMcollab ZOOM ist ein IFC-Viewer, um das IFC-Modell zu validieren. Allerdings muss die Weiterverarbeitung des IFC-Modells bezahlt werden. Die zeitlichen Daten werden nicht in BIMcollab Zoom dargestellt. Daneben werden die IFC-Eigenschaften automatisch auf ein proprietäres, internes Datenmodell abgebildet, siehe Abbildung 3-8.

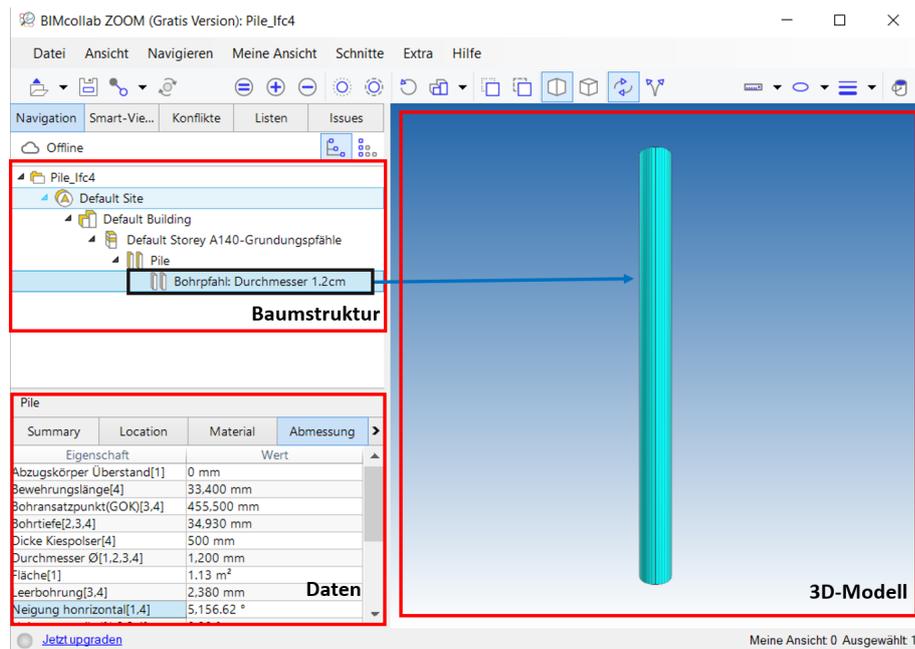


Abbildung 3-8: 4D-Modell in BIMcollab ZOOM in der vorliegenden Arbeit

## xBIM Xplorer

xBIM Xplorer gehört zum xBim-Toolkit. Die IFC-Datei und die ifcXML-Datei werden mit xBIM Xplorer dargestellt. Das bringt den Vorteil mit sich, dass neben den IFC-Eigenschaften die Objektbeziehungen mit xBIM Xplorer vollständig dargestellt werden. Allerdings können die zeitlichen Daten nicht mit xBIM Xplorer gezeigt werden, siehe Abbildung 3-9.

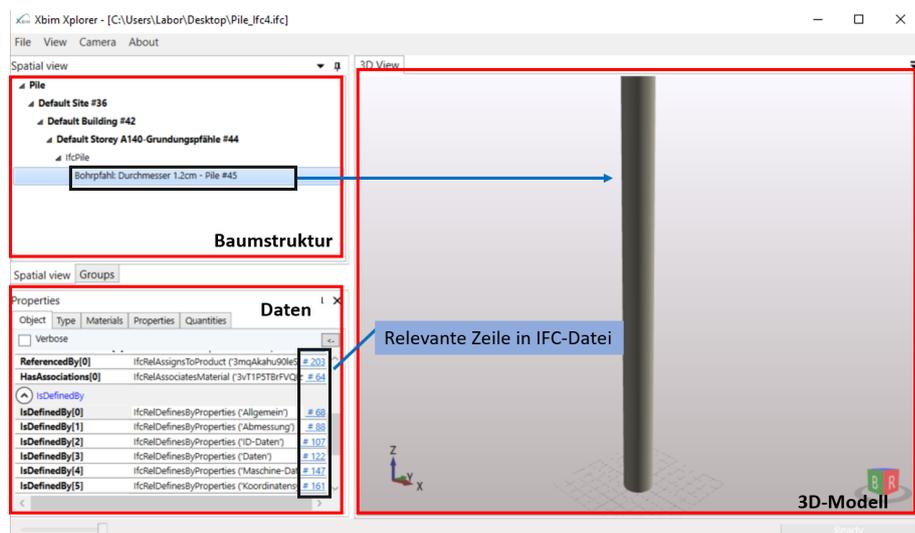


Abbildung 3-9: 4D-Modell in xBIM Xplorer in der vorliegenden Arbeit

## **Simplebim**

Simplebim wird als eine leistungsstarke Software angesehen, mit der die Konstruktionsinformationen und -modelle eingelesen, geändert und verwaltet werden können. Allerdings sind die Funktionen kostenpflichtig.

## **Zusammenfassung**

In diesem Kapitel wurde ein Überblick über die Vielzahl an Softwarelösungen für die Validierung des BIM-Modells gegeben. Im Unterschied zu anderen Softwarelösungen wird die Visualisierung des IFC-Modells mit der Software Constructivity übersichtlich und vollständig dargestellt. Darüber hinaus sind die IFC-Eigenschaften mithilfe der Software Constructivity gut lesbar. Deshalb wird die Software Constructivity als Model Checker in der vorliegenden Arbeit verwendet.

Darüber hinaus werden der Aufbau der IFC-Datei und der ifcXML-Datei sowie der XML-Datei mit Notepad++ in der vorliegenden Arbeit gesehen und geschrieben. Außerdem wird die Software IFC Quick Browser verwendet, um die Objektbeziehungen des IFC-Modells zu lesen. Zusätzlich wird die Funktion Abfragen von FZK Viewer in der vorliegenden Arbeit angewendet, um die Klassifizierung der Eigenschaften eindeutig darzustellen.

Nach erfolgreicher Validierung wird das IFC-Modell akzeptiert, sonst muss das IFC-Modell wieder modifiziert und validiert werden.

## 4 BIM-Modell

---

In diesem Kapitel wird die Modellierung des IFC-Formats, das in der vorliegenden Arbeit im Programm MVS erstellt wird, ausführlich erläutert. Darüber hinaus werden die Eigenschaften des IFC-Modells zuerst eindeutig klassifiziert. Dann kann das IFC-Modell anhand des UML-Diagramms aufgebaut werden. Die erfolgreiche Validierung des IFC-Modells wird mithilfe der Software Constructivity durchgeführt, siehe Kapitel 3.3. Zuletzt werden die Eigenschaften als Eingangsparameter aus dem IFC-Modell in die Software Plant Simulation von Siemens PLM exportiert und dort abgespeichert.

### 4.1 Datenbeschaffung

Mithilfe des Telematikmoduls TC3G von Sensor-Technik Wiedemann GmbH (STW) werden die Daten in den Drehbohrgeräten von Bauer mit ihrem Server verbunden [STW-2020]. Dabei wird die Datenübertragung entweder nach einem bestimmten Ereignis, z. B. nach Beendigung eines Bohrpfahls, nach Überschreitung eines bestimmten Datenvolumens oder in einem bestimmten Zeitintervall von bis zu 15 min, durchgeführt. Zusätzlich wird die proprietäre Webapplikation „WEB-BGM“ von Bauer für die Übertragung der Daten im .json-Format eingesetzt. Beim WEB-BGM handelt es sich um drei verschiedene Daten: „B-Report“-Daten, „B-Activity“-Daten sowie „ISO“-Daten. Außerdem werden die „MiC 4.0“-Daten und das Positionspapier Spezialtiefbau vom Hauptverband der Deutschen Bauindustrie sowie die IFC-Dateien aus dem Pilotprojekt Autobahndirektion Südbayern in der vorliegenden Arbeit verwendet. Diese genannten Daten sind für die Bestimmung der IFC-Eigenschaften von großer Bedeutung. Im Folgenden werden die verschiedenen Daten und das Positionspapier sowie das Pilotprojekt Autobahndirektion Südbayern erläutert; sie werden als unterschiedliche Quellen für die Beschreibung eines BIM-Modells im Spezialtiefbau betrachtet, um das BIM-Modell zu erstellen. Dafür verfügen sie über eigene Quelle-Nummern, um sie zu identifizieren, sonst werden die Daten selbst als Quelle [5] nummeriert [Fis-2020, HDB-2019, Bun-2017]:

- **„B-Report“-Daten [1]:** Alle erfassten Maschinendaten gehören zu diesen Daten, wie beispielsweise Bohrtiefe, Status Gerät, Seilkraft Hauptwinde, Neigung X-Y Mast und

Vorschubkraft. Dabei lassen sich die Daten im WEB-BGM unter „B-Report“ lesen und exportieren.

- **„B-Activity“-Daten [1]:** Mit dem B-Activity-Modul wird es ermöglicht, dass einzelne zuvor definierte Ereignisse während der Ausführung quittiert werden, wie zum Beispiel Bohren, Ziehen, Leeren oder Ablassen des Werkzeuges. Dabei sind diese Ereignisse sowohl mit dem Zeitstempel als auch der Identifikationsnummer des Drehbohrgeräts versehen. Es ist möglich, dass die Daten im WEB-BGM unter „B-Activity“ gelesen und exportiert werden.
- **„ISO“-Daten [1]:** Hier handelt es sich um die in der ISO 15143-3 definierten Werte sowie deren Austausch mit Dritten. Es gibt die Einrichtung einer API im WEB-BGM mit dem „Data Exchanger“ [Fis-2020].
- **Positionspapier [2]:** Das Positionspapier dient dazu, dass die Anforderungen im BIM-Prozess an andere Baubeteiligte und die Schnittstellen zu anderen Baubeteiligten sowie die zu liefernden Daten definiert werden. Außerdem werden die allgemeinen Definitionen von BIM im Bauprozess präsentiert und ergänzt. Im Hinblick auf das Positionspapier liegt der Fokus in der vorliegenden Arbeit überwiegend auf den Mindestanforderungen an Modellelemente. Abbildung 4-1 zeigt ein Beispiel eines Pfahls und den Bauteilcode 140 und verschiedene Eigenschaften, wie beispielsweise Typ, Nummer, Länge und Durchmesser sowie die Einheiten.
- **„MiC-4.0“-Daten [3]:** Diese Daten beziehen sich auf das MiC 4.0 Cluster Spezialtiefbau auf Basis von ISO 15143-3.
- **Pilotprojekt Autobahndirektion Südbayern [4]:** Das Pilotprojekt mit dem Projektnamen „A 99 BW 27/1 Unterführung der DB + S 8 Flughafen Ersatzneubau“ wurde von der Autobahndirektion Südbayern durchgeführt. Der Projekttyp wird als Brücke klassifiziert. In der vorliegenden Arbeit wird die IFC-Datei dieses Pilotprojekts verwendet. Die Abbildung 4-2 stellt ein IFC-Modell der Brücke aus diesem Pilotprojekt in der Software Constructivity dar. Zudem wird dieses IFC-Modell nach der Konvertierung in das XML-Format auch in die Software Plant Simulation importiert.

140	Pfahl	
	Typ	-
	Nummer	-
	Material (für alle Bestandteile separat anzugeben)	-
	Materialgüte (für alle Materialien separat anzugeben)	-
	Durchmesser	mm/cm
	Pfahlkopflage	-
	Länge	m
	Neigung vertikal	Grad
	Neigungsrichtung	-
	Bewehrungsgehalt	kg/m <sup>3</sup>

Abbildung 4-1: Mindestanforderungen an Pfahl mit dem Bauteilcode 140, den Eigenschaften und den Einheiten im Positionspapier [HDB-2019]

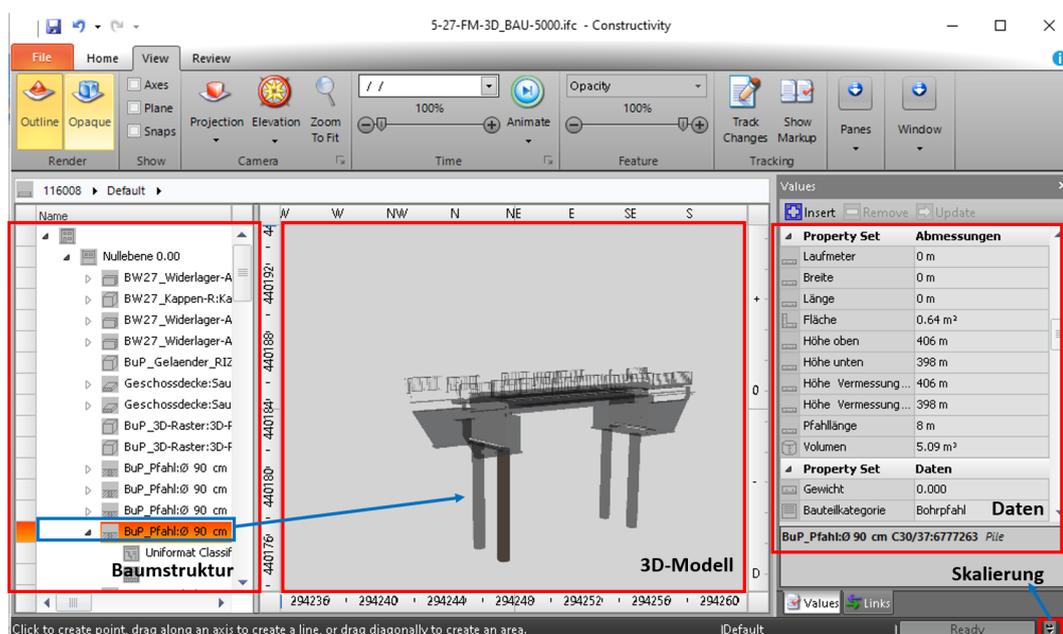


Abbildung 4-2: IFC-Modell der Brücke aus Pilotprojekt Autobahndirektion Südbayern

Im Rahmen des Brückenbauprojekts „WTRO“ werden die IFC-Eigenschaften für die IFC-Datei anhand der Quellen, wie bereits beschrieben, bestimmt und anschließend eindeutig im IFC-Modell klassifiziert. Der Fokus liegt auf den IFC-Eigenschaften, die mithilfe der Subklassen `IfcPropertySingleValue` definiert und im Instanzmodell Bohrpfahl A140.1 eingeführt werden. Insgesamt stehen acht Kategorien für die IFC-Eigenschaften zur Verfügung:

- **Allgemein:** allgemeine textuelle Daten, wie beispielsweise Bauvorhaben, Projektname und Architekt

- **Abmessung:** ermittelte Daten durch Messung, wie beispielsweise Pfahllänge, Durchmesser und Bohrtiefe
- **ID-Daten:** die Daten, um das Instanzmodell zu klassifizieren und zu identifizieren, wie beispielsweise Nummer, Typname und Methode
- **Daten:** ermittelte Daten ohne Messung, wie beispielsweise Klassifikation, Gewerk und Betriebskilometer
- **Maschinendaten:** die Daten über die Maschinen, wie beispielsweise Bohrgeschwindigkeit, Typ des Bohrgeräts und Anzahl der Betonmischer
- **Terminplan:** die zeitlichen Daten, wie beispielsweise frühester Anfang und spätestes Ende
- **Koordinatensystem:** zum Beispiel X-Hoch und Y-Rechts
- **Kosten:** Baukosten, wie beispielsweise Materialkosten, Personalkosten oder Maschinenkosten

Im Rahmen des Brückenbauprojekts „WTRO“ werden die Soll-Daten in die IFC-Datei eingeführt. Die geeigneten Eigenschaften aus der IFC-Datei werden in die Software Plant Simulation importiert.

Abbildung 4-3 zeigt ein Beispiel für die IFC-Eigenschaften in der Kategorie Abmessung in der Software Constructivity und dabei wird der Vergleich der IFC-Eigenschaften mit unterschiedlichen Quellen in der Kategorie Abmessung dargestellt, siehe Tabelle 4-1. Zudem werden die in die Software Plant Simulation exportierten IFC-Eigenschaften in rotem Rechteck dargestellt. Außerdem handelt es sich im IFC-Modell um die verschiedenen Wertetypen, wie beispielsweise `IfcLengthMeasure` (Real) und `IfcLabel` (String). Daneben wird die Darstellung der IFC-Eigenschaften in den anderen Kategorien im Anhang A1 beigefügt.

Property Set	Abmessung	
Pfahllänge[1,2,3,4]	32.050 m	IfcLengthMeasure(Real)
Durchmesser Ø[1,2,3,4]	1.2 m	
Volumen[1,3,4]	36.25 m <sup>3</sup>	IfcVolumeMeasure (Real)
Fläche[1]	1.13 m <sup>2</sup>	IfcAreaMeasure(Real)
Bohrtiefe[2,3,4]	34.93 m	
Neigung horizontal[1,4]	90 °	IfcPlaneAngleMeasure (Real)
Neigung vertikal[1,2,3,4]	0 °	
Pfahluferkante[1,3,4]	421.07 müNN	
Pfahloberkante[1,3,4]	453.12 müNN	
Bohransatzpunkt(GOK)[3,4]	455.5 müNN	
Pfahloberkante/Pfahluferkante/GOK Ei...	müNN	IfcLabel(String)
Abzugskörper Überstand[1]	0 m	
Pfahllänge Überstand[1]	0 m	
Pfahlachsüberstand[1]	0 m	Typ
Leerbohrung[3,4]	2.38 m	
Dicke Kiespolser[4]	0.5 m	Legende
Bewehrungslänge[4]	33.4 m	In Plant Simulation importieren

Abbildung 4-3: IFC-Eigenschaften in der Kategorie Abmessung. Rot umrandet sind die importierten Eigenschaften

Tabelle 4-1: IFC-Eigenschaften mit unterschiedlichen Quellen in der Kategorie Abmessung

IFC-Eigenschaften	Wert	[1]	[2]	[3]	[4]
Pfahllänge	32,05 m	x	x	x	x
Durchmesser Ø	1,2 m	x	x	x	x
Volumen	36,25 m <sup>3</sup>	x		x	x
Fläche	1,13 m <sup>2</sup>	x			
Bohrtiefe	34,93 m		x	x	x
Neigung horizontal	90°	x			x
Neigung vertikal	0°	x	x	x	x
Pfahluferkante	421,07 müNN	x		x	x
Pfahloberkante	453,12 müNN	x		x	x
Bohransatzpunkt (GOK)	455,50 müNN			x	x
Abzugskörper Überstand	0 m	x			
Pfahllänge Überstand	0 m	x			
Leerbohrung	2,38 m			x	x
Dicke Kiespolser	0,5 m				x
Bewehrungslänge	33,40 m				x

## 4.2 Datenaufbereitung

### 4.2.1 IFC-Modell

Anhand des UML-Diagramms wird die IFC-Datei des Gebäudes für die Bohrfahlerstellung erstellt. Die Beispieldatei wird im Programm MVS mithilfe des xBIM-Toolkits erstellt.

Die .ifc-Datei wird in zwei Abschnitte eingeteilt, die Kopfzeile (Header) und den Datenabschnitt (Data), siehe Abbildung 4-4. Im DATA-Abschnitt steht am Anfang jeder Zeile in der IFC-Datei ein Rauten-Symbol „#“. Die Entitäten-Nummer „#-Zeichen“ entspricht der Zeilennummer in der IFC-Datei. Dabei handelt es sich um lediglich eine Entität mit seinen Attributen (Klammerwerte) oder Verweise auf Zeilennummern, mit denen diese Entität mit anderen Entitäten verbunden wird. Beim Erstellen der Beispieldatei stehen mit dem Symbol „\$“ gekennzeichnete Attribute zur Verfügung. Das bedeutet, dass diese Attribute nicht befüllt werden [Gün-2018].

