



Group Formation Among P2P Agents

Alexander Hamann

2006-05-16

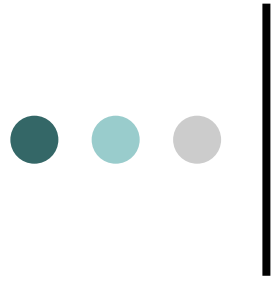
Seminar:

Self-* Properties in Complex Information Systems



Outline

- I. Data Clustering
- II. Clusteringalgorithmen
- III. Dezentralisierte Gruppenbildung



Data Clustering



Data Clustering

disambiguation

○ Clusteranalyse

- Verfahren der Datenanalyse, Finden von Gruppen zusammengehöriger Objekte (*cluster*) in Daten
- Objekte in der selben Gruppe sind gleichartig
- Objekte in jeweils verschiedenen Gruppen sollen so verschieden wie möglich sein



Data Clustering

classification

- Klassifikation gleichartiger Objekte in Gruppen...
 - ... ist eine bedeutende menschliche Aktivität
 - ... hat schon immer eine essentielle Rolle in der Wissenschaft gespielt
 - Biologie (Taxonomie: Gliederung der Organismen)
 - Geologie (System der Mineralien)
 - Marketing (Marktsegmentierung)
 - Geographie, Psychologie, Psychiatrie, ...
 - Mustererkennung, Data Mining, ...



Data Clustering

data and data structure

- multidimensionale Daten
- Aufdecken vorhandener Datenstrukturen
- homogener Datenmenge Struktur aufprägen



Data Clustering

cluster analysis <> discriminant analysis

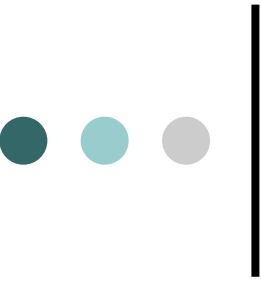
- Clusteranalyse: Gruppen etablieren
- Diskriminantenanalyse: Objekte zu vorher definierten Gruppen zuweisen



Data Clustering

synonyms

- Synonyme für Clusteranalyse
 - unsupervised learning
 - numerical taxonomy
 - vector quantization
 - learning by observation
 - automatic classification
 - typological analysis
 - botryology



Clusteringalgorithmen



Clusteringalgorithmen

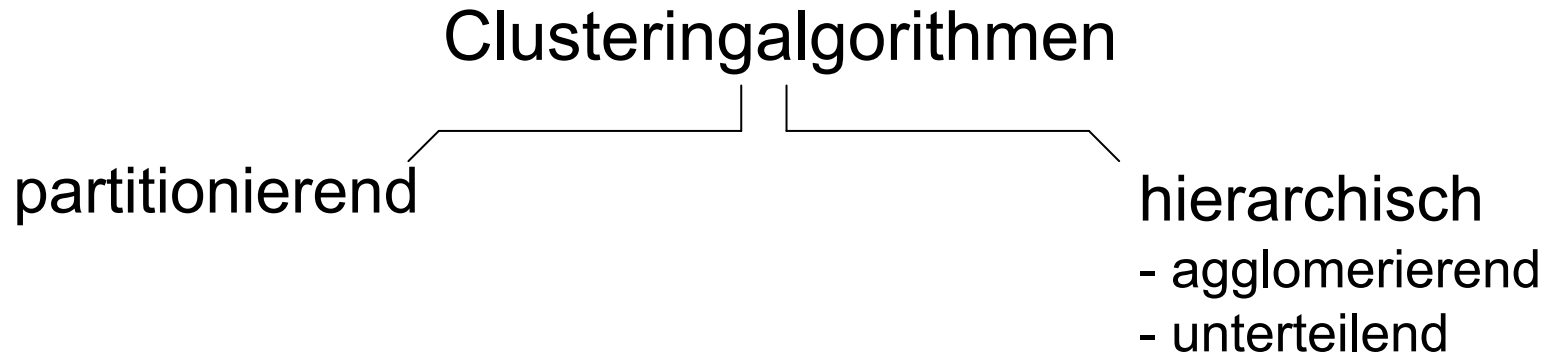
cluster

- Cluster?
 - intuitiv: dichte Region verbundener Punkte, umgeben von einer weniger dichten Region
 - Clusteringalgorithmen enthalten oft implizite Annahmen über die Form eines Clusters



