

Spielerverhalten und Storyerzeugung in interaktiven Spielumgebungen am Beispiel der Grimmix Story-Engine

Kathrin Gerling, Anja Thieme, Jörg Niesenhaus, Markus Specker,
Jürgen Ziegler

Universität Duisburg-Essen, Abt. Informatik, Interaktive Systeme und Interaktionsdesign

Zusammenfassung

Aktuelle Computerspiele eröffnen dem Spieler immer größer werdende Handlungs- und Bewegungsräume mit einer Vielzahl von Freiheitsgraden. Doch die steigenden Möglichkeiten des Spielers schaffen neue Probleme für das Story-Telling im Rahmen des digitalen Spiels. Daraus ergibt sich die Fragestellung, wie die Story dem neuen Freiheitsdenken in der Spielwelt angepasst werden kann.

Diesem Problem versucht *Grimmix* entgegenzuwirken: Das auf den Märchen der Gebrüder Grimm basierende Spiel ist mit einer Story-Engine verbunden, die immer dann, wenn es im Spiel zu einer Interaktion zwischen Spieler und anderen Charakteren kommt, angesprochen wird und dafür zuständig ist, auf das Spielerverhalten zugeschnittene, aber dennoch logische Elemente der Story zurückzugeben.

1 Einleitung

Grimmix ist ein interaktives Computerspiel, das im Rahmen eines Praxisprojektes mit 15 Studierenden der Universität Duisburg-Essen entwickelt wurde. Die in *Grimmix* enthaltene Story-Engine ermöglicht eine dynamische Generierung von Spielabläufen. Inhaltlich ist das Spiel an die Märchen der Gebrüder Grimm angelehnt. Diese bilden aufgrund mehrerer Faktoren eine geeignete Grundlage für eine Beeinflussung des Spielablaufs durch das Spielerverhalten.

In den Märchen ist meist eine für den Spieler leicht nachvollziehbare recht eindeutige ethisch-moralische Polarisierung der Protagonisten identifizierbar. Auch die Spielwelt stellt ein für die meisten Spieler thematisch bekanntes Terrain dar. Ein weiterer Vorzug ist die ähnliche Grundstruktur der verschiedenen Märchen der Gebrüder Grimm, die eine Struktu-

rierung des Spiels in Module und *Quests* (Aufgaben für den Spieler) begünstigt. Im Rahmen üblicher Computerspiel-Taxonomien ist Grimmix dem Genre der Rollenspiele¹ zuzuordnen.

Während der Entwicklung stand vor allem die Frage nach Konsequenzen des Spielerverhaltens für die Spielwelt und für die in ihr stattfindende Geschichte im Vordergrund. Im Laufe des Projekts bewies sich die Interaktion des Spielers mit computergesteuerten Charakteren der Spielwelt, so genannten *NPCs*², als geeignete Grundlage für eine interaktive und dynamische Storyworld.

Diese Erkenntnis findet sich in der aktuellen Version des Spiels wieder: Trifft der Spieler auf einen NPC, überprüft die Story-Engine, ob der NPC in der Lage ist, Aufgaben, auch Quests genannt, zu vergeben. Ist dies der Fall, wird anschließend anhand verschiedener Bedingungen ein auf den Spieler abgestimmtes Quest ausgewählt.

2 Verwandte Arbeiten

Die Märchenwelt der Gebrüder Grimm wurde bereits im Projekt *InterTale* (Berndtsson & Kindmark 2003) verwendet. Hierbei wurde der ähnliche Aufbau der Märchen ausgenutzt und diese als Datenbasis für Geschichten benutzt. Über 40 Märchen wurden isoliert und mittels einer *Narrative Grammar*, die sich die typische Struktur der Grimmschen Märchen zu Nutze macht, neue Geschichten generiert.

Ebenfalls mit Märchen beschäftigte sich der russische Folklorist Vladimir Propp. Er analysierte in der ersten Hälfte des vergangenen Jahrhunderts über 100 russische Volksmärchen und stellte dabei fest, dass die Märchen viele ähnliche Komponenten aufweisen, die er als *Funktionen* bezeichnete (Propp 1968). Die Propp'schen Funktionen wurden bereits mehrfach als Grundlage für Forschungsansätze im Bereich des interaktiven digitalen Storytellings verwendet, z.B. bei Fairclough und Cunningham (2002) in ihrem System OPIATE (*Open-Ended Proppian Interactive Adaptive Tale Engine*).