Des Weiteren werden die Informationen, wie Besitzerhistorie (#2) und Einheitenbeschreibung (#7, #8, #9), wie beispielsweise Meter, GradSekunde, Kubikmeter und Kilogramm, in der Abbildung 4-4 ermittelt [Bor-2015]. In der Besitzerhistorie werden alle geschichts- und identifikationsbezogenen Informationen definiert. Es wird dadurch ermöglicht, dass die Besitzerhistorie unmittelbar mit allen unabhängigen Objekten, Beziehungen und Eigenschaften verbunden wird [Bui-2021].

```

ISO-10303-21;
HEADER;
FILE_DESCRIPTION ((' ', '2;1');
FILE_NAME (' ', '2021-05-04T20:38:38', (' ', (' ', 'Processor version 5.1.0.0', 'Xbim.IO.MemoryModel', ''));
FILE_SCHEMA (('IFC4'));
ENDSEC;
DATA;
#1=IFCPROJECT('OdjYpcf2r7UvnNPQSiMBNg', #2, 'Pile', $, $, 'B15, Westtangente Rosenheim', 'Projektstatus', (#19, #22), #7);
#2=IFCOWNERHISTORY(#5, #6, $, .ADDED., 1620160719, $, $, 0);
#3=IFCPERSON($, 'Team', 'xBIM', $, $, $, $);
#4=IFCORGANIZATION($, 'fml', $, $, $);
#5=IFCPERSONANDORGANIZATION(#3, #4, $);
#6=IFCAPPLICATION(#4, '1.0', 'Pile Application', 'Pile.exe');
#7=IFCUNITASSIGNMENT((#8, #9, #10, #11, #12, #13, #14, #15, #16, #23, #24, #31, #35));
#8=IFCSIUNIT(*, .LENGTHUNIT., $, .METRE.);
#9=IFCSIUNIT(*, .AREAUNIT., $, .SQUARE_METRE.);

```

Abbildung 4-4: IFC-Beispielcode aus der vorliegenden Arbeit

Mithilfe des Tools IfcConvert von IfcOpenShell-Projekt wird das Standardformat der IFC, das IFC-Format, in das XML-Format konvertiert. Es enthält die gleichen und vollständigen Informationen wie das IFC-Format. In Abbildung 4-5 wird eine durch Konvertieren aus der IFC-Datei erzeugte XML-Datei dargestellt. Sie beinhaltet die gleichen Informationen wie

das IFC-Format. Ein Beispiel hierfür sind die Daten aus dem HEADER-Abschnitt und die Einheitenbeschreibung. Allerdings sind die Informationen im XML-Format unvollständig im Gegensatz zum ifcXML-Format.

```

▼<ifc xmlns:xlink="http://www.w3.org/1999/xlink">
  ▼<header>
    ▼<file_description>
      <description/>
      <implementation_level>2;1</implementation_level>
    </file_description>
    ▼<file_name>
      <author/>
      <organization/>
      <name/>
      <time_stamp>2021-05-04T20:38:38</time_stamp>
      <preprocessor_version>Processor version 5.1.0.0</preprocessor_version>
      <originating_system>Xbim.IO.MemoryModel</originating_system>
      <authorization/>
    </file_name>
    ▼<file_schema>
      <schema_identifiers>IFC4</schema_identifiers>
    </file_schema>
  </header>
  ▼<units>
    <IfcSIUnit UnitType="LENGTHUNIT" Name="METRE" SI_equivalent="1"/>
    <IfcSIUnit UnitType="AREAUNIT" Name="SQUARE_METRE" SI_equivalent="1"/>
  </units>

```

**HEADER**

**Grundmaßeinheit**

Abbildung 4-5: XML-Beispielcode aus der vorliegenden Arbeit

Durch die hierarchische Verbindung in Abbildung 4-6 werden die Entitäten dieser Gebäudestruktur gezeigt. Die Beziehungen zwischen diesen Entitäten, wie IfcProject, IfcSite, IfcBuilding, IfcBuildingStorey und IfcPile, entsprechen einer 1:1-Verbindung. Im Prinzip ist es möglich, dass es um 1:n-Verbindungen geht [Gün-2018]. Das bedeutet, dass eine übergeordnete Entität für jede Entität zur Verfügung steht. Dadurch wird es ermöglicht, dass viele Entitäten IfcPile unter einer Entität IfcBuildingStorey im IFC-Modell vorhanden sind.

Durch die Beziehungsobjekte IfcRelAggregates und IfcRelContainedInSpatialStructure wird der Gesamtzusammenhang der verschiedenen Entitäten und die hierarchische IFC-Struktur dargestellt, siehe Abbildung 4-6.

Die Beziehungsobjekte, wie IfcRelAggregates und IfcRelContainedInSpatialStructure, haben folgenden Aufbau [Gün-2018]:

**#[Zeilennummer]=** Name des Beziehungsobjekts([1], [2], [3], [4], [5], [6])

[1] = Globale Identifizierungsnummer

[2] = Verweis auf Besitzerhistorie

[3] = Name

[4] = Beschreibung

[5] = betreffendes Objekt

[6] = zugeordnete / untergeordnete Objekte

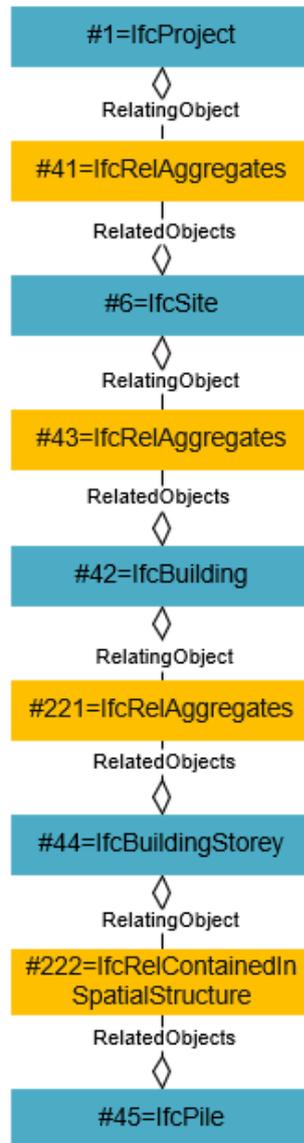


Abbildung 4-6: die Gebäudestruktur aus der zugehörigen IFC-Datei am Beispiel vom Bohrpfahl A140.1 im Rahmen des Brückenbauprojekts „WTRO“ in Anlehnung an [Bor-2015]

Die hierarchische Verbindung wird durch die Beziehungszusammenhänge in [5] und [6] hergestellt. Objekt [5] wird als ein Container für das Objekt [6] betrachtet, siehe Abbildung 4-7. Die Zeile #41 verweist auf Zeile #1 mit dem IfcProject und auf die Zeile #36 mit dem IfcSite. Dadurch wird der Zusammenhang zwischen den beiden Entitäten erzeugt. Die nächste Zeile #43 verbindet die IfcSite-Zeile #36 und die IfcBuilding-Zeile #42 und die Zeile #221 führt die IfcBuilding-Zeile #42 und die IfcBuildingStorey-Zeile (#44) zusammen. In der Zeile #222

wird die IfcPile-Entität (#45) dem Objekt im Container IfcBuildingStorey (#44) untergeordnet [Gün-2018].

```
#41=IFCRELAGGREGATES('2wksqVltXFaq3j2BNlhlc',#2,$,#1,(#36));
#43=IFCRELAGGREGATES('2gGTgx5u92ChO5ldn8N0lg',#2,$,#36,(#42));
#221=IFCRELAGGREGATES('2jG6$Z_FT34hL$R18SqllB',#2,$,#42,(#44));
#222=IFCRELCONTAINEDINSPATIALSTRUCTURE('04BAR65ffFmfcRkcsI75aD',#2,$,(#45),#44);
```

Abbildung 4-7: Hierarchische Beziehung in der IFC-Datei

Auf der obersten Hierarchieebene wird das Projekt in IfcProject dargestellt. Unter diesem Objektnamen Pile wird die Gebäudestruktur aufgebaut.

In der Abbildung 4-8 wird gezeigt, dass die Zeile #1 in ihrer Beschreibung auf weitere Zeilen verweist, wie #2 (Besitzerhistorie), #7 (Einheitenbeschreibung) und #19 und #22 (geometrische Daten), die sowohl die grundsätzlichen Daten als auch Informationen zum gesamten Projekt beinhalten [Gün-2018]. Dazu handelt sich es in der Beschreibung um den Projekt-namen B15, Westtangente Rosenheim und Projektstatus.

```
#1=IFCPROJECT(2j3LVmpr9BsP7JomqkBGf$,#2,'Pile',$,$,'B15, Westtangente Rosenheim','Projektstatus',(#19,#22),#7);
#2=IFCOWNERHISTORY(#5,#6,$,.ADDED.,1621250966,$,$,0);
#7=IFCUNITASSIGNMENT((#8,#9,#10,#11,#12,#13,#14,#15,#16,#23,#24,#31,#35));
#19=IFCGEOMETRICREPRESENTATIONCONTEXT('Building Model','Model',3,1.E-05,#18,$);
#22=IFCGEOMETRICREPRESENTATIONCONTEXT('Building Plan View','Plan',2,1.E-05,#21,$);
```

Abbildung 4-8: IfcProject und seine Verweise

Die Entität von IfcSite beinhaltet die Informationen über die geografische Lage und die genauen Bruchlinien der Liegenschaft sowie die Siteadresse. Zudem wird das Attribut CompositionType auf ELEMENT gesetzt, siehe Abbildung 4-9. Durch IfcLocalPlacement wird auf die globale Position verwiesen. Des Weiteren wird das Gebäude, das hierarchisch unter der Liegenschaft liegt, durch IfcBuilding repräsentiert. Außerdem wird ein Geschoss durch IfcBuildingStorey dargestellt [Gün-2015].

```
#36=IFCSITE(3CmA2_koP4YeF18DWZ3RCI',#2,'Default Site',$,$,#38,$,$,ELEMENT,(0,0,0,0),(0,0,0,0),0,$,#37);
#2=IFCOWNERHISTORY(#5,#6,$,.ADDED.,1621250966,$,$,0);
#38=IFCLOCALPLACEMENT($,#39);
#37=IFCPOSTALADDRESS($,$,$,'<AddressForInterMailDelivery>',('<Projektaddress>'),$,'Rosenheim','Bayern','830
22','Deutschland');
#42=IFCBUILDING('0C0pqpUyV9vuMFRBrLnMc',#2,'Default Building',$,$,$,$,ELEMENT,$,$,$);
#2=IFCOWNERHISTORY(#5,#6,$,.ADDED.,1621250966,$,$,0);
#44=IFCBUILDINGSTOREY('2JNLcS6XX1mwmXE6fDsqGa',#2,'Default Storey A140-GrundungspfX\E4hle',$,$,$,$,EL
EMENT,0.);
#2=IFCOWNERHISTORY(#5,#6,$,.ADDED.,1621317934,$,$,0);
```

Abbildung 4-9: IfcSite, IfcBuilding und IfcBuildingstore

Das Element `IfcPile` wird als ein Pfahl in einem Gebäude beschrieben und es geht um die Beziehung mit der `IfcBuildingStorey`. Zusätzlich hat das Element `IfcPile` an dritter Stelle in der Klammer einen Namen, in diesem Fall „Bohrpfahl: Durchmesser 1.2cm“. Zudem werden das Attribut `IfcPileTypeEnum` auf `BORED` und `ConstructionType` auf `CAST_IN_PLACE` gesetzt, siehe Abbildung 4-10. Weiterhin wird bei der Entität `IfcPile` sowohl auf die Zeile #54 zur Produktionsdefinition (`IfcProduktdefinitionsshape`) verwiesen als auch auf die Zeile #2 Besitzerhistorie. Außerdem hat `IfcLocalPlacement` (#55) an der ersten Stelle in der Klammer das Symbol „\$“ und damit wird relativ zur Lage des Baufelds `IfcSite` positioniert. Darüber hinaus wird die geometrische Repräsentation für den Pfahl definiert (#54). Ein dreidimensionaler Volumenkörper wird als ‚SweptSolid‘ beschrieben (#53, #49) und wird durch die Extrusion der Grundfläche (#46) erzeugt, die durch ein Kreisprofil mit dem Durchmesser 1,2 m definiert wird. Dadurch erfolgt die Extrusion entlang des Vektors (0.0, 0.0, 1.0) (#96) mit einer Höhe von 32,05 m (#49).

```
#45=IFCPILE('1JQEdlt6L49x7usZeO8Cem',#2,'Bohrpfahl: Durchmesser 1.2cm',$,#55,#54,$,.BORED,.CAST_IN_PLAC
E.);
#2=IFCOWNERHISTORY(#5,#6,$,.ADDED.,1621317934,$,$,0);
#54=IFCPRODUCTDEFINITIONSHAPE($,$,#53);
#53=IFCSHAPEREPRESENTATION(#19,'Body','SweptSolid',(#49));
#49=IFCEXTRUDEDAREASOLID(#46,#52,#50,32.05);
#46=IFCCIRCLEPROFILEDEF(.AREA.,'CircleProf',#48,1.2);
#50=IFCDIRECTION((0.,0.,1.));
#52=IFCAXIS2PLACEMENT3D(#51,$,$);
#55=IFCLOCALPLACEMENT($,#56);
```

Abbildung 4-10: `IfcPile` in IFC-Datei

Weiterhin kommt es zum Definieren der neuen Objekteigenschaft mit einfacher Name-Wert-Datentyp-Einheit-Tupel. Die Basisentitäten der Eigenschaften, also z.B. `IfcPropertySingleValue` mit den Attributen ‚Name‘, ‚NominalValue‘, ‚Type‘ und ‚Unit‘, müssen implementiert werden, wie „Name: ‚Bohrtiefe [2, 3, 4]‘; NominalValue: ‚34.93‘; Type: ‚IfcLengthMeasure‘; Unit: ‚m‘“, siehe Abbildung 4-11. Einzelne `IfcProperty`s werden zu einer Gruppe (`IfcPropertySet`) zusammengefasst und anschließend einem Objekt (`IfcPile`) zugeordnet (`IfcRelAssignsProperties`).

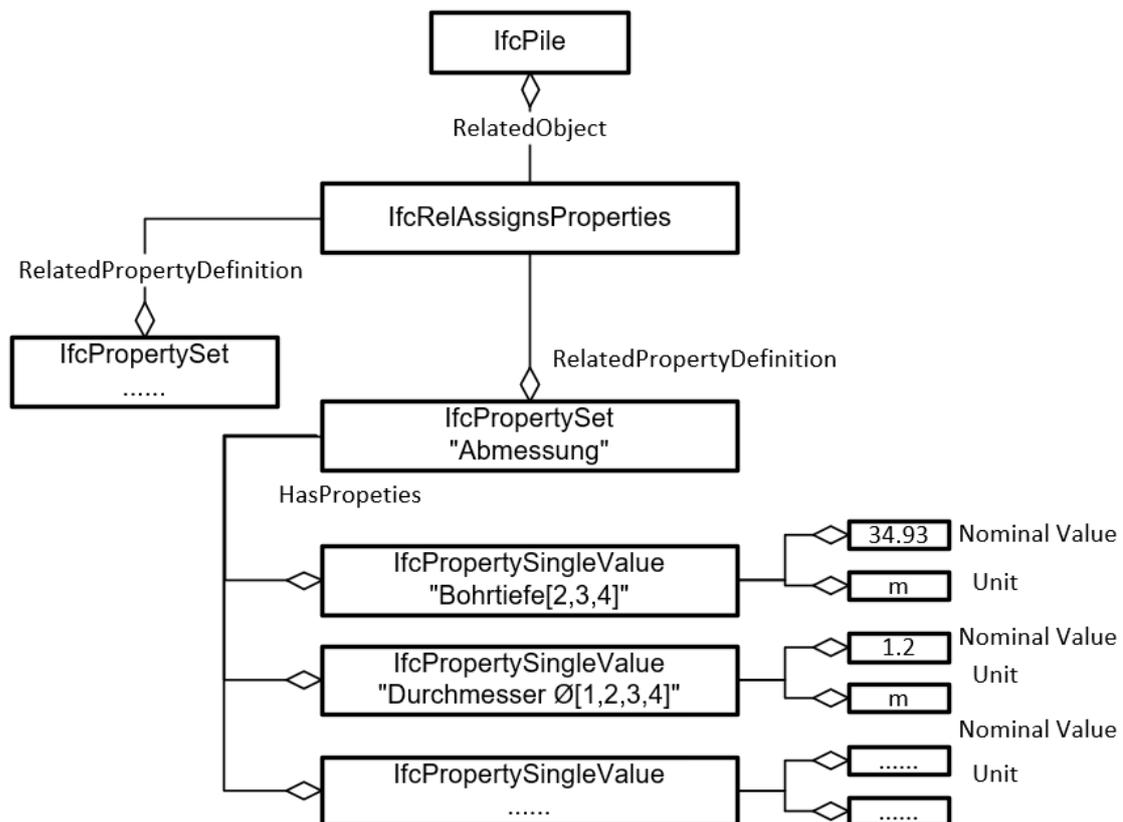


Abbildung 4-11: Beispielhafte Verwendung von Properties (Eigenschaften) aus der vorliegenden Arbeit in Anlehnung an [Bor-2015]

Darüber hinaus werden alphanumerische Eigenschaften festgelegt. Dafür erfolgt deren Definition über `IfcPropertySet`. Dies zeigt ein Beispiel: In Abbildung 4-12 ist ein Satz an Eigenschaften (`IfcPropertySet`, #69) dargestellt, der dem Pfahlobjekt (#45) durch Beziehungsobjekte (`IfcRelDefinesByProperties`, #68) zugeordnet ist. In den Zeilen #70 bis #87 werden Eigenschaftswerte definiert, wie zum Beispiel der Projektleiter [1] (#71), der eine eigene Quellen-Nummer besitzt [1].

```
#68=IFCRELDEFINESBYPROPERTIES('2jPf_o20z8cPyJEL3wzhQU',#2,$,($,#45),#69);
    #45=IFCPPILE('1JQEdlt6L49x7usZeO8Cem',#2,'Bohrpfahl: Durchmesser 1.2cm',$,$,#55,#54,$,.BORED.,CAST_
    IN_PLACE.);
#69=IFCPROPERTYSET('1dedGLJKH42fbDFU6$Dkdp',#2,'Allgemein',$,($,#70,#71,#72,#73,#74,#75,#76,#77,#78,#79,#80,
#81,#82,#83,#84,#85,#86,#87));
    #2=IFCOWNERHISTORY(#5,#6,$,.ADDED.,1621317934,$,$,0);
    #70=IFCPROPERTYSINGLEVALUE('Bauvorhaben(BV)[4]',$,IFCTEXT('Westtangete Rosenheim(WTRO)'),$);
    #71=IFCPROPERTYSINGLEVALUE('Projektleiter[1]',$,IFCTEXT(''),$);
    .....
    #86=IFCPROPERTYSINGLEVALUE('Modellierer[1]',$,IFCTEXT(''),$);
    #87=IFCPROPERTYSINGLEVALUE('zentrale Projektdatei[1]',$,IFCTEXT(''),$);
```

Abbildung 4-12: Eigenschaften (`IfcProperty`), die in `IfcPropertySets` Allgemein gruppiert sind

In der Abbildung 4-13 ist dargestellt, dass ein Satz an Maß- und Mengenangaben (`IfcElementQuantity`, #212) definiert und dem Pfahlobjekt (#45) durch Beziehungsobjekte (`IfcRel-`

DefinesByProperties, #213) zugeordnet wird. Zudem werden die Eigenschaftswerte, wie beispielsweise die Pfahlänge (#210), in den Zeilen #208 bis #210 spezifiziert.

```
#213=IFCRELDEFINESBYPROPERTIES('0LYh$gyNHEmBb3kxzOX7LI',#2,'Area Association','IfcElementQuantity
associated to wall',(#45),#212);
#212=IFCELEMENTQUANTITY('3$dviXNEH77vx7kzRz0LfY',#2,'Base Quantities','Measurement
quantity',$(#208,#209,#210));
#208=IFCQUANTITYAREA('FIX\E4che',$$,1.131,$);
#209=IFCQUANTITYVOLUME('Volumen',$$,36.2477,$);
#210=IFCQUANTITYLENGTH('Pfahl\X\E4nge',$$,32.05,$);
```

Abbildung 4-13: Zuordnung von Maß- und Mengenwerten über ein IfcElementQuantity-Objekt

#### 4.2.2 Terminplan

Anhand des UML-Diagramms wird zusätzlich der Terminplan während der Bohrpfahlherstellung innerhalb der IFC-Datei erstellt, siehe Abbildung 4-14. Im Anhang B wird der Terminplan in der IFC-Datei ausführlich dargestellt.

Der Aufbau der Beziehungsobjekte wurde in diesem Kapitel bereits erläutert. Das Bauteilobjekt Pfahl (#45), wie bereits beschrieben, wird über das Beziehungsobjekt des Typs IfcRelAssignsToProcess (#199) mit einem IfcTask (#188) verknüpft, der als Terminplanvorgang betrachtet wird. Unter diesem Terminplanvorgang werden mehrere Vorgänge zur Verfügung gestellt. Ein IfcRelAssignsToProduct (#203) verknüpft dabei ein IfcTask (#197) mit einem IfcPile-Objekt (#45). Eine Zuordnung von Teilobjekten zum Originalobjekt wird mittels eines Beziehungsobjekts vom Typ IfcRelNests erzeugt [Tul-2008]. Die zwei IfcTasks (#188 und #197) werden durch IfcRelNests mit einem IfcTask (#195) verbunden. Daher wird das IfcTask (#197) ein Terminplanvorgang, der dem Vorgang im Container IfcTask (#188) untergeordnet wird. Darüber hinaus werden die IfcWorkPlan (#186) durch IfcRelNests (#204) mit IfcWorkCalendar (#191) und IfcWorkSchedule (#187) verknüpft. Zum einen wird IfcWorkCalendar (#191) durch IfcRelAssignsToControl(#199) mit IfcTask (#188) verbunden. Zum anderen wird durch IfcRelAssignsToControl der Zusammenhang zwischen IfcWorkCalendar (#191) und IfcPile (#45) und daneben der zwischen IfcWorkCalendar (#191) und IfcTask (#188) hergestellt.

Mittels eines Beziehungsobjekts vom Typ IfcRelDeclares wird IfcProject mit IfcTask (#188) und IfcWorkPlan (#186) verknüpft. Das IfcTask (#188) wird durch IfcRelAssignsToProduct (#202) mit IfcBuildingStorey (#44) verbunden.