Clusteringalgorithmen

schema





Clusteringalgorithmen

partitional clustering

Clusteringalgorithmen

partitionierend

hierarchisch

- agglomerierend
- unterteilend



Clusteringalgorithmen

partitional clustering

- Partitionierende Algorithmen
 - klassifizieren Daten in k Gruppen
 - jede Gruppe muss mind. ein Objekt enthalten
 - jedes Objekt muss zu genau einer Gruppe gehören

- ➔ Ergebnis: einzelne Partition der Daten



Clusteringalgorithmen

partitional clustering: k-means method

- k-means method (I)
 - MacQueen, 1967
 - Objekte durch Zahlenvektor ($p \times 1$) beschrieben
 - *Centroid*
 - charakterisiert Cluster
 - Punkt im p -dimensionalen Raum
 - Komponenten = Durchschnitte der jeweiligen Komponenten aller Clusterobjekte
 - nicht notwendigerweise Teil der Objekte im Datensatz!



Clusteringalgorithmen

partitional clustering: k-means method

- k-means method (II)

- Maß für Dichte eines Clusters: $E^2(C_v)$

- $$E^2(C_v) = \sum_{i \in C_v} \sum_{f=1}^p (x_{if} - \bar{x}_f(v))^2$$

- Summe der quadrierten Abweichungen von Centroid und Clusterobjekten

- Maß für die Güte des Clusterings: E^2

- $$E^2 = \sum_{v=1}^k E^2(C_v)$$

- Summe der $E^2(C_v)$ des gesamten Clusterings



Clusteringalgorithmen

partitional clustering: k-means method

- k-means method (III)
 - Ziel: $\min E^2$
 - Algorithmus:
 - 1) generiere zufällige Partition mit k Clustern
 - 2) berechne Centroide der Cluster der aktuellen Partition als Seeds
 - 3) weise jedes Objekt demjenigen Cluster zu, dessen Seed am wenigsten weit entfernt liegt
 - 4) vergleiche neue Partition mit der Vorherigen
 - 1) wenn keine Änderung: STOP
 - 2) sonst: GOTO (2)



Clusteringalgorithmen

partitional clustering: k-means method

- k-means method (IV)
 - Anzahl der Cluster bleibt immer erhalten
 - das letzte Objekt eines Clusters fällt mit dem Centroid zusammen
 - kann nicht entfernt werden
 - Qualität der Partition stark von zufälliger Anfangskonfiguration abhängig



Clusteringalgorithmen

hierarchical clustering

Clusteringalgorithmen

partitionierend

hierarchisch

- agglomerierend
- unterteilend



Clusteringalgorithmen

hierarchical clustering

○ Hierarchische Algorithmen

- behandeln alle möglichen Werte für k (Anzahl der Cluster)
- $k=1$: alle Objekte zusammen in einem Cluster
- $k=n$: jedes Objekt bildet eigenen Cluster

➡ Ergebnis: verschachtelte Serie von Partitionen



Clusteringalgorithmen

hierarchical clustering: agglomerative approach

- agglomerierend
 - bottom-up, startet mit n Clustern
 - in jedem Schritt werden 2 Cluster zusammengefasst,
 - ... bis Abbruchkriterium erreicht ist (oder nur noch ein Cluster übrig ist)



Clusteringalgorithmen

hierarchical clustering: divisive approach

- unterteilend
 - top-down, startet mit einem Cluster
 - in jedem Schritt wird ein Cluster geteilt,
 - ... bis Abbruchkriterium erreicht ist
(oder nur einelementige Cluster übrig sind)



