

Bauablaufsimaulation und -animation für die Planung von Brückenbauvorhaben

Gergó Dori, André Borrmann
Lehrstuhl für Computation in Engineering, Technische Universität München
{dori, borrmann}@bv.tum.de

Kurzfassung: Im Paper wird ein Projekt vorgestellt, dessen Ziel es ist, für 3D-Brückenmodelle eine automatische Bauablaufsimaulation mit Zeitoptimierung zu realisieren. Das Projekt wird im Rahmen des Forschungsverbunds ForBAU¹ durchgeführt und gliedert sich in drei Teile: Im ersten Teil wird ein 3D-Modell um Bauprozesswissen erweitert. In diesem Teil wird das prototypisch entwickelte Programm Preparator vorgestellt. Der zweite Teil besteht in der Bauablaufsimaulation. Auf Grundlage der vom Preparator gelieferten Daten wird hier eine möglichst optimale Reihenfolge der Bauablaufsimaulation ermittelt. Dafür wird eine Ereignis-diskrete Simulation benutzt, die um ein System zur Constraintprüfung erweitert wurde. Es werden Präzedenzbeziehungen eingeführt, die verschiedene zeitliche Verbindungen zwischen mehreren Bauablaufsimaulationsschritten repräsentieren. Im letzten Teil des Projekts wird aus den Ergebnissen der Simulation automatisiert eine Animation abgeleitet. Dazu werden parametrisch vordefinierte Ablaufsimaulationstücke benutzt. Mit Hilfe der Animation soll die berechnete Ablauffolge visualisiert werden, um Planer schon in frühen Phasen der Planung auf Schwierigkeiten und mögliche Kollisionen beim Bauablauf aufmerksam zu machen.

1 Einleitung

Die Planung von Bauabläufen ist eine komplexe und zeitaufwändige Aufgabe. Um diese Arbeit zu erleichtern, wird im Rahmen des ForBAU-Projekts ein Forschungsprojekt mit der Zielsetzung durchgeführt, eine halbautomatisierte Bauablaufsimaulation und Ablaufsimaulation für Brückenbauvorhaben zu entwickeln. Das Projekt gliedert sich in drei Teile:

- I. Erstellen eines intelligenten 3D-Modells, dessen Komponenten mit Informationen zu Konstruktionsmethoden und Bauablauffolgen verknüpft werden
- II. Ablaufsimaulation und Zeitoptimierung: Mit Hilfe einer Diskrete-Ereignis-Simulation wird die Gesamtprojektdauer aus Teilprozesszeiten bestimmt. Auf Basis eines Monte-Carlo-Ansatzes werden Reihenfolgen variiert und eine möglichst optimale Lösung bestimmt
- III. Animation: Aus der bestimmten Ablauffolge wird eine Animation erzeugt und als 4D-Modell gespeichert

In diesem Paper werden die genannte Projektteile mit Schwerpunkt auf dem Erstellen des intelligenten 3D-Modells und der Bauablaufsimaulation vorgestellt. Im ersten Teil wird das Constraint-basierte Konzept für die Bauablaufsimaulation vorgestellt und die Idee einer hierarchischen Lösungsmethode eingeführt, welche eine kürzere Berech-

¹ Der Forschungsverbund ForBAU wird von der Bayerischen Forschungsfoundation gefördert.

nungszeit ermöglicht. Im zweiten Teil wird das prototypisch entwickelte Programm *Preparator* mit seinen Aufgaben und Lösungskonzepten detailliert vorgestellt. Im dritten Teil werden Ziele und Schwierigkeiten einer automatisierten Ablaufanimation diskutiert. Am Ende des Papers werden die Erweiterungsideen und Ziele zusammengefasst.