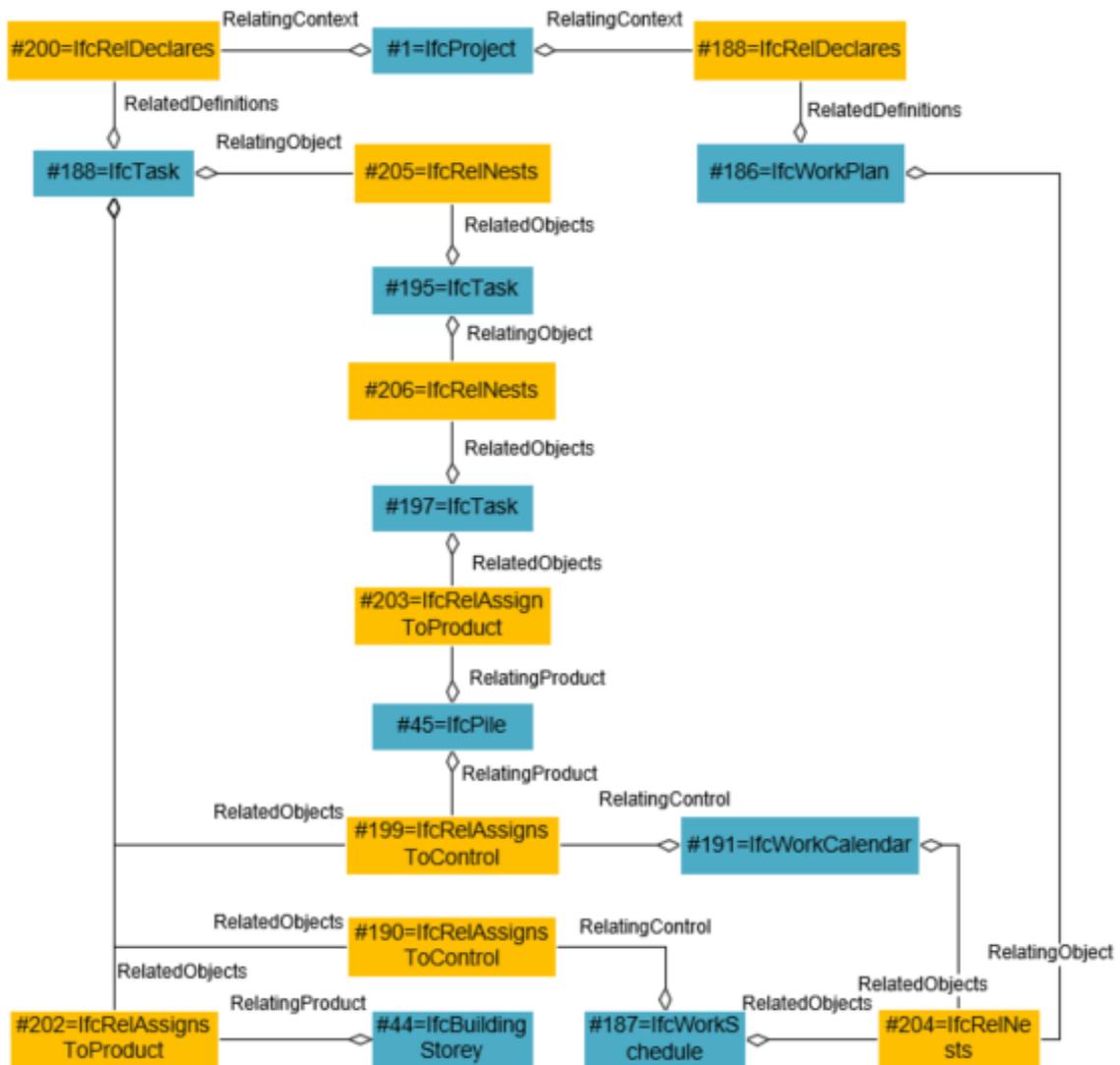


Abbildung 4-14: Terminplan innerhalb der IFC-Datei

Darüber hinaus wird das `IfcWorkCalendar`-Objekt (#191) dem neu eingeführten `IfcWorkTime`-Objekt (#192) zugeordnet, in das das neue `IfcRecurrencePattern`-Objekt (#193) eingeführt wird, siehe Abbildung 4-15. Das `IfcRecurrencePattern`-Objekt weist die Eigenschaften auf, wie `RecurrenceType` und `WeekdayComponent`. Zudem werden die Attribute `WeekdayComponent` auf (1, 2, 3, 4, 5) und `RecurrenceType` auf `Weekly` gesetzt.

```
#191=IFCWORKCALENDAR('0AvbhUS9jDgeTa_ZLbx3q9',#2,$,$,$,(#192),$.NOTDEFINED.);
#192=IFCWORKTIME('Standard',$,$,#193,$);
#193=IFCRECURRENCEPATTERN(.WEEKLY,$,(1,2,3,4,5),$,$,$,(#194));
#194=IFCTIMEPERIOD('08:00:00','17:00:00');
```

Abbildung 4-15: `IfcWorkCalendar` in IFC-Datei

Die Nummer im `WeekdayComponent` wird im Folgenden ermittelt [Bui-2021]:

- **1:** Montag
- **2:** Dienstag
- **3:** Mittwoch
- **4:** Donnerstag
- **5:** Freitag
- **6:** Samstag
- **7:** Sonntag

Außerdem verweist die Zeile #193 auf Zeile #194; damit wird eine neue Eigenschaft, wie Arbeitszeit, dem IfcRecurrencePattern-Objekt zugeordnet. Dafür ist die Arbeitszeit von 8 Uhr bis 17 Uhr als ein Beispiel definiert.

Das IfcTask-Objekt (#188) wird dem neu eingeführten IfcTaskTime-Objekt (#189) zugeordnet (siehe Abbildung 4-16), dessen Attribute im Folgenden erläutert werden [Bui-2021]:

- **DataOrigin:** PREDICTED
- **DurationType:** WORKTIME
- **ScheduleDuration:** 6D18H
- **ScheduleStart:** 2019-08-13T10:24:00
- **ScheduleFinish:** 2019-08-22T08:24:00
- **EarlyStart:** 2019-07-15T00:00:00
- **EarlyFinish:** 2019-07-22T00:00:00
- **LateStart:** 2019-07-15T00:00:00
- **LateFinish:** 2019-07-22T00:00:00

```
#188=IFCTASK('3XbmyiArnBkAH1KFSr7JLW',#2,'Default Storey A140-GrundungspfXIE4hle',,$,$,$,$,$,.F.,$,#189,.CO
NSTRUCTION.);
#189=IFCTASKTIME($,.PREDICTED.,$,WORKTIME.,'6D18H','2019-08-13T10:24:00','2019-08-
22T08:24:00','2019-07-15T00:00:00','2019-07-22T00:00:00','2019-07-15T00:00:00','2019-07-
22T00:00:00',$,$,$,$,$,$,$);
```

Abbildung 4-16: IfcTask in IFC-Datei

Die Daten sind der Quelle [1] entnommen, die bereits im Unterkapitel 4.2 erläutert wird. Weiterhin ist die Schreibweise der Zeit in der IFC-Datei YYYY-MM-DDThh:mm:ss. Dabei repräsentiert YYYY das Jahr, MM den Monat und DD den Tag [Bui-2021]. Die Dauer des Schedules ist 6 Tage 18 Stunden (6D18H).



Tabelle 4-2: Vergleich der unterschiedlichen Quellen mit den in die Plant Simulation exportierenden IFC-Eigenschaften

IFC-Eigenschaften	Wert	[1]	[2]	[3]	[4]	[5]
Bauvorhaben(BV)	Westtangete Rosenheim (WTRO)				x	
Projektadresse	Rosenheim, Bayern, 83022 Deutschland	x				
Nummer	A140.1	x	x	x	x	
Pfahllänge	32,05 m	x	x	x	x	
Durchmesser	1,2 m	x	x	x	x	
Bohrtiefe	34,93 m		x	x	x	
Neigung horizontal	90°	x			x	
Überbetonvolumen	1,7 m <sup>3</sup>					x
Bewehrt	true	x	x	x	x	
Anzahl der Bewehrungskorb	3					x
Anzahl der Rohrschüsse	8					x
Verrohrungsanlage	true			x	x	
Bohrgeräte-Typ	BG46					x
Bohrgeräte-Anzahl	-					x
Betonmischer-Typ	-					x
Trommelvolumen	7 m <sup>3</sup>					x
Betonmischer-Anzahl	-					x
Radlader-Typ	-					x
Radlader-Anzahl	-					x
Schaufelvolumen	-					x
Y-Rechts	4506249.795				x	
X-Hoch	5301308.650				x	
Start	2019-08-13 10:24:00			x	x	
Ende	2019-08-22 08:24:00			x	x	

## 5 Simulationsstudie am Fallbeispiel „Westtangente Rosenheim (WTRO)“

---

Das Ziel der Simulationsstudie auf Datenbasis eines realen Anwendungsfalls ist, eine optimale Maschineneinsatzplanung sowie verschiedene Produktionsstrategien zu testen und als Robustheitsprüfung das DES-Modell zu verifizieren. Dabei handelt es sich um ein modulares DES-Modell für den Spezialtiefbau, in dem BIM integriert wird. In diesem Kapitel wird eine Simulationsstudie durchgeführt, bei der an einem realen Fallbeispiel „WTRO“ die Bauzeit unter verschiedenen Szenarien des Projekts vorhergesagt wird. Dabei kann zwischen Pull- und Push-Steuerung unterschieden werden. Für beide Szenarien werden die Simulationsstudien durchgeführt. Im Folgenden werden zuerst die Datenbeschaffung und die Datenaufbereitung beschrieben. Dann wird die Modellierung anhand des Flussdiagramms erklärt. Zuletzt werden die Simulationsergebnisse erläutert.

### 5.1 Datenbeschaffung

Die verschiedenen Daten sind über das WEB-BGM abrufbar, wie „B-Report“-Daten, „B-Activity“-Daten und Plandaten (siehe Kapitel 4.1).

### 5.2 Datenaufbereitung

Für die Simulationsstudie am Fallbeispiel „WTRO“ wird die Herstellungszeit des Bohrpfahls auf Basis der Taktzeit untersucht. Es wurde lediglich beachtet, dass die Daten aus dem Jahr 2019 übernommen wurden. Das Fallbeispiel „WTRO“ beinhaltet 232 Bohrpfähle der Achsen 10 bis 190. Aus den manuell erfassten Herstellprotokollen wird die Dauer für die folgenden Teilprozesse ermittelt:

- Bohren
- Wartezeit 1 zwischen Bohren und Bewehren
- Bewehren
- Wartezeit 2 zwischen Bewehren und Kontraktorrohr Installieren

- Kontraktorrohr Installieren
- Wartezeit 3 zwischen Kontraktorrohr Installieren und Betonieren
- Einzelnes Einbauen, das als Betonieren betrachtet wird, während des Betoniervorgangs
- Einzelnes Warten, das nach individuellem Einbauen geschehen ist, während des Betoniervorgangs

Die Dauer der einzelnen Prozessschritte zur Bohrpfahlherstellung steht bei den Daten zur Verfügung, welche im Rahmen der Masterarbeit [Ji-2020] bearbeitet wurden. Die Herstellungszeit jedes Bohrpfahls ist abhängig von der Pfahllänge bzw. der Bohrtiefe. Die DES auf Basis von BIM wird daher mithilfe der Taktzeit entwickelt. Dadurch kann eine optimale Dauer für die Teilprozesse auf Basis der Taktzeit ermittelt werden, um in DES verwendet zu werden [Abb-2020]. Wie in Hopp und Spearman [Hop-2008], definiert die Taktzeit die Zeiteinheit, in der ein Produkt produziert werden muss (Produktionsrate), damit die Produktionsrate mit der Nachfragerate übereinstimmen kann. Tatsächlich wird die Taktzeit wie in der folgenden Formel berechnet [Abb-2020].

$$T = \frac{T_a}{P_u} \tag{5-1}$$

<i>T</i>	<i>Taktzeit</i>
<i>T<sub>a</sub></i>	<i>verfügbare Arbeitszeit</i>
<i>P<sub>u</sub></i>	<i>Anzahl der Einheiten für Produktion nach Bedarf</i>

Mithilfe der aus dem BIM-Modell extrahierten Geometriedaten und der Formel 5-1 werden die oben aufgezählten Taktzeiten berechnet. Die Berechnungsmethode wird beispielhaft für den Bohrpfahl A140.1 gezeigt. Die folgende Tabelle 5-1 stellt die Dauer für die acht Teilprozesse am Beispiel von Bohrpfahl A140.1 dar.

Tabelle 5-1: Dauer [min] für die acht Teilprozesse am Beispiel von Bohrpfahl A140.1

	Bohren	Warten1	Bewehren	Warten2	Kontraktorrohr	Warten3	Einbau	Warten
Dauer	204	117	23	8	20	28	16	12

Mit der folgenden Formel wird beispielhaft die Taktzeit für den Teilprozess Bohren berechnet. In der Formel ist dargestellt, dass die Bohrdauer in der Simulation für jeden Meter der

Pfahllänge berechnet wird.

$$T = \frac{Dauer}{Pfahllänge} = \frac{204 \text{ min}}{32,05 \text{ m}} = 6,36 \text{ min/m} \quad (5-2)$$

In der vorliegenden Arbeit wird die Simulationsstudie am Beispiel von Bohrpfahl A140.1 durchgeführt. Die Tabelle 5-2 stellt die Taktzeiten für die acht Teilprozesse am Beispiel von Bohrpfahl A140.1 dar. Dadurch wird die Taktzeit des Teilprozesses, die als Simulationsdauer betrachtet wird, auf der jeweiligen Station generiert.

Tabelle 5-2: Taktzeit [min/m] für die acht Teilprozesse am Beispiel von Bohrpfahl A140.1

	Bohren	Warten1	Bewehren	Warten2	Kontraktorrohr	Warten3	Einbau	Warten
Taktzeit	6,36	3,65	0,72	0,25	0,62	0,88	0,48	0,37

Weiterhin können für diese Daten repräsentative Wahrscheinlichkeitsdichtefunktionen als Eingangsgrößen für die Simulation dienen. Eine Dauer, die dem Eingangsparameter dieser Verteilungsfunktion entspricht, wird zufällig in der Simulation ausgewählt, um die Herstellungszeit vorherzusagen. Dabei handelt es sich in der vorliegenden Arbeit um die symmetrisch verteilte Dichteverteilung der Normalverteilung, auch bekannt unter dem Begriff der „Gaußschen Glockenkurve“, auf Basis der Taktzeit.

## 5.3 Modellierung

### 5.3.1 Modulares DES-Modell

In der vorliegenden Arbeit wird ein modulares DES-Modell unter Berücksichtigung der Betonmischer für eine bestimmte Tätigkeit Betonieren aufgebaut. Hierbei geht es im modularen DES-Modell um die drei Bereiche des Zustandsautomaten, des Ressourcenhandlings und der Visualisierung, siehe Kapitel 2.1.1. Im Zustandsautomaten durchläuft die Prozessmarke mit sämtlichen Informationen des Auftrags die aufeinander folgenden Elementartigkeiten. Zudem können die angeforderten Ressourcen für das Ressourcenhandling nicht nur für einen bestimmten Tätigkeitstyp gebucht, sondern auch wieder freigegeben

werden. Weiterhin können eine 4D-Visualisierung oder aussagekräftige Diagramme für die Ergebnisse in der Visualisierung eingesetzt werden.

Zum Aufbau des modularen DES-Modells sind bestimmte Annahmen der einzelnen Prozesse notwendig. Der Simulation liegen fünf Annahmen zugrunde, diese sind:

1. Am Anfang der Simulation werden das Material Beton und die Geräte Betonmischer gebucht und auf dem Lager platziert, um sie nach dem Bewehren freizugeben. Anschließend fahren die Betonmischer mit dem Frischbeton los.
2. Im Szenario mit Push-Steuerung fährt ein Betonmischer mit Frischbeton wieder zur Baustelle los, nachdem der Betonmischer nach der Entladung zum Mischwerk zurückgefahren ist.
3. Eine Kollision zwischen den Betonmischem wird vollständig vernachlässigt.
4. Der Transportweg zwischen der Baustelle und dem Mischwerk wird nicht berücksichtigt. Als Fahrzeit wird 16 min als Eingangsparameter in der Simulation festgestellt.
5. Die Simulation läuft ab, indem jede Komponente des Systems, wie Prozessmarke, Beton und Betonmischer, in bestimmter Sequenzierung gemäß dem Prinzip „First-In-First-Out“ (FIFO) aufgerufen wird. Nachdem der Beton somit transportiert ist, wird er nach diesem Prinzip beim Betoniervorgang abgeladen, um über ein Betonierrohr mit dem Betonmischer in die Bohrung verfüllt zu werden.

Anhand des Flussdiagramms (siehe Abbildungen 5-2, 5-3 wird ein modulares DES-Modell modelliert, siehe Abbildung 5-1. Diese zeigt ein solches Modell, welches in zwei Ebenen gegliedert ist:

- **Ebene 1:** Prozessbaustein unter Berücksichtigung der Bohrpfahlherstellung
- **Ebene 2:** Prozessbaustein unter Berücksichtigung der Betonmischer

### **Ebene 1**

Auf der Ebene 1 handelt es sich um zwei Bereiche der Visualisierung und des Zustandsautomaten. Die Ergebnisse können durch eine 2D-Visualisierung im Bereich der Visualisierung einfach interpretierbar sein. Dafür ist Bohrpfahl BP\_1 als ein Beispiel vom Bohrpfahl A 140.1 dargestellt. Im Bereich Zustandsautomaten durchläuft die Prozessmarke von links

nach rechts nacheinander verschiedene Elementartätigkeiten, wie beispielsweise Bohren und einzelnes Einbauen. Die Teilprozesse werden von den angebotenen Methoden gesteuert. Der Teilprozess einzelnes Einbauen wird als wiederholbarer Prozess angesehen. Wenn die Bohrung über das Betonierrohr des Betonmischers mit Beton vollständig verfüllt wurde, wird der Betonvorgang beendet. Das heißt, dass der Bohrpfehl fertig hergestellt ist. Darüber hinaus wird zum einen der Radlader (RL) während des Bohrens abgerufen, zum anderen der Betonmischer (BM) nach dem Bewehren, siehe Abbildung 5-2.

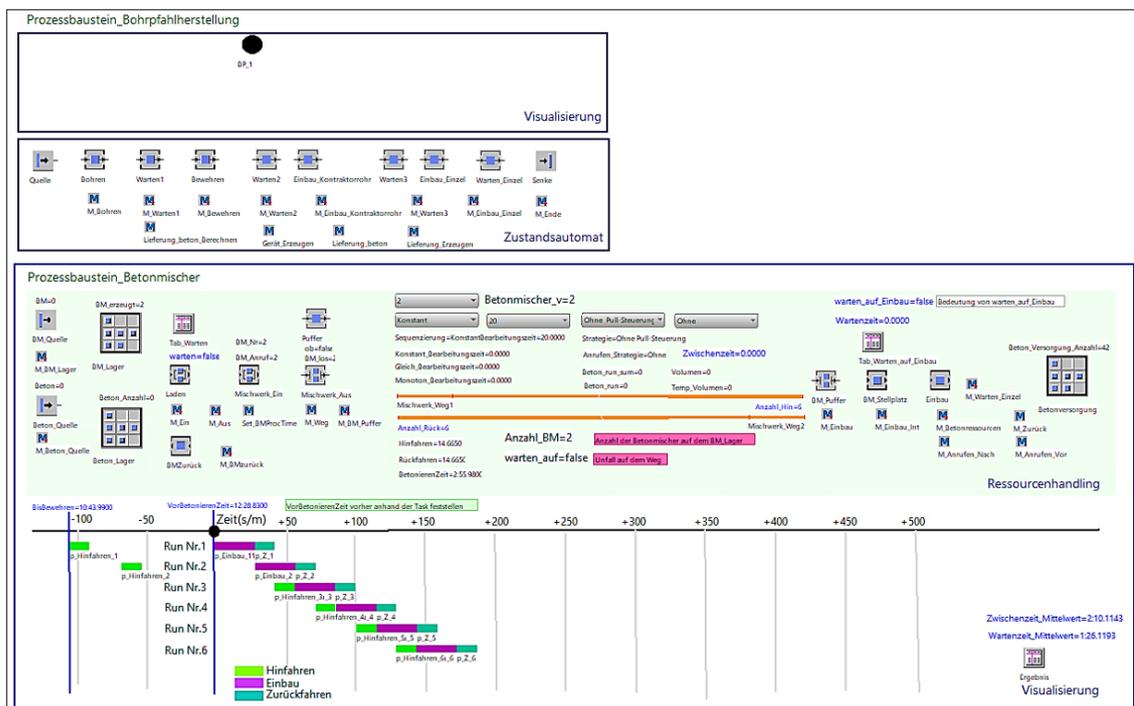


Abbildung 5-1: Modulares DES-Modell unter Berücksichtigung der Betonmischer für eine bestimmte Tätigkeit Betonieren

## Ebene 2

Auf der Ebene 2 geht es um zwei Bereiche der Visualisierung und des Ressourcenhandlings. Die Ergebnisse können durch ein aussagekräftiges Diagramm erfolgen. Die Generierung des Zeitplans für die Teilprozesse Hinfahren, einzelnes Einbauen und Zurückfahren unter Berücksichtigung der Betonmischer wird visuell dargestellt. Außerdem handelt es sich im Diagramm um die Anzahl der Läufe des Betonmischers. Im Bereich des Ressourcenhandlings gibt es Elementartätigkeiten wie beispielsweise Laden und einzelnes Einbauen. Die Geräte Betonmischer sowie das Material Beton können am Anfang gebucht und wieder vom Lager freigegeben werden. Danach fährt der Betonmischer mit dem Beton

vom Mischwerk zur Baustelle hin, siehe Abbildungen 5-3. Die wesentlichen Strukturen der Baustelle an dem Bohrpfahl werden über einen Betonmischer-Puffer, auf dem mehrere Betonmischer zur Verfügung stehen, und einen Betonmischer-Stellplatz abgebildet, auf dem sich maximal ein Betonmischer befinden kann.

