

Dynamische Partitionierung von Multiagentensystemen in einem Evakuierungsszenario

Christian Thiel

HAW Hamburg
Fakultät für Technik und Informatik

26.11.2010

1 Das Projekt

2 Partitionierung

- Warum Partitionierung?
- Partitionierungsverfahren

3 Clusterverfahren

4 Lösungssuche

Simulation von Massenverhalten durch Multiagentensysteme

Ziel des Projektes

- Entwicklung eines Tools zur Simulation von Massenpaniken
- Simulation auf Basis von Agenten (bisher vorwiegend nur durch Partikel- oder Flusssysteme)
- vorerst Anwendung nur in kleinen Szenarien wie U-Bahn-Stationen
- Kooperation mit der HAW München

Vorteile von Agentensystemen

- Individuelle Behandlung jeder einzelnen Person
- individuelle Festlegung von Gefühlen, biologischen oder sozialen Eigenschaften festlegbar
 - ▶ einige Personen geraten leichter in Panik als andere
 - ▶ Panik äussert sich bei jedem Menschen anders (Aggression, Tunnelblick etc.)
 - ▶ unterschiedlich alte, schnelle und starke Menschen
- Modellierung spezieller Szenarien
 - ▶ Familie wird während der Simulation getrennt -> Vater sucht panisch nach seinen Kindern und stellt so ein Hindernis dar
 - ▶ Sturz auf Treppen oder Rolltreppen als Hindernis
 - ▶ speziell auf Bahnsteigen Gefahr auf das Gleis zu stürzen

Modellierungsansätze

- Modellierung der KI
 - ▶ Modellierung des Verhaltens mit Hilfe von Emotionen (Wut, Ekel, Angst, Freude, Trauer etc.)
 - ▶ verschiedene Voreinstellung für verschiedene Charaktere
 - ▶ Ereignisse in der Umgebung beeinflussen Emotionszustände
 - ▶ Entscheidungsfindung auf Basis von Wahrnehmung, jedoch stark beeinflusst von Emotionen (siehe dazu [2])
- 3D-GIS als ebenenbasiertes Geoinformationssystem

Überblick

1 Das Projekt

2 Partitionierung

- Warum Partitionierung?
- Partitionierungsverfahren

3 Clusterverfahren

4 Lösungssuche

Warum Partitionierung?

- Agenten sind deutlich aufwändiger als reine Bewegungsroboter
- einzelne Serverplattform wird bei größeren Szenarien schnell überlastet
- Verteilung der Agenten auf mehrere Server notwendig
- Partitionierungsalgorithmen erforderlich

Warum intelligent?

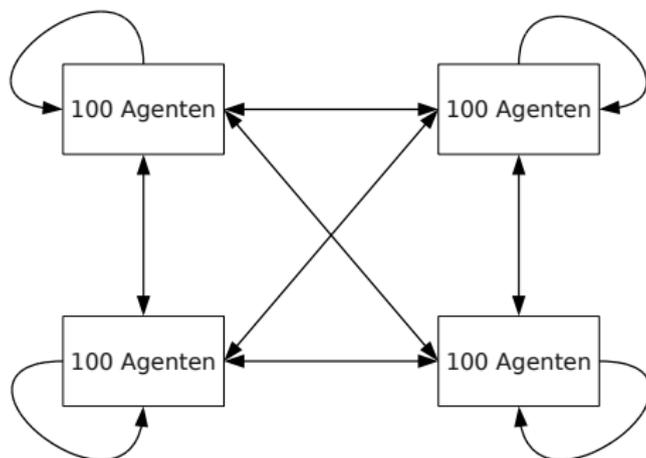


Abbildung: Unbalancierte Verteilung der Agenten

- Überall gleich viel Kommunikation zwischen Agenten bzw. Servern
- Netzwerk könnte durch zu hohe Last zum Flaschenhals werden
- höhere Latenzzeiten könnten Simulation beeinflussen

Warum intelligent? (2)

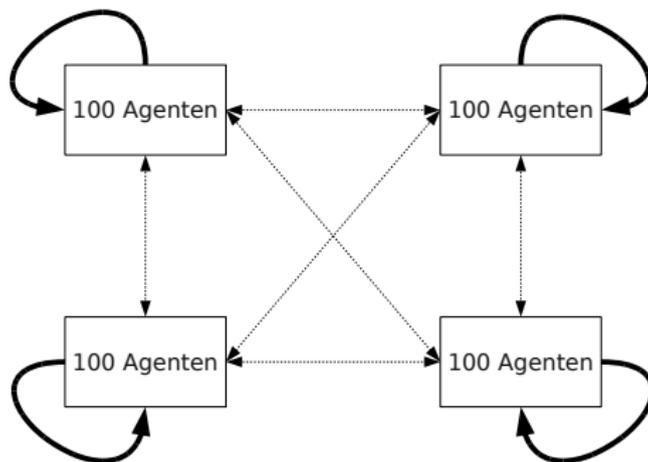


Abbildung: Balancierte Verteilung der Agenten

- Kommunikationspfade hauptsächlich lokal
- deutlich weniger Netzwerklast
- reduzierte Latenzzeiten