2 Bauablaufsimulation

Die wesentlichen Eingabedaten einer Animation sind die Anfangs- und Endzeitpunkte der einzelnen Ablaufschritte. Um diese zu ermitteln, muss zunächst eine Ablaufsimulation durchgeführt werden. Mit einer Constraint-basierten Simulation können zuverlässige Ablaufergebnisse unter Beachtung der Verfügbarkeit der verschiedenen Ressourcen generiert werden (Beißert et al. 2007, Beißert et al. 2008, Wu et al. 2009). Der wesentliche Vorteil des Constraint-Ansatzes ist, dass die Reihenfolge der einzelnen Arbeitsschritte nicht fest vorgegeben, sondern dynamisch auf der Basis von Zufallsselektionen bestimmt wird. Im hier verfolgten Konzept werden die Eingabedaten für die Simulation mit Hilfe des Preparators (Abbildung 1) erzeugt. Dabei werden neben atomaren Aktivitäten deren auch Reihenfolgebeziehungen sowie die zur Ausführung notwendigen Mitarbeiter, Maschinen und Materialien definiert.

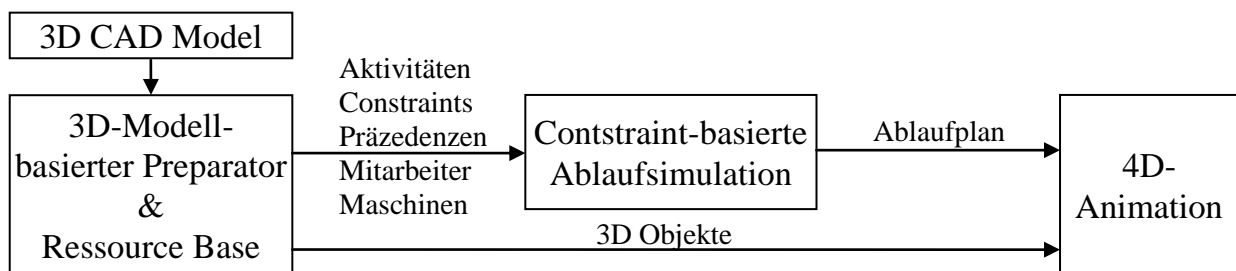


Abbildung 1: Aufbau des Projekts

2.1 Level of detail - Ansatz

Um die Komplexität der Aufbereitung der Simulationsparameter zu reduzieren, wurde ein hierarchischer Ansatz (Level of Detail) umgesetzt (Wu et al. 2010). Der Bauablauf wird top-down von groben Prozessschritten hin zu atomaren Aktivitäten durch Auswahl von Bauverfahren aufgebaut und in einer Prozesshierarchie geordnet. Die Hierarchie besteht aus vier Levels. Auf dem höchsten Level steht das ganze *Bauwerk*: die Brücke. In jedem weiteren Level werden die Schritte weiter verfeinert (Abbildung 2). Die Brücke besteht aus *Bauelementen*, wie z.B. Widerlagern und Stützen. Diese Elemente setzen sich aus kleineren *Unterelementen* zusammen, wie z.B. bei einer Stütze das Unterfundament, das Fundament, die Stütze usw. Nach der Auswahl des eingesetzten *Bauverfahrens* werden die Elemente im nächsten Level automatisch generiert, sind jedoch nachträglich manuell veränderbar. Auf dem vierten Level befinden sich vordefinierte *Prozessmuster*, die aus verschiedenen atomaren *Aktivitäten* bestehen. Das hierarchische Produktmodell (auf der rechten Seite in Abbildung 2) ist das Ergebnis des Bauablaufs, quasi die fertig gestellte Brücke.

