FAKULTÄT FÜR INFORMATIK
UNIVERSITÄT KARLSRUHE

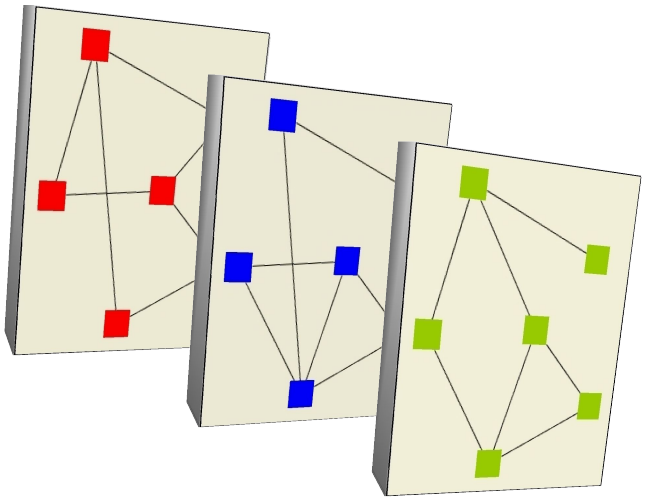
Zeitexpandiertes Graphenclustern - Modellierung und Experimente

Diplomarbeit

von

Dieter Glaser

(d.glaser@gmx.de)



Februar 2008

Ausgeführt unter der Leitung von Robert Görke

Referenten/Betreuer:
Prof. Dr. Dorothea Wagner
Robert Görke
Fakultät für Informatik
Universität Karlsruhe

Inhaltsverzeichnis

1. Einleitung **1**
1.1. Motivation . . . 1
1.2. Related Work . . . 2
1.3. Ziele . . . 4
1.4. Aufbau der Arbeit . . . 4

2. Grundlagen und Definitionen **5**
2.1. Begriffe und Definitionen zu Graphen . . . 5
2.2. Zeitexpandierte Graphen . . . 7
2.2.1. Gewichteter zeitexpandierter Graph mit Schwelle p . . . 11
2.3. Clusterung . . . 11
2.3.1. Dichte einer Clusterung . . . 12
2.4. Indizes zur Bewertung von Clusterungen . . . 12
2.4.1. Coverage . . . 12
2.4.2. Performance . . . 13
2.4.3. Conductance . . . 14
2.4.4. Modularity . . . 17
2.5. Cluster-Verfahren . . . 18
2.5.1. Greedy-Significance-Clustering . . . 18
2.5.2. Iterative-Conductance-Cutting . . . 19
2.5.3. Markov-Clustering . . . 19
2.6. Cosine-Similarity . . . 21
2.6.1. Adapted-Cosine-Similarity . . . 21
2.7. Cosine Similarity Matrix . . . 23
2.8. Vergleichsmaße für Clusterungen . . . 23
2.8.1. Das Vergleichsmaß *bestmatch* . . . 23
2.8.2. Van Dongen . . . 25
2.8.3. Cosine-comparison . . . 26
2.8.4. Variation of Information . . . 27

3. Entwurf von zeitexpandierten Graphen **29**
3.1. Das E-Mail-Netzwerk der Fakultät für Informatik . . . 29
3.2. Ablauf . . . 30
3.2.1. Design . . . 31
3.2.2. Analyse . . . 34
3.2.3. Implementierung . . . 35
3.2.4. Experimente . . . 35
3.3. Ziele . . . 35
3.4. Beispiele . . . 35
3.4.1. Spaltung einer Gruppe . . . 36
3.4.2. Umbruch einer Gruppe . . . 36
3.4.3. Zeitlich begrenztes Abweichen von der Norm . . . 36

4. Testreihe mit der Methode Normal **39**
4.1. Design . . . 39
4.2. Analyse . . . 39
4.3. Experimente . . . 41
4.3.1. Überprüfung der Referenz-Clusterung . . . 41
4.3.2. Greedy-Significance-Clustering . . . 42
4.3.3. Iterative-Conductance-Cutting . . . 46
4.3.4. Markov-Clustering . . . 50
4.4. Fazit . . . 51

5. Testreihe mit der Methode Normed **57**
5.1. Design . . . 58
5.2. Analyse . . . 58
5.3. Experimente . . . 58
5.3.1. Überprüfung der Referenz-Clusterung . . . 59
5.3.2. Greedy-Significance-Clustering . . . 59
5.3.3. Iterative-Conductance-Cutting . . . 61
5.4. Fazit . . . 63

6. Testreihe mit der Methode Cosine-Time **65**
6.1. Design . . . 65
6.2. Analyse . . . 66
6.3. Experimente . . . 66
6.3.1. Überprüfung der Referenz-Clusterung . . . 66
6.3.2. Greedy-Significance-Clustering . . . 67
6.3.3. Iterative-Conductance-Cutting . . . 71
6.4. Fazit . . . 72

7. Bewertung der Ergebnisse **75**
7.1. Interpretation der Ergebnisse . . . 75
7.2. Vergleich zur zeitlich flachen Clusterung . . . 78
7.3. Anwendung auf die Beispiele . . . 82
7.4. Fazit der Testreihen . . . 86

8. Zusammenfassung und Ausblick **89**
8.1. Zusammenfassung . . . 89
8.2. Ausblick . . . 91

A. Anhang **98**
A.1. Abbildungen . . . 98
A.2. Tabellen . . . 112

1. Einleitung

1.1. Motivation

Mit Hilfe von Graphen lassen sich viele Netzwerke beschreiben. So können soziale Netzwerke, Co-Autoren- oder Kollaborations-Netzwerke, Routing-Netzwerke, Verkehrs- oder Transport-Netzwerke als Graphen dargestellt werden. Diese Netzwerke sind nicht statisch und starr, sondern ändern sich mit der Zeit. Solche veränderlichen Netzwerke kann man daher als dynamische Graphen ansehen. Dabei beschreibt der dynamische Graph das zugrundeliegende Netzwerk in Abhängigkeit von der Zeit. Wir nennen den dynamischen Graphen für einen bestimmten Zeitpunkt t die Ausprägung des dynamischen Graphen zum Zeitpunkt t.

Zum Auffinden natürlicher Gruppen innerhalb von Graphen benutzt man zur Partitionierung der Knotenmenge die unterschiedlichsten Cluster-Verfahren. Das Ergebnis dieser Cluster-Verfahren ist eine Clusterung, das heißt, eine Partitionierung der Knotenmenge des Graphen in paarweise disjunkte Teilmengen, sogenannte Cluster. Untersucht man die zeitliche Entwicklung der Gruppierungen innerhalb eines dynamischen Graphen, so steht man vor dem Problem, dass bereits kleine Veränderungen innerhalb eines Graphen zu großen Veränderungen der gefundenen Clusterungen führen können. Dies erschwert die Interpretierbarkeit und Verständlichkeit der Clusterungen der unterschiedlichen Zeitpunkte für den Nutzer. Die Beantwortung der Frage, welche Cluster zweier aufeinanderfolgenden Clusterungen auseinander hervorgehen, ist nicht trivial. Entscheidend für die Erfassung der zeitlichen Entwicklung der verschiedenen Gruppen ist die korrekte Überführung der Clusterungen der aufeinanderfolgenden Zeitschritte. Beim Clustern zeitlich veränderlicher Graphen unterscheiden wir zwei Arten von Cluster-Verfahren, zum einen die *online*-Verfahren, bei denen nur die Daten der vorangehenden Zeitschritte verfügbar sein müssen, zum anderen die *offline*-Verfahren, bei denen schon zu Beginn der Clusterung alle Daten zur Verfügung stehen müssen.

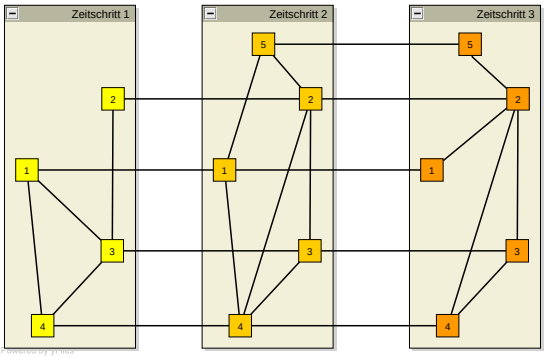


Abbildung 1: *Beispiel für einen zeitexpandierten Graphen mit drei Zeitschritten. Dabei sind die drei Ausprägungen des zugrundeliegenden dynamischen Graphen durch die Kästen repräsentiert.*

In dieser Arbeit beschäftigen wir uns mit einer offline-Methode, dem Clustern von zeitexpandierten Graphen. Diese Idee wurde erstmals in [GGWW06] vorgestellt. Ähnliche Graphen wurden bisher lediglich für die Lösung von Transportproblemen durch Flussalgorithmen genutzt [FF58, FF62]. Zur Erzeugung der zeitexpandierten Graphen werden für eine Folge von Zeitschritten die zugehörigen Ausprägungen eines dynamischen Graphen übereinandergelegt. Die

sich entsprechenden Kopien der Knoten werden über Kanten verbunden. Dadurch ensteht ein zusammenhängender Graph, der die zeitliche Entwicklung des dynamischen Graphen widerspiegelt (siehe Abbildung 1). Die Clusterung dieses zeitexpandierten Graphen liefert uns die Entwicklung von natürlichen Gruppen innerhalb des dynamischen Graphen. Ebenso erhoffen wir uns durch die Clusterung ein besseres Verständnis der Struktur des dynamischen Graphen. Das Clustern von zeitexpandierten Graphen macht die komplizierte Zusammenführung der Clusterungen der verschiedenen Zeitschritte überflüssig.

Ein Beispiel für einen dynamischen Graphen, für den die Erzeugung eines zeitexpandierten Graphen sinnvoll ist, wäre eine Web-Community wie StudiVZ. Die Mitglieder einer Web-Community stehen miteinander durch Mitteilungen oder gegenseitiges Besuchen der persönlichen Seiten in ständigem Kontakt. Die Anzahl der Kontakte eines bestimmten Zeitbereiches könnten dabei die Gewichte der Kanten einer Ausprägung des dynamischen Graphen bilden. Will man die Entwicklung von Gruppen innerhalb der Community untersuchen, wäre die Clusterung eines zeitexpandierten Graphen der Community eine vielversprechende Vorgehensweise.

Unserer Kenntnis nach ist dies der erste Ansatz, um eine Clusterung im offline-Verfahren gleichzeitig für alle Zeitschritte eines dynamischen Graphen zu finden. Wir werden zeigen, dass die Clusterung von zeitexpandierten Graphen ein geeignetes Mittel darstellt, Veränderungen innerhalb eines dynamischen Graphen zu erfassen. Dabei liefert ein Cluster des zeitexpandierten Graphen eine Beschreibung der Entwicklung einer Gruppe über die Zeit hinweg. In dieser Arbeit entwickeln wir zunächst ein Modell für die zeitexpandierten Graphen. Mit Hilfe dieses Modells erzeugen wir anschließend zeitexpandierte Graphen für eine konkrete Anwendung. Durch die Clusterung der erzeugten zeitexpandierten Graphen erhoffen wir uns ein besseres Verständnis für unser Modell.

1.2. Related Work

Hopcroft et al. [HKKS04] versucht natürliche Gruppen innerhalb der NEC CiteSeer Literaturdatenbank zu finden. Der aus einem Datensatz erzeugte Graph enthält als Knoten die einzelnen Paper, die in der Datenbank gespeichert sind. Es gibt eine gerichtete Kante von Paper A zu Paper B, falls Paper A auf Paper B verweist. Um die natürlichen Gruppen zu finden, werden aus jeweils 95 % der Knoten des Graphen n verschiedene Subgraphen gebildet. Auf jedem dieser Subgraphen wird ein hierarchisches Cluster-Verfahren ausgeführt. Alle Cluster der verschiedenen Zwischenschritte der hierarchischen Clusterung des Subgraphen x bilden die Menge $\mathcal{T}_{\mathrm{x}}$. Die Basis für die Suche der natürlichen Gruppen liefert die Menge $\mathcal{T}_1$ des ersten Subgraphen.

Zur Bestimmung der natürlichen Gruppen wird zunächst das Vergleichsmaß $\mathrm{bestmatch}$ definiert (siehe dazu Abschnitt 2.8.1), das für einen Cluster C und eine Clustermenge $\mathcal{T}_{\mathrm{x}}$, den zu C ähnlichsten Cluster aus $\mathcal{T}_{\mathrm{x}}$ bestimmt. Eine natürliche Gruppe ist ein Cluster C aus T_1, der für einen festgelegten Anteil f der n Clustermengen $\mathcal{T}_{\mathrm{x}}$ einen $\mathrm{bestmatch}$ erreicht, der höher ist als ein zuvor festgelegter Schwellenwert p. Mit anderen Worten: ein Cluster aus T_1 ist genau dann eine natürliche Gruppe, wenn er in den meisten der anderen Clustermengen in ähnlicher Form vorkommt. Die Festlegung von T_1 als Basis für alle natürlichen Gruppen erfolgt ohne weitere Begründung.

Es werden zwei Datensätze gebildet. Der 1998-Datensatz enthält alle Daten von 1990 bis 1998, während der 2001-Datensatz alle Daten von 1990 bis 2001 beinhaltet. Für jeden dieser Datensätze

werden die natürlichen Gruppen gesucht. Die Überführung der natürlichen Gruppen der beiden Zeitschritte erfolgt über deren Übereinstimmung anhand der definierten Vergleichsmaße. Dabei wird jedoch aufgrund der Größe der Graphen nur ein kleiner Teil der Cluster untersucht.

Palla et al. [PBV07] stellt ein Clusterverfahren auf Basis der *Clique Percolation Method* CPM vor. Dabei werden die Cluster eines Zeitschrittes mit Hilfe der im Graphen enthaltenen k-Cliquen gefunden. Eine *k-Clique* ist eine Gruppe von k vollständig miteinander verbundenen Knoten. Zwei k-Cliquen sind adjazent, wenn sie $\mathrm{k}-1$ Knoten gemeinsam haben. Daraus ergibt sich eine *k-Cliquen-Kette* aus der Vereinigung einer Folge von adjazenten Cliquen. Sind zwei k-Cliquen Teil der selben k-Cliquen-Kette heißen sie *k-Cliquen-connected*. Der Cluster einer speziellen Clique entsteht bei der CPM aus der Vereinigung der Clique mit allen Cliquen, die zu ihr k-Cliquen-connected sind. Dabei können Knoten mehreren Clustern angehören. Die Überführung der Clusterungen zweier aufeinanderfolgender Zeitschritte wird hier durch ein *joint network* erreicht. Das joint network bildet sich aus der Vereinigung der Kantenmengen der beiden Zeitschritte. Auf diesem joint network wird nun erneut die CPM durchgeführt. Jeder Cluster der beiden Zeitschritte ist in genau einem Cluster des joint networks enthalten. Ist in einem Cluster des joint networks aus beiden Zeitschritten genau ein Cluster enthalten, werden diese einander zugeordnet. Sind in einem Cluster des joint networks mehrere Cluster eines der beiden Zeitschritte enthalten, so werden die Cluster aufgrund ihrer gegenseitigen Überdeckungen zugeordnet.

Beim *Evolutionary Clustering* von Chakrabarti et al. [CKT06] wird ein online-Verfahren vorgestellt, bei dem sich für jeden Zeitschritt t die Bewertung einer Clusterung $\mathrm{C_t}$ aus zwei Teilen zusammensetzt. Aus der *Snapshot Quality* sq, also der Güte bezüglich der $\mathrm{n} \times \mathrm{n}$ Matrix $\mathrm{M_t}$, die die Beziehungen der n verschiedenen Objekte in Zeitschritt t beschreibt, und den *Temporal Costs* ct, die hoch sind, falls sich die Clusterung stark von der vorhergehenden Clusterung $\mathrm{C_{t-1}}$ unterscheidet. Die verwendeten Cluster-Verfahren versuchen eine optimale Clusterung zu finden, indem sie den Term

$$\overbrace{\mathrm{sq}(\mathrm{C_t}, \mathrm{M_t})}^{\mathrm{SnapshotQuality}} - \overbrace{\mathrm{hc}(\mathrm{C_{t-1}}, \mathrm{C_t})}^{\mathrm{TemporalCosts}}$$

maximieren. Mit Hilfe dieser dualen Bewertung gibt es keine ständigen Schwankungen in den Clusterungen der einzelnen Abschnitte. Es wird eine Glättung der Clusterungen der einzelnen Zeitabschnitte erreicht, das heißt, die Clusterungen aufeinanderfolgender Zeitpunkte ähneln einander und erlauben je nach dem Anteil der Temporal Costs mehr oder weniger starke Veränderungen. Eine Zuordnung der Cluster der verschiedenen Zeitschritte erfolgt dabei noch nicht. In [CSZ+07] wird eine spektrale Variante des Evolutionary Clustering eingeführt.

Die bisherigen Ansätze versuchen, die Entwicklung von Gruppen anhand der Clusterung einzelner Zeitschritte zu erfassen. Dabei verwenden sie unterschiedliche Methoden. Bei Palla et al. [PBV07] und Hopcroft et al. [HKKS04] finden die Clusterungen der einzelnen Zeitschritte zunächst unabhängig voneinander statt. Danach werden für die benachbarten Zeitschritte Cluster gesucht, die auseinander hervorgehen. Chakrabarti et al. [CKT06] wendet sich einem anderen Lösungsansatz zu. Die Clusterungen der vorigen Zeitschritte haben einen direkten Einfluss auf das Ergebnis des aktuellen Zeitschrittes. Dennoch liefert keines der erwähnten Verfahren eine vollständige Beschreibung der zeitlichen Entwicklung von Gruppen. Bei unserem Ansatz dagegen

werden alle Zeitschritte des dynamischen Graphen in einem Graphen erfasst. Die Untersuchung der Entwicklung der Gruppen erfolgt direkt durch die Clusterung des zeitexpandierten Graphen.

1.3. Ziele

In dieser Arbeit soll zunächst ein Modell für die zeitexpandierten Graphen aufgestellt werden. Auf Grundlage dieses Modells werden wir zeitexpandierte Graphen für das E-Mail-Netzwerk innerhalb der *Fakultät für Informatik* an der *Universität Karlsruhe (TH)* erzeugen und die Auswirkungen der verschiedenen Parameter und Methoden des Modells untersuchen.

1.4. Aufbau der Arbeit

Im folgenden Kapitel 2 werden grundlegende Begriffe und Definitionen aufgeführt. Auf Basis des in Kapitel 3 vorgestellten Modells und des E-Mail-Netzwerks wird in den darauf folgenden Kapiteln (4, 5 und 6) der mögliche Aufbau eines zeitexpandierten Graphen untersucht. Kapitel 7 bewertet und interpretiert die erreichten Ergebnisse. Den Abschluss bildet Kapitel 8 mit einer kurzen Zusammenfassung und einem Ausblick.

2. Grundlagen und Definitionen

Dieses Kapitel liefert einige grundlegende Definitionen. Gängige Definitionen sind aus der Literatur entnommen.

2.1. Begriffe und Definitionen zu Graphen

Wenn wir im Folgenden von Graphen sprechen, so sind damit stets *einfache Graphen* gemeint, das heißt, innerhalb der Graphen existieren weder *Mehrfachkanten* zwischen zwei Knoten noch *Schlingen*.

Definition 2.1 *Ein* gerichteter Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *ist eine Menge von* Knoten $\mathcal{V}$*, die über eine Menge von* Kanten $\mathcal{E}$ *miteinander verbunden sind. Jede Kante* $\mathrm{e} \in \mathcal{E}$ *lässt sich schreiben durch ein geordnetes Tupel* (v_i, v_j) *mit dem* Startknoten $v_i \in \mathcal{V}$ *und dem* Endknoten $v_j \in \mathcal{V}$*. Die* Anzahl der Knoten *des Graphen bezeichnen wir mit* n*, die* Anzahl der Kanten *mit* m.

Die maximale *Anzahl von Kanten* eines gerichteten Graphen $\mathcal{G}$ ist $\mathrm{m}_{max} = \mathrm{n} \cdot (\mathrm{n}-1)$. Sei $\mathcal{U} \subset \mathcal{V}$ eine Teilmenge aller Knoten, dann ist $\mathcal{E}\ (\mathcal{U})$ die Menge aller Kanten, deren Ziel- und Endknoten in $\mathcal{U}$ enthalten sind. Ein Eintrag $\mathcal{A}_{\mathrm{i,j}}$ der *Adjazenzmatrix* $\mathcal{A} \in \{0,1\}^{\mathrm{n}\times\mathrm{n}}$ eines Graphen $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ hat den Wert 1, falls die Kante $\mathrm{e} = (v_i, v_j)$ in der Menge $\mathcal{E}$ enthalten ist. Existiert keine Kante von Knoten v_i zu Knoten v_j, enthält die Adjazenzmatrix eine 0 an der betreffenden Stelle $\mathcal{A}_{\mathrm{i,j}}$, daraus folgt

$$\mathcal{A} = \begin{pmatrix} A_{1,1} & A_{1,1} & \dots & A_{1,n} \\ A_{2,1} & A_{2,1} & \dots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,1} & \dots & A_{n,n} \end{pmatrix} \text{, wobei}$$

$$A_{\mathrm{i,j}} = \begin{cases} 1 & \text{, falls Kante mit Startknoten } v_i \text{ und Endknoten } v_j \text{ existiert} \\ 0 & \text{, sonst} \end{cases} .$$

Definition 2.2 *Ein* ungerichteter Graph *ist ein Graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$*, dessen Kanten ungerichtet sind. Das heißt*

$$\forall \mathrm{e} \in \mathcal{E} : \mathrm{e} = \{v_\mathrm{i}, v_\mathrm{j}\} \quad .$$

Die Kante $\mathrm{e} = \{v_i, v_j\}$ hat die *Endknoten* v_i und v_j. Die Menge $\mathcal{E}(\mathcal{U})$ enthält alle Kanten, deren Endknoten Elemente der Knotenmenge $\mathcal{U}$ sind. Die maximale Anzahl von Kanten eines ungerichteten Graphen $\mathcal{G}$ ist

$$\mathrm{m}_{max} = \begin{pmatrix} \mathrm{n} \\ 2 \end{pmatrix} = \frac{\mathrm{n} \cdot (\mathrm{n}-1)}{2} \quad .$$

Die Matrix eines ungerichteten Graphen ist symmetrisch und die Einträge der Adjazenzmatrix ergeben sich durch

$$A_{\mathrm{i,j}} = \begin{cases} 1 & \text{, falls Kante zwischen Knoten } v_i \text{ und Knoten } v_j \text{ existiert} \\ 0 & \text{, sonst} \end{cases} .$$

Wenn wir im Folgenden von Graphen sprechen, so sind immer ungerichtete Graphen gemeint.

Definition 2.3 *Unter einem* Subgraphen *eines Graphen $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ versteht man den durch die Knotenmenge $\mathcal{U}$ induzierten Graphen $\mathcal{G}[\mathcal{U}] = (\mathcal{U}, \mathcal{E}(\mathcal{U}))$.*

Definition 2.4 *Die* Nachbarschaft $\mathcal{N}(v_\mathrm{i})$ *eines Knoten v_i setzt sich aus den Knoten zusammen, die über eine Kante mit ihm verbunden sind:*

$$\mathcal{N}(v_\mathrm{i}) = \{v \in \mathcal{V} \mid \exists \mathrm{e} \in \mathcal{E} : \mathrm{e} = \{v_\mathrm{i}, v\}\} \quad .$$

Definition 2.5 *Die Anzahl der Kanten, die zu Knoten v_i inzident sind, heißt der* Grad $g(v_\mathrm{i})$ *von v_i und ergibt sich zu*

$$g(v_\mathrm{i}) = \sum_{k=1}^{n} (A_{\mathrm{i,k}}) \quad .$$

Definition 2.6 *Ein* gewichteter Graph *ist ein Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, dessen Kanten alle mit einer Gewichtung versehen sind. Das heißt, es gibt eine Funktion $\omega : \mathcal{E} \to \mathcal{R}$ mit $\omega : \ \mathrm{e} \mapsto \mathrm{w_e}$ und $\mathrm{w_e} \in \mathcal{R}$.*

Oft entspricht $\mathcal{R}$ dem Intervall $[0, 1]$, aber es sind ebenso andere Wertebereiche möglich, wie z.B. die Menge der reellen Zahlen $\mathbb{R}$. Der Wertebereich $[0, 1]$ ermöglicht einen Vergleich der verschiedenen Gewichte. Wäre der Wertebereich die Menge der natürlichen Zahlen $\mathbb{N}$, ließen sich einzelne Gewichte ohne genauere Kenntnis der Verteilung der Gewichte nur schwer einordnen. Berechnen sich die Gewichte der Kanten aus einer Kostenfunktion oder Ähnlichem, ist dies für die Interpretation der Kanten von Vorteil. Die *gewichtete Adjazenzmatrix* $\mathcal{A}^\omega \in \mathcal{R}^{\mathrm{n}\times\mathrm{n}}$ enthält an der Position $\mathcal{A}^\omega_{\mathrm{i,j}}$ bzw. $\mathcal{A}^\omega_{\mathrm{j,i}}$ das Gewicht der Kante $\mathrm{e} = \{v_i, v_j\}$, falls diese existiert, ansonten eine 0. Mit $\mathrm{v_i(j)}$ bezeichnen wir das Gewicht der Kante von Knoten v_i nach Knoten v_j. Bei ungerichteten Graphen gilt immer $\mathrm{v_i(j)} = \mathrm{v_j(i)}$.

Definition 2.7 *Unter dem* Gewicht eines Knoten $\omega(v_\mathrm{i})$ *versteht man die Summe aller Gewichte der Kanten, die zu Knoten v_i inzident sind. Sei $\mathcal{E}(v_\mathrm{i}) = \{\mathrm{e} \in \mathcal{E} \,|\exists v \in \mathcal{V} : \mathrm{e} = \{v, v_\mathrm{i}\}\}$, dann ist das Gewicht von Knoten v_i gegeben durch*

$$\omega(v_\mathrm{i}) = \sum_{\mathrm{e}\in\mathcal{E}(v_\mathrm{i})} \omega(\mathrm{e}) \quad .$$

Definition 2.8 *Unter der* Dichte $D(\mathcal{G})$ *eines Graphen $\mathcal{G}$ versteht man das Verhältnis von der Anzahl der Kanten m zu der maximalen Kantenanzahl $\mathrm{m_{max}}$:*

$$D(\mathcal{G}) = \frac{\mathrm{m}}{\frac{(\mathrm{n}-1)\mathrm{n}}{2}} = \frac{2\mathrm{m}}{(\mathrm{n}-1)\mathrm{n}} \quad .$$

Analog ergibt sich die Dichte eines Subgraphen $\mathcal{G}[\mathcal{U}] = (\mathcal{U}, \mathcal{E}(\mathcal{U}))$ von $\mathcal{G}$ aus

$$D(\mathcal{G}[\mathcal{U}]) = \frac{|\mathcal{E}(\mathcal{U}|}{\frac{|\mathcal{U}|\cdot(|\mathcal{U}|-1)}{2}} = \frac{2\,|\mathcal{E}(\mathcal{U}|}{|\mathcal{U}|\cdot(|\mathcal{U}|-1)} \quad .$$

Für weiterführende Informationen zur Graphentheorie sei hier auf die Bücher von Diestel [Die06] und Turau [Tur04] verwiesen.

2.2. Zeitexpandierte Graphen

Ein *dynamischer Graph* ist ein Graph, der sich über die Zeit verändert. Der dynamische Graph $\mathcal{G}(\mathrm{t}_0)$ zum Zeitpunkt t_0 heißt die *Ausprägung* des dynamischen Graphen zum Zeitpunkt t_0. Hat man verschiedene Ausprägungen des dynamischen Graphen in chronologischer Reihenfolge gegeben (siehe Abbildung 2), stellen diese die zeitliche Entwicklung des Graphen dar. Verbindet man die einander entsprechenden Knoten der verschiedenen Ausprägungen, entsteht ein *zeitexpandierter Graph* (siehe Abbildung 3a). Mit Hilfe von zeitexpandierten Graphen wollen wir die Veränderungen in der Struktur eines dynamischen Graphen erfassen. Mit den Clustern dieser zeitexpandierten Graphen versuchen wir, zeitliche Trends im Graphen zu erkennen und zu identifizieren. Für die Definition eines zeitexpandierten Graphen führen wir zunächst einige notwendige Begriffe ein.

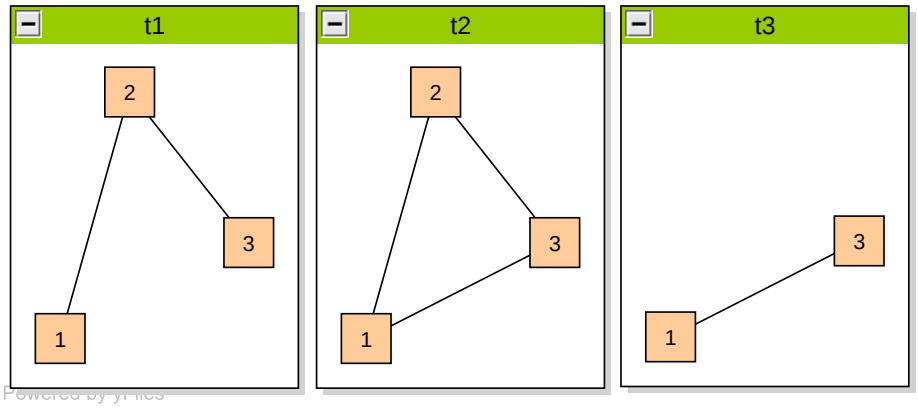
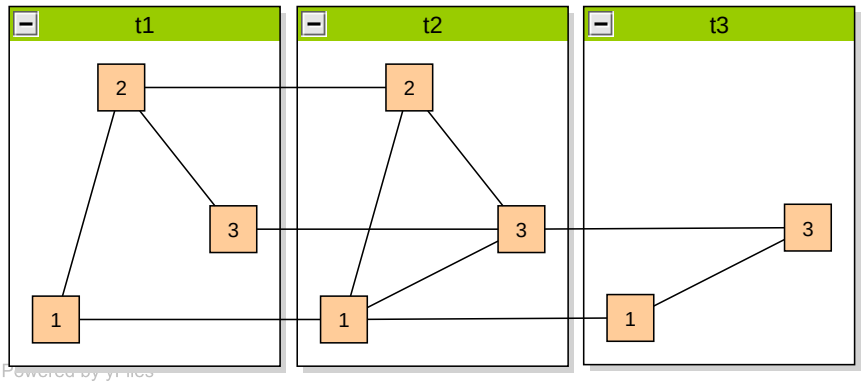
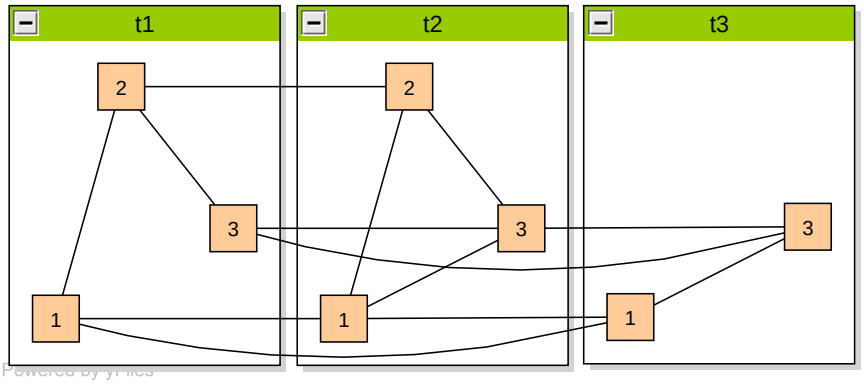


Abbildung 2: *Drei Ausprägungen* $\mathcal{G}(\mathrm{t}_1)$, $\mathcal{G}(\mathrm{t}_2)$ *und* $\mathcal{G}(\mathrm{t}_3)$ *eines dynamischen Graphen mit Knotenmenge* $\mathcal{V} = \{1, 2, 3\}$.

Definition 2.9 *Ein* dynamischer Graph *ist ein Graph* $\mathcal{G}(\mathrm{t}) = (\mathcal{V}_\mathrm{t}, \mathcal{E}_\mathrm{t})$ *auf einer Knotenmenge* $\mathcal{V}$*, der abhängig von einer Folge diskreter Zeitpunkte* $\mathrm{T} = (\mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t}_\mathrm{d})$ *unterschiedliche Ausprägungen haben kann. Die Menge dieser diskreten Zeitpunkte bezeichnen wir mit* $\mathcal{T} = \{\mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t}_\mathrm{d}\}$*. Das heißt, es gibt eine Funktion* $f : \mathcal{T} \to \mathcal{E}(\mathrm{P})$ *mit* $f : \mathrm{t} \mapsto \mathcal{E}_\mathrm{t}$ *und eine Funktion* $g : \mathcal{T} \to \mathcal{V}(\mathrm{P})$ *mit* $g : \mathrm{t} \mapsto \mathcal{V}_\mathrm{t}$*. Dabei steht die Menge* $\mathcal{E}(\mathrm{P})$ *für die Menge aller möglichen Kantenmengen und die Menge* $\mathcal{V}(\mathrm{P})$ *für die Menge aller möglichen Knotenmengen über der Knotenmenge* $\mathcal{V}$*.*



(a) Zeitexpandierter Graph $\mathcal{G}_1^3$ für die Graphenfolge $\hat{\mathcal{G}} = (\mathcal{G}(\mathrm{t}_1), \mathcal{G}(\mathrm{t}_2), \mathcal{G}(\mathrm{t}_3))$ des dynamischen Graphen in Abbildung 2.



(b) Zeitexpandierter Graph $\mathcal{G}_2^3$ für die Graphenfolge $\hat{\mathcal{G}} = (\mathcal{G}(\mathrm{t}_1), \mathcal{G}(\mathrm{t}_2), \mathcal{G}(\mathrm{t}_3))$ des dynamischen Graphen in Abbildung 2.

Abbildung 3: *Beispiele für zeitexpandierte Graphen mit* Variante 1 *der Interzeitkanten. Aus den drei Ausprägungen eines dynamischen Graphen in Abbildung 2 wurden zwei zeitexpandierte Graphen, einer mit Reichweite* $\mathrm{k} = 1$ *(3a) und einer mit Reichweite* $\mathrm{k} = 2$ *(3b), erzeugt.*

Die Knotenmengen $\mathcal{V}_\mathrm{t}$ sind also beliebige Teilmengen der Knotenmenge $\mathcal{V}$. Für die Kantenmengen $\mathcal{E}_\mathrm{t}$ gilt

$$\forall \mathrm{e} \in \mathcal{E}_\mathrm{t}\ \exists v_\mathrm{i}, v_\mathrm{j} \in \mathcal{V}_\mathrm{t} : \mathrm{e} = \{v_\mathrm{i}, v_\mathrm{j}\} \quad .$$

Bei dem Verbinden der einzelnen Ausprägungen müssen wir Knoten aus der Menge $\mathcal{V}$, die in mehreren Zeitpunkten vorkommen, unterscheiden. Sie werden bei den zeitexpandierten Graphen nicht zu einem Knoten verschmolzen. Wir nennen den Knoten $v_\mathrm{i} \in \mathcal{V}_\mathrm{t}$ den Repräsentanten von v_i in Zeitpunkt t und bezeichnen ihn im Folgenden mit v_i^t. Aus den Knotenmengen $\mathcal{V}_\mathrm{t}$ der d diskreten Zeitpunkte bilden wir die Knotenmenge $\mathcal{V}_\mathrm{d}$ über die disjunkte Vereinigung

$$\mathcal{V}_\mathrm{d} = \dot{\bigcup_{\mathrm{t_i} \in \mathcal{T}}} \mathcal{V}_{\mathrm{t_i}} \text{ mit } \mathrm{n_d} = |\mathcal{V}_\mathrm{d}| \leq \mathrm{d} \cdot |\mathcal{V}| \quad .$$

In Abbildung 2 sind drei Ausprägungen eines dynamischen Graphen abgebildet. Aus einem dynamischen Graphen lässt sich über eine Folge diskreter Zeitpunkte eine Folge von Graphen ableiten, die aus den Ausprägungen der verschiedenen Zeitpunkte der Folge bestehen.

Definition 2.10 *Eine* Graphenfolge $\hat{\mathcal{G}}$ *ist eine Folge von Graphen mit* $\hat{\mathcal{G}} = (\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\mathrm{d})$. *Wir nennen*

$$\mathrm{ver}(\mathcal{G}_\mathrm{i}, \mathrm{k}) = \left\{ \mathcal{G}_\mathrm{j} \in \hat{\mathcal{G}} \mid (|\mathrm{i} - \mathrm{j}| \leq \mathrm{k}) \wedge (\mathrm{i} \neq \mathrm{j}) \right\}$$

die k-Verwandtschaft *von* $\mathcal{G}_\mathrm{i}$*, wobei* $\mathrm{k} \in \mathbb{N}$.

Es sei $\mathcal{G}(\mathrm{t})$ ein dynamischer Graph und $\mathrm{T} = (\mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t}_\mathrm{d})$ eine Folge diskreter Zeitpunkte. Daraus bilden wir die Graphenfolge $\hat{\mathcal{G}} = (\mathcal{G}(\mathrm{t}_1), \mathcal{G}(\mathrm{t}_2), \ldots, \mathcal{G}(\mathrm{t}_\mathrm{d}))$ mit $\mathcal{G}(\mathrm{t}_\mathrm{i}) = (\mathcal{V}_{\mathrm{t}_\mathrm{i}}, \mathcal{E}_{\mathrm{t}_\mathrm{i}})$. Analog zu $\mathcal{V}_\mathrm{d}$ ergibt sich die Menge $\mathcal{E}_\mathrm{intra}$ aller Kanten der verschiedenen Ausprägungen aus der disjunkten Vereinigung

$$\mathcal{E}_\mathrm{intra} = \dot{\bigcup_{\mathrm{t}_\mathrm{i} \in \mathcal{T}}} \mathcal{E}_{\mathrm{t}_\mathrm{i}} \ .$$

Die Kanten der Menge $\mathcal{E}_\mathrm{intra}$ nennen wir Intrazeitkanten. Intrazeitkanten sind die Kanten innerhalb des zeitexpandierten Graphen, die nicht zwischen Knoten aus verschiedener Zeitpunkten verlaufen. Jetzt können wir anhand der eingeführten Begriffe einen zeitexpandierten Graphen definieren.

Definition 2.11 *Ein* zeitexpandierter Graph $\mathcal{G}^\mathrm{d}_\mathrm{k} = (\mathcal{V}_\mathrm{d}, \mathcal{E})$ *mit* $\mathcal{E} = \mathcal{E}_\mathrm{intra} \cup \mathcal{E}_\mathrm{inter}$ *entsteht aus der Graphenfolge* $\hat{\mathcal{G}} = (\mathcal{G}(\mathrm{t}_1), \mathcal{G}(\mathrm{t}_2), \ldots, \mathcal{G}(\mathrm{t}_\mathrm{d}))$ *eines dynamischen Graphen* $\mathcal{G}(\mathrm{t})$ *durch das Verbinden der Graphen* $\mathcal{G}(\mathrm{t}_\mathrm{i})$ *mit den Graphen in ihrer k-Verwandtschaft* $\mathrm{ver}(\mathcal{G}_{\mathrm{t}_\mathrm{i}}, \mathrm{k})$ *über eine Menge von* Interzeitkanten $\mathcal{E}_\mathrm{inter}$. *Den Parameter* k *nennen wir die* Reichweite *des zeitexpandierten Graphen.*

Interzeitkanten sind Kanten, deren Endknoten in verschiedenen Graphen aus $\hat{\mathcal{G}}$ liegen. Für die Festlegung, zwischen welchen Knoten der Graphen aus der Graphenfolge $\hat{\mathcal{G}}$ Interzeitkanten verlaufen, gibt es mehrere Möglichkeiten:

Variante 1 Sei $\mathrm{k} < \mathrm{d}$ beliebig fest. Es existiert zwischen zwei Knoten $v_\mathrm{x}^{\mathrm{t}_\mathrm{i}} \in \mathcal{V}_{\mathrm{t}_\mathrm{i}}$ und $v_\mathrm{y}^{\mathrm{t}_\mathrm{j}} \in \mathcal{V}_{\mathrm{t}_\mathrm{j}}$ genau dann eine Kante, wenn sie Repräsentanten des selben Knotens der Knotenmenge $\mathcal{V}$ des dynamischen Graphen $\mathcal{G}(\mathrm{t})$ sind, das heißt, wenn $\mathrm{x} = \mathrm{y}$ und der Graph $\mathcal{G}(\mathrm{t}_\mathrm{j})$ in der k-Verwandtschaft $\mathrm{ver}(\mathcal{G}_{\mathrm{t}_\mathrm{i}}, \mathrm{k})$ von Graph $\mathcal{G}(\mathrm{t}_\mathrm{i})$ liegt.

$$\begin{aligned} & \exists v_\mathrm{x}^{\mathrm{t}_\mathrm{i}} \in \mathcal{V}_{\mathrm{t}_\mathrm{i}} \ \exists v_\mathrm{y}^{\mathrm{t}_\mathrm{j}} \in \mathcal{V}_{\mathrm{t}_\mathrm{j}} : (\mathcal{G}(\mathrm{t}_\mathrm{j}) \in \mathrm{ver}(\mathcal{G}(\mathrm{t}_\mathrm{i}), \mathrm{k})) \wedge (\mathrm{x} = \mathrm{y}) \\ \iff & \exists \mathrm{e} \in \mathcal{E}_\mathrm{inter} : \mathrm{e} = \left\{ v_\mathrm{x}^{\mathrm{t}_\mathrm{i}}, v_\mathrm{y}^{\mathrm{t}_\mathrm{j}} \right\} \ . \end{aligned}$$

Variante 2 Sei $\mathrm{k} < \mathrm{d}$ beliebig fest. Es existiert zwischen zwei Knoten $v_\mathrm{x}^{\mathrm{t}_\mathrm{i}} \in \mathcal{V}_{\mathrm{t}_\mathrm{i}}$ und $v_\mathrm{y}^{\mathrm{t}_\mathrm{j}} \in \mathcal{V}_{\mathrm{t}_\mathrm{j}}$ genau dann eine Kante, wenn eine Distanzfunktion $\mathrm{dis} : \mathcal{V}_\mathrm{d} \times \mathcal{V}_\mathrm{d} \mapsto \mathcal{R}$ bzw. eine Ähnlichkeitsfunktion $\mathrm{sim} : \mathcal{V}_\mathrm{d} \times \mathcal{V}_\mathrm{d} \mapsto \mathcal{R}$ mit Wertebereich $\mathcal{R}$ für die beiden Knoten einen

Wert $\mathrm{r} \in \mathcal{R}$ zurückliefert, der kleiner bzw. größer als ein festgelegter Schwellenwert $\mathrm{p_i}$ ist und $\mathcal{G}(\mathrm{t_j}) \in \mathrm{ver}(\mathcal{G}_{\mathrm{t_i}}, \mathrm{k})$. Für die Ähnlichkeitsfunktion sim gilt:

$$\begin{aligned} &\exists v_{\mathrm{x}}^{\mathrm{t_i}} \in \mathcal{V}_{\mathrm{t_i}}\ \exists v_{\mathrm{y}}^{\mathrm{t_j}} \in \mathcal{V}_{\mathrm{t_j}} : (\mathcal{G}(\mathrm{t_j}) \in \mathrm{ver}(\mathcal{G}(\mathrm{t_i}), \mathrm{k})) \wedge (\mathrm{sim}(v_{\mathrm{x}}^{\mathrm{t_i}}, v_{\mathrm{y}}^{\mathrm{t_j}}) > \mathrm{p_i}) \\ \Longleftrightarrow\ &\exists \mathrm{e} \in \mathcal{E}_{\mathrm{inter}} : \mathrm{e} = \left\{ v_{\mathrm{x}}^{\mathrm{t_i}}, v_{\mathrm{y}}^{\mathrm{t_j}} \right\} \quad . \end{aligned}$$

Analog gilt für die Distanzfunktion dis:

$$\begin{aligned} &\exists v_{\mathrm{x}}^{\mathrm{t_i}} \in \mathcal{V}_{\mathrm{t_i}}\ \exists v_{\mathrm{y}}^{\mathrm{t_j}} \in \mathcal{V}_{\mathrm{t_j}} : (\mathcal{G}(\mathrm{t_j}) \in \mathrm{ver}(\mathcal{G}(\mathrm{t_i}), \mathrm{k})) \wedge (\mathrm{dis}(v_{\mathrm{x}}^{\mathrm{t_i}}, v_{\mathrm{y}}^{\mathrm{t_j}}) < \mathrm{p_i}) \\ \Longleftrightarrow\ &\exists \mathrm{e} \in \mathcal{E}_{\mathrm{inter}} : \mathrm{e} = \left\{ v_{\mathrm{x}}^{\mathrm{t_i}}, v_{\mathrm{y}}^{\mathrm{t_j}} \right\} \quad . \end{aligned}$$

Variante 3 Sei $\mathrm{k} < \mathrm{d}$ beliebig fest. Für jedes mögliche Knotenpaar $\{v_{\mathrm{x}}^{\mathrm{t_i}}, v_{\mathrm{y}}^{\mathrm{t_j}}\}$ mit $v_{\mathrm{x}}^{\mathrm{t_i}} \in \mathcal{V}_{\mathrm{t_i}}$ und $v_{\mathrm{y}}^{\mathrm{t_j}} \in \mathcal{V}_{\mathrm{t_j}}$ existiert eine Interzeitkante, falls $\mathcal{G}(\mathrm{t_j}) \in \mathrm{ver}(\mathcal{G}_{\mathrm{t_i}}, \mathrm{k})$.

$$\begin{aligned} &\exists v_{\mathrm{x}}^{\mathrm{t_i}} \in \mathcal{V}_{\mathrm{t_i}}\ \exists v_{\mathrm{y}}^{\mathrm{t_j}} \in \mathcal{V}_{\mathrm{t_j}} : (\mathcal{G}(\mathrm{t_j}) \in \mathrm{ver}(\mathcal{G}(\mathrm{t_i}), \mathrm{k})) \\ \Longleftrightarrow\ &\exists \mathrm{e} \in \mathcal{E}_{\mathrm{inter}} : \mathrm{e} = \left\{ v_{\mathrm{x}}^{\mathrm{t_i}}, v_{\mathrm{y}}^{\mathrm{t_j}} \right\} \quad . \end{aligned}$$

Variante 4 Die Interzeitkanten werden durch externe Informationen bestimmt.