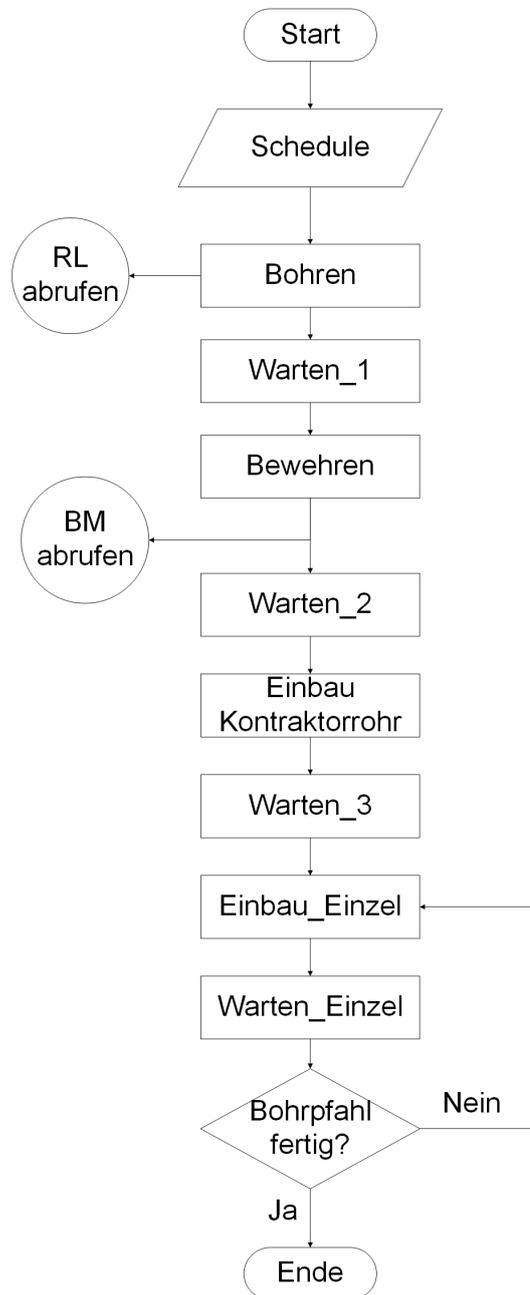


Abbildung 5-2: Prozessablauf für die Bohrpfahlherstellung. RL: Radlader, BM: Betonmischer

Darüber hinaus wird der einzelne Einbauvorgang fertig ausgeführt und der Betonmischer fährt wieder zurück, sobald er leer ist. Wenn der Pfahl vollständig betoniert ist, wird das Herstellungsverfahren beendet. Sonst wird das oben genannte Verfahren wiederholt durch-

geführt, siehe Abbildungen 5-3.

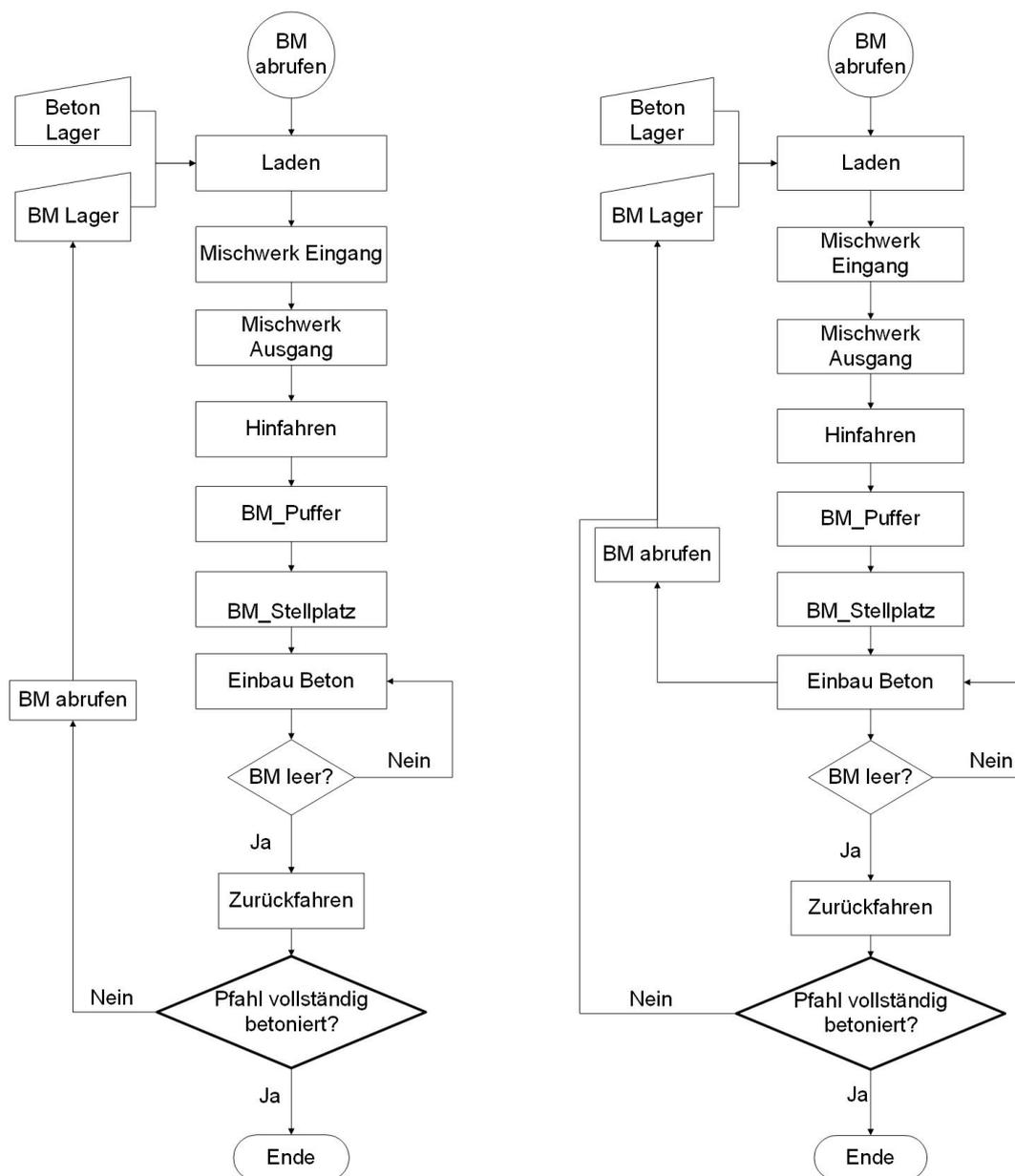


Abbildung 5-3: Szenario mit Push- (links) und Push-Steuerung (rechts)

Weiterhin wird das erste Einbauen in der zweiten Ebene begonnen, sobald die Prozessmarke in der ersten Ebene erstmals auf der Station Einbau\_Einzel steht. Der Teilprozess einzelnes Einbauen wird als wiederholbarer Prozess betrachtet. Die Anzahl der Läufe des Betonmischers wird mit den folgenden Formeln berechnet, in der  $d$  für die Durchmesser des Pfahls steht.

$$\text{Anzahl der Läufe} = \frac{\text{Volumen des Bohrlochs} + \ddot{\text{Überbetonvolumen}}}{\text{Trommelvolumen des Betonmischers}} \quad (5-3)$$

$$\text{Volumen des Bohrlochs} = \frac{\pi d^2}{4} \times \text{Bohrtiefe} \quad (5-4)$$

### 5.3.2 Modellierung der Szenarien

Die vorliegende Arbeit zeigt zwei Szenarien mit unterschiedlicher Produktionssteuerung auf. Für beide Szenarien werden die Simulationsstudien durchgeführt. Somit ergeben sich vielfältige Variationsmöglichkeiten, beispielsweise [Gün-2011]:

- Eingesetzte Ressourcen, z. B. Beton und Betonmischer
- Zuordnung der Ressourcen zu einer einzelnen Tätigkeit, z. B. Art und Anzahl des Betonmischers
- Zuordnung des Transports, z. B. Transportzeiten, Transportnetze und Transportwege

#### Erstes Szenario mit Push-Steuerung

In Abbildung 5-3 ist dargestellt, dass der Betonmischer abgerufen wird, solange der Pfahl unvollständig betoniert ist. In der vorliegenden Arbeit wird die Simulationsstudie im Szenario mit Push-Steuerung mit verschiedenen Variationsmöglichkeiten durchgeführt:

1. Anzahl der verfügbaren Betonmischer  
Die Betonmischeranzahl entspricht dem Volumen des Bohrlochs. In der vorliegenden Arbeit stehen maximal sechs Betonmischer zur Verfügung.
2. Zeitlicher Abstand der Betonmischer  
Jeder Betonmischer im Mischwerk wird in jedem Zeitintervall, wie 10 min, 20 min, 30 min oder 60 min, in bestimmter Sequenzierung abgerufen, um den Frischbeton zur Baustelle zu transportieren.
3. Fahrzeit des Betonmischers  
Es handelt sich um die Summe der Hinfahrzeit und der Zurückfahrzeit.

#### Zweites Szenario mit Pull-Steuerung

In der vorliegenden Arbeit wird der Betonmischer nach oder vor dem Betoniervorgang abgerufen, siehe Abbildung 5-3. Hierfür wird die Simulationsstudie im zweiten Szenario mit ver-

schiedenen Variationsmöglichkeiten durchgeführt. Diese Produktionsprozesse sind analog zur Umsetzung des Push-Prinzips.

Im Folgenden werden die verschiedenen Variationsmöglichkeiten erläutert:

1. Anzahl der verfügbaren Betonmischer  
Analog zur Umsetzung des Push-Prinzips.
2. Abrufen des Betonmischers  
Jeder Betonmischer im Mischwerk wird vor oder nach dem Betonieren in bestimmter Sequenzierung abgerufen. Dadurch wird der Betonmischer mit dem Frischbeton zur Baustelle transportiert, um den Bohrpfahl zu betonieren.
3. Fahrzeit des Betonmischers  
Analog zur Umsetzung des Push-Prinzips.

## **5.4 Simulationsergebnisse und Auswertung**

Eine gute Produktion zeichnet sich in der vorliegenden Arbeit durch folgende Punkte aus:

- geringe Kosten, z. B. geringe Mietkosten der Betonmischer
- geringe Bauzeit
- geringer Platzbedarf, z. B. geringer Platzbedarf des Betonmischer-Puffers
- hohe Qualität, z. B. hohe Betonqualität

Hierfür werden die Simulationsstudien in zwei Szenarien mit vielfältigen Variationsmöglichkeiten anhand der genannten Punkte verglichen, um eine optimale Maschineneinsatzplanung sowie verschiedene Produktionsstrategien zu testen, und um als Robustheitsprüfung das DES-Modell zu verifizieren.

### **5.4.1 Vergleich der Bauzeit im ersten Szenario**

Mit dem erstellten Simulationsmodell und der Hilfe des Experimentverwalters wird es ermöglicht, verschiedene Eingabeparameter automatisch zu ändern, damit die Auswirkungen

der Betonmischeranzahl und des zeitlichen Abstands zwischen den Betonmischern auf die Bauzeit in der vorliegenden Arbeit untersucht werden kann.

Mithilfe des Experimentverwalters in dem erstellten Simulationsmodell wird die Betonmischeranzahl zwischen 1 und 6 und der zeitliche Abstand zwischen den Betonmischern zwischen 0 und 60 min variiert und alle anderen Simulationsparameter beibehalten. Abbildung 5-4 zeigt die Ergebnisse eines Simulationsexperiments hinsichtlich der Betonmischeranzahl und des zeitlichen Abstands zwischen den Betonmischern und wie sich durch Erhöhung dieser beiden Parameter die Bauzeit zeitlich ändern.



Abbildung 5-4: Simulationsstudie im ersten Szenario hinsichtlich der Betonmischeranzahl und des zeitlichen Abstands zwischen den Betonmischern

Aus Abbildung 5-4 wird ersichtlich, wie viele Betonmischer bevorzugt zum Einsatz kommen. Um eine optimale Maschineneinsatzplanung zu gewährleisten, stehen zwei Betonmischer während des Betoniervorgangs zur Verfügung. Mit zwei Betonmischern besitzt der Gesamtprozess die geringste Dauer von 8,23 h. Dazu hat der zeitliche Abstand keine Auswirkung auf die Dauer des Gesamtprozesses.

### Analyse hinsichtlich der Betonmischeranzahl

In Abbildung 5-4 ist dargestellt, dass zwischen der Dauer von 9,54 h für den Gesamtprozess mit einem Betonmischer und 8,23 h für den mit zwei Betonmischern mit unterschiedlichem zeitlichen Abstand zwischen diesen ein großer Unterschied besteht. Im Vergleich zu zwei Betonmischern ist die Dauer des Gesamtprozesses mit einem Betonmischer mehr als 1,31 h länger. Die Gründe dafür werden anhand der Abbildung 5-5 erklärt.

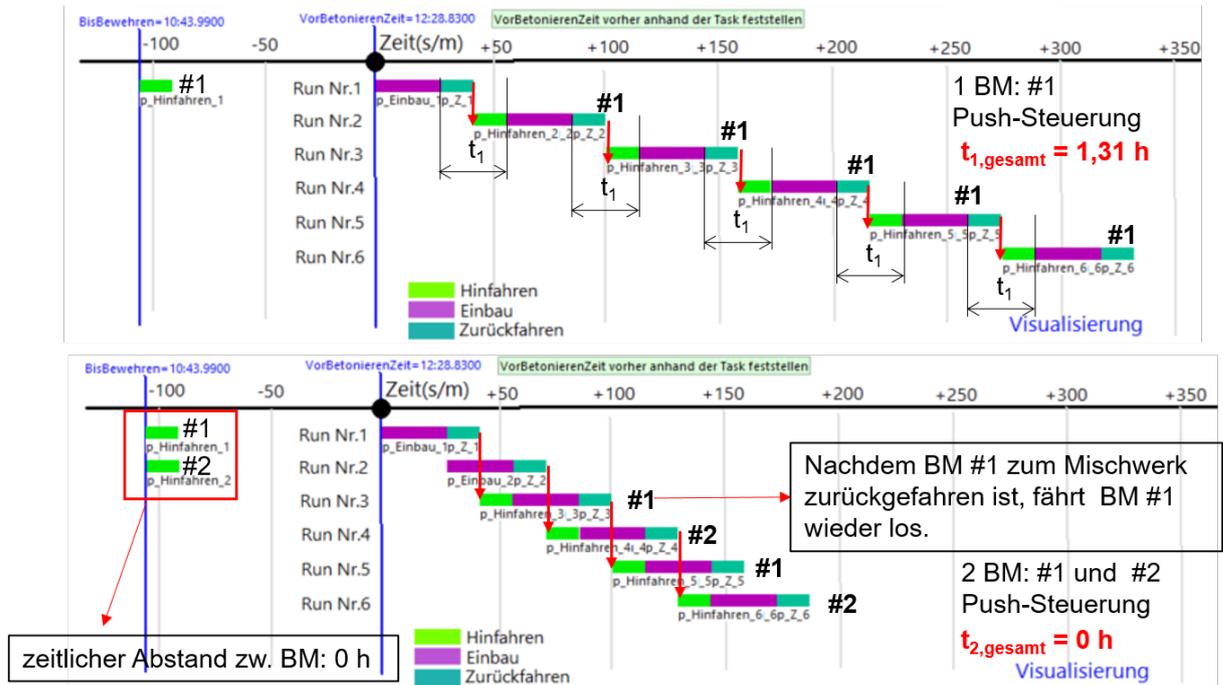


Abbildung 5-5: Visualisierung des ersten Szenarios mit einem (#1) oder zwei Betonmischern (#1 und #2) bei zeitlichem Abstand von 0 min zwischen den Betonmischern

Abbildung 5-5 verdeutlicht hier grafisch die Gründe anhand der Wartezeit  $t_i$ , die als die Zeit zwischen dem nacheinander einzelnen Einbauen beschrieben wird. Dazu werden die Visualisierungen unterschiedlicher Varianten hinsichtlich der Betonmischeranzahl im ersten Szenario mit zeitlichem Abstand von 0 h zwischen den Betonmischern dargestellt. Die Wartezeit von 1,31 h entsteht für das Szenario mit einem Betonmischer. Dagegen liegt die Dauer im Szenario mit zwei Betonmischer, wie #1 und #2, bei 0 h. Diese wird über nachstehende Formel erreicht. Dabei betragen die Hinfahrenzeit und die Zurückfahrenzeit jeweils ca. 0,13 h. In nachfolgender Formel steht  $t_{i,gesamt}$  für Gesamte Wartezeit  $i$ ,  $i = 1, \dots, i$ , wobei  $i$  die Betonmischeranzahl ist.

$$t_{1,gesamt} = (\text{Hinfahrenzeit} + \text{Zurückfahrenzeit}) \times 5 \approx 1,31 h \quad (5-5)$$

$$t_{2, \text{gesamt}} = 0 \text{ h} \quad (5-6)$$

### **Analyse hinsichtlich des zeitlichen Abstands zwischen den Betonmischern**

Der zeitliche Abstand zwischen den Betonmischern wird als Simulationsparameter betrachtet und variiert zwischen 0 und 60 min. Hier werden die Visualisierungen im ersten Szenario mit sechs Betonmischern mit unterschiedlichem zeitlichen Abstand zwischen den Betonmischern in Abbildung 5-6 gezeigt. Wenn dieser festgestellte Simulationsparameter keinen Einfluss auf die Bauzeit im ersten Szenario mit sechs Betonmischern hat, lautet eine Simulationsaussage: Dieser festgestellte Simulationsparameter hat keine Auswirkung auf die Bauzeit. Hierfür ist in Abbildung 5-6 sichtbar, dass der zeitliche Abstand von 0 bis 20 min zwischen den Betonmischern die Bauzeit nicht beeinflusst.

Im Gegensatz zur Variante 1 bis 3 handelt es sich bei den Varianten 4 und 5 um eine Wartezeit zwischen dem einzelnen Einbauen. Diese Wartezeit führt zu zeitlichen Verzögerungen; diese werden in den roten Rechtecken in Abbildung 5-6 gezeigt und führen zu hoher Bauzeit in den Varianten 4 und 5. Weiterhin ist sichtbar, dass es im ersten Szenario zu hohem Platzbedarf auf der Baustelle kommen kann.

Weiterhin beeinflusst der zeitliche Abstand von 60 min zwischen den Betonmischern im ersten Szenario mit mehr als 2 Betonmischern stark die Bauzeit, was in den blauen Rechtecken in Abbildung 5-4 gezeigt wird. Dagegen wird die Bauzeit im ersten Szenario mit mehr als 4 Betonmischern durch den zeitlichen Abstand von 30 min zwischen den Betonmischern schwach beeinflusst. Die oben genannten zeitlichen Verzögerungen führen zu schwachen oder starken Auswirkungen auf die Bauzeit. Daher sind die optimalen zeitlichen Abstände zwischen den Betonmischern 0 min, 10 min und 20 min. Für den weiteren Vergleich der beiden Szenarien in der vorliegenden Arbeit wird der zeitliche Abstand von 20 min zwischen den Betonmischern ausgewählt. Im Anhang D befinden sich die weiteren Visualisierungen.