Ein in der Art der Umsetzung mit Grimmix verwandtes Storytelling-System, HEFTI (*Hybrid Evolutionary-Fuzzy Time-based Interactive*) (Ong & Leggett 2004), basiert auf genetischen Algorithmen und Fuzzy Logic. Das System zeichnet sich durch eine Unterteilung in unterschiedliche Komponenten aus: einer Wissensbasis, einer Auswahl und einer Generierung von Storytelling-Elementen. Die Datenbasis des Systems, die Teil der Wissensbasis ist, besteht aus so genannten *Story Components*, welche Events, Scripts und Charaktere beinhalten. Die Wissensbasis erhält über ein in einem XML-Dialekt beschriebenes Authoring Environment jeweils neue Daten. Anhand der Spielerinteraktionen und des Fuzzy Decision Bases System werden aufgrund unterschiedlicher Regeln, Variablen und Constraints mit Hilfe von Fuzzy Logic neue Stories ausgewählt.

¹ Spiele, in denen der Spieler eine neue Rolle annimmt und aus dieser Perspektive handelt

² Non Player Character, computergesteuerte Charaktere im Spiel, mit denen der Spieler interagieren kann

3 Das Spiel

Grimmix bietet dem Spieler die Möglichkeit, die Märchenwelt mit seinem Avatar aus einer Draufsicht zu erforschen. Der Spieler kann seine Figur frei in der Spielwelt bewegen, Personen ansprechen, Gegenstände aufnehmen, diese konsumieren oder abgeben und kämpfen (siehe Abbildung 1). Der Spielaufbau ist modular, so dass das Spiel jederzeit erweitert werden kann. Dennoch arbeiten die Module nicht völlig unabhängig voneinander, sondern sind durch unterschiedliche Bedingungen miteinander verknüpft, auf die wir später detaillierter eingehen werden. Diese Bedingungen bleiben dem Spieler jedoch verborgen, ihm wird der Eindruck vermittelt, dass es sich um eine einzige, große Spielwelt handelt.

Jedes Modul stellt jeweils ein Märchen der Gebrüder Grimm in den Vordergrund. Das Spiel besteht bisher aus insgesamt drei Modulen. Neben „Schneewittchen“ sind dies „Hänsel und Gretel“ sowie „Aschenputtel“. Die aus diesen Märchen bekannten Personen und Vorgänge werden für das Storytelling aufgegriffen. Im Modul „Schneewittchen“ ist dies beispielsweise Schneewittchen als Hauptperson; die „Sieben Zwerge“ und die „böse Stiefmutter“ sind Nebenakteure. Je nach Modul sind auch „märchenfremde“ Questgeber eingebunden, die die Spielwelt glaubwürdiger und lebendiger erscheinen lassen sollen.



Abbildung 1: Screenshots aus Grimmix

Für die grafische Umsetzung von Grimmix wurde die Golden T Game Engine (GTGE)³ benutzt, eine kostenlose Game-Engine zur Entwicklung von Java-Spielen für verschiedene Genres. Sie bietet unter anderem eine 2D-Grafikengine, die auf einem Tilesset-Raster mit multiplen Ebenen beruht und sämtliche Eingaben des Nutzers, die ausschließlich über die Tastatur erfolgen, verarbeiten kann.

3.1 Die Story-Engine

Die eigens entwickelte Story-Engine ist für die Aufgabenvergabe an den Spieler sowie für die Überprüfung des Status des jeweiligen Aufgabenfortschritts verantwortlich. Sobald der

³ Informationen und Download unter www.goldenstudios.or.id/products/GTGE

Spieler auf einen NPC trifft, wird die Story-Engine über RMI⁴ von der Game-Engine angesprochen, um zu ermitteln, ob der Spieler eine neue Aufgabe (*Quest*) vom NPC erhalten soll und wenn ja, welche. Aufgrund der Kommunikation mittels RMI zwischen Game-Engine und Story-Engine kann das Spiel auch problemlos über das Internet gespielt werden.

Bevor ein NPC eine Aufgabe vergibt, wird mit dem Spieler ein *Gesinnungsdialog* mit vorgefertigten Antwortmöglichkeiten geführt. Dabei ist jede Antwortmöglichkeit einem Gesinnungstyp zugeordnet, so dass nicht nur die bereits bearbeiteten *Quests*, sondern auch das momentane Verhalten des Spielers einen Einfluss auf die Gesinnung des Spielers nehmen können.