Zwei Beispiele für zeitexpandierte Graphen mit Variante 1 sind in Abbildung 3 zu sehen. Dabei basieren sie auf den drei Ausprägungen des dynamischen Graphen aus Abbildung 2. Die Knotenmengen der einzelnen diskreten Zeitpunkte sind eine Teilmenge der Knotenmenge $\mathcal{V} = \{1, 2, 3\}$. In der Abbildung 3a sieht man das Resultat der Erzeugung des ungewichteten zeitexpandierten Graphen mit Reichweite $\mathrm{k} = 1$ aus der Graphenfolge $\hat{\mathcal{G}} = (\mathcal{G}(\mathrm{t_1}), \mathcal{G}(\mathrm{t_2}), \mathcal{G}(\mathrm{t_3}))$ des dynamischen Graphen. Der zeitexpandierte Graph mit Reichweite $\mathrm{k} = 2$ ist in Abbildung 3b zu sehen.

Sind die Graphen der Graphenfolge $\hat{\mathcal{G}}$ gewichtet, bilden wir aus den Funktionen $\omega_{\mathrm{t_i}} : \mathcal{E}_{\mathrm{t_i}} \rightarrow \mathcal{R}$, die für die einzelnen Graphen $\mathcal{G}(\mathrm{t_i})$ die Gewichte der Kanten liefern, die Funktion $\omega_{\mathrm{intra}} : \mathcal{E} \rightarrow \mathcal{R}$ mit

$$\omega_{\mathrm{intra}}(\mathrm{e}) = \begin{cases} \omega_{\mathrm{t_1}}(\mathrm{e}) & , \text{falls } \mathrm{e} \in \mathcal{E}_{\mathrm{t_1}} \\ \quad\vdots & \\ \omega_{\mathrm{t_d}}(\mathrm{e}) & , \text{falls } \mathrm{e} \in \mathcal{E}_{\mathrm{t_d}} \end{cases} .$$

Die Funktion $\omega : \mathcal{E} \rightarrow \mathcal{R}$ liefert für jede Kante $\mathrm{e} \in \mathcal{E}$ des zeitexpandierten Graphen das Gewicht der Kante zurück

$$\omega(\mathrm{e}) = \begin{cases} \omega_{\mathrm{intra}}(\mathrm{e}) & , \text{falls } \mathrm{e} \in \mathcal{E}_{\mathrm{intra}} \\ \omega_{\mathrm{inter}}(\mathrm{e}) & , \text{falls } \mathrm{e} \in \mathcal{E}_{\mathrm{inter}} \end{cases} ,$$

wobei $\omega_{\text{inter}} : \mathcal{E}_{\text{inter}} \rightarrow \mathcal{R}$, eine Funktion ist, die für eine Interzeitkante e deren Gewicht im zeitexpandierten Graphen $\mathcal{G}_{\mathrm{k}}^{\mathrm{d}}$ zurückgibt.

2.2.1. Gewichteter zeitexpandierter Graph mit Schwelle p

Hat ein Graph sehr viele Kanten mit kleinem Gewicht, kann das mehrere Gründe haben. Die kleinen Kanten können durch verrauschte Daten entstanden sein oder durch geringe Ähnlichkeiten zwischen den Knoten. In diesem Fall ist es nützlich, eine Schwelle für die Kantengewichte einzuführen.

Definition 2.12 *Für einen zeitexpandierten Graphen $\overline{\mathcal{G}}_{\mathrm{k,p}}^{\mathrm{d}}$ mit Schwelle $\mathrm{p} \in \mathcal{R}$ ersetzt die Funktion $\overline{\omega} : \mathcal{E} \rightarrow \mathcal{R}$ die Funktion ω:*

$$\overline{\omega}(\mathrm{e}) = \begin{cases} \omega(\mathrm{e}) & \text{, falls } \omega(\mathrm{e}) > \mathrm{p} \\ 0 & \text{, sonst} \end{cases} .$$

Das heißt, Kante $\mathrm{e} \in \mathcal{E}$ existiert in $\overline{\mathcal{G}}_{\mathrm{k,p}}^{\mathrm{d}}$, falls ihr Kantengewicht größer als Schwellenwert p ist. Alle Kantengewichte, die kleiner als Schwelle p sind werden aus dem zeitexpandierten Graphen entfernt. Dies reduziert die Komplexität der Operationen auf dem zeitexpandierten Graphen. In unseren Testreihen werden wir erkennen, dass sich eine Schwelle sehr positiv auf die Ergebnisse von Cluster-Verfahren auswirken kann.

2.3. Clusterung

Definition 2.13 *Eine* Clusterung $\mathcal{C} = \{\mathrm{C}_1, \mathrm{C}_2 \ldots \mathrm{C}_{\mathrm{cl}}\}$ *des Graphen $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ ist eine Partitionierung der Knotenmenge $\mathcal{V}$ von $\mathcal{G}$ in paarweise disjunkte* Cluster C_{i}. *Dabei ist jeder Cluster C_{i} eine Teilmenge von $\mathcal{V}$. Es gilt:*

$$\forall \mathrm{i} : \mathrm{C}_{\mathrm{i}} \subset \mathcal{V} \text{ und } \dot{\bigcup_{\mathrm{C}_{\mathrm{i}} \in \mathcal{C}}} \mathrm{C}_{\mathrm{i}} = \mathcal{V} .$$

Intraclusterkanten sind Kanten, die innerhalb eines Clusters verlaufen, während *Interclusterkanten* zwischen Knoten zweier verschiedener Cluster verlaufen. Die *Menge der Intraclusterkanten* bezeichnen wir mit $\mathcal{E}(\mathcal{C})$, die der *Interclusterkanten* mit $\overline{\mathcal{E}}(\mathcal{C})$. Die *Anzahl der Intraclusterkanten bzw. Interclusterkanten* ist $\mathrm{m}(\mathcal{C})$ bzw. $\overline{\mathrm{m}}(\mathcal{C})$. Ist der Graph gewichtet, so ist $\mathrm{w}(\mathcal{C})$ die *Summe der Gewichte der Intraclusterkanten* und $\overline{\mathrm{w}}(\mathcal{C})$ die *Summe der Gewichte der Interclusterkanten*. Einen guten Überblick über das Thema Clusterung liefert [Gae05].

Es gibt zwei triviale Clusterungen, zum einen das

1-Clustering $\mathcal{C}^1$, das alle Knoten in einem einzelnen Cluster C vereinigt, und das

Singleton-Clustering $\mathcal{C}^{\mathrm{s}}$, in dem jeder Knoten $v_{\mathrm{i}} \in \mathcal{V}$ einen eigenen Cluster C_{i} bildet.

2.3.1. Dichte einer Clusterung

Eine Dichtefunktion für Clusterungen $\mathcal{C}$ eines Graphen $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ wird in [DGG+06] vorgestellt. Der Index $\mathrm{density}$ spiegelt das Paradigma *„intracluster-density versus intercluster-sparsity“* wider:

$$\mathrm{density}(\mathcal{C}) = \frac{1}{2}\overbrace{\left(\frac{1}{|\mathcal{C}|}\sum_{\mathrm{C_i}\in\mathcal{C}}\frac{|\mathcal{E}(\mathrm{C_i})|}{\binom{|\mathrm{C_i}|}{2}}\right)}^{\text{Intraclusterdichte}} + \frac{1}{2}\left(1 - \frac{\overbrace{\mathrm{m} - \sum_{\mathrm{C_i}\in\mathcal{C}}|\mathcal{E}(\mathrm{C_i})|}^{\text{Interclusterdichte}}}{\binom{\mathrm{n}}{2} - \sum_{\mathrm{C_i}\in\mathcal{C}}\binom{|\mathrm{C_i}|}{2}}\right) \quad . \tag{1}$$

Eine *signifikante Clusterung* ist eine Clusterung, die eine hohe Intraclusterdichte und eine nur geringe Interclusterdichte erzielt. Das heißt eine Clusterung ist signifikant, wenn die Knoten eines Clusters ein kompaktes Netzwerk ergeben, während die Cluster untereinander nur über wenige Kanten bzw. Gewichte verknüpft sind. Die Knoten eines Clusters sollen viele Kanten bzw. starke Gewichte untereinander, aber nur wenige Kanten bzw. geringe Gewichte zu Knoten außerhalb des Clusters aufweisen. Der Index $\mathrm{density}$ ist nicht der einzige Index zur Bewertung der Güte einer Clusterung. Obwohl der $\mathrm{density}$-Index ein guter Indikator für signifikante Clusterungen ist, verwenden wir bei unseren Testreihen Indizes, die in der Literatur weiter verbreitet sind.

2.4. Indizes zur Bewertung von Clusterungen

Im Folgenden werden wir die in dieser Arbeit verwendeten Indizes kurz vorstellen.

2.4.1. Coverage

Einer der weitverbreitesten Indizes ist die *Coverage(Abdeckung)* einer Clusterung. Darunter versteht man das Verhältnis der Anzahl der Intraclusterkanten zu der Anzahl aller Kanten

$$\mathrm{cov}(\mathcal{C}) = \frac{\mathrm{m}(\mathcal{C})}{\mathrm{m}} = \frac{\mathrm{m}(\mathcal{C})}{\mathrm{m}(\mathcal{C}) + \overline{\mathrm{m}}(\mathcal{C})} \quad . \tag{2}$$

Verläuft der größte Teil der Kanten innerhalb der Cluster, so hat die Clusterung eine gute Coverage. Für gewichtete Graphen kann man die Coverage auf den Gewichten der Kanten definieren:

$$\mathrm{cov_w}(\mathcal{C}) = \frac{\mathrm{w}(\mathcal{C})}{\mathrm{w}(\mathcal{C}) + \overline{\mathrm{w}}(\mathcal{C})} \quad . \tag{3}$$

Der Wertebereich der Coverage ist $\mathcal{R} = [0, 1]$. Die Coverage ist ein naheliegender Index, hat jedoch einige Nachteile, wie zum Beispiel die maximale Bewertung des 1-Clustering. Ebenso hat eine Clusterung $\mathcal{C}$, deren Cluster $\mathrm{C_i}$ alle in Clustern $\mathrm{C_j}$ von Clusterung $\mathcal{C}'$ enthalten sind,

immer einen niedrigeren Coverage-Wert als Clusterung $\mathcal{C}'$ (siehe Abbildung 4). Der minimale Wert 0 wird vom Singleton-Clustering $\mathcal{C}^s$ erreicht. Signifikante Clusterungen haben immer einen hohen Coverage-Wert, lassen sich aus den angegebenen Gründen aber nicht immer durch die Maximierung der Coverage finden.

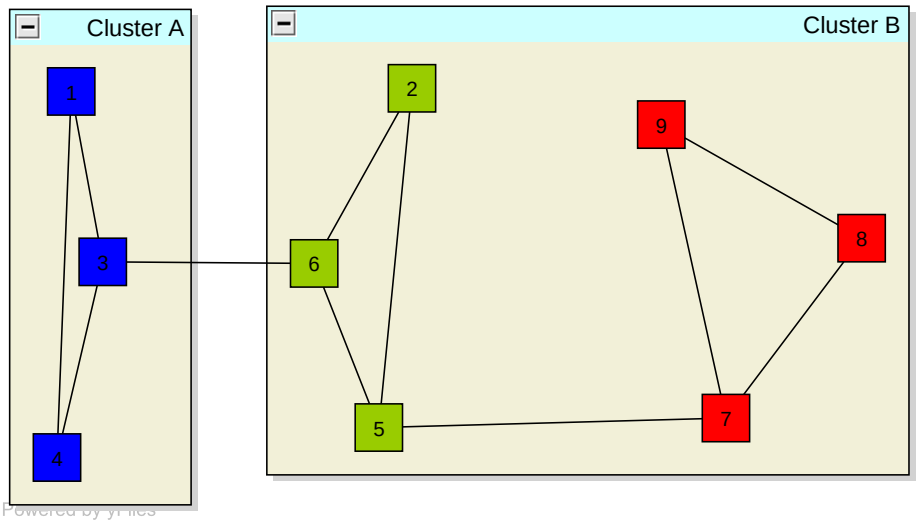


Abbildung 4: *Die durch die Kästen induzierte Clusterung $\mathcal{C}$' in zwei Cluster A und B erzielt den Wert* $\mathrm{cov}(\mathcal{C}')=\frac{10}{11}$, *während die intuitiv bessere Clusterung* $\mathcal{C} = \{\{1,3,4\},\{2,5,6\},\{7,8,9\}\}$, *welche durch die Knotenfarbe induziert wird, lediglich einen Wert von* $\mathrm{cov}(\mathcal{C}) = \frac{9}{11}$ *liefert.*

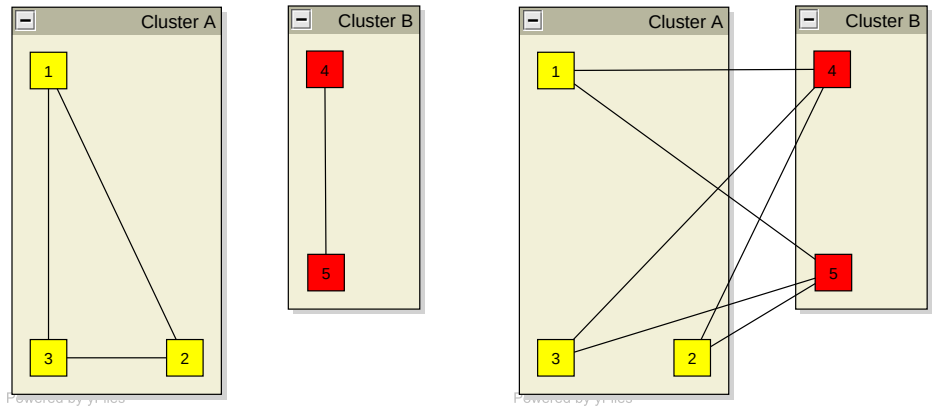
2.4.2. Performance

Die *Performance* $\mathrm{per}(\mathcal{C})$ ist das Verhältnis der Anzahl *korrekt zugeordneter Knotenpaare* zu allen möglichen Knotenpaaren

$$\mathrm{per}(\mathcal{C}) = \frac{\overbrace{\mathrm{m}(\mathcal{C})}^{(a)} + \overbrace{\frac{1}{2}\sum_{\mathrm{C_i}\in\mathcal{C}} (|\mathrm{C_i}| \cdot (\mathrm{n} - |\mathrm{C_i}|)) - \overline{\mathrm{m}}(\mathcal{C})}^{(b)}}{\frac{1}{2}\mathrm{n}(\mathrm{n}-1)} \quad . \tag{4}$$

Ein Knotenpaar ist korrekt zugeordnet, wenn eine Kante zwischen beiden existiert und beide demselben Cluster $\mathrm{C_i}$ angehören *(a)*, oder keine Kante zwischen den beiden Knoten existiert und sie verschiedenen Clustern $\mathrm{C_i}$ und $\mathrm{C_j}$ zugeordnet wurden *(b)*.

Der maximale Wert wird für Clusterungen von Graphen zurückgeliefert, bei denen die Cluster vollständig verbundene Cliquen sind, die untereinander keine Verbindungen haben (siehe Abbildung 5a). Gibt es keine Intraclusterkanten und ist jeder Knoten mit allen Knoten außerhalb seines Clusters verbunden, liefert dies den minimalen Wert 0. Dies gilt für kanonische Clusterungen vollständig bipartiter bzw. vollständig multipartiter Graphen (siehe Abbildung 5b). Die Aussagekraft der Performance ist für Graphen mit mittlerer oder geringer Dichte und vielen Knoten



(a) Clusterung $\mathcal{C}$ (b) Clusterung $\mathcal{C}'$

Abbildung 5: *Abbildungen zur Performance einer Clusterung. (a) Bei der linken Clusterung $\mathcal{C}$ entsprechen die Cluster Cliquen, die untereinander nicht verbunden sind. Es gilt* $\mathrm{per}(\mathcal{C}) = 1$. *(b) Bei der rechten Clusterung $\mathcal{C}'$ existieren keine Kanten innerhalb der Cluster und jeder Knoten ist mit allen Knoten außerhalb des Clusters verbunden. Es gilt* $\mathrm{per}(\mathcal{C}') = 0$.

problematisch, da hier der Anteil der korrekt zugeordneten Knotenpaare mit Kante sehr gering ist. Zum Beispiel sei $\mathcal{G}$ ein Graph mit $\mathrm{n} = 9000$ und durchschnittlich 18 Kanten pro Knoten, also insgesamt 81000 Kanten. Die maximale Veränderung der Performance einer Clusterung des Graphen aufgrund der Kanten beträgt lediglich $2 \cdot (81000/40500000) = 0,004$. Den größten Einfluss auf die Performance von dünnen Graphen hat die Anzahl der Cluster und deren Mächtigkeit, denn sie bestimmen die Anzahl der möglichen Kanten zwischen den verschiedenen Clustern. Diese fließt in die Berechnung der Performance ein (Teilterm (b) in Gleichung 4) und ist für dünne Graphen deutlich höher als die Anzahl aller Kanten des Graphen.

2.4.3. Conductance

Ein Schnitt $\mathcal{S}_{\mathcal{G}}\left[\mathrm{C}_1\right]$ eines Graphen ist eine Clusterung des Graphen $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ in genau zwei Cluster C_1 und C_2, wobei $\mathrm{C}_1 \subset \mathcal{V}$ und $\mathrm{C}_2 = \mathcal{V} \backslash \mathrm{C}_1$. Unter der *Conductance (Leitfähigkeit)* eines Schnittes $\mathcal{S}_{\mathcal{G}}\left[\mathrm{C}_1\right] = \{\mathrm{C}_1, \mathcal{V} \backslash \mathrm{C}_1\}$ versteht man das Verhältnis der Anzahl der Interclusterkanten $\overline{\mathrm{m}}(\mathcal{S}_{\mathcal{G}}\left[\mathrm{C}_1\right])$ (Kanten zwischen C_1 und $\mathcal{V} \backslash \mathrm{C}_1$) zu dem Minimum der Summe aller Grade der Knoten eines der beiden Cluster:

$$\phi(\mathcal{S}_{\mathcal{G}}\left[\mathrm{C}_1\right]) = \begin{cases} 1 & \text{, falls } \mathrm{C}_1 \in \{\mathcal{V}, \emptyset\} \\ 0 & \text{, falls } \mathrm{C}_1 \notin \{\mathcal{V}, \emptyset\} \text{ und } \overline{\mathrm{m}}(\mathcal{S}_{\mathcal{G}}\left[\mathrm{C}_1\right]) = 0 \\ \dfrac{\overline{\mathrm{m}}(\mathcal{S}_{\mathcal{G}}[\mathrm{C}_1])}{\min\left(\sum_{v_\mathrm{i} \in \mathrm{C}_1} g(v_\mathrm{i}), \sum_{v_\mathrm{j} \in (\mathcal{V} \backslash \mathrm{C}_1)} g(v_\mathrm{j})\right)} & \text{, sonst} \end{cases}.$$

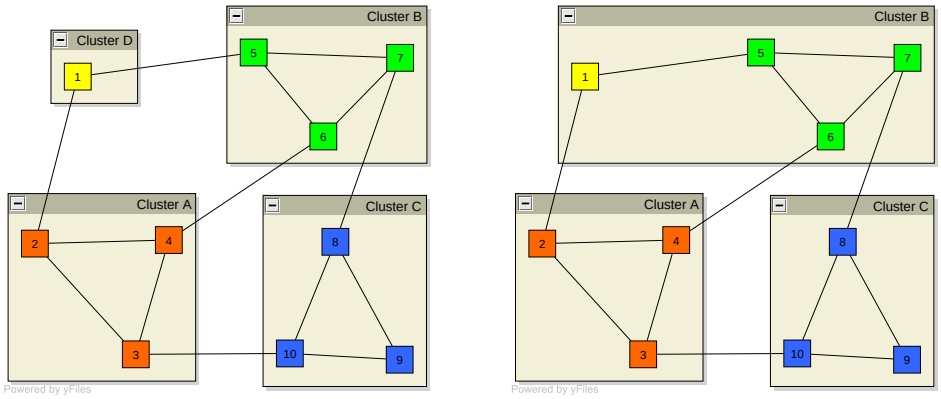
Für einen gewichteten Graphen $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ definieren wir die Conductance nach [KVV00] wie folgt:

$$\phi(\mathcal{S}_\mathcal{G}[\mathrm{C}_1]) = \begin{cases} 1 & , \text{falls } \mathrm{C}_1 \in \{\mathcal{V}, \emptyset\} \\ 0 & , \text{falls } \mathrm{C}_1 \notin \{\mathcal{V}, \emptyset\} \text{ und } \overline{\mathrm{w}}(\mathcal{S}_\mathcal{G}[\mathrm{C}_1]) = 0 \\ \frac{\overline{\mathrm{w}}(\mathcal{S}_\mathcal{G}[\mathrm{C}_1])}{\min\left(\sum_{v_\mathrm{i} \in \mathrm{C}_1} \omega(v_\mathrm{i}), \sum_{v_\mathrm{j} \in (\mathcal{V} \setminus \mathrm{C}_1)} \omega(v_\mathrm{j})\right)} & , \text{sonst} \end{cases} .$$

Der Wertebereich der Conductance ist $[0, 1]$. Aus der Conductance lassen sich zwei Kennwerte einer Clusterung $\mathcal{C}$ berechnen, die *Interclusterconductance* und die *Intraclusterconductance*. Die *maximale Interclusterconductance* δ_m berechnet sich aus der maximalen Conductance eines Clusters C_i von $\mathcal{C}$:

$$\delta_\mathrm{m}(\mathcal{C}) = 1 - \max_{\mathrm{C}_\mathrm{i} \in \mathcal{C}}(\phi(\mathcal{S}_\mathcal{G}[\mathrm{C}_\mathrm{i}])) \quad . \tag{5}$$

Ein niedriger Interclusterconductancewert von $\mathcal{C}$ bedeutet, dass es einen Cluster C_i gibt, für den die meisten Kanten einer der beiden Hälften des Schnittes $\mathcal{S}_\mathcal{G}[\mathrm{C}_\mathrm{i}]$ Interclusterkanten sind. Der Cluster C_i ist daher nicht gut gewählt im Sinne einer signifikanten Clusterung $\mathcal{C}$.



(a) Clusterung $\mathcal{C}$ (b) Clusterung $\mathcal{C}'$

Abbildung 6: *(a) Die maximale Conductance der Clusterung $\mathcal{C}$ wird für Cluster D mit $\phi(\mathcal{S}_\mathcal{G}[\mathrm{D}]) = 1$ erreicht. Die minimale Conductance ist $\phi(\mathcal{S}_\mathcal{G}[\mathrm{C}]) = 0,25$. (b) Bei der rechten Clusterung $\mathcal{C}'$ ergibt sich die minimale Conductance aus $\phi(\mathcal{S}_\mathcal{G}[\mathrm{C}]) = 0,25$ und die maximale aus $\phi(\mathcal{S}_\mathcal{G}[\mathrm{A}]) = 0,\overline{3}$.*

Nachteil dieses Qualitätsindex ist die Abhängigkeit des Index von einem einzelnen Cluster. Ein kleiner Cluster mit einer hohen Conductance zum restlichen Graphen führt automatisch zu einer schlechten Bewertung der Clusterung. Sei ein Cluster der Clusterung $\mathcal{C}$ gegeben durch einen einzelnen Knoten mit einer Kante zum restlichen Graphen, dann führt dieser Cluster zu einer minimalen Interclusterconductance von $\delta_\mathrm{m}(\mathcal{C}) = 0$, egal wie signifikant die restliche Clusterung ist. Ein weiteres Beispiel liefert Abbildung 6a. Aufgrund der Conductance $\phi(\mathcal{S}_\mathcal{G}[\mathrm{D}]) = 1$ von Cluster D, liefert die Interclusterconductance der Clusterung den Wert 0. Die starke Abhängigkeit

bezüglich eines einzigen Clusters der Clusterung wird bei der *durchschnittlichen Interclusterconductance* durch die Bildung des Mittelwerts der Conductance über alle Cluster vermieden. Sie berechnet sich aus

$$\delta_{\mathrm{d}}(\mathcal{C}) = 1 - \frac{1}{|\mathcal{C}|} \sum_{\mathrm{C_i} \in \mathcal{C}} \phi(\mathcal{S}_{\mathcal{G}}\left[\mathrm{C_i}\right]) \ . \tag{6}$$

Für das Beispiel in Abbildung 6a erhalten wir eine durchschnittliche Interclusterconductance von $\delta_{\mathrm{d}}(\mathcal{C}) \approx 0,521$. Ein hoher Wert der durchschnittlichen Interclusterconductance bedeutet, dass die Schnitte der Cluster eine hohe Intraclusterdichte haben, verglichen mit der Interclusterdichte der Schnitte. Auch mit der Normierung bezüglich der Anzahl der Cluster liefert die durchschnittliche Interclusterconductance nicht immer repräsentative Werte. In Abbildung 7 sieht man eine Clusterung $\mathcal{C}$ mit dem Cluster A, dessen Knoten durch keine Kanten verbunden sind. Trotzdem liefert $\delta_{\mathrm{d}}(\mathcal{C}) = 1 - \frac{1}{7}(6 \cdot \frac{1}{6} + 1) = \frac{5}{7}$. Eine hohe durchschnittliche Interclusterconductance spricht nicht automatisch für eine signifikante Clusterung, ist aber ein guter Indikator, denn eine signifikante Clusterung hat eine hohe durchschnittliche Interclusterconductance.

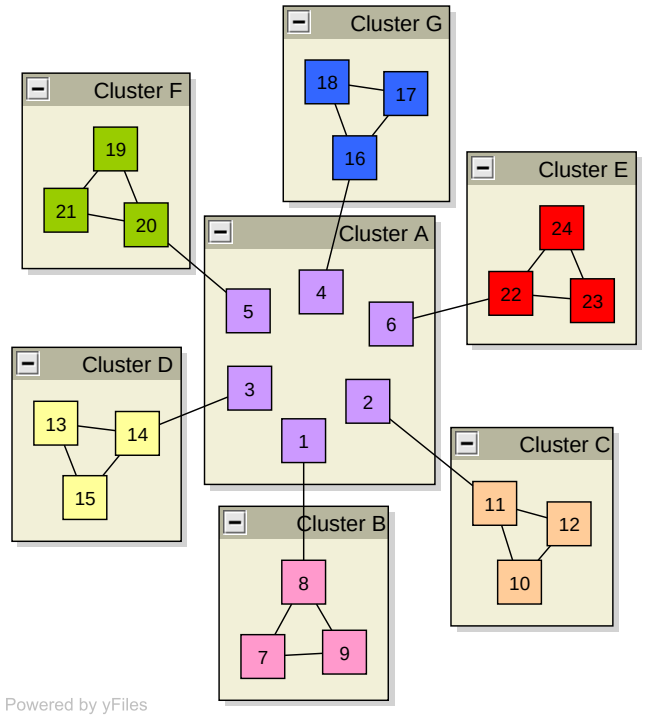


Abbildung 7: *Clusterung $\mathcal{C}$ eines Graphen mit hoher durchschnittlicher Conductance.*

Die *Intraclusterconductance* $\alpha(\mathcal{C})$ einer Clusterung $\mathcal{C}$ ergibt sich aus der minimalen Conductance eines durch die Cluster $\mathrm{C_i}$ induzierten Subgraphen $\mathcal{G}\left[\mathrm{C_i}\right] = (\mathrm{C_i}, \mathcal{E}(\mathrm{C_i}))$ des Graphen $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Die Conductance für Graphen ist gegeben durch:

$$\phi(\mathcal{G}) = \min_{\mathrm{C} \subset \mathcal{V}} \phi(\mathcal{S}_{\mathcal{G}}\left[\mathrm{C}\right]) \ .$$

Daraus ergibt sich die Intraclusterconductance der Clusterung $\mathcal{C}$:

$$\alpha(\mathcal{C}) = \min_{\mathrm{C_i} \in \mathcal{C}} \left(\phi(\mathcal{G}\left[\mathrm{C_i}\right])\right) \quad . \tag{7}$$

Sie liefert die Conductance des *besten* Schnittes innerhalb eines Clusters der Clusterung zurück. Ist dieser Wert gering, enthält die Clusterung einen Cluster, der durch einen *kleinen* Schnitt in zwei Cluster mit geringer Leitfähigkeit zwischeneinander zerteilt werden könnte. Die Berechnung der Intraclusterconductance ist NP-schwer [GA99].

2.4.4. Modularity

In [NG04] führen Newman und Girvan den Qualitätsindex *Modularity* ein, der die Coverage einer Clusterung in Relation zu ihrem Erwartungswert setzt. Genauer gesagt, wird für jeden Cluster $\mathrm{C_i}$ einer Clusterung $\mathcal{C}$ die Differenz von dem Anteil der Intraclusterkanten *(a)* und dem erwarteten Anteil der Intraclusterkanten *(b)* gebildet. Diesem Erwartungswert liegt folgendes Wahrscheinlichkeitsmodell zu Grunde: Die Knoten, die Clusterung und die erwarteten Kantengrade werden festgehalten und die m Kanten dann zufällig eingefügt. Dabei ist die Wahrscheinlichkeit, dass eine Kante inzident zu einem Knoten v ist proportional zu $g(v)$. Daraus ergibt sich die Wahrscheinlichkeit einer Intraclusterkante aus dem Quadrat des Verhältnisses der Summe aller Grade der Knoten des Clusters zu $2\mathrm{m}$, und man erhält

$$\mathrm{mod}\left(\mathcal{C}\right) = \sum_{\mathrm{C_i} \in \mathcal{C}} \left(\overbrace{\frac{|\mathcal{E}(\mathrm{C_i})|}{\mathrm{m}}}^{(a)} - \overbrace{\left(\frac{\sum_{v_\mathrm{j} \in \mathrm{C_i}} g(v_\mathrm{j})}{2\mathrm{m}} \right)^2}^{(b)} \right) \quad . \tag{8}$$

Für gewichtete Graphen ergibt sich die Modularity aus

$$\mathrm{mod_w}\left(\mathcal{C}\right) = \sum_{\mathrm{C_i} \in \mathcal{C}} \left(\frac{\sum_{\mathrm{e} \in \mathcal{E}(\mathrm{C_i})} \omega(\mathrm{e})}{\mathrm{w}(\mathcal{C}) + \overline{\mathrm{w}}(\mathcal{C})} - \left(\frac{\sum_{v_\mathrm{j} \in \mathrm{C_i}} \omega(v_\mathrm{j})}{2(\mathrm{w}(\mathcal{C}) + \overline{\mathrm{w}}(\mathcal{C}))} \right)^2 \right) \quad . \tag{9}$$

In [BDG+08] werden die Eigenschaften der Modularity genauer untersucht. Der Wertebereich der Modularity entspricht $[-\frac{1}{2}, 1)$. Ist die Coverage höher als zu erwarten war, ist die Modularity positiv. Wenn sie niedriger als erwartet ist, nimmt die Modularity negative Werte an. Den niedrigsten Wert der Modularity liefert eine kanonische Clusterung eines bipartiten Graphen. Das heißt, die Clusterung hat genau zwei Cluster, deren Knoten nur mit Knoten außerhalb des Clusters verbunden sind und in denen keine Intraclusterkanten existieren. Ein Beispiel für solch eine Clusterung ist in Abbildung 5b zu sehen. Für eine Clusterung, die vielen gleich großen unverbundenen k-Cliquen entspricht mit $k > 1$, geht der Wert der Modularity gegen 1. Dabei ist die

Modularity nicht abhängig von der Größe der Cliquen, sondern hängt nur von der Anzahl der Cliquen ab. Die Clusterung $\mathcal{C}$ mit x k-Cliquen ohne Interclusterkanten hat die Modularity

$$\mathrm{mod}\,(\mathcal{C}) = \frac{\mathrm{x}-1}{\mathrm{x}} \quad .$$

Das Finden einer Clusterung mit maximaler Modularity ist NP-vollständig [BDG+08].

2.5. Cluster-Verfahren

Dieser Abschnitt stellt die benutzten Cluster-Verfahren kurz vor. Die ersten beiden Verfahren basieren auf bereits vorgestellten Kennwerten, während das dritte Verfahren aus statistischen Überlegungen hervorgeht. Dabei werden die Cluster über das Simulieren von *Random Walks* auf dem Graphen gefunden.

Algorithmus 1 Greedy-Significance-Clustering

Eingabe: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
Ausgabe: Clusterung $\mathcal{C}$ von Graph $\mathcal{G}$

1: $\mathcal{C} \leftarrow$ Singleton-Clustering von Graph $\mathcal{G}$
2: $\mathrm{dendrogramm} \leftarrow$ speichere Tupel $(\mathcal{C}, \mathrm{mod}(\mathcal{C}))$
3: **while** $|\mathcal{C}| \neq 1$ **do**
4: $\quad \mathrm{maxvalue} = -\frac{1}{2}$
5: $\quad$ **for all** $\mathrm{C_i}, \mathrm{C_j} \in \mathcal{C}$ mit $\mathrm{i} \neq \mathrm{j}$ **do**
6: $\quad\quad \mathcal{C}' = (\mathcal{C} \setminus \{\mathrm{C_i}, \mathrm{C_j}\}) \cup \{\mathrm{C_i} \cup \mathrm{C_j}\}$
7: $\quad\quad \mathrm{value} \leftarrow \mathrm{mod}(\mathcal{C}') - \mathrm{mod}(\mathcal{C})$
8: $\quad\quad$ **if** $\mathrm{value} > \mathrm{maxvalue}$ **then**
9: $\quad\quad\quad \mathrm{candidate1} \leftarrow \mathrm{C_i}$
10: $\quad\quad\quad \mathrm{candidate2} \leftarrow \mathrm{C_j}$
11: $\quad\quad\quad \mathrm{maxvalue} \leftarrow \mathrm{value}$
12: $\quad\quad$ **end if**
13: $\quad$ **end for**
14: $\quad \mathcal{C} \leftarrow (\mathcal{C} \setminus \{\mathrm{candidate1}, \mathrm{candidate2}\}) \cup \{\mathrm{candidate1} \cup \mathrm{candidate2}\}$
15: $\quad \mathrm{dendrogramm} \leftarrow$ speichere Tupel $(\mathcal{C}, \mathrm{mod}(\mathcal{C}))$
16: **end while**
17: wähle aus $\mathrm{dendrogramm}$ die Clusterung $\mathcal{C}$ mit maximaler Modularity $\mathrm{mod}(\mathcal{C})$

2.5.1. Greedy-Significance-Clustering

Da das Finden einer Clusterung mit maximaler Modularity NP-vollständig ist [BDG+08], benutzt das *Greedy-Significance-Clustering* [GGW07] einen Greedy-Algorithmus, der eine Heuristik darstellt. Tests lassen vermuten, dass der Algorithmus Ergebnisse liefert, die nahe am Optimum liegen [BDG+08]. Das Greedy-Significance-Clustering ist ein hierarchisches Cluster-Verfahren, welches ausgehend vom Singleton-Clustering $\mathcal{C} = \mathcal{C}^{\mathrm{s}}$ iterativ jeweils die beiden Cluster $\mathrm{C_i}$ und

C_j vereinigt, die die Differenz

$$\mathrm{mod}(\mathcal{C}') - \mathrm{mod}(\mathcal{C}) \text{ mit } \mathcal{C}' = (\mathcal{C} \setminus \{C_i, C_j\}) \cup \{C_i \cup C_j\}$$

maximieren. Man könnte diese Differenz die Tendenz von Clusterung $\mathcal{C}'$ bezüglich Clusterung $\mathcal{C}$ nennen, da ein positiver bzw. negativer Wert eine Verbesserung bzw. Verschlechterung der Modularity bedeutet. Das Ganze wird solange wiederholt, bis alle Knoten in einem Cluster zusammengefasst sind. Das Verfahren benutzt ein *Dendrogramm*, um abschließend die Clusterung mit maximaler Modularity aller Teilschritte auszuwählen. Nutzt man geeignete Datenstrukuren und eine effiziente Version von Algorithmus 1, ist die Laufzeit des Greedy-Significance-Clustering in $\mathcal{O}(n^2 \log n)$ [GGW07].

2.5.2. Iterative-Conductance-Cutting

Algorithmus 2 Iterative-Conductance-Cutting (ICC)

Eingabe: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, Conductance-Schwellenwert $0 < a^* < 1$
Ausgabe: Clusterung $\mathcal{C}$ von Graph $\mathcal{G}$

1: $\mathcal{C} \leftarrow \{\mathcal{V}\}$
2: **while** es gibt ein $C_i \in \mathcal{C}$ mit $\phi(\mathcal{G}[C_i]) < a^*$ **do**
3: $\quad x \leftarrow$ Eigenvektor des zweitgrößten Eigenwertes von $M(\mathcal{G}[C_i])$
4: $\quad \mathcal{S} \leftarrow \{S \subset C_i \mid \max_{v \in \mathcal{S}} \{x_v\} < \min_{w \in C_i \setminus \mathcal{S}} \{x_w\}\}$
5: $\quad C \leftarrow$ ein beliebiges Element der Menge $\{C \subset C_i \mid \phi(C) = \min_{S \in \mathcal{S}} \phi(S)\}$
6: $\quad \mathcal{C} \leftarrow (\mathcal{C} \setminus C_i) \cup \{C, C_i \setminus C\}$
7: **end while**

Das Verfahren *Iterative-Conductance-Cutting* (ICC) [KVV00] basiert auf der in 2.4.3 vorgestellten Conductance. Ausgehend vom 1-Clustering sucht es Schnitte mit kleiner Conductance und verfeinert dabei schrittweise die Clusterung. Allerdings ist es NP-schwer, einen Schnitt mit minimaler Conductance zu finden [GA99]. Deshalb wird eine Heuristik benutzt. Das spektrale Verfahren verwendet den zweitgrößten Eigenwert für eine Ordnung der Knoten bezüglich des zugehörigen Eigenvektors. Zur Erklärung sei gesagt, dass die Vielfachheit des ersten Eigenwertes einer Adjazenzmatrix eines ungerichteten Graphen die Anzahl der Komponenten des Graphen liefert. Die weiteren Eigenwerte tragen für die Clusterung des Graphen nützliche Informationen. Wir werden hier nicht weiter auf diesen Sachverhalt eingehen. Der interessierte Leser sei auf [Gae02, KVV00, vL07] verwiesen. Der Algorithmus sucht einen minimalen Schnitt, der die Knoten getreu der gefundenen Ordnung in zwei Cluster aufteilt. Diese Schritte werden wiederholt, bis es keinen Schnitt durch einen Cluster mehr gibt, dessen heuristisch gefundene Conductance kleiner als der festgelegte Schwellenwert a^* ist. Durch die Reduzierung des Problems auf eine lineare Separierung der Knotenmenge, ist die Laufzeit des Iterative-Conductance-Cutting polylogarithmisch. Der Pseudo-Code ist in Algorithmus 2 zu finden.

2.5.3. Markov-Clustering

Wie bereits erwähnt, basiert das *Markov-Clustering* (MCL) [vD98] auf sogenannten *Random Walks*. Das sind zufällige Traversierungen eines Graphen entlang von Kanten. Wobei sich die

Wahrscheinlichkeit ausgehend von Knoten v_i als nächstes Knoten v_j zu besuchen, aus dem Anteil des Gewichtes von Kante $\mathrm{e} = \{v_\mathrm{i}, v_\mathrm{j}\}$ am Gewicht $\omega(v_\mathrm{i})$ von v_i ergibt. Im Folgenden ist $\mathcal{M}(\mathcal{A}^\omega) = \mathcal{D}(\mathcal{A}^\omega)^{-1}\mathcal{A}^\omega$ die *normalisierte gewichtete Adjazenzmatrix* der gewichteten Adjazenzmatrix $\mathcal{A}^\omega$ mit Diagonalmatrix $\mathcal{D}(\mathcal{A}^\omega)$ der Gewichte der Knoten. Dabei ergeben die Summen der Einträge jeder Zeile der Matrix $\mathcal{M}(\mathcal{A}^\omega)$ immer 1. Die Einträge in Zeile i geben den Anteil des jeweiligen Kantengewichts am Gewicht von Knoten v_i wieder.

Beginnend mit der Matrix $\mathcal{M} = \mathcal{M}(\mathcal{A}^\omega)$ werden zwei Schritte wiederholt, bis ein Abbruchkriterium erfüllt ist. Im ersten Schritt, der *Expansion* genannt wird, wird die stochastische Matrix $\mathcal{M}$ in die e-te Potenz erhoben, wobei $\mathrm{e} \in \mathbb{N}_{>1}$. Dies entspricht der Simulation eines Random Walks der Länge e, da die Matrix $\mathcal{M}(\mathcal{A}^\omega)$ die Übergangswahrscheinlichkeiten bezüglich der Gewichtung der Kanten enthält und die Zeilensummen jeweils den Wert 1 ergeben. In anderen Worten enthält der Eintrag der i-ten Zeile und j-ten Spalte von $\mathcal{M}(\mathcal{A}^\omega)$ die Wahrscheinlichkeit, ausgehend von Knoten i den Knoten j als nächstes zu besuchen. Der zweite Schritt *Inflation* bildet die r-te Potenz für jeden Eintrag der Matrix. Anschließend werden die Einträge der Matrix bezüglich ihrer Zeilensummen normalisiert. Das Ergebnis dieser Normalisierung ist erneut eine stochastische Matrix. Durch die Expansion wird der Fluss innerhalb des Graphen erhöht. Die darauf folgende Inflation verringert für $\mathrm{r} > 1$ den Fluss innerhalb des Graphen. Je höher der Eintrag, desto weniger schwächt er sich durch die Inflation ab, während niedrige Einträge der Matrix stärker abgeschwächt werden. Die beiden Schritte werden solange wiederholt, bis sich die Matrix $\mathcal{M}$ durch Expansion und Inflation nicht mehr verändert. Sie bildet einen sogenannten *Fixpunkt*. Das Ergebnis der Clusterung sind die Komponenten des Graphen, der duch die Einträge der Matrix $\mathcal{M}$ gegeben ist, die nicht null sind. Der Pseudo-Code ist in Algorithmus 3 zu sehen.

Algorithmus 3 Markov-Clustering (MCL)

Eingabe: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $\mathrm{e} \in \mathbb{N}_{>1}$, $\mathrm{r} \in \mathbb{R}^+$
Ausgabe: Clusterung $\mathcal{C}$ von Graph $\mathcal{G}$

1: $\mathcal{M} \leftarrow \mathcal{M}(\mathcal{A}^\omega)$
2: **while** $\mathcal{M}$ ist kein Fixpunkt **do**
3: $\quad\mathcal{M} \leftarrow \mathcal{M}^\mathrm{e}$
4: $\quad$**for all** $\mathrm{u} \in \mathcal{V}$ **do**
5: $\quad\quad$**for all** $\mathrm{v} \in \mathcal{V}$ **do**
6: $\quad\quad\quad\mathcal{M}_{\mathrm{u,v}} \leftarrow \mathcal{M}^\mathrm{r}_{\mathrm{u,v}}$
7: $\quad\quad$**end for**
8: $\quad\quad$**for all** $\mathrm{v} \in \mathcal{V}$ **do**
9: $\quad\quad\quad\mathcal{M}_{\mathrm{u,v}} \leftarrow \frac{\mathcal{M}_{\mathrm{u,v}}}{\sum_{\mathrm{w}\in\mathcal{V}} \mathcal{M}_{\mathrm{u,w}}}$
10: $\quad\quad$**end for**
11: $\quad$**end for**
12: **end while**
13: $\mathcal{C} \leftarrow$ Clusterung, deren Cluster aus den Komponenten des durch $\mathcal{M}$ induzierten Graphen bestehen

Die Laufzeit des MCL-Algorithmus liegt in $\mathcal{O}(\mathrm{n}^3)$. Der Faktor n^3 resultiert aus der Multiplikation zweier $\mathrm{n} \times \mathrm{n}$ Matrizen während der Expansion. Die Inflation liegt in $\mathcal{O}(\mathrm{n}^2)$. Zur Reduzierung der Laufzeit existiert eine Pruning-Variante des Algorithmus, bei der nur die $\kappa \in \mathbb{N}$ jeweils höchsten Elemente einer Zeile in die Berechnung einfließen. Die Pruning-Variante reduziert die

Komplexität des Algorithmus auf $\mathcal{O}(\mathrm{n}\ \kappa^2)$ [vD98].

2.6. Cosine-Similarity

Die *Cosine-Similarity* ist ein Ähnlichkeitsmaß für die Richtung von Vektoren in $\mathbb{R}^\mathrm{n}$ mit $\mathrm{n} \in \mathbb{N}$. Sie berechnet sich aus dem Cosinus des Winkels α, den zwei Vektoren aus $\mathbb{R}^\mathrm{n}$ einschließen. Für den Winkel zwischen zwei Vektoren $\mathrm{v_i}$ und $\mathrm{v_j}$ ergibt sich die Cosine Similarity aus

$$\mathrm{sim}(\mathrm{v_i}, \mathrm{v_j}) = \cos\alpha = \frac{\sum_{\mathrm{x}=1}^{\mathrm{n}} (\mathrm{v_i}(x) \cdot \mathrm{v_j}(x))}{\sqrt{\sum_{\mathrm{x}=1}^{\mathrm{n}} \mathrm{v_i}(x)^2 \cdot \sum_{\mathrm{x}=1}^{\mathrm{n}} \mathrm{v_j}(x)^2}} \quad . \tag{10}$$

Zwei Vektoren liefern den Wert 0 bzw. 1, falls sie den Winkel $\alpha = 90°$ einschließen bzw. falls sie in die gleiche Richtung verlaufen. Für zwei Knoten v_i und v_j eines Graphen berechnet sich die Cosine Similarity aus ihren Spaltenvektoren $\mathrm{v_i}$ und $\mathrm{v_j}$ der Adjazenzmatrix $\mathcal{A}$, das heißt $\mathrm{sim}(v_\mathrm{i}, v_\mathrm{j}) = \mathrm{sim}(\mathrm{v_i}, \mathrm{v_j})$. Bei vielen Anwendungen, wie bei der Berechnung von Ähnlichkeiten zwischen Merkmalsvektoren ist die Verwendung der Cosine Similarity sinnvoll.