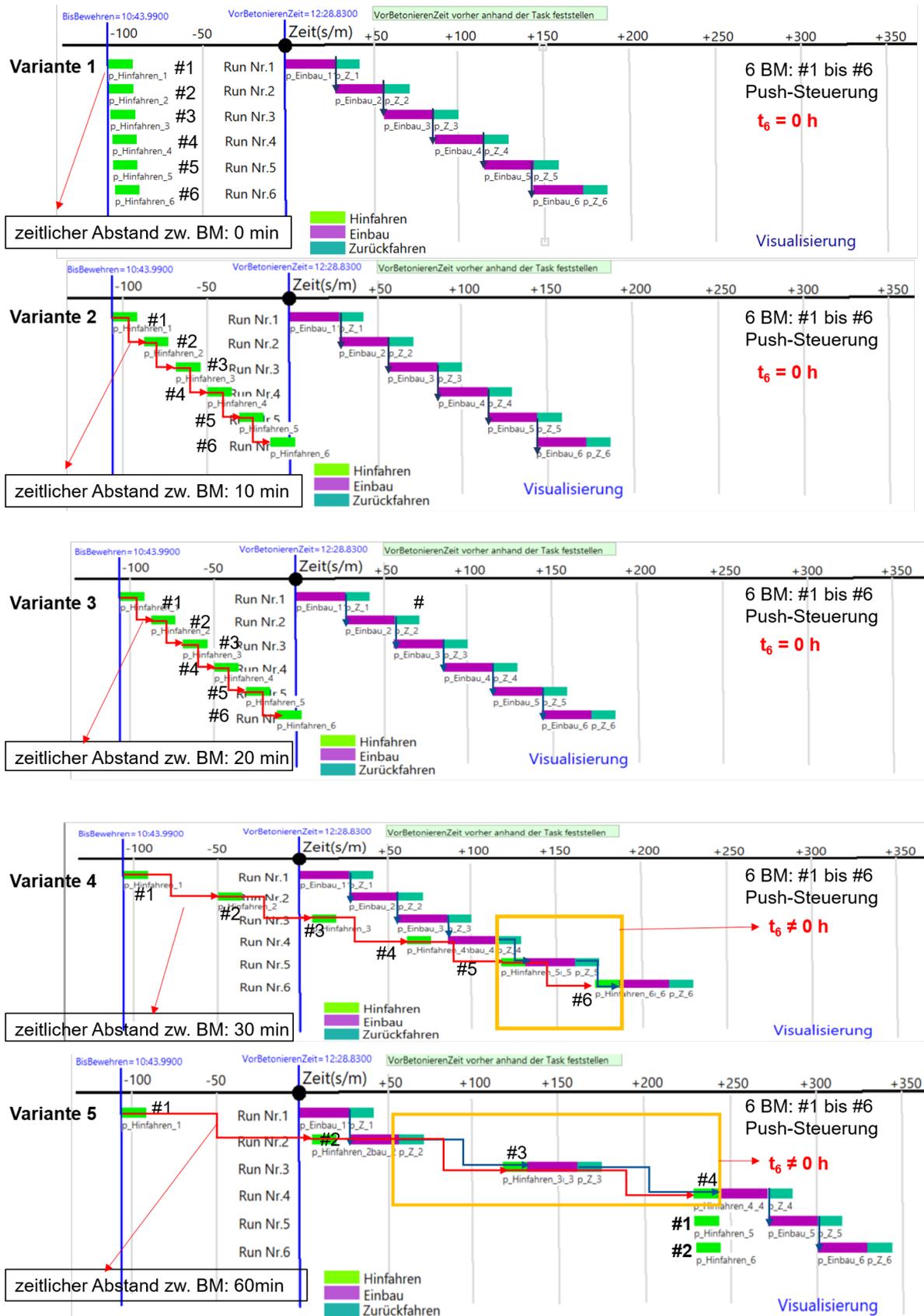


Abbildung 5-6: Visualisierung des ersten Szenarios mit sechs Betonmischern (#1 bis #6) bei zeitlichem Abstand von 0 bis 60 min zwischen den Betonmischern

## Analyse hinsichtlich der Fahrzeit

Im ersten Szenario mit zwei Betonmischern mit dem zeitlichen Abstand von 20 min wird die Simulationsstudie hinsichtlich der Fahrzeit durchgeführt. In Abbildung 5-7 ist dargestellt, dass die Bauzeit mit zunehmender Fahrzeit stark steigt und die Fahrzeit unter 16 min keine Auswirkungen auf die Bauzeit, die bei 8,23 h liegt, in dieser Situation hat. Die optimale Bauzeit ist 8,23 h.

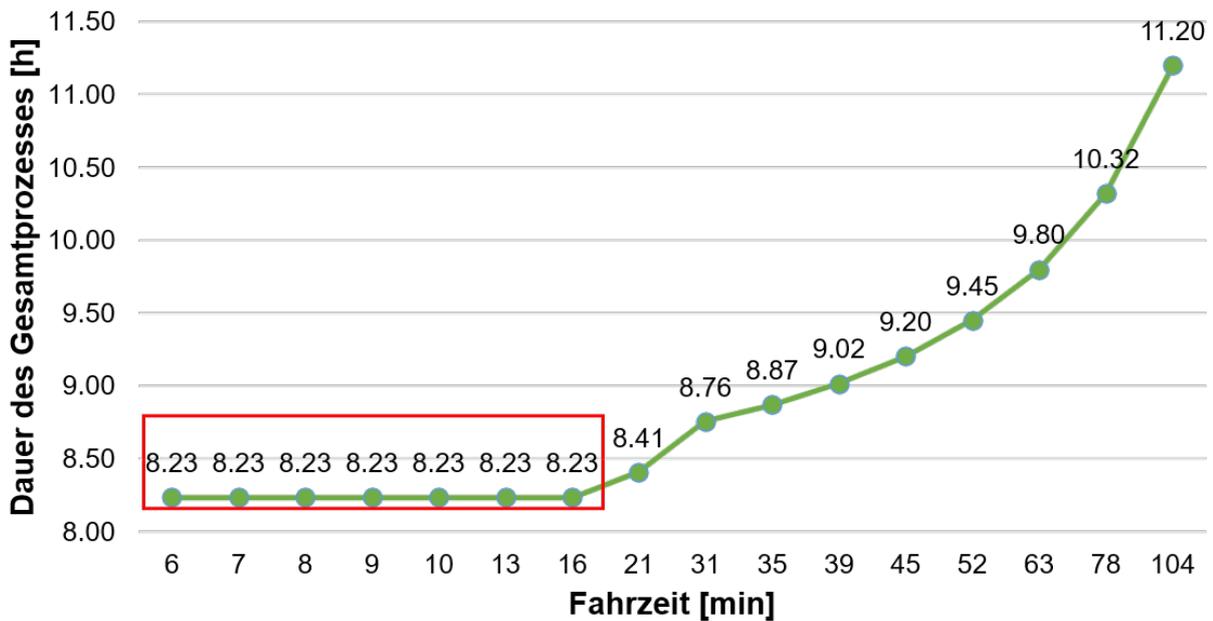


Abbildung 5-7: Simulationsstudie im zweiten Szenario hinsichtlich der Fahrzeit der Betonmischer

## Analyse hinsichtlich der Betonqualität

In Abbildung 5-8 ist die Transportzeit der Betonmischer im ersten Szenario mit zwei Betonmischern mit einem zeitlichen Abstand von 20 min dargestellt. Die Betonqualität ist von der Transportzeit der Betonmischer abhängig, die nicht mehr als 90 min sein darf. Das heißt, dass die Transportzeit der Betonmischer unter 90 min sein muss, um die Betonqualität zu gewährleisten. Die Transportzeiten der Betonmischer liegen unter 90 min. Deswegen steht eine gute Betonqualität in der Situation zur Verfügung.

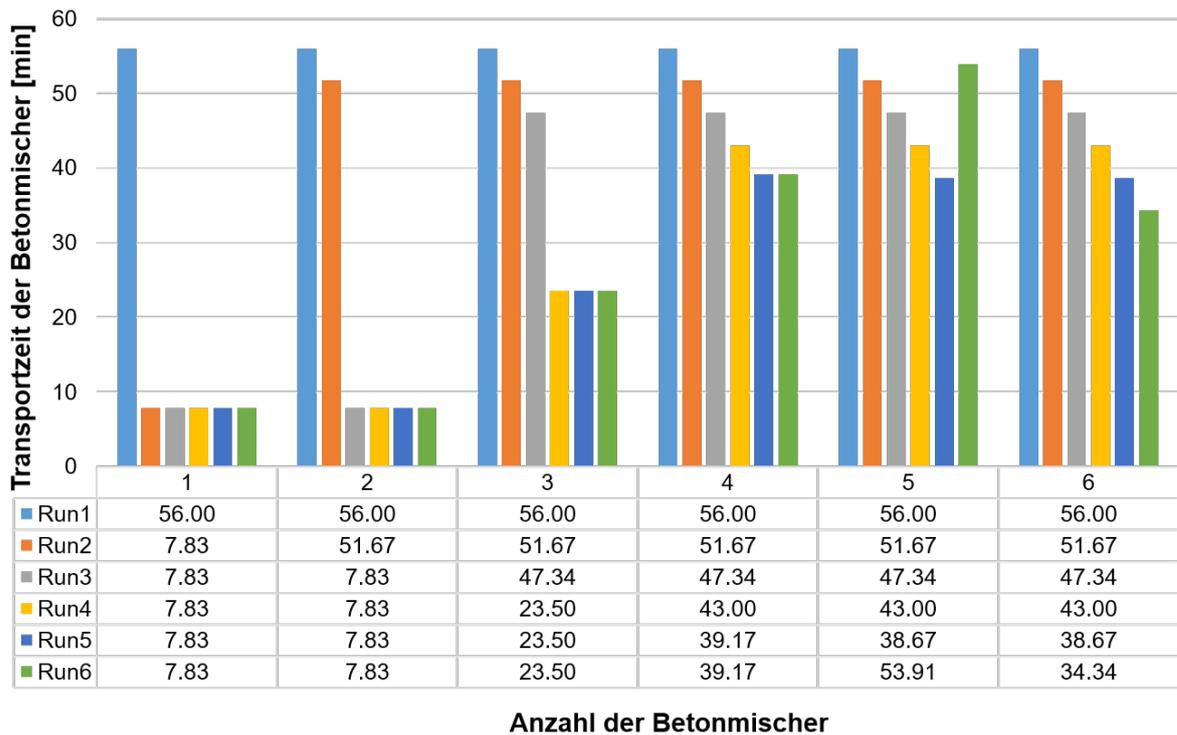


Abbildung 5-8: Simulationsstudie im zweiten Szenario hinsichtlich der Betonmischeranzahl

#### 5.4.2 Vergleich der Bauzeit im zweiten Szenario

Für das zweite Szenario wird die Betonmischeranzahl als Simulationsparameter betrachtet. Diese Produktionsprozesse sind analog zur Umsetzung des Push-Prinzips. Dabei handelt es sich im zweiten Szenario um zwei Varianten:

- **Variante 1:** Der Betonmischer im Mischwerk wird vor dem Betonieren in bestimmter Sequenzierung abgerufen.
- **Variante 2:** Der Betonmischer im Mischwerk wird nach dem Betonieren in bestimmter Sequenzierung abgerufen.

Mit dem erstellten Simulationsmodell und der Hilfe des Experimentverwalters wird die Simulationsstudie hinsichtlich der Betonmischeranzahl unter den verschiedenen Varianten durchgeführt, siehe Abbildung 5-9. Hier sind die Bauzeiten im zweiten Szenario mit 2 bis 6 Betonmischern in der Variante 1 höher als in der Variante 2. Daher wird der Betonmischer bevorzugt vor dem Betonieren abgerufen. Des Weiteren kommen die zwei Betonmischer im zweiten Szenario aufgrund der geringen Bauzeit und der geringen Mietkosten der Betonmischer zum Einsatz. Außerdem hat die Betonmischeranzahl keinen Einfluss auf die Bauzeit

unter beiden Varianten im zweiten Szenario, wenn mehr als ein Betonmischer zum Einsatz kommt.

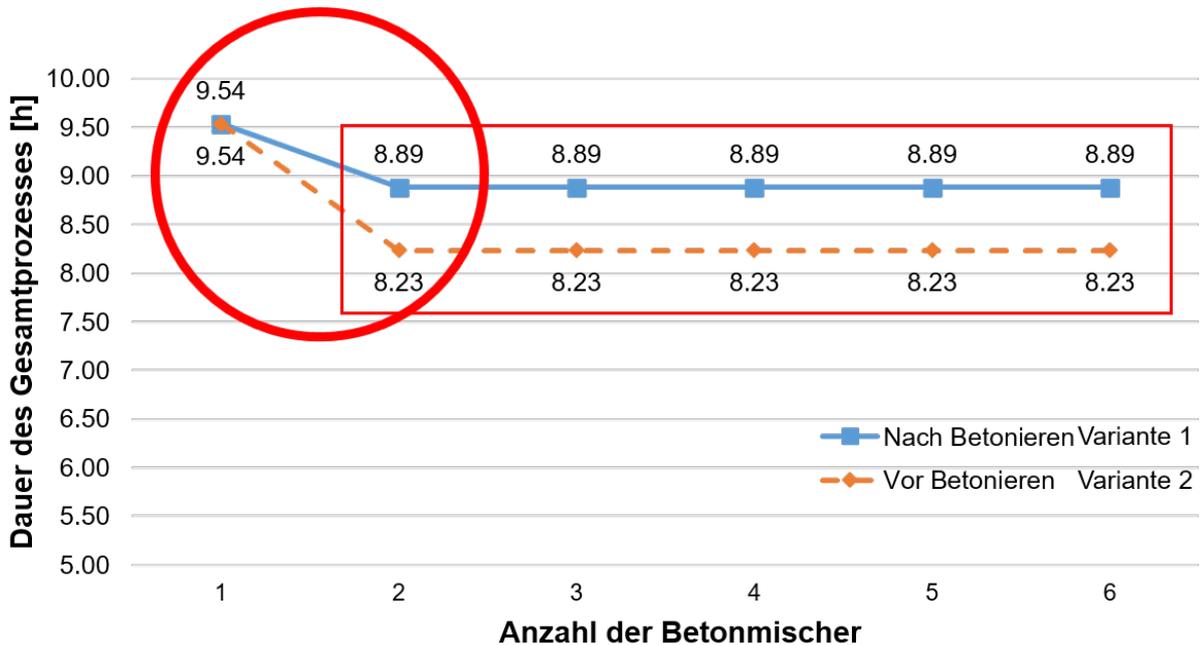


Abbildung 5-9: Simulationsstudie im zweiten Szenario hinsichtlich der Betonmischeranzahl

### Analyse hinsichtlich der verschiedenen Varianten

Die Ursache für die starke Absenkung der Bauzeit im roten Kreis in Abbildung 5-9 wird durch die Visualisierung in der Simulation in Abbildung 5-10 dargestellt. Im zweiten Szenario mit einem Betonmischer steht eine niedrigere Wartezeit  $t_1$  als mit zwei Betonmischern. Aufgrund der geringen Wartezeit führt das zu geringen Verzögerungen und damit zu einer geringeren Bauzeit als im zweiten Szenario mit zwei Betonmischern.

### Analyse hinsichtlich der Fahrzeit der Betonmischer

Die Simulationsstudie wird hinsichtlich der Fahrzeit im zweiten Szenario mit zwei Betonmischern unter verschiedenen Varianten durchgeführt. In Abbildung 5-11 ist dargestellt, dass die Bauzeit mit zunehmender Fahrzeit im zweiten Szenario mit zwei Betonmischern unter der Variante 1 und 2 steigt und daneben die Fahrzeit unter 16 min keine Auswirkungen auf die Bauzeit, die bei 8,23 h liegt, in dieser Situation unter der Variante 2 hat. Im Vergleich mit Variante 1 besitzt der Bauprozess in der Situation unter Variante 2 eine niedrigere Bauzeit,

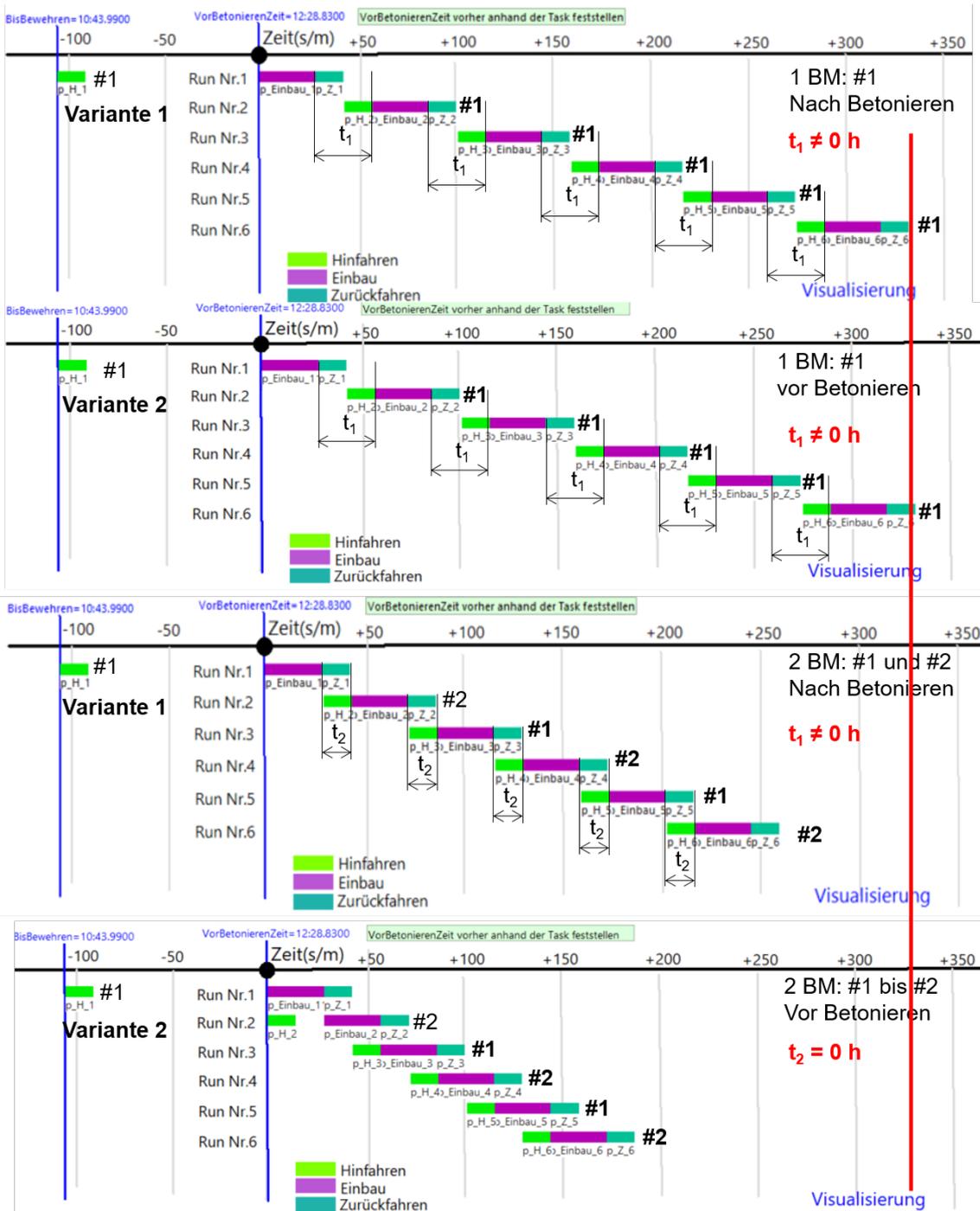


Abbildung 5-10: Visualisierung des zweiten Szenarios mit ein (#1) und zwei Betonmischern (#1 und #2)

siehe Abbildung 5-11. Daher wird die optimale Variante 2 im zweiten Szenario in der vorliegenden Arbeit ausgewählt. Das heißt, der Betonmischer wird bevorzugt vor dem Betonieren abgerufen.

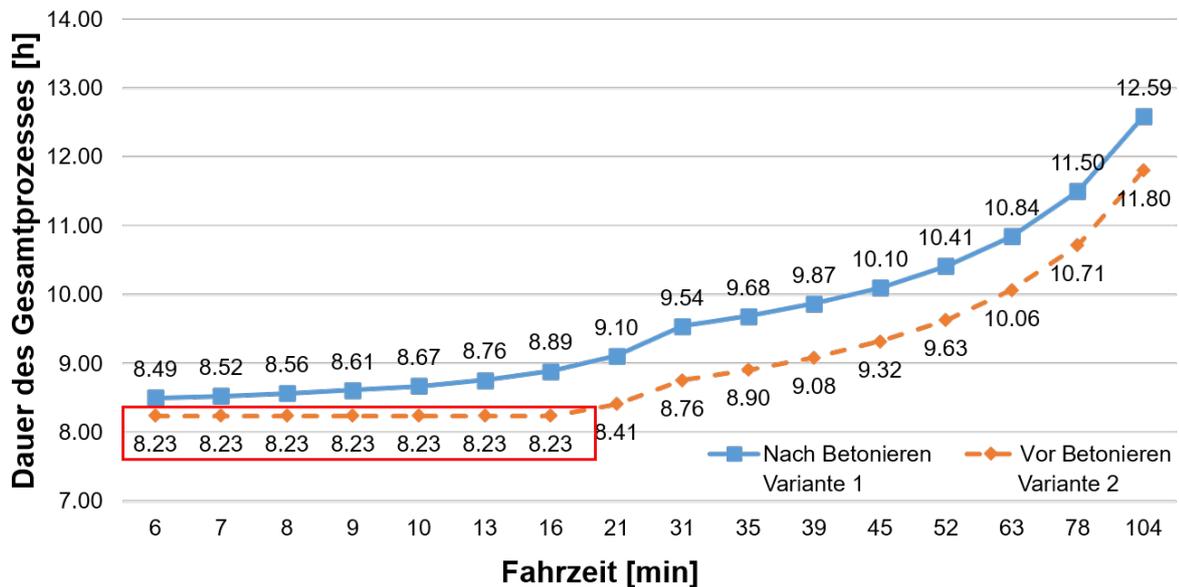


Abbildung 5-11: Simulationsstudie im zweiten Szenario hinsichtlich der Fahrzeit der Betonmischer

### 5.4.3 Vergleich der Bauzeit in beiden Szenarien

Im Folgenden werden zwei Szenarien mit verschiedenen Produktionssteuerungen verglichen. Dabei werden die folgenden Simulationsparameter festgestellt:

- Zeitlicher Abstand zwischen den Betonmischern im ersten Szenario: 20 min
- Abrufen der Betonmischer: Abrufen vor Betonieren (Variante 2) im zweiten Szenario

### Analyse hinsichtlich der Fahrzeit der Betonmischer

In Abbildung 5-12 ist dargestellt, dass die Bauzeiten in beiden Szenarien mit zwei Betonmischern bei einer Fahrzeit unter 31 min gleich sind. Daneben ist sichtbar, dass die Fahrzeit unter 16 min in beiden Szenarien mit zwei Betonmischern keine Auswirkungen auf die Bauzeit, die bei 8,23 h liegt, hat.

