

Verifikation von Multiagentensystemen

Peter Salchow

INF-M3 - Anwendungen II
Wintersemester 2008
Department Informatik
HAW Hamburg

07. Januar 2009



Inhalt

- 1 Einleitung
 - Vorhaben
 - Zielsetzung
- 2 Related Work
 - Agent PathFinder
 - AgentSpeak
 - AVACS
- 3 Revision
- 4 Ausblick
- 5 Quellen



Inhalt

- 1 Einleitung
 - Vorhaben
 - Zielsetzung
- 2 Related Work
 - Agent PathFinder
 - AgentSpeak
 - AVACS
- 3 Revision
- 4 Ausblick
- 5 Quellen



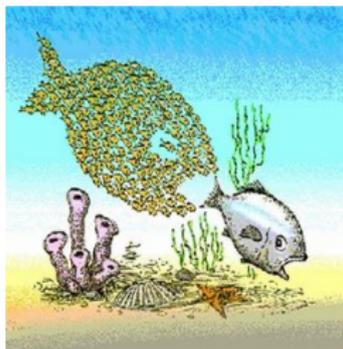
Vorhaben I

- Viele Einsatzgebiete von Multiagentensystemen (MAS)
 - Unabhängige Einheiten im Meer zur Kontrolle von Ölnfällen (Uni Oldenburg)
 - Stundenplanerstellung über Agenten (FH Gelsenkirchen)
 - Optimierung des Paketversands von DHL durch Agenten (HAW)
 - Autonome Raumschiffsteuerung durch Agenten (Liverpool)
- Hohes Potenzial komplexe Systeme über MAS zu realisieren



Vorhaben II

Zusammenspiel vieler Agenten kann emergentes Verhalten hervorrufen - Schwarmverhalten



Vorhaben III

- Für „normale“ Softwaresysteme existieren Ansätze und Methoden zur Verifikation
- Übertragung dieser Mechanismen auf Multiagentensysteme
- Nachweis der Korrektheit solcher Systeme

⇒ **Verifikation von Multiagentensystemen**



Zielsetzung

Ziel 1

Wie können Multiagentensysteme verifiziert werden (allgemein)?

- Unterschiede zwischen verteilten Systemen und MAS
- Formalismen (Methoden) dafür identifizieren
- Welche Vorgehensmodelle erscheinen geeignet?



Zielsetzung

Ziel 2

Kombination von Multiagentensystemen und formaler Verifikation

- Erarbeiten bis zu welchem Grad Agenten verifiziert werden können
- Verifizierbare Eigenschaften identifizieren
- Einteilung in lokale und globale Eigenschaften



Zielsetzung

Ziel 3

Prüfung der Formalismen auf Tauglichkeit zur Verifikation von MAS

- Erarbeiten von bereits existierenden Formalismen
- Vor- und Nachteile der Formalismen identifizieren
- Formalismen mit direkter Codeverifikation kombinieren



Zielsetzung

Ziel 4

Erarbeiten eigener Methoden zur erfolgreichen Verifikation

- Betrachtung bestimmter Bereiche von MAS
- Pattern für bestimmte Probleme entwickeln



Inhalt

- 1 Einleitung
 - Vorhaben
 - Zielsetzung
- 2 Related Work
 - Agent PathFinder
 - AgentSpeak
 - AVACS
- 3 Revision
- 4 Ausblick
- 5 Quellen



Agent PathFinder

- Automatische Verifikation von Agentensystemen (BDI-Agenten)
- Integration der BDI-Semantik in den Java-Code
- Basiert auf bestehenden Java-Erweiterungen für Agenten (z.B. JACK)
- Direktes Model Checking von Java-Programmen (Java PathFinder)



Java PathFinder

- Verifiziert Eigenschaften von Java-Programmen
- Untersucht Bytecode (eigene VM über Java VM)
- Eigenschaft wird geprüft, in dem alle Ausführungspfade durchlaufen werden
- Durch Abstraktion kann Suchraum eingeschränkt werden
- Abstraktion nur unter Berücksichtigung der zu prüfenden Eigenschaft
- Bei gefundenen Fehler wird Weg dorthin ausgegeben