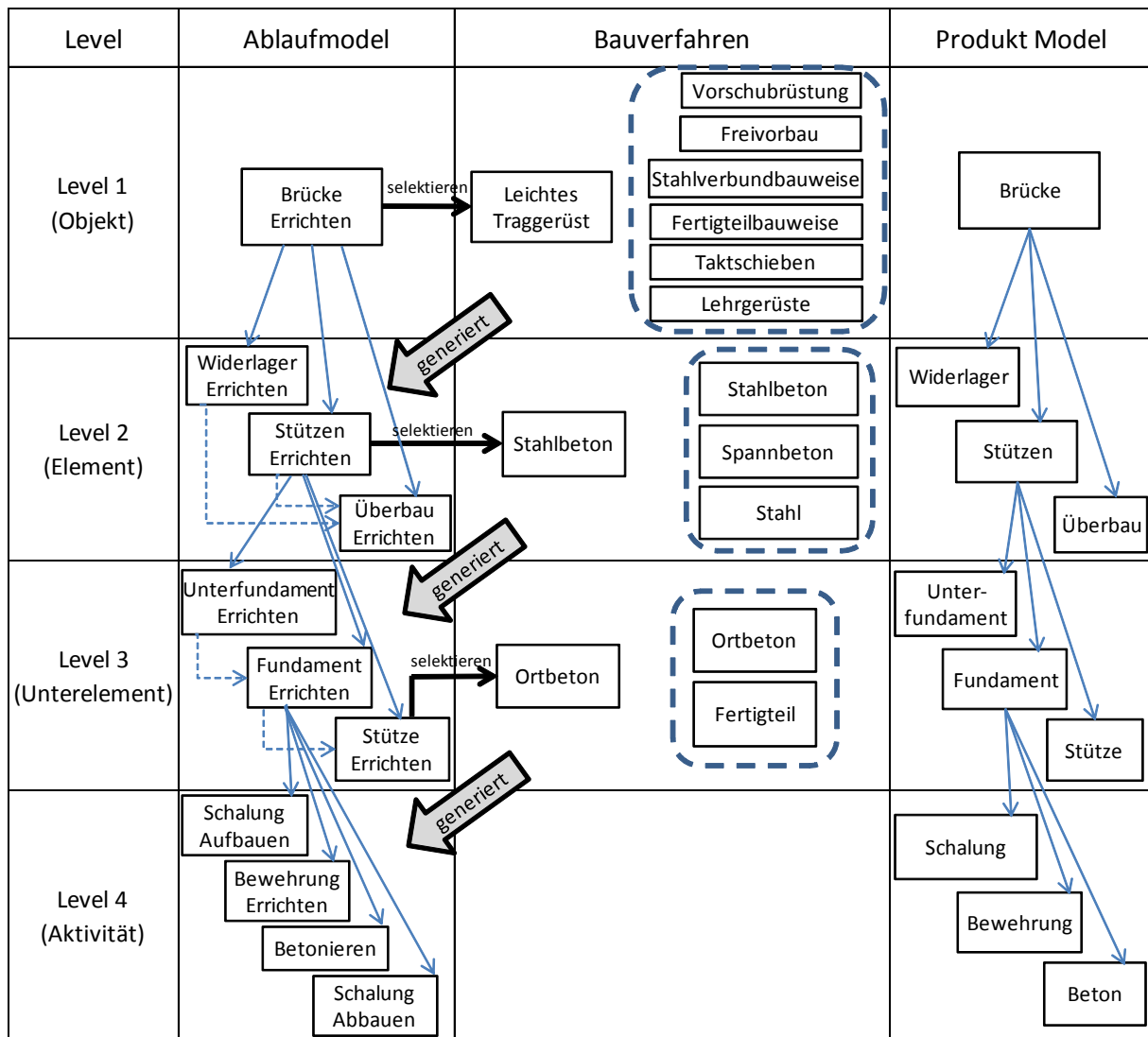


Abbildung 2: „Level of Detail“ Hierarchie

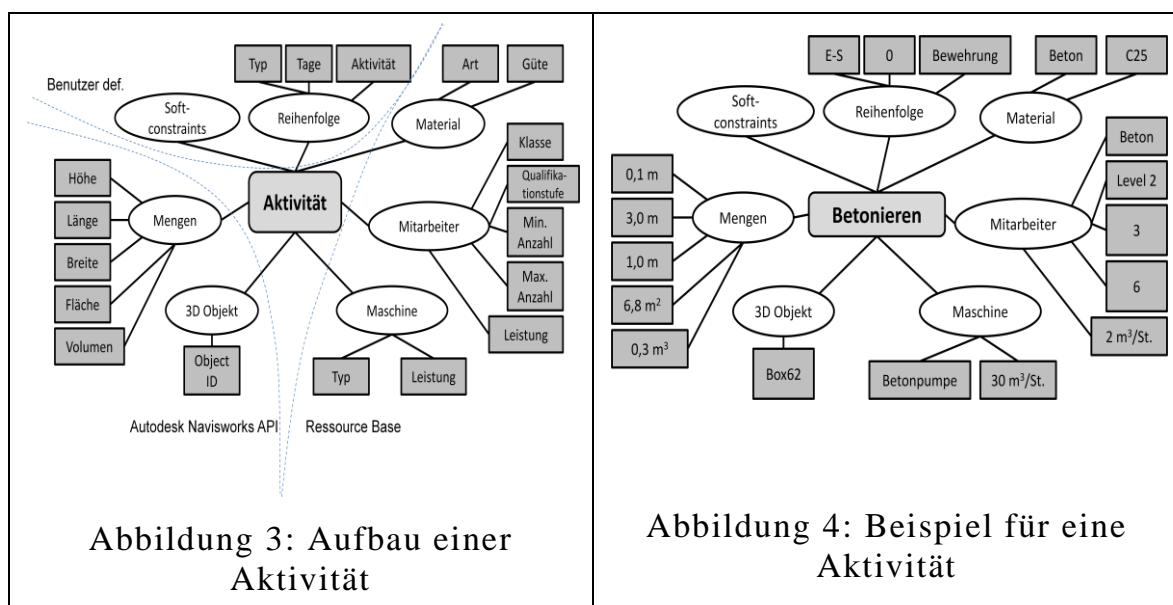
2.1.1 Prozesspatterns und Aktivitäten

Der Bau eines Unterelements wird durch Anwendung von *Prozesspatterns* in einzelne Aktivitätsschritte aufgelöst. Diese Patterns ergeben aus dem angewendeten Bauverfahren. Wenn beispielsweise ein Unterelement aus Ortbeton hergestellt wird, sind die sich ergebenden Schritte immer die folgenden: 1. Schalung aufbauen, 2. Bewehrung einbauen, 3. Betonieren und am Ende (4.) die Schalung entfernen. Diese Schritte sind die *Aktivitäten*. Die Aktivitäten sind mit allen Informationen verbunden, die zu ihrer Ausführung nötig sind (Abbildung 3). Die Attribute einer Aktivität können wie folgt kategorisiert werden:

1. Nutzerdefinierte Attribute
2. durch Autodesk Navisworks API eingelesene Attribute
3. Attribute in Verbindung mit der Ressource Base

In den nutzerdefinierten Attributen werden die *Präzedenzen* und die *Soft Constraints* abgebildet (Beißert et al. 2008). Unter Präzedenzen verstehen wir eine zeitliche

Beziehung zwischen mehreren Aktivitäten. Bei der Zuweisungsphase der Konstruktionsmethoden werden die Verknüpfungen interaktiv hergestellt. Dabei wird Autodesk Navisworks für die Visualisierung des 3D-Modells benutzt. Die einzelnen Bauteile werden im 3D-Modell ausgewählt und ihren Konstruktionsmethoden zugewiesen. Mit Hilfe der Autodesk Navisworks API Funktionen werden die geometrischen Eigenschaften des ausgewählten 3D-Objekts eingelesen. Sie werden nur einmal in einem Unterelement gespeichert; alle assoziierten Aktivitäten lesen die benötigten Mengen dort aus. Diese Daten (Mengen) sind eine wichtige Eingangsgröße für die Berechnung der Dauer der Aktivitäten im Rahmen der Ablaufsimulation. Die benötigten Ressourcen (Mitarbeiter, Maschinen und Material) werden ebenfalls mit den Aktivitäten verknüpft. Diese Angaben werden beim Durchführen der Ablaufsimulation mit den in der Ressource Base (Abschnitt 3.2) definierten Daten verglichen und die beste Möglichkeiten für den Arbeitsschritt ausgewählt.