2.6.1. Adapted-Cosine-Similarity

Sei der Vektor eines Knotens eines Graphen aus den Kantenwerten des Knotens zu den n Knoten des Graphen erzeugt, so steht für Knoten v_i der Wert $\mathrm{v_i(j)}$ für den Wert der Kante $\mathrm{e} = \{v_\mathrm{i}, v_\mathrm{j}\}$. Wenn sich ein Eintrag der Vektoren jeweils auf den Knoten selbst bezieht, ist das Ergebnis der Cosine Similarity fragwürdig. In vielen Anwendungen existieren in den Graphen keine Schlingen, oder sie haben eine andere Bedeutung, als die anderen Kanten. Ein Beispiel hierfür ist unser E-Mail-Graph. Kanten zwischen Knoten haben hier den Wert, der gleich der Anzahl der ausgetauschten E-Mails ist. E-Mails, die von einer Person an sich selbst geschickt werden, haben keine Aussagekraft über die Zugehörigkeit einer Person bzw. eines Accounts zu einer bestimmten Gruppe. Sie dienen meist zur Datenspeicherung, als Erinnerung oder zur Bestätigung, dass eine E-Mail verschickt wurde. Deshalb werden wir in unseren Betrachtungen diese E-Mails nicht berücksichtigen. Dies führt zu folgendem Problem:

Für ein Knotenpaar eines Graphen ohne Schlingen kann die Berechnung der Cosine-Similarity nie den Wert 1 zurückliefern, falls das Knotenpaar über eine Kante verbunden ist. Denn der Wert $\mathrm{w_e}$ des Gewichts dieser Kante verschwindet bei der Berechnung des Skalarproduktes der beiden Vektoren. Bei der Normierung wird er jedoch für beide Vektoren berücksichtigt. Dieses Problem lässt sich lösen, indem wir vor der Normierung zum Skalarprodukt der beiden Vektoren das Quadrat von $\mathrm{w_e}$ hinzuaddieren. Daraus ergibt sich die folgende Änderung der Cosine-Similarity:

$$\mathrm{sim_{ad}}(v_\mathrm{i}, v_\mathrm{j}) = \frac{\sum_{\mathrm{x}=1}^{\mathrm{n}} (\mathrm{v_i}(x) \cdot \mathrm{v_j}(x)) + \mathrm{v_j}(i) \cdot \mathrm{v_i}(j)}{\sqrt{\sum_{\mathrm{x}=1}^{\mathrm{n}} \mathrm{v_i}(x)^2 \cdot \sum_{\mathrm{x}=1}^{\mathrm{n}} \mathrm{v_j}(x)^2}} \quad . \tag{11}$$

Diese Variante der Cosine-Similarity nennen wir *Adapted-Cosine-Similarity*. Das folgende Beispiel illustriert das Problem und die durch die Adapted-Cosine-Similarity erreichte Verbesserung.

Beispiel

Gegeben sei ein E-Mail-Graph mit 4 Knoten v_1, v_2, v_3 und v_4 und der Adjazenzmatrix

$$\mathcal{A} = \begin{pmatrix} 7 & 1 & 100 & 40 \\ 1 & 55 & 0 & 3 \\ 100 & 0 & 0 & 40 \\ 40 & 3 & 40 & 34 \end{pmatrix}.$$

Beim Berechnen der Ähnlichkeiten zwischen den Knoten möchten wir die E-Mails der Knoten an sich selbst nicht berücksichtigen. Dies führt zu einer Matrix

$$\mathcal{A}' = \begin{pmatrix} 0 & 1 & 100 & 40 \\ 1 & 0 & 0 & 3 \\ 100 & 0 & 0 & 40 \\ 40 & 3 & 40 & 0 \end{pmatrix}.$$

Die Vektoren der Knoten des Graphen lauten:

$$v_1 = \begin{pmatrix} 0 \\ 1 \\ 100 \\ 40 \end{pmatrix}, v_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 3 \end{pmatrix}, v_3 = \begin{pmatrix} 100 \\ 0 \\ 0 \\ 40 \end{pmatrix} \textit{ und } v_4 = \begin{pmatrix} 40 \\ 3 \\ 40 \\ 0 \end{pmatrix} .$$

Daraus ergeben sich folgende Cosine-Similarity-Werte:

$$\mathrm{sim}(v_1, v_2) = \frac{(0 \cdot 1) + (1 \cdot 0) + (100 \cdot 0) + (40 \cdot 3)}{\sqrt{(1^2 + 100^2 + 40^2) \cdot (1^2 + 3^2)}} \approx 0,35232$$

$$\mathrm{sim}(v_1, v_3) = \frac{(0 \cdot 100) + (1 \cdot 0) + (100 \cdot 0) + (40 \cdot 40)}{\sqrt{(1^2 + 100^2 + 40^2) \cdot (100^2 + 40^2)}} \approx 0,13793$$

Eigentlich sollte zwischen den Knoten v_1 und v_3 eine größere Ähnlichkeit bestehen als zwischen Knoten v_1 und Knoten v_2, da sie viele E-Mails ausgetauscht und gleich viele E-Mails an Knoten v_4 geschrieben haben.

Deshalb benutzen wir die Adapted-Cosine-Similarity zur Berechnung der Ähnlichkeit. Die daraus resultierenden Werte spiegeln die starke Ähnlichkeit der Knoten v_1 und v_3 wider:

$$\mathrm{sim}_{\mathrm{ad}}(v_1, v_2) = \frac{(0 \cdot 1) + (1 \cdot 0) + (100 \cdot 0) + (40 \cdot 3) + (1 \cdot 1)}{\sqrt{(1^2 + 100^2 + 40^2) \cdot (1^2 + 3^2)}} \approx 0,35525 \text{ und}$$

$$\mathrm{sim}_{\mathrm{ad}}(v_1, v_3) = \frac{(0 \cdot 100) + (1 \cdot 0) + (100 \cdot 0) + (40 \cdot 40) + (100 \cdot 100)}{\sqrt{(1^2 + 100^2 + 40^2) \cdot (100^2 + 40^2)}} \approx 0,99997 \ .$$

2.7. Cosine Similarity Matrix

Die *Cosine Similarity Matrix* $\mathcal{A}_{\text{sim}}$ enthält die Adapted Cosine Similarity-Werte der Adjazenzmatrix $\mathcal{A}$ eines Graphen:

$$\mathcal{A}_{\text{sim}} = \begin{pmatrix} \text{sim}_{\text{ad}}(v_1, v_1) & \text{sim}_{\text{ad}}(v_1, v_2) & \dots & \text{sim}_{\text{ad}}(v_1, v_n) \\ \text{sim}_{\text{ad}}(v_2, v_1) & \text{sim}_{\text{ad}}(v_2, v_2) & \dots & \text{sim}_{\text{ad}}(v_2, v_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{sim}_{\text{ad}}(v_n, v_1) & \text{sim}_{\text{ad}}(v_n, v_2) & \dots & \text{sim}_{\text{ad}}(v_n, v_n) \end{pmatrix} .$$

2.8. Vergleichsmaße für Clusterungen

Die von uns genutzen Vergleichsmaße werden hier kurz eingeführt. Für einen detaillierten Überblick über Vergleichsmaße empfehlen wir [Del06] und [Mei07, Mei05].

2.8.1. Das Vergleichsmaß *bestmatch*

In [HKKS04] werden auseinander hervorgehende Cluster über das Vergleichsmaß bestmatch bestimmt. Dieses Maß ist ein Schnittmaß, dessen Wert sich aus den Überdeckungen der Cluster ergibt. Der bestmatch von Cluster C_i bezüglich der Clusterung $\mathcal{C}'$ berechnet sich aus dem maximalen $\text{match}_{\text{old}}$ des Clusters mit einem Cluster C_j aus $\mathcal{C}'$. Dabei ergibt sich der $\text{match}_{\text{old}}$ zweier Cluster C_i und C_j aus der bezüglich der Mächtigkeit des Größeren der beiden Cluster normierten Größe der Schnittmenge

$$\text{match}_{\text{old}}(\text{C}_\text{i}, \text{C}_\text{j}) = \min\left(\frac{|\text{C}_\text{i} \cap \text{C}_\text{j}|}{|\text{C}_\text{i}|}, \frac{|\text{C}_\text{i} \cap \text{C}_\text{j}|}{|\text{C}_\text{j}|}\right) .$$

Haben zwei Cluster C_{k1} und C_{k2} jeweils die selbe Schnittmenge mit einem größeren oder zumindest gleich großen Cluster C_g (siehe Abbildung 8), so gilt unabhängig von der Mächtigkeit der Mengen $\text{C}_{\text{k1}} \backslash (\text{C}_{\text{k1}} \cap \text{C}_\text{g})$ und $\text{C}_{\text{k1}} \backslash (\text{C}_{\text{k2}} \cap \text{C}_\text{g})$:

$$\text{match}_{\text{old}}(\text{C}_{\text{k1}}, \text{C}_\text{g}) = \text{match}_{\text{old}}(\text{C}_{\text{k2}}, \text{C}_\text{g}) .$$

Da dies für ein Vergleichsmaß nicht intuitiv ist, normieren wir den match, statt ihn bezüglich des Maximums der Mächtigkeit der beiden verglichenen Cluster zu normieren, bezüglich der Mächtigkeit der Vereinigung beider Cluster. Für zwei Cluster $\text{C}_\text{i} \in \mathcal{C}$ und $\text{C}_\text{j} \in \mathcal{C}'$ ergibt sich der match aus folgender Gleichung:

$$\text{match}(\text{C}_\text{i}, \text{C}_\text{j}) = \frac{|\text{C}_\text{i} \cap \text{C}_\text{j}|}{|\text{C}_\text{i} \cup \text{C}_\text{j}|} .$$

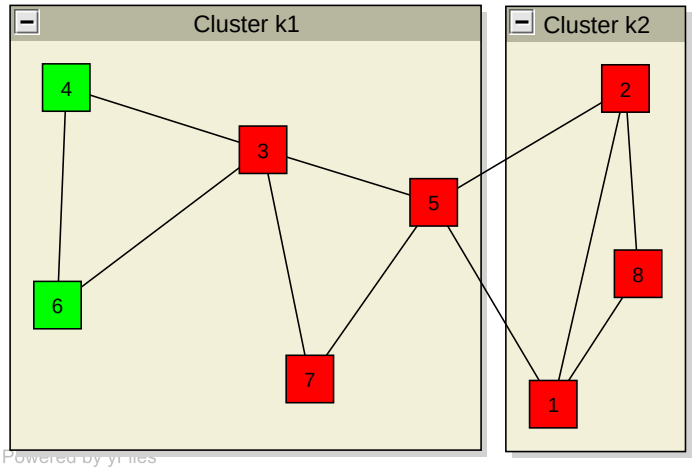


Abbildung 8: *Die Clusterung $\mathcal{C}'$ ist gegeben durch die Färbung der Knoten, dabei nennen wir den roten Cluster $\mathrm{C_g}$. Die Clusterung $\mathcal{C}$ ist gegeben durch die Cluster $\mathrm{C_{k1}}$ und $\mathrm{C_{k2}}$. Für beide Cluster aus $\mathcal{C}$ ergibt sich* $\mathrm{match_{old}}(\mathrm{C_{k1}}, \mathrm{C_g}) = \mathrm{match_{old}}(\mathrm{C_{k2}}, \mathrm{C_g}) = 0,5$.

Der $\mathrm{bestmatch}(\mathrm{C_i}, \mathcal{C}')$ des Clusters $\mathrm{C_i}$ bezüglich Clusterung $\mathcal{C}'$ ist der höchste match eines Clusters aus $\mathcal{C}'$ mit dem Cluster $\mathrm{C_i}$

$$\mathrm{bestmatch}(\mathrm{C_i}, \mathcal{C}') = \max_{\mathrm{C_j} \in \mathcal{C}'} (\mathrm{match}(\mathrm{C_i}, \mathrm{C_j})) \ . \tag{12}$$

Ausgehend von diesem Vergleichsmaß definieren wir den $\mathrm{bestmatch_{av}}$ zweier Clusterungen $\mathcal{C}$ und $\mathcal{C}'$ als Durchschnittswert aller $\mathrm{bestmatch}(\mathrm{C_i}, \mathcal{C}')$:

$$\mathrm{bestmatch_{av}}(\mathcal{C}, \mathcal{C}') = \frac{\sum_{\mathrm{C_i} \in \mathcal{C}} (\mathrm{bestmatch}(\mathrm{C_i}, \mathcal{C}'))}{|\mathcal{C}|} \ . \tag{13}$$

Eine Schwachstelle dieses Vergleichsmaßes ist in Abbildung 9 aufgezeigt. Vergleicht man die linke Clusterung $\mathcal{C}$ der Abbildung mit der rechten Clusterung $\mathcal{C}'$ erhält man

$$\mathrm{bestmatch_{av}}(\mathcal{C}, \mathcal{C}') = \frac{1 + 1 + \frac{3}{9}}{3} = \frac{7}{9} \approx 0,778 \ .$$

Die Clusterungen mit vielen kleinen übereinstimmenden Clustern erhalten immer einen hohen $\mathrm{bestmatch_{av}}$. Deshalb normieren wir die $\mathrm{bestmatch}$-Werte bezüglich der Clustergrößen der Cluster $\mathrm{C_i}$ und definieren den

$$\mathrm{bestmatch_{no}}(\mathcal{C}, \mathcal{C}') = \sum_{\mathrm{C_i} \in \mathcal{C}} \left(\frac{|\mathrm{C_i}|}{\mathrm{n}} \cdot \mathrm{bestmatch}(\mathrm{C_i}, \mathcal{C}') \right) \ . \tag{14}$$

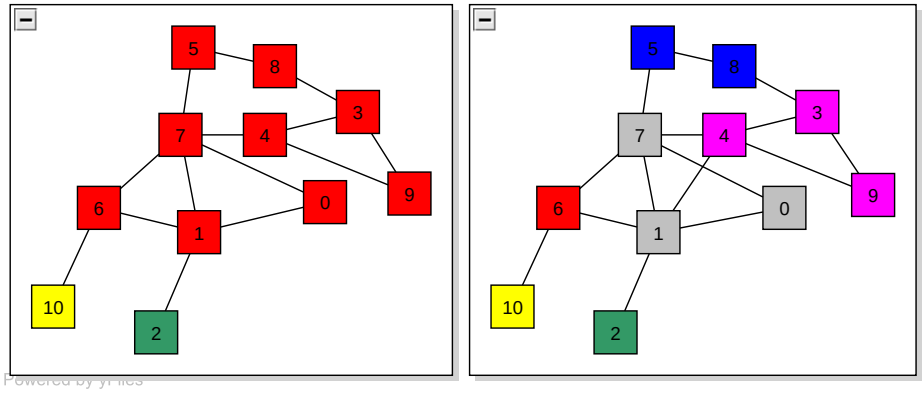


Abbildung 9: *Zwei verschiedene Clusterungen eines Graphen. Die linke Clusterung sei $\mathcal{C}$ und die rechte Clusterung sei $\mathcal{C}'$.*

Für das Beispiel in Abbildung 9 ist der

$$\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C},\mathcal{C}') = \frac{1}{11}\cdot 1 + \frac{1}{11}\cdot 1 + \frac{9}{11}\cdot\frac{3}{9} = \frac{5}{11} \approx 0,455 \ .$$

Der $\mathrm{bestmatch}_{\mathrm{av}}$ und der $\mathrm{bestmatch}_{\mathrm{no}}$ sind nicht symmetrisch. Das heißt $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C},\mathcal{C}') \neq \mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}',\mathcal{C})$. Eine mögliche symmetrische Variante ist gegeben durch:

$$\mathrm{bestmatch}_{\mathrm{sy}}(\mathcal{C},\mathcal{C}') = \frac{\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C},\mathcal{C}') + \mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}',\mathcal{C})}{2} \ . \tag{15}$$

Wir werden das Vergleichsmaß $\mathrm{bestmatch}_{\mathrm{no}}$ benutzen, um die Ähnlichkeit einer Clusterung $\mathcal{C}'$ zu einer Referenz-Clusterung $\mathcal{C}$ mittels $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C},\mathcal{C}')$ zu bestimmen. Wir verwenden nicht die symmetrische Variante, da es uns beim Vergleich von $\mathcal{C}'$ mit der Referenz-Clusterung vor allem darauf ankommt, dass es für jeden Referenzcluster eine möglichst genaue Entsprechung in der Clusterung $\mathcal{C}'$ gibt. Das Maß $\mathrm{bestmatch}_{\mathrm{no}}$ ähnelt dem nun folgenden *van Dongen*-Maß, ist im Gegensatz dazu aber ein Ähnlichkeitsmaß. Es hat den Wertebereich $[0,1]$ und liefert für sehr ähnliche Clusterungen einen Wert nahe 1.

2.8.2. Van Dongen

Das von van Dongen [vD00] vorgeschlagene Maß $\mathcal{VD}(\mathcal{C},\mathcal{C}')$ ist ein symmetrisches Schnittmaß. Auch hier berechnet sich der Wert aus den Überdeckungen der Cluster der beiden Clusterungen $\mathcal{C}$ und $\mathcal{C}'$. Der genaue Wert ergibt sich aus der Differenz von $2\mathrm{n}$ und der Summe der maximalen Schnittmengen aller Cluster beider Clusterungen $\mathcal{C}$ und $\mathcal{C}'$. Dabei sucht man ähnlich zum $\mathrm{bestmatch}$ für jeden Cluster $\mathrm{C_i}$ aus $\mathcal{C}$ den Cluster $\mathrm{C_j}$ in $\mathcal{C}'$ mit maximaler Schnittmenge $\mathrm{C_i} \cap \mathrm{C_j}$. Analog dazu sucht man für jeden Cluster $\mathrm{C_j}$ aus $\mathcal{C}'$ den Cluster $\mathrm{C_i}$ in $\mathcal{C}$ mit maximaler Schnittmenge $\mathrm{C_i} \cap \mathrm{C_j}$. Damit ergibt sich das van Dongen-Maß aus

$$\mathcal{VD}(\mathcal{C},\mathcal{C}') = 2\mathrm{n} - \sum_{\mathrm{C_i}\in\mathcal{C}} \max_{\mathrm{C_j}\in\mathcal{C}'} (|\mathrm{C_i}\cap\mathrm{C_j}|) - \sum_{\mathrm{C_j}\in\mathcal{C}'} \max_{\mathrm{C_i}\in\mathcal{C}} (|\mathrm{C_i}\cap\mathrm{C_j}|) \tag{16}$$

$$= \underbrace{\mathrm{n} - \sum_{\mathrm{C_i}\in\mathcal{C}} \max_{\mathrm{C_j}\in\mathcal{C}'} (|\mathrm{C_i}\cap\mathrm{C_j}|)}_{(1)} + \underbrace{\mathrm{n} - \sum_{\mathrm{C_j}\in\mathcal{C}'} \max_{\mathrm{C_i}\in\mathcal{C}} (|\mathrm{C_i}\cap\mathrm{C_j}|)}_{(2)} \quad .$$

Der vordere Teil der Summe (1) von $\mathcal{VD}(\mathcal{C},\mathcal{C}')$ ergibt sich aus der Differenz der Anzahl der Knoten und maximaler Überdeckung von $\mathcal{C}$ mit $|\mathcal{C}|$ Clustern von $\mathcal{C}'$. Analog dazu ergibt sich der hintere Teil der Summe (2). Dabei erfolgt die anschließend mögliche Normierung über die doppelte Anzahl der Knoten

$$\mathcal{NVD}(\mathcal{C},\mathcal{C}') = \frac{\mathcal{VD}(\mathcal{C},\mathcal{C}')}{2\mathrm{n}} \quad . \tag{17}$$

Das normierte van Dongen-Maß ist ein Distanzmaß mit Wertebereich $[0,1]$. Ähnliche Clusterungen erhalten daher Werte nahe 0.

2.8.3. Cosine-comparison

In 2.6 stellten wir das Ähnlichkeitsmaß Cosine-Similarity vor. Ausgehend davon definieren wir ein Vergleichsmaß für Clusterungen. Sieht man die Cluster $\mathrm{C_i}\in\mathcal{C}$ und $\mathrm{C_j}\in\mathcal{C}'$ eines Graphen $\mathcal{G}$ mit n Knoten als Vektoren aus $\{0,1\}^n$, dann haben die Vektoren an der Stelle i den Wert 1, falls Knoten v_i im Cluster enthalten ist, ansonsten eine 0. Die Cosine Similarity der Clustervektoren $\mathrm{v_{C_i}}$ und $\mathrm{v_{C_j}}$ entspricht

$$\mathrm{sim}(\mathrm{v_{C_i}},\mathrm{v_{C_j}}) = \frac{|\mathrm{C_i}\cap\mathrm{C_j}|}{\sqrt{|\mathrm{C_i}|\cdot|\mathrm{C_j}|}} \quad .$$

Daraus leiten wir das Maß $\mathrm{coscom_{max}}$ ab

$$\mathrm{coscom_{max}}(\mathrm{C_i},\mathcal{C}') = \max_{\mathrm{C_j}\in\mathcal{C}'} \left(\frac{|\mathrm{C_i}\cap\mathrm{C_j}|}{\sqrt{|\mathrm{C_i}|\cdot|\mathrm{C_j}|}} \right) \quad .$$

Das Vergleichsmaß *Cosine-comparison* $\mathrm{coscom_{no}}$ ergibt sich aus der $\mathrm{coscom_{max}}$ der Cluster $\mathrm{C_i}$ einer Clusterung $\mathcal{C}$ und der Clusterung $\mathcal{C}'$

$$\mathrm{coscom_{no}}(\mathcal{C},\mathcal{C}') = \sum_{\mathrm{C_i}\in\mathcal{C}} \left(\frac{|\mathrm{C_i}|}{\mathrm{n}} \cdot \mathrm{coscom_{max}}(\mathrm{C_i},\mathcal{C}')\right) \quad . \tag{18}$$

Das Maß Cosine-comparison ist ein Ähnlichkeitsmaß mit Wertebereich $[0,1]$.

2.8.4. Variation of Information

Mit Hilfe der *Entropie* aus der *Informationstheorie* lassen sich ebenfalls Vergleichsmaße für Clusterungen herleiten [Del06, Mei03]. Die Entropie einer Clusterung berechnen wir über die Quotienten aus der jeweiligen Clustergröße und der Gesamtknotenzahl n. Sie entspricht der Entropie nach Shannon und ergibt sich zu

$$\mathcal{H}(\mathcal{C}) = -\sum_{\mathrm{C_i}\in\mathcal{C}} \left(\frac{|\mathrm{C_i}|}{\mathrm{n}} \cdot \log_2 \left(\frac{|\mathrm{C_i}|}{\mathrm{n}} \right) \right) \ . \tag{19}$$

Dabei ist die Entropie $\mathcal{H}(\mathcal{C})$ ein Maß für den Informationsgehalt einer Clusterung. Die Unsicherheit, dass ein zufällig ausgewählter Knoten in einem bestimmten Cluster $\mathrm{C_i}$ der Clusterung $\mathcal{C}$ enthalten ist, steigt bei steigender Entropie. Die Entropie einer Clusterung hängt nicht von der Anzahl der Knoten, sondern von der Aufteilung der Cluster ab. Sind die Knoten gleichmäßig auf alle Cluster verteilt, ist die Unsicherheit maximal. Dabei ist der maximale Wert der Entropie einer Clusterung mit c Clustern $\log_2 \mathrm{c}$.

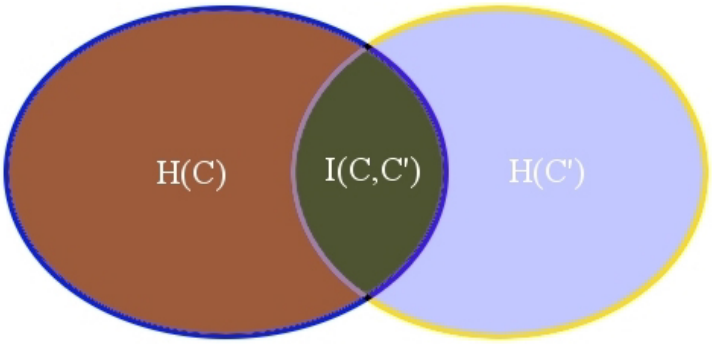


Abbildung 10: *Venn-Diagramm der Entropien zweier Clusterungen $\mathcal{C}$ und $\mathcal{C}'$. Der Anteil der Kreuzkorrelation ist in schwarz gehalten.*

Die Entropie der beiden Clusterungen $\mathcal{C}$ und $\mathcal{C}'$ allein erlaubt keine Aussage darüber, wie ähnlich die beiden Clusterungen sind. Die Kreuzkorrelation $\mathcal{I}(\mathcal{C}, \mathcal{C}')$ der beiden Clusterungen ist gegeben durch

$$\begin{aligned} \mathcal{I}(\mathcal{C}, \mathcal{C}') &= -\sum_{\mathrm{C_i}\in\mathcal{C}} \sum_{\mathrm{C_j}\in\mathcal{C}'} \left(\frac{|\mathrm{C_i} \cap \mathrm{C_j}|}{\mathrm{n}} \cdot \log_2 \left(\frac{\frac{|\mathrm{C_i}\cap\mathrm{C_j}|}{\mathrm{n}}}{\frac{|\mathrm{C_i}|}{\mathrm{n}} \cdot \frac{|\mathrm{C_j}|}{\mathrm{n}}} \right) \right) \\ &= -\frac{1}{\mathrm{n}} \sum_{\mathrm{C_i}\in\mathcal{C}} \sum_{\mathrm{C_j}\in\mathcal{C}'} \left(|\mathrm{C_i} \cap \mathrm{C_j}| \cdot \log_2 \left(\frac{|\mathrm{C_i} \cap \mathrm{C_j}|}{|\mathrm{C_i}| \cdot |\mathrm{C_j}|} \right) \right) \ . \end{aligned} \tag{20}$$

Sie berechnet sich aus den Schnittmengen der Cluster der beiden Clusterungen $\mathcal{C}$ und $\mathcal{C}'$. Meila [Mei03] definiert mit Hilfe der Entropie und der Kreuzkorrelation die *Variation of Information* zweier Clusterungen durch

$$\mathcal{VI}(\mathcal{C},\mathcal{C}') = \overbrace{(\mathcal{H}(\mathcal{C}) - \mathcal{I}(\mathcal{C},\mathcal{C}'))}^{\mathcal{H}(\mathcal{C}|\mathcal{C}')} + \overbrace{(\mathcal{H}(\mathcal{C}') - \mathcal{I}(\mathcal{C},\mathcal{C}'))}^{\mathcal{H}(\mathcal{C}'|\mathcal{C})}$$

$$\text{und normiert} \quad \mathcal{NVI}(\mathcal{C},\mathcal{C}') = \frac{\mathcal{H}(\mathcal{C}) + \mathcal{H}(\mathcal{C}') - 2\mathcal{I}(\mathcal{C},\mathcal{C}')}{\log_2(\mathrm{n})} \ . \tag{21}$$

Dabei steht $\mathcal{H}(\mathcal{C} \mid \mathcal{C}')$ für die Information, die wir beim Übergang von Clusterung $\mathcal{C}$ auf Clusterung $\mathcal{C}'$ verlieren. Und $\mathcal{H}(\mathcal{C}' \mid \mathcal{C})$ steht für den noch nötigen Informationszuwachs beim Übergang von Clusterung $\mathcal{C}$ auf Clusterung $\mathcal{C}'$ (siehe Abbildung 10). Der interessierte Leser sei auf [Mei03, Mei07] verwiesen. Die Variation of Information ist ein symmetrisches Distanzmaß, welches in der normierten Form den Wertebereich $[0, 1]$ hat.

Da die vier Vergleichsmaße bei unseren Testreihen meist sehr ähnliche Tendenzen aufweisen, geben wir häufig nur eines dieser Vergleichsmaße an.

3. Entwurf von zeitexpandierten Graphen

In dem Entwurf von zeitexpandierten Graphen sehen wir die Chance, eine möglichst exakte Wiedergabe der natürlichen Gruppen innerhalb eines Netzwerkes zu erreichen. Betrachten wir zum Beispiel die Entwicklung von Internet-Foren. Sei das Kantengewicht zweier Foren-Mitglieder die Anzahl der Themen (Threads), zu denen beide Mitglieder in einem gewissen Zeitfenster Beiträge beigesteuert haben, dann ist der resultierende dynamische Graph ständigen Veränderungen unterworfen. Erzeugt man aus diesem dynamischen Graphen einen zeitexpandierten Graphen, kann man nicht nur Veränderungen, sondern ebenfalls stabile Gruppen innerhalb des dynamischen Graphen erkennen. Das Netzwerk, auf dem wir unsere Testreihen durchführen, ist das E-Mail-Netzwerk der *Fakultät für Informatik* an der *Universität Karlsruhe (TH)*.

3.1. Das E-Mail-Netzwerk der Fakultät für Informatik

Innerhalb der *Fakultät für Informatik* an der *Universität Karlsruhe (TH)* werden seit über einem Jahr die Kontakt-Daten von E-Mails, die *intern* verschickt werden, aufgezeichnet. Dabei wird außer dem Datum und dem E-Mail-Account der Sender und Empfänger zusätzlich deren *LehrstuhlID* gespeichert. Aus Datenschutzgründen werden die Daten anonymisiert und die Originaldaten anschließend vernichtet. Das heißt, jeder E-Mail-Account und jede LehrstuhlID hat einen eindeutigen Identifier, der von uns nicht mehr zurückverfolgt werden kann. E-Mails werden in der Form

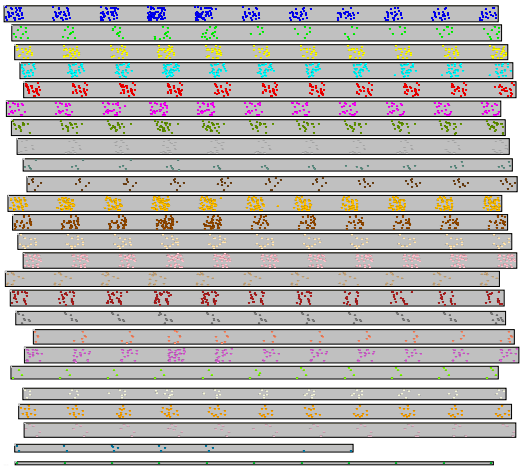
```
Datum, AccountID.InstitutID(Absender), AccountID.InstitutID(Empfänger)
```

oder, falls mehrere Empfänger vorhanden sind, in der Form

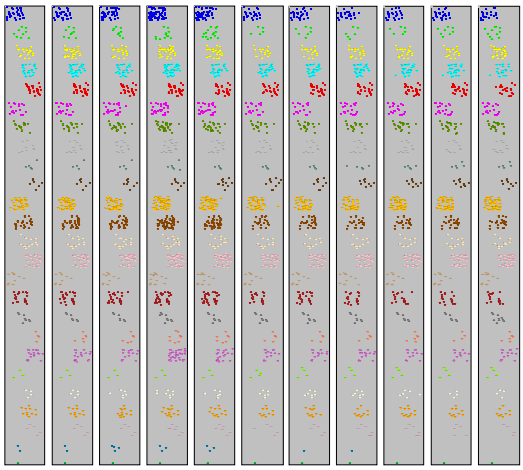
```
Datum, AccountID.LehrstuhlID(Absender), AccountID.LehrstuhlID(Empfänger 1)
Datum, AccountID.LehrstuhlID(Absender), AccountID.LehrstuhlID(Empfänger 2)
...
Datum, AccountID.LehrstuhlID(Absender), AccountID.LehrstuhlID(Empfänger X)
```

gespeichert, wenn sie von einem Account der Fakultät an einen anderen Account der Fakultät geschickt werden. Das heißt, anstatt einer Zeile pro E-Mail bei mehreren Empfängern erzeugen wir für jeden Empfänger eine Zeile. Das Ergebnis ist eine Menge von Sender-Empfänger-Paaren. Die E-Mails eines Accounts an sich selbst tragen keine sinnvolle Information zur Clusterung der Accounts bei. Sie werden meist zur Erinnerung, Datensicherung oder Überprüfung, ob eine E-Mail wirklich verschickt wurde, genutzt. Ist einer der Empfänger der Absender, wird für ihn deshalb keine Zeile erzeugt. Aus den gesammelten Daten einer Zeitspanne von 308 Tagen wollen wir unsere zeitexpandierten Graphen erzeugen. Die Knotenmenge $\mathcal{V}$ besteht aus allen in den Daten vorkommenden Identifiern *AccountID.LehrstuhlID*. Für unseren Untersuchungszeitraum enthält die Menge $\mathcal{V}$ 785 Knoten. Wir unterteilen die Zeitspanne in elf Zeitschritte zu je 28 Tagen. Diese Zeitspanne ist nicht willkürlich gewählt, sondern aufgrund des deutlich geringeren E-Mail-Aufkommens während der Wochenenden ein Vielfaches der Anzahl der sieben Wochentage. Für jeden dieser elf Zeitschritte erzeugen wir eine Matrix $\mathcal{A}(\mathrm{t}) \in \mathbb{N}^{785\times785}$, deren Einträge sich aus der Anzahl der ausgetauschten E-Mails ergeben. Die resultierenden symmetrischen Matrizen haben an der Stelle $\mathcal{A}(\mathrm{t})_{\mathrm{i,j}}$ den Eintrag x, wobei x die Anzahl aller Sender-Empfänger-Paare bestehend aus Knoten v_{i} und v_{j} in Zeitschritt t ist. Diese Matrizen sind im Folgenden Ausgangspunkt aller zeitexpandierten Graphen. Der Aufwand zur Erzeugung dieser Matrizen liegt in $\Theta(\mathrm{M})$, wobei M die Anzahl aller Sender-Empfänger-Paare ist.

Die unterschiedlichen Gruppierungen, die mit Hilfe der LehrstuhlID angegeben werden, stehen für unterschiedliche Lehrstühle oder zentrale Einrichtungen. In der Regel bilden zwei bis fünf Lehrstühle ein Institut. Dies ist aus den Daten durch die Anonymisierung allerdings nicht ersichtlich. In dem untersuchten Zeitraum gab es 26 verschiedene Gruppen innerhalb des internen E-Mail-Netzwerkes. Die Vorgabe der LehrstuhlID für jeden Repräsentanten des zeitexpandierten Graphen kann als Clusterung aufgefasst werden. Diese bezeichnen wir als Institute-Clusterung. Zur besseren Lesbarkeit haben wir die verschiedenen Zeitschritte, Knoten und Lehrstühle beginnend mit 0 durchnummeriert. Dadurch ergeben sich die Namen für die Repräsentanten der Knoten in den zeitexpandierten Graphen aus der Knotennummer, der Lehrstuhlnummer und der Nummer des Zeitschrittes. Dabei ist *12$19#2* der Name des Repräsentanten von Knoten 12 aus dem Lehrstuhl 19 der Institute-Clusterung in Zeitschritt 2. In den Abbildungen erfolgt die Anordnung der Lehrstühle von oben nach unten (siehe Abbildung 11a), beginnend mit Lehrstuhl 0. Die Zeitschritte werden chronologisch von links nach rechts angeordnet (siehe Abbildung 11b). Eine Clusterung der Knoten gemäß der in 11b angegebenen Zeitschritte nennen wir die Zeit-Clusterung $\mathcal{C}_{\text{time}}$ des zeitexpandierten Graphen. Die Anordnung der Knoten innerhalb einer Gruppe ist zur besseren Interpretierbarkeit für alle Zeitschritte identisch. Aus Gründen der Übersichtlichkeit sind die Kanten aus den Bildern der zeitexpandierten Graphen entfernt.



(a) Die Kästen entsprechen den verschiedenen Lehrstühlen der Institute-Clusterung.



(b) Die Kästen entsprechen den verschiedenen Zeitschritten.

Abbildung 11: *In der linken Abbildung repräsentieren die Kästen die verschiedenen Lehrstühle der Institute-Clusterung. In der rechten Abbildung stehen die Kästen für die verschiedenen Zeitschritte innerhalb des zeitexpandierten Graphen. Die Farben der Knoten stehen in beiden Abbildungen für die Zugehörigkeit zu den verschiedenen Lehrstühlen.*

3.2. Ablauf

Auf Grundlage des in 2.2 vorgestellten Modells werden wir gewichtete zeitexpandierte Graphen $\overline{\mathcal{G}}^{\text{d}}_{\text{k,p}}$ aus unseren E-Mail-Daten generieren und untersuchen, wie sich Veränderungen im Aufbau des Graphen auf die verwendeten Cluster-Verfahren und die Eigenschaften des Graphen auswirken. Ähnlich zum *Algorithm Engineering* werden wir unser Vorgehen in vier Phasen unterteilen (siehe Abbildung 12). Die *Design-Phase*, in der das Modell sowie die verwendeten Methoden

des Modells ausgewählt werden. In dieser Phase werden zusätzlich die Wertebereiche der Parameter und verwendete Cluster-Verfahren festgelegt. In der folgenden *Analyse-Phase* werden die Vor- und Nachteile des Modells und der verwendeten Methoden erörtert. Ebenso betrachten wir die Laufzeit für den Aufbau und die Clusterung der Graphen. Während der *Implementierungs-Phase* erfolgt die Implementierung in Java unter Benutzung der *yfiles* der yWorks GmbH und eines Clustering-Frameworks (siehe Abschnitt 3.2.3), das am *Institut für Theoretische Informatik* an der Universität Karlsruhe (TH) entwickelt wurde. Abschließend werden die zeitexpandierten Graphen in der *Experimente-Phase* erzeugt. Die erzeugten Graphen werden mit den in der Design-Phase gewählten Verfahren geclustert. Anhand von Kennwerten, Qualitätsindizes und Vergleichsmaßen werden die Graphen und Clusterungen analysiert und wir ziehen Rückschlüsse bezüglich unseres verwendeten Modells. Schließlich gehen wir wieder in die Design-Phase über.

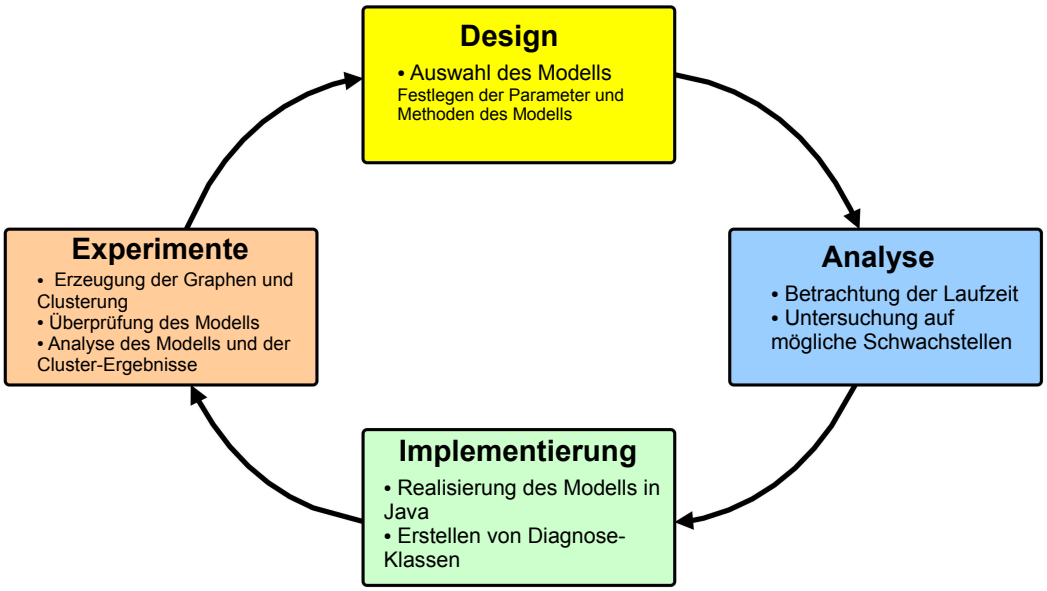


Abbildung 12: *Vorgehen beim Entwurf.*

3.2.1. Design

In der Design-Phase werden die benutzen Methoden und Parameter festgelegt. Die Anzahl der unterschiedlichen Knoten $\mathcal{V}$ des zugrundeliegenden dynamischen Graphen sei n. Dabei gehen wir davon aus, dass für jeden Zeitschritt t_0 eine Adjazenzmatrix $\mathcal{A}(t_0) \in \mathbb{R}^{n \times n}$ mit den durch die Anwendung vorgegebenen Kantengewichten gegeben ist. Ein Beispiel hierfür wären die verschiedenen Momentaufnahmen des Transportaufkommens in einem Transportnetzwerk. Der Eintrag $\mathcal{A}(t_0)_{i,j}$ der Matrix $\mathcal{A}(t_0)$ für den Zeitpunkt t_0 enthält dabei das Transportaufkommen zwischen Knoten v_i und v_j zum Zeitpunkt der Momentaufnahme. In unserer Anwendung sind es die zuvor

beschriebenen Matrizen $\mathcal{A}(\mathrm{t})$ (siehe Abschnitt 3.1), die aus den Sender-Empfänger-Paaren unserer E-Mail-Daten erzeugt werden. Im Folgenden gehen wir stets von positiven Kantengewichten aus. Die Kantengewichte eines zeitexpandierten Graphen können daraus auf verschiedene Arten berechnet werden. Die Methoden für die Intrazeitkantengewichte sind:

Normal Das heißt, die Intrazeitkantengewichte entsprechen den Einträgen der Matrizen $\mathcal{A}(\mathrm{t})$. Es ist offensichtlich, dass diese Methode keinen zusätzlichen Aufwand benötigt, da die Matrizen $\mathcal{A}(\mathrm{t})$ unverändert bleiben.

Normed Hier erfolgt eine logarithmische Normierung bezüglich des maximalen Kantengewichtes. Die neuen Kantengewichte ergeben sich aus

$$\omega_{\mathrm{no}}(\mathrm{e}) = \frac{\mathrm{log1p}(\omega(\mathrm{e}))}{\mathrm{log1p}(\max_{\mathrm{e_i}\in\mathcal{E}}(\omega(\mathrm{e_i})))} \text{ mit } \mathrm{log1p}(x) = \ln(x+1).$$

Der Aufwand der Normierung der Kanten eines Zeitschrittes ist $\mathcal{O}(\mathrm{n}^2)$. Daraus ergibt sich ein Gesamtaufwand der Normierung für alle d Zeitschritte von $\mathcal{O}(\mathrm{n}^2\mathrm{d})$.

Cosine Mit Hilfe der in Abschnitt 2.6.1 vorgestellten Adapted-Cosine-Similarity wird aus jeder Matrix $\mathcal{A}(\mathrm{t}_0)$ eine Cosine Similarity Matrix $\mathcal{A}_{\mathrm{sim}}(\mathrm{t}_0)$ generiert. Dabei ergibt sich das Gewicht der Kante zwischen den Knoten $v_{\mathrm{i}}^{\mathrm{t}_0}$ und $v_{\mathrm{j}}^{\mathrm{t}_0}$ durch $\mathrm{sim}_{\mathrm{ad}}(v_{\mathrm{i}}^{\mathrm{t}_0}, v_{\mathrm{j}}^{\mathrm{t}_0})$. Dies ist ein sehr aufwändiges Verfahren. Der Aufwand zur Berechnung einer Kante mit der Adapted-Cosine-Similarity ist $\mathcal{O}(\mathrm{n})$. Für die Berechnung aller Werte einer quadratischen Cosine Similarity Matrix $\mathcal{A}_{\mathrm{sim}}$ mit n Knoten liegt der Aufwand damit in $\mathcal{O}(\mathrm{n}^3)$. Daraus ergibt sich ein Gesamtaufwand für die Cosine-Methode für alle d Zeitschritte von $\mathcal{O}(\mathrm{n}^3\mathrm{d})$.

Wir beschränken uns bei unseren Testreihen auf diese Methoden. Die Gewichte von *Cosine*, *Normed* und *Normal* haben unterschiedliche Bedeutungen. Die normalen Kantengewichte sind die unveränderten Kantengewichte, ihre Bedeutung ergibt sich aus der Anwendung. In unserem Fall entsprechen sie der Anzahl der ausgetauschten E-Mails im betreffenden Zeitschritt. Bei dieser Methode ist es ohne zusätzliche Informationen nicht möglich, das Gewicht einer Kante zu beurteilen.

Bei der Methode Normed werden die Gewichte logarithmisch normiert. Das bedeutet, dass das maximale Kantengewicht den Wert 1 erhält, während alle anderen Gewichte an diesem Gewicht gemessen werden. Der Wertebereich der Kantengewichte wird also auf das Intervall $[0, 1]$ skaliert. Die Abbildung der Gewichte ist streng monoton, je größer das ursprüngliche Gewicht, desto größer der neue Kantenwert. Dadurch erreicht man eine bessere Vergleichbarkeit der Gewichte. Durch die logarithmische Normierung verringern wir die Auswirkungen von Ausreißern innerhalb der Gewichte.

Das Kantengewicht zweier Endknoten bei der Cosine-Methode ist im Gegensatz dazu genau dann nahe am maximalen Wert 1, wenn die beiden Knotenvektoren, die sich aus den Spaltenvektoren der Matrizen $\mathcal{A}(\mathrm{t})$ ergeben, in eine ähnliche Richtung zeigen. Anders ausgedrückt erhält das Kantengewicht einen hohen Wert, falls die beiden Endknoten eine ähnliche Ausrichtung bezüglich ihrer Nachbarschaft haben. Selbst Knoten ohne gemeinsame Kante im ursprünglichen Graphen können aufgrund ihrer ähnlichen Nachbarschaft ein hohes Kantengewicht erhalten (siehe rote Kante in Abbildung 13).

Aufgrund der unterschiedlichen Bedeutung der Normed- und der Cosine-Gewichte führen wir eine weitere Methode zur Berechnung der Intrazeitkantengewichte ein.

Mixed Bei Mixed wird das Kantengewicht von Normed $\mathrm{w}_{\mathrm{e}_{\mathrm{normed}}}$ und Cosine $\mathrm{w}_{\mathrm{e}_{\mathrm{cosine}}}$ zu einem neuen Kantengewicht kombiniert. Die Berechnung der Gewichte erfolgt über die Konvexkombination $\mathrm{w}_{\mathrm{e}_{\mathrm{mix}}} = \lambda \mathrm{w}_{\mathrm{e}_{\mathrm{cosine}}} + (1-\lambda)\mathrm{w}_{\mathrm{e}_{\mathrm{normed}}}$ mit $\lambda \in [\,0, 1]$.

Der Aufwand von $\mathcal{O}(\mathrm{n}^3\mathrm{d})$ für diese Methode wird durch die Berechnung der Cosine Similarity Matrizen der d Zeitschritte bestimmt. Durch die Konvexkombination kann man die Anteile der Normed- und der Cosine-Gewichte am Gesamtgewicht steuern. Dies ist dann hilfreich, wenn beides, die Ähnlichkeit zweier Knoten und das ursprüngliche Kantengewicht der Anwendung für die Einordnung der Knoten von Bedeutung ist.

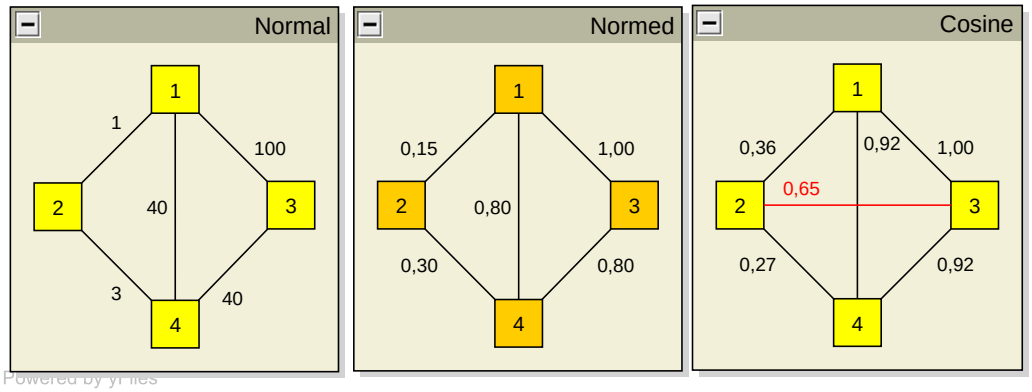


Abbildung 13: *Die Kantengewichte von Normal, Normed und Cosine gerundet auf zwei Dezimale für das Beispiel aus 2.6.1.*

Ein Beispiel für die unterschiedlichen Möglichkeiten zur Berechnung der Intrazeitkanten ist in Abbildung 13 zu sehen. Bei den in 2.2 vorgestellten Varianten für den Verlauf der Interzeitkanten beschränken wir uns auf die *Variante 1*, jeder Knoten wird mit den Knoten, die denselben Account repräsentieren, verbunden, sofern sie in seiner Reichweite liegen. Für die Berechnung der Interzeitkanten unterscheiden wir folgende Methoden:

Alpha Alle existierenden Interzeitkanten erhalten einen fixen Wert α. Der Aufwand zur Erzeugung der Interzeitkanten liegt hierbei für ein festes $\mathrm{k} \leq \mathrm{d}$ in $\mathcal{O}(\mathrm{ndk})$.