Java PathFinder

Vorteile:

- Direkte Verifikation von Java-Code
- Automatisches Traversieren durch gesamten Zustandsraum

Nachteile:

- Zustandsraum muss endlich sein
- BDI-Semantik für Java-Programme muss definiert werden
- Abstraktion ist sehr komplex



Inhalt

- 1 Einleitung
 - Vorhaben
 - Zielsetzung
- 2 Related Work
 - Agent PathFinder
 - AgentSpeak
 - AVACS
- 3 Revision
- 4 Ausblick
- 5 Quellen



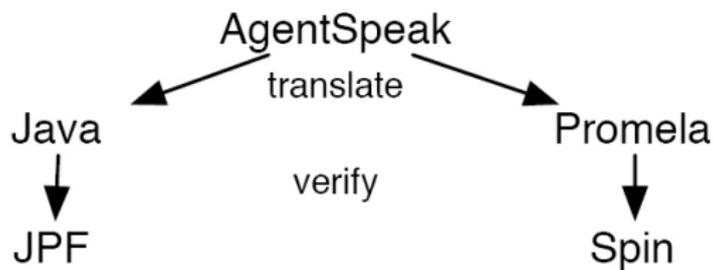
AgentSpeak

- Regelbasierte Programmiersprache für BDI-Agenten
- Beliefs werden als Fakten vorgegeben (ähnlich Prolog)
- Entwicklung eines gesamten Systems in einer formalen Sprache
- Plattformen für Agenten führen MAS direkt aus

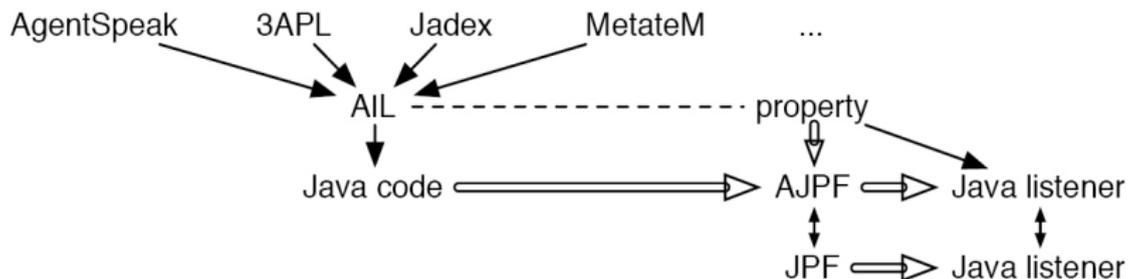


AgentSpeak

- Mehrere Ansätze AgentSpeak zu verifizieren
- Meist nicht direkt verifiziert
- Überführung in andere Sprache z.B. PROMELA oder Java



Framework zur Verifikation



- 1 MAS in beliebiger Sprache
- 2 Zu testende Eigenschaften festlegen
- 3 Übersetzen des Programms in AIL-Repräsentation
- 4 Übersetzen der Eigenschaften in PSL
- 5 Umgebung definieren
- 6 Programm in AJPF ausführen / verifizieren



Inhalt

- 1 Einleitung
 - Vorhaben
 - Zielsetzung
- 2 Related Work
 - Agent PathFinder
 - AgentSpeak
 - AVACS
- 3 Revision
- 4 Ausblick
- 5 Quellen



Automatic Verification and Analysis of Complex Systems

- Entwicklung von Methoden und Verfahren zur mathematischen Verifikation
- Analyse von Modellen komplexer, sicherheitskritischer, eingebetteter Systeme
- Bisher nur eingeschränkte Analyse von Teilgebieten (Nebenläufigkeit, Zeitverhalten, Stabilität ...)
- Ziele:
 - Umfassende, ganzheitliche Verifikation dieser Systeme
 - Entwicklung automatischer Verifikationstechniken
 - Vollständige Überdeckung des Modellraums



Inhalt

- 1 Einleitung
 - Vorhaben
 - Zielsetzung
- 2 Related Work
 - Agent PathFinder
 - AgentSpeak
 - AVACS
- 3 **Revision**
- 4 Ausblick
- 5 Quellen



Revision der Ziele

Ziel 1

Wie können Multiagentensysteme verifiziert werden (allgemein)?