Clusteringalgorithmen

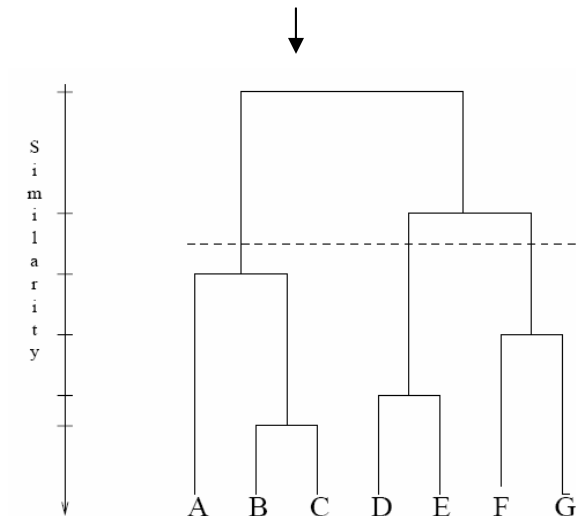
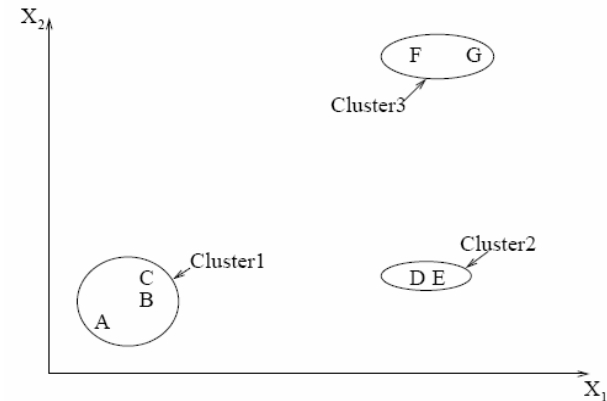
hierarchical clustering: agglomerative vs divisive

- Resultate beider Ansätze unterscheiden sich i. A.

Clusteringalgorithmen

hierarchical clustering: dendrogram

- *Dendrogramm*
 - dendros (griech.) = Baum
 - Visualisierung einer hierarchischen Clusterstruktur
 - zeigt, welche Cluster in welchem Schritt zusammengefasst wurden





Clusteringalgorithmen

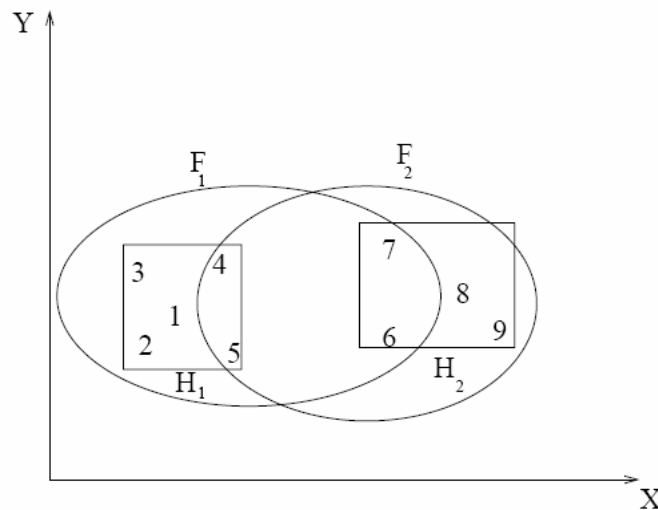
hierarchical clustering: distances

- Maß der Unterschiedlichkeit: „Distanz“
- ... von einzelnen Objekten
 - typischerweise: euklidische Distanz
- ... von Clustern
 - graph metrics (Distanz: Abstand bestimmter Objekte)
 - single linkage
 - complete linkage
 - average linkage
 - geometric metrics (Distanz: Abstand der cluster center)
 - centroid
 - median
 - minimum variance

Clusteringalgorithmen

monothetic vs polythetic / hard vs fuzzy

- Clusteringalgorithmen lassen sich auch in andere Klassen unterteilen
 - monothetisch
 - polythetisch
 - hard
 - fuzzy

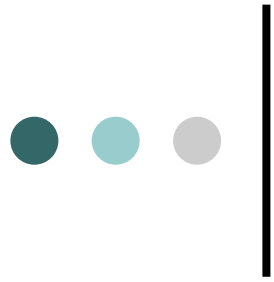




Clusteringalgorithmen

conclusion

- Clusteringalgorithmen traditionell zentral
- es gibt i. A. keinen „besten“ Algorithmus, Wahl abhängig von den Eigenheiten der zugrunde liegenden Daten
- Algorithmen benutzen Parameter, um das Konzept des Clusters abzubilden
 - Zielanzahl der Cluster / Clustergröße
 - minimaler Abstand zwischen Clustern
 - minimale Dichte



Dezentralisierte Gruppenbildung



Dezentralisierte Gruppenbildung

background

- Gruppenbildung anhand ähnlicher Zielsetzungen/Daten
- ➔ Clustering-Problem
- Peer-To-Peer Multi-Agenten-System
- dezentral

- Vorteil der Einteilung in Gruppen
 - Alternative zu zentralen Verzeichnisdiensten
 - Vorstufe zu kooperativer Arbeit



