

Ameisenalgorithmen

Fachhochschule Köln, Campus GM
Algorithmische Anwendungen
WS 05/06
Prof. Heiner Klocke

Karsten Will, Mat. Nr 11038919

Wer hat's erfunden

- Marco Dorigo, italienischer Mathematiker
- 1991 in Milan, Italien
- Heute an der Universität Brüssel
- Literatur : AntNet – A mobile agents approach to adaptive Routing

Routing in Kommunikationsnetzen



Shortest Path Routing

- Start-Ziel-Paar Perspektive
- Entscheidet über kürzesten Weg zwischen zwei Knoten anhand eines Kostenkriteriums (Laufzeit)
- Probleme:
 - Statisches Kriterium -> passt nicht auf alle Umstände
 - Dynamische Berücksichtigung -> Oszillationseffekt

Ablauf SPR

- Jeder Knoten kennt Kosten (statisch o. dynamisch) seiner Nachbarknoten
- Periodisch wird an alle Nachbarn ein Info-Paket geschickt (Laufzeiten, Entfernung)
- Diese integrieren das in ihr eigenes Wissen
- Routing-Entscheidungen können deterministisch (anhand des Wissens) getroffen werden

Routing im Internet (Historie)

- Arpanet (1969) dynamische Verbindungskosten
- insbesondere: wieviele Pakete sind in der Warteschlange
- Führt zu starken Oszillationseffekten
- Grosse Konstante addiert
- Resultat: Weniger empfindlich für Veränderungen

Routing im Internet (Historie)(2)

- 1980 neuer Algorithmus : Shortest Path First
- Immer noch dynamische Verbindungskosten-Metrik
- Jetzt unter Einbeziehung von Übertragungs- und Propagations- Verzögerungen
- Besser aber immer noch Oszillation unter starker Last

Routing im Internet (heute)

- OSPF (Open Shortest Path First)
- Im wesentlichen statisch (Metrik von Administratoren ermittelt)
- Nur topologische Änderungen werden dynamisch berücksichtigt
- Nicht „best practice“ sondern „best compromise“

Sinnfrage

- Entwicklung bisher sehr konservativ
- Trotzdem „geht“ alles
- Braucht man eigentlich mehr?

Routing im Internet (Ausblick)

- Mit der Einführung von QOS wird eine bessere Lösung benötigt
- 2 Aufgaben
 - Einhalten von Vorgaben einer QOS-Session (end-to-end-delay, Bandbreite, Paketverlust)
 - Globale Optimierung der Netzwerkressourcen
- 2 Im folgenden Vorstellung einer möglichen Lösung
- Siehe auch RFC der Network Working Group

AntNet – Adaptive Agent Based Routing

- In Intervallen werden von jedem Knoten Agenten (forward Ant) zu einem zufälligen Zielknoten geschickt
- Jeder besuchte Knoten mit Zeit gemerkt
- Jeder Hop (Wechsel von einem zum nächsten Knoten) anhand der Routing-Tabelle des aktuellen Knotens

AntNet – Adaptive Agent Based Routing(2)

- Dabei zufällig auch eine nicht optimale Entscheidung möglich (Entdeckungswahrscheinlichkeit)
- Wenn optimaler ausgewählter Knoten schon besucht dann Zufallsauswahl
- Wird ein Kreis festgestellt wird der gegangene Weg ab dort gelöscht

AntNet – Adaptive Agent Based Routing(3)

- Ziel erreicht -> Entsendung eines Agenten in Gegenrichtung (backward Ant)
- Klärt auf dem Weg alle besuchten Knoten über die Kosten zu seinem Ursprung auf
- Knoten enthält 2 Arten von Informationen
 - Routing-Tabelle (Paar Nachbar-Ziel mit Güte)
 - Mittelwerte zur Entfernung (Zeit) aller Knoten (Gedächtnis des Netzwerkzustands für diesen Knoten)

AntNet – Adaptive Agent Based Routing(4)

- Wahrscheinlichkeit des besuchten Pfades inkrementiert (abhängig von Güte), andere dekrementiert
- Laufzeit einziges aber gutes Kriterium
 - Aussagekräftig für physische Entfernung (Anzahl der Hops, Verarbeitungsgeschwindigkeit besuchter Knoten, Bandbreite)
 - Und für den Netzwerkzustand (Wartezeit)

AntNet – Adaptive Agent Based Routing(5)

- Forward Ants bewegen sich nach den gleichen Kriterien wie Datenpakete
 - Evt. Wartezeit wird auch in der Wegdauer berücksichtigt -> geringere Chance, bester Weg zu werden
 - Rückpropagierung erfolgt später

AntNet – Adaptive Agent Based Routing(6)

- Nur 2 Parameter :
 - Generierungsintervall (Verhältnis von Agenten zu Datenpaketen)
 - Entdeckungswahrscheinlichkeit (Wahrscheinlichkeit, sich im Einzelfall für einen nicht optimalen Weg zu entscheiden)

Realisierung im Praktikum

- Hinzufügen eines weiteren Parameters
- TimeToLive soll die Transitionsphase (Zeit bis der Algorithmus konkurrenzfähige Wege findet) verkürzen
- Setzt Wissen über die zu erwartende maximale Entfernung (Netzwerkgröße) voraus
- Verhindert völlig ineffiziente Routen



DEMO

Bestwegsuche im
Vergleich zu Dijkstra



Erläuterung

- Ist Entfernung das einzige Kriterium und ändert es sich nie dann ist Dijkstra per Definition optimal
- Bei AntNet grundsätzlich lange Transitionsphase
- Hier also nur Beweis für grundsätzliche Tauglichkeit zur Bestwegsuche, nicht Effizienz

DEMO

Einfügen des
Parameters TimeToLive



Erläuterung

- Kein wirklich sinnvoller Parameter
 - Viel zu hoch für kürzere Wege (die allermeisten)
 - Greift wegen „Gedächtnisverlust“ sowieso sehr selten

Dorigos Tests

- AntNet mit verschiedenen anderen Routing-
Algorithmen verglichen u.a. :
 - OSPF
 - SPF_1F (propagiert nur an direkte Nachbarn)
 - Daemon (optimaler Algorithmus, der jederzeit den Status des gesamten Netzwerks in Echtzeit kennt und deswegen „echte“ Kosten berechnen kann) -
> Vergleichswert

Dorigos Tests (2)

- Kriterien :
 - Durchsatz (bit/s)
 - Average Delay (s)
 - Kapazitätsnutzung (global:
gesamt_verfügbare/tatsächlich genutzte)

Dorigos Tests (3)

- 2 Szenarien :
 - CBR (Alle Sessions starten und enden gleichzeitig mit der Anwendung)
 - VBR (zeitlich und örtlich irreguläre Datenflüsse)

Dorigos Tests (4)

- Resultat
 - AntNet und SPF_1F die besten „realistischen“ Algorithmen
 - AntNet deutlich im Vorteil bei CBR
 - SPF_1F gleich bis leicht im Vorteil bei VBR
 - AntNet nahezu gleiche Ergebnisse wie Daemon in einem Drittel der Fälle (Messgenauigkeit)

Fazit



AntNet starker Mitbewerber um den Routing-Algorithmus des zukünftigen Internet

Fragen ???

