

Interaktive Evolution zur Assistenz bei der Einrichtungsplanung

Masterarbeit, vorgelegt von Stephan Dreyer



Zielstellung

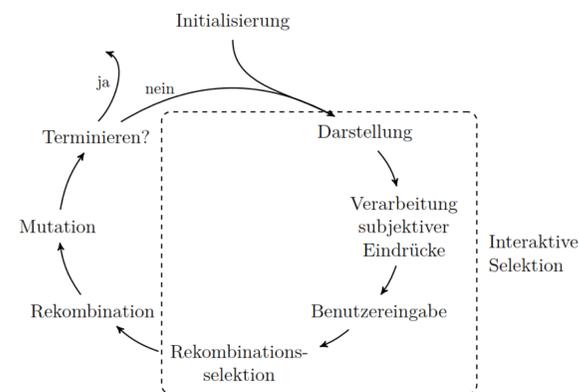
Die Planung der Inneneinrichtung von Räumen ist oft aufwändig und zeitintensiv, da ein weites Spektrum verschiedenster Einrichtungsgegenstände verfügbar ist und diese fast beliebig kombiniert werden können. Klassische Systeme zur Inneneinrichtungsplanung nutzen oft einen lösungsorientierten Ansatz, bei dem der Benutzer die Einrichtungsgegenstände auswählt und im Raum platziert. Vom Benutzer wird verlangt, dass er sich bereits ein Ziel gesetzt hat. Ein hohes Maß an Eigenleistung ist ebenfalls erforderlich. Bei einem problemorientierten Ansatz hingegen werden dem Benutzer Lösungsvorschläge präsentiert, die er anschließend bewertet. Das Ziel der Arbeit war, einen solchen Ansatz mit Hilfe interaktiver evolutionärer Algorithmen umzusetzen. Dazu musste zunächst eine geeignete Kodierung der Lösungskandidaten gefunden werden. Die Bewertung sollte den Benutzer nicht überfordern, daher war die Benutzerschnittstelle einfach zu halten. Die Zusammenstellungen der Einrichtungsgegenstände sollten möglichst realitätsgetreu dargestellt werden.

Interaktive evolutionäre Algorithmen

Bei der biologischen Evolution nach Charles Darwin findet eine Anpassung von Lebewesen (Individuen) an die Umwelt statt, wenn folgende Bedingungen erfüllt sind:

- Individuen einer Population können sich reproduzieren
- Es treten Veränderungen auf, welche die Reproduktionsrate beeinflussen
- Die Veränderungen können vererbt werden
- Die Ressourcen sind begrenzt

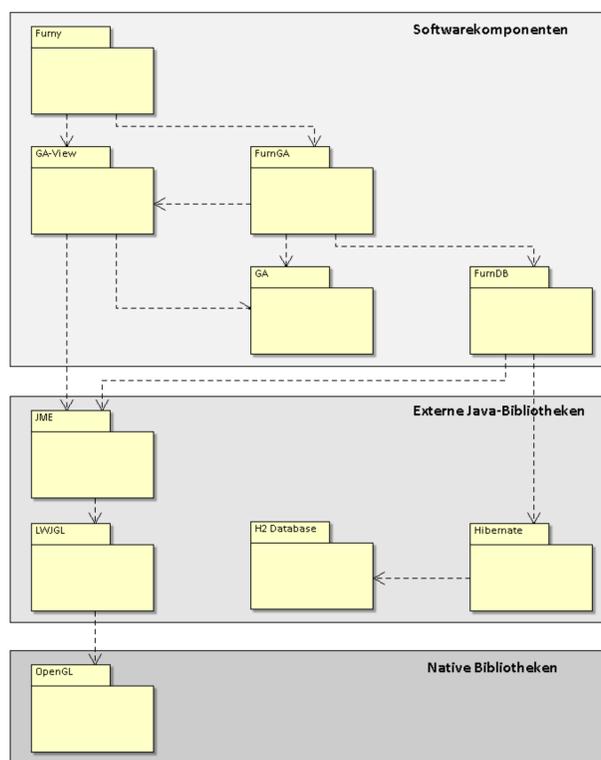
Evolutionäre Algorithmen übertragen dieses Prinzip auf technische Systeme. Individuen sind dort keine Lebewesen, sondern potentielle Problemlösungen. Die Population wird meist zufällig erzeugt. Gute Problemlösungen haben auch eine höhere Chance zur Reproduktion. Bei interaktiven evolutionären Algorithmen beurteilt der Benutzer die Güte der Lösungskandidaten. Bei der Reproduktion findet die Rekombination (lokale Suche, Eigenschaften werden kombiniert) sowie die Mutation (globale Suche, Eigenschaften werden zufällig verändert) mit definierten Wahrscheinlichkeiten statt. Die Optimierung wird meist beim Erreichen eines Ziels oder den Abbruch durch den Benutzer beendet.



Der interaktive evolutionäre Zyklus definiert einen festen Ablauf der einzelnen Schritte. Der Benutzer wird an der interaktiven Selektion beteiligt.

Systemarchitektur

Zur Umsetzung der Modifizierbarkeit, Wiederverwendbarkeit und der losen Kopplung wurde das System in verschiedene Module aufgeteilt. Die Anwendung Furny bildet die oberste Schicht und setzt auf allen Modulen auf. Diese beinhaltet auch die Hauptklasse, mit der alle Teilkomponenten gestartet werden können. Die Bibliothek GA bietet generische Klassen und Schnittstellen zur Implementierung evolutionärer Algorithmen. FurnGA ist die Ausprägung eines interaktiven evolutionären Algorithmus für Einrichtungspläne. FurnDB ist die Persistenzschicht und speichert Einrichtungsgegenstände mittels Hibernate in einer relationalen Datenbank ab. GA-View enthält verschiedene Benutzerschnittstellen, die auf der jMonkeyEngine, einer in Java geschriebenen 3D-Bibliothek basieren.



Softwaremodule und Bibliotheken in der Übersicht.

Kodierung der Individuen

Ein Individuum entspricht einem einzelnen Einrichtungsplan. Dieser ist genetisch als Liste von Einrichtungsgegenständen mit der jeweiligen Position und dem Winkel im Raum kodiert. Diese Liste hat theoretisch keine begrenzte Länge, jedoch wird der Benutzer Einrichtungspläne mit zu vielen Gegenständen schlecht bewerten, sodass diese sich nicht reproduzieren werden. Weiterhin wird die Anzahl der Einrichtungsgegenstände durch die begrenzte Größe des Raums eingeschränkt.

Sparse Fitness Evaluation

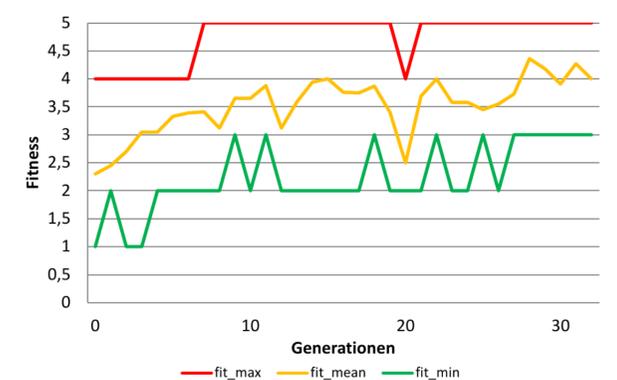
Zur Verringerung der Benutzerermüdung wurde die Methode der Sparse Fitness Evaluation umgesetzt. Diese besteht aus zwei Schritten: im ersten Schritt wird ein Clustering der Population durchgeführt, so dass ähnliche Individuen gruppiert werden. Nur die Zentren der Cluster werden vom Benutzer direkt bewertet. Im zweiten Schritt wird allen anderen Individuen im Cluster eine Fitness in Abhängigkeit vom Abstand des Individuums zum Zentrum zugewiesen. Dies ist eine Form der Fitnessprognose. Zur Umsetzung musste eine Abstandsfunktion entworfen werden.

Benutzerschnittstelle

Um der Forderung nach einer realitätsgetreuen Darstellung nachzukommen, wurde die Benutzerschnittstelle mit der Grafikkibliothek jMonkeyEngine realisiert (siehe Abbildung links oben). Die Vogelperspektive bietet eine gute Übersicht und ist dreh- und schwenkbar. Die Bewertung erfolgt in fünf Abstufungen. Zur Vergabe der Punkte muss lediglich eine der Zahlen im rechten Bereich des Bildschirms angeklickt werden. Der Abbruch des Algorithmus ist jederzeit mit der Schaltfläche rechts oben möglich.

Versuche

Zum Nachweis der Funktion des interaktiven evolutionären Algorithmus wurden verschiedene Versuche durchgeführt. Beim ersten Versuchsaufbau wurde gezeigt, dass der verwendete k-Means++-Clusterer die Population nach der subjektiven Ähnlichkeit der Individuen gruppiert. Der zweite Versuchsaufbau diente dem Nachweis der Funktion des Algorithmus unter Vorgabe eines festen Ziels. Die Bewertung der Güte von Individuen erfolgte hier automatisch auf Basis der Abstandsfunktion. Das Ziel wurde in 7 von 12 Versuchen ohne Sparse Fitness Evaluation erreicht. Mit der Sparse Fitness Evaluation wurde das Ziel nie erreicht, es fand jedoch eine Annäherung an das Optimum statt. Im dritten Versuchsaufbau wurde dem Benutzer ein Ziel-Individuum vorgegeben, das er durch die interaktive Bewertung erreichen sollte. In einem der acht Versuche konnte eine zufriedenstellende Lösung gefunden werden. Auch hier lieferte die Sparse Fitness Evaluation insgesamt schlechtere Ergebnisse.



Fitnessverlauf aus den durchgeführten Versuchen mit manueller Evaluation. Die minimale, durchschnittliche und maximale Fitness verbessert sich mit den Generationen.

Ergebnisse

In der Arbeit wurde ein interaktiver evolutionärer Algorithmus zur Planung der Inneneinrichtung entworfen und umgesetzt. Die Methode der Sparse Fitness Evaluation wurde ebenfalls implementiert. In den durchgeführten Versuchen konnte bestätigt werden, dass eine Optimierung stattfindet. Ein vorgegebenes Ziel konnte dabei jedoch nicht immer erreicht werden. Als potentielle Schwachstellen konnten die variable Struktur des Genotyps, die Suche nach einer allgemeingültigen Definition der Abstandsfunktion und die Fülle an Parametern, die der Algorithmus bietet, adressiert werden. Im Ausblick wurden zahlreiche Vorschläge zur Verbesserung des Algorithmus gemacht.