Dezentralisierte Gruppenbildung

background

- Clustering als P2P-Suchproblem: Agenten suchen ihnen ähnliche Agenten
- hier:
 - Agenten mit geringen Aktionsmöglichkeiten
 - Agenten mit stark lokaler Sicht (nur wenige Nachbarn bekannt)



Dezentralisierte Gruppenbildung

general model

○ Modell (I)

- jeder Agent hat eine kleine Anzahl an *links* zu anderen Agenten (Nachbarschaft)
- Agenten eines Clusters poolen ihre links
- Agenten wählen individuell links zu „ähnlichen“ Agenten, formen daraus *matched links*



Dezentralisierte Gruppenbildung

general model

○ Modell (II)

- Cluster wählen die besten matched links ihrer Agenten aus und formen daraus *connected links*
- durch connections verbundene Agenten bilden Cluster



Dezentralisierte Gruppenbildung

general model

○ Modell (III)

- Limit der Clustergröße verhindert Bildung eines einzigen, riesigen Clusters
- Cluster können schwächere connections trennen, um verfügbare stärkere matched links in connections umzuwandeln



Dezentralisierte Gruppenbildung

general model

○ Modell (IV)

- jeder Agent hat ein *main attribute*
 - beschreibt die Eigenschaften des Agenten
 - Gegenstand des Clusterings
- jeder Agent hat mehrere *objectives*
 - derzeitige Zielsetzungen, basierend auf dem main attribute
- Suche nach ähnlichen Agenten über objectives



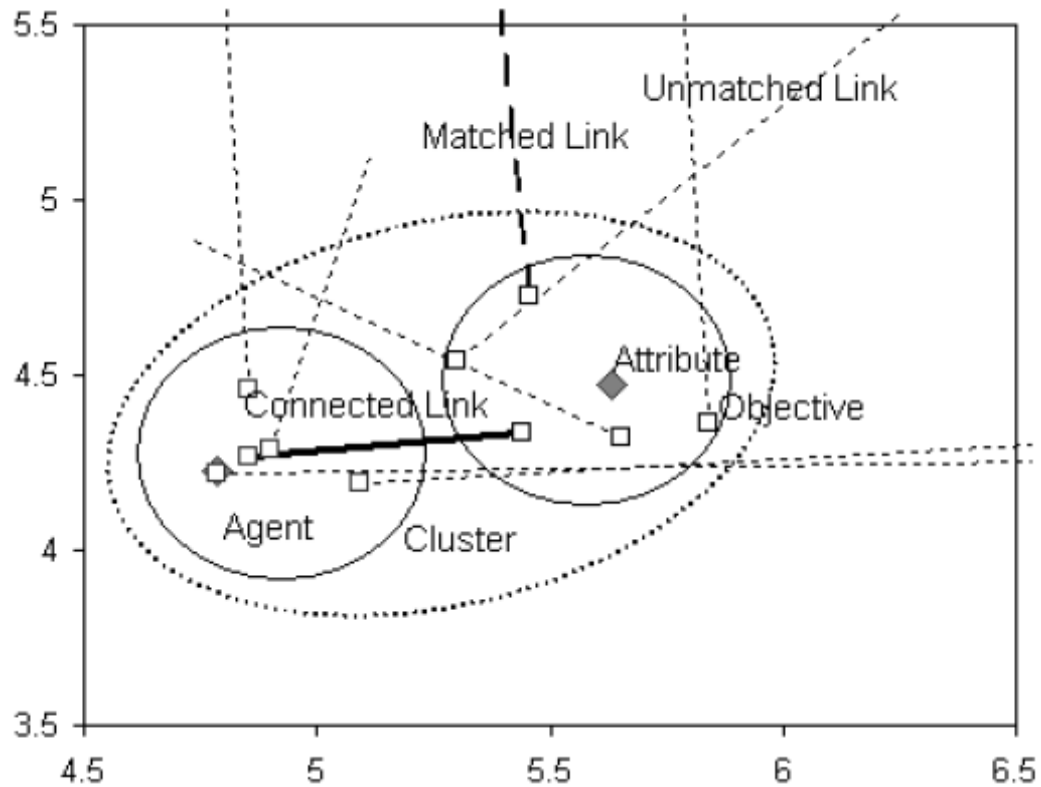
Dezentralisierte Gruppenbildung

general model

- Modell (V)
 - Agenten bestimmen matched links
 - mit Hilfe „passender“ objectives
 - Cluster bestimmen connections
 - können „Nähe“ der Agenten auf dem Attributlevel besser einschätzen