Revision der Ziele

Ziel 2

Kombination von Multiagentensystemen und formaler Verifikation

- Erarbeiten bis zu welchem Grad Agenten verifiziert werden können
- Verifizierbare Eigenschaften identifizieren
- Einteilung in lokale und globale Eigenschaften



Revision der Ziele

Ziel 3

Prüfung der Formalismen auf Tauglichkeit zur Verifikation von MAS

- Erarbeiten von bereits existierenden Formalismen
- Vor- und Nachteile der Formalismen identifizieren
- Formalismen mit direkter Codeverifikation kombinieren



Revision der Ziele

Ziel 4

Erarbeiten eigener Methoden zur erfolgreichen Verifikation

- Betrachtung bestimmter Bereiche von MAS
- Pattern für bestimmte Probleme entwickeln



Inhalt

- 1 Einleitung
 - Vorhaben
 - Zielsetzung
- 2 Related Work
 - Agent PathFinder
 - AgentSpeak
 - AVACS
- 3 Revision
- 4 Ausblick**
- 5 Quellen



Ziele und Grenzen sind bekannt! Was muss jetzt getan werden...

- Weitere Quellen recherchieren
- Diese gezielt auswerten
- Mit Betreuer Thema konkretisieren
- In konkrete Gebiete tiefer einarbeiten
- Kontakt zu Spezialisten auf den Gebieten aufnehmen



Quellen I

[Alechina06]

Alechina, Natasha: Verifying space and time requirements for resource-bounded agents. In: International Conference on Autonomous Agents. ACM, 2006

[Bordini03]

Bordini, Rafael H. ; Fisher, Michael [u.a.]: Model Checking AgentSpeak. In: AAMAS'03. ACM, 2003

[Bordini04]

Bordini, Rafael H. ; Fisher, Michael [u.a.]: Model checking rational agents. In: IEEE Intelligent Systems. IEEE, 2004

[Dennis08]

Dennis, Louise A. [u.a.]: A flexible framework for verifying agent programs. In: International Conference on Autonomous Agents. ACM, 2008



Quellen II

[Fisher02]

Fisher, Michael ; Visser, Willem: Verification of Autonomous Spacecraft Control - A logical vision of the future. 2002

[Fisher04]

Bordini, Rafael H. ; Fisher, Michael [u.a.]: State-Space Reduction Techniques in Agent Verification. In: AAMAS'04. ACM, 2004

[Henesey08]

Henesey, Lawrence: Agent based simulation architecture for evaluating operational policies in transshipping containers. Springer, 2008

[Holzmann04]

Holzmann, Gerard J.: The Spin Model Checker : Primer and Reference Manual. 2004



Quellen III

[Kramer07]

Kramer, Jeff: Is abstraction the key to computing?. In: Communications of the ACM: 2007.

[Lam04]

Lam, Dung N. [u.a.]: Verifying and Explaining Agent Behavior in an Implemented Agent System. In: International Conference on Autonomous Agents. ACM, 2004

[Roos07]

Roos, Nico; Witteveen, Cees: Models and methods for plan diagnosis. Springer, 2007

[SpinWeb]

<http://spinroot.com/spin/whatispin.html>



Quellen IV

[Sudeikat08]

Sudeikat, Jan; Renz, Wolfgang: A Systemic Approach to the Validation of Self-Organizing Dynamics within MAS. 2008

[Wooldridge02]

Wooldridge, Michael ; Fisher, Michael [u.a.]: Model Checking Multi-Agent Systems with MABLE. ACM, 2002

[Wooldridge06]

Wooldridge, Michael: An Introduction to Multiagent Systems. 2006



Vielen Dank für die Aufmerksamkeit!
Fragen?