Cosine-Time Bei dieser Variante werden die Interzeitkantengewichte über die in Abschnitt 2.6 vorgestellte Cosine Similarity berechnet. Zwei Knoten $v_\mathrm{i} \in \mathcal{V}_{\mathrm{t}_\mathrm{i}}$ und $v_\mathrm{j} \in \mathcal{V}_{\mathrm{t}_\mathrm{j}}$ werden über eine Interzeitkante verbunden, falls sie den selben Knoten des zu Grunde liegenden dynamischen Graphen $\mathcal{G}(\mathrm{t})$ repräsentieren und $\mathcal{G}(\mathrm{t}_\mathrm{i}) \in \mathrm{ver}(\mathcal{G}_{\mathrm{t}_\mathrm{j}}, \mathrm{k})$. Das Gewicht der Kante berechnet sich aus $\mathrm{sim}(\mathrm{v}_\mathrm{i}, \mathrm{v}_\mathrm{j})$, wobei v_i der Spaltenvektor von v_i in der Matrix $\mathcal{A}(\mathrm{t}_\mathrm{i})$ und v_j der Spaltenvektor von v_j in der Matrix $\mathcal{A}(\mathrm{t}_\mathrm{j})$ ist. Der Aufwand zur Berechnung einer Interzeitkante mit der Cosine-Similarity liegt in $\mathcal{O}(\mathrm{n})$. Daraus ergibt sich ein Gesamtaufwand zur Erzeugung der Interzeitkanten für ein festes $\mathrm{k} \leq \mathrm{d}$ von $\mathcal{O}(\mathrm{n}^2\mathrm{dk})$.

Bei unseren Testreihen beschränken wir uns auf diese beiden Methoden, da sie sowohl den Fall statischer Interzeitkanten, wie auch den Fall dynamischer Zeitkanten abdecken. Die Verwendung der Methode Alpha erfordert wenig Aufwand, allerdings tragen die Interzeitkantengewichte keine nützliche Information zur Clusterung der Daten bei. Durch die Verwendung der Methode Cosine-Time enthält der zeitexpandierte Graph zusätzliche Information über die Ähnlichkeit der Repräsentanten der einzelnen Zeitschritte.

Die Parameter eines zeitexpandierten Graphen $\overline{\mathcal{G}}^{\mathrm{d}}_{\mathrm{k,p}}$ sind die Anzahl der Zeitschritte d, die Reichweite der Interzeitkanten k und die Schwelle p. Bei unseren Untersuchungen haben alle unsere Graphen die feste Anzahl von elf Zeitschritten. Die veränderlichen Parameter der Graphen sind die Reichweite k, Schwelle p und bei fixen Interzeitkantenwerten der Parameter α. Wenn die Mixed-Kantenberechnung verwendet wird, kommt zusätzlich der Parameter λ hinzu. Die Schwelle p verringert die Dichte innerhalb des Graphen. Sie filtert unwichtige oder auch verrauschte Daten heraus und reduziert die Anzahl der Kanten. Die Reichweite k bestimmt die Sichtweite der Interzeitkanten.

Für unsere Anwendung erfolgt die Wahl der Kantenmengen $\mathcal{E}_{\mathrm{t}}$ und Knotenmengen $\mathcal{V}_{\mathrm{t}}$ aufgrund der zum jeweiligen Zeitpunkt existierenden Kanten des E-Mail-Netzwerkes. Sei $\mathcal{A}(\mathrm{t})$ die Matrix des E-Mail-Netzwerkes zum Zeitpunkt t, dann ergibt sich die Knotenmenge $\mathcal{V}_{\mathrm{t}}$ aus

$$\mathcal{V}_{\mathrm{t}} = \left\{ v_{\mathrm{i}}^{\mathrm{t}} \mid v_{\mathrm{i}} \in \mathcal{V} \wedge (\exists \mathrm{j} \in \{1, \ldots, \mathrm{n}\} : (v_{\mathrm{j}} \in \mathcal{V} \wedge \mathcal{A}(\mathrm{t})_{\mathrm{i,j}} > 0)) \right\}.$$

Die Kantenmenge ergibt sich analog aus

$$\mathcal{E}_{\mathrm{t}} = \{ \mathrm{e} = \{v_{\mathrm{i}}, v_{\mathrm{j}}\} \mid \exists \mathrm{i}, \mathrm{j} \in \{1, \ldots, \mathrm{n}\} : v_{\mathrm{i}}, v_{\mathrm{j}} \in \mathcal{V} \wedge \mathcal{A}(\mathrm{t})_{\mathrm{i,j}} > 0 \}.$$

Wenn die Kante $\mathrm{e} \in \mathcal{E}_{\mathrm{t}_0}$ mit $\mathrm{e} = \{v_{\mathrm{i}}^{\mathrm{t}_0}, v_{\mathrm{j}}^{\mathrm{t}_0}\}$ in Zeitpunkt t_0 existiert, dann sind die Knoten $v_{\mathrm{i}}^{\mathrm{t}_0}$ und $v_{\mathrm{j}}^{\mathrm{t}_0}$ in der Menge $\mathcal{V}_{\mathrm{t}_0}$ enthalten. Gibt es einen Zeitpunkt t_0 zu dem keine Kante $\mathrm{e} \in \mathcal{E}_{\mathrm{t}_0}$ inzident zu Repräsentant $v_{\mathrm{i}}^{\mathrm{t}_0}$ ist, so ist $v_{\mathrm{i}}^{\mathrm{t}_0}$ nicht in der Menge $\mathcal{V}_{\mathrm{t}_0}$ enthalten. Würden wir für alle Zeitschritte die Mengen $\mathcal{V}_{\mathrm{t}} = \mathcal{V}$ setzen, gäbe es in jedem Zeitschritt eine Menge von isolierten Knoten, die die Anzahl der Knoten des zeitexpandierten Graphen vergrößern aber keine zusätzlichen Informationen liefern würden. Das Ergebnis einer Clusterung wäre dadurch möglicherweise verzerrt. Durch die von uns getroffene Wahl kann man für jeden Knoten feststellen, in welchen Zeitschritten er am E-Mail-Verkehr des Netzwerkes beteiligt war.

Bei den verwendeten Cluster-Verfahren beschränken wir uns auf die in 2.5 vorgestellten Verfahren Greedy-Significance-Clustering, Iterative-Conductance-Cutting und Markov-Clustering. Sei $\mathrm{n_d}$ die Anzahl aller Knoten des zeitexpandierten Graphen. Die Laufzeit des Iterative-Conductance-Cutting ist polylogarithmisch in $\mathrm{n_d}$ mit $\mathrm{n_d} \leq \mathrm{nd}$. Für die Laufzeit des Greedy-Significance-Clustering erhalten wir $\mathcal{O}(\mathrm{n_d^2} \log \mathrm{n_d})$. Analog dazu ergibt sich der Aufwand für das Markov-Clustering der zeitexpandierten Graphen mit Pruning-Faktor κ aus $\mathcal{O}(\mathrm{n_d}\kappa^2)$.

3.2.2. Analyse

In der Analyse betrachten wir die Laufzeit der Erzeugung und Clusterung der Graphen. Wir diskutieren mögliche Schwachstellen und Vorteile der verwendeten Methoden, sowie die Bedeutung bzw. Unterschiede der Kantengewichte. Des Weiteren stellen wir Hypothesen auf, die in der Experimente-Phase überprüft werden und wählen eine Referenz-Clusterung für unsere erzeugten Graphen.

3.2.3. Implementierung

Ausgehend von unserem Modell haben wir ein Framework implementiert, welches uns ermöglicht, ganze Serien von Graphen für die vorgestellten Methoden zu generieren und zu clustern. Bei der Realisierung benutzen wir das Clustering-Framework und die yfiles-Bibliotheken. Das am ITI Wagner entwickelte Clustering-Framework stellt grundlegende Datenstrukturen und Methoden für den Umgang mit geclusterten Graphen zur Verfügung. Es ermöglicht die Manipulation von Clusterungen, wie die Vereinigung zweier Cluster. Außerdem liefert das Clustering-Framework eine Implementierung der hier benutzen Cluster-Verfahren und Indizes.

Unser Framework ermöglicht die genaue Festlegung der Wertebereiche für die variablen Parameter und der verwendeten Methoden zur Erzeugung der zeitexpandierten Graphen unseres E-Mail-Netzwerkes. Außerdem erlauben uns einige Hilfsklassen die Berechnung von Kennwerten, Qualitätsindizes und Vergleichsmaßen, die in Log-Dateien gespeichert werden. In den einzelnen Testreihen werden wir nicht mehr auf die Implementierung eingehen.

3.2.4. Experimente

In dieser Phase erfolgt die Generierung der Graphen. Dabei erzeugen wir zunächst die Matrizen $\mathcal{A}(\mathrm{t})$ der einzelnen Zeitschritte. Aufgrund der in der Design-Phase gewählten Methoden zur Berechnung der Gewichte werden aus diesen Matrizen die neuen Kantenwerte berechnet. Die daraus entstehenden zeitexpandierten Graphen werden mit den verschiedenen Cluster-Verfahren geclustert. Die gefundenen Clusterungen werden mit der Referenz-Clusterung verglichen und es wird untersucht, wie sich Veränderungen der Parameter auf das Ergebnis der Clusterung auswirken. Wir versuchen gewünschte, wie auch unerwünschte Zusammenhänge zu erkennen. Die in der Analyse-Phase aufgestellten Hypothesen werden überprüft und an die Ergebnisse angepasst. In Kapitel 7 werden wir die Ergebnisse zusammenfassen und eine Gegenüberstellung von zeitexpandiertem Clustern und dem Clustern der einzelnen Ausprägungen vornehmen.

3.3. Ziele

Die Ziele der Testreihen sind zum einen ein besseres Verständnis des E-Mail-Netzwerkes innerhalb der Fakultät für Informatik und zum anderen eine Bewertung unserer Ergebnisse bezüglich des Modells eines zeitexpandierten Graphen. Wir wollen mit Hilfe des Modells Veränderungen innerhalb von Netzwerken erkennen. Über die Clusterung erhoffen wir uns eine Zuordnung der Repräsentanten der verschiedenen Zeitschritte zu bestimmten Gruppierungen innerhalb des Graphen. Anhand dieser Clusterungen sollten spezielle Veränderungen innerhalb eines Graphen ablesbar sein.

3.4. Beispiele für Veränderungen innerhalb eines dynamischen Graphen

Wir erwarten von unserem Modell spezielle Eigenschaften. So soll der zeitexpandierte Graph die Veränderungen des dynamischen Graphen erfassen und für die Cluster-Verfahren interpretierbar machen. Hierzu wollen wir im Folgenden einige typische Beispiele für die Entwicklung von Gruppen innerhalb eines Graphen vorstellen.

3.4.1. Spaltung einer Gruppe

Innerhalb eines dynamischen Graphen kann es einzelne Gruppen geben, die aufgrund von wegfallenden Kanten oder zentralen Knoten des dynamischen Graphen in mehrere Gruppen zerfallen. Ein Beispiel für eine solche Spaltung einer Gruppe ist in Abbildung 14 zu beobachten.

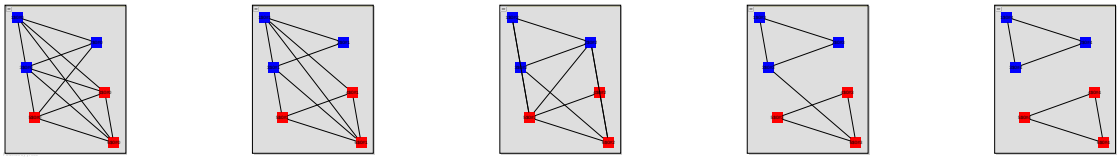


Abbildung 14: *Beispiel für die Spaltung einer Gruppe. Es sind die Teilgraphen für die fünf Zeitschritte 0 bis 4 angegeben. Sämtliche Kanten haben das Gewicht* $0,7$.

Die beiden 3-Cliquen sind zunächst stark verknüpft. Dieser Zusammenhang reduziert sich, bis in Zeitschritt 4 keine Kanten zwischen den beiden Cliquen mehr existieren. Analog dazu, nur in umgekehrter Reihenfolge der Zeitschritte, könnte die Vereinigung zweier Gruppen erfolgen.

3.4.2. Umbruch einer Gruppe

Ein weiteres typisches Beispiel für die Veränderungen innerhalb eines Graphen ist der Umbruch einer Gruppe. Ähnlich zur Spaltung trennt sich ein Teil der Gruppe von ihr ab, aber gleichzeitig kommen neue Knoten zur Gruppe hinzu. In Abbildung 15 sehen wir einen solchen Umbruch einer Gruppe. Zunächst bilden die rote und die grüne Clique eine Gruppe, deren Zusammenhalt sich langsam auflöst, währenddessen sich die blaue Clique der grünen annähert.

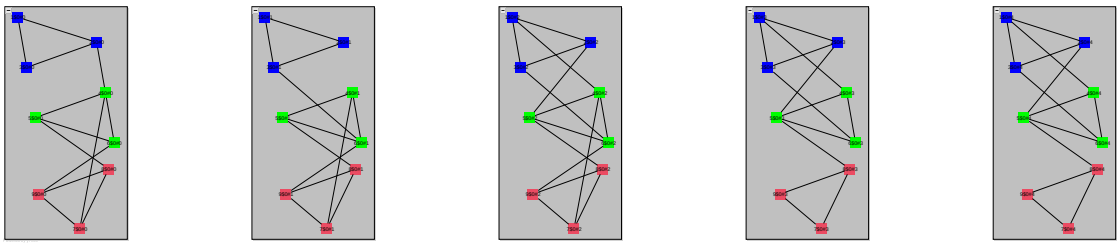


Abbildung 15: *Beispiel für den Umbruch einer Gruppe. Es sind die Teilgraphen für die fünf Zeitschritte 0 bis 4 angegeben. Sämtliche Kanten haben das Gewicht* $0,7$.

3.4.3. Zeitlich begrenztes Abweichen von der Norm

Viele Gruppen verändern ihr Verhalten nur wenig über die Zeit. In Folge von verrauschten oder fehlerhaften Daten eines Zeitschrittes kann es zu einer zeitlich begrenzten Abweichung des Verhaltens von der Norm kommen. Denkbar wären hier ebenfalls spezielle Ereignisse, die zu einer starken Abweichung führen. Sind zum Beispiel in unserem E-Mail-Netzwerk mehrere Besitzer zentraler Accounts einer der Gruppierungen im Urlaub, liefert die Clusterung dieses einzelnen Zeitschrittes unter Umständen ein völlig anderes Ergebnis, als die der anderen Zeitschritte. Ein

ähnliches Beispiel ist in Abbildung 16 zu sehen. Die drei 3-Cliquen sind eng miteinander verknüpft. Die einzigen Abweichungen von diesem Zusammenhang sind in den Zeitschritten 2 und 3 zu finden. Hier gibt es deutlich weniger Kanten zwischen diesen drei Cliquen. In Zeitschritt 2 gibt es keine Kanten von der magentafarbenen Clique zu den anderen beiden. Und in Zeitschritt 3 gilt das Gleiche für die blaue Clique. Die rote 4-Clique, die ansonsten keine Kanten zu den anderen Cliquen hat, hat in diesen Zeitschritten einige Kanten, die sie eng mit der blauen Clique und in Zeitschritt 2 zusätzlich mit der grünen Cliquen verbindet. In den Zeitschritten 4 und 5 entspricht die Situation den ersten beiden Zeitschritten.

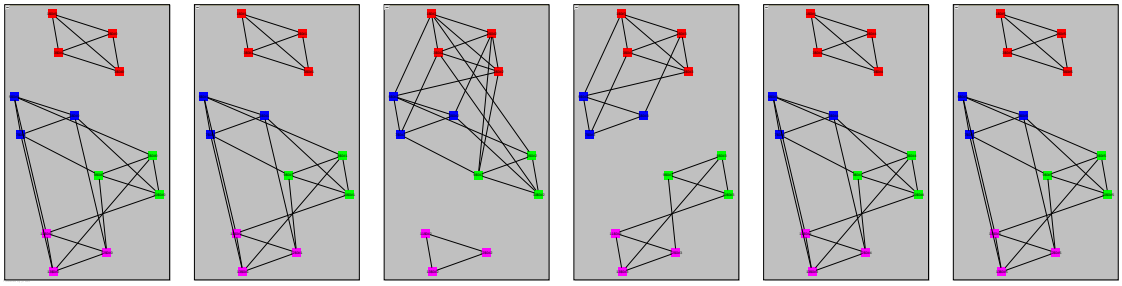


Abbildung 16: *Beispiel für die kurzeitige Änderung eines ansonsten gleichbleibenden Verhaltens. Es sind die Teilgraphen für die sechs Zeitschritte 0 bis 5 angegeben. Sämtliche Kanten haben das Gewicht* $0,7$.

Natürlich sind dies nicht die einzig möglichen Veränderungen innerhalb eines Graphen. Denkbar wären weitere Szenarien, wie beispielsweise das Auftauchen bzw. die Auflösung einer Gruppe.

Die aufgeführten Beispiele deuten darauf hin, welche Chancen wir in der Clusterung von zeitexpandierten Graphen sehen. Unser Ziel ist es zu zeigen, dass mit Hilfe der zeitexpandierten Graphen eine gute Dokumentation der zeitlichen Entwicklung des dynamischen Graphen möglich ist. In den folgenden Kapiteln werden wir einige Testreihen durchführen, deren Ergebnisse wir anschließend in Kapitel 7 bewerten und interpretieren. In diesem Zusammenhang werden wir noch einmal auf die vorgestellten Beispiele eingehen (siehe Abschnitt 7.3).

4. Beginn der Testreihen: Die Methode Normal

In diesem Kapitel werden wir mit Hilfe der Normal-Methode zeitexpandierte Graphen des E-Mail-Netzwerkes erzeugen.

4.1. Design

Zu Beginn wählen wir den denkbar einfachsten Aufbau eines zeitexpandierten Graphen mit den Methoden Normal und Alpha aus unserem Modell. Dabei entspricht jedes Intrazeitkantengewicht der Anzahl der ausgetauschten E-Mails der beiden Knoten. Wir begründen unsere Wahl mit folgender Annahme: je mehr E-Mails in einem Zeitbereich ausgetauscht werden, desto stärker ist auch die Zusammenarbeit bzw. der soziale Kontakt der beiden Account-Besitzer. Die meisten E-Mail-Accounts werden in den verschiedenen Zeitschritten ein ähnliches E-Mail-Verhalten aufweisen. Daher wählen wir bei dieser ersten Testreihe ein fixes Interzeitkantengewicht α. Es ist unwahrscheinlich, dass die verschiedenen Lehrstuhl-Mitglieder hauptsächlich Kontakte außerhalb des Lehrstuhls pflegen. Weiterhin vermuten wir eine geringere Dynamik in unserem Graphen, so dass ein fixer Wert α einen kleineren Fehler verursacht, als bei einem Graphen mit hoher Dynamik.

Wir veranschaulichen die Struktur eines zeitexpandierten Graphen mit d Zeitschritten durch die gewichtete Adjazenzmatrix

$$\mathcal{A}^{\omega}_{\mathrm{d}} = \begin{pmatrix} \mathcal{A}(\mathrm{t}_1) & \mathcal{D}(1,2) & \dots & \mathcal{D}(1,\mathrm{d}) \\ \mathcal{D}(1,2) & \mathcal{A}(\mathrm{t}_2) & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathcal{D}(\mathrm{d}-1,\mathrm{d}) \\ \mathcal{D}(1,\mathrm{d}) & \dots & \mathcal{D}(\mathrm{d}-1,\mathrm{d}) & \mathcal{A}(\mathrm{t}_{\mathrm{d}}) \end{pmatrix}.$$

Dabei ergibt sich die Matrix $\mathcal{A}^{\omega}_{\mathrm{d}}$ des zeitexpandierten Graphen für unser E-Mail-Netzwerk aus den im Kapitel 3.1 beschriebenen Matrizen $\mathcal{A}(\mathrm{t})$ und den Diagonalmatrizen $\mathcal{D}(\mathrm{i},\mathrm{j})$. Deren Diagonalelemente sind die Kantengewichte der Interzeitkanten zwischen zeitlich benachbarten Repräsentanten $v^{\mathrm{i}}_{\mathrm{x}} \in \mathcal{V}_{\mathrm{i}}$ und $v^{\mathrm{j}}_{\mathrm{x}} \in \mathcal{V}_{\mathrm{j}}$ des selben Knotens $v_{\mathrm{x}} \in \mathcal{V}$. Hierbei haben alle Diagonalelemente einen festen Wert α, falls gilt $0 < |\mathrm{i}-\mathrm{j}| \leq \mathrm{k}$. Dies ist gleichbedeutend damit, dass Zeitschritt $\mathrm{t_i}$ in Reichweite von Zeitschritt $\mathrm{t_j}$ liegt. Ansonsten enthalten alle Einträge der Matrix $\mathcal{D}(\mathrm{i},\mathrm{j})$ den Wert 0.

Bei der Erzeugung der Graphen werden wir die veränderlichen Parameter Reichweite k, Interzeitkantengewicht α und Schwelle p gemäß dem in Algorithmus 4 beschriebenen Ablauf durchlaufen. In dieser ersten Testreihe entscheiden wir uns zur Clusterung der Graphen mit allen drei vorgestellten Cluster-Verfahren (siehe Abschnitt 2.5). Die drei Verfahren basieren auf sehr unterschiedlichen Methoden der Clusterung und ermöglichen uns eine breitere Entscheidungsgrundlage für die weiteren Testreihen.

4.2. Analyse

Unabhängig von der Anwendung liegen die Nachteile der gewählten Methoden zum einen in der Starrheit der Interzeitkanten. Die Interzeitkantengewichte sind für alle Knoten gleich, egal wie

stark sich die Ausrichtung bzw. das Verhalten des Knoten zwischen zwei Zeitschritten verändert. Selbst wenn es für einen Knoten v_x in zwei aufeinanderfolgenden Zeitpunkten t und $\mathrm{t}+1$ keinen Knoten v_y gibt, dessen Repräsentant v_y^t ein Nachbar von v_x^t und dessen Repräsentant $v_\mathrm{y}^{\mathrm{t}+1}$ ein Nachbar von $v_\mathrm{x}^{\mathrm{t}+1}$ ist, hat die Kante den gleichen Wert wie die Interzeitkante zwischen zwei Repräsentanten v_z^t und $v_\mathrm{z}^{\mathrm{t}+1}$ des selben Knotens v_z, bei denen alle Nachbarn und alle Gewichte übereinstimmen. Dies ist nicht intuitiv und spiegelt die Veränderungen innerhalb des Graphen nicht wider. Die Interzeitkanten sollten eine starke Verbindung nur zwischen den Repräsentanten eines Knotens herstellen, die viel gemeinsam haben. Bei einer starken Veränderung der Ausrichtung bzw. des Verhaltens des Knotens sollte sich dieses im betreffenden Interzeitkantengewicht niederschlagen. Ein weiteres Problem besteht in der Wahl von α. Wählen wir das Gewicht α zu klein werden die meisten Cluster nur Knoten eines einzelnen Zeitschrittes enthalten, was der flachen Clusterung der einzelnen Zeitschritte entsprechen würde. Wählen wir die beiden Parameter Interzeitkantengewicht α und Reichweite k zu hoch, besteht die Gefahr von Zeitschläuchen, die jeweils alle Repräsentanten eines Knotens enthalten.

Der Vorteil der gewählten Methode ist der geringe Aufwand zur Erzeugung des zeitexpandierten Graphen. Da wir den Aufwand der Erzeugung der Graphen ausgehend von den Matrizen $\mathcal{A}(\mathrm{t})$ betrachten, liegt der Aufwand zur Erzeugung eines zeitexpandierten Graphen mit den gewählten Methoden in $\mathcal{O}(\mathrm{n_d k})$ mit $\mathrm{n_d} = |\mathcal{V}_\mathrm{d}|$. Auch die Laufzeit der Clusterung des zeitexpandierten Graphen ist abhängig von der Knotenanzahl $\mathrm{n_d}$ des zeitexpandierten Graphen und dem verwendeten Cluster-Verfahren. Mit der Abschätzung $\mathrm{n_d} \leq \mathrm{nd}$ liegt die Laufzeit für das Greedy-Significance-Clustering in $\mathcal{O}(\mathrm{n^2 d^2 \log(nd)})$, die für das Markov-Clustering in $\mathcal{O}(\mathrm{nd}\kappa^2)$. Die Laufzeit des Iterative-Conductance-Cutting ist polylogarithmisch in Abhängigkeit von nd.

Für unsere Anwendung wählen wir als Referenz-Clusterung die Zugehörigkeit der E-Mail-Accounts zu den verschiedenen Lehrstühlen. Die Begründung ist naheliegend, denn die Mitglieder eines Lehrstuhles arbeiten an gemeinsamen Projekten, müssen viele Termine abstimmen und arbeiten alle an miteinander verwandten Themen. Ebenso dürften sich bei den meisten Lehrstühlen während eines Jahres weniger Veränderungen zwischen den Mitgliedern ergeben, als zwischen Mitgliedern verschiedener Lehrstühle. Auch ist innerhalb eines Lehrstuhles die Wahrscheinlichkeit sozialer Kontakte höher als außerhalb. Wir werden daher versuchen, die Granularität der Clusterungen der verschiedenen Verfahren der Referenz-Clusterung anzupassen, damit ein besserer Vergleich möglich ist.

Dabei ist die gewählte Referenz-Clusterung nicht als die optimale Clusterung zu verstehen. Sie erfasst weder mögliche Kooperationen der Lehrstühle untereinander noch berücksichtigt sie, dass sich diese Kooperationen mit der Zeit verändern. Ein weiterer Nachteil der Referenz-Clusterung ist die Ausblendung sozialer Kontakte und räumlicher Nähe. Die vorgegebene Einteilung der Accounts berücksichtigt nicht, ob die Personen sich privat kennen oder ob sie in benachbarten Räumen des Fakultätsgebäudes sitzen. Alles dies ist bei den Clusterungs-Ergebnissen zu beachten. Wir gehen jedoch aufgrund des relativ kurzen, betrachteten Zeitbereichs davon aus, dass die Referenz-Clusterung eine gute Vergleichsbasis ist.

Wir nehmen an, dass sich ein hohes α positiv auf die Ähnlichkeit der gefundenen Clusterung und der Referenz-Clusterung auswirkt, da ein höheres α eine höhere Intraclusterdichte aller Referenz-Cluster bedeutet. Die gleiche positive Auswirkung auf die Referenz-Clusterung müsste eine Erhöhung der Reichweite des zeitexpandierten Graphen haben. Hierbei sei angemerkt, dass sich die Erhöhung der Reichweite bei zeitexpandierten Graphen mit einer hohen Dynamik durchaus auch negativ auswirken kann. Hier finden starke Veränderungen der Kantengewichte und Nach-

barschaften innerhalb der Zeitschritte statt. Diese starken Veränderungen werden durch konstante Interzeitkantengewichte nicht erfasst. Die hohe Reichweite führt zusätzlich zu einer starken Konnektivität der Repräsentanten eines Knotens. Die verschiedenen Knoten werden trotz eines stark veränderten Verhaltens mit hoher Wahrscheinlichkeit dem selben Cluster zugeordnet.

Algorithmus 4 Ablauf bei der Erzeugung und Clusterung der Normal-Graphen

Eingabe: E-Mail-Daten der Fakultät für Informatik eines Zeitraumes von 308 Tagen // siehe 3.1
Ausgabe: geclusterte zeitexpandierte Graphen in Verzeichnis dir

```
1: d ← 11    // Festlegung der Anzahl der Zeitschritte
2: for i = 1 to d do
3:     erzeuge die Matrix A(t)    // Erzeugung wie in 3.1 beschrieben
4: end for

5: k_max ← 9    // Festlegung der maximalen Reichweite der zu erzeugenden Graphen
6: k ← 1    // Reichweite k wird auf 1 gesetzt
7: α ← 1    // α wird auf 1 gesetzt

8: while k ≤ k_max do    // Schleife: bei jedem Schleifendurchlauf wird die Reichweite k um den Wert 2 erhöht
9:     while α ≤ 10 do    // Schleife: bei jedem Schleifendurchlauf wird α um den Wert 1 erhöht
10:        p ← 0
11:        while p < α do    // Schleife: bei jedem Schleifendurchlauf wird die Schwelle p um den Wert 3 erhöht
12:            procedure CREATE GRAPH(d, k, α, p)    // siehe 2.2 und 3.2
13:                erzeuge Graph G^d_{k,p} mit den Matrizen A(t) und den festgelegten Parametern
14:                speichere den Graphen in Verzeichnis dir
15:            end procedure
16:            p ← p + 3
17:        end while    // Ende der Schwellen-Schleife
18:        α ← α + 1
19:    end while    // Ende der Interzeitkanten-Schleife
20:    k ← k + 2
21: end while    // Ende der Reichweite-Schleife

22: for all Graphen G in dir do
23:    clustere G mit Greedy-Significance-Clustering
24:    clustere G mit Iterative-Conductance-Cutting
25:    clustere G mit Markov-Clustering
26: end for
```

4.3. Experimente

Mit Hilfe unseres Frameworks erzeugen wir 108 zeitexpandierte Graphen. Anschließend werden die Graphen mit den Clusterverfahren Greedy-Significance-Clustering, Markov-Clustering und Iterative-Conductance-Cutting geclustert. Es sei angemerkt, dass die verwendeten Cluster-Verfahren keine normierten Kantengewichte der Graphen voraussetzen. Zunächst untersuchen wir die Wahl unserer Referenz-Clusterung.

4.3.1. Überprüfung der Referenz-Clusterung

Die durchschnittliche gewichtete Coverage der Referenz-Clusterung beträgt ungefähr $0,9$. Dieser Wert allein würde nicht für eine signifikante Clusterung sprechen, ist aber ein gutes Indiz. Des-

halb berechnen wir zusätzlich die Performance, die gewichtete Modularity und die durchschnittliche Interclusterconductance (siehe Tabelle 1). Die hohen Durchschnittswerte der verschiedenen Indizes sprechen für die gewählte Referenz-Clusterung. Sie ist sicherlich nicht die optimale Clusterung, liefert aber für alle Graphen akzeptable Indexwerte, wie die Minima in Tabelle 1 belegen.

Im Weiteren werden wir die von den Cluster-Verfahren gefundenen Clusterungen anhand der Referenz-Clusterung und ihrer Indexwerte bewerten. Wir werden die Auswirkungen der Veränderung der verschiedenen Parameter auf das Ergebnis der Clusterungen untersuchen und hoffen so Rückschlüsse auf unser Modell ziehen zu können.

Index	cov_w	cov	per	mod_w	mod	δ_d
1. Quartil	0,8785	0,7995	0,9350	0,7760	0,7316	0,7131
3. Quartil	0,9310	0,9251	0,9388	0,8357	0,8534	0,8538
Minimum	0,8146	0,7104	0,9299	0,7022	0,6398	0,5889
Maximum	0,9535	0,9619	0,9392	0,8596	0,8817	0,8723
Mittelwert	0,9022	0,8558	0,9365	0,8031	0,7843	0,7686

Tabelle 1: *Indizes für die Referenz-Clusterung der 108 zeitexpandierten Graphen der Normal-Testreihe.*

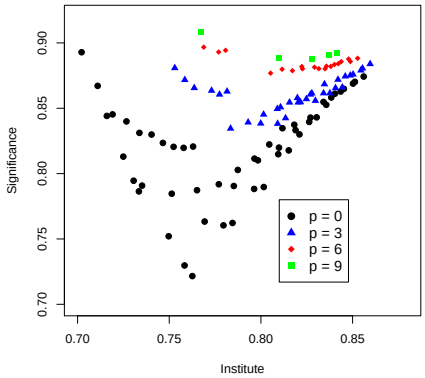
Index	cov_w	per	mod_w	δ_d	$\mathrm{bestmatch}_\mathrm{no}$ Institute
1. Quartil	0,8958	0,9250	0,8283	0,7366	0,4109
3. Quartil	0,9414	0,9393	0,8788	0,8906	0,6649
Minimum	0,7802	0,9086	0,7216	0,5194	0,0945
Maximum	0,9596	0,9555	0,9087	0,9464	0,7041
Mittelwert	0,9137	0,9330	0,8473	0,8042	0,5275

Tabelle 2: *Indizes für die Greedy-Significance-Clusterungen der 108 zeitexpandierten Graphen der Normal-Testreihe.*

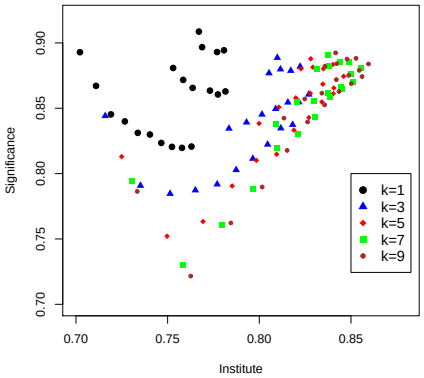
4.3.2. Greedy-Significance-Clustering

Das Greedy-Significance-Clustering (siehe Abschnitt 2.5.1), das wir zunächst betrachten, benötigt keine Eingabeparameter. Wir berechnen die Durchschnittswerte der Indizes der gefundenen Clusterungen (siehe Tabelle 2). Um diese Ergebnisse besser interpretieren zu können, setzen wir sie in Relation zu den Ergebnissen der Referenz-Clusterung. Wir vergleichen die Indizes gewichtete Modularity (17), gewichtete Coverage (19) und durchschnittliche Interclusterconductance (18). Die Werte der Performance eignen sich aus den in Kapitel 2.4.2 beschriebenen Gründen nicht für einen Vergleich. Für jeden Index erzeugen wir drei Abbildungen, die alle den selben Sachverhalt wiedergeben. Jeder der 108 Graphen liefert einen Punkt, dessen Koordinaten sich aus dem Wert des Index für die Referenz-Clusterung und dem Wert des Index für die Significance-Clusterung ergeben. Jede dieser Abbildungen visualisiert die Graphen in Abhängigkeit eines anderen variablen Parameters. Mit Hilfe dieser Abbildungen hoffen wir die Ergebnisse besser interpretieren zu können.

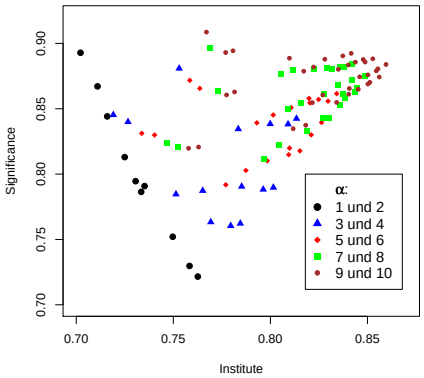
Interessant ist, dass die Werte für beide Clusterungen nicht gleichmäßig ansteigen. Sortiert man die Graphen nach den Werten der Referenz-Clusterung, fallen zu Beginn die Werte für die Significance-Clusterungen, trotz steigender Werte der Referenz-Clusterung. Dies widerspricht



(a) Farben repräsentieren die verschiedenen Werte der Schwelle p.



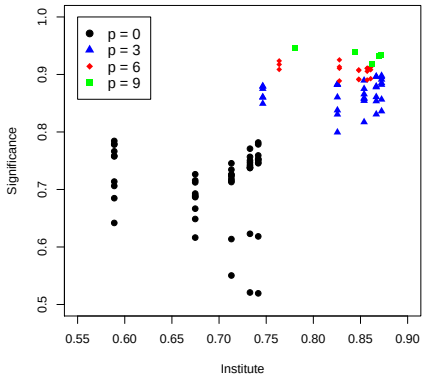
(b) Farben repräsentieren die verschiedenen Werte der Reichweite k.



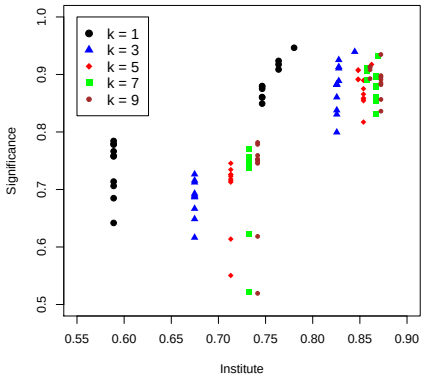
(c) Farben repräsentieren die verschiedenen Werte der Interzeitkantengewichte α.

Abbildung 17: *Vergleich der Referenz-Clusterung und der Significance-Clusterung anhand der gewichteten Modularity. Jeder Punkt steht für einen Graphen der Testreihe. Die Koordinaten ergeben sich aus der Modularity der Referenz-Clusterung und der Significance-Clusterung. Dabei stellen alle drei Abbildungen den selben Sachverhalt dar. Der Unterschied liegt in der Einfärbung der Plots. In der Abbildung 17a stehen die verschiedenen Farben für die unterschiedlichen Schwellen der Graphen. Die Abbildung 17b unterscheidet die unterschiedlichen Reichweiten und die Abbildung 17c die verschiedenen Werte der Interzeitkantengewichte α.*

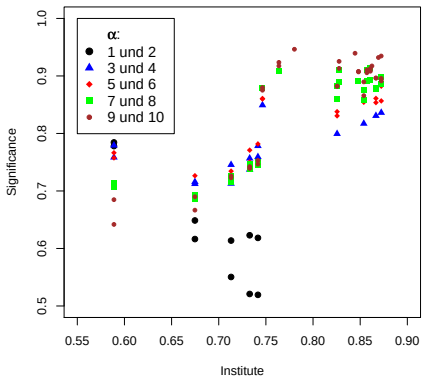
unserer Erwartung, dass die durch die Lehrstühle vorgegebene Clusterung eine stark signifikante Maßgabe ist. Eigentlich hätten wir bei einer Verbesserung der Bewertung der Referenz-Clusterung ebenfalls eine Verbesserung der gefundenen Clusterung erwartet.



(a) Farben repräsentieren die verschiedenen Werte der Schwelle p.



(b) Farben repräsentieren die verschiedenen Werte der Reichweite k.

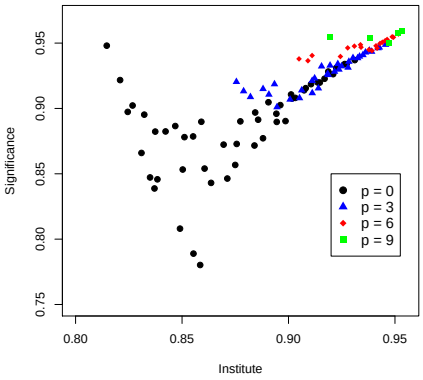


(c) Farben repräsentieren die verschiedenen Werte der Interzeitkantengewichte α.

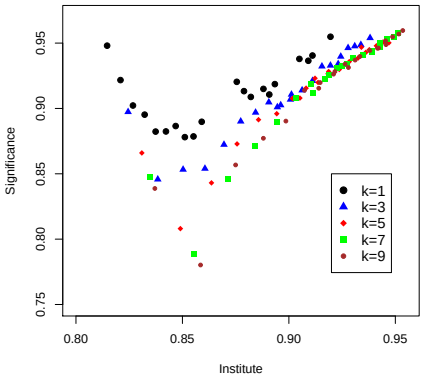
Abbildung 18: *Ähnlich wie in Abbildung 17 vergleichen wir die Referenz-Clusterung mit der Significance-Clusterung. Im Unterschied zu Abbildung 17 vergleichen wir sie anhand der Werte der durchschnittlichen Interclusterconductance.*

Eine Erklärung dieser widersprüchlichen Entwicklung liefern uns die Abbildungen (17,18,19). Die Graphen mit den niedrigsten Werten bezüglich der starren Referenz-Clusterung haben ein kleines Interzeitkantengewicht α und eine kleine Reichweite k. Das bedeutet, dass diese Graphen die geringste Interzeitdichte aufweisen. Sie haben die wenigsten Interzeitkanten und die kleinsten Interzeitkantengewichte. Um den Einfluss der Interzeitdichte auf die Clusterung genauer zu untersuchen, vergleichen wir den $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$ der Significance-Clusterungen (siehe Abbildung 20d) anhand des Produkts von α und k, wobei die Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ alle Knoten eines diskreten Zeitpunktes in einem Cluster vereinigt und $\mathcal{C}'$ die Significance-Clusterung

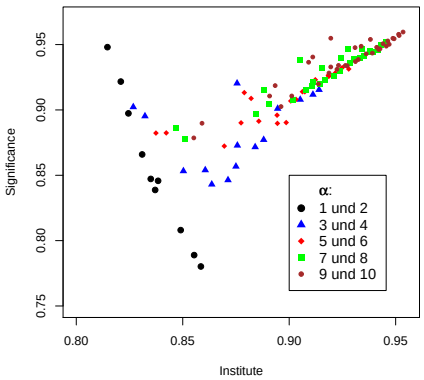
des jeweiligen Graphen ist. Das Ergebnis zeigt uns, dass das Greedy-Significance-Clustering bei niedrigen Werten von α und kleiner Reichweite dazu neigt, Cluster zu finden, die eine starke Ähnlichkeit zu den Clustern der Zeit-Clusterung $\mathcal{C}_{\text{time}}$ aufweisen (Abbildung 40 im Anhang).



(a) Farben repräsentieren die verschiedenen Werte der Schwelle p.



(b) Farben repräsentieren die verschiedenen Werte der Reichweite k.



(c) Farben repräsentieren die verschiedenen Werte der Interzeitkantengewichte α.

Abbildung 19: *Ähnlich wie in Abbildung 17 vergleichen wir die Referenz-Clusterung mit der Significance-Clusterung. Im Unterschied zu Abbildung 17 vergleichen wir sie anhand der Werte der gewichteten Coverage.*

Dies erklärt den Abfall der Significance-Werte. Für kleine α und ein kleines k verlaufen die meisten der gefundenen Cluster innerhalb eines Zeitschrittes. Durch die Erhöhung der Parameter α und k werden diese Cluster automatisch schlechter bewertet, gleichzeitig steigt die Bewertung der Referenz-Cluster (Abbildung 20b). Nach und nach geht die Significance-Clusterung von Intrazeit-Clustern über in Interzeit-Cluster. Dabei nimmt die Ähnlichkeit zur Referenz-Clusterung zu (Abbildungen 20a und 20c). Hierbei sei erwähnt, dass die Clusterung $\mathcal{C}'$ bereits bei einem $\text{bestmatch}_{\text{no}}(\mathcal{C}, \mathcal{C}') \geq 0,4$ eine hohe Ähnlichkeit zur der Referenz-Clusterung aufweist (siehe Abbildung 50 im Anhang).

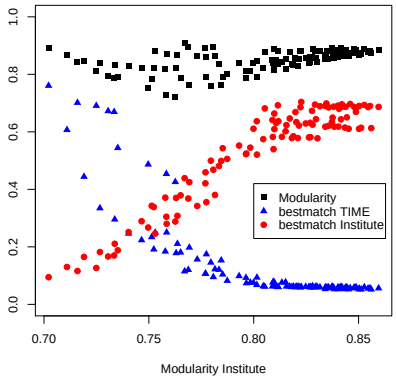
Für höhere Werte der Indizes der Referenz-Clusterung stellt sich dann die erwartete Korrelation zwischen den Indizes der Referenz-Clusterung und der Significance-Clusterungen ein. Besonders deutlich ist dieser Sachverhalt für die Coverage (siehe Abbildung 19) und die Modularity (siehe Abbildung 17), bei denen für höhere Werte der Referenz-Clusterung ein proportionaler Anstieg der Indexwerte beider Clusterungen zu beobachten ist. Ursache dafür ist das verwendete Greedy-Significance-Clustering. Das hierarchische Cluster-Verfahren basiert auf einer schrittweisen Maximierung der Modularity, welche eng mit der Coverage verknüpft ist (siehe 2.4.4).

Auffallend ist die Auswirkung der Schwelle des Graphen auf die Ähnlichkeit zur Referenz-Clusterung. Ein Graph mit Schwelle $\mathrm{p} = 3$ oder höher reduziert die Ähnlichkeit der gefundenen Significance-Clusterung mit der Zeit-Clusterung und erhöht die Ähnlichkeit zur Referenz-Clusterung (Abbildung 20). Das führen wir darauf zurück, dass die Kanten innerhalb eines Lehrstuhles im Schnitt einen höheren Wert haben als Kanten, die zwischen verschiedenen Lehrstühlen verlaufen. Also filtert die Schwelle vor allem Kanten zwischen verschiedenen Lehrstühlen heraus. Hinzu kommt, dass die Schwelle immer kleiner ist, als das gewählte Interzeitkantengewicht α. Das heißt, die Anzahl der Intrazeitkanten fällt, während die Anzahl der Interzeitkanten konstant bleibt. Dies führt zu einer generell besseren Bewertung von Interzeit-Clustern. Vor allem für dichte Graphen ist eine Schwelle von großem Vorteil. Sie verringert die Komplexität, da sie die Anzahl der Kanten und eventuell auch der Knoten reduziert. Zudem profitieren effiziente Implementierungen der Cluster-Algorithmen von einer Ausdünnung der Kantenmenge. Weiterhin vermindert sie die Wahrscheinlichkeit, dass Clusterungs-Ergebnisse aus reinen Intrazeit-Clustern bestehen.

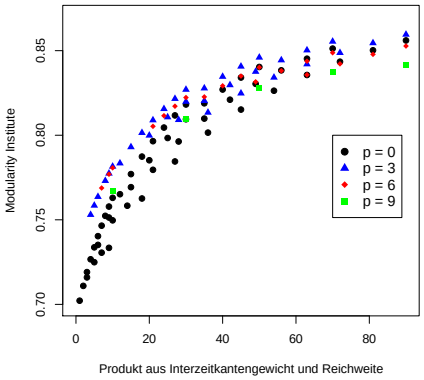
Allerdings ist sie mit Bedacht zu wählen. Eine Schwelle bedeutet immer auch einen Verlust von Information. Wählt man sie zu hoch, verliert das Ergebnis an Aussagekraft. Bei unseren E-Mail-Graphen dieser Testreihe bedeutet Schwelle $\mathrm{p} = 9$ eine Reduzierung der Knoten von 4677 auf 1918, da nur Knoten mit mindestens einer Kante im Graph existieren. So haben die Graphen mit Schwelle $\mathrm{p} = 9$ eine geringere Ähnlichkeit zur Referenz-Clusterung, als Graphen mit Schwelle $\mathrm{p} = 6$ (siehe dazu Abbildung 20c). Bei der Clusterung mit dem Greedy-Significance-Clustering hätten wir auf eine Schwelle verzichten können. Wir werden jedoch sehen, dass es Fälle gibt, in denen eine Schwelle von hohem Vorteil ist. Generell bleibt festzuhalten, dass eine Schwelle immer im Kontext der Anwendung gewählt werden muss.