Dezentralisierte Gruppenbildung

general model





Dezentralisierte Gruppenbildung

simulation

○ Simulation (I)

● Initialzustand

- *unmatched links*
 - objectives eines Agenten werden randomisiert mit objectives der anderen Agenten assoziiert
 - jede objective ist mit genau einer anderen objective gepaart
- matched links
 - \emptyset
- connected links
 - \emptyset



Dezentralisierte Gruppenbildung

simulation

- Simulation (II)
 - Algorithmus
 - 1) Connecting
 - 2) Mixing
 - 3) Matching
 - 4) Breaking



Dezentralisierte Gruppenbildung

simulation

○ Simulation (III)

● Algorithmus

1) Connecting

- Cluster wählen die stärksten matched links und bauen sie zu connections aus

2) Mixing

3) Matching

4) Breaking



Dezentralisierte Gruppenbildung

simulation

○ Simulation (IV)

● Algorithmus

1) Connecting

2) Mixing

- objectives die adjazent zu unmatched links sind, werden permutiert

- neue Menge an unmatched links entsteht

3) Matching

4) Breaking



Dezentralisierte Gruppenbildung

simulation

○ Simulation (V)

● Algorithmus

1) Connecting

2) Mixing

3) Matching

- Agenten testen ihre unmatched links
- matched links entstehen mit rundenabhängiger Wahrscheinlichkeit p_t

4) Breaking



Dezentralisierte Gruppenbildung

simulation

○ Simulation (VI)

● Algorithmus

1) Connecting

2) Mixing

3) Matching

4) Breaking

- Cluster wählen mit Wahrscheinlichkeit p_b matched links aus und stufen sie zu unmatched links herab
- wird ein connected link getrennt, so entsteht u. U. ein neuer Cluster



Dezentralisierte Gruppenbildung

simulation

- Simulation (VII)
 - matching probability
 - $p_t(\alpha, \alpha') = 1 - d(\alpha, \alpha') / R_t$
 - je weiter objectives entfernt sind, desto geringer die Chance auf einen matched link
 - breaking probability
 - $p_b(C) = \lambda * N_c / L$
 - λ ... Speed-Faktor
 - je näher der Cluster C am Limit L ist, desto höher die Wahrscheinlichkeit, dass matched links getrennt werden



Dezentralisierte Gruppenbildung

simulation

- Simulation (VIII)
 - 4 Datensätze
 - 5x5, 10x10, 20x20, 40x40 Cluster
 - generiert, kreisförmig
 - auf Raster angeordnet
 - Agenten pro Cluster: 100
 - Gütemaß der Clusterings: E^2
 - Achtung: minimales E^2 führt nicht zwingend zu geeignetstem Clustering!



Dezentralisierte Gruppenbildung

experimental results

- Simulationsergebnisse (I)
 - E^2 anfangs, wenn Cluster weitläufig sind, hoch
 - E^2 konvergiert sehr schnell in die Nähe des optimalen Werts
 - Auslaufverhalten: nach Phase rascher Konvergenz graduelle Verbesserung bis zum Gleichgewicht



