

# dezentrale dynamische Optimierung autonomer Systeme

Willi Engel

HU Berlin

02.05.2006

# Inhalt

- 1 Charakterisierung dezentralisierter autonomer System
- 2 Optimierung
- 3 Beispiele
- 4 Zusammenfassung

## autonome Systeme

### Definition

Erreichen eines gegebenen Auftrags durch selbständiges Treffen von Entscheidungen

## autonome Systeme

### Definition

Erreichen eines gegebenen Auftrags durch selbständiges Treffen von Entscheidungen

### Beispiele

- die Roboter der Mars Pathfinder Expedition
- die RoboCup- Teilnehmer.

## dezentralisierte Systeme

### Definition

einzelne kommunizierende Module bewältigen gemeinsam eine Anwendung

## dezentralisiert/verteilt

# dezentralisierte Systeme

## Definition

einzelne kommunizierende Module bewältigen gemeinsam eine Anwendung

## Beispiele

- SETI@home

## dynamische Systeme

### Definition

Parameter (besonders Eingangsparameter) ändern sich in zeitlicher Abhängigkeit

# dynamisch

## dynamische Systeme

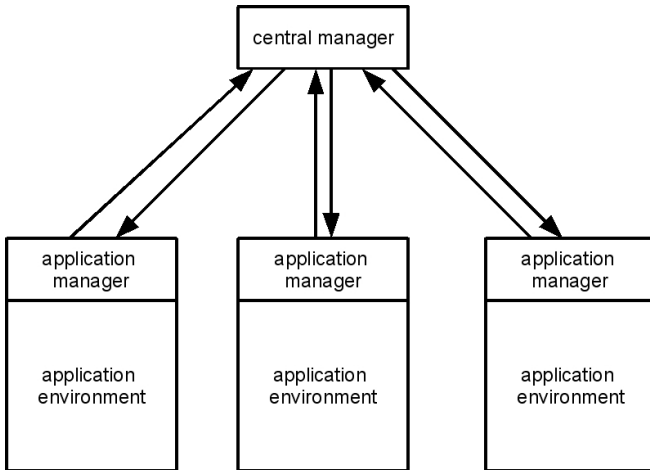
### Definition

Parameter (besonders Eingangsparameter) ändern sich in zeitlicher Abhängigkeit

### Beispiele

- alle Mess-/ Regelsysteme

# Systemmodell



## allgemeiner Ansatz

Minimierung einer Kostenfunktion:

$$h_{opt} = \min_{x \in \mathfrak{R}} h(f_1(x_1) \dots f_n(x_n))$$

# Vorgehen

- Minimierung der Kostenfunktion für jedes application environment
- Minimierung der Gesamtkostenfunktion für das System

# Kostenfunktion für $AE_i$

## lokale Kostenfunktion

$$g_i(r_i, u_i) = \min_{x_i} f_i(x_i, r_i, u_i)$$

mit  $x_i \in C_i(r_i, u_i)$

$x_i$  veränderbare Variablen in AE

$u_i$  äußere Parameter

$r_i$  zugeteilte Ressourcen

# Minimierung der Gesamtkostenfunktion

## Gesamtkostenfunktion

$$h_d = \min_{r_i} h(g_1(r_1, u_1), \dots, g_n(r_n, u_n))$$

mit  $(r_1 \dots r_n) \in R$

## dezentrale vs. zentrale Optimierung

dezentral=zentral ?

$$h_d = \min_{r_i} h(g_1(r_1, u_1), \dots, g_n(r_n, u_n))$$

$$g_i(r_i, u_i) = \min_{x_i} f_i(x_i, r_i, u_i)$$
$$=$$

$$h_{\text{zentral}} = \min_{x_i, r_i} h(f_1(x_1, r_1, u_1) \dots f_n(x_n, r_n, u_n))$$

# dezentrale vs. zentrale Optimierung

dezentral=zentral ?

$$h_d = \min_{r_i} h(g_1(r_1, u_1), \dots, g_n(r_n, u_n))$$

$$g_i(r_i, u_i) = \min_{x_i} f_i(x_i, r_i, u_i)$$
$$=$$

$$h_{\text{zentral}} = \min_{x_i, r_i} h(f_1(x_1, r_1, u_1) \dots f_n(x_n, r_n, u_n))$$

Resultat

**dezentral=zentral !**

wenn  $h$  ordnungserhaltend ist

# Ablauf des Optimierungsprozesses in statischer Umgebung

- 1  $AE_i$  erhält Ressourcen ( $r_i$ ) vom CM
- 2 äußeren Parameter ( $u_i$ ) werden gelesen vom  $AE_i$  oder durch CM übermittelt
- 3  $AE_i$  berechnet die lokale Kostenfunktion ( $g_i(x_i, r_i)$ )
- 4 CM berechnet die Gesamtkostenfunktion  $h_d$

# Erweiterung des statischen Verfahrens

- $u_i$  stellt die Dynamik in dem System dar
- Berechnung der Kostenfunktionen zu bestimmten Zeiten:  
 $t_l, l = 0, 1 \dots$  (mit  $t_1 < t_2 < \dots$ )
- Verwendung der Messwerte aus  $t_{k-1}$

# Probleme der Dynamik

- ausreichende Größe des Intervalls um stabiler Zustand des Systems zu sichern (begrenzte Funktionen)
- Instabilität und Chaos möglich
- Berücksichtigung bei Umsetzung

# IBM - Serviceprovider

## Systemkonfiguration

- $M$  heterogenen Servern ( $S_1, \dots, S_M$ ) und einer Menge von Routern
- $N$  client application environments ( $E_1, \dots, E_N$ ) auf den Servern
- Ziel: Erfüllung von QoS- Bedingungen basierend auf Antwortzeiten
- Ein AE besteht aus einem Router und mehreren Servern
- Anfragen werden durch den Router nach AM Vorgaben an die Server geleitet

# Allgemeine Kostengleichung

## Kostenfunktion basierend auf Antwortzeiten

$$f_i(x_i, r_i, u_i) = f_i(ET_i(x_i, r_i, u_i))$$

- $r_i$  Menge der Server und der Router vom CM an  $AE_i$  zugewiesen
- $x_i$  Menge der Router und Schedulingvariablen (pro Server) (änderbar in  $AE_i$ )
- $u_i$  Menge der Charakteristiken  $AE_i$  (Workload)
- $R$  Menge aller möglichen N- Wege Partitionen der Server  $S_1, \dots, S_M$  mit entsprechendem Router

$ET_i(x_i, r_i, u_i)$  Erwartete Antwortzeit

# Gesamtkostenfunktion

## Gesamtkostenfunktion allgemein

$$h(g_1(r_1, u_1), \dots, g_1(r_N, u_N)) = \sum_{i=0}^N \hat{H}_i g_i(r_i, u_i)$$

$\hat{H}_i$ ; Haltekosten pro Zeiteinheit für  $AE_i$

# Gesamtkostenfunktion

## Gesamtkostenfunktion allgemein

$$h(g_1(r_1, u_1), \dots, g_1(r_N, u_N)) = \sum_{i=0}^N \hat{H}_i g_i(r_i, u_i)$$

$\hat{H}_i$ : Haltekosten pro Zeiteinheit für  $AE_i$

## Gesamtkostenfunktion aufgelöst

$$h_d = \min_{(r_1, \dots, r_N)} \sum_{i=1}^N \hat{H}_i \min_i \underbrace{\sum_{S_j \in r_i} H_j \left( \frac{1}{\mu_{i,j}} + \frac{x_{i,j} \alpha_i \beta_i}{\mu_{i,j} - \lambda_i x_{i,j}} \right)}_{g_i(r_i, u_i)} x_{i,j}$$

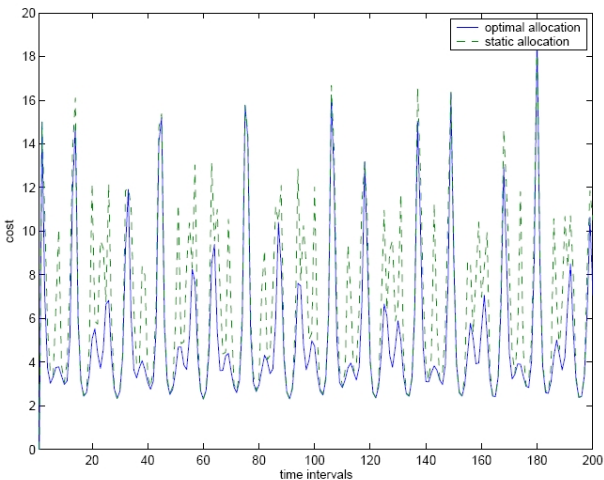
## dynamische Komponente - Problem

- zu geringe Intervallzeiten
- nicht alle Anfragen im Intervall abgearbeitet
- Überhang der Anfragen
- Steigerung durch Wiederholung
- Erliegen des Systems

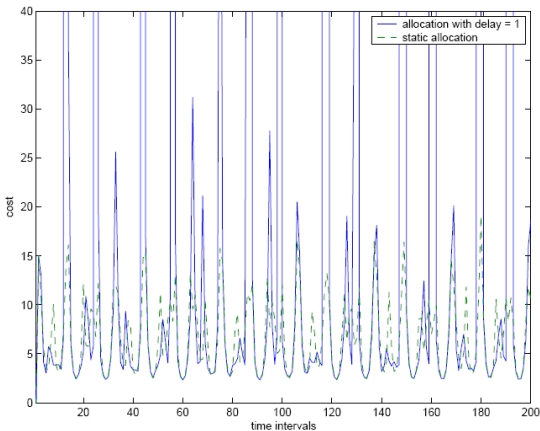
# Simulation

- Simulation mit 2 AE
- Sinusförmige Trafficintensität
- Vergleich dynamische Ressourcenzuteilung zu statischer Ressourcenzuteilung
- dynamische Verteilung (wie bisher behandelt)
- statische Zuteilung auf den durchschnittlich zu erwartenden Traffic optimiert