2.2 Constraint-basierte Ablaufsimulation

Die Ereignis-diskrete Ablaufsimulation wurde grundsätzlich für die stationäre Industrie entwickelt. Mit der Erweiterung um den Constraint-basierten Ansatz ist sie aber auch für das Bauwesen gut verwendbar (Beißert et al. 2007). Dabei wird der Bauablauf als ein *Constraint Satisfaction Problem* definiert: alle Voraussetzungen zum Durchführen einer Aktivität werden als Constraints festgehalten, darunter die benötigten Mitarbeiter, Maschinen, Material, Platz und welche Aktivitäten vorher beendet sein müssen (Präzedenzen). Es gibt zwei verschiedene Typen von Constraints: *Hard Constraints* und *Soft Constraints*.

Hard Constraints müssen zwingend erfüllt sein, um eine Aktivität starten zu können. Soft Constraints hingegen dienen der Umsetzung einer Strategie bei der Wahl von Ausführungsreihenfolgen. Mit ihrer Hilfe werden *Präferenzen* zwischen ausführbaren Aktivitäten abgebildet. Die Aktivität mit der höchsten Präferenz wird ausgeführt und die verwendeten Ressourcen gesperrt (Beißert et al. 2008).

2.2.1 Einsatz der Prozesshierarchie zur Prüfung von Hard Constraints

Wenn die Liste der ausführbaren Aktivitäten aktualisiert wird, wird für alle definierten Aktivitäten bei jedem Schritt geprüft, ob alle Hard Constraints erfüllt sind (bei einer große Brücken können das mehr als 2-3000 sein). Um diese Prozedur zu beschleunigen, kann der in Abbildung 2 eingeführte hierarchische Aufbau auch für die Prüfung der Constraints benutzt werden. Das bedeutet, dass die Constraints nicht nur mit Aktivitäten verbunden sind, sie werden stattdessen auch auf höheren Levels eingesetzt. Beispielsweise kann die Arbeit an einem Überbauteil nur dann begonnen werden, wenn das Widerlager auf einer Seite und die Stützen auf der anderen Seite bereits fertiggestellt sind. Deswegen werden die Constraints der betreffenden Unterelemente und die Aktivitäten des Überbaus nicht geprüft, solange die Stützen und das Widerlager nicht fertiggestellt sind. Wir prüfen die Constraints zunächst auf dem höchsten Level und nur wenn alle erfüllt sind, wird das nächste Level untersucht. Mit dieser Methode kann die Performance stark erhöht werden. Die Implementierung wird durch Verwendung der Präzedenzen möglich.

2.2.2 Präzedenzen

Die Präzedenzen sind eine spezielle Art von Hard Constraints, die verschiedene Zeitbeziehungen zwischen Prozessschritten des gleichen Levels repräsentieren. Aus diesem Grund können Präzedenzen nur zwischen Elementen, zwischen Unterelementen oder zwischen Aktivitäten definiert werden. Wir unterscheiden *Start-Start*-, *Ende-Start*-, *Start-Ende*- und *Ende-Ende*-Verbindungen. Die Verbindung kann darüber hinaus mit einer Zeitdauer versehen werden.

Zum Beispiel kann mit dem Betonieren direkt nach dem Fertigstellen der Bewehrung (und der Schalung) begonnen werden (Abbildung 4 – End-Start-Verbindung, 0 Tage nach Bewehren). Eine Start-Start 3 Verbindung bedeutet entsprechend, dass der nachfolgende Prozess erst nach drei Tagen Pause begonnen werden kann.

Es können auch mehrere Präzedenzen gleichzeitig definiert werden, was dann mehrere Constraints für den Prozess bedeutet. Zur abermaligen Reduzierung der Rechenzeit kann auf dem Level Unterelement festgelegt werden, dass nur die erste Aktivität ausgeführt werden darf. Die Constraints der weiteren Aktivitäten werden dabei nur dann geprüft, wenn sie mit der vorherigen in Verbindung stehen.

3 Einsatz des Preparators zum Erzeugen eines intelligenten 3D Modells

In unserem Konzept werden die Eingangsdaten für die Bauablaufsimulation auf der Basis eines 3D-Modells der Brücke erzeugt. Wenn dieses Modell ausreichend detailliert ist, können einzelne Objekte (Element/Unterelement) identifiziert und mit entsprechenden Bauablaufschritten verknüpft werden. Das ist die Hauptaufgabe des Preparators.