Trifft der Spieler beispielsweise auf die Hexe, die auf der Suche nach Hänsel und Gretel ist, so kann diese ihn zu seiner Meinung über Kinder befragen. Wählt der Spieler dann die negativ belegte Antwortoption „Ich habe Kinder zum Fressen gern...“, steigt die Wahrscheinlichkeit, dass die Hexe ihm ein „böses“ *Quest* anbietet.

Die *Gesinnung* des Spielers wird über dessen Spielverhalten bestimmt und hat einen großen Einfluss auf die Auswahl der *Quests* durch die Story-Engine. Zu Beginn des Spiels wird dem Spieler eine neutrale Gesinnung zugeordnet, die sich im Laufe des Spiels jedoch in Abhängigkeit der von ihm bearbeiteten Aufgaben verändern kann. Dabei werden die Spielerhandlungen jeweils als *gut*, *böse* und *neutral* bewertet. Sowohl dem Spieler als auch den *Quests* sind Gesinnungswerte zugeordnet, die bei erfolgreicher Beendigung einer Aufgabe miteinander verrechnet werden. Durch den Gesinnungswert übt das Spielerverhalten einen Einfluss auf die Spielwelt und die weitere Entwicklung der Story aus.

Die *Quests*, die durch die Story-Engine ausgewählt werden, liegen als vorgefertigte Konstrukte im XML-Format vor. Jeder *Questgeber* verfügt über mindestens ein *Quest* pro Gesinnungsausrichtung, so dass der Spieler entsprechend seiner eigenen Gesinnung stets gute, neutrale oder böse Aufgaben erhalten kann. Durch einen dazu gehörenden XML-Editor können *Quests* auch später noch eingefügt werden.

Um zur Spielergesinnung passende *Quests* zu erhalten, wurden probabilistische Bayes-Netze modelliert, die es ermöglichen, eine Wahrscheinlichkeitsverteilung der jeweils zur Auswahl stehenden *Quests* zu erhalten. Dabei wird berücksichtigt, dass der Spieler ein *Quest* nicht annehmen kann, sofern sich dieses von einem bereits absolvierten *Quest* aus einem der vorangegangenen Module thematisch nicht grundlegend unterscheidet. Dadurch soll gewährleistet sein, dass das Spiel dem Spieler möglichst abwechslungsreiche *Quests* offeriert. Insgesamt gibt es je Modul ein Hauptquest-Netz und vier Nebenquest-Netze für die vier Hauptcharaktere eines jeden Märchenmoduls.

Abbildung 2 zeigt, dass bei einem neutralen Spielerverhalten und einem als „böse“ bewerteten *Gesinnungsdialog* die Wahrscheinlichkeit steigt, ein negatives Hauptquest (S_H_Mieder) zu erhalten. Bei erfolgreicher Beendigung wirkt sich die negative Gewichtung des *Quests* wiederum auf die Gesinnung des Spielers (Spielerstatus) aus.

⁴ Remote Method Invocation, Methodenaufruf eines entfernten Java-Objekts

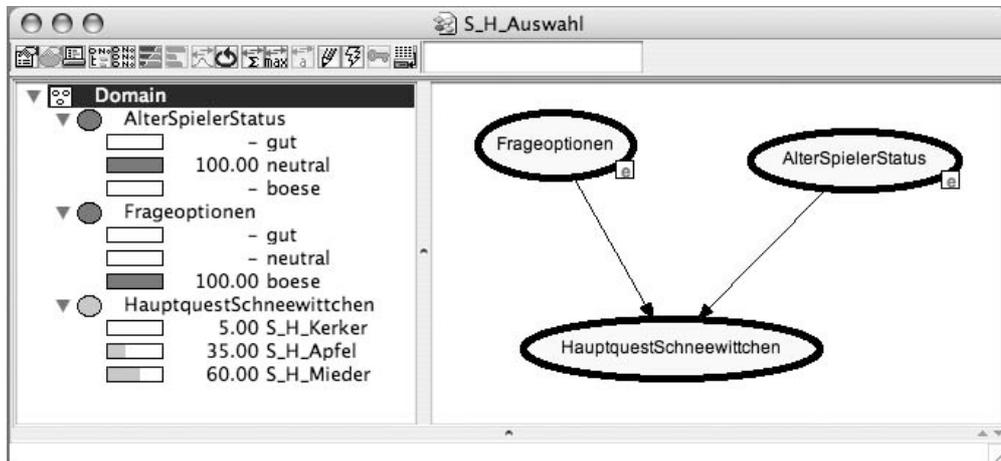


Abbildung 2: Beispiel zur Gesinnungsbeeinflussung durch Fragendialoge

Für die Auswahl der Quests spielen neben der Gesinnung noch andere Einflussfaktoren eine Rolle. Diese werden im Folgenden näher erläutert.