4.3.3. Iterative-Conductance-Cutting

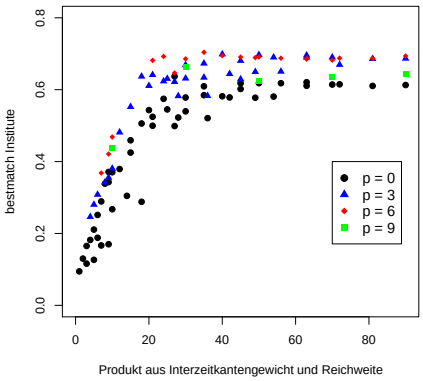
Bei der Clusterung mit den verschiedenen Verfahren haben wir bei dem Markov-Clustering und dem Iterative-Conductance-Cutting das Problem, adäquate Werte für die Parameter der Cluster-Verfahren zu bestimmen. Wir führen die Clusterung mit dem Iterative-Conductance-Cutting (siehe Abschnitt 2.5.2) für alle Graphen mit vier verschiedenen Schwellen a^* durch. Für $\mathrm{a}^* = 0,20$ erhalten wir Clusterungen mit durchschnittlich 200 Clustern bei einer durchschnittlichen Knotenzahl der zeitexpandierten Graphen von 3700. Je kleiner wir den Wert a^* wählen, desto früher bricht der Algorithmus ab und desto weniger Cluster hat das Ergebnis der Clusterung. Da wir uns größere Cluster wünschen, führen wir das Iterative Conductance Cutting mit drei weiteren Schwellenwerten $\mathrm{a}^* = 0,10$, $\mathrm{a}^* = 0,075$ und $\mathrm{a}^* = 0,05$ durch. Dabei beobachten wir, dass die Durchschnittswerte für die Coverage und die durchschnittliche Interclusterconductance für kleinere a^* ansteigen. Die Modularity und der $\mathrm{bestmatch}_{\mathrm{no}}$ steigen beim Übergang von $\mathrm{a}^* = 0,20$ nach $\mathrm{a}^* = 0,10$ an. Für $\mathrm{a}^* = 0,10$ und $\mathrm{a}^* = 0,075$ sind die beiden annähernd konstant und fallen



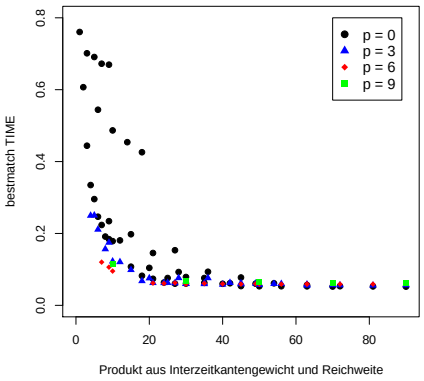
(a) In dieser Abbildung erfolgt eine Gegenüberstellung verschiedener Kennwerte anhand der gewichteten Modularity der Referenz-Clusterung.



(b) Einfluss von Reichweite k und Interzeitkantengewicht α auf die gewichtete Modularity der Referenz-Clusterung. Vergleiche hierzu Abbildung 20a.



(c) Einfluss von Reichweite k und Interzeitkantengewicht α auf die Ähnlichkeit der Significance-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$, angegeben durch den $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$.



(d) Einfluss von Reichweite k und Interzeitkantengewicht α auf die Ähnlichkeit der Significance-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$, angegeben durch den $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$.

Abbildung 20: *Die Abbildungen verdeutlichen die Auswirkungen der Parameter α und k auf die Clusterungen des Significance-Verfahrens. Bei einer Erhöhung einer der beiden Parameter erhöht sich die Ähnlichkeit der gefundenen Significance-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ und gleichzeitig nimmt die Ähnlichkeit zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ ab. (a) Hier findet eine Gegenüberstellung der gewichteten Modularity $\mathrm{mod}_{\mathrm{w}}(\mathcal{C}')$ (schwarz) der Significance-Clusterung, der Ähnlichkeit $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$ (blau) der Significance-Clusterung zur Zeit-Clusterung und der Ähnlichkeit $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ (rot) der Significance-Clusterung zur Referenz-Clusterung anhand der gewichteten Modularity $\mathrm{mod}_{\mathrm{w}}(\mathcal{C})$ der Referenz-Clusterung statt. (b) Auswirkungen der Parameter Interzeitkantengewicht α und Reichweite k auf die Modularity der Referenz-Clusterung. (c)-(d) Auswirkungen der Parameter Interzeitkantengewicht α und Reichweite k auf die Ergebnisse der Significance-Clusterungen.*

für $a^* = 0,05$ wieder etwas ab. Die Tabellen 18, 19, 20 und 21 mit diesen Durchschnittswerten für die vier verschiedenen Werte des Parameters a^* sind im Anhang zu finden. Die Clusterungen für $a^* = 0,05$, $a^* = 0,075$ und $a^* = 0,1$ werden wir jetzt eingehend untersuchen.

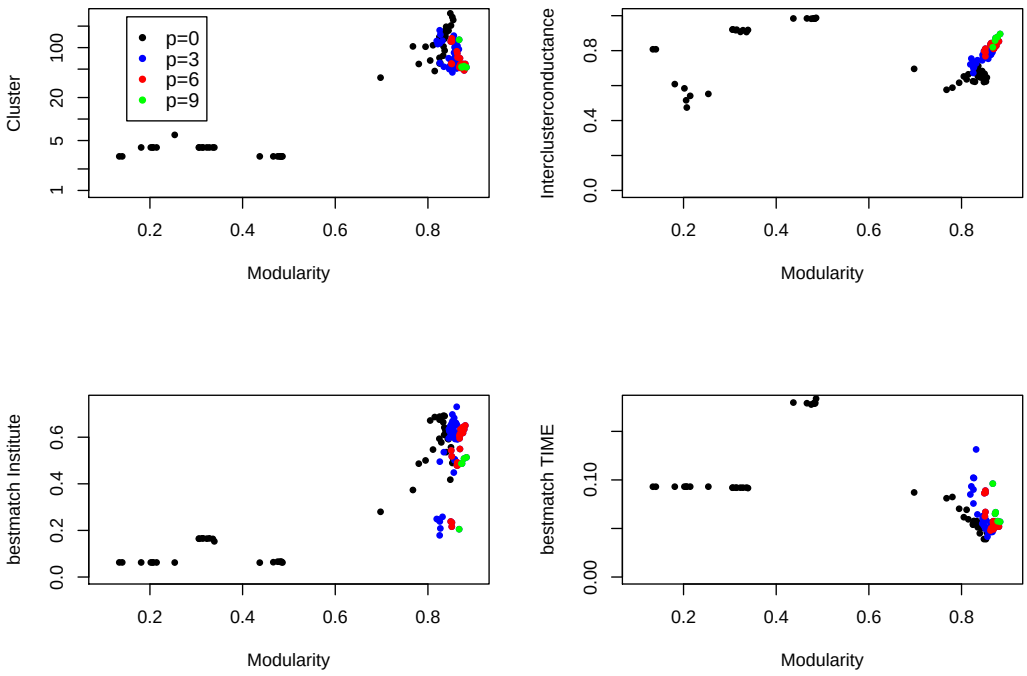


Abbildung 21: *Die Plots zeigen für die ICC-Clusterungen mit* $a^* = 0,1$ *die Anzahl der Cluster, die durchschnittliche Interclusterconductance, den* $\mathrm{bestmatch}_{\mathrm{no}}$ *bezüglich der Referenz-Clusterung und den* $\mathrm{bestmatch}_{\mathrm{no}}$ *bezüglich der Zeit-Clusterung. Die Ordnung erfolgt aufgrund der Modularity der ICC-Clusterungen. Dabei stehen die Farben für die unterschiedlichen Schwellen der Graphen.*

Bei der Untersuchung der ICC-Clusterung mit $a^* = 0,10$ ist auffällig, dass ein Viertel der Clusterungen eine deutlich niedrigere gewichtete Modularity aufweisen. Eine Gemeinsamkeit dieser Clusterungen $\mathcal{C}$ ist, dass sie aus sehr wenigen Clustern bestehen. Dies ist dann der Fall, wenn der Algorithmus schon früh abbricht und nur wenige Schnitte durchgeführt werden. Diese Clusterungen gehören alle zu Graphen mit Schwelle $p = 0$ (siehe Abbildung 21). Bei $a^* = 0,075$ haben 28 % der Clusterungen weniger als zehn Cluster. Für $a^* = 0,05$ sind es bereits 36 %.

Im Gegensatz zu den Significance-Clusterungen gibt es nur wenige Graphen mit einer ICC-Clusterung mit erhöhter Ähnlichkeit zur Zeit-Clusterung. Es sind gerade die Graphen mit Schwelle $p = 0$ und Reichweite $k = 1$. Bei den Graphen mit Modularity $\mathrm{mod}_w(\mathcal{C}) > 0,8$ gibt es eine kleine Gruppe mit geringer Ähnlichkeit zur Referenz-Clusterung. In Abbildung 22 wird deutlich, dass diese Graphen alle die minimale Reichweite $k = 1$ haben. Das führt uns zu dem Schluss, dass der stärkste Einfluss auf die ICC-Clusterung, abgesehen von a^*, von der Reichweite k und der Schwelle p ausgeht.

Die Betrachtung der Clusterungen für $a^* = 0,075$ und $a^* = 0,05$ (siehe Abbildungen 44 und

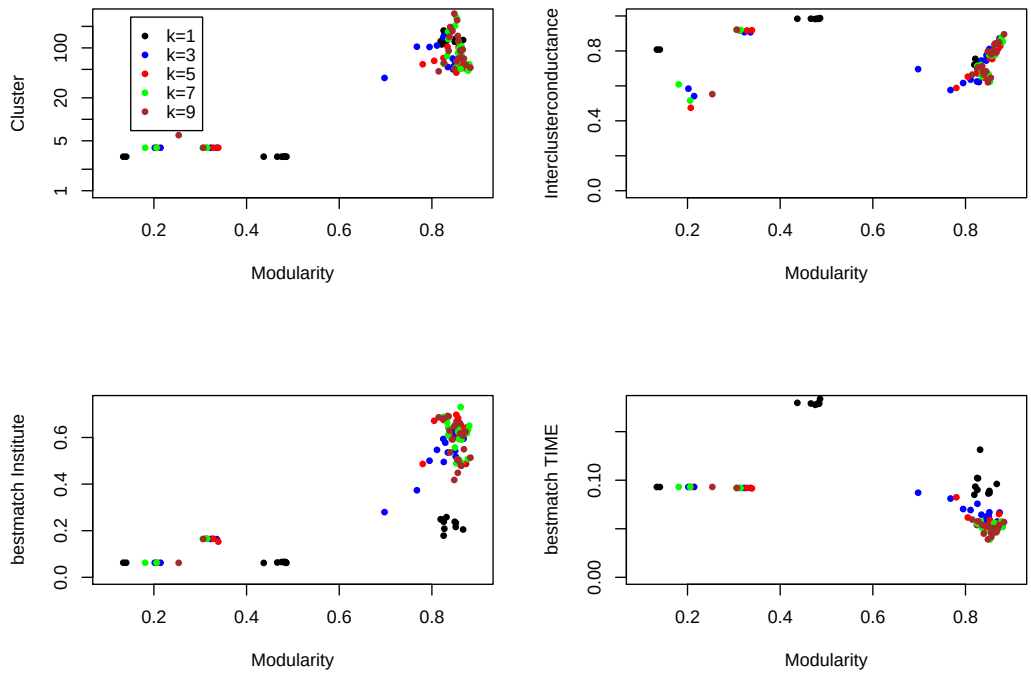


Abbildung 22: *Die Plots zeigen für die ICC-Clusterungen mit* $\mathrm{a}^* = 0,1$ *die Anzahl der Cluster, die durchschnittliche Interclusterconductance, den* $\mathrm{bestmatch_{no}}$ *bezüglich der Referenz-Clusterung und den* $\mathrm{bestmatch_{no}}$ *bezüglich der Zeit-Clusterung. Die Ordnung erfolgt aufgrund der Modularity der ICC-Clusterungen. Dabei stehen die Farben für die unterschiedlichen Reichweiten der Graphen.*

43 im Anhang) zeigen ähnliche Resultate. Wir vergleichen die Ähnlichkeit der Clusterungen zur Referenz-Clusterung in Abhängigkeit von der Schwelle. Das Ergebnis in Tabelle 3 zeigt, dass die Erhöhung der Schwelle von 0 auf 3 eine deutliche Verbesserung des $\mathrm{bestmatch_{no}}$ bedeutet. Eine weitere Erhöhung bringt hingegen nur noch geringfügige Verbesserungen oder gar eine Verschlechterung der Durchschnittswerte bei einem hohen Informationsverlust, da sich die Anzahl der Knoten stark reduziert.

Schwelle p	0	3	6	9
$\mathrm{a}^* = 0,05$	0,2319	0,5321	0,6071	0,5738
$\mathrm{a}^* = 0,075$	0,2964	0,6161	0,5793	0,5177
$\mathrm{a}^* = 0,1$	0,3465	0,5474	0,5253	0,5738

Tabelle 3: *Durchschnittswerte des* $\mathrm{bestmatch_{no}}$ *der ICC-Clusterungen bezüglich der Referenz-Clusterung für die verschiedenen Schwellen* p *und Werte des Parameters* a^*.

Den maximalen $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung $\mathcal{C}$ liefert der Graph mit den Parametern $\mathrm{k} = 3$, $\mathrm{p} = 3$ und $\alpha = 9$. Für $\mathrm{a}^* = 0,075$ ergibt der $\mathrm{bestmatch_{no}}$ der ICC-Clusterung $\mathcal{C}'$ dieses Graphen den Wert $bestmatch_{\mathrm{no}}(\mathcal{C}, \mathcal{C}') \approx 0,78$. Für den interessierten Leser ist das Bild dieser Clusterung im Anhang 46 zu finden.

4.3.4. Markov-Clustering

Beim Markov-Clustering (siehe Abschnitt 2.5.3) gibt es den Expansionsparameter $\mathrm{e} \in \mathbb{N}_{>1}$, den Inflationsparameter $\mathrm{r} \in \mathbb{R}^+$ und den Pruningparameter $\kappa \in \mathbb{N}$. Dabei steht e für die Anzahl der Schritte des Random Walks. Der Inflationsparameter r reduziert die Auswirkung kleiner Einträge der Matrix, während der Pruningparameter die Anzahl der Elemente je Zeile der Matrix angibt, die maximal benutzt werden. Das heißt, wenn eine Zeile der Matrix mehr als κ von 0 verschiedene Einträge enthält, werden nur die κ höchsten Werte zur Berechnung der neuen Matrix $\mathcal{M}$ verwendet.

Typ	e	r	κ	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	$\mathrm{bestmatch_{no}}$ Institute	Cluster
A	3	2	120	0,7616	0,7420	0,6160	0,2717	211,4
A	4	2	120	0,7799	0,7434	0,4902	0,3584	180,7
B	5	1,5	150	0,9585	0,7077	0,8991	0,3803	11,2
B	3	2	50	0,7613	0,7418	0,6579	0,2888	158,2
B	3	1,5	50	0,8801	0,8186	0,7893	0,4768	50,6
B	3	1,2	50	0,9619	0,7176	0,9426	0,4351	10,3
C	5	1,5	50	0,9565	0,7225	0,9104	0,4272	10,2
C	5	1,8	50	0,9019	0,8104	0,8277	0,5248	25,8
C	5	1,9	50	0,8906	0,8099	0,8083	0,5457	33,0
C	5	2,0	50	0,8779	0,8075	0,7897	0,5505	40,8

Tabelle 4: *Die Abbildung enthält die Durchschnittswerte einiger Kennwerte der MCL-Clusterungen der verschiedenen Durchläufe. Nur bei den Durchläufen von Typ A werden alle* 108 *Graphen geclustert. Bei Typ B werden nur die Graphen mit Schwelle* $\mathrm{p} = 3$ *geclustert. In Typ C werden nur die fünf Graphen geclustert, die für den letzten Durchlauf von Typ B den höchsten* $\mathrm{bestmatch_{no}}$ *bezüglich der Referenz-Clusterung aufweisen.*

Zunächst führen wir die Clusterung der 108 Graphen mit den Parametern $\mathrm{e} = 3$, $\mathrm{r} = 2$ und $\kappa = 120$ durch. Wir berechnen einige Durchschnittswerte verschiedener Kennwerte (siehe Tabelle 4), um die Clusterungen qualitativ mit den Ergebnissen der beiden anderen Cluster-Verfahren zu vergleichen. Die Durchschnittswerte sind deutlich geringer als bei den anderen Cluster-Verfahren. Der Durchschnittswert der MCL-Clusterungen für den $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung ist ähnlich zu den schlechten Ergebnissen der ICC-Clusterungen für $\mathrm{a}^* = 0,20$ (siehe Tabelle 18). Daher betrachten wir die durchschnittliche Clusterzahl der Clusterungen. Der hohe Durchschnittswert von über 200 Clustern erklärt die geringe Ähnlichkeit zur Referenz-Clusterung.

Daraufhin führen wir die Clusterung der Graphen mit den Parametern $\mathrm{e} = 4$, $\mathrm{r} = 2$ und $\kappa = 120$ durch. Die durchschnittliche Anzahl der Cluster nimmt dabei nur leicht ab. Dennoch erhöht sich der durchschnittliche $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung der Lehrstühle auf $0,3584$. Um einen besseren Wert zu erreichen, führen wir eine erneute Clusterung mit den Werten $\mathrm{e} = 5$, $\mathrm{r} = 1.5$ und $\kappa = 150$ durch. Durch die Reduzierung von r dürfte sich die Anzahl der Cluster verringern, da die Einträge der stochastischen Matrix weniger stark ausgedünnt werden. Bei diesem und den weiteren Durchläufen verwenden wir nur die Graphen mit Schwelle $\mathrm{p} = 3$. Diesen Entschluss treffen wir aufgrund unserer Beobachtungen bezüglich der Schwelle (siehe Tabelle 5) und der erhöhten Laufzeit durch die hohen Parameterwerte für e und κ. Die durchschnittliche Laufzeit der Clusterung eines Graphen mit den gewählten Parametern beträgt mit

unserem AMD Opteron 2218 Prozessor mit 2,6 GHz ungefähr anderthalb Stunden, während die Laufzeit bei $\mathrm{e} = 3$, $\mathrm{r} = 2$ und $\kappa = 120$ zwischen drei und 20 Minuten liegt. Der durchschnittliche $\mathrm{bestmatch_{no}}$ von $0,3803$ rechtfertigt den erhöhten Aufwand leider nicht. Der Parameter $\mathrm{r} = 1.5$ ist zu niedrig gewählt, was zu einer durchschnittlichen Clusterzahl von $11,2$ führt.

Drei weitere Durchläufe mit $\mathrm{e} = 3$ und $\kappa = 50$ zeigen, dass eine geeignete Wahl von r entscheidend für einen Vergleich mit der Referenz-Clusterung ist. Während der durchschnittliche $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung bei $\mathrm{r} = 2$ ungefähr $0,29$ beträgt, ist er bei $\mathrm{r} = 1,5$ ungefähr $0,48$. Für $\mathrm{r} = 1,2$ sinkt der durchschnittliche $\mathrm{bestmatch_{no}}$ wieder auf $0,44$.

Wir wählen die fünf Graphen mit den höchsten $\mathrm{bestmatch_{no}}$-Werten des letzten Durchlaufs für vier weitere Durchläufe. Dabei bestätigt sich die Abhängigkeit des $\mathrm{bestmatch_{no}}$ von der Granularität, die wir mit Parameter κ steuern. Wir erreichen bei einer Erhöhung des Parameters e auf den Wert 5 einen durchschnittlichen $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung von $0,55$. Die drei variablen Parameter machen eine geeignete Wahl schwierig. Für jeden Graphen und jede Kombination von e und κ muss der Parameter r so angepasst werden, dass die Granularität der Clusterung ähnlich zur Referenz-Clusterung ist. Deshalb werden wir in den folgenden Testreihen nicht mehr auf dieses Cluster-Verfahren zurückgreifen.

4.4. Fazit

Trotz des simplen Modells, das wir in dieser ersten Testreihe verwenden, erzielen wir Clusterungen, die der Referenz-Clusterung sehr ähnlich sind. Und das nicht nur bei einem speziellen Cluster-Verfahren, sondern bei geeigneter Wahl der Parameter des Modells und der Parameter der Cluster-Verfahrens bei allen drei Verfahren. Der höchste $\mathrm{bestmatch_{no}}$ wird mit Hilfe des ICC-Verfahrens erreicht. Das Significance-Verfahren liefert uns die zuverlässigsten Ergebnisse. Die verwendete Referenz-Clusterung der Lehrstühle ist für die Analyse der Clusterungen ein wichtiges Vergleichsmaß. Es ist jedoch zu beachten, dass eine vollständige Übereinstimmung der Clusterungen mit der Referenz-Clusterung nicht wünschenswert ist. Die dynamischen Veränderungen innerhalb der untersuchten Zeitspanne werden nicht von ihr erfasst. Außerdem werden soziale Kontakte der Mitarbeiter verschiedener Lehrstühle und Kooperationen der Lehrstühle nicht berücksichtigt. Dennoch hilft uns die Referenz-Clusterung in dieser Testreihe, einige interessante Zusammenhänge zu entdecken.

Der Schwellenwert hat von den drei variablen Parametern unserer Testreihe den stärksten Einfluss auf das Ergebnis der Clusterungen. Eine Erhöhung der Schwelle $\mathrm{p} = 0$ auf den Wert 3 führt bei allen drei Verfahren zu einer starken Erhöhung der Ähnlichkeit zur Referenz-Clusterung (siehe Tabelle 5). Aufgrund dieser Ergebnisse erzeugen wir eine kleine Testreihe bei der das Interzeitkantengewicht α und die Reichweite k konstant belassen werden, und nur die Schwelle verschiedene Werte durchläuft. Das Ergebnis dieser Testreihe wird durch Abbildung 41 im Anhang illustriert. Der optimale Schwellenwert scheint ungefähr bei dem Wert 3 zu liegen. Man erkennt anhand der Abbildungen, dass eine kleine Schwelle die Modularity und Interclusterconductance der Referenz-Clusterung verbessert. Die Erhöhung der Coverage zeigt, dass die Gewichte der Kanten innerhalb der Lehrstühle im Durchschnitt größer sind, als Gewichte zwischen den verschiedenen Lehrstühlen. Das ist eine weitere Bestätigung unserer Referenz-Clusterung. Wir können erkennen, dass eine weitere Erhöhung der Schwelle die Coverage weiter verbessert, aber die anderen Indizes verschlechtert. Das führen wir auf die stärkere Reduzierung der Intracluster-

kanten im Vergleich zu den Interclusterkanten in diesem Bereich zurück (siehe Abbildung 41f), sowie die Annäherung des durchschnittlichen Gewichtes der Interclusterkanten an das Interzeitkantengewicht α aufgrund der erhöhten Schwelle p (siehe Abbildung 41e). Dadurch verwischen sich die Grenzen der einzelnen Lehrstühle. Durch die starke Reduzierung der Knotenzahl kann es ebenfalls zur Zersplitterung des zeitexpandierten Graphen kommen. Beides führt bei einer hohen Schwelle zu einer Abnahme der Ähnlichkeit der mit Hilfe der Cluster-Verfahren gefundenen Clusterungen zur Referenz-Clusterung (siehe Tabelle 5).

Schwelle p	0	3	6	9
ICC ($\mathrm{a}^* = 0,05$)	0,2319	0,5321	0,6071	0,5738
Significance	0,43560	0,5865	0,6444	0,6011
MCL ($\mathrm{e} = 4$, $\mathrm{r} = 2$ und $\kappa = 120$)	0,3013	0,4077	0,4184	0,3958

Tabelle 5: *Durchschnittswerte des* $\mathrm{bestmatch}_{\mathrm{no}}$ *der Clusterungen der verschiedenen Cluster-Verfahren bezüglich der Referenz-Clusterung abhängig von der Höhe der Schwelle.*

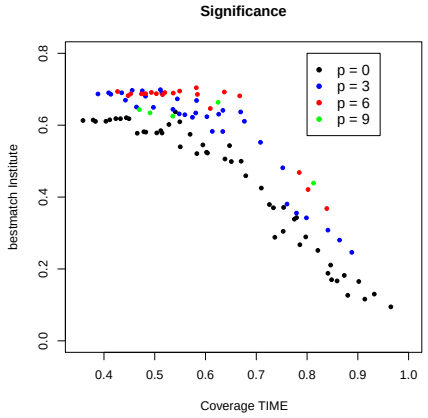
Eine Erhöhung der Reichweite von $\mathrm{k} = 1$ auf den Wert 3 bringt ebenfalls eine deutliche Verbesserung der Ähnlichkeit der Clusterungen zur Referenz-Clusterung. Weitere Erhöhungen bringen hingegen keine bzw. kaum mehr Verbesserungen. Um diesen Sachverhalt besser zu verstehen, führen wir eine weitere kleine Testreihe mit variabler Reichweite durch, für die wir die Indizes der Referenz-Clusterung berechnen. In Abbildung 42 sind die dazugehörigen Abbildungen zu finden. Anhand dieser Daten wird deutlich, dass der Anstieg der Indizes für größere Werte der Reichweite immer stärker abnimmt. Nur für kleine Werte der Reichweite gibt es eine deutliche Verbesserung der Indizes, da hier der Anstieg der Interzeitkanten am höchsten ist. Ein Vergleich mit den Durchschnittswerten der Coverage der Zeit-Clusterung unserer 108 Graphen für die unterschiedlichen Reichweiten bestätigt diesen Zusammenhang. So ist bei den Graphen die durchschnittliche Coverage der Zeit-Clusterung ungefähr $0,83$ für die Graphen mit Reichweite $\mathrm{k} = 1$, $0,57$ für die Graphen mit Reichweite $\mathrm{k} = 5$ und $0,51$ für die Graphen mit Reichweite $\mathrm{k} = 9$. Für kleine Reichweiten nimmt die Interzeitdichte bei einer Erhöhung der Reichweite also am stärksten zu. Es ist offensichtlich, das für beliebige Werte der Reichweite eine maximale Anzahl von $\mathrm{d} - 1$ Interzeitkanten pro Knoten möglich ist (siehe Tabelle 17). Die Gesamtzahl der Interzeitkanten steigt dadurch für eine sehr hohe Reichweite nicht mehr so stark wie zuvor.

Reichweite k	1	3	5	7	9
ICC ($\mathrm{a}^* = 0,05$)	0,1792	0,4575	0,4313	0,4589	0,4935
Significance	0,3006	0,5393	0,5871	0,5920	0,5980
MCL ($\mathrm{e} = 4$, $\mathrm{r} = 2$ und $\kappa = 120$)	0,20230	0,3711	0,3992	0,4024	0,4029

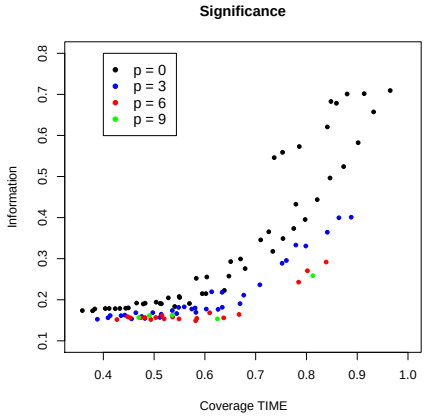
Tabelle 6: *Durchschnittswerte des* $\mathrm{bestmatch}_{\mathrm{no}}$ *der Clusterungen der verschiedenen Cluster-Verfahren bezüglich der Referenz-Clusterung abhängig von der Höhe der Reichweite.*

Eine hohe Reichweite reduziert die mögliche Dynamik einer Clusterung. Will man die dynamischen Veränderungen in den einzelnen Zeitschritten durch die Clusterung wiedergeben, sollte man die Reichweite deshalb nicht zu hoch wählen. Bei unseren Ergebnissen zeigen nur Graphen mit kleiner und mittlerer Reichweite Cluster, die nicht durchgängig durch alle Zeitpunkte verlaufen. Es ist klar, dass sich eine hohe Reichweite bei der Ähnlichkeit bezüglich unserer Referenz-Clusterung nicht negativ auswirkt, da unsere Referenz-Clusterung starr ist und keine dynamischen Elemente enthält (siehe Tabelle 6). Der Vorteil einer Reichweite $\mathrm{k} > 1$ ist, dass kurzzeiti-

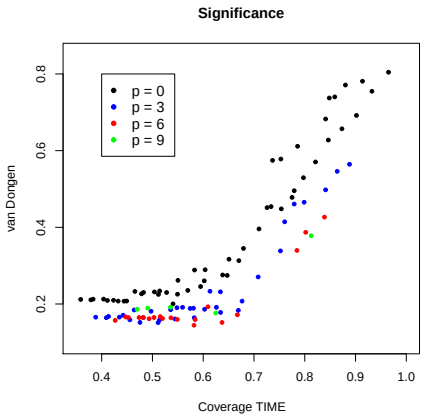
ge Abweichungen und teilweise verrauschte Daten geringere Auswirkungen auf die Clusterung haben und dadurch verlässlichere Ergebnisse liefern. Sind zum Beispiel die Daten eines Zeitschrittes fehlerhaft, kann die Clusterung bei einer Reichweite $k > 1$ trotz allem ein aktzeptables Ergebnis liefern, da die existierenden Interzeitkanten dem Cluster-Verfahren eine erweiterte Sicht ermöglichen.



(a) Vergleich mit dem Ähnlichkeitsmaß $\mathrm{bestmatch}_{\mathrm{no}}$ (2.8.1).



(b) Vergleich mit dem Distanzmaß Variation of Information $\mathcal{NVI}$ (2.8.4).

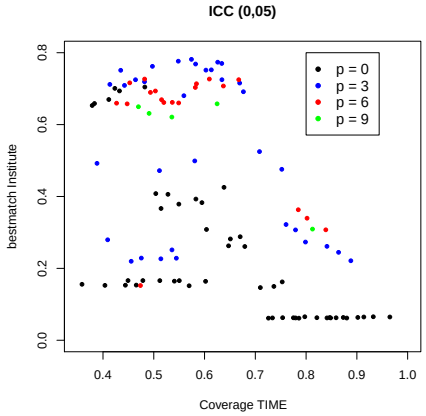


(c) Vergleich mit dem Distanzmaß van Dongen $\mathcal{NVD}$ (2.8.2).

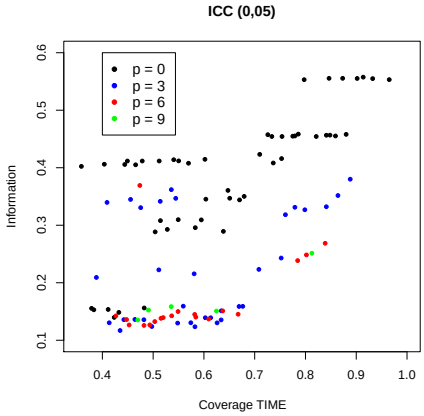
Abbildung 23: *Die Abbildungen stellen die Ähnlichkeit der Significance-Clusterungen zur Referenz-Clusterung in Abhängigkeit von der Coverage der Zeit-Clusterung anhand verschiedener Vergleichsmaße dar. Die Farbe der Punkte steht für die verschiedenen Werte der Schwelle der Graphen.*

Der Parameter α sollte abhängig von dem durchschnittlichen Kantengewicht und der Wahl der Reichweite gewählt werden. Zeitexpandierte Graphen mit kleiner Reichweite benötigen ein größeres Interzeitkantengewicht α als Graphen mit hoher Reichweite, um eine akzeptable Interzeitdichte zu erreichen. Problematisch bleibt weiterhin die fixe Festlegung von α für alle Interzeitkanten. Diese Festlegung bedeutet eine weitere Einschränkung der Dynamik der Clusterung. Es ist offensichtlich, dass die beiden Parameter Interzeitkantengewicht α und Reichweite k einen

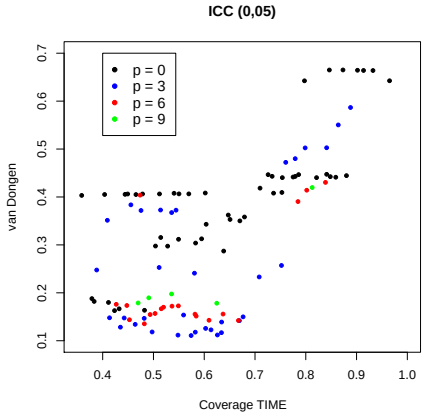
starken Einfluss auf die Coverage der Zeit-Clusterung haben (siehe Abbildung 49 im Anhang).



(a) Vergleich mit dem Ähnlichkeitsmaß $\mathrm{bestmatch}_{\mathrm{no}}$.



(b) Vergleich mit dem Distanzmaß Variation of Information $\mathcal{NVI}$.

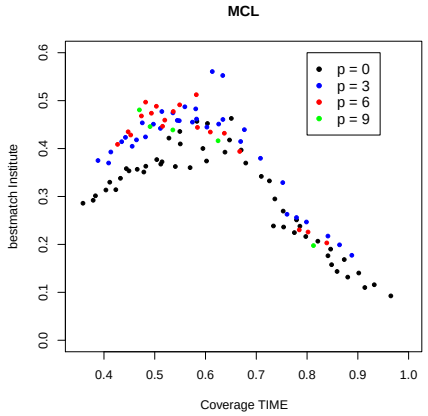


(c) Vergleich mit dem Distanzmaß van Dongen $\mathcal{NVD}$.

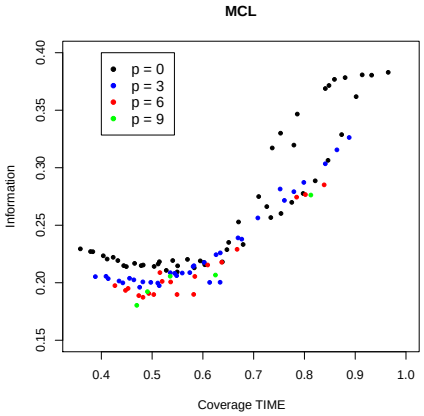
Abbildung 24: *Die Abbildungen stellen die Ähnlichkeit der ICC-Clusterungen mit* $(\mathrm{a}^* = 0,05)$ *zur Referenz-Clusterung in Abhängigkeit von der Coverage der Zeit-Clusterung anhand verschiedener Vergleichsmaße dar. Die Farbe der Punkte steht für die verschiedenen Werte der Schwelle der Graphen.*

Eine Gemeinsamkeit der Ergebnisse der drei Cluster-Verfahren ist die Abhängigkeit der Ähnlichkeit bezüglich der Referenz-Clusterung von dieser Coverage der Zeit-Clusterung. Das heißt, für einen Coveragewert der Zeit-Clusterung zwischen $0,4$ und $0,7$ weisen die Ergebnisse der Cluster-Verfahren die zur Referenz-Clusterung ähnlichsten Ergebnisse auf. Werte unter $0,4$ kommen bei den 108 Graphen allerdings keine vor. Illustriert wird dieser Zusammenhang durch die Abbildung 23. Dabei haben die Significance-Clusterungen in diesem Wertebereich durchgängig eine hohe Ähnlichkeit zu den Lehrstühlen. Beim Markov-Clustering bildet sich eine Glockenkurve aus (siehe Abbildung 25), die ihr Maximum bei einer Coverage der Zeit-Clusterung um $0,6$ erreicht. Sehr unterschiedliche Ergebnisse liefert der ICC-Algorithmus (siehe Abbildung 24) in diesem Wertebereich. Dies wird durch die geringe Anzahl der Cluster bei einigen Ergebnissen

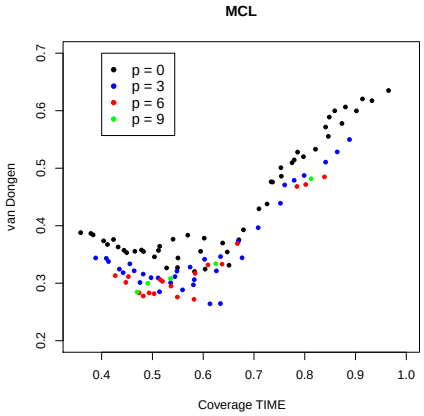
der Clusterungen verursacht (siehe Abbildung 43b, 44b und 45b im Anhang).



(a) Vergleich mit dem Ähnlichkeitsmaß $\mathrm{bestmatch}_{\mathrm{no}}$.



(b) Vergleich mit dem Distanzmaß Variation of Information $\mathcal{NVI}$.



(c) Vergleich mit dem Distanzmaß van Dongen $\mathcal{NVD}$.

Abbildung 25: *Die Abbildungen stellen die Ähnlichkeit der MCL-Clusterungen* ($\mathrm{e} = 3$, $\mathrm{r} = 2$ *und* $\kappa = 120$) *zur Referenz-Clusterung in Abhängigkeit von der Coverage der Zeit-Clusterung anhand verschiedener Vergleichsmaße dar. Die Farbe der Punkte steht für die verschiedenen Werte der Schwelle der Graphen.*

Der Zusammenhang zwischen der Coverage der Zeit-Clusterung und der Ähnlichkeit zur Referenz-Clusterung ist durch den Einfluss der Interzeitdichte auf die Clusterung zu erklären. Die Dichte innerhalb der einzelnen Zeitschritte muss durch die Interzeitkanten aufgewogen werden. Eine niedrigere Coverage der Zeit-Clusterung bedeutet einen höheren Anteil der Interzeitkantengewichte an der Summe aller Gewichte des Graphen. Ist die Coverage der Zeit-Clusterung zu niedrig, droht eine Zerschneidung des Graphen in kleine Zeitschläuche. Ist sie zu hoch, findet das Cluster-Verfahren nur Intrazeit-Cluster. Die optimalen Werte der Coverage liegen unserer Ansicht nach nicht immer zwischen 0,4 und 0,7. Sie sind abhängig von der Dichte der Ausprägungen des dynamischen Graphen für die einzelnen diskreten Zeitpunkte und der Anzahl der diskreten Zeitpunkte d.

Überraschenderweise eignet sich dieses einfache Modell bei einer geeigneten Wahl der Parameter gut zum Erkennen der Strukturen innerhalb des zeitexpandierten Graphen. Bei den beiden Verfahren Iterative-Conductance-Cutting und Greedy-Significance-Clustering erreichen wir hohe Übereinstimmungen mit der Referenz-Clusterung. Es sei vorweg genommen, dass der maximale $\mathrm{bestmatch}_{\mathrm{no}}$ der ICC-Clusterungen von $0,78$ (siehe Ende Abschnitt 4.3.3) bei keiner der anderen Testreihen erreicht wird. Ebenfalls interessante Ergebnisse liefern die Graphen mit geringer Reichweite, da die resultierenden Clusterungen mehr Veränderungen zeigen, als die geglätteten Clusterungen für hohe Reichweiten.

5. Die Methode Normed

Aufgrund der in der ersten Testreihe gewonnenen Erkenntnisse suchen wir eine Möglichkeit, wie die Interzeitkanten dynamischer gestaltet werden können. Ein Problem hierbei sind die nicht normierten Intrazeitgewichte, da die Interzeitkantengewichte den selben Wertebereich haben sollten, wie die Intrazeitkanten. Außerdem erschweren die nicht normierten Gewichte die Interpretation und Vergleichbarkeit der Kanten. Deshalb benutzen wir in der zweiten Testreihe die Methode Normed um die Intrazeitgewichte zu normieren. Mit dieser Testreihe verfolgen wir zwei Ziele. Zum einen wollen wir überprüfen, ob die Normierung der Kanten einen starken Einfluss auf die Cluster-Verfahren ausübt, und zum anderen wollen wir das Verhalten der Clusterungen bezüglich der Coverage der Zeit-Clusterung untersuchen und Vergleiche zur ersten Testreihe aufstellen.

Algorithmus 5 Ablauf bei der Erzeugung und Clusterung der Normed-Graphen

Eingabe: E-Mail-Daten der Fakultät für Informatik eines Zeitraumes von 308 Tagen // siehe 3.1
Ausgabe: geclusterte zeitexpandierte Graphen in Verzeichnis dir

1: $\mathrm{d} \leftarrow 11$ // Festlegung der Anzahl der Zeitschritte
2: **for** $i = 1$ **to** d **do**
3: erzeuge die Matrix $\mathcal{A}(\mathrm{t})$ // Erzeugung wie in 3.1 beschrieben
4: **end for**

5: **for all** Matrizen $\mathcal{A}(\mathrm{t})$ **do**
6: normiere die Einträge der Matrix $\mathcal{A}(\mathrm{t})$ bezüglich des globalen Maximums
7: speichere das Ergebnis als Matrix $\mathcal{A}_{\mathrm{normed}}(\mathrm{t})$
8: **end for**

9: $\mathrm{k_{max}} \leftarrow 9$ // Festlegung der maximalen Reichweite der zu erzeugenden Graphen
10: $\mathrm{k} \leftarrow 1$ // Reichweite k wird auf 1 gesetzt
11: $\alpha \leftarrow 0,1$ // α wird auf $0,1$ gesetzt

12: **while** $\mathrm{k} \leq \mathrm{k_{max}}$ **do** // Schleife: bei jedem Schleifendurchlauf wird die Reichweite k um den Wert 2 erhöht
13: **while** $\alpha \leq 1,0$ **do** // Schleife: bei jedem Schleifendurchlauf wird α um den Wert $0,1$ erhöht
14: $\mathrm{p} \leftarrow 0$ // Schwelle p wird auf 0 gesetzt
15: **while** $\mathrm{p} < \alpha$ **do** // Schleife: bei jedem Schleifendurchlauf wird die Schwelle p um den Wert $0,1$ erhöht
16: **procedure** CREATE GRAPH($\mathrm{d}, \mathrm{k}, \alpha, \mathrm{p}$) // siehe 2.2 und 3.2
17: erzeuge Graph $\overline{\mathcal{G}}^{\mathrm{d}}_{\mathrm{k,p}}$ mit den Matrizen $\mathcal{A}_{\mathrm{normed}}(\mathrm{t})$ und den festgelegten Parametern
18: speichere den Graphen in Verzeichnis dir
19: **end procedure**
20: $\mathrm{p} \leftarrow \mathrm{p} + 0,1$
21: **end while** // Ende der Schwellen-Schleife
22: $\alpha \leftarrow \alpha + 0,1$
23: **end while** // Ende der Interzeitkanten-Schleife
24: $\mathrm{k} \leftarrow \mathrm{k} + 2$
25: **end while** // Ende der Reichweiten-Schleife

26: **for all** Graphen $\mathcal{G}$ in dir **do**
27: clustere $\mathcal{G}$ mit *Greedy-Significance-Clustering*
28: clustere $\mathcal{G}$ mit *Iterative-Conductance-Cutting*
29: **end for**

5.1. Design

Der Unterschied zu den Graphen der ersten Testreihe sind die logarithmisch normierten Intrazeitgewichte (siehe Abschnitt 3.2.1). Der Wertebereich der neuen Intrazeitgewichte ist $[0, 1]$. Auch in dieser Testreihe verwenden wir ein starres Interzeitgewicht α. Dabei durchläuft α alle Vielfachen von $0,1$ im Intervall $(0, 1]$. Die Schwelle p nimmt alle Vielfachen von $0,1$ im Intervall $(\alpha, 1)$ an. Der genaue Ablauf der Erzeugung der Graphen ist in Algorithmus 5 beschrieben. In dieser Testreihe entscheiden wir uns zur Clusterung der Graphen mit den Verfahren Iterative-Conductance-Cutting und Greedy-Significance-Clustering. Das MCL-Verfahren verwenden wir aufgrund der problematischen Anpassung der Parameter nicht mehr.

5.2. Analyse

Schwachstelle des verwendeten Modells sind erneut die starren Interzeitkantengewichte. Die Intrazeitkantengewichte sind jetzt normiert und ermöglichen eine bessere Interpretation einzelner Gewichte. Die logarithmische Normierung schwächt die Gewichtung von sehr hohen Kantengewichten ab. Dies hat den Vorteil, dass bei einem einzelnen sehr hohen Kantengewicht nicht alle anderen Gewichte gegen den Wert 0 tendieren. Der zusätzliche Aufwand der Normierung der Intrazeitgewichte liegt in $\mathcal{O}(\mathrm{n}^2\mathrm{d})$ oder falls dies echt kleiner ist in $\Theta(\mathrm{m}_{\mathrm{intra}})$, wobei $\mathrm{m}_{\mathrm{intra}}$ die Anzahl der Intrazeitkanten, n die Anzahl der Knoten der Knotenmenge $\mathcal{V}$ des zugrundeliegenden dynamischen Graphen und d die Anzahl der Zeitschritte des zeitexpandierten Graphen ist. Der Gesamtaufwand der Erzeugung eines zeitexpandierten Graphen mit den gewählten Methoden ergibt sich aus dem Aufwand der Normierung der Intrazeitkantengewichte $\mathcal{O}(\mathrm{n}^2\mathrm{d})$ und dem Aufwand für die Interzeitkantengewichte $\mathcal{O}(\mathrm{n_d k})$. Es gilt $\mathrm{n_d k} < \mathrm{ndk}$, somit dominiert der Aufwand zur Berechnung der Intrazeitkantengewichte den Gesamtaufwand. Er liegt in $\mathcal{O}(\mathrm{n}^2\mathrm{d})$. Die Laufzeit der Clusterung der zeitexpandierten Graphen bleibt unverändert.

Wir vermuten, dass mit den normierten Kanten ähnlich gute Ergebnisse erreicht werden können, wie bei den Graphen der ersten Testreihe. Falls die Normierung sich nicht negativ auswirkt, werden wir eine weitere Testreihe mit normierten Intrazeitkantengewichten und mit veränderlichen Interzeitkanten durchführen. Ebenfalls wollen wir die in der ersten Testreihe entdeckten Zusammenhänge weiter untersuchen. Als Referenz-Clusterung dient weiterhin die Institute-Clusterung, die durch die Lehrstühle vorgegeben ist.

5.3. Experimente

Mit Hilfe unseres Frameworks erzeugen wir 275 Graphen. Allerdings verwerfen wir alle 75 Graphen mit Schwelle $\mathrm{p} = 0,5$ und höher. Der Grund dafür ist die stark reduzierte Anzahl von Knoten bei diesen Werten. Von den anfangs 4677 Knoten existieren bei Schwelle $\mathrm{p} = 0,5$ lediglich noch 728. Für die Schwelle $\mathrm{p} = 0,4$ existieren ähnlich zur Schwelle $\mathrm{p} = 9$ bei der ersten Testreihe 1610 Knoten. Die Graphen mit Schwelle $\mathrm{p} = 0,1$ stimmen mit Graphen ohne Schwelle überein, also haben alle Kanten ein Gewicht größer als $0,1$.