3.1 Aufgaben

Der Preparator liefert alle Eingabedaten für die Ablaufsimulation. Deswegen müssen alle Eigenschaften, Attribute und Ressourcen, die bei einem Bauablauf wichtig sind, zusammengesammelt und in einer Datei gespeichert werden. Das Programm besteht aus zwei Teilen. Der erste Teil ist der eigentliche Preparator (Abbildung 5), der zweite Teil ist die Ressource Base. Zunächst muss in den Preparator ein 3D-Brückenmodell eingelesen werden. Das Modellfenster zur Anzeige des Modells befindet sich in der Mitte unten. Die zugehörige Liste der Objekte (linke Seite), aus denen das Modell besteht, wird automatisch generiert. Oben links befinden sich Steuerelemente zum Aufbau der in Abschnitt 2.1 vorgestellten Hierarchie. Hier können die *vordefinierten* Elemente, Bauverfahren und Unterelemente zugeordnet werden. Oben in der Mitte werden die Aktivitäten mit den zugehörigen Präzedenzen dargestellt. Unter Verwendung der Steuerelemente unterhalb der Listen können neue Aktivitäten und Präzedenzen hinzugefügt oder gelöscht werden. Auf der rechten Seite ist das Ergebnis der Zuweisung dargestellt: die hierarchische Liste der Prozessschritte zum Bau der Brücke. Diese Liste muss per Unterelement aufgebaut werden. Zunächst wird ein Element (oben links) mit dem zugehörigen Bauverfahren ausgewählt, dann ein Unterelement. Dieses Unterelement wird mit einem 3D-Objekt verknüpft. Es werden die Präzedenzbeziehungen zwischen den Aktivitäten überprüft. Im Fall, dass die vordefinierte Präzedenzen bzw. benötigte Ressourcen nicht passend sind, können diese undefiniert werden.