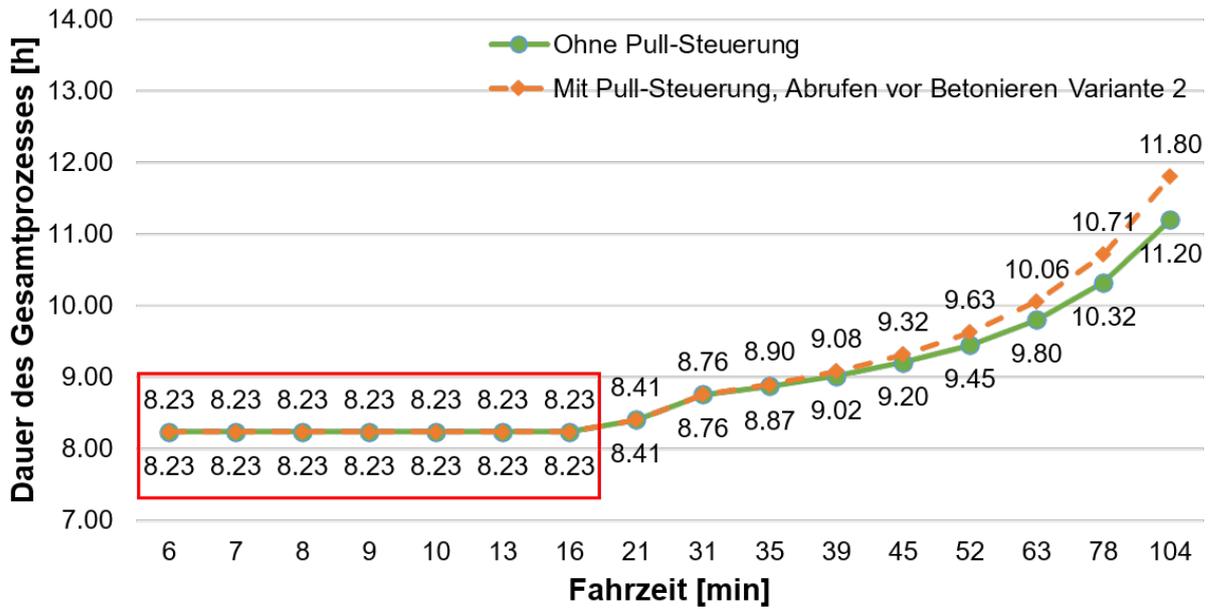


Abbildung 5-12: Simulationsstudie in beiden Szenarien hinsichtlich der Fahrzeit der Betonmischer

### Analyse hinsichtlich der Betonmischeranzahl

Die Bauzeiten in beiden Szenarien sind gleich, siehe Abbildung 5-13. Aus Abbildung 5-13 wird ersichtlich, dass die Betonmischer in beiden Szenarien keine Auswirkungen auf die Bauzeit, die bei 8,23 h liegt, haben. Zwei Betonmischer kommen in beiden Szenarien zum Einsatz, um die optimale Maschineneinsatzplanung zu erreichen.

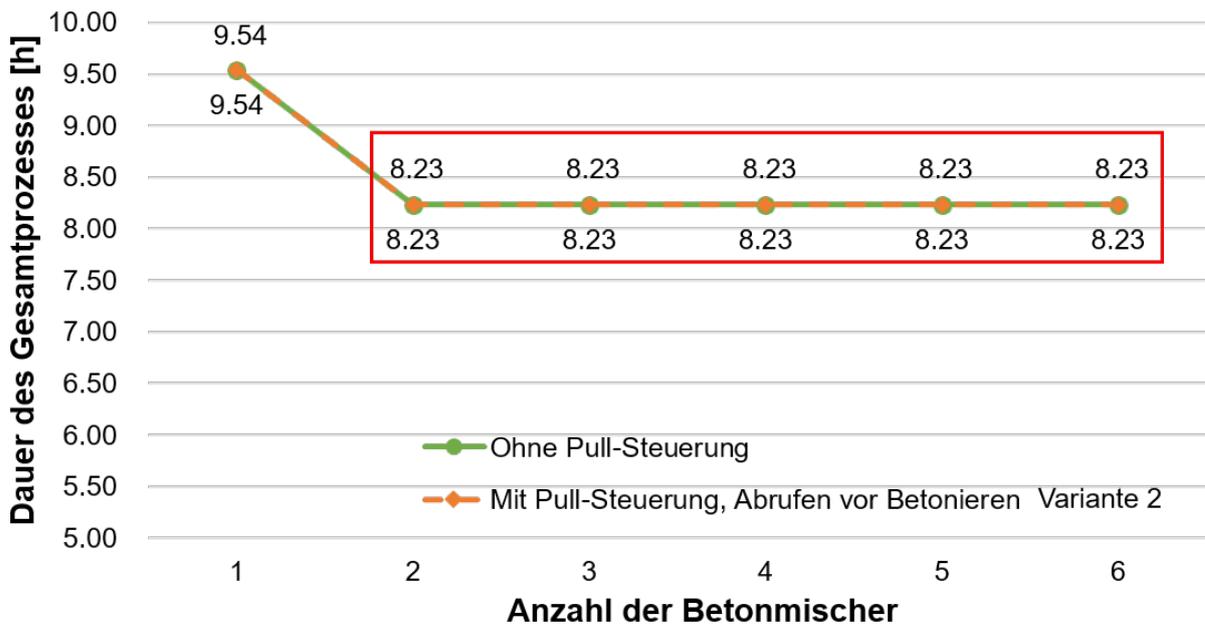


Abbildung 5-13: Simulationsstudie in beiden Szenarien hinsichtlich der Betonmischeranzahl

### Analyse in beiden Szenarien mit Stau

Es gibt keinen großen Unterschied zwischen den beiden Szenarien mit zwei Betonmischern und Stau, in denen gleiche Standardabweichungen zur Verfügung stehen, siehe Tabelle 5-3. Aufgrund der wenigen Bohrpfähle und damit geringen Anzahl der Transporte kann es zu keinem Unterschied führen.

Tabelle 5-3: Experiment mit 10.000 Beobachtungen in beiden Szenarien mit Stau

Experiment	Dauer des Gesamtprozesses [h]	Standardabweichung [h]	Minimum [h]	Maximum [h]
Push	8,87	0,13	8,43	9,24
Pull	8,89	0,13	8,43	9,33

### Analyse hinsichtlich der Betonmischeranzahl auf Basis der Daten einer Normalverteilung

Mithilfe der Normalverteilung wird die Dauer für die Teilprozesse in der Simulation repräsentiert. In Abbildung 5-14 ist dargestellt, dass die Bauzeit in den beiden Szenarien mit zwei Betonmischern auf Basis der Daten einer Normalverteilung in der Simulation nicht am niedrigsten ist. Daher ist es möglich, dass drei Betonmischer zum Einsatz kommen, um zeitliche Verzögerungen zu vermeiden.

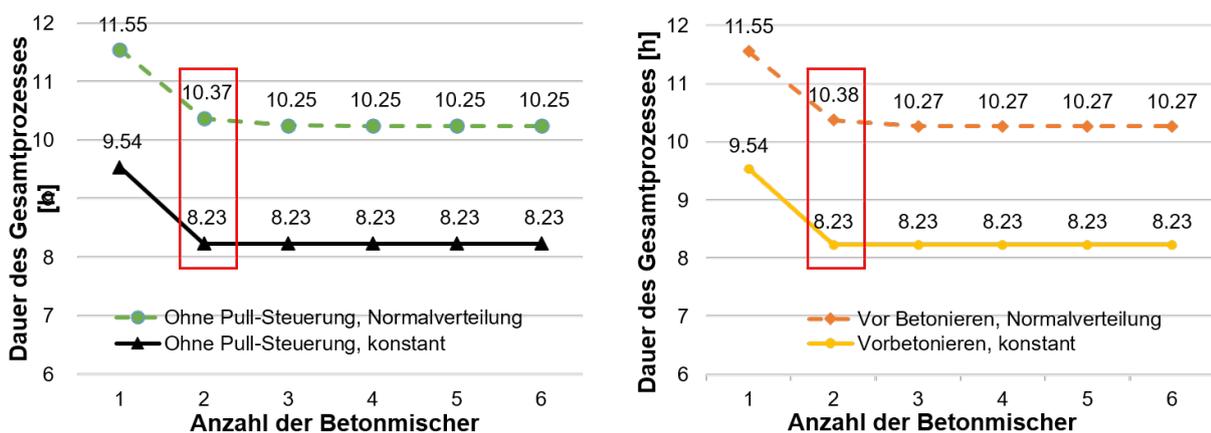


Abbildung 5-14: Experiment mit 10.000 Beobachtungen in beiden Szenarien hinsichtlich der Betonmischer

## 6 Zusammenfassung und Ausblick

---

### 6.1 Zusammenfassung

In der vorliegenden Arbeit wird BIM in ein DES-Modell zur Abbildung des Spezialtiefbauprozesses im Rahmen des Brückenbauprojekts „WTRO“ effektiv integriert.

In Kapitel 2 werden die für diese Arbeit wesentlichen Grundlagen aus den Bereichen BIM und DES im Bauwesen vorgestellt. Hierzu werden die Optimierungsansätze in Bezug auf den Maschineneinsatz und die Modellierungsansätze im Bauwesen (Kapitel 2.1), wie DES, ABM, SD und hybride Simulation vorgestellt. Die Umsetzung der Integration der verschiedenen Baufortschrittsdaten (Kapitel 2.2), wie Photogrammetrie, Videoanalyse und Sensordaten und die Grundlagen von BIM im Bauwesen (Kapitel 2.3), wie zugrundeliegende Begriffe und Einordnung zu Vorgängerprojekten werden erläutert. Zudem wird IFC-Format (Kapitel 2.4) ausführlich beschrieben.

In Kapitel 3 werden die für die Modellierung des IFC-Formats benötigten Technologien beschrieben. Darin werden die Programmiersprachen (Kapitel 3.1), die IFC-Bibliotheken (Kapitel 3.2) sowie die Softwarelösungen für die Modellierung (Kapitel 3.3) vorgestellt.

In Kapitel 4 und 5 liegt der Fokus der vorliegenden Arbeit; sie widmen sich der Beantwortung der in Kapitel 1.2 gestellten Fragen.

In Kapitel 4 wird die Modellierung des IFC-Formats an einem Fallbeispiel durchgeführt. Darüber hinaus werden die Eigenschaften des IFC-Modells zuerst eindeutig klassifiziert (Kapitel 4.1). Danach kann das IFC-Modell anhand des UML-Diagramms aufgebaut werden (Kapitel 4.2). Zuletzt werden die Eigenschaften als Eingangsparameter aus dem IFC-Modell in die Software Plant Simulation von Siemens PLM exportiert und abgespeichert (Kapitel 4.3).

In Kapitel 5 wird die Simulationsstudie auf Basis des IFC-Modells durchgeführt. Dabei handelt es sich zunächst um die Datenbeschaffung (Kapitel 5.1) und die Datenaufbereitung (Kapitel 5.2). Anschließend wird das Simulationsmodell auf Basis des IFC-Modells entwickelt (Kapitel 5.3). Dazu wird ein modulares DES-Modell für den Spezialtiefbau entwickelt.

Weiterhin wird eine Simulationsstudie (Kapitel 5.4) auf Datenbasis eines realen Anwendungsfalls durchgeführt, um eine optimale Maschineneinsatzplanung sowie verschiedene Produktionsstrategien zu testen, und um als Robustheitsprüfung das DES-Modell zu verifizieren.

Darüber hinaus kommen zwei Betonmischer unter den Annahmen der vorliegenden Arbeit aufgrund der geringen Bauzeit und der geringen Mietkosten zum Einsatz, um die optimale Maschineneinsatzplanung zu erreichen. Im Szenario mit Push-Steuerung kann es zu hohem Platzbedarf im Gegensatz zum Szenario mit Pull-Steuerung kommen und es ist optimal, dass ein zeitlicher Abstand von 0 bis 20 min ausgewählt wird. Im Szenario mit Pull-Steuerung wird der Betonmischer vor dem Betonieren bevorzugt abgerufen. Es ist möglich, dass drei Betonmischer aufgrund der Vermeidung von zeitlichen Verzögerungen zum Einsatz kommen. Die Empfehlungen in der vorliegenden Arbeit werden nicht mit der Praxis abgeglichen. Zudem handelt es sich in der Simulation um fünf Annahmen. Erstens: Das Material Beton und die Geräte Betonmischer werden am Anfang der Simulation zuerst erzeugt und gelagert, damit die Betonmischer mit dem Frischbeton nach dem Bewehren vom Mischwerk zur Baustelle hinfahren. Zweitens: Der Betonmischer mit dem Frischbeton fährt im Szenario mit Push-Steuerung wieder zur Baustelle hin, nachdem ein Betonmischer nach dem Betoniervorgang zum Mischwerk zurückgefahren ist. Drittens: Die Simulationsstudie wird unter Vernachlässigung der Kollision zwischen den Betonmischern durchgeführt. Viertens: Der Transportweg zwischen der Baustelle und dem Mischwerk wird nicht beachtet. Hierbei wird die Fahrzeit von 16 min als Eingangsparameter in der Simulation festgestellt. Zuletzt werden die Betonmischer in der Reihenfolge der Transportanforderungen (FIFO) in der Simulation abgerufen.

Weiterhin wird ein IFC-Format als Schnittstelle für den Informationsaustausch zwischen dem BIM-Modell und dem DES-Modell betrachtet. Dadurch werden die Informationen aus dem BIM-Modell in dem DES-Modell abgespeichert. Hierfür wird das IFC-Format als Datenbasis des digitalen Zwillings auf Basis der DES im Bauwesen entwickelt. Außerdem können die BIM-Daten in der Ausführungsphase durch Einbindung der Sensordaten in der DES aktualisiert werden. Das führt zur Vereinfachung des digitalen Zwillings im Bauwesen. In der Zukunft spielt die Einbindung von BIM-Daten in den digitalen Zwilling sowohl in der Planungsphase als auch in der Ausführungsphase eine wesentliche Rolle. Des Weiteren

gibt es für den Spezialtiefbau bis dato keinen entsprechenden Ansatz in der Literatur, um BIM-Daten für die Simulation aufzubereiten.

Neben der BIM Interoperabilität, zeigt der vorgestellte Modellierungsansatz eine hohe Skalierung auf. Durch die Modellierung des Bohrablaufs, ergänzt um die Arbeitsschritte des Drehbohrgeräts, des Radladers und der Betonmischer, wird eine Anwendung auf weitere Projekte zur Bohrpfahlerstellung ermöglicht.

## 6.2 Ausblick

Mit der vorliegenden Arbeit wird BIM auf Basis von DES an einem Fallbeispiel im Bauwesen entwickelt. Dazu wird das IFC-Format als die Schnittstelle für den Informationsaustausch zwischen dem BIM-Modell und dem DES-Modell entwickelt. Darüber hinaus wird das Simulationsmodell manuell anhand des Flussdiagramms hinsichtlich der Betonmischer generiert und anschließend wird die Simulationsstudie mit unterschiedlicher Produktionssteuerung durchgeführt. Dazu ergeben sich verschiedene Variationsmöglichkeiten.

Als erstes könnten in einer weiteren Arbeit die Logistikprozesse mit der BPMN modelliert werden; dabei könnte das Simulationsmodell mit Hilfe von BPMN möglicherweise automatisiert generiert werden. Weiterhin könnte die IFC-Datei entwickelt werden und alle Bohrpfähle im Rahmen des Brückenbauprojekts „WTRO“ beinhalten.

Eine weitere Möglichkeit zur Optimierung wäre, dass die Daten aus der IFC-Datei sowohl in DES exportiert als auch umgekehrt durch DES aktualisiert würden. Das bedeutet, dass die Daten von DES nach BIM exportiert werden. Zudem können die benötigten Daten für die Simulation aus dem BIM-Modell schnell und effektiv durch einem BIM Server extrahiert und aufbereitet werden [Ism-2014]. Außerdem könnten das Positionssystem und das Sensorsystem, die mit der DES verknüpft werden können, im Bauwesen zum Einsatz kommen. Dadurch können die BIM-Daten aktualisiert werden. Hierfür ist es für den digitalen Zwilling von Bedeutung, die BIM-Daten in der Planungsphase und in der Ausführungsphase einzubinden. Das führt zu einem effizienten, wirtschaftlichen und sicheren Bauverfahren.

Darüber hinaus könnten die diskreten Daten eines Datensatzes als Häufigkeitsdichte betrachtet werden. Es handelt sich um die fünf an die Häufigkeitsdichte angepassten gängigen

Wahrscheinlichkeitsdichtefunktionen: Normal, Lognormal, Weibull, Gamma und Exponential. Die geeigneten Wahrscheinlichkeitsverteilungen werden für jeden Teilprozess in der Simulation auf Basis der Taktzeit durch den Anpassungstest ausgewählt.

## Literaturverzeichnis

---

- [Abb-2020] Abbasi, S. et al.: BIM-Based Combination of Takt Time and Discrete Event Simulation for Implementing Just in Time in Construction Scheduling under Constraints. In: Journal of Construction Engineering & Management 146.12 (2020), S. 04020143.
- [Abd-2020] Abdelmegid, M. A. et al.: The roles of conceptual modelling in improving construction simulation studies: A comprehensive review. In: Advanced Engineering Informatics 46 (2020), S. 101175.
- [Alk-2019] Alkaissy, M. et al.: Process Modelling in Civil Infrastructure Projects: A Review of Construction Simulation Methods. In: Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC). Hrsg. von Al-Hussein, Mohamed. Proceedings of the International Symposium on Automation and Robotics in Construction (IAARC). International Association for Automation and Robotics in Construction (IAARC): 2019.
- [Aza-2011] Azadeh, A. et al.: A fuzzy simulation algorithm for estimating availability functions in time-dependent complex systems. In: International Journal of Industrial and Systems Engineering 7.4 (2011), S. 429.
- [Ban-2013] Banks, J.: Discrete-event system simulation: Pearson new international edition. fifth. Pearson Education (US). In: (2013).
- [Bau-2021] BAUER Maschinen GmbH: BAUER B-Tronic. [https://www.bauer.de/export/shared/documents/pdf/bma/datenblatter/B-Tronic\\_System\\_DE\\_EN\\_905\\_796\\_1\\_2.pdf](https://www.bauer.de/export/shared/documents/pdf/bma/datenblatter/B-Tronic_System_DE_EN_905_796_1_2.pdf), Zugriff am 09.05.2021. 2021.
- [Ben-2015] Ben-Alon, L. et al.: Simulating the behavior of trade crews in construction using agents and building information modeling. In: Automation in Construction 74 (2015), S. 12–27.
- [Bon-1999] Bonney, M. C. et al.: Are push and pull systems really so different? In: International Journal of Production Economics 59.1-3 (1999), S. 53–64.

- [Bor-2021] Borrmann, A.: ProgressTrack. <https://www.cms.bgu.tum.de/de/17-research-projects/47-progresstrack-automatisierte-baufortschrittskontrolle>, Zugriff am 09.05.2021.
- [Bor-2015] Borrmann, A. et al.: Building Information Modeling. Springer Fachmedien Wiesbaden, Wiesbaden: 2015.
- [Bor-2018] Borrmann, A. et al., Hrsg.: Building Information Modeling: Technology Foundations and Industry Practice. Springer, Cham: 2018.
- [Bra-2014] Brailsford, S. et al.: Discrete-event simulation and system dynamics for management decision making. Wiley, Chichester West Sussex: 2014.
- [Bra-2013] Braun, A.: Entwicklung eines 4D-BIM-Viewers mit graphbezogener Darstellung von Bauabläufen und -alternativen. Masterthesis. München: Technische Universität München, 2013.
- [BS-1993] BS 5192: Guide to production Control Part I. Entwurf. London: 1993.
- [Büg-2017] Bügler, M. et al.: Fusion of Photogrammetry and Video Analysis for Productivity Assessment of Earthwork Processes. In: Computer-Aided Civil and Infrastructure Engineering 32.2 (2017), S. 107–123.
- [Bui-2021] BuildingSMART: IfcTask. <https://standards.buildingsmart.org/>, Zugriff am 13.05.2021. 2021.
- [Bun-2017] Bundesministerium für Verkehr und digitale Infrastruktur: Pilotprojekt Autobahndirektion Südbayern: A 99 BW 27/1 Unterführung der DB + S 8 Flughafen Ersatzneubau. In: (2017).
- [Dal-2018] Dallasega, P. et al.: Untersuchung des Potenzials für KMU zur Unterstützung der Baulieferkette mit Building Information Modeling: eine Fallstudie eines ETO-Fassadenlieferanten. In: KMU 4.0 - Digitale Transformation in kleinen und mittelständischen Unternehmen. Hrsg. von Matt, Dominik T. GITO Verlag: 2018, S. 244–266.
- [Eld-2016] Eldabi, T., Hrsg.: Simulating complex service systems: Proceedings of the 2016 Winter Simulation Conference. IEEE und Omnipress, Piscataway, NJ und Madison, WI: 2016.