5.3.1. Überprüfung der Referenz-Clusterung

Wir berechnen die Qualitätsindizes der Referenz-Clusterung für die verbliebenen Graphen. Die Durchschnittswerte sind nahezu identisch mit den Werten der ersten Testreihe und bestätigen die gewählte Referenz-Clusterung.

Index	$\mathrm{cov}_{\mathrm{w}}$	$\mathrm{mod}_{\mathrm{w}}$	δ_{d}
1. Quartil	0,8782	0,8079	0,7131
3. Quartil	0,9521	0,8783	0,8469
Minimum	0,7371	0,6625	0,5889
Maximum	0,9811	0,9017	0,8729
Mittelwert	0,9088	0,8364	0,7621

Tabelle 7: *Indizes für die Referenz-Clusterung.*

Index	$\mathrm{cov}_{\mathrm{w}}$	$\mathrm{mod}_{\mathrm{w}}$	δ_{d}	$\mathrm{bestmatch}_{\mathrm{no}}$ Institute
1. Quartil	0,8967	0,7939	0,7365	0,5186
3. Quartil	0,9556	0,8996	0,9205	0,6494
Minimum	0,7169	0,6287	0,5474	0,0656
Maximum	0,9773	0,9343	0,9786	0,7442
Mittelwert	0,9135	0,8344	0,8145	0,5501

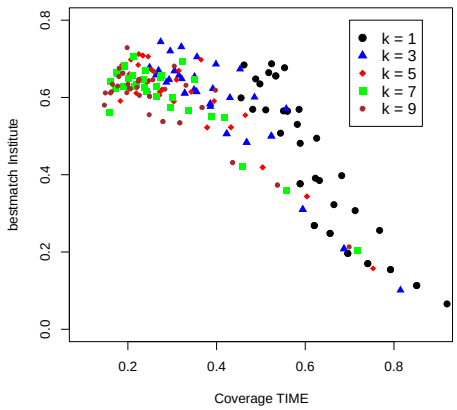
Tabelle 8: *Indizes für die Significance-Clusterungen der zeitexpandierten Graphen.*

5.3.2. Greedy-Significance-Clustering

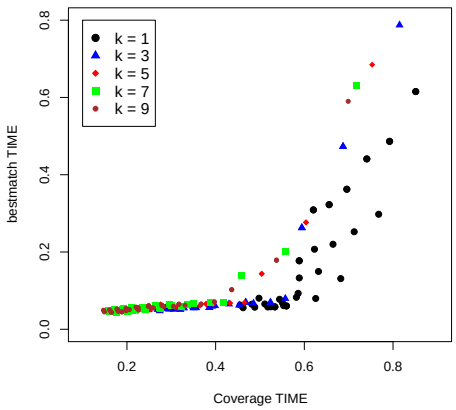
Die Ergebnisse des Greedy-Significance-Clustering gleichen den Werten aus der Normal-Testreihe (siehe Tabelle 8). Das spricht dafür, dass sich die Methode ebenso gut zur Erzeugung unseres Graphen eignet wie die Methode Normal.

Zusätzlich wollen wir in dieser Testreihe den Einfluss der Coverage der Zeit-Clusterung auf die Cluster-Verfahren betrachten. Wie wir in der ersten Testreihe gesehen haben, wird diese Coverage stark von den Parametern α und k beeinflusst. Dass die Erhöhung der Schwelle sich unterschiedlich auf die Coverage der Zeit-Clusterung auswirken kann, soll nicht unerwähnt bleiben.

Die Abbildung 26 bestätigt den Zusammenhang zwischen der Coverage der Zeit-Clusterung und der Güte der Clusterung bezüglich der Referenz-Clusterung. Eine zu niedrige Interzeitdichte führt zur Ausprägung von Intrazeit-Clustern. Die zur Referenz-Clusterung ähnlichsten Ergebnisse werden für Graphen, deren Coverage der Zeit-Clusterung kleiner als $0,4$ ist, erreicht. Dieser Wertebereich unterscheidet sich von dem Wertebereich aus der ersten Testreihe. Das heißt, es gibt keinen fixen Wertebereich der Coverage der Zeit-Clusterung, für die Clusterungen erreicht werden können, die über die Zeit hinweglaufen, sondern dieser Wertebereich ist abhängig von den Eigenschaften des Graphen. Für kleine Werte um $0,2$ der Coverage der Zeit-Clusterung erkennen wir einen leichten Abfall des $\mathrm{bestmatch}_{\mathrm{no}}$ bezüglich der Referenz-Clusterung. Denn eine zu starke Erhöhung von α und k kann zu Clustern führen, die aus allen Repräsentanten eines Knotens bestehen. Dies ist gleichbedeutend mit einer Reduzierung der Ähnlichkeit der Clusterung zur Referenz-Clusterung.

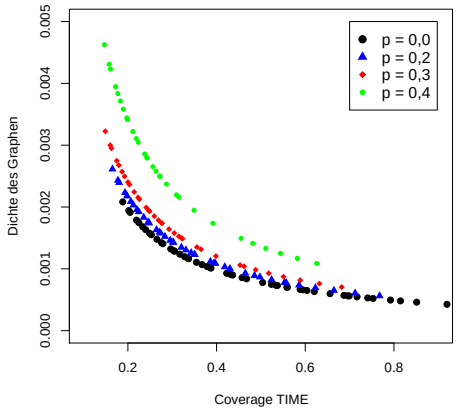


(a) Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der Significance-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$.

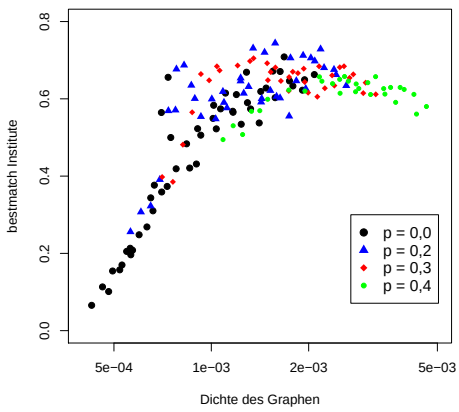


(b) Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der Significance-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$.

Abbildung 26: *Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der Significance-Clusterungen zur Referenz-Clusterung und zur Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Reichweiten der Graphen.*



(a) Die Dichte der Graphen abhängig von der Coverage der Zeit-Clusterung.



(b) Die Ähnlichkeit der Significance-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$.

Abbildung 27: *Die Plots zeigen die Dichte der Graphen abhängig von der Coverage der Zeit-Clusterung und den Zusammenhang von der Dichte und dem* $\mathrm{bestmatch}_{\mathrm{no}}$ *der Significance-Clusterungen bezüglich der Referenz-Clusterung. Wir benutzen in 27b eine logarithmische Skala für die Dichte. Die Farben stehen in beiden Plots für die unterschiedlichen Schwellen der Graphen.*

Die Coverage der Zeit-Clusterung ist nicht der einzige Index, anhand dessen man eine Einteilung der Clusterungen bezüglich der Ähnlichkeit zur Referenz-Clusterung finden kann. Es gibt ebenfalls einen Zusammenhang zwischen dem $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung und der Dichte des Graphen (siehe dazu Abbildung 27). Die Veränderungen der Dichte bei konstanter Schwelle können nur durch Änderungen in der Interzeitdichte entstehen, da alle Intrazeitkanten unverändert bleiben. Die Veränderungen der Interzeitdichte werden hier vollständig von den Parametern α und k hervorgerufen. Eine Anhebung eines dieser Parameter erhöht die Dichte, eine Senkung verringert sie. Der optimale Wert für die Dichte des Graphen scheint sich in unserer Testreihe bei ungefähr $0,0018$ zu befinden. Dieser Sachverhalt wird durch die Abbildung 27b illustriert. Ohne Referenz-Clusterung wäre es schwierig, Werte für die Parameter Reichweite k und Interzeitkantengewicht α zu bestimmen, mit denen aussagekräftige Clusterungen erreicht werden können.

5.3.3. Iterative-Conductance-Cutting

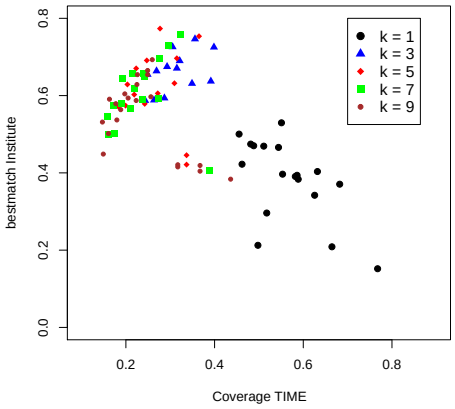
Bei der Verwendung des Iterative-Conductance-Cutting clustern wir die 200 Graphen mit Schwelle $\mathrm{p} \leq 0,4$ zunächst mit Parameter $\mathrm{a}^* = 0,05$. Ähnlich zu der ersten Testreihe gibt es viele Clusterungen mit weniger als zehn Clustern. Diese werden von uns im Folgenden nicht berücksichtigt. Es bleiben 83 Graphen mit zehn und mehr Clustern.

Das Iterative-Conductance-Cutting liefert mit der Methode Normed ähnlich gute Ergebnisse wie in der ersten Testreihe (siehe Tabelle 9). Allerdings ist der Anteil der Clusterungen mit sehr wenigen Clustern bei gleichem Parameter a^* stark gestiegen. Der Algorithmus bricht bei der Modellierung mit der Normed-Methode im Schnitt früher ab. Sind es bei der Methode Normal noch ein Viertel der Clusterungen, die bei $\mathrm{a}^* = 0,10$ weniger als zehn Cluster aufweisen, sind es jetzt bereits 43 %. Für $\mathrm{a}^* = 0,05$ haben 56 % der Clusterungen weniger als zehn Cluster. Dies könnte auf die geringeren Unterschiede der Intrazeitkantengewichte bei der Verwendung der Methode Normed zurückzuführen sein.

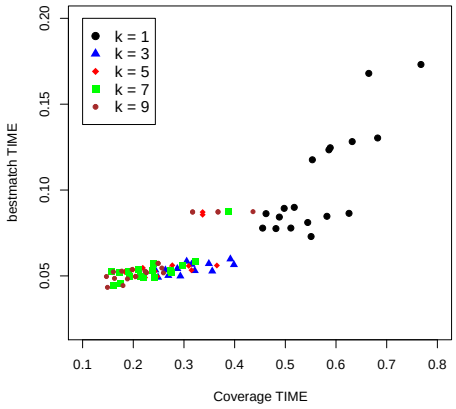
Index	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	$\mathrm{bestmatch_{no}}$ Institute
1. Quartil	0,9465	0,8158	0,7365	0,4213
3. Quartil	0,9652	0,9113	0,9205	0,6502
Minimum	0,8966	0,5303	0,5474	0,1519
Maximum	0,9766	0,9273	0,9786	0,7734
Mittelwert	0,9523	0,8391	0,8145	0,5395

Tabelle 9: *Indizes für das Iterative-Conductance-Cutting der* 83 *ausgewählten zeitexpandierten Graphen für* $\mathrm{a}^* = 0,05$.

Das Verhalten hinsichtlich der Coverage der Zeit-Clusterung ist analog zum Greedy-Significance-Clustering (siehe Abbildung 28). Die Ausnahme bildet eine kleine Ausreißergruppe, deren Graphen alle die Schwelle $\mathrm{p} = 0,0$ haben (siehe Markierung in Abbildung 29). Diese hat trotz vergleichbarer Werte der Coverage der Zeit-Clusterung einen niedrigeren $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung als die anderen Clusterungen. Die Clusterungen der Ausreißergruppe enthalten alle einen Cluster, der die Mehrheit der Referenz-Cluster in sich vereint. Das führt zu einer erhöhten Ähnlichkeit zur Zeit-Clusterung, und zu einer Abnahme des $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung. Die Wahl des Parameter $\mathrm{a}^* = 0,05$ des ICC-Verfahrens scheint für

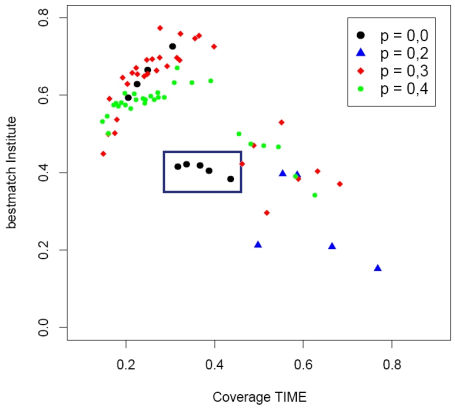


(a) Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$.

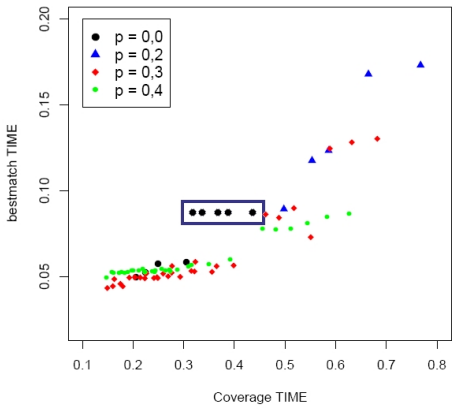


(b) Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$.

Abbildung 28: *Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der ICC-Clusterungen zur Referenz-Clusterung und zur Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Reichweiten der Graphen.*



(a) Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$.



(b) Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$.

Abbildung 29: *Die Plots entsprechen den Plots in Abbildung 28, anstatt der unterschiedlichen Reichweiten repräsentieren die Farben hier jedoch die unterschiedlichen Schwellen der Graphen.*

diese Graphen zu niedrig gewählt.

Die Ähnlichkeit zur Referenz-Clusterung steigt für eine sinkende Coverage der Zeit-Clusterung. Für hohe Werte der Reichweite k und einen hohen Wert für Interzeitkantengewicht α besteht allerdings die Gefahr einer Zersplitterung der Clusterung in Knotenschläuche, die aus allen Repräsentanten eines Knotens bestehen. Aus diesem Grund nimmt die Ähnlichkeit zur Referenz-Clusterung für sehr kleine Werte der Coverage der Zeit-Clusterung wieder ab (siehe Abbildung 28a). Für die Graphen mit Reichweite $\mathrm{k} = 1$ besteht eine erhöhte Ähnlichkeit der Clusterung zur Zeit-Clusterung. Dadurch verschlechtern sich die Werte des $\mathrm{bestmatch}_{\mathrm{no}}$ bezüglich der Referenz-Clusterung.

5.4. Fazit

Es gibt einen deutlichen Zusammenhang zwischen der Ähnlichkeit der Clusterung bezüglich der Referenz-Clusterung und der Coverage der Zeit-Clusterung. Gleiches gilt für die Dichte des Graphen. Dies ist wenig verwunderlich, da diese beiden Indizes direkt von den Parametern der zeitexpandierten Graphen abhängen. Zwischen der Reichweite k und dem Interzeitkantengewicht α und den beiden Indizes besteht ein klarer Zusammenhang. Erhöhen wir einen der zwei Parameter, so erhöht sich die Dichte des Graphen und die Coverage der Zeit-Clusterung verringert sich. Interessant dabei ist, dass die Coverage der Zeit-Clusterung sich nicht proportional zur Schwelle verhält. Durch die Erhöhung der Schwelle fallen Intrazeitkanten weg, deren Gewicht kleiner ist als die festgelegte Schwelle. Ist ein Knoten zu keiner Intrazeitkante mehr inzident nachdem die Schwelle erhöht wurde, wird er aus dem Graphen entfernt. Logischerweise verschwinden damit alle Interzeitkanten des Knotens zu seinen anderen Repräsentanten im Graphen. Wenn das Verhältnis der Intrazeitgewichte zu den Gesamtgewichten der wegfallenden Knoten größer bzw. kleiner als die Coverage der Zeit-Clusterung ist, verkleinert bzw. vergrößert sich diese.

Eine Möglichkeit zur Bewertung einer Clusterung eines zeitexpandierten Graphen ohne Referenz-Clusterung könnte ihre Ähnlichkeit bezüglich der Zeit-Clusterung sein, da sie unabhängig von der Struktur des Graphen ist. Eine zu hohe Ähnlichkeit zur Zeit-Clusterung bedeutet eine Tendenz der Clusterung hin zu reinen Intrazeitclustern. Daher erscheint ein Vergleich verschiedener Clusterungen eines zeitexpandierten Graphen mit Hilfe der Ähnlichkeit zur Zeit-Clusterung sinnvoll.

Die Methode Normed liefert uns ähnliche Ergebnisse wie die erste Testreihe. Sie eignet sich ähnlich gut zur Modellierung, wie die Methode Normal. Daher nehmen wir nun unser nächstes Ziel in Angriff, das Problem der starren Interzeitkanten zu beseitigen.

6. Die Methode Cosine-Time

Bisher führten wir unsere Testreihen mit starren Interzeitkantengewichten α durch. Ein weitverbreitetes Ähnlichkeitsmaß ist die in Abschnitt 2.6 vorgestelle Cosine-Similarity. Mit ihrer Hilfe werden wir in dieser Testreihe die Interzeitkantengewichte berechnen. Die Gewichte spiegeln daher die Ähnlichkeit der Repräsentanten des Knotens in den zwei entsprechenden Zeitpunkten wider.

Algorithmus 6 Ablauf bei der Erzeugung und Clusterung der Cosine-Time-Graphen

Eingabe: E-Mail-Daten der Fakultät für Informatik eines Zeitraumes von 308 Tagen // siehe 3.1
Ausgabe: geclusterte zeitexpandierte Graphen in Verzeichnis dir

1: $\mathrm{d} \leftarrow 11$ // Festlegung der Anzahl der Zeitschritte
2: **for** $i = 1$ **to** d **do**
3: erzeuge die Matrix $\mathcal{A}(\mathrm{t})$ // Erzeugung wie in 3.1 beschrieben
4: **end for**

5: **for all** Matrizen $\mathcal{A}(\mathrm{t})$ **do**
6: normiere die Einträge der Matrix $\mathcal{A}(\mathrm{t})$ bezüglich des globalen Maximums
7: speichere das Ergebnis als Matrix $\mathcal{A}_{\mathrm{normed}}(\mathrm{t})$
8: **end for**

9: $\mathrm{k_{max}} \leftarrow 9$ // Festlegung der maximalen Reichweite der zu erzeugenden Graphen
10: **for all** Knoten v in Knotenmenge $\mathcal{V}$ **do**
11: berechne alle Interzeitkantengewichte für die maximale Reichweite $\mathrm{k_{max}} = 9$ der Repräsentanten des Knotens
12: speichere die Ergebnisse in Matrix $\mathcal{A}_v \in \mathbb{R}^{\mathrm{d} \times \mathrm{d}-1}$
13: **end for**

14: $\mathrm{k} \leftarrow 1$ // Reichweite wird auf 1 gesetzt
15: **while** $\mathrm{k} \leq \mathrm{k_{max}}$ **do** // Schleife: bei jedem Schleifendurchlauf wird die Reichweite k um den Wert 2 erhöht
16: $\mathrm{p} \leftarrow 0$ // Schwelle wird auf 0 gesetzt
17: **while** $\mathrm{p} \leq 0,45$ **do** // Schleife: bei jedem Schleifendurchlauf wird die Schwelle p um den Wert $0,05$ erhöht
18: **procedure** CREATE GRAPH(d, k, p) // siehe 2.2 und 3.2
19: erzeuge Graph $\overline{\mathcal{G}}^{\mathrm{d}}_{\mathrm{k,p}}$ mit den Matrizen $\mathcal{A}_{\mathrm{normed}}(\mathrm{t})$, den Matrizen $\mathcal{A}_v$ und den festgelegten Parametern
20: speichere den Graphen in Verzeichnis dir
21: **end procedure**
22: $\mathrm{p} \leftarrow \mathrm{p} + 0,05$
23: **end while** // Ende der Schwellen-Schleife
24: $\mathrm{k} \leftarrow \mathrm{k} + 2$
25: **end while** // Ende der Reichweiten-Schleife

26: **for all** Graphen $\mathcal{G}$ in dir **do**
27: clustere $\mathcal{G}$ mit *Greedy-Significance-Clustering*
28: clustere $\mathcal{G}$ mit *Iterative-Conductance-Cutting*
29: **end for**

6.1. Design

In dieser Testreihe werden wir die Methode Cosine-Time (siehe Abschnitt 3.2.1) zur Berechnung der Interzeitkantengewichte verwenden. Für die Berechnung der Intrazeitkantengewichte

verwenden wir die Methode Normed. Durch die Methode Cosine-Time erhoffen wir uns eine höhere Dynamik der Clusterung. Wir wählen nicht etwa einen fixen Interzeitkantenwert, der ausreicht die Cluster über die Zeit hinweg auszudehnen, vielmehr ergibt sich die Zeitkante zwischen Knoten $v_{\mathrm{x}}^{\mathrm{t_i}}$ und $v_{\mathrm{x}}^{\mathrm{t_j}}$ aus der Ähnlichkeit der E-Mail-Kontakte des Knotens in den Zeitpunkten $\mathrm{t_i}$ und $\mathrm{t_j}$, falls die Reichweite $\mathrm{k} \geq |\mathrm{j} - \mathrm{i}|$ und $\mathrm{j} \neq \mathrm{i}$:

$$\mathrm{sim}(v_{\mathrm{x}}^{\mathrm{t_i}}, v_{\mathrm{x}}^{\mathrm{t_j}}) = \frac{\sum_{\mathrm{l}=1}^{\mathrm{n}} \left(\mathrm{v}_{\mathrm{x}}^{\mathrm{t_i}}(l) \cdot \mathrm{v}_{\mathrm{x}}^{\mathrm{t_j}}(l) \right)}{\sqrt{\sum_{\mathrm{l}=1}^{\mathrm{n}} \mathrm{v}_{\mathrm{x}}^{\mathrm{t_i}}(l)^2 \cdot \sum_{\mathrm{l}=1}^{\mathrm{n}} \mathrm{v}_{\mathrm{x}}^{\mathrm{t_j}}(l)^2}}.$$

Die Anzahl der variablen Parameter unserer Graphen reduziert sich auf zwei. Die Reichweite, welche die Werte 1, 3, 5, 7 und 9 durchlaufen wird und die Schwelle p. Diese wird schrittweise um $0,05$ angehoben. Der genaue Ablauf der Erzeugung und Clusterung der zeitexpandierten Graphen kann man Algorithmus 6 entnehmen.

6.2. Analyse

Unsere Erwartungen an diese Testreihe sind die Folgenden: Durch die variablen Interzeitkanten werden wir Veränderungen innerhalb der Struktur des Graphen erkennen. Verändern Gruppierungen innerhalb des Graphen ihr Verhalten oder ändert sich ihre Zusammensetzung, sollte es nun möglich sein, dieses zu erkennen.

Ausgehend von den Matrizen $\mathcal{A}(\mathrm{t})$ ergibt sich die Laufzeit der Erzeugung des zeitexpandierten Graphen aus dem Aufwand zur Erzeugung der normierten Matrizen $\mathcal{O}(\mathrm{n}^2\mathrm{d})$ und aus dem Aufwand zur Erzeugung der Interzeitkantengewichte. Dieser ergibt sich aus dem Aufwand der Cosine-Similarity $\mathcal{O}(\mathrm{n})$ und der Anzahl z der Interzeitkanten, wobei $\mathrm{z} < \mathrm{ndk}$. Damit ist der Gesamtaufwand zur Erzeugung des zeitexpandierten Graphen in $\mathcal{O}(\mathrm{n}^2\mathrm{d}) + \mathcal{O}(\mathrm{n}^2\mathrm{dk}) = \mathcal{O}(\mathrm{n}^2\mathrm{d} + \mathrm{n}^2\mathrm{dk}) = \mathcal{O}(\mathrm{n}^2\mathrm{dk})$. Dabei dominiert der Aufwand zur Berechnung der Interzeitkantengewichte die Erzeugung des zeitexpandierten Graphen. Die Laufzeit der Clusterung der zeitexpandierten Graphen bleibt unverändert. Die Referenz-Clusterung ist wieder die durch die Lehrstühle vorgegebene Institute-Clusterung.

6.3. Experimente

Wir erzeugen 50 Graphen mit unserem Framework und clustern diese mit dem Iterative-Conductance-Cutting und dem Greedy-Significance-Clustering. Wir verwenden nur Graphen mit Schwelle $\mathrm{p} = 0,45$ und geringer, da ansonsten der Informationsverlust zu hoch ist. Bei einem Graphen mit Schwelle $\mathrm{p} = 0,50$ existieren lediglich 728 Knoten von ursprünglich 4677. Die Schwellen $\mathrm{p} = 0,05$ und $\mathrm{p} = 0,10$ haben nur geringe Auswirkungen auf den Graphen. Die Anzahl der Knoten wird durch sie nicht verringert und die Anzahl der Kanten ändert sich minimal.

6.3.1. Überprüfung der Referenz-Clusterung

Zunächst berechnen wir die Durchschnittswerte für die gewichtetete Coverage, die gewichtete Modularity und die durchschnittliche Interclusterconductance der Referenz-Clusterung. Alle Werte überzeugen und bestätigen die Referenz-Clusterung als signifikante Clusterung (siehe

Tabelle 10). Das Durchschnittsgewicht der Intrazeitkanten eines zeitexpandierten Graphen mit Reichweite $\mathrm{k} = 1$ und Schwelle $\mathrm{p} = 0$ ist $0,22$. Im Gegensatz dazu beträgt das Durschnittsgewicht der Interzeitkanten $0,69$. Bei Schwelle $\mathrm{p} = 0,45$ ist das durchschnittliche Intrazeitkantengewicht $0,54$ und das durchschnittliche Interzeitkantengewicht $0,84$. Diese relativ hohen Werte der Interzeitkantengewichte sprechen dafür, dass sich das Verhalten der Repräsentanten eines Knoten innerhalb der verschiedenen Zeitschritte unserer zeitexpandierten Graphen kaum verändert. Dieser Sachverhalt ist ein weiterer Indikator für die Güte der Institute-Clusterung als Referenz-Clusterung.

Index	cov_w	mod_w	δ_d
1. Quartil	0,9125	0,8373	0,7218
3. Quartil	0,9659	0,8832	0,8514
Minimum	0,8171	0,7472	0,5836
Maximum	0,9839	0,8986	0,8717
Mittelwert	0,9308	0,8548	0,7835

Tabelle 10: *Indizes für die Referenz-Clusterung der Graphen der Testreihe.*

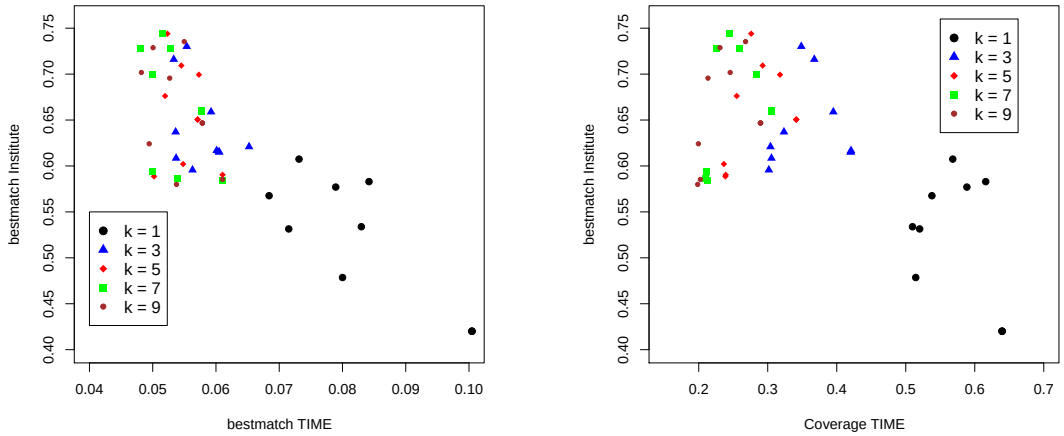
Index	cov_w	mod_w	δ_d	$\mathrm{bestmatch}_\mathrm{no}$ Institute
1. Quartil	0,9246	0,8582	0,7738	0,5877
3. Quartil	0,9693	0,9104	0,9640	0,6681
Minimum	0,8569	0,7668	0,7180	0,4201
Maximum	0,9927	0,9276	0,9953	0,7443
Mittelwert	0,9422	0,8770	0,8799	0,6247

Tabelle 11: *Indizes für die Significance-Clusterungen der Graphen der Testreihe.*

6.3.2. Greedy-Significance-Clustering

Wir clustern die 50 Graphen zunächst mit dem Greedy-Significance-Clustering (siehe Abschnitt 2.5.1) und berechnen erneut die Durchschnittswerte der ausgewählten Bewertungsindizes. Sie sind in Tabelle 11 zu finden. Auffällig sind die deutlich geringeren Differenzen der Minima und Maxima der Indizes. Das heißt, die Ergebnisse dieser Testreihe weisen beim Greedy-Significance-Clustering geringere Unterschiede auf als bei den bisherigen Testreihen. Durch die durchschnittlich sehr hohen Werte der Interzeitkantengewichte gibt es keine Graphen, die der Zeit-Clusterung stark ähneln. Alle Graphen, die eine höhere Reichweite haben, weisen durchgängig einen sehr hohen $\mathrm{bestmatch}_\mathrm{no}$ bezüglich der Referenz-Clusterung auf (siehe Abbildung 30).

Für einen Vergleich der Unterschiede zwischen der Verwendung der statischen Interzeitkantengewichte α und der dynamischen Interzeitkantengewichte mit der Methode Cosine-Time untersuchen wir die Ergebnisse der Clusterungen der Graphen der Cosine-Time-Testreihe und der Normed-Testreihe. Dabei verwenden wir die Graphen der Cosine-Time-Testreihe mit Reichweite $\mathrm{k} = 1$ und den Schwellen $\mathrm{p} = 0,0$, $\mathrm{p} = 0,1$, $\mathrm{p} = 0,2$, $\mathrm{p} = 0,3$ und $\mathrm{p} = 0,4$, sowie die entsprechenden Graphen der Normed-Testreihe, die die starren Interzeitkantengewichte $\alpha = 0,7$ und $\alpha = 1,0$ haben. Die Wahl auf die beiden Werte der Interzeitkantengewichte erfolgt aufgrund



(a) Vergleich der Ähnlichkeit der Significance-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\text{time}}$ mit der Ähnlichkeit zur Referenz-Clusterung $\mathcal{C}$ anhand des $\text{bestmatch}_{\text{no}}(\mathcal{C}_{\text{time}}, \mathcal{C}')$ und des $\text{bestmatch}_{\text{no}}(\mathcal{C}, \mathcal{C}')$.

(b) Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der Significance-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des $\text{bestmatch}_{\text{no}}(\mathcal{C}, \mathcal{C}')$.

Abbildung 30: *Einfluss der Reichweite* k *auf den* $\text{bestmatch}_{\text{no}}$ *der Significance-Clusterungen bezüglich der Referenz-Clusterung. Dabei stehen die Farben für die verschiedenen Reichweiten der Graphen.*

des berechneten Durchschnittswertes der Interzeitkantengewichte von $0,69$ der Cosine-Time-Graphen und der Tatsache, dass hohe Interzeitkantengewichte zu einer besseren Bewertung der Referenz-Clusterung führen. Zunächst vergleichen wir die Normed-Graphen mit Interzeitkantengewicht $\alpha = 0,7$ mit den gewählten Graphen der Cosine-Time-Testreihe. Für niedrige Schwellen haben die Clusterungen der Normed-Graphen eine starke Ähnlichkeit zur Zeit-Clusterung. Es gibt in jedem Zeitschritt einen Cluster, der die Mehrheit der Referenz-Cluster enthält. Teilweise ziehen sich diese Cluster über zwei oder drei Zeitschritte hinweg. Ab dem Zeitschritt 4 gibt es zumindest einige Referenz-Cluster, die über den gesamten restlichen Zeitbereich einen Cluster ausbilden. Insgesamt gesehen haben diese Clusterungen der Normed-Graphen mit einem $\text{bestmatch}_{\text{no}}$ kleiner als $0,3$ nur eine geringe Ähnlichkeit zur Referenz-Clusterung. Für höhere Werte der Schwellen wird die Ähnlichkeit zur Referenz-Clusterung wieder signifikanter. Im Gegensatz dazu hat die Clusterung der Cosine-Time-Graphen schon für niedrige Schwellenwerte eine deutliche Ähnlichkeit zur Referenz-Clusterung. Für den Cosine-Time-Graphen mit Schwelle $\text{p} = 0$ beträgt der $\text{bestmatch}_{\text{no}}$ bezüglich der Referenz-Clusterung bereits $0,42$. Auffällig bei den Clusterungen der gewählten Cosine-Time-Graphen ist ein starker Umbruch einiger Cluster ab Zeitschritt 4. Diesen kann man in den Normed-Graphen mit höherer Schwelle ebenfalls feststellen. Dieser Umbruch wird im folgenden Kapitel Gegenstand genauerer Untersuchungen. Wir halten fest, dass die Clusterung der Cosine-Time-Graphen bei ähnlichem Durchschnittsgewicht der Interzeitkantengewichte, aussagekräftigere Ergebnisse liefert als die Normed-Graphen mit statischen Interzeitkanten.

Betrachten wir nun die Normed-Graphen mit maximalem Interzeitkantengewicht $\alpha = 1,0$. Die Ähnlichkeit zur Referenz-Clusterung ist durchgängig sehr hoch. Der $\text{bestmatch}_{\text{no}}$ bezüglich der

Referenz-Clusterung liegt zwischen $0,56$ und $0,75$. Bei einer näheren Betrachtung ergeben sich kaum Veränderungen der Clusterungen zwischen den einzelnen Zeitschritten. Im Vergleich zu den Clusterungen der Cosine-Time-Graphen ist hier ein Umbruch selbst für höhere Schwellen kaum feststellbar. Die Ergebnisse im folgenden Kapitel 7.1 werden uns bestätigen, dass die Methode Cosine-Time hier eine bessere Wiedergabe der dynamischen Veränderungen unseres E-Mail-Netzwerkes liefert. Für höhere Reichweiten ist der Umbruch bei fast allen Graphen der Testreihen nicht mehr feststellbar.

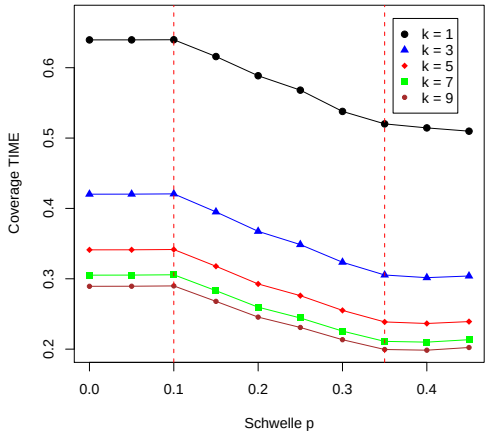
Im Folgenden untersuchen wir den Einfluss der Reichweite auf die Clusterung der Graphen der Cosine-Time-Testreihe. Der Durchschnitt des $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung beträgt bei Reichweite $\mathrm{k} = 1$ ungefähr $0,51$. Für $\mathrm{k} = 3$ erhöht er sich auf $0,64$ und bleibt bei einer weiteren Erhöhung der Reichweite annähernd konstant (siehe dazu Abbildung 31c). Eine Begründung dafür ist, dass durch die erhöhte Reichweite die Sichtweite des Cluster-Verfahrens vergrößert wird und kurzzeitige Veränderungen der Ausrichtung der Knoten weniger ins Gewicht fallen. Die Clusterungen werden konformer gegenüber dem durchschnittlichen Verhalten der Knoten. Ein weiterer Beleg für diesen Zusammenhang liefert uns eine Betrachtung der *stabilen Knotenmenge* eines Clusters $\mathrm{C_i}$. Diese beinhaltet alle Knoten des dynamischen Graphen, deren Repräsentanten in jedem Zeitschritt in Cluster $\mathrm{C_i}$ enthalten sind, über den der Cluster hinwegreicht. Wir bezeichnen diese Menge als *Kern* von $\mathrm{C_i}$. Das heißt, der Kern eines Clusters $\mathrm{C_i}$ einer Clusterung $\mathcal{C}$ eines zeitexpandierten Graphen ist die Menge der Knoten des dynamischen Graphen $\mathcal{G}(\mathrm{t})$ mit Knotenmenge $\mathcal{V}$, deren Repräsentanten in jedem Zeitschritt der vom Cluster überdeckten Zeitspanne in Cluster $\mathrm{C_i}$ vorkommen. Sei $[\mathrm{t_a}, \mathrm{t_e}]$ der Zeitbereich über den sich der Cluster $\mathrm{C_i}$ erstreckt, dann ergibt sich der Kern aus

$$\mathrm{kernel}(\mathrm{C_i}) = \left\{ v_\mathrm{i} \in \mathcal{V} \mid \forall \mathrm{t_i} \in [\mathrm{t_a}, \mathrm{t_e}] : v_\mathrm{i}^{\mathrm{t_i}} \in \mathrm{C_i} \right\} \quad . \tag{22}$$

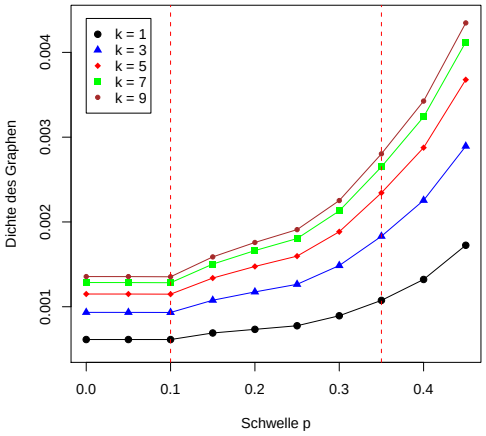
Der Anteil des Kerns an Cluster $\mathrm{C_i}$ ergibt sich aus dem Verhältnis der Mächtigkeit von $\mathrm{kernel}(\mathrm{C_i})$ und der Anzahl aller Knoten aus $\mathcal{V}$, die einen Repräsentanten in Cluster $\mathrm{C_i}$ haben. Dieses Verhältnis ist ein Maß für die stabilen Elemente eines Clusters. Erhöht man die Reichweite von $\mathrm{k} = 1$ auf $\mathrm{k} = 3$ findet eine deutliche Vergrößerung des Kerns statt. So erhöht sich für Schwelle $\mathrm{p} = 0$ der durchschnittliche Anteil von $0,17$ bei $\mathrm{k} = 1$ auf $0,21$ bei $\mathrm{k} = 3$. Für die Schwellenwerte $\mathrm{p} = 0,15$ und $\mathrm{p} = 0,20$ ist der Anstieg noch deutlicher. Erhöht man die Reichweite weiter, ergeben sich keine großen Veränderungen des Anteils der Kerne mehr.

Im Gegensatz zu der zuvor verwendeten Methode der starren Interzeitkantengewichte sollte sich bei der Methode Cosine-Time eine hohe Reichweite nicht negativ auf das Ergebnis auswirken. Verändert sich die Ausrichtung der Repräsentanten $v_\mathrm{x}^{\mathrm{t_i}}$ eines Knotens stark, so werden die Interzeitkantengewichte der Repräsentanten in diesen Bereichen gegen den Wert 0 tendieren. Wir stellen jedoch fest, dass der Umbruch ab Zeitschritt 4 für höhere Reichweiten, auch für Graphen der Cosine-Time-Testreihe, nicht mehr bemerkbar ist. Offensichtlich führt die Erhöhung der Reichweite zu einer stärkeren Glättung der Clusterung, so dass der Umbruch nicht mehr zu erkennen ist.

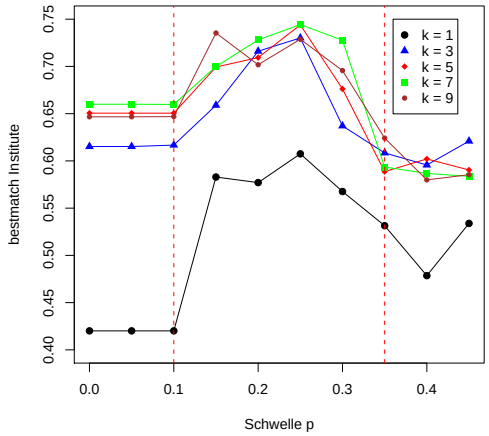
In Abbildung 31 ist der Einfluss der Reichweite k und der Schwelle p auf die Clusterung gut abzulesen. Für niedrige Schwellen ergeben sich kaum Veränderungen. Da die Erhöhung der Schwelle gleichzeitig einen Informationsverlust bedeutet, erscheint für das Greedy-Significance-Clustering



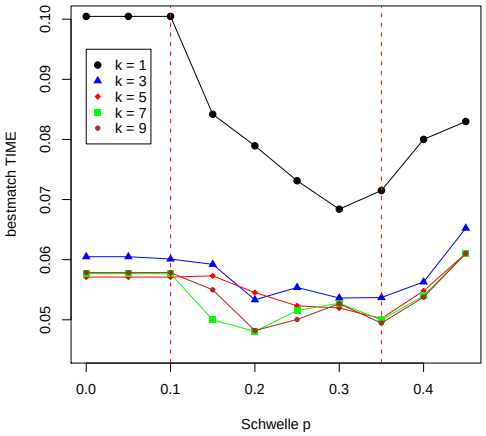
(a) Einfluss der Parameter auf die Coverage der Zeit-Clusterung.



(b) Einfluss der Parameter auf die Dichte des Graphen.



(c) Einfluss der Parameter auf die Ähnlichkeit der Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$.



(d) Einfluss der Parameter auf die Ähnlichkeit der Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$.

Abbildung 31: *Einfluss der Parameter Reichweite* k *und Schwelle* p *auf die Significance-Clusterungen der Cosine-Time-Testreihe.*

die Schwelle $p = 0,15$ die optimale Wahl. Die Zahl der Knoten reduziert sich bei dieser Schwelle auf 4075 von ursprünglich 4677 und der $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung ist für alle Werte der Reichweite k sehr hoch. Hier halten sich der Informationsverlust und die Verbesserung bezüglich der Referenz die Waage. Auffallend ist die deutliche Annäherung an die Referenz-Clusterung für $k = 1$ (siehe dazu Abbildung 31). Eine weitere Erhöhung der Schwelle bringt kaum mehr eine Verbesserung des $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung. Erhöht man die Schwelle über den Wert $0,3$ wird der Informationsverlust durch die Reduzierung der Knoten und Kanten so groß, dass die Ähnlichkeit zur Referenz-Clusterung sogar abnimmt. Ein zeitexpandierter Graph mit Schwelle $0,35$ hat nur ungefähr 2000 Knoten. Die Zahl der Kanten reduziert sich von ursprünglich 23000 auf 4000. Ebenso steigt die Ähnlichkeit zur Zeit-Clusterung für sehr hohe Schwellen leicht an.

Im Graphen mit den Reichweite $k = 1$ ist die Dynamik der Clusterung höher als bei den Graphen mit höherer Reichweite. Nicht alle Cluster verlaufen durchgängig über alle Zeitschritte hinweg, sondern es gibt Cluster, deren Grenze zwischen den verschiedenen Zeitschritten verläuft (siehe Abbildung 34).

6.3.3. Iterative-Conductance-Cutting

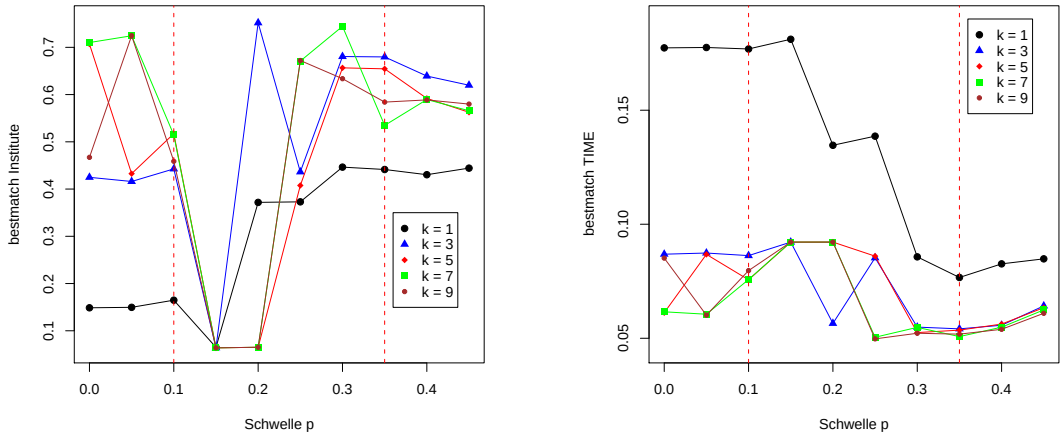
Wir clustern die 50 Graphen mit Hilfe des Iterative-Conductance-Cutting (siehe Abschnitt 2.5.2) zunächst mit den Parametern $a^* = 0,10$, $a^* = 0,07$, $a^* = 0,05$ und $a^* = 0,04$. Beim Vergleich der Durchschnittswerte der einzelnen Durchläufe (siehe Tabelle 12) entscheiden wir uns aufgrund des höchsten durchschnittlichen $\mathrm{bestmatch_{no}}$ bezüglich der Referenz-Clusterung für eine eingehendere Betrachtung der Clusterungen mit dem Parameter $a^* = 0,05$.

Index	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	Cluster	$\mathrm{bestmatch_{no}}$ Institute
$a^* = 0,10$	0,9039	0,7203	0,7992	106,2	0,3712
$a^* = 0,07$	0,9451	0,7157	0,8767	52,4	0,4332
$a^* = 0,05$	0,9626	0,7035	0,9168	32,8	0,4485
$a^* = 0,04$	0,9742	0,6420	0,9328	25,8	0,4191

Tabelle 12: *Durchschnittswerte der Indizes der ICC-Clusterungen der Graphen der Cosine-Time-Testreihe für die verschiedenen Schwellen* a^*.

Auch bei der Clusterung mit dem ICC-Verfahren untersuchen wir den Einfluss der Reichweite und der Schwelle auf die Clusterung (siehe Abbildung 32). Graphen mit der Reichweite $k = 1$ und niedrigen Schwellen liefern Clusterungen mit einer erhöhten Ähnlichkeit zur Zeit-Clusterung. Offensichtlich ist hier die Interzeitdichte für das ICC-Verfahren zu niedrig. Für größere Reichweiten haben die Clusterungen generell eine hohe Ähnlichkeit zur Referenz-Clusterung. Die Erhöhung der Reichweite von $k = 1$ auf $k = 3$ führt zu einer starken Erhöhung des Anteils der Kerne an den Clustern. Das heißt, der Anteil der Knoten, die über die ganze Zeitspanne dem selben Cluster angehören, steigt an und es gibt innerhalb der Cluster weniger Veränderungen der Repräsentantenmengen der einzelnen Zeitschritte. Wie in den Testreihen zuvor führt eine Erhöhung der Schwelle zu einer höheren Ähnlichkeit zur Referenz-Clusterung.