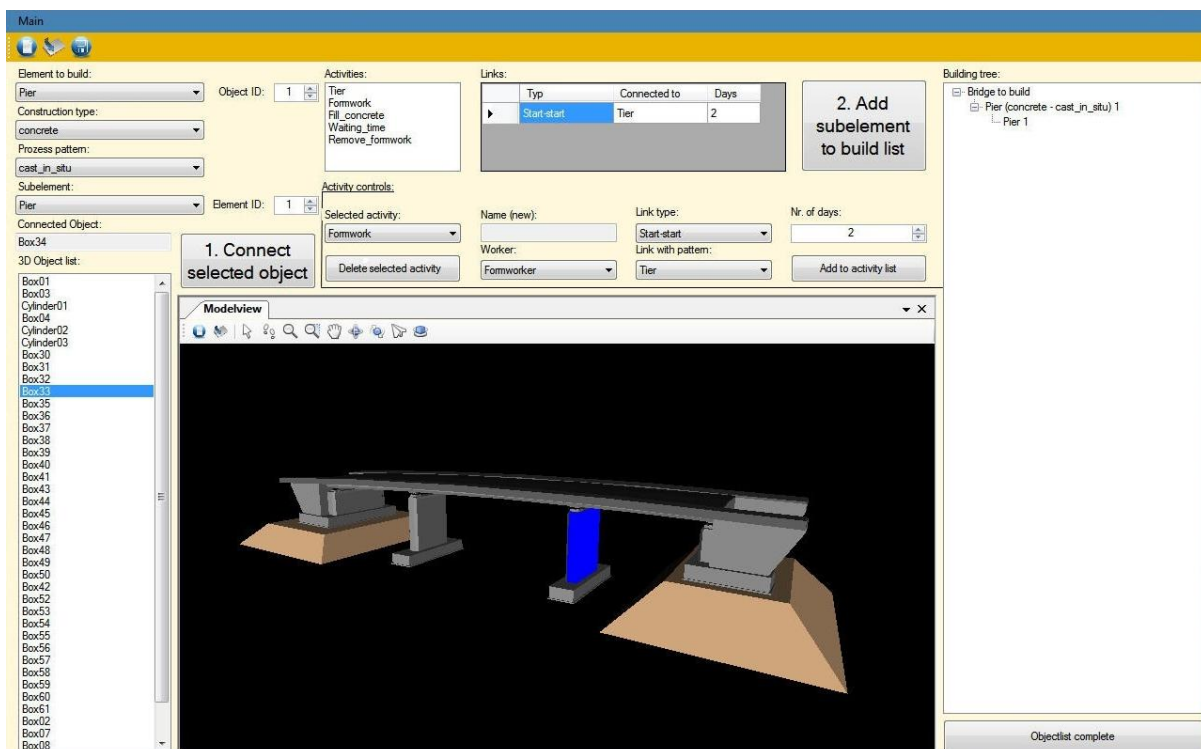


Abbildung 5: Aufbau des Preparators

Sind alle Angaben richtig, wird durch Betätigen des Knopfs „Add subelement to process list“ das entsprechende Unterelement der Brücke zur Hierarchie auf der rechten Seite hinzugefügt. Jetzt kann ein neues Unterelement ausgewählt und zur

Hierarchie hinzugefügt werden usw. Nach erfolgter Verknüpfung werden die Objekte von der Objektliste gelöscht. Dieser Vorgang wird wiederholt, bis alle Objekte mit Informationen verknüpft sind und der Bauprozess vollständig aufgebaut ist. Im Folgenden müssen noch die verfügbaren Ressourcen definiert werden.

3.2 Ressource Base

Das zweite Hauptfenster des Preparators ist die Ressource Base. Hier können die verfügbaren Mitarbeiter, Maschinen und Materialien mit ihren Eigenschaften in tabellarischer Form definiert werden. Die wichtigsten Attribute eines Mitarbeiters sind sein Arbeitsbereich, seine Ausbildungsstufe und seine Arbeitsleistung. Hinzu kommt eine ID, mit deren Hilfe verfolgbar ist, für welche Arbeitsschritte er eingesetzt wurde. Daneben ist noch definierbar, wann er verfügbar ist. Ist ein Mitarbeiter beispielsweise im Urlaub, kann er während dieser Zeit nicht auf der Baustelle eingesetzt werden. Das gleiche gilt für eine Baufirma, die verschiedene Unterauftragnehmer beschäftigt. Deren Mitarbeiter sind i.d.R. ebenfalls nur für eine begrenzte Zeit verfügbar, weil sie bei weiteren Bauprojekten arbeiten. Für die Maschinen werden ähnliche Attribute gespeichert: Typ, Leistung und Verfügbarkeit. Materialien werden durch einen Namen und eine Qualität beschrieben. Weitere Attribute sind die Menge des verfügbaren Materials sowie die Menge der täglichen Zulieferung bzw. Produktion. Alle diese Daten werden mit der aufgebauten Prozesshierarchie in einer Datei gespeichert.

4 Animation

Ergebnis der Ablaufsimulation ist nicht nur die Reihenfolge der Bauschritte, sondern auch die Anfangs- und Endzeitpunkte der einzelnen Aktivitäten. Wenn diese Zeiten mit den 3D-Objekten verknüpft werden, wird es möglich, eine Ablaufanimation automatisch zu generieren. Daher werden diese Zeitpunkte in der Datei zusammen mit der Brückenhierarchie für jeden Aufbauschritt gespeichert und später in die Animation eingelesen. Für die einzelnen Schritte können verschiedene Aufbauanimationen parametrisch vordefiniert und später mit der Anfangs- und Endzeitpunkte verbunden werden.