- [Ell-2010] Eller, F.: Visual C# 2010: Grundlagen, Programmier-techniken, Datenbanken. Addison-Wesley, Munich: 2010. Url: <http://proquest.tech.safaribooksonline.de/9783827329165>.
- [Fis-2020] Fischer, A., Hrsg.: 8. Fachtagung Baumaschinentechnik 2020 - Automatisierung, Antriebssysteme, Bauverfahren. 2020.
- [Fis-1973] Fishman, G. S. et al.: Concepts and methods in discrete event digital simulation. John Wiley & Sons, New York: 1973.
- [Fra-2011] Franz, V.: Simulation von Unikatprozessen: Neue Anwendungen aus Forschung und Praxis ; 24. März 2011 an der Universität Kassel. Hrsg. von Franz, Volkhard. Bd. Band 8. Schriftenreihe Bauwirtschaft 3: Tagungen und Berichte. Kassel University Press, Kassel: 2011.
- [Gal-2018] Galvão Scheidegger, A. et al.: An introductory guide for hybrid simulation modelers on the primary simulation methods in industrial engineering identified through a systematic review of the literature. In: Computers & Industrial Engineering 124 (2018), S. 474–492.
- [Gün-2018] Günther, M. et al.: Durchgängiges modellbasiertes Engineering von Gebäudeautomationssystemen. In: at - Automatisierungstechnik 64.6 (2018).
- [Gün-2015] Günthner, W. A. et al.: FAUST - Fertigungssynchrone Ablaufsimulation von Unikatbaustellen im Spezialtiefbau. In: (2015).
- [Gün-2011] Günthner, W. A. et al.: Digitale Baustelle- innovativer Planen, effizienter Ausführen. Springer Berlin Heidelberg, Berlin, Heidelberg: 2011.
- [Gün-2010] Günthner, W. A. et al.: ForBau - Virtuelle Baustelle Abschlussbericht 2010: Digitale Werkzeuge für die Bauplanung und -abwicklung. 2010.
- [Har-2014] Hartmann, R., Hrsg.: HOAI 2013: Band 4: Ingenieurbauwerke, Verkehrsanlagen, Tragwerksplanung, Technische Ausrüstung. 1. Aufl. Beuth Recht. Beuth Verlag GmbH, s.l.: 2014. Url: <http://gbv.ebib.com/patron/FullRecord.aspx?p=2033202>.
- [HDB-2019] HDB: BIM im Spezialtiefbau - Technisches Positionspapier der Bundesfachabteilung Spezialtiefbau im Hauptverband der Deutschen Bauindustrie e.V. In: (2019).

- [Hol-2007] Hollmann, M.: Ein systemdynamischer Modellierungsansatz zur Untersuchung des technischen und ökologischen Potenzials dezentraler Energieversorgung. Zugl.: Paderborn, Univ., Diss., 2006. Göttingen: Cuvillier, 2007.
- [Hop-2008] Hopp, W. J. et al.: Factory physics. 3. ed., internat. ed. The McGraw-Hill/Irwin series. McGraw-Hill/Irwin, Boston: 2008.
- [Hub-2019] Huber, M.: Das Potenzial digitaler Zwillinge. In: (2019).
- [lfc-2021] lfc-dotnet: lfc-dotnet. <https://code.google.com/archive/p/lfc-dotnet/>, Zugriff am 15.05.2021. 2021.
- [IFC-2021] IFCEngine: IFCEngine. <http://rdf.bg/product-list/ifc-engine/>, Zugriff am 15.05.2021. 2021.
- [lfc-2021] lfcOpenShell: lfcOpenShell. <http://ifcopenshell.org/>, Zugriff am 15.05.2021. 2021.
- [lfc-2021] lfcPlusPlus: lfcPlusPlus. <https://ifcquery.com/>, Zugriff am 15.05.2021. 2021.
- [IFC-2021] IFCToolsProject: IFCToolsProject. [http://www.openifctools.org/Open\\_IFC\\_Tools/Home.html](http://www.openifctools.org/Open_IFC_Tools/Home.html), Zugriff am 15.05.2021. 2021.
- [lsm-2017] Ismail, A. et al.: Integrated and collaborative process-based simulation framework for construction project planning. In: International Journal of Modelling and Simulation (2017).
- [lsm-2014] Ismail, A. et al.: Simulation von Bauausführungsvarianten mit einem prozessbasierten Simulationsbaukasten. In: Informationssysteme im Bauwesen 2. Hrsg. von Scherer, Raimar J.; Schapke, Sven-Eric. Springer Berlin Heidelberg, Berlin, Heidelberg: 2014, S. 107–128.
- [ISO-2020] ISO Internationale Organisation für Normung: Earth-moving machinery and mobile road construction machinery — Worksite data exchange. ISO 15143-3:2016-12. 2016.
- [Jab-2017] Jabri, A. et al.: Agent-based modeling and simulation of earthmoving operations. In: Automation in Construction 81 (2017), S. 210–223.

- [Jac-2017] Jäckle, S.: Neue Trends in den Sozialwissenschaften: Innovative Techniken für qualitative und quantitative Forschung. Hrsg. von Jäckle, Sebastian. 1. Auflage 2017. Springer Fachmedien Wiesbaden, Wiesbaden: 2017.
- [Jac-2011] Jacob, A. et al.: 20111129\_AutoBauLog\_Zweijahresbericht\_website. In: (2011).
- [Ji-2020] Ji, S.: Analyse des aktuellen Stands der ereignisdiskreten Simulation im Bauwesen und Entwicklung einer Simulationsstudie anhand eines konkreten Fallbeispiels. In: (2020).
- [JSD-2021] JSDAI: JSDAI. <https://www.jsdai.net/>, Zugriff am 15.05.2021. 2021.
- [Kap-2016] Kapmeier, F.: Vom systemischen Denken zur Methode System Dynamics. In: (2016).
- [Kar-2015] Kargul A. et al.: Web based field data analysis and data-driven simulation application for construction performance prediction. In: Electronic Journal of Information Technology in Construction (2015).
- [Kes-2020] Kessler, S.: Bauen 4.0 - die Weiterführung der Digitalisierung der Bauprozesse und des Baustellenbetriebes. In: (2020).
- [KIM-2007] KIMURA, O.; TERADA, HIROSUKE: Design and analysis of Pull System, a method of multi-stage production control. In: International Journal of Production Research 19.3 (1981), S. 241–253.
- [Kla-2008] Klaus, P. et al.: Gabler Lexikon Logistik. 4., komplett durchges. und aktualisierte Aufl. Gabler, Wiesbaden: 2008.
- [Klo-2020] Klostermeier, R. et al.: Geschäftsmodelle digitaler Zwillinge: HMD Best Paper Award 2018. 1st ed. 2020. Essentials. Springer Fachmedien Wiesbaden und Imprint: Springer Vieweg, Wiesbaden: 2020.
- [Kud-2000] Kudlich, T.: Optimierung von Materialflusssystemen mit Hilfe der Ablaufsimulation. Dissertation. Technische Universität München, 10.2.
- [Kuh-2006] Kühn, W.: Digitale Fabrik: Fabriksimulation für Produktionsplaner. Hanser, München und Wien: 2006.
- [Lou-2011] Louis, D.: C++: Programmieren mit einfachen Beispielen. easy. Pearson Deutschland und Markt+Technik, München: 2011.

- [Mar-2012] Markl-Hummel, L.: Multikriterielle Entscheidungsunterstützung für kommunale Klimaschutzmaßnahmen. Zugl.: Göttingen, Univ., Diss., 2012. Göttingen: Cuvillier, 2012.
- [MAT-2018] MATEJEVIĆ, B. et al.: The Simulation Model for Predicting the Productivity of the Reinforced Concrete Slabs Concreting Process. In: Tehnicki vjesnik - Technical Gazette 25.6 (2018).
- [Mit-2016] Mittal, A.: Hybrid simulation modeling for regional food systems. Diss. Dunwoody College of Technology, 2016.
- [Nas-2018] Nasirzadeh, F. et al.: A hybrid simulation framework for modelling construction projects using agent-based modelling and system dynamics: an application to model construction workers' safety behavior. In: International Journal of Construction Management 18.2 (2018), S. 132–143.
- [Nos-2015] Noskin, A.: Die Entwicklung der Wasserstoffwirtschaft: Modelle zur Wasserstoffinfrastruktur. Diplomica-Verl., Hamburg: 2015. Url: <http://www.diplomica-verlag.de/>.
- [Pil-2017] Pilling, A. et al.: BIM - Das digitale Miteinander: Planen, Bauen und Betreiben in neuen Dimensionen. 2nd ed. Beuth Innovation. Beuth Verlag, Berlin: 2017. Url: <https://ebookcentral.proquest.com/lib/gbv/detail.action?docID=5107580>.
- [Rao-2016] Raoufi, M.: Overview of Fuzzy Simulation Techniques in Construction Engineering and Management. IEEE: 2016. Url: <http://ieeexplore.ieee.org/servlet/opac?punumber=7819724>.
- [Sad-2015] Sadeghi, N. et al.: Queue performance measures in construction simulation models containing subjective uncertainty. In: Automation in Construction 60 (2015), S. 1–11.
- [Sal-2005] Saleck, T.: Chefsache open source: Kostenvorteile und Unabhängigkeit durch open source ; mit Online-Service zum Buch. 1. Aufl. Edition CIO. Vieweg, Wiesbaden: 2005.
- [Sch-2014] Scherer, R. et al., Hrsg.: Informationssysteme im Bauwesen 1: Modelle, Methoden und Prozesse. Aufl. 2014. VDI-Buch. Springer Berlin Heidelberg, Berlin, Heidelberg: 2014.

- [Sch-2014] Scherer, R. et al.: Informationssysteme im Bauwesen 2: Anwendungen. VDI-Buch. Springer Vieweg, Berlin: 2014.
- [Sch-2020] Schöberl, M.: Current Research and Developments for Civil Engineering in Germany. In: Current Trends in Civil & Structural Engineering 5.1 (2020).
- [Sch-2009] Schönit, M. et al.: Bohren: Verfahren, Geräte, Anwendungen, IT-Lösungen. 1. Auflage. Ernst & Sohn und Liebherr, Berlin und Lenzing: 2019.
- [Sch-2016] Schreyer, M.: BIM - Einstieg kompakt für Bauunternehmer: BIM-Methoden in der Bauausführung. 1. Auflage. Beuth Pocket. Beuth Verlag GmbH, Berlin: 2016.
- [Sch-2019] Schröder, M. et al.: Logistik im Wandel der Zeit – Von der Produktionssteuerung zu vernetzten Supply Chains: Festschrift für Wolfgang Kersten zum 60. Geburtstag. Springer Gabler, Wiesbaden: 2019.
- [Sha-2005] Shaheen, A. et al.: A Framework for Integrating Fuzzy Expert Systems and Discrete Event Simulation. In: Construction Research Congress 2005. Hrsg. von Tommelein, Iris D. American Society of Civil Engineers, Reston, VA: 82005, S. 1–11.
- [Sie-2011] Siepermann, C. et al., Hrsg.: Logistik - gestern, heute, morgen: Festschrift für Richard Vahrenkamp zur Vollendung des 65. Lebensjahres. Gito-Verl., Berlin: 2011.
- [Sin-2016] Singer, D. et al.: Machbarkeitsstudie BIM für Bestandsbrücken. 2016.
- [STE-2021] STEPcode: STEPcode. <https://stepcode.github.io/>, Zugriff am 15.05.2021. 2021.
- [STW-2020] STW Sensor-Technik Wiedemann GmbH: Datenblatt. 2020.
- [Tul-2008] Tulke, J. et al.: Decomposition of BIM objects for scheduling and 4D simulation. In: ECPPM - eWork and eBusiness in Architecture, Engineering and Construction. In: (2008).
- [Tul-2010] Tulke, J. et al.: Kollaborative Terminplanung auf Basis von Bauwerksinformationsmodellen: Zugl.: Weimar, Bauhaus-Univ., Fak. Bauing., Diss., 2010. Bd. 4. Informatik in Architektur und Bauwesen. Verl. der Bauhaus-Univ, Weimar: 2010.

- [Tus-2014] Tuschy, S.: Building Information Modeling (BIM): Eine bisher weitgehend unbekannte Größe in der TGA. In: Bundesindustrieverband Technische Gebäudeausrüstung e. V. (2014).
- [Val-2019] Valluru, P.: An approach to Open-BIM based Construction Project Management. Diss. Technische Universität Dresden, 1.2019.
- [VDI-2018] VDI 2552, Building Information Modeling Begriffe. Entwurf. VDI-Richtlinien. Beuth Verlag GmbH, Berlin: 2018.
- [VDI-2000] VDI 3633. VDI-Richtlinien. Beuth Verlag GmbH, Berlin: 2020.
- [VDM-2021] VDMA e. V.: Machines in Construction - MiC 4.0. <https://bub.vdma.org/viewer/-/v2article/render/31371535>, Zugriff am 09.05.2021. 2021.
- [Wil-2012] Wilms, F.: Wirkungsgefüge: Einsatzmöglichkeiten und Grenzen in der Unternehmensführung. Hrsg. von Wilms, Falko E. P. 1. Aufl. Haupt Verl., Bern: 2012.
- [Wim-2014] Wimmer, J.: Ereignisorientierte Simulation und Optimierung im Erdbau. Dissertation. Technische Universität München, 2014.
- [Wis-2018] Wissel, J. et al.: BIM-Grundlagen für die WSV-Pilotprojekte. 2018.
- [xBI-2021] xBIM: xBIM. <https://docs.xbim.net/>, Zugriff am 15.05.2021. 2021.
- [Xu-2018] Xu, X. et al.: Schedule risk analysis of infrastructure projects: A hybrid dynamic approach. In: Automation in Construction 95 (2018), S. 20–34.

# Abbildungsverzeichnis

---

Abbildung 1-1	Vorgehensweise für die Entwicklung von 4D- und 5D-Modellen in Anlehnung an [Tus-2014, Val-2019]	4
Abbildung 1-2	Simulationsvorgehensmodell [Kud-2000]	5
Abbildung 2-1	Klassifikation von Simulationsmodellen [Gün-2011]	9
Abbildung 2-2	Prozess, Aktivität und Ergebnisse [Fis-1973]	10
Abbildung 2-3	Aufbau eines Prozessbausteins [Gün-2011]	11
Abbildung 2-4	Interaktion heterogener Agenten [Mit-2016]	12
Abbildung 2-5	Lager, Rate und Rückkopplungen in SD [Nas-2018]	14
Abbildung 2-6	Building Information Modelling über den gesamten Lebenszyklus eines Bauwerks [Bor-2015]	21
Abbildung 2-7	Versionsgeschichte des IFC-Modells [Bor-2015]	22
Abbildung 2-8	Ausschnitt aus dem IFC-Datenmodell mit den wichtigsten Entitäten (engl. Entity) der obersten Ebenen der Vererbungshierarchie [Bor-2015]	24
Abbildung 2-9	Die Vererbungshierarchie der Beziehungsklassen des IFC-Datenmodells [Bor-2015]	27
Abbildung 2-10	IfcTask Instantiation Diagram [Bui-2021]	29
Abbildung 2-11	Prozessbeziehungen [Bui-2021]	30
Abbildung 3-1	Aufbau der IFC-Datei mit Notepad++ in der vorliegenden Arbeit	37
Abbildung 3-2	Aufbau der IFC-Datei mit IFC Quick Browser in der vorliegenden Arbeit	37
Abbildung 3-3	4D-Modell in der Software Constructivity in der vorliegenden Arbeit	38
Abbildung 3-4	Darstellung des Gantt-Diagramms in der vorliegenden Arbeit	38
Abbildung 3-5	4D-Modell in Solibri in der vorliegenden Arbeit	39
Abbildung 3-6	4D-Modell in FZK Viewer in der vorliegenden Arbeit	40
Abbildung 3-7	Funktion Abfragen in FZK Viewer	40
Abbildung 3-8	4D-Modell in BIMcollab ZOOM in der vorliegenden Arbeit	41
Abbildung 3-9	4D-Modell in xBIM Xplorer in der vorliegenden Arbeit	41
		91

Abbildung 4-1	Mindestanforderungen an Pfahl mit dem Bauteilcode 140, den Eigenschaften und den Einheiten im Positionspapier [HDB-2019]	45
Abbildung 4-2	IFC-Modell der Brücke aus Pilotprojekt Autobahndirektion Südbayern	45
Abbildung 4-3	IFC-Eigenschaften in der Kategorie Abmessung. Rot umrandet sind die importierten Eigenschaften	47
Abbildung 4-4	IFC-Beispielcode aus der vorliegenden Arbeit	48
Abbildung 4-5	XML-Beispielcode aus der vorliegenden Arbeit	49
Abbildung 4-6	die Gebäudestruktur aus der zugehörigen IFC-Datei am Beispiel vom Bohrpfahl A140.1 im Rahmen des Brückenbauprojekts „WTRO“ in Anlehnung an [Bor-2015]	50
Abbildung 4-7	Hierarchische Beziehung in der IFC-Datei	51
Abbildung 4-8	IfcProject und seine Verweise	51
Abbildung 4-9	IfcSite, IfcBuilding und IfcBuildingstore	51
Abbildung 4-10	IfcPile in IFC-Datei	52
Abbildung 4-11	Beispielhafte Verwendung von Properties (Eigenschaften) aus der vorliegenden Arbeit in Anlehnung an [Bor-2015]	53
Abbildung 4-12	Eigenschaften (IfcProperty), die in IfcPropertySets Allgemein gruppiert sind	53
Abbildung 4-13	Zuordnung von Maß- und Mengenwerten über ein IfcElementQuantity-Objekt	54
Abbildung 4-14	Terminplan innerhalb der IFC-Datei	55
Abbildung 4-15	IfcWorkCalendar in IFC-Datei	55
Abbildung 4-16	TfcTask in IFC-Datei	56
Abbildung 4-17	Darstellung der IFC-Eigenschaften in der Software Plant Simulation	57
Abbildung 5-1	Modulares DES-Modell unter Berücksichtigung der Betonmischer für eine bestimmte Tätigkeit Betonieren	63
Abbildung 5-2	Prozessablauf für die Bohrpfahlherstellung. RL: Radlader, BM: Betonmischer	64
Abbildung 5-3	Szenario mit Push- (links) und Push-Steuerung (rechts)	65

---

Abbildung 5-4	Simulationsstudie im ersten Szenario hinsichtlich der Betonmischeranzahl und des zeitlichen Abstands zwischen den Betonmischern	68
Abbildung 5-5	Visualisierung des ersten Szenarios mit einem (#1) oder zwei Betonmischern (#1 und #2) bei zeitlichem Abstand von 0 min zwischen den Betonmischern	69
Abbildung 5-6	Visualisierung des ersten Szenarios mit sechs Betonmischern (#1 bis #6) bei zeitlichem Abstand von 0 bis 60 min zwischen den Betonmischern	71
Abbildung 5-7	Simulationsstudie im zweiten Szenario hinsichtlich der Fahrzeit der Betonmischer	72
Abbildung 5-8	Simulationsstudie im zweiten Szenario hinsichtlich der Betonmischeranzahl	73
Abbildung 5-9	Simulationsstudie im zweiten Szenario hinsichtlich der Betonmischeranzahl	74
Abbildung 5-10	Visualisierung des zweiten Szenarios mit ein (#1) und zwei Betonmischern (#1 und #2)	75
Abbildung 5-11	Simulationsstudie im zweiten Szenario hinsichtlich der Fahrzeit der Betonmischer	76
Abbildung 5-12	Simulationsstudie in beiden Szenarien hinsichtlich der Fahrzeit der Betonmischer	77
Abbildung 5-13	Simulationsstudie in beiden Szenarien hinsichtlich der Betonmischeranzahl	77
Abbildung 5-14	Experiment mit 10.000 Beobachtungen in beiden Szenarien hinsichtlich der Betonmischer	78



## Tabellenverzeichnis

---

Tabelle 3-1	Übersicht über verschiedene STEP/IFC-Bibliotheken [Bor-2018]	32
Tabelle 3-2	Übersicht über verschiedene Softwarelösungen für die Validierung des BIM-Modells	36
Tabelle 4-1	IFC-Eigenschaften mit unterschiedlichen Quellen in der Kategorie Abmessung	47
Tabelle 4-2	Vergleich der unterschiedlichen Quellen mit den in die Plant Simulation exportierenden IFC-Eigenschaften	58
Tabelle 5-1	Dauer [min] für die acht Teilprozesse am Beispiel von Bohrpfahl A140.1	60
Tabelle 5-2	Taktzeit [min/m] für die acht Teilprozesse am Beispiel von Bohrpfahl A140.1	61
Tabelle 5-3	Experiment mit 10.000 Beobachtungen in beiden Szenarien mit Stau	78



# A Anhang - Darstellung der IFC-Eigenschaften in der Software Constructivity

Property Set	Allgemein
Bauvorhaben(BV)[4]	Westtangete Rosenheim(WTRO)
Projektleiter[1]	
Projektname[1]	
Projektnummer[1]	
Projektadresse[1]	Rosenheim,Bayern,83022 Deutschland
Projektstatus[1]	
Architekt[1]	
Architekt E-Mail[1]	
Architekt PLZ + Stadt[1]	
Architekt Strasse[1]	
Architekt Telefon[1]	
Bauausführung[1]	
Bauherr[1]	
Bauherr PLZ+Stadt[1]	
Bauherr Strasse[1]	
Verwendungszweck (Modell)[1]	Typ: IfcText (String)
Modellierer[1]	Legende
zentrale Projektdatei[1]	In Plant Simulation importieren

Abbildung A-1: IFC-Eigenschaften in der Kategorie Allgemein. Rot umrandet sind die importierten Eigenschaften