Eine Ausnahme bilden die Graphen mit Schwelle $p = 0,15$ und $p = 0,20$. Bei diesen Graphen unterscheiden sich die Clusterungen sehr stark von der Referenz-Clusterung. Die Abbildung 33



(a) Einfluss der Parameter auf die Ähnlichkeit der Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$.

(b) Einfluss der Parameter auf die Ähnlichkeit der Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$.

Abbildung 32: *Einfluss der Parameter Reichweite* k *und Schwelle* p *auf die ICC-Clusterungen der Cosine-Time-Graphen.*

zeigt uns, dass die geringe Anzahl der Cluster der Auslöser für diese Abweichung ist. Offensichtlich ist der Parameter a^* für diese Graphen zu niedrig gewählt. Eine erneute Clusterung der Graphen mit Schwelle $\mathrm{p} = 0,15$ mit verschiedenen Werten von a^* liefert jedoch keine Verbesserung. Selbst bei $\mathrm{a}^* = 0,70$ haben alle Clusterungen dieser Graphen nur drei Cluster, obwohl sich keine unerwarteten Veränderungen der Graphenstruktur im Vergleich zu den anderen Schwellenwerten ergeben. Es ist dem ICC-Verfahren aufgrund der verwendeten Heuristik (siehe Abschnitt 2.5.2) nicht möglich, einen weiteren Schnitt innerhalb der Cluster zu finden, mit einer heuristisch gefundenen Conductance, die niedriger ist als a^*.

6.4. Fazit

Bei unserer Testreihe mit der Methode Cosine-Time liefert das Greedy-Significance-Clustering die konstantesten Ergebnisse. Bis auf die Clusterungen der Graphen mit den Schwellen $\mathrm{p} = 0,15$ und $\mathrm{p} = 0,20$ erreicht das ICC-Clustering Ergebnisse mit einer hohen Ähnlichkeit zur Referenz-Clusterung. Die Vorteile der Methode Cosine-Time gegenüber der Methode Alpha, der starren Interzeitkantengewichte, sind der mögliche Vergleich der verschiedenen Interzeitkanten anhand der Gewichte und die dadurch erhöhte Aussagekraft unseres zeitexpandierten Graphen. Hat die Interzeitkante zwischen $v_{\mathrm{x}}^{\mathrm{t_i}}$ und $v_{\mathrm{x}}^{\mathrm{t_j}}$ ein hohes Gewicht, so besteht in unserer Anwendung eine hohe Ähnlichkeit des E-Mail-Verkehrs des Knotens v_{x} in den Zeitschritten $\mathrm{t_i}$ und $\mathrm{t_j}$. Durch diese dynamischen Gewichte wird es leichter möglich, Veränderungen innerhalb des Graphen durch Cluster-Verfahren sichtbar zu machen. Ein Vergleich mit Clusterungen von Graphen mit starren Interzeitkanten bestätigte die Vermutung, dass durch die Methode Cosine-Time eine bessere Erfassung der Dynamik des zeitexpandierten Graphen möglich ist. Der Nachteil der Methode Cosine-Time ist der erhöhte Aufwand der Berechnung der Interzeitkantengewichte.

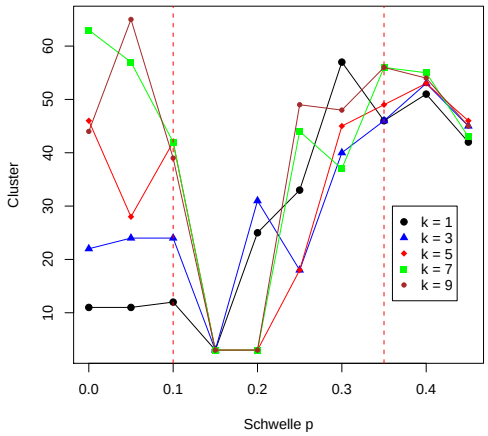
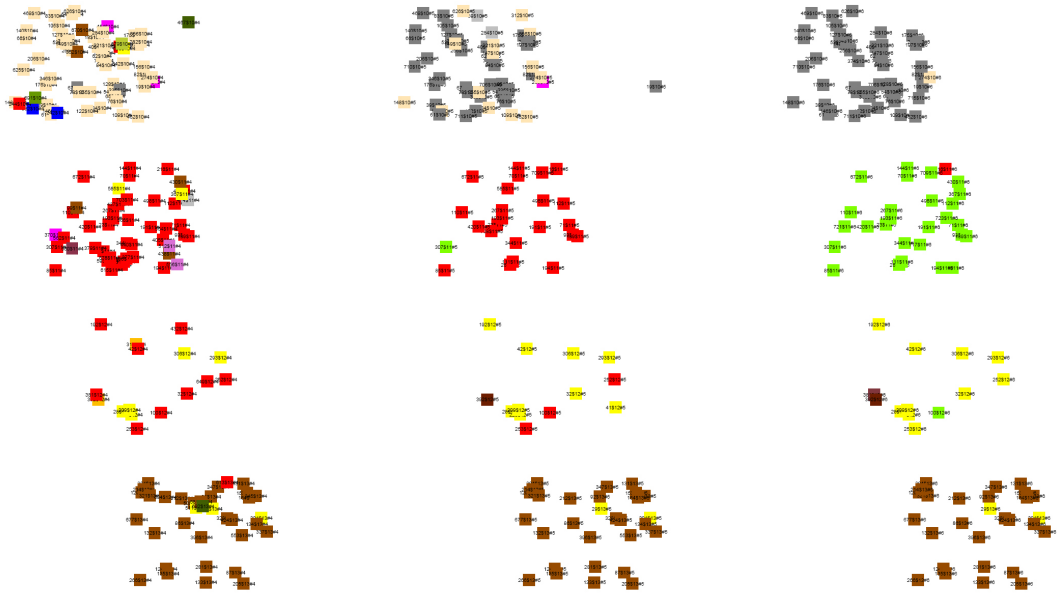


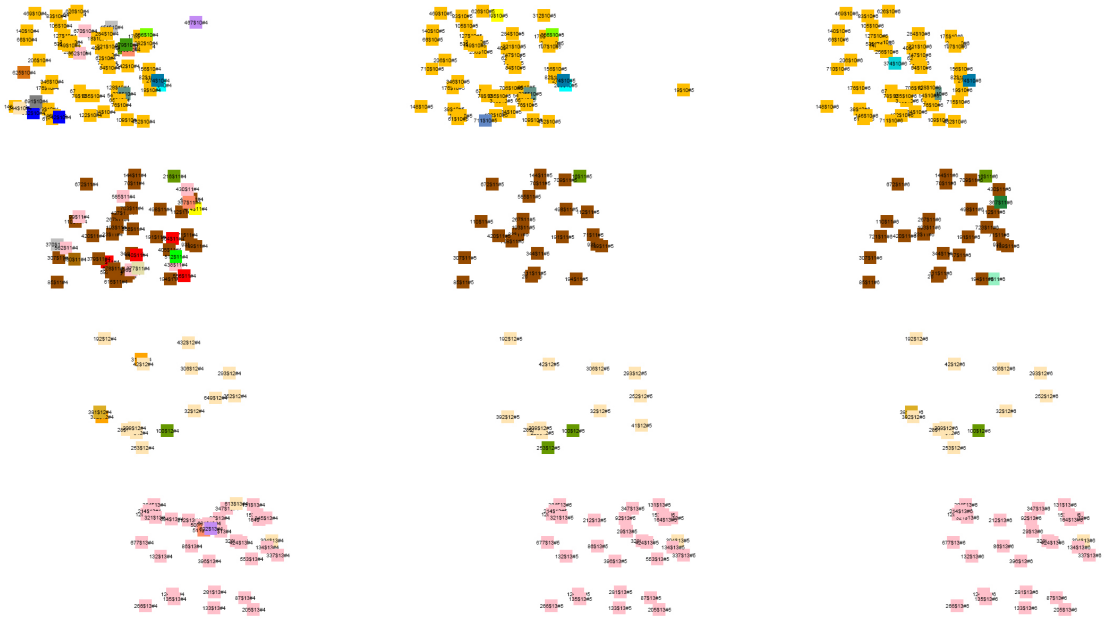
Abbildung 33: *Auswirkungen der Anzahl der Cluster auf die Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand des* $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$. *Vergleiche hierzu Abbildung 32.*

Bei der Verwendung der Methode Cosine-Time reduzieren sich die variablen Parameter auf zwei. Dadurch vereinfacht sich die Handhabung unseres Modells. Bei der Schwelle des Graphen gilt wie bei den ersten beiden Testreihen, dass sich eine kleine Schwelle positiv auf das Erkennen der Referenz-Clusterung durch das Cluster-Verfahren auswirken kann. Erhöht man die Schwelle zu stark, gehen zu viele Informationen bzw. Knoten und Kanten verloren und die Aussagekraft des Graphen nimmt ab. Manche Zusammenhänge können aufgrund der fehlenden Kanten nicht mehr erkannt werden.

Je höher wir die Reichweite wählen, desto stabiler wird die Clusterung. In Abbildung 34 sieht man den gleichen Ausschnitt der Clusterung zweier zeitexpandierter Graphen unserer Testreihe. Zum einen die Clusterung $\mathcal{C}'$ des zeitexpandierten Graphen mit $\mathrm{k} = 1$ und $\mathrm{p} = 0$ mit Hilfe des Significance-Verfahrens (siehe Abbildung 34a). Visuell ist eine deutliche Ähnlichkeit zur Referenz-Clusterung $\mathcal{C}$ zu erkennen, der $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ beträgt $0,42$. Durch die minimale Reichweite $\mathrm{k} = 1$ werden die Cluster häufig an den Übergängen der verschiedenen Zeitschritte getrennt. Nur der braune Cluster erfasst einen Lehrstuhl durchgängig. Bei der Clusterung des zeitexpandierten Graphen mit $\mathrm{k} = 3$ und $\mathrm{p} = 0$ (siehe Abbildung 34b) verschwinden diese Trennungen und es entstehen zu den Lehrstühlen sehr ähnliche Cluster. Für jeden Lehrstuhl gibt es einen Cluster, der durchgängig die deutliche Mehrheit aller E-Mail-Accounts des Lehrstuhls enthält. Die Repräsentanten der einzelnen Accounts sind für höhere Reichweiten stärker verknüpft, sofern sich ihr Verhalten nicht stark verändert und damit die Interzeitkantengewichte sehr klein werden. Durch diese stärkere Verknüpfung bilden die Repräsentanten eines Knotens eine kompakte Gruppe. Die einzelnen Gruppen, bestehend aus den Repräsentanten der Knoten, werden von den Cluster-Verfahren nur dann dem selben Cluster zugewiesen, wenn sie über einen längeren Zeitraum eng miteinander verbunden sind. Dadurch werden Veränderungen, die nur eine kurze Zeitspanne anhalten, weniger stark berücksichtigt. Das entspricht einer Glättung der Clusterung bezüglich der typischen Ausprägung des dynamischen Graphen. Will man kurzfristige Änderungen erfassen, darf die Reichweite daher nicht zu hoch gewählt werden. Allerdings wirkt sich eine höhere Reichweite bei unserer Anwendung positiv auf das Erkennen der Lehrstuhl-Zugehörigkeit aus.



(a) Ausschnitt der Significance-Clusterung des zeitexpandierten Graphen mit $k = 1$ und $p = 0$. Die verschiedenen Farben der Knoten stehen für die verschiedenen Cluster.



(b) Ausschnitt der Significance-Clusterung des zeitexpandierten Graphen mit $k = 3$ und $p = 0$. Die verschiedenen Farben der Knoten stehen für die verschiedenen Cluster.

Abbildung 34: *In der oberen Abbildung sehen wir einen Ausschnitt der Significance-Clusterung eines zeitexpandierten Graphen mit Reichweite* $k = 1$. *Zwischen den einzelnen Zeitschritten bestehen deutliche Unterschiede. Bei dem identischen Ausschnitt der Clusterung des zeitexpandierten Graphen mit Reichweite* $k = 3$ *zeigt sich ein deutlich konformeres Bild der einzelnen Zeitschritte zueinander und es kommt zu weniger Trennungen der Cluster entlang der Zeitgrenzen.*

7. Bewertung der Ergebnisse

Das Netzwerk, auf dem wir unsere Testreihen durchführten, ist das E-Mail-Netzwerk der *Fakultät für Informatik* an der *Universität Karlsruhe (TH)*. Die *Fakultät für Informatik* besteht aus verschiedenen Instituten und zentralen Einrichtungen, wie der *Abteilung Technische Infrastruktur*, kurz *ATIS* oder dem Dekanat. Dabei setzen sich die Institute aus mehreren Lehrstühlen zusammen. Anhand der anonymen Identifier lassen sich die einzelnen E-Mails-Accounts und Lehrstühle unterscheiden. Wir verwendeten die anonymen E-Mail-Kontaktdaten eines Zeitraumes von 308 Tagen zur Erzeugung unserer zeitexpandierten Graphen, wobei wir den Zeitraum in elf Zeitschritte zu je 28 Tagen aufteilten. Dabei wurden nur die E-Mails erfasst, die innerhalb der Fakultät verschickt wurden. Mit Hilfe der in Abschnitt 3.1 beschriebenen Sender-Empfänger-Paare erzeugten wir für jeden Zeitschritt t eine symmetrische Matrix $\mathcal{A}(\mathrm{t})$. Der Eintrag $\mathcal{A}(\mathrm{t})_{\mathrm{i,j}}$ entspricht der Anzahl der in Zeitschritt t ausgetauschten E-Mails zwischen Account i und Account j. Die Zugehörigkeit der Accounts zu den einzelnen Lehrstühlen nannten wir Institute-Clusterung und verwendeten sie als Referenz-Clusterung.

Zur Verfügung gestellt wurden uns diese Daten durch die ATIS. Die ATIS ist eine Abteilung der Fakultät für Informatik. Sie ist innerhalb der Fakultät für die Infrastruktur der Informations- und Kommunikationstechnik zuständig. Bedanken möchten wir uns in diesem Zusammenhang bei den Mitarbeitern der ATIS, im Speziellen bei Herrn Hopp, die uns die anonymisierten Daten des E-Mail-Netzwerkes innerhalb der Informatik-Fakultät zur Verfügung gestellt haben. Herr Hopp konnte uns auf einige konkrete Anfragen zusätzliche Informationen über die Zusammenhänge zwischen den einzelnen Lehrstühlen liefern, ohne die Anonymität zu gefährden. Diese zusätzlichen Informationen ermöglichten uns eine bessere Interpretation der gefundenen Clusterungen.

In den verschiedenen Testreihen erzeugten wir aus den Matrizen $\mathcal{A}(\mathrm{t})$ mit Hilfe der in Abschnitt 3.2.1 vorgestellten Methoden die zeitexpandierten Graphen. In diesem Kapitel wollen wir die Ergebnisse der Testreihen zusammenfassen. Dazu werden wir zunächst die Ergebnisse unserer Clusterungen interpretieren. Anschließend werden wir untersuchen, wie stark sich die Clusterung der einzelnen Zeitschritte von der Clusterung des gesamten zeitexpandierten Graphen unterscheidet. Zuletzt werden wir noch einmal den Einfluss der Parameter erläutern und die Vor- und Nachteile der Methoden unseres Modells diskutieren.

7.1. Interpretation der Ergebnisse

Die bei der Mehrzahl der erzeugten Graphen hohen Ähnlichkeiten der Clusterungen mit der durch die Lehrstühle vorgegebenen Referenz-Clusterung zeigen, dass das E-Mail-Verhalten in unserem Netzwerk stark von der Zugehörigkeit zu den verschiedenen Lehrstühlen bestimmt wird. Dabei gibt es einige Auffälligkeiten. Die Clusterungen fassen in vielen Graphen der verschiedenen Testreihen bestimmte Referenz-Cluster zusammen. In unseren Abbildungen 51 und 52 im Anhang sind diese häufig vorkommenden Clusterungen der Lehrstühle farblich hervorgehoben. Die Bezeichnung der Knoten in den Abbildungen setzt sich folgendermaßen zusammen: *knoten$lehrstuhl#time*. Die Nummer *lehrstuhl* bestimmt die Zugehörigkeit der Knoten zu den verschiedenen Referenz-Clustern. Es gibt 26 verschiedene Referenz-Cluster, die durch die Zahlen von 0 bis 25 repräsentiert werden. Dabei steht jeder dieser Cluster für einen Lehrstuhl oder eine zentrale Einrichtung.

Bei vielen Graphen der Testreihen werden die Referenz-Cluster 5 und 8 in einem Cluster vereinigt, teilweise ist der Referenz-Cluster 3 ebenfalls in diesen Clustern enthalten. Dabei ist Referenz-Cluster 8 ein kleinerer Lehrstuhl, der organisatorisch mit Referenz-Cluster 5 zusammengehört. Referenz-Cluster 3 und 5 sind Lehrstühle, die vor nicht allzu langer Zeit zusammen gehörten. Diese zusätzlichen Informationen bestätigen, dass eine Clusterung dieser Lehrstühle aufgrund der bestehenden sozialen Kontakte und der engen organisatorischen Zusammengehörigkeit sinnvoll ist. Interessant wäre hierbei die Betrachtung eines längeren Zeitraumes. Sollten sich die Kontakte zwischen den Referenz-Clustern 3 und 5 abschwächen, könnte eine Clusterung eines entsprechenden zeitexpandierten Graphen diese Entwicklung bestätigen.

Eine weitere häufig als Cluster vorkommende Kombination der Referenz-Cluster sind die Referenz-Cluster 6 und 9. Hierbei handelt es sich um zwei Lehrstühle innerhalb des selben Instituts. Teilweise werden die beiden Referenz-Cluster mit Referenz-Cluster 11 im gleichen Cluster zusammengefasst. Eine Erklärung hierfür liefert die Tatsache, dass Referenz-Cluster 6 und Referenz-Cluster 11 früher gemeinsam ein Institut bildeten. Weitere, immer wiederkehrende Kombinationen sind die Referenz-Cluster 10 und 21, sowie 12 und 17. Dabei sind die Referenz-Cluster 10 und 21 Lehrstühle des selben Instituts, die räumlich nah beieinander liegen. Bei Referenz-Cluster 12 handelt es sich um eine zentrale Einrichtung, die sich personell mit Referenz-Cluster 17 überschneidet. Auch Referenz-Cluster 4 ist eine zentrale Einrichtung. Die Informationen darüber, welche der Referenz-Cluster zentralen Einrichtungen entsprechen, helfen uns die Ergebnisse besser zu verstehen. Die Referenz-Cluster 4 und 12 werden häufig solchen Clustern zugeordnet, die bereits mehrere Lehrstühle enthalten. So gibt es einige Cluster, die außer den Referenz-Clustern 6, 9 und 11 zusätzlich den Referenz-Cluster 4 enthalten. Die zentralen Einrichtungen kommunizieren viel mit Knoten außerhalb der Einrichtung. Daher verwundert es nicht, dass sie hauptsächlich großen Clustern zugeordnet werden. Dieser Zusammenhang zeigt sich in Abbildung 52. Der große gelbe Cluster in dieser Abbildung beinhaltet beide zentralen Einrichtungen.

Die zweite Auffälligkeit der Clusterungen der zeitexpandierten Graphen lässt sich ebenfalls an dieser Abbildung erkennen. Viele Clusterungen der Graphen mit niedriger Reichweite haben auffällig viele Clustergrenzen zwischen den Zeitschritten 4 und 5. So enden viele Cluster, die über mehrere Zeitschritte hinwegreichen, zwischen diesen Zeitschritten. Wir beobachteten diesen Sachverhalt bereits bei dem Vergleich der Clusterungen von Graphen mit statischen und dynamischen Interzeitkantengewichten (siehe Abschnitt 6.3.2).

Um eine Begründung für diese Ergebnisse zu finden, untersuchen wir die Teilgraphen der einzelnen Zeitschritte. In den Zeitschritten 3 und 4 ist die Anzahl der Knoten und der Kanten zwischen den verschiedenen Lehrstühlen drastisch erhöht (siehe rote Makierung in Tabelle 16). Eine Erklärung hierfür ist uns leider nicht möglich. Die Erhöhung der Kanten- und Knotenzahl beschränkt sich nicht auf wenige Lehrstühle, sondern ist bei allen größeren Lehrstühlen festzustellen. Es ist keinerlei Systematik innerhalb der Anstiege zu erkennen. Eine mögliche Begründung ist ein Anstieg von SPAM-E-Mails, welcher in diesem Zeitraum auftrat, obwohl nur interne E-Mails erfasst werden. Die zeitliche Lage des Zeitschrittes 3 um den Jahreswechsel ist eine weitere mögliche Begründung für den Anstieg. Die verschiedenen Personen verschickten eventuell an alle Accounts in ihrem Adressbuch Grußkarten zu Weihnachten und Neujahr. Allerdings ist dies keine Erklärung für die ebenfalls erhöhten Werte in Zeitschritt 4. Die Güte der Referenz-Clusterung ist in diesen beiden Zeitschritten deutlich geringer, als in den anderen Zeitschritten. Zwar können wir nicht genau erklären, welche Ursache dieser Anstieg hat, jedoch erklärt er, warum so viele

Zeitschritt	Knoten	Intra	Inter	Coverage Institute
0	393	1101	447	0, 75
1	444	1259	603	0, 74
2	456	1238	645	0, 72
3	589	1257	1430	0, 53
4	513	1325	971	0, 65
5	379	1147	497	0, 75
6	384	1069	444	0, 74
7	380	1013	399	0, 78
8	383	1102	422	0, 78
9	376	1109	401	0, 79
10	380	1142	471	0, 77

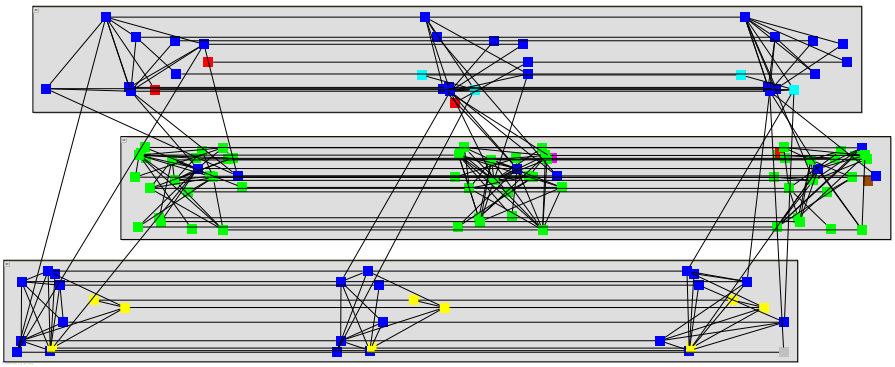
Tabelle 13: *Vergleich des E-Mail-Netzwerks in den verschiedenen Zeitschritten anhand der in die einzelnen Zeitschritte zerschnittenen Graphen der Cosine-Time-Testreihe. Für jeden Zeitschritt werden die Anzahl der Knoten, die Anzahl der Kanten innerhalb der Lehrstühle und die Anzahl der Kanten zwischen den verschiedenen Lehrstühlen, sowie die gewichtete Coverage der Referenz-Clusterung angegeben. Die Daten ergeben sich aus den Graphen mit Schwelle* $\mathrm{p} = 0$.

Cluster zwischen den Zeitschritten 4 und 5 enden. Viele Knoten und Kanten, die zuvor existierten, verschwinden in Zeitschritt 5. Die erhöhten Knoten- und Kantenzahlen in Zeitschritt 3 und 4 führen dazu, dass am Übergang zu Zeitschritt 5 im Vergleich zu den anderen Übergängen pro Intrazeitkante weniger Interzeitkanten existieren. Deshalb stellen die Übergänge für die Clusterung günstige Schnitte dar. Aufgrund der in Zeitschritt 1 und 2 bereits angestiegenen Knoten- und Kantenzahlen findet am Übergang von Zeitschritt 2 zu Zeitschritt 3 nichts Vergleichbares statt. Betrachten wir noch einmal Abbildung 52. In den Zeitschritten 0 bis 4 wird eine genaue Abgrenzung der einzelnen Gruppen durch die erhöhte Kommunikation zwischen den verschiedenen Lehrstühlen schwieriger. Das führt zu dem gelben Cluster, der die Lehrstühle 6, 9 und 11 sowie die zentralen Einrichtungen 4 und 12 zusammenfasst. In den späteren Zeitschritten normalisiert sich die Situation wieder. Der entsprechende blaue Cluster enthält nur die Referenz-Cluster 6, 9 und 11, ohne die zentralen Einrichtungen. Ähnliches ist für den grünen Cluster feststellbar. Er beinhaltet über alle Zeitschritte hinweg die Referenz-Cluster 5 und 8. Aber erst ab dem 5. Zeitschritt, also nach der Normalisierung der Knoten- und Kantenzahlen wird Referenz-Cluster 3 dem Cluster hinzugefügt. Die Clusterung des zeitexpandierten Graphen zeigt hier deutlich, dass man diese Veränderungen innerhalb des Graphen erfassen kann. Die hier diskutierte Anomalie und die gemeinsame Clusterung bestimmter Cluster der Referenz-Clusterung beschränken sich nicht auf eine einzelne Testreihen, sondern sind in allen Testreihen zu beobachten.Die Ergebnisse der Cosine-Time-Testreihe belegen die Veränderungen innerhalb des Graphen am deutlichsten. Für die Testreihen mit starren Interzeitkantengewichten ist zum Erkennen der Veränderungen in den Zeitschritten 3 und 4 durch die Clusterung eine geeignete Wahl von Interzeitkantengewicht α und Schwelle p entscheidend.

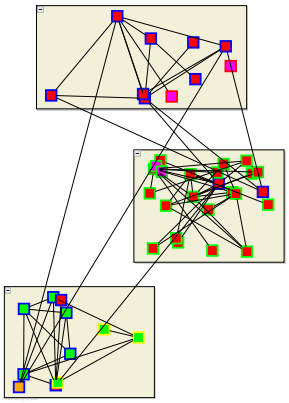
Mit der Clusterung von zeitexpandierten Graphen versuchen wir die Veränderungen der Gruppierungen über die Zeit zu erfassen. Bei den bisherigen Verfahren werden die Ausprägungen der einzelnen Zeitschritte zunächst einzeln geclustert und erst danach miteinander verknüpft. Im Folgenden werden wir die Unterschiede zwischen dieser zeitlich flachen und der zeitexpandierten Clusterung bei unserem E-Mail-Netzwerk untersuchen.

7.2. Vergleich der zeitlich flachen Clusterungen mit den Clusterungen der zeitexpandierten Graphen

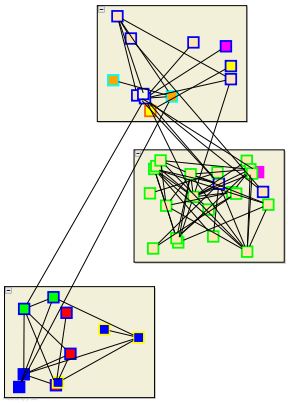
Wir vergleichen die Ergebnisse der zeitlich flachen und der zeitexpandierten Clusterungen anhand der Significance-Clusterung der zeitexpandierten Graphen der Cosine-Time-Testreihe, da diese die konstantesten Ergebnisse liefert. Dazu nehmen wir die 50 Graphen der Cosine-Time-Testreihe und zerschneiden jeden dieser Graphen in die elf Zeitschritte, für die wir die Significance-Clusterungen der zeitexpandierten Graphen speicherten. Dadurch erhalten wir 550 Graphen, von denen jeder einen Zeitschritt eines der zeitexpandierten Graphen repäsentiert. Im Folgenden wird die gespeicherte Clusterung der zeitexpandierten Graphen als *expandierte Clusterung* bezeichnet. Jeder dieser 550 Graphen wird anschließend mit dem Significance-Verfahren geclustert. Diese Clusterungen der einzelnen Zeitschritte wollen wir mit den expandierten Clusterungen der 550 Graphen vergleichen.



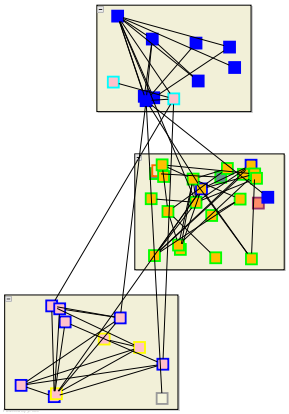
(a) Ausschnitt eines zeitexpandierten Graphen, der sich über die drei Zeitschritte 5 bis 7 und drei Lehrstühle ausdehnt. Dabei stehen die Farben der Knoten für die verschiedenen Cluster der Significance-Clusterung des zeitexpandierten Graphen.



(b) Zeitschritt 5 des Ausschnittes des Graphen.



(c) Zeitschritt 6 des Ausschnittes des Graphen.



(d) Zeitschritt 7 des Ausschnittes des Graphen.

Abbildung 35: *Illustration der Zerschneidung der zeitexpandierten Graphen. Bei den Abbildungen (b)-(d) repräsentieren die verschiedenen Farben der Knotenränder die verschiedenen Cluster der Significance-Clusterung des gesamten zeitexpandierten Graphen (vergleiche dazu 35a), während die Farben der Knoten für die Cluster der Significance-Clusterung der einzelnen Teilgraphen stehen.*

Illustriert wird unser Vorgehen durch Abbildung 35. Durch das Zerschneiden des zeitexpandierten Graphen 35a in die einzelnen Zeitschritte (35b, 35c und 35d) erhalten wir für jeden zeitexpandierten Graphen elf Teilgraphen. Diese Teilgraphen enthalten die Informationen der Clusterung des gesamten zeitexpandierten Graphen (in der Abbildung an den Farben der Knotenränder nachvollziehbar). Die Clusterung der einzelnen Zeitschritte wird in den Abbildungen 35b, 35c und 35d durch die verschiedenen Knotenfarben dargestellt. Wir bezeichnen die Significance-Clusterung der einzelnen Zeitschritte im Folgenden als *zeitlich flache Clusterung*.

Reichweite k	1	3	5	7	9
$\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C},\mathcal{C}')$	0,6972	0,6638	0,6304	0,6044	0,6002
$\mathcal{NVD}(\mathcal{C},\mathcal{C}')$	0,1618	0,1657	0,1823	0,1912	0,1932
$\mathcal{NVI}(\mathcal{C},\mathcal{C}')$	0,1318	0,1274	0,1354	0,1405	0,1405

Tabelle 14: *Durchschnittswerte einiger Vergleichsmaße der expandierten und zeitlich flachen Clusterungen der Teilgraphen. Dabei ist der* $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C},\mathcal{C}')$ *die Ähnlichkeit der zeitlich flachen Clusterung* $\mathcal{C}'$ *bezüglich der expandierten Clusterung* $\mathcal{C}$*. Van Dongen* $\mathcal{NVD}$ *und die Variation of Information* $\mathcal{NVI}$ *sind hingegen symmetrische Distanzmaße (siehe Abschnitt 2.8). Die Ähnlichkeit der expandierten und der zeitlich flachen Clusterung nimmt mit Erhöhung der Reichweite ab.*

Die durchschnittliche Clusterzahl $29,23$ der expandierten Clusterungen ist höher, als die $18,81$ der zeitlich flachen Significance-Clusterungen. Dieser Sachverhalt könnte auf die in [FB07, BDG+08] beschriebene Eigenschaft der Modularity zurückzuführen sein. Die Granularität einer Clusterung durch die Optimierung der Modularity ist demnach abhängig von der Kanten- und Knotenzahl des Graphen. Daher kommt der Unterschied zwischen der Anzahl der Cluster bei beiden Clusterungen. Die Clusterung des gesamten zeitexpandierten Graphen enthält aufgrund der gestiegenen Kanten- und Knotenzahl eine höhere Clusterzahl als die Clusterung eines einzelnen Zeitschrittes. Da die meisten Cluster über die Zeit hinweg verlaufen, haben die expandierten Significance-Clusterungen der einzelnen Zeitschritte mehr Cluster als die zeitlich flachen Clusterungen. Dennoch besteht im Schnitt eine hohe Ähnlichkeit der beiden Clusterungen, wie die Vergleichsmaße in Tabelle 14 und Tabelle 23 im Anhang belegen.

Reichweite k	1	3	5	7	9
$\mathrm{cov}_{\mathrm{w}}$	0,8691	0,8477	0,8244	0,8147	0,8142
$\mathrm{mod}_{\mathrm{w}}$	0,7552	0,7527	0,7364	0,7321	0,7307
δ_{d}	0,7642	0,6541	0,6101	0,6005	0,5819

Tabelle 15: *Durchschnittliche Indexwerte der in die einzelnen Zeitschritte zerschnittenen Significance-Clusterungen der zeitexpandierten Graphen. Dabei berechnen wir die Durchschnittswerte aller Teilgraphen mit derselben Reichweite. Die Ergebnisse stehen daher für jeweils 110 Teilgraphen. Die Güte der expandierten Clusterungen nimmt mit steigender Reichweite ab.*

Für beide im Graphen enthaltenen Clusterungen werden die Durchschnittswerte für die gewichtete Coverage, die gewichtete Modularity und die durchschnittliche Interclusterconductance berechnet. Die Ergebnisse der expandierten Clusterungen sind in Tabelle 15 zu finden. Wir berechnen die Durchschnittswerte in Abhängigkeit der Reichweite. Es fällt auf, dass die Werte, bis auf die gewichtete Modularity, für die Reichweite $\mathrm{k} = 1$ deutlich besser ausfallen, als für die anderen Reichweiten. Die gewichtete Modularity bleibt für alle Werte der Reichweite relativ konstant.

Die zeitlich flache Clusterung der Teilgraphen ergibt die Durchschnittswerte $0,8939$ für die gewichtete Coverage, $0,7844$ für die gewichtete Modularity und $0,8869$ für die durchschnittliche Interclusterconductance. Wir stellen fest, dass die Werte der expandierten Clusterungen für die gewichtete Coverage und durchschnittliche Interclusterconductance deutlich schlechter sind wie die Werte der zeitlich flachen Clusterungen der Teilgraphen. Die Verschlechterung der Coverage ist nicht verwunderlich, da die durchschnittliche Anzahl der Cluster der expandierten Clusterungen deutlich höher ist als die der zeitlich flachen Clusterungen. Dahingegen sind die Unterschiede für die Modularity gering. Dies ist überraschend, da das hierarchische Greedy-Significance-Clustering die Clusterung über die Maximierung der Modularity in den einzelnen Teilschritten findet. Offensichtlich ist die Güte der expandierten Clusterung der Graphen bezüglich der Modularity nahe an der Güte der zeitlich flachen Clusterung.

Je höher die Reichweite des zeitexpandierten Graphen ist, desto niedriger ist die Coverage und die Interclusterconductance der expandierten Clusterung (siehe Tabelle 15). Hier wird deutlich, dass eine Erhöhung der Reichweite mit einem höheren Unterschied zwischen den beiden Clusterungen einhergeht (siehe Tabelle 14). Beim $\mathrm{bestmatch_{no}}$ lässt sich dieser Zusammenhang besonders gut erkennen. Die Ähnlichkeit nimmt bis zu einem gewissen Grad ab, ab dem sich eine weitere Erhöhung der Reichweite nicht mehr auf die Ähnlichkeit auswirkt (hier bei der Erhöhung der Reichweite von $\mathrm{k} = 7$ auf $\mathrm{k} = 9$). Bereits zuvor sprachen wir an (siehe Abschnitt 6.4), dass eine Erhöhung der Reichweite auch eine Glättung der Clusterung bedeutet. Denn das durchschnittliche Verhalten des Knotens gewinnt durch die starke Konnektivität der Repräsentanten eines Knotens mehr an Bedeutung als das zeitlich lokale Verhalten des Knotens. Durch diese Mittelbildung entstehen für hohe Werte der Reichweite bei deren weiterer Zunahme nur noch geringfügige Veränderungen der Clusterungen, falls sich das Verhalten der Knoten über die Zeit nicht stark verändert.

Die Schwelle führt zu einem Anstieg der Ähnlichkeit der expandierten Clusterungen zur Referenz-Clusterung. Gleiches gilt für die zeitlich flachen Clusterungen. Die logische Folge ist eine Erhöhung der Ähnlichkeit der beiden Clusterungen zueinander (siehe Tabelle 23).

Aufgrund unserer Beobachtungen, dass die einzelnen Zeitschritte teilweise sehr hohe Unterschiede in der Knoten- und Kantenanzahl haben, vergleichen wir für jeden Zeitschritt die beiden Clusterungen anhand einiger Vergleichsmaße (siehe Tabelle 16). In den Zeitschritten 3 und 4, die die stärkste Abweichung gegenüber den anderen Zeitschritten aufweisen, ist die Ähnlichkeit der beiden Clusterungen am geringsten. Bei den expandierten Clusterungen werden die Nachbarzeitschritte mit in die Betrachtung einbezogen. Es wird ein Kontext zwischen den verschiedenen Zeitschritten hergestellt, während das Significance-Clustering der einzelnen Zeitschritte versucht, das lokale Optimum zu finden. Durch die starke Abweichung in den Zeitschritten 3 und 4 fällt hier der Unterschied zwischen der Betrachtung des ganzen Kontextes und der lokalen Teilgraphen am stärksten aus. Eine weitere Auffälligkeit ist die leichte Zunahme der Ähnlichkeit der Clusterungen an den zeitlichen Enden (hier Zeitschritte 9 und 10). Dies führen wir auf die geringere Anzahl der Interzeitkanten in diesen Bereichen zurück. Durch die starken Unterschiede in den frühen Zeitschritten ist für die Zeitschritte 0 und 1 dieser Zusammenhang nicht feststellbar.

Die expandierten Clusterungen und die Clusterungen der einzelnen Teilgraphen weisen viele Gemeinsamkeiten auf. Dennoch unterscheiden sie sich. Die expandierte Clusterung verbindet die Informationen der einzelnen Zeitschritte zu einem Gesamtbild. Im Unterschied zur lokalen Optimierung der zeitlich flachen Clusterungen wird hier eine Clusterung gesucht, die über die gesamte Zeitspanne hinweg ein im Hinblick auf die Bewertungsfunktion des Cluster-Verfahrens gutes

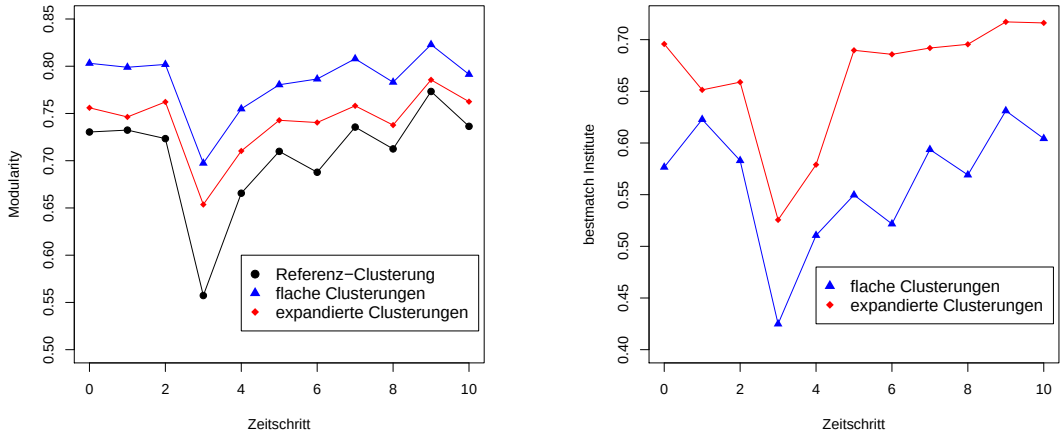
Zeitschritt	$\mathcal{NVD}(\mathcal{C}, \mathcal{C}')$	$\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$	$\mathcal{NVI}(\mathcal{C}, \mathcal{C}')$
0	$0,1832$	$0,6347$	$0,1347$
1	$0,1688$	$0,6532$	$0,1293$
2	$0,1668$	$0,6610$	$0,1259$
3	$0,2383$	$0,5532$	$0,1808$
4	$0,1924$	$0,6098$	$0,1476$
5	$0,1862$	$0,6217$	$0,1403$
6	$0,1855$	$0,6136$	$0,1420$
7	$0,1761$	$0,6578$	$0,1315$
8	$0,1738$	$0,6507$	$0,1306$
9	$0,1511$	$0,6908$	$0,1113$
10	$0,1447$	$0,6847$	$0,1125$

Tabelle 16: *Vergleichsmaße für die expandierten Clusterungen und für die zeitlich flachen Clusterungen der Teilgraphen in Abhängigkeit der Zeitschritte. Dabei ist der* $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ *die Ähnlichkeit der zeitlich flachen Clusterung* $\mathcal{C}'$ *bezüglich der expandierten Clusterung* $\mathcal{C}$*. Die Maße van Dongen* $\mathcal{NVD}$ *und Variation of Information* $\mathcal{NVI}$ *sind hingegen symmetrische Distanzmaße.*

Ergebnis liefert. Dabei gilt, je höher die Reichweite, desto stärker die Glättung bezüglich des durchschnittlichen Verhaltens des Knotens. Wollen wir kurzzeitige und kleinere Veränderungen erfassen, muss die Reichweite möglichst klein gewählt werden. Suchen wir nach einer möglichst guten Repräsentation des durchschnittlichen Verhaltens, muss die Reichweite höher gewählt werden. Für die Reichweite $\mathrm{k} = 1$ weist die Clusterung des zeitexpandierten Graphen die höchste Ähnlichkeit zur zeitlich flachen Clusterung auf.

Es ist zu beachten, dass bei der Verwendung der Methode Cosine-Time eine starke Veränderung des Verhaltens der Repräsentanten des Knotens zu sehr kleinen Interzeitkantengewichten führt. Das heißt, je stärker sich das Verhalten der Repräsentanten zwischen den verschiedenen Zeitschritten ändert, desto schwächer ist der Einfluss aufeinander. Dadurch lassen sich Veränderungen des dynamischen Graphen mit Hilfe der Methode Cosine-Time besser erkennen. Generell ist zu beobachten, dass starke Veränderungen innerhalb des dynamischen Graphen zu größeren Unterschieden zwischen der zeitlich flachen Clusterung und der Clusterung des zeitexpandierten Graphen führen.

In unserem E-Mail-Netzwerk zeigt sich, dass die expandierte Clusterung der einzelnen Zeitschritte eine deutlich höhere Ähnlichkeit zur Referenz-Clusterung aufweist, als die zeitlich flache Clusterung (siehe Abbildung 36b). Der durchschnittliche $\mathrm{bestmatch}_{\mathrm{no}}$ der zeitlich flachen Clusterungen aller 550 Teilgraphen bezüglich der Referenz-Clusterung ergibt $0,56$ gegenüber einem deutlich höheren durchschnittlichen $\mathrm{bestmatch}_{\mathrm{no}}$ von $0,66$ der expandierten Clusterungen. In allen Zeitschritten bis auf Zeitschritte 3 und 4 ist die Ähnlichkeit der beiden Clusterungen zur Referenz-Clusterung relativ konstant. Für die Zeitschritte 3 und 4 führt die starke Erhöhung der Knoten- und Kantenzahl zu einem starken Rückgang der Ähnlichkeit beider Clusterungen zur Referenz-Clusterung. Der $\mathrm{bestmatch}_{\mathrm{no}}$ der expandierten Clusterung bezüglich der Referenz-Clusterung fällt von $0,66$ in Zeitschritt 2 auf $0,52$ in Zeitschritt 3. Bei der zeitlich flachen Clusterung fällt der $\mathrm{bestmatch}_{\mathrm{no}}$ bezüglich der Referenz-Clusterung von $0,58$ auf $0,42$. Generell fallen die Bewertungen der Clusterungen in Zeitschritt 3 deutlich schlechter aus (siehe



(a) Vergleich der Durchschnittswerte der gewichteten Modularity für die verschiedenen Zeitschritte.

(b) Vergleich der Durchschnittswerte des $\mathrm{bestmatch}_{\mathrm{no}}$ bezüglich der Referenz-Clusterung für die verschiedenen Zeitschritte.

Abbildung 36: *Vergleich der expandierten Clusterungen und der zeitlich flachen Clusterungen für die verschiedenen Zeitschritte. Dabei wird in Abbildung 36a die gewichtete Modularity der expandierten Clusterungen, der zeitlich flachen Clusterungen und der Referenz-Clusterung verglichen. In Abbildung 36b wird die Ähnlichkeit der expandierten und der zeitlich flachen Clusterungen zur Referenz-Clusterung dargestellt.*

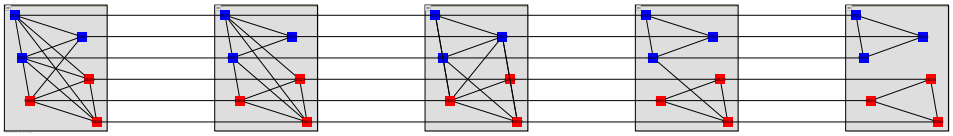
Abbildung 36a). Offensichtlich erschwert die starke Erhöhung der Kanten zwischen den verschiedenen Lehrstühlen das Finden einer signifikanten Clusterung. Die höhere Ähnlichkeit der expandierten Clusterung zur Referenz-Clusterung ist auf das Kontextwissen, das in der Struktur des zeitexpandierten Graphen enthalten ist, zurückzuführen. Einmalig auftretende Kontakte zwischen Knoten haben weniger Einfluss auf das Resultat. Überraschend war, dass die Modularity für beide Clusterungen ähnliche Werte zurückliefert. Das heißt, die gefundene Clusterung des gesamten zeitexpandierten Graphen hat im Hinblick auf die Modularity der einzelnen Zeitschritte ähnlich gute Ergebnisse wie die zeitlich flache Clusterung.

7.3. Anwendung auf die Beispiele

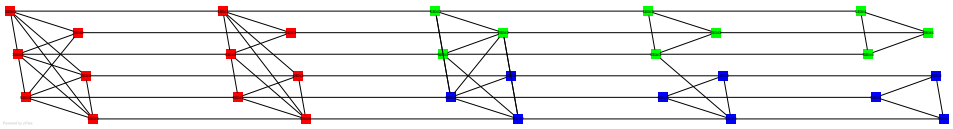
Das Clustern von zeitexpandierten Graphen soll uns helfen, die Änderungen von Gruppen innerhalb des Graphen zu erfassen. Es wäre wünschenswert, Vorgänge wie die Spaltung oder den Umbruch einer Gruppe zu erkennen. Des Weiteren wäre es von Vorteil, wenn kurzfristige Veränderungen eines ansonsten gleichbleibenden Verhaltens erfasst und diese nicht zu unerwünschten Nebeneffekten führen würden.

In Abschnitt 3.4 zeigten wir einige Beispiele für diese Problemstellungen auf. Für diese Beispiele haben wir nun aufgrund der Ergebnisse unserer Testreihen zeitexpandierte Graphen mit der Methode Cosine-Time und Reichweite $\mathrm{k} = 1$ erzeugt. Alle Intrazeitkanten haben in den Beispielen das Gewicht $\mathrm{w} = 0,7$.

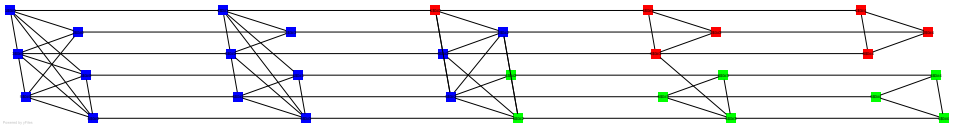
In Abbildung 37a ist der zeitexpandierte Graph für das Beispiel der Spaltung einer Gruppe zu sehen. Zunächst sind die beiden Cliquen stark miteinander verknüpft. Ab dem dritten Zeitschritt beginnt dieser Zusammenhang nachzulassen, bis sie im letzten Zeitschritt völlig getrennt sind. Die Abbildungen darunter zeigen die Clusterungen des zeitexpandierten Graphen, die wir mit den drei Cluster-Verfahren erzielten. Dabei überzeugen vor allem die Clusterungen mit dem ICC- und dem MCL-Verfahren mittels derer sich die Spaltung der beiden Cliquen gut nachvollziehen lässt. Bei der Significance-Clusterung erscheint die Trennung des blauen und des roten Clusters nicht intuitiv.



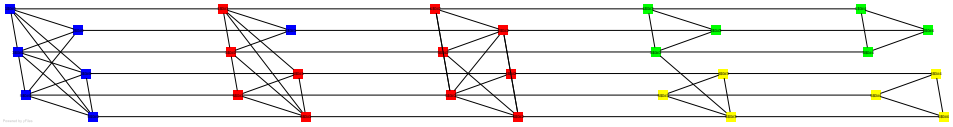
(a) Hier ist der Ausgangsgraph abgebildet. Die Farben stehen für die zwei Cliquen und die Kästen für die einzelnen Zeitschritte.



(b) Das Ergebnis der Clusterung mit dem ICC-Verfahren mit $\mathrm{a}^* = 0,22$. Die Farben stehen für die verschiedenen Cluster.



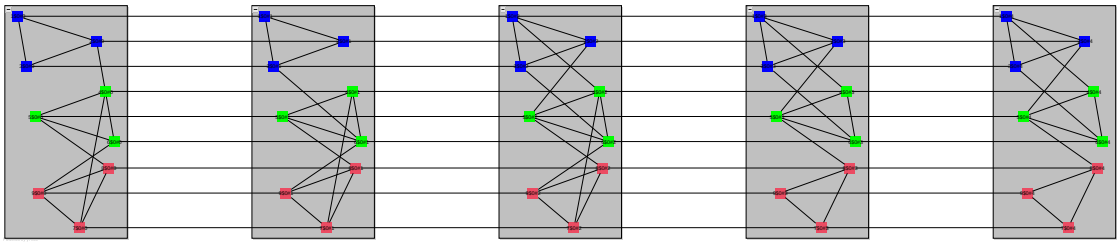
(c) Das Ergebnis der Clusterung mit dem MCL-Verfahren mit $\mathrm{e} = 3$ und $\mathrm{r} = 2$. Die Farben stehen für die verschiedenen Cluster.



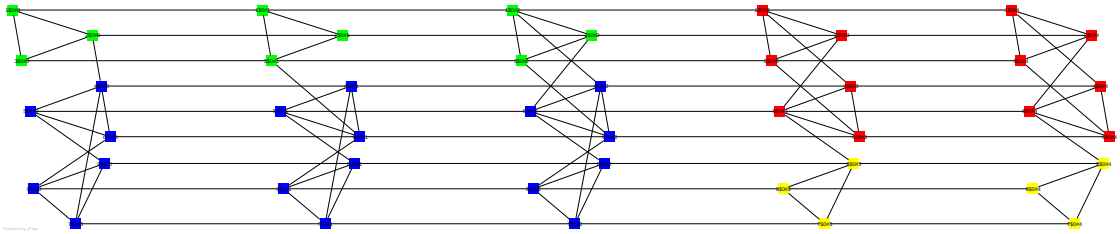
(d) Das Ergebnis der Clusterung mit dem Significance-Verfahren. Die Farben stehen für die verschiedenen Cluster.

Abbildung 37: *Die Abbildungen zeigen die Clusterungen des zeitexpandierten Graphen für unser Beispiel aus 3.4.1, die Spaltung einer Gruppe. Bei allen drei Clusterungen ist die Spaltung gut zu erkennen. Allerdings erschwert bei der Significance-Clusterung (37d) die Aufteilung der ersten drei Zeitschritte in den roten und den blauen Cluster die Interpretation der Clusterung.*

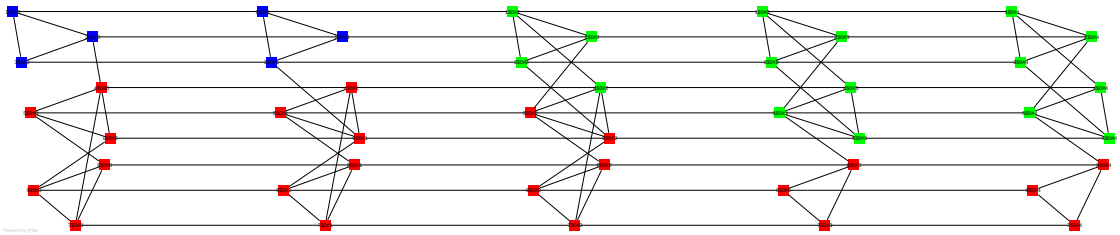
Bei dem Beispiel für den Umbruch einer Gruppe (siehe Abbildung 38 und 39) verhält es sich ähnlich. Hier besteht in den ersten zwei Zeitschritten ein starker Zusammenhang der unteren beiden Cliquen. Ab dem dritten Zeitschritt schwächt sich dieser Zusammenhang ab, während die oberen beiden Cliquen stärker zusammenwachsen. An den Clusterungen mit dem ICC- und MCL-Verfahren lässt sich dieser Umbruch gut nachvollziehen. Hingegen trennt das Significance-Verfahren den grünen und den türkisfarbenen Cluster. Wie im ersten Beispiel ist auch dies kein intuitives Ergebnis.



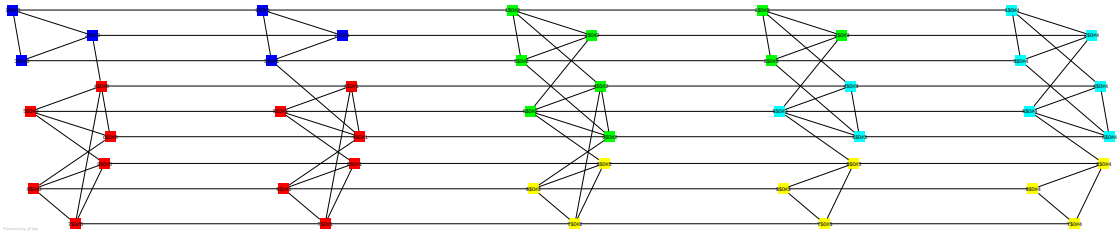
(a) Hier ist der Ausgangsgraph abgebildet. Die Farben stehen für die drei Cliquen und die Kästen für die einzelnen Zeitschritte.



(b) Das Ergebnis der Clusterung mit dem ICC-Verfahren mit $\mathrm{a}^* = 0,21$. Die Farben stehen für die verschiedenen Cluster.



(c) Das Ergebnis der Clusterung mit dem MCL-Verfahren mit $\mathrm{e} = 3$ und $\mathrm{r} = 2$. Die Farben stehen für die verschiedenen Cluster. Die gleiche Clusterung ist in Abbildung 39 vergrößert zu sehen.



(d) Das Ergebnis der Clusterung mit dem Significance-Verfahren. Die Farben stehen für die verschiedenen Cluster.

Abbildung 38: *Die Abbildungen zeigen die Clusterungen des zeitexpandierten Graphen für unser Beispiel aus Abschnitt 3.4.2, den Umbruch einer Gruppe. An den Clusterungen mit dem ICC- und MCL-Verfahren lässt sich dieser Umbruch der Gruppe gut nachvollziehen. Das Gleiche gilt für die Significance-Clusterung, jedoch gibt es in den letzten drei Zeitschritten zwei Cluster für die oberen beiden Cliquen. Eine gemeinsame Clusterung der beiden Cliquen in einem gemeinsamen Cluster wäre hier für das Verständnis hilfreicher.*

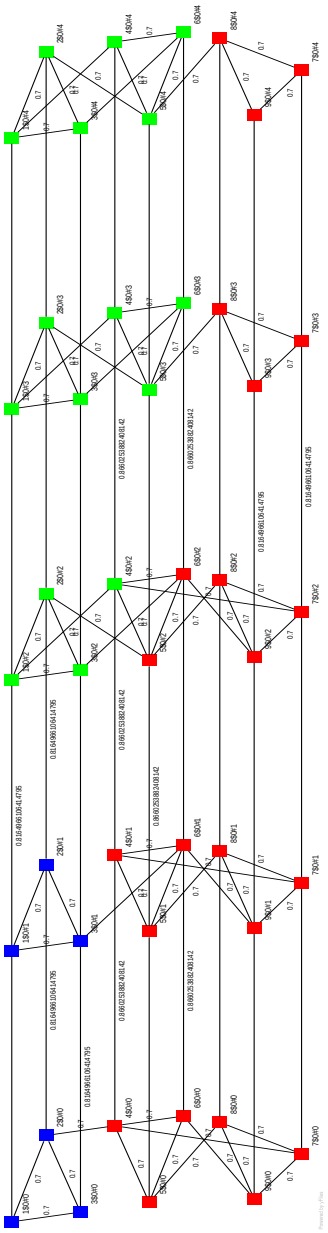


Abbildung 39: *Das Ergebnis der Clusterung mit dem MCL-Verfahren mit* $\mathrm{e} = 3$ *und* $\mathrm{r} = 2$ *für unser Beispiel 2 aus 3.4.2, den Umbruch einer Gruppe. Die Farben stehen für die verschiedenen Cluster. Man kann anhand der Clusterung gut erkennen, wie sich die Ausrichtung der mittleren Clique von der einen auf die andere Clique verlagert.*

Abbildung 53 im Anhang zeigt Beispiel 3 und dessen Clusterungen für einen zeitexpandierten Graphen mit Reichweite $k = 2$. Durch die erhöhte Reichweite fällt die zeitweise Veränderung des Verhaltens der Cliquen nicht so stark ins Gewicht. Die Clusterungen der Beispiele liefern uns viele Erkenntnisse über die Veränderungen der Graphen, doch zeigen die Ergebnisse der Significance-Clusterung bei den ersten beiden Beispielen, dass nicht immer ein hilfreiches Ergebnis zurückgeliefert wird.

7.4. Fazit der Testreihen

Die Interpretation der Clusterungen zu Beginn des Kapitels zeigt, dass die Ergebnisse sinnvoll und nachvollziehbar sind. Die Ergebnisse sind keine zufällig zusammengewürfelten Gruppen, sondern ähneln den von der Referenz-Clusterung vorgegebenen Lehrstühlen. Eine perfekte Übereinstimmung gibt es nicht, diese wäre auch nicht wünschenswert. Mit Hilfe unseres Modells wollen wir gerade die dynamischen Veränderungen erfassen.

Die ersten beiden Testreihen, bei denen wir die Methode Alpha für die Interzeitkantengewichte benutzten, lieferten die Erkenntnis, dass eine geeignete Wahl der Parameter α und k entscheidend für das Ergebnis der Clusterung sind (siehe Abbildung 48 im Anhang). Wird das starre Interzeikantengewicht α zu klein gewählt, gibt es keine Cluster über die Zeitgrenzen hinweg. Wählen wir α zu hoch im Vergleich zu den existierenden Intrazeitkantengewichten und der gewählten Reichweite k, entstehen aus den Repräsentanten der Knoten viele kleine Zeitschläuche.

Die Wahl der Reichweite hängt stark davon ab, ob die Clusterung möglichst nah an der Clusterung der einzelnen Zeitschritte bleiben soll, oder ob die Clusterung kurzzeitige Änderungen weniger berücksichtigen sollte. Unsere Untersuchung der einzelnen Zeitschritte offenbart eine klare Abhängigkeit der Ähnlichkeit zwischen den expandierten Clusterungen und den zeitlich flachen Clusterungen von der Reichweite des Graphen. Erhöhen wir die Reichweite, so nimmt die Ähnlichkeit der Clusterung des zeitexpandierten Graphen zur zeitlich flachen Clusterung ab. Legt eine Gruppe kurzzeitig durch eventuell verrauschte Daten ein stark verändertes Verhalten an den Tag, kann eine Reichweite $k > 1$ dazu führen, diese *Abnormalität* zu überbrücken oder zu überspringen (siehe dazu Lehrstuhl 6 an den Übergängen von Zeitschritt 3 bei der Clusterung in Abbildung 50 oder die Clusterungen zu Beispiel 3 in Abbildung 53). Die Betrachtung der Kerne (siehe Gleichung 22 in Abschnitt 6.3.2) innerhalb der Cluster zeigt, dass eine höhere Reichweite zu größeren Kernen innerhalb der Cluster führt. Dies entspricht einer größeren Konstanz der Clusterung über die Zeit.

Die Wahl der Schwelle ist stark anwendungsabhängig. Die Schwelle hilft die Anzahl der Intrazeitkanten bei dichten Graphen zu reduzieren. Bei verrauschten Daten kann die Schwelle zur Filterung benutzt werden. Sind die Daten exakt oder wird die Schwelle zu hoch gewählt, führt sie unweigerlich zu einem Informationsverlust. Die Ergebnisse spiegeln nicht mehr die korrekten Zusammenhänge wider. Bei unseren Daten mussten wir die Schwelle gut abwägen. Eine kleine Schwelle führte bei einem vertretbaren Informationsverlust bei allen Cluster-Verfahren zu einer Clusterung, die der Referenz-Clusterung deutlich ähnlicher war. Für höhere Schwellen kam es neben einem hohen Informationsverlust zu einer leichten Reduzierung dieser Ähnlichkeit.

Aber die Wahl der Parameter hängt ebenso von der Anwendung und dem verwendeten Cluster-Verfahren ab. Bei dem Iterative-Conductance-Cutting hat die Reichweite einen stärkeren Einfluss auf das Ergebnis der Clusterung als beim Greedy-Significance-Clustering. Dies alles muss bei der

Erzeugung der zeitexpandierten Graphen berücksichtigt werden. Generell sind die beiden Verfahren Iterative-Conductance-Cutting und Greedy-Significance-Clustering leichter handzuhaben als das Markov-Clustering. Für viele Graphen der Testreihen liefert das ICC-Verfahren für das gewählte Abbruchkriterium nur sehr grobe Clusterungen mit weniger als zehn Clustern. Versuche haben gezeigt, dass bei einigen dieser Graphen selbst eine starke Erhöhung des Schwellenwertes a^*, welcher das Abbruchkriterium für das ICC-Verfahren ist, zu keinen weiteren Verbesserungen führen. Offensichtlich schafft es die verwendete Heuristik in diesen Fällen nicht, weitere geeignete Schnitte zu finden. Es wäre interessant zu untersuchen, ob die spezielle Struktur des zeitexpandierten Graphen eine mögliche Ursache hierfür ist. Allerdings treten bei der Clusterung von zeitlich kollabierten Graphen des E-Mail-Netzwerkes vereinzelt ähnliche Probleme bei der Verwendung des ICC-Verfahrens auf. Das Greedy-Significance-Clustering liefert für die Cosine-Time-Testreihe durchgängig Ergebnisse, die eine starke Ähnlichkeit mit der Referenz-Clusterung aufweisen. Beim Markov-Clustering erschweren die drei Parameter die Handhabung des Verfahrens. Hier müssen die Parameter für jeden Graphen einzeln angepasst werden.

Betrachtet man die Laufzeit, so schlägt das ICC-Verfahren die beiden anderen deutlich. Die Clusterung von zehn Graphen der Cosine-Time-Testreihe dauerte mit dem ICC-Verfahren und Schwellenparameter $\mathrm{a}^* = 0,20$ knappe zwölf Minuten auf unserem AMD Opteron 2218 Prozessor mit 2,6 GHz. Das Significance-Verfahren benötigte für die zehn Graphen 48 Minuten. Beim Markov-Clustering führten wir zwei Durchläufe durch. Die Clusterung mit den Parametern $\mathrm{e} = 4$, $\mathrm{r} = 2$ und $\kappa = 50$ dauerte 78 Minuten. Beim zweiten Durchlauf mit den Parametern $\mathrm{e} = 3$, $\mathrm{r} = 2$ und $\kappa = 50$ benötigte das Verfahren 22 Minuten.

Die Testreihen begannen wir mit dem denkbar einfachsten Aufbau der zeitexpandierten Graphen, den Methoden Normal und Alpha. Das heißt, wir übernahmen bei den Intrazeitkanten die Einträge der Matrizen $\mathcal{A}(\mathrm{t})$ und legten ein fixes α für die Gewichte der Interzeitkanten fest. Bei der zweiten Testreihe erreichten wir durch die Verwendung der Methode Normed eine bessere Interpretierbarkeit der Intrazeitkantengewichte bei gleichbleibend guten Resultaten bezüglich unserer Referenz-Clusterung. Desgleichen machten sich vorhandene Ausreißer der Kantengewichte bei dieser Methode nicht mehr so stark bemerkbar. Die normierten Werten für die Intrazeitkantengewichte ermöglichten uns, die Verwendung der Methode Cosine-Time zur Berechnung der Interzeitkanten. Bei der dritten Testreihe, bei der wir die Methoden Cosine-Time und Normed benutzten, erhielten wir die konstantesten Ergebnisse. Viele Clusterungen wiesen eine erhöhte Ähnlichkeit zur Referenz-Clusterung auf. Durch den Einsatz der Cosine-Time Methode reduzierte sich die Anzahl der variablen Parameter auf zwei, die Schwelle und die Reichweite. Die Interzeitkantengewichte werden bei dieser Methode nicht willkürlich gewählt, sondern zeigen die Ähnlichkeit zweier Repräsentanten desselben Knotens in verschiedenen Zeitschritten auf. Repräsentanten eines Knotens mit geringer Ähnlichkeit sprechen dafür, dass sich die Ausrichtung des Knotens geändert hat und er in den verschiedenen Zeitschritten unterschiedlichen Gruppen angehört. Durch die Verwendung der Cosine-Time Methode spiegelt der zeitexpandierte Graph die dynamischen Veränderungen innerhalb des Graphen auch in den Interzeitkanten wider. Ein Vergleich anhand der Clusterungen zeigte, dass die Methode Cosine-Time eine bessere Wiedergabe der Veränderungen des dynamischen Graphen liefert, als starre Interzeitkantengewichte.

Es sei erwähnt, dass wir zwei weitere Testreihen durchführten. Eine Testreihe mit den Methoden Cosine und Cosine-Time und eine weitere mit den Methoden Mixed und Cosine-Time. Deren genauere Analyse würde den Rahmen dieser Arbeit sprengen. Beide Testreihen bestätigen jedoch die bisherigen Erkenntnisse im Bezug auf unser Modell. Die Verwendung der Methoden Cosi-

ne und Mixed für die Berechnung der Intrazeitkantengewichte führt zu einer starken Erhöhung der Anzahl der Intrazeitkanten. Dadurch liefert die Clusterung der erzeugten Graphen häufiger reine Intrazeit-Cluster zurück. Die Graphen wiesen im Vergleich zur Cosine-Time-Testreihe vor allem für kleinere Werte der Reichweite eine deutlich höhere Ähnlichkeit zur Zeit-Clusterung auf. Gleichfalls erscheinen uns die direkten Kontaktdaten der einzelnen Knoten sinnvoller zur Einteilung der zusammengehörigen Knoten innerhalb unseres E-Mail-Netzwerkes als deren Ähnlichkeit. Deshalb haben wir uns für eine ausführlichere Betrachtung der Cosine-Time-Testreihe entschieden.

Für die Beispiele in 7.3 lieferten die zeitexpandierten Graphen mit der Methode Cosine-Time interessante Ergebnisse. Die MCL- und ICC-Clusterungen spiegelten die Veränderungen innerhalb des dynamischen Graphen wider, während die Ergebnisse des Significance-Verfahrens nicht immer überzeugten.

Abschließend bleibt festzuhalten, dass sich die Clusterung eines zeitexpandierten Graphen gut eignet um die verborgenen Strukturen innerhalb unseres E-Mail-Netzwerkes zu erkennen. Zur Modellierung unserer Anwendung eignet sich die Verwendung der Methode Cosine-Time in Verbindung mit der Methode Normed am besten. Durch die Verwendung der Methode Cosine-Time ist eine bessere Wiedergabe der Veränderungen innerhalb des Netzwerkes anhand von Clusterungen möglich. Gleichzeitig erleichtert sich durch die Reduzierung der variablen Parameter die Handhabung unseres Modells.

Um kurzfristige Veränderungen innerhalb unseres E-Mail-Netzwerkes zu erfassen, bevorzugen wir Graphen mit der Reichweite $\mathrm{k} = 1$. Bei der Verwendung des Greedy-Significance-Clustering benötigen wir bei unserer Anwendung keine Schwelle für die erzeugten Graphen, um aussagekräftige Ergebnisse zu erzielen. Würden wir längere Zeitbereiche für unsere Zeitschritte festlegen, müssten wir aufgrund der höheren Dichte entweder eine Schwelle einführen oder die Reichweite erhöhen, um keine reinen Intrazeit-Cluster zu erhalten.

Soll eine bestimmte Granularität der Clusterung erreicht werden, wäre die Verwendung anderer Cluster-Verfahren, wie dem Markov-Clustering oder dem Iterative-Conductance-Cutting sinnvoll. Für zeitexpandierte Graphen mit vielen Zeitschritten d oder Anwendungen mit höheren Knotenzahlen wäre generell eine Verwendung eines Cluster-Verfahrens mit geringer Laufzeit wünschenswert. Hier überzeugt vor allem das Iterative-Conductance-Cutting mit seiner Geschwindigkeit.

8. Zusammenfassung und Ausblick

8.1. Zusammenfassung

Ziel der vorliegenden Arbeit war die konzeptionelle Entwicklung eines Modells für zeitexpandierte Graphen, deren theoretische Analyse sowie eine experimentelle Untersuchung und Anpassung des Modells anhand einer Anwendung im Sinne des Algorithm Engineering. Das von uns vorgestellte Modell vernetzt die verschiedenen zeitlichen Ausprägungen eines dynamischen Graphen über eine Menge von Interzeitkanten, welche Knoten verschiedener Zeitschritte verbinden. Dabei erfolgt die Wahl, zwischen welchen Repräsentanten der Knoten der einzelnen Zeitschritte diese Interzeitkanten verlaufen, anwendungsabhängig. Es ensteht ein zusammenhängender Graph, der zeitexpandierte Graph, der die zeitliche Entwicklung des dynamischen Graphen widerspiegelt.

Die Clusterung der zeitexpandierten Graphen eröffnet neue Möglichkeiten zur Untersuchung der Entwicklung von Gruppen innerhalb von Netzwerken. Die durch die Clusterung eines zeitexpandierten Graphen gefundenen Cluster verlaufen über die Zeitgrenzen hinweg und liefern so ein Bild einer Gruppe innerhalb des dynamischen Graphen und deren Veränderung im Lauf der Zeit. Gleichzeitig erhält man Aussagen über die Struktur des dynamischen Graphen. Im Gegensatz zu bisherigen Verfahren macht das Clustern von zeitexpandierten Graphen die komplizierte Zusammenführung der Clusterungen der verschiedenen Zeitschritte überflüssig. Bei unserem Ansatz werden alle Zeitschritte des dynamischen Graphen in einem Graphen erfasst. Die Untersuchung der Entwicklung der Gruppen erfolgt direkt durch die Clusterung eines zeitexpandierten Graphen. Im Idealfall liefern die Cluster eines zeitexpandierten Graphen eine Beschreibung der zeitlichen Entwicklung einer Gruppe des zugrundeliegenden dynamischen Graphen. Keines der bisherigen Verfahren liefert eine solche vollständige Beschreibung der zeitlichen Entwicklung von Gruppen.

Ein weiterer Vorteil des Modells ist der modulare Aufbau, sowie die mögliche Nutzung von Standard-Cluster-Verfahren. Die Festlegung der Interzeitkantenmenge, der verschiedenen Parameter und Methoden des Modells kann getrennt voneinander erfolgen. Mit Hilfe dieses Modells sollte es möglich sein, beliebige dynamische Graphen in einen zeitexpandierten Graphen zu überführen.

Mit Hilfe einer Reichweitenvariablen wird gesteuert, wie viele Zeitschritte die Interzeitkanten überbrücken dürfen, also wie langfristig zeitliche Zusammengehörigkeit in Betracht gezogen werden soll. Die Berechnung der konventionellen Kanten, der sogenannten Intrazeitkanten, und der Interzeitkanten erfolgt getrennt voneinander durch verschiedene Methoden. Diese Methoden werden ebenfalls anwendungsabhängig gewählt. Auf der Grundlage dieses Modells führten wir zahlreiche Testreihen durch, in deren Verlauf sich gewisse Eigenschaften, Stärken und Schwächen des Modells herauskristallisierten. Ähnlich zum Algorithm Engineering gliederte sich dabei der Ablauf der Testreihen in vier Phasen.

Wir begannen jede Testreihe mit einer Design-Phase, in der wir das Modell, die Methoden, die Wertebereiche der Parameter und die verwendeten Cluster-Verfahren festlegten. In der anschließenden Analyse-Phase wurden die Schwachstellen und Vorteile des Modells und der verwendeten Methoden diskutiert, sowie die Laufzeit zur Erzeugung und Clusterung der Graphen betrachtet. Ebenfalls stellten wir Hypothesen im Bezug auf das verwendete Modell auf. Während der Implementierungs-Phase erfolgte die Umsetzung der gewählten Methoden in unserem Java-

Framework. Das Ende jeder Testreihe bildete die Experimente-Phase. In ihr erfolgte eine Analyse der Clusterungen, sowie eine Überprüfung der aufgestellten Hypothesen. Abschließend fassten wir die Ergebnisse in einem kurzen Fazit zusammen.

Generell war festzustellen, dass eine hohe Reichweite der Interzeitkanten des Graphen eine Glättung der Clusterung zur Folge hat. Das durchschnittliche Verhalten der Knoten gewinnt für höhere Reichweiten an Bedeutung, gegenüber einer sinkenden Relevanz der kurzzeitigen Veränderungen. Clusterungen von Graphen mit Reichweite $\mathrm{k} = 1$ weisen daher die höchste Ähnlichkeit zur zeitlich flachen Clusterung auf. Dabei sind die Auswirkungen, die sich durch die Änderung der Reichweite ergeben, für kleine Werte am stärksten. Bei höheren Werten fallen die Veränderungen bei einer weiteren Anhebung der Reichweite schwächer aus.

Die von uns zunächst verwendeten statischen Interzeitkantengewichte eignen sich nicht für eine angemessene Beschreibung der Zusammenhänge zwischen den einzelnen Repräsentanten. Durch die Verwendung der Methode Cosine-Time enthält der zeitexpandierte Graph zusätzlich zu den Daten der einzelnen Zeitschritte die Information, wie sehr sich die Repräsentanten eines Knotens in den verschiedenen Zeitschritten gleichen. Wenn die Repräsentanten zeitlich benachbarter Zeitschritte eine starke Ähnlichkeit aufweisen, gehören sie mit hoher Wahrscheinlichkeit der selben Gruppierung an. Nachteil der Methode Cosine-Time ist, neben dem erhöhten Aufwand zur Erzeugung der zeitexpandierten Graphen, die nicht berücksichtigte Änderung der Knotenmenge zwischen den verschiedenen Zeitschritten.

Unser Modell hat einige Schwachstellen. Ein Hauptproblem der Verwendung der zeitexpandierten Graphen zur Analyse der zeitlichen Entwicklung besteht in dem hohen Aufwand für die Clusterung. Da sich die Anzahl der Knoten und Kanten gegenüber dem dynamischen Graphen um ein Vielfaches erhöht, steigt die Laufzeit im Vergleich zur Clusterung des dynamischen Graphen stark an. Bisher verwendeten wir Standard-Cluster-Verfahren für die Clusterung der zeitexpandierten Graphen. Ohne die Verwendung effizienter Cluster-Verfahren, die die besondere Struktur des zeitexpandierten Graphen berücksichtigen, ist es fraglich, ob das Modell für dynamische Graphen mit sehr hohen Knotenzahlen Anwendung findet.

Einen starken Einfluss auf das Ergebnis der Clusterung hat die Dichte der einzelnen Ausprägungen des dynamischen Graphen. Ist die Dichte der Ausprägungen zu hoch, das heißt, ist die Anzahl der Kanten innerhalb des Zeitschrittes im Vergleich zur Anzahl der Interzeitkanten sehr hoch, liefern die Cluster-Verfahren häufig die einzelnen Ausprägungen oder reine Intrazeit-Cluster als Cluster zurück. Unser Modell kann diese Unterschiede der Graphen bisher nicht gut kompensieren. Die Einführung eines Schwellenwertes p für die zeitexpandierten Graphen kann hier in einem gewissen Rahmen regulierend wirken. Allerdings ist die Einführung einer Schwelle immer mit einem Informationsverlust verbunden. Abhilfe könnte in solchen Fällen die Verwendung einer anderen Variante für die Festlegung der Interzeitkanten bringen.

Für unser E-Mail-Netzwerk lieferte die Clusterung der zeitexpandierten Graphen interessante Erkenntnisse. Die Cluster-Verfahren fanden, bei geeigneter Wahl der Parameter, Cluster, die über den gesamten Zeitbereich hinweglaufen. Diese Cluster wiesen eine hohe Ähnlichkeit zu den Lehrstühlen unserer Fakultät auf. Wir erkannten Verbindungen zwischen einzelnen Lehrstühlen, die organisatorisch oder thematisch zusammenhängen. Aufgrund der Clusterung bemerkten wir eine zeitweise starke Erhöhung der Knoten- und Kantenzahl und ein damit verbundenes, von der Norm abweichendes Verhalten. Interessant wäre in diesem Zusammenhang eine Betrachtung eines längeren Zeitbereiches über mehrere Jahre, diese könnte mögliche Veränderungen innerhalb

der Ausrichtungen der einzelnen Lehrstühle oder mögliche Kooperationen zwischen den Lehrstühlen sichtbar machen.

8.2. Ausblick

Nach dieser anwendungsbezogenen Untersuchung, wäre eine weitere theoretische Analyse des Modells anhand eines randomisierten Graph-Generators für zeitexpandierte Graphen denkbar. Eine mögliche Fragestellung wäre hierbei, ob die verborgene Struktur durch die Clusterung des zeitexpandierten Graphen erkannt wird. Ebenfalls interessant wäre die Untersuchung eines Netzwerkes mit einer höheren Dynamik als die unseres E-Mail-Netzwerkes und eine damit verbundene Analyse der Ergebnisse der Clusterungen.

Die von uns verwendeten Interzeitkanten geben die Ähnlichkeit der verbundenen Repräsentanten ohne Berücksichtigung der überbrückten Zeitspanne wieder. Möglich wäre hier eine Abschwächung der Interzeitkanten proportional zu der überbrückten Zeitspanne, da die Relevanz der Ähnlichkeit der Repräsentanten für zeitlich nähere Bereiche von größerer Bedeutung ist als für zeitlich entferntere Bereiche.

Weiter wäre es vorstellbar, spezielle Cluster-Verfahren für unser Modell zu entwerfen, welche die besondere Struktur berücksichtigen und die Probleme der reinen Intrazeit-Cluster beseitigen. Denkbar wäre zum Beispiel eine spezielle Variante des Markov-Clustering, das für die Bereiche der stochastischen Matrix, die für die Interzeit-Verbindungen stehen, einen anderen Inflationsparameter r benutzt wie für die Bereiche der Intrazeit-Verbindungen. Dadurch könnten auch für dynamische Graphen mit dichteren Ausprägungen verwertbare Clusterungen gefunden werden. Eine andere Variante für ein Cluster-Verfahren zeitexpandierter Graphen bestünde darin, die durch die Methode Cosine-Time berechneten Gewichte zu benutzen, um zeitlich aufeinander folgende sehr ähnliche Repräsentanten eines Knotens zu einem Cluster zu vereinen. Auf Basis dieser Vor-Clusterung wird dann die weitere Clusterung des zeitexpandierten Graphen durchgeführt.

Mit Hilfe der Clusterung eines zeitexpandierten Graphen könnte die Visualisierung eines dynamischen Graphen auf der Grundlage der gefundenen Cluster vereinfacht werden. Die Knotenmengen der verschiedenen Zeitschritte werden aufgrund ihrer Zugehörigkeit zu den Clustern positioniert. Dabei muss bedacht werden, dass sich die Zugehörigkeit eines Knotens über die Zeit ändern kann. Eine solche graphische Darstellung eines zeitexpandierten Graphen würde die visuelle Interpretation des dynamischen Graphen vereinfachen.

Es sind viele Anwendungen denkbar, für die diese Offenlegung der Entwicklung der Gruppen interessant wäre. Da wäre beispielsweise die Entwicklung von Gruppen innerhalb von sozialen Netzwerken, Web-Communities, Telekommunikations- oder Recommendation-Netzwerken. Die Clusterung von zeitexpandierten Graphen stellt hier eine interessante Alternative zu den bisherigen Verfahren dar.

Literatur

[BDG+08] Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Gorke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering*, 20(2):172–188, 2008.

[CKT06] Deepayan Chakrabarti, Ravi Kumar, and Andrew Tomkins. Evolutionary clustering. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 554–560, New York, NY, USA, 2006. ACM Press.

[CSZ+07] Yun Chi, Xiaodan Song, Dengyong Zhou, Koji Hino, and Belle L. Tseng. Evolutionary spectral clustering by incorporating temporal smoothness. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 153–162, New York, NY, USA, 2007. ACM Press.

[Del06] Daniel Delling. Analyse und Evaluierung von Vergleichsmaßen für Graphclusterungen. Diplomarbeit, Institut für Theoretische Informatik - Universität Karlsruhe (TH), February 2006.

[DGG+06] Daniel Delling, Marco Gaertler, Robert Görke, Zoran Nikoloski, and Dorothea Wagner. How to Evaluate Clustering Techniques. Technical Report 2006-24, ITI Wagner, Faculty of Informatics, Universität Karlsruhe (TH), 2006.

[Die06] Reinhard Diestel. *Graphentheorie*. Springer Verlag, 2006.

[FB07] Santo Fortunato and Marc Barthelemy. Resolution limit in community detection. *PROC.NATL.ACAD.SCI.USA*, 104:36, 2007.

[FF58] L. R. Ford and D. R. Fulkerson. Constructing maximal dynamic flows from static flows. *Operations Research*, 6(3):419–433, 1958.

[FF62] L.R. Ford and D.R. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton, N.J., U.S.A., 1962.

[GA99] G. Gambosi V. Kann A. Marchetti-Spaccamela M. Protasi G. Ausiello, P. Crescenzi. *Complexity and Approximation - Combinatorial optimization problems and their approximability properties*. Springer Verlag, 1999.

[Gae02] Marco Gaertler. Clustering with Spectral Methods. Diplomarbeit, Fachbereich Informatik und Informationswissenschaft, Universität Konstanz, March 2002.

[Gae05] Marco Gaertler. Clustering. In *Network Analysis: Methodological Foundations*, volume 3418 of *Lecture Notes in Computer Science*, pages 178–215. Springer-Verlag, February 2005.

[GGW07] Marco Gaertler, Robert Görke, and Dorothea Wagner. Significance-Driven Graph Clustering. In *Proceedings of The Third International Conference on Algorithmic Aspects in Information and Management (AAIM)*, pages 11–26, Portland, USA, 6–8 June 2007.

[GGWW06] Marco Gaertler, Robert Görke, Dorothea Wagner, and Silke Wagner. How to Cluster Evolving Graphs. In *Proceedings of the European Conference of Complex Systems (ECCS'06)*, September 2006.

[HKKS04] John Hopcroft, Omar Khan, Brian Kulis, and Bart Selman. Tracking evolving communities in large linked networks. *Proceedings of the National Academy of Sciences*, 101:5249–5253, April 2004.

[HLW07] Ya Huang, Shixia Liu, and Yi Wang. Online detecting and tracking of the evolution of user communities. *icnc*, 2:681–685, 2007.

[KVV00] Ravi Kannan, Santosh Vempala, and Adrian Vetta. On clusterings: Good, bad, and spectral. In *Proceedings of the 41st Annual Symposium on the Foundation of Computer Science*, pages 367–380. IEEE Computer Society, November 2000.

[Mei03] Marina Meila. Comparing clusterings by the variation of information. In *Proceedings of the Sixteenth Annual Conference on Computational Learning Theory (COLT)*. Springer, 2003.

[Mei05] Marina Meila. Comparing clusterings - an axiomatic view. In *International Conference on Machine Learning (ICML)*, 2005.

[Mei07] Marina Meilă. Comparing clusterings - an information based distance. *J. Multivar. Anal.*, 98(5):873–895, 2007.

[NG04] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69:026113, 2004.

[PBV07] Gergely Palla, Albert-Laszlo Barabasi, and Tamas Vicsek. Quantifying social group evolution. *Nature*, 446(7136):664–667, April 2007.

[Tur04] Volker Turau. *Algorithmische Graphentheorie*. Oldenbourg Wissenschaftsverlag, 2004.

[vD98] Stijn van Dongen. A new cluster algorithm for graphs. In *281*, page 42. Centrum voor Wiskunde en Informatica (CWI), ISSN 1386-3681, 31 1998.

[vD00] Stijn M. v. van Dongen. *Graph Clustering by Flow Simulation*. PhD thesis, University of Utrecht, May 2000.

[vL07] Ulrike von Luxburg. A tutorial on spectral clustering, Nov 2007.

Tabellenverzeichnis

1. Indizes der Referenz-Clusterung der Normal-Testreihe. 42
2. Indizes der gefundenen Greedy-Significance-Clusterungen der Normal-Testreihe. 42
3. Durchschnittswerte des $\mathrm{bestmatch_{no}}$ der ICC-Clusterungen bezüglich der Referenz-Clusterung der Normal-Testreihe für verschiedene Werte von a^*. 49
4. Vergleich der verschiedene Durchläufe mit dem MCL-Verfahren bei der Normal-Testreihe. 50
5. Durchschnittswerte des $\mathrm{bestmatch_{no}}$ der Clusterungen der verschiedenen Cluster-Verfahren bezüglich der Referenz-Clusterung in der Normal-Testreihe äbhängig von der Höhe der Schwelle. 52
6. Durchschnittswerte des $\mathrm{bestmatch_{no}}$ der Clusterungen der verschiedenen Cluster-Verfahren bezüglich der Referenz-Clusterung in der Normal-Testreihe äbhängig von der Höhe der Reichweite. 52
7. Indizes der Referenz-Clusterung der Normed-Testreihe. 59
8. Indizes der gefundenen Significance-Clusterungen der Normed-Testreihe. 59
9. Indizes der ICC-Clusterungen der Normed-Testreihe für $\mathrm{a}^* = 0,05$. 61
10. Indizes der Referenz-Clusterung der Cosine-Time-Testreihe. 67
11. Indizes der Significance-Clusterungen der Cosine-Time-Testreihe. 67
12. Durchschnittswerte der Indizes der ICC-Clusterungen der Graphen der Cosine-Time-Testreihe. 71
13. Vergleich des E-Mail-Netzwerks in den verschiedenen Zeitschritten. 77
14. Durchschnittswerte der Vergleichsmaße für die zeitexpandierten und zeitlich flachen Clusterungen abhängig von der Reichweite. 79
15. Indexwerte der expandierten Clusterungen. 79
16. Durchschnittswerte der Vergleichsmaße für die zeitexpandierten und zeitlich flachen Clusterungen. 81
17. Maximale Anzahl der Interzeitkanten eines zeitexpandierten Graphen mit Variante 1. 112
18. Indizes der ICC-Clusterungen der Graphen der Normal-Testreihe für $\mathrm{a}^* = 0,20$. . 112
19. Indizes der ICC-Clusterungen der Graphen der Normal-Testreihe für $\mathrm{a}^* = 0,10$. . 112
20. Indizes der ICC-Clusterungen der Graphen der Normal-Testreihe für $\mathrm{a}^* = 0,075$. 112
21. Indizes der ICC-Clusterungen der Graphen der Normal-Testreihe für $\mathrm{a}^* = 0,05$. . 113
22. Indizes der MCL-Clusterungen der Graphen der Normal-Testreihe. 113
23. Durchschnittswerte der Vergleichsmaße für die zeitexpandierten und zeitlich flachen Clusterungen in Abhängigkeit von der Schwelle. 113

Abbildungsverzeichnis

1. Beispiel für einen zeitexpandierten Graphen. 1
2. Beispiel für einen dynamischen Graph. 7
3. Beispiele für zeitexpandierte Graphen. 8
4. Abbildung zur Coverage. 13
5. Abbildungen zur Performance einer Clusterung. 14
6. Abbildung zur Conductance. 15
7. Clusterung $\mathcal{C}$ eines Graphen mit hoher durchschnittlicher Conductance. 16
8. Abbildung zu dem Schnittmaß $\mathrm{match}_{\mathrm{old}}$ zweier Cluster. 24
9. Abbildung zum $\mathrm{bestmatch}_{\mathrm{no}}$. 25
10. Venn-Diagramm der Entropien zweier Clusterungen $\mathcal{C}$ und $\mathcal{C}'$. 27
11. Erläuterungen zu den Abbildungen der zeitexpandierten Graphen. 30
12. Vorgehen beim Entwurf. 31
13. Die Kantengewichte von Normal, Normed und Cosine gerundet auf zwei Dezimale für das Beispiel aus 2.6.1. 33
14. Beispiel für die Spaltung einer Gruppe. 36
15. Beispiel für den Umbruch einer Gruppe. 36
16. Beispiel für die kurzeitige Änderung eines ansonsten gleichbleibenden Verhaltens. 37
17. Vergleich der Referenz-Clusterung und der Significance-Clusterung der Graphen der Normal-Testreihe anhand der Modularity. 43
18. Vergleich der Referenz-Clusterung und der Significance-Clusterung der Graphen der Normal-Testreihe anhand der Interclusterconductance. 44
19. Vergleich der Referenz-Clusterung und der Significance-Clusterung der Graphen der Normal-Testreihe anhand der Coverage. 45
20. Einfluss der Parameter α und k auf das Greedy-Significance-Clustering der Graphen der Normal-Testreihe. 47
21. Einfluss der Schwelle auf die ICC-Clusterungen der Graphen der Normal-Testreihe mit $\mathrm{a}^* = 0,10$. 48
22. Einfluss der Reichweite auf die ICC-Clusterungen der Graphen der Normal-Testreihe mit $\mathrm{a}^* = 0,10$. 49
23. Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der Significance-Clusterungen zur Referenz-Clusterung bei den Graphen der Normal-Testreihe. . . 53
24. Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der ICC-Clusterungen $(\mathrm{a}^* = 0,05)$ zur Referenz-Clusterung bei den Graphen der Normal-Testreihe. 54
25. Vergleich der Coverage der Zeit-Clusterung mit der Ähnlichkeit der MCL-Clusterungen zur Referenz-Clusterung bei den Graphen der Normal-Testreihe. . . 55
26. Auswirkung der Coverage der Zeit-Clusterung auf die Significance-Clusterungen der Normed-Testreihe. 60
27. Betrachtung der Dichte der Graphen für die Normed-Testreihe. 60
28. Auswirkung der Coverage der Zeit-Clusterung auf die ICC-Clusterungen der Normed-Testreihe in Abhängigkeit der Reichweite. 62
29. Auswirkung der Coverage der Zeit-Clusterung auf die ICC-Clusterungen der Normed-Testreihe in Abhängigkeit der Schwelle. 62
30. Einfluss der Reichweite auf die Ähnlichkeit der Significance-Clusterungen zur Referenz-Clusterung bei der Cosine-Time-Testreihe. 68

31. Einfluss der Parameter auf die Significance-Clusterungen der Cosine-Time-Testreihe. 70
32. Einfluss der Parameter auf die Clusterung der Cosine-Time-Graphen. 72
33. Auswirkungen der Anzahl der Cluster auf die Ähnlichkeit zur Referenz-Clusterung. 73
34. Ausschnitt der Significance-Clusterung zweier zeitexpandierter Graphen mit verschiedener Reichweite. 74
35. Illustration der Zerschneidung der zeitexpandierten Graphen. 78
36. Vergleich der expandierten und der zeitlich flachen Clusterungen für die verschiedenen Zeitschritte. 82
37. Der zeitexpandierte Graph für unser Beispiel 1 aus 3.4.1. 83
38. Der zeitexpandierte Graph für unser Beispiel 2 aus 3.4.2. 84
39. Das Ergebnis der Clusterung mit dem MCL-Verfahren mit $\mathrm{e} = 3$ und $\mathrm{r} = 2$ für unser Beispiel 2 aus 3.4.2. 85
40. Significance-Clusterung eines zeitexpandierten Graphen der Normal-Testreihe mit zu klein gewähltem α. 98
41. Einfluss der Schwelle auf die Referenz-Clusterung. 99
42. Einfluss der Reichweite auf die Referenz-Clusterung. 100
43. Plots zu den ICC-Clusterungen mit $\mathrm{a}^* = 0,05$. 101
44. Plots zu den ICC-Clusterungen mit $\mathrm{a}^* = 0,075$. 102
45. Plots zu den ICC-Clusterungen mit $\mathrm{a}^* = 0,10$. 103
46. ICC-Clusterung eines zeitexpandierten Graphen der ersten Testreihe mit $\mathrm{a}^* = 0,075$ und hoher Ähnlichkeit zur Referenz-Clusterung. 104
47. Vergleich der Significance- und ICC-Clusterungen der Normal-Testreihe. 105
48. Auswirkungen durch Veränderungen der statischen Interzeitkantengewichte α auf die Ähnlichkeit der Clusterungen zur Referenz-Clusterung. 106
49. Einfluss der variablen Parameter auf die Coverage der Zeit-Clusterung bei den Graphen der Normal-Testreihe. 107
50. Significance-Clusterung eines zeitexpandierten Graphen mit Reichweite $\mathrm{k} = 3$, Schwelle $\mathrm{p} = 0$ und Interzeitkantengewicht $\alpha = 5$. 108
51. Häufig auftretente Gemeinsamkeiten der Clusterungen. 109
52. Häufig auftretente Gemeinsamkeiten der Clusterungen. 110
53. Der zeitexpandierte Graph mit der Methode Cosine-Time und Reichweite $\mathrm{k} = 2$ für unser Beispiel 3 aus 3.4.3. 111

A. Anhang

A.1. Abbildungen

Die Graphiken der Gesamtgraphen sind beim Ausdruck auf DinA4 wohl nicht hilfreich. Allderings kann man im PDF schön reinzoomen.