4.1 Schwierigkeiten

Eine wesentliche Schwierigkeit beim automatisierten Erzeugen einer Animation sind Bauteile und Hilfskonstruktionselemente, die im 3D-CAD-Modell nicht enthalten sind. Dabei handelt es sich u.a. um Schalungselemente, Rüstzeug, Erdbaukörper und i.d.R. auch um die Bewehrung. Mangelnde Modellierung der Bewehrung führt nicht nur zu Problemen beim Erzeugen der Animation, sondern bereits bei der Mengenermittlung im Preparator. Kenntnis über die Menge des einzubauenden Stahls ist notwendig, um die zugehörige Konstruktionszeit zu berechnen. Wenn diese Information nicht verfügbar ist, muss sie manuell im Preparator eingegeben werden. Hilfskonstruktionen (z.B. Schalung), die nicht modelliert sind, müssen erst mit dem zugehörigen Unterelement (z.B. Stütze 12) und einem Ablaufschritt (z.B. Schalen) parametrisch verknüpft werden, dann als neues 3D-Objekt definiert und am Ende animiert werden. Die eingesetzten Parameter sind die Größe des Objekts und die Aufbauzeit.

4.2 Ziele

Ziel der Entwicklung einer automatisierten Animation ist nicht nur, den Aufbau einer Brücke zu visualisieren, sondern die Planer bereits in der Planungsphase auf potentielle Probleme und Schwierigkeiten aufmerksam zu machen. Auftretende Schwierigkeiten können beispielsweise sein: zu wenig Platz für ein Element, schwere Erreichbarkeit oder mögliche Elementkollisionen. Um diese visualisieren zu können, sind zusätzliche Informationen zur Baustelleneinrichtung, insbesondere zur Position von Maschinen, Kränen etc. notwendig.

5 Zusammenfassung

Im Paper wurden erste Schritte im Rahmen eines Projektes vorgestellt, dessen Ziel es ist, Ablaufanimationen für Brückenbauvorhaben automatisch zu generieren. Um dieses Ziel zu erreichen, wird zunächst ein detailliertes 3D-Modell der Brücke benötigt. Im nächsten Schritt müssen die 3D-Objekte mit Informationen, Materialien und Bauverfahren verknüpft werden. Hierzu wurde das prototypische Programm Preparator entwickelt. Dieses Programm benutzt vordefinierte Bauverfahren auf verschiedenen Detaillierungsebenen, mit deren Hilfe eine hierarchische Struktur von Bauprozessschritten erstellt werden kann. Im Preparator können zusätzlich die Mengen der Materialien sowie die Eigenschaften der verfügbaren Mitarbeiter und Maschinen definiert werden. Mit Hilfe dieser Daten kann die Ablaufsimulation ausgeführt werden. Für die Simulation wird eine neue hierarchische Constraint-Prüfungsmethode eingeführt, die auf der Basis von Präzedenzbeziehungen auf höheren Detaillierungsebenen arbeitet. Ziel des Durchführens der Simulation ist, eine durchführbare und möglichst optimale Reihenfolge von Prozessschritten im Bauablauf zu generieren. Als Ergebnis werden die Anfangs- und Endzeiten für die Animation geliefert, die dann mit 3D-Objekten verknüpft werden. Die Animation wird anschließend automatisch durch vordefinierte und parametrische Ablaufanimationselemente erstellt. Mit Hilfe der erzeugten Animation soll nicht nur der Bauablauf visualisiert, sondern auch Kollisionen und potentielle Schwierigkeiten aufgezeigt werden, damit diese schon in der Planungsphase berücksichtigt werden und der Bauablauf entsprechend angepasst werden kann.

6 Referenzen

Beißert U., König M., Bargstädt H.-J., (2008), Generation and Local Improvement of Execution Schedules Using Constraint-Based Simulation. XIIth International Conference on Computing in Civil and Building Engineering, Beijing, 2008

Beißert U., König M., Bargstädt H.-J., (2007), Constraint-Based Simulation of Outfitting Processes in Building Engineering, Proc. of the CIB W78 Conference, Maribor, Slovenia

Wu I-C., Borrmann A., Rank E., Beißert U., König M., (2009), A Pattern-Based Approach for Facilitating Schedule Generation and Cost Analysis in Bridge Construction Projects, Proc. of the 26th CIB-W78 Conference on Managing IT Construction. Istanbul, Turkey

Wu I-C., Borrmann A., Beißert U., König M., Rank E. (2010), Bridge Construction Schedule Generation with Pattern-based Construction Methods and Constraint-based Simulation, Advanced Engineering Informatics, in press. doi:10.1016/j.aei.2010.07.002