Property Set	ID-Daten
SPTB1.0-Code[2,3]	10-50-070-140
Gewerk[1,2,3,4]	Spezialtiefbau
Funktion[1,2,3,4]	Gründung
Bauteilgruppe[1,2,3,4]	Pfahl
Methode[3,4]	Kelly
Nummer[1,2,3,4]	A140.1
BW-Achse[1,2,3,4]	A140
Baugruppenbeschreibung[1]	Piles-Steel Pipe
Baugruppenkennzeichen[1]	NOTDEFINED
Bearbeitungsbereich[1,4]	Bauwerk BW3.2-3.4
Typname[1,2,3,4]	Bohrpfahl Ø 90 cm C30/37
Auftragsnummer[4]	A2112320
gem.Plan Nr.[4]	NOTDEFINED

Abbildung A-2: IFC-Eigenschaften in der Kategorie ID-Daten. Rot umrandet sind die importierten Eigenschaften

Property Set	Daten	
LV-Position[1,2,3,4]	NOTDEFINED	
Klassifikation[1,2,3,4]	Pfahl	
Bauteilkategorie[1,2,3,4]	Bohrpfahl	
Bauteilkategorie[1,2,3,4]	Bohrpfahl-bewehrt	
Gewerk[1,2,3,4]	Spezialtiefbau	
Fachmodell[1,2,3,4]	Bauwerk	
Bauwerk[1,2,3,4]	BW3.2-3.4	
Bereich[1,2,3,4]	A140	
Betriebskilometer[1,4]	Baukm 3+800-4+480	
Material[1,2,3,4]	Beton,Betonstahl	
Materialgüte[1,2,3,4]	C30/37,B500B	
Überbetonhöhe in Leerbohrung[3,4]	1.5 m	
Überbetonvolumen[4]	1.70 m³	
Verfülllänge[3]	0 m	
Bewehrt[1,2,3,4]	True	IfcBoolean (Boolean)
Expositionsklasse[1,4]	XC2-XD2-XF2-WA	
Bewehrungsgewicht[2,3,4]	0.00 kg	IfcMassMeasure (Real)
Bewehrungsgrad[1]	0.000	
Bewehrungsgehalt[2]	0 kg/m³	IfcReal (Real)
Anzahl der Bewehrungskorb[4]	3	IfcInteger (Integer)
Arbeitsschicht[5]	NOTDEFINED	Legende
Anzahl der Rohrschüsse[4]	8	
LOD[2]	0	In Plant Simulation importieren

Abbildung A-3: IFC-Eigenschaften in der Kategorie Daten. Rot umrandet sind die importierten Eigenschaften

Property Set	Maschinendaten	
Verrohrungsanlage[3,4]	True	
Verrohrungsanlage-Typ[5]	NOTDEFINED	
Bohrgeräte-Typ[4]	BG46	
Bohrgeräte-Anzahl[4]	1	
Bohrgeschwindigkeit[3]	0 m/s	
Einheit der Bohrgeschwindigkeit	m/s	
Betonmischer-Typ[5]	NOTDEFINED	
Betonmischer-Max.Ladevolumen[5]	7.00 m³	
Betonmischer-Anzahl[5]	6	
Radlader-Typ[5]	NOTDEFINED	Legende
Radlader-Anzahl[5]	0	
Radlader-Max.Ladevolumen[5]	0.00 m³	In Plant Simulation importieren

Abbildung A-4: IFC-Eigenschaften in der Kategorie Maschinendaten. Rot umrandet sind die importierten Eigenschaften

Property Set	Koordinatensystem	
Definition Koordinatensystem[3,4]	GK-System	
Y-Rechts[4]	4,506,249.795	Legende
X-Hoch[4]	5,301,308.650	
Höhe OK Pfahl[1,3,4]	453.12 mÜNN	In Plant Simulation importieren
Höhe OK Pfahl Einheit	mÜNN	

Typ

Abbildung A-5: IFC-Eigenschaften in der Kategorie Koordinatensystem. Rot umrandet sind die importierten Eigenschaften

Property Set	Terminplan	
Start[3,4]	13.08.2019 10:24:00.00	
Ende[3,4]	22.08.2019 08:24:00.00	
FrühesterAnfang[4]	15.07.2019	Legende
FrühesterEnde[4]	22.07.2019	
SpätesterAnfang[4]	15.07.2019	In Plant Simulation importieren
SpätesterEnde[4]	22.07.2019	
Duration[3,4]	ifcTimeMeasure (Real)	6.75 d

Typ

Abbildung A-6: IFC-Eigenschaften in der Kategorie Terminplan. Rot umrandet sind die importierten Eigenschaften

Property Set	Kosten	
Beton[5]	0 Euro	
Betonstahl[5]	0 Euro	
Bohrgeräte[5]	0 Euro	
Radlader[5]	0 Euro	Legende
Betonmischer[5]	0 Euro	
Werker[5]	0 Euro	In Plant Simulation importieren
Verrohrungsanlage[5]	0 Euro	

Typ

Abbildung A-7: IFC-Eigenschaften in der Kategorie Kosten. Rot umrandet sind die importierten Eigenschaften



## B Anhang - IFC-Datei

```
#186=IFCWORKPLAN('0RH$VOulz05xXgt8jwLtDY',#2,'Work Plan',,$,$,$,'2018-11-29T00:00:00',,$,$,$,'2018-11-29T00:00:00',,$,$);
#187=IFCWORKSCHEDULE('1V_f86idX0xxwc$TpuHbK',#2,$,$,$,'2019-08-13T00:00:00',,$,$,$,'2019-08-13T00:00:00',,$,$,PLANNED.);
#188=IFCTASK('3XbmyiAmBkAH1KFSr7JLW',#2,'Default Storey A140-GrundungspfXIE4hle',,$,$,$,$,$,F.,,$,#189,.CONSTRUCTION.);
    #189=IFCTASKTIME($,PREDICTED.,$,WORKTIME.,'6D18H','2019-08-13T10:24:00','2019-08-22T08:24:00','2019-07-15T00:00:00','2019-07-22T00:00:00','2019-07-15T00:00:00','2019-07-22T00:00:00',,$,$,$,$,$,$,$);
#191=IFCWORKCALENDAR('0AvbhUS9jDgeTa_ZLbx3q9',#2,$,$,$,($192),$,NOTDEFINED.);
    #192=IFCWORKTIME('Standard',,$,#193,$,$);
        #193=IFCRECURRENCEPATTERN(.WEEKLY.,$(1,2,3,4,5),,$,$,$,($194));
        #194=IFCTIMEPERIOD('08:00:00','17:00:00');
#195=IFCTASK('11nCBzO$f1TxO1Ao98aeZF',#2,'Pile(Standard) A140-GrXIFCndungspfXIE4hle',,$,$,$,$,$,F.,,$,#196,.CONSTRUCTION.);
#197=IFCTASK('0Ud0HfTBj6sPLTqIbF7ZYW',#2,'A140.1 GrXIFCndungspfXIE4hle',,$,$,$,$,'Kelly',F.,,$,#198,.CONSTRUCTION.);
    #198=IFCTASKTIME($,PREDICTED.,$,WORKTIME.,'P0Y0M0DT0H0M0S','2019-08-14T08:00:00','2019-08-14T17:00:00',,$,$,$,$,'P0Y0M0DT0H0M0S','P0Y0M0DT0H0M0S',.T.,,$,$,$,$,$);
#190=IFCRELASSIGNSTOCONTROL('2H34gpgBTEyOjpJM2qLqCy',#2,$,$,($188),,$,#187);
#199=IFCRELASSIGNSTOCONTROL('0Bin28Yzj0WAx$gJFCfcl',#2,$,$,($188),,$,#191);
#200=IFCRELDECLARES('3qzwNyPM54N9hT4W0tgvjS',#2,'PROCESS',,$,#1,($188));
#201=IFCRELDECLARES('05VBWV$dbB59yGt87qC6yZ',#2,'CONTROL',,$,#1,($186));
#202=IFCRELASSIGNSTOPRODUCT('3XuiTHFJ5AD9qg78B9qVlz',#2,$,$,($188),,$,#44);
#203=IFCRELASSIGNSTOPRODUCT('2F2XDP1VP1aBKQbuH1hsKK',#2,$,$,($197),,$,#45);
#204=IFCRELNESTS('2T6R84Vrb3ff6olSQxOdpG',#2,$,$,#186,($191,#187));
#205=IFCRELNESTS('1ucbNYHcj0HRSIOPjsQ7o5',#2,$,$,#188,($195));
#206=IFCRELNESTS('3llmosg0T29vMxLRYOZK_z',#2,$,$,#195,($197));
```

Abbildung B-1: Abbildung des Terminplans in IFC-Datei



## C Anhang - Entitäten aus dem IFC-Datenmodell im Programm MVS

<b>XbimEditorCredentials</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	ApplicationDevelopersName	xBimTeam
	ApplicationFullName	Pile Application
	ApplicationIdentifier	Pile.exe
	ApplicationVersion	1.0
	EditorsFamilyName	Team
	EditorsGivenName	xBIM
	EditorsOrganisationName	fml
<b>IfcProject</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Pile
	LongName	B15, Westtangente Rosenheim
	Phase	Projektstatus
<b>IfcSite</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Default Site
	CompositionType	ELEMENT
	RefLatitude	0,0,0,0
	RefLongitude	0,0,0,0
	RefElevation	0
	AddressLines	<Projektaddress>
	InternalLocation	<AddressForInternalMailDelivery>
	Town	Rosenheim
	Region	Bayern
	PostalCode	83022
	Country	Deutschland
<b>IfcBuilding</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Default Building
	CompositionType	ELEMENT
<b>IfcStorey</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Default Storey A140-Grundungspfähle
	Elevation	0.0
	CompositionType	ELEMENT

Abbildung C-1: Allgemeine Entitäten - TeilA

<b>IfcPile</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Bohrpfahl: Durchmesser 1.2cm
	PredefinedType	BORED
	ConstructionType	CAST IN PLACE
<b>Representation</b>	Body	Extruded Solid
<b>Material</b>	Materialschicht	Beton C30/37
	Materialschicht	Betonstahl B500B
<b>IfcUnit</b>	<b>UnitType</b>	<b>UnitName</b>
<b>SIUnit</b>	LENGTHUNIT	METRE
	AREAUNIT	SQUARE METRE
	VOLUMEUNIT	CUBIC METRE
	SOLIDANGLEUNIT	STERADIAN
	PLANEANGLEUNIT	RADIAN
	MASSUNIT	KILO GRAM
	TIMEUNIT	SECOND
	THERMODYNAMICTEMPERATUREUNIT	DEGREE CELSIUS
	LUMINOUSINTENSITYUNIT	LUMEN
<b>ConversionBasedUnit</b>	PLANEANGLEUNIT	°
<b>ContextDependentUnit</b>	LENGTHUNIT	müNN
<b>DerivedUnit</b>	MASSDENSITYUNIT	kg/m3
	LINEARVELOCITYUNIT	m/s
<b>MonetaryUnit</b>	MONETARYUNIT	Euro

Abbildung C-2: Allgemeine Entitäten - TeilB

<b>IfcWorkPlan</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Work Plan
	CreationDate	2018-11-29T00:00:00
	StartTime	2018-11-29T00:00:00
<b>IfcWorkSchedule</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	PredefinedType	PLANNED
	CreationDate	2019-08-13T00:00:00
	StartTime	2019-08-13T00:00:00
<b>IfcWorkCalendar</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	PredefinedType	NOTDEFINED
	Name	Standard
<b>IfcRecurrencePattern</b>	RecurrenceType	WEEKLY
	StartTime	8:00:00
	EndTime	17:00:00
	WeekdayComponent	Montag(1) bis Freitag(5)

Abbildung C-3: Entitäten hinsichtlich des Terminplans - TeilA

<b>IfcTask</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Default Storey A140-Grundungspfähle
	PredefinedType	CONSTRUCTION
<b>IfcTask Time</b>	DataOrigin	PREDICTED
	DurationType	WORKTIME
	ScheduleStart	2019-08-13T10:24:00
	ScheduleFinish	2019-08-22T08:24:00
	ScheduleDuration	POYOMODTOHOMOS
	EarlyFinish	2019-07-22T00:00:00
	EarlyStart	2019-07-15T00:00:00
	LateFinish	2019-07-22T00:00:00
	LateStart	2019-07-15T00:00:00
<b>IfcTask</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	Pile(Standard) A140-Grundungspfähle
	PredefinedType	CONSTRUCTION
<b>IfcTask Time</b>	DataOrigin	PREDICTED
	DurationType	WORKTIME
	ScheduleStart	2019-08-13T10:24:00
	ScheduleFinish	2019-08-22T08:24:00
	ScheduleDuration	6D18H
	EarlyFinish	2019-07-22T00:00:00
	EarlyStart	2019-07-15T00:00:00
	LateFinish	2019-07-22T00:00:00
	LateStart	2019-07-15T00:00:00
<b>IfcTask</b>	<b>IFC Parameter</b>	<b>Wert</b>
<b>Allgemein</b>	Name	A140.1 Grundungspfähle
	WorkMethod	Kelly
	PredefinedType	CONSTRUCTION
<b>IfcTask Time</b>	DataOrigin	PREDICTED
	DurationType	WORKTIME
	ScheduleStart	2019-08-13T10:24:00
	ScheduleFinish	2019-08-23T08:24:00
	ScheduleDuration	POYOMODTOHOMOS
	FreeFloat	POYOMODTOHOMOS
	TotalFloat	POYOMODTOHOMOS
	IsCritical	true
<b>Assignments</b>	Pile:BORED	Bohrpfahl: Durchmesser 1.2cm

Abbildung C-4: Entitäten hinsichtlich des Terminplans - TeilB



# D Anhang - Visualisierung in der Software Plant Simulation

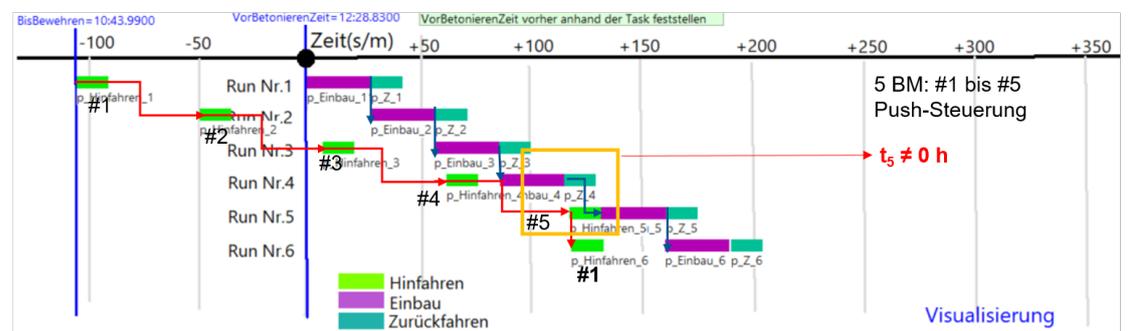
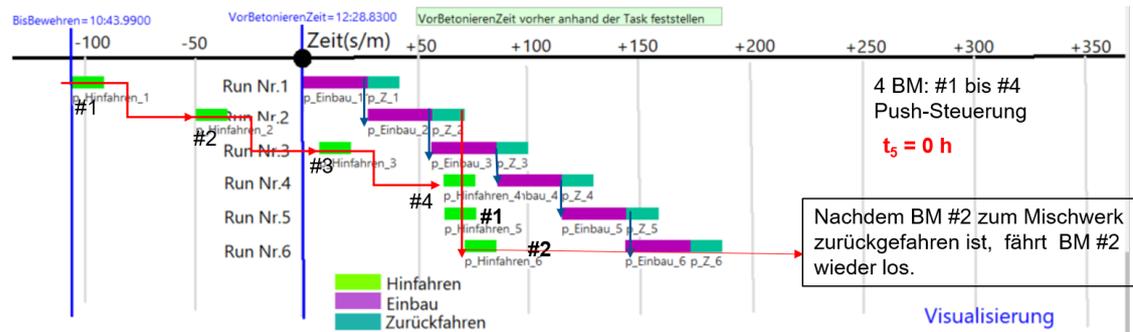


Abbildung D-1: Visualisierung des ersten Szenarios mit vier (#1 bis #4) und fünf Betonmischern (#1 bis #5) bei zeitlichem Abstand von 30 min zwischen den Betonmischern

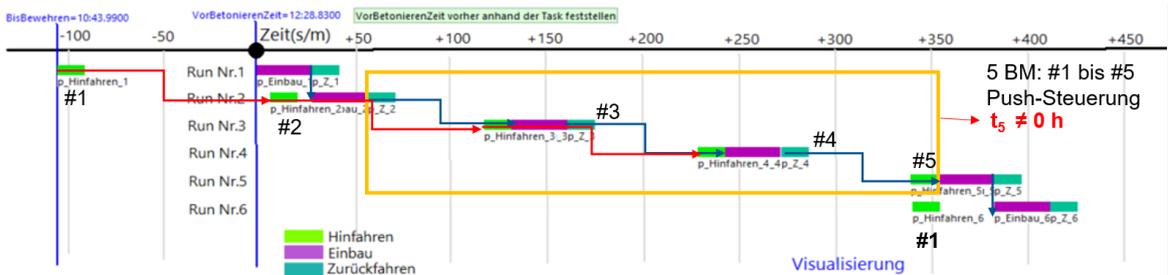


Abbildung D-2: Visualisierung des ersten Szenarios mit fünf Betonmischern (#1 bis #5) bei zeitlichem Abstand von 60 min zwischen den Betonmischern

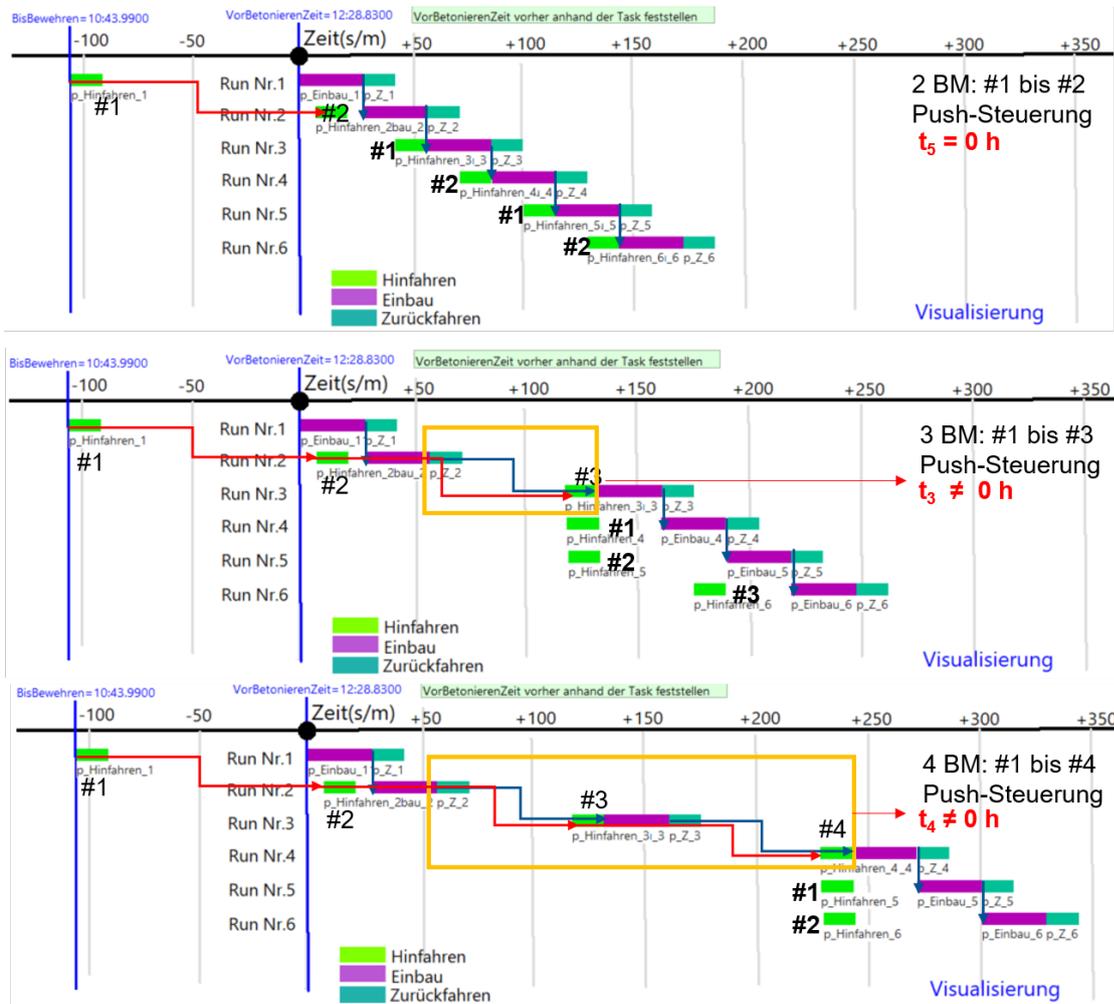


Abbildung D-3: Visualisierung des ersten Szenarios mit zwei (#1 und #2) bis vier Betonmischern (#1 bis #4) bei zeitlichem Abstand von 60 min zwischen den Betonmischern

## Erklärung

---

Ich versichere hiermit, dass ich die von mir eingereichte Abschlussarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Garching, 01.06.2021. Shuoka

---

Ort, Datum, Unterschrift