In der *Spielerfahrung* wird das Wissen des Spielers über die Spielwelt und die Vorgänge in ihr zusammengefasst. Die Spielerfahrung setzt sich aus unterschiedlichen Eigenschaften zusammen, wie die vom Spieler im Spiel bisher getroffenen NPCs, die von ihm gesammelten Gegenstände sowie die Anzahl der bereits absolvierten Aufgaben. Da davon auszugehen ist, dass sich die Gesinnung eines Spielers während des Spielverlaufs festigt, wird ihr Einfluss in Abhängigkeit vom Faktor Spielerfahrung im Laufe des Spiels gesteigert.

Weiterhin können *frei gewählte Bedingungen* die Questwahl beeinflussen. So kann zum Beispiel die Bearbeitung eines bestimmten Quests oder die Ausführung einer bestimmten Handlung im Spiel Einfluss auf die spätere Questwahl der Story-Engine haben, so dass der Spieler zu einem möglichst abwechslungsreichen Spielerlebnis gelangt.

Alle Bayes-Netze wie auch die Module lassen sich nahezu beliebig durch weitere Einflusskriterien erweitern.

Neben der Möglichkeit, dass die Storyengine Quests über probabilistische Bayes-Netze aus einem Pool auswählt, besteht zusätzlich die Möglichkeit, mittels Prolog dynamisch Quests zu *generieren*. Diese werden frei aus einem Personen-, Item- und Aktionspool zusammengestellt und dann erzeugt, wenn der Spieler alle regulären Quests bereits bearbeitet oder sie abgelehnt hat. Die Prolog-Quests folgen fest definierten Regeln, und gewährleisten damit die Konsistenz der Story.

Ein mittels Prolog generiertes Quest besteht aus vier Faktoren: Zum einen muss es Informationen enthalten, um was für eine Aufgabenart es sich handeln soll, zum anderen benötigt die Game-Engine Informationen über im Quest eingebundene Personen, das Ziel des Quests sowie die Belohnung, die der Spieler bei Beendigung erhalten soll. Für die Umsetzung wurde SWI-Prolog in Verbindung mit JPL gewählt.

4 Fazit

Die in das Spiel Grimmix eingebundene Story-Engine ist ein neuer Ansatz, der sich unterschiedlicher Methoden und Technologien bedient, um diese in Kombination zur Story-Erzeugung einzusetzen. Die Verbindung einer Datenbasis aus Quests, die im XML-Format gespeichert werden, probabilistischer Bayes-Netze zur Auswahl der Quests und eines regelbasierten Systems zur dynamischen Questerzeugung mittels Prolog bilden die Grundlage für einen auf das Verhalten des Spielers abgestimmten Spielverlauf, der damit Einfluss auf Spielwelt und Storytelling gewinnt.

Vorteil dieses Ansatzes ist, dass der Aufgabenpool der Story-Engine jederzeit – und je nach Implementierung auch unabhängig vom Spiel – erweitert werden kann. Die Trennung von Story-Engine und Game-Engine erlaubt zusätzlich den Anschluss unterschiedlicher Visualisierungsmethoden, angefangen bei der Text-Ausgabe bis hin zu einer 3D-Spielumgebung.

Literaturverzeichnis

- Berndtsson, K.; Kindmark, L. (2003): Intertale – A new way to create interactive narratives. In: Proceedings of Technologies for Interactive Digital Storytelling and Entertainment. Darmstadt: Teubner, S. 9-24.
- Fairclough, C.; Cunningham, P. (2002): "An interactive Story Engine", AICS 2002, LNAI 2464, S 171-176.
- Ong, T.; Leggett, J. (2004): A genetic algorithm approach to interactive narrative generation. In: Proceedings of the fifteenth ACM Conference on Hypertext and Hypermedia. New York: ACM Press.
- Propp, V.: (1968): Morphology of the Folktale. Austin, TX: University of Texas Press.

Danksagung und Kontaktinformation

Wir danken allen Teilnehmern des Praxisprojektes für ihren engagierten Einsatz am Grimmix-Projekt.

Jörg Niesenhaus, Markus Specker, Jürgen Ziegler
{niesenhaus, specker, ziegler}@interactivesystems.info