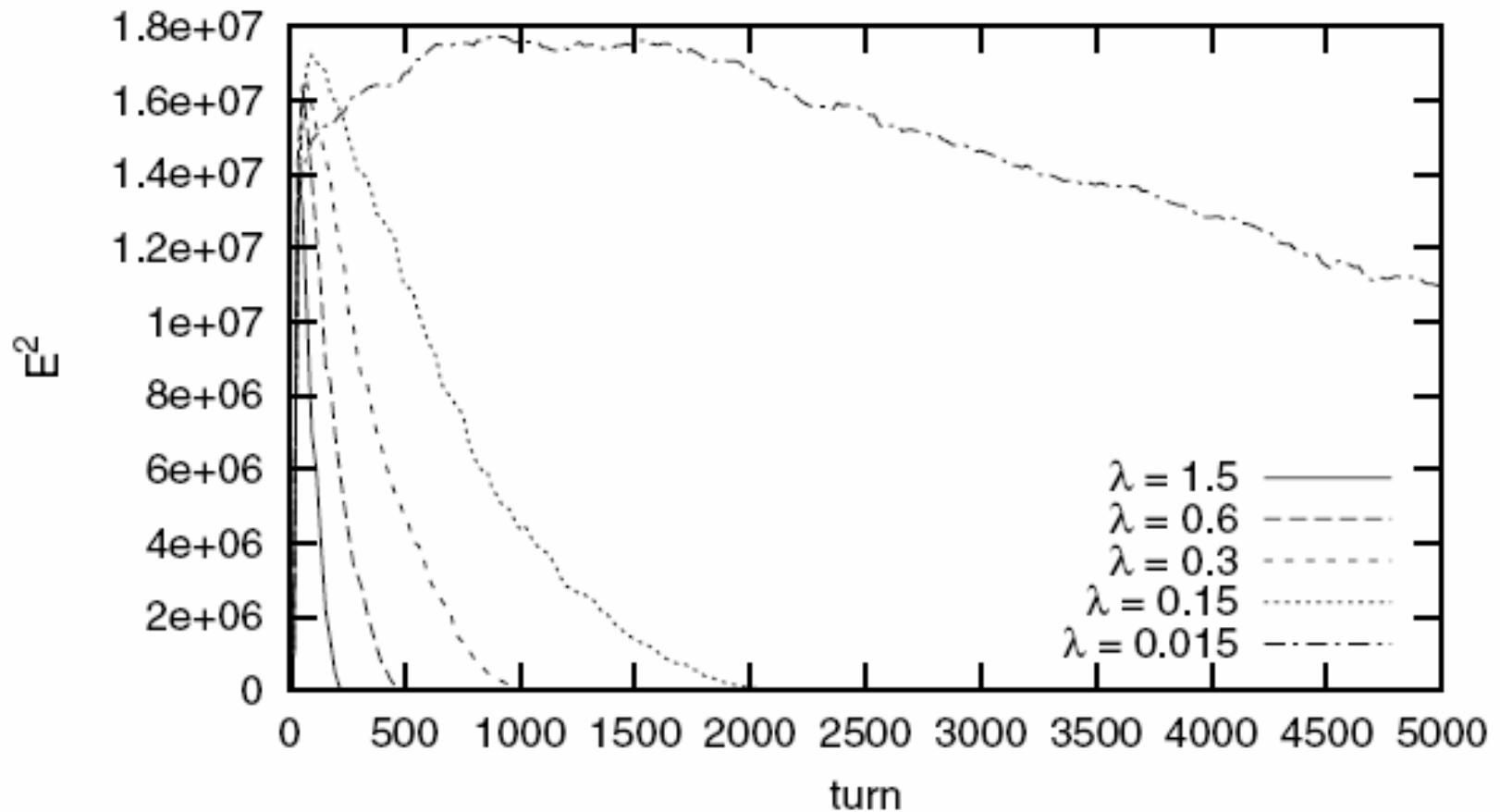
Dezentralisierte Gruppenbildung

experimental results

- Simulationsergebnisse (II)
 - hoher Speed-Faktor λ
 - raschere Konvergenz
 - $\lambda = 1,5$ minimiert E^2
 - niedriger Speed-Faktor λ
 - höhere Stabilität des Gleichgewichts in der Auslaufphase
 - $\lambda = 0,15$ liefert höchste Gleichgewichtsstabilität und qualitativ bestes Clustering