Überblick

- 1 Das Projekt
- 2 Partitionierung**
 - Warum Partitionierung?
 - Partitionierungsverfahren**
- 3 Clusterverfahren
- 4 Lösungssuche

Lösung 1: Feste geografische Aufteilung

Lösung:

- der gesamte Lageplan des Gebäudes wird in feste Bereiche geteilt, welche strategisch gut verteilt sind (z.B. Wände oder Treppen als Grenze nutzen)
- Agenten werden anhand ihrer aktuellen Position aufgeteilt
- Positionswechsel kann zu Migration der Agenten führen

Probleme:

- Partitionierung statisch, es droht mögliche Überlastung einzelner Partitionen (z.B. Ausgänge von Gebäuden)

Lösung 2: Bäume

Lösung:

- Wie Lösung 1, nur diesmal als Baum
- Aufteilung einzelner Partitionen bei Überfüllung
- Verfahren:
 - ▶ Quad-Tree
 - ▶ K-d-Tree

Probleme:

- Anzahl der Partitionen ist nicht stabil, es kann also nicht entsprechend skaliert werden

Lösung 3: Clustering

Lösung:

- Agenten werden mit Hilfe von Clusteralgorithmen intelligent aufgeteilt
- abstrakte Ähnlichkeitsfunktion bestimmt Interessensbereich der Agenten
- Clusterverfahren (Auswahl)
 - ▶ Graphentheoretisch: DBSCAN
 - ▶ Hierarchisch
 - ▶ Partitionierend: K-Means

DBSCAN

Density-Based Spatial Clustering of Applications with Noise

- Menge aller Objekte als Graph aufgespannt
- Basiert auf Dichteverbundenheit der einzelnen Knoten
- Drei verschiedene Arten von Knoten
 - ▶ **Kernobjekte** Objekte, die selbst dicht sind
 - ▶ **dichte-erreichbar** von Kernobjekten erreichbar
 - ▶ **Noise** nicht von Kernobjekten erreichbar

Dichteverbundenheit

Ein Objekt ist *dicht*, wenn es mindestens *minPt* Nachbarn hat.

Zwei Objekte gelten als *dichteverbunden*, wenn sie durch eine Kette von bereits *dichten* Objekten miteinander verbunden werden.

DBSCAN (2)

- Verfahren ist deterministisch
- lineare Komplexität
- parallelisierbar

Probleme:

- Clusteranzahl ist variabel
- es kann vorkommen, dass alle Objekte in einem Cluster sind
- Rauschobjekte werden nicht zugeordnet

Hierarchische Cluster

- Ähnlichkeitsfunktion von zwei Agenten symbolisiert logischen Abstand (Kommunikationsaufkommen, Distanz)
- Zu Beginn ist jeder Agent ein eigenes Cluster
- Berechnung der Distanzen zwischen allen Clustern
- Schrittweise zusammenführung der Cluster mit den geringsten Abständen bis nur noch n Cluster vorhanden sind

Hierarchische Cluster (2)

- Verfahren ist deterministisch
- Clusteranzahl ist stabil

Probleme:

- sehr hoher Rechenaufwand und Speicherbedarf
- nur bedingt parallelisierbar wegen Sortierung bei Zusammenführung
- Clustergrößen sind sehr variabel

K-Means

- Wahl von k Clusterzentren zufällig aus der Menge aller Objekte
- Zuordnung aller Objekte zu Clusterzentren anhand einer Ähnlichkeitsfunktion
- Neuberechnung der jeweiligen Clusterzentren
- Erneute Zuteilung aller Objekte
- Wiederholung bis keine Änderung mehr auftritt

K-Means (2)

- Clusteranzahl ist stabil
- Aufwand ist weitgehend linear
- iteratives, abbrechbares Verfahren
- parallelisierbar

Probleme:

- Verfahren ist nicht deterministisch
- Clustergrößen sind sehr variabel

Statische Partitionen und Bäume sind ungeeignet

Eigenschaften des gesuchten Clusteralgorithmus:

- Algorithmus muss schnell Ergebnisse liefern
- Clusteranzahl muss konstant sein
- Anzahl der Agenten innerhalb eines Clusters darf nicht zu sehr vom Durchschnitt abweichen
- Parallelisierbarkeit wünschenswert, aber nicht zwingend notwendig
- muss nicht zwingend deterministisch sein

Quellen



T. Clusteranalyse.
K-Means.



A. Klingenberg.
Prototypische Entwicklung eines emotionalen Agenten auf der Basis des Goal Oriented Action Plannings.
Master's thesis, HAW Hamburg, 2009.



T. Schlegel, P. Braun, and R. Kowalczyk.
Towards autonomous mobile agents with emergent migration behaviour.
In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems, AAMAS '06*, page 585–592, New York, NY, USA, 2006. ACM.



J. Schäfer.
KD-Bäume.



A. Steed and R. Abou-Haidar.
Partitioning crowded virtual environments.
In *Proceedings of the ACM symposium on Virtual reality software and technology, VRST '03*, page 7–14, New York, NY, USA, 2003. ACM.