# Simulationsergebnisse - 1



## Simulationsergebnisse - 2



# Bluetooth

## Aufbau

- 8 Knoten bilden ein piconet (sternförmig)
- 1 Knoten ist master, die anderen slave
- scatternet Bildung über Brückenknoten (master/slave&bridge)
- Ziel: Minimierung der Pfadkosten
- periodische Neuoptimierung des scatternet Topologie

# scatternet Bildung

## Bildung des scatternet

- initial Aufbau eines scatternet
- periodische Rekonfiguration der Knoten und ihrer Umgebung (Anschlußänderung)

## Übergänge

- SS** S löst Anschluss zum aktuellen M, schließt sich anderen M an
- SM** S löst sich von seinem M, erzeugt neues piconet (andere Knoten an sich binden)
- MS** M wird S (Abgabe aller seiner Knoten an andere M)
- MM** Zusammenfassung zweier piconet (ein M übernimmt alle Knoten)

## Optimierung

### Min. der Pfadkosten durch min. Anzahl an Etappen

#### Min. Pfad zw. s und d

$$h_{sd} = \min_{K \in R_{sd}} |K|$$

mit  $|K|$  die Anzahl der Etappen in der Sequenz  $K \in R_{sd}$   
 $R_{sd}$  ist die Menge aller möglichen Pfade zwischen s und d

#### min. Pfade des ges. scatternet

$$F = \sum_{(s,d) \in C} h_{sd}$$

mit  $C$  Menge der Verbindungen auf dem scatternet

$$P : \min F$$

## Vergleich zu IBM

- Veränderter CM (k. Ressourcenzuteilung, nur Übergangspezifikation)
- gleiche AE mit AM (Min. der lokalen Kostenfunktion)
- Kommunikation zw. AE/AM und CM (Topologieänderung)

## Umsetzung/Simulation

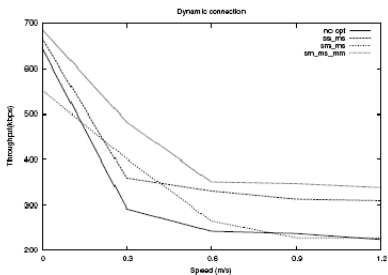
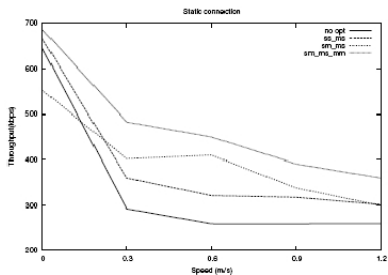
### Realisierung

- Löschen/Erstellen neuer Übergänge
- Vergleich der Etappen vorher und nacher
- mögl. Topologieänderung

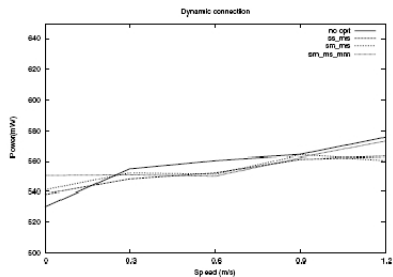
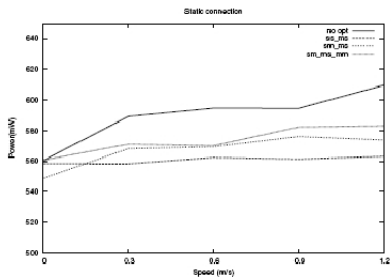
### Simulation

- 50 Knoten verteilt über  $22m \times 22m$
- Zeitraum von 300s
- Knotengeschwindigkeit: 0 m/s bis 1.2 m/s
- statische Verbindungen: Verbindung für ges. Simulationszeit
- dynamische Verbindungen: Verbindung für max. 30 sec
- immer 25 Verbindungen

# Simulationsergebnisse - 1



## Simulationsergebnisse - 2



# Zusammenfassung

- 2 stufige Berechnung einer Kostenfunktion
- beliebige Gestaltung der Kostenfunktion:
  - Realisierung von versch. self\*-Eigenschaften
  - Anpassung an verschiedene Systeme (Systemmodell entsprechend)
- Umsetzung in Serverumgebung bei optimalen Bedingungen gut
- Umsetzung für Bluetoothnetze bei kl. Geschwindigkeiten gut
- Problem der Dynamik

# Fragen

*? Fragen ?*