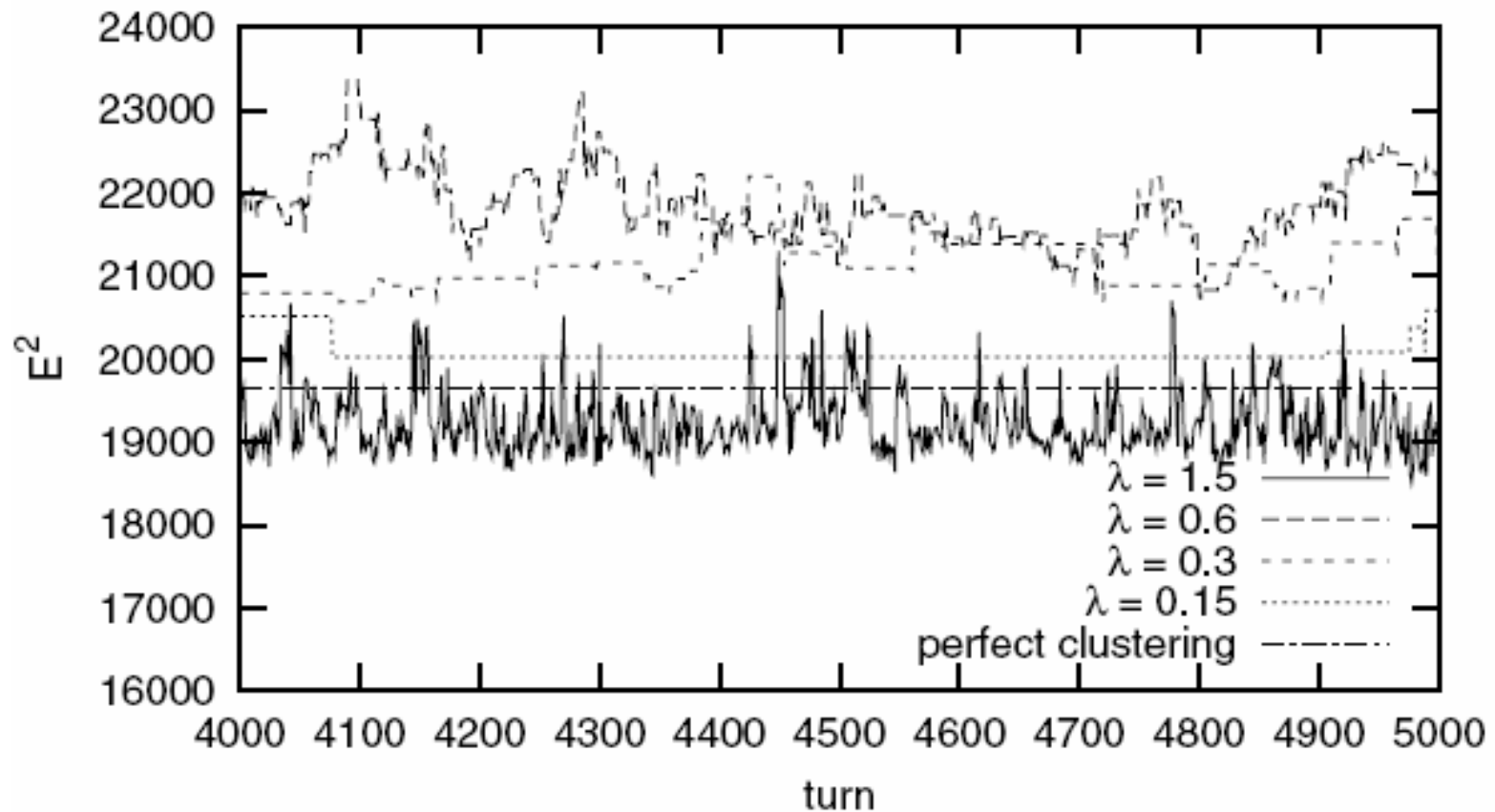
Dezentralisierte Gruppenbildung

experimental results



Dezentralisierte Gruppenbildung

experimental results





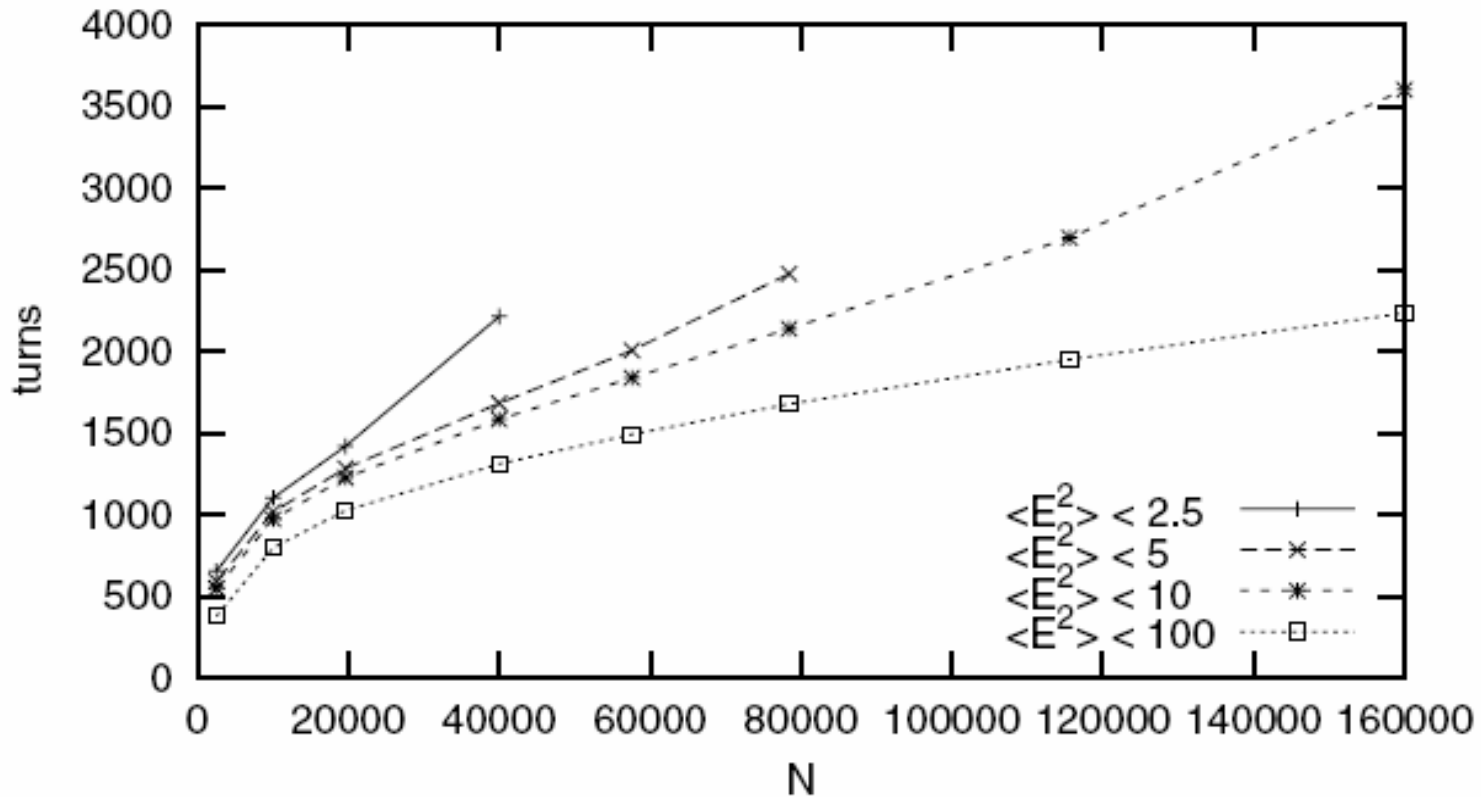
Dezentralisierte Gruppenbildung

experimental results

- Simulationsergebnisse (III)
 - Vergrößerung des Systems
 - Konvergenz bleibt hoch
 - Auslaufphase wird länger, d.h. bessere Lösungen dauern immer länger
 - Kosten, ein bestimmtes Qualitätslevel zu erreichen wachsen aber sublinear mit der Systemgröße!
 - zum Vergleich: k-means hat $O(n)$
 - Qualität
 - kleinere Systeme: 99% richtig gruppiert
 - großes System: ~95% richtig gruppiert

Dezentralisierte Gruppenbildung

experimental results



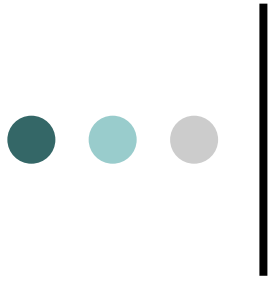


Dezentralisierte Gruppenbildung

conclusion

○ Fazit

- dezentrale Agentensysteme sind in der Lage
 - mit vernünftigem Zeitaufwand,
 - in überraschend guter Qualität
 - große Datenmengen zu clustern!



Finis



Quellen

- [1] **Ogston, Elth et al.:**
A Method for Decentralized Clustering in Large Multi-Agent Systems.
Second International Joint Conference on Autonomous Agents and Multi-Agent Systems, 2003.

- [2] Kaufman, Leonard / Rousseeuw, Peter J.:
Finding Groups in Data. An Introduction to Cluster Analysis.
John Wiley & Sons, Inc., 1990.

- [3] Jain, A.K. / Murty, M.N. / Flynn, P.J.:
Data Clustering: A Review.
In: ACM Computing Surveys, Vol. 31, No. 3, 1999, S. 264-322.

- [4] Olson, Clark F.:
Parallel Algorithms for Hierarchical Clustering.
In: Parallel Computing, Vol. 21, 1995, S. 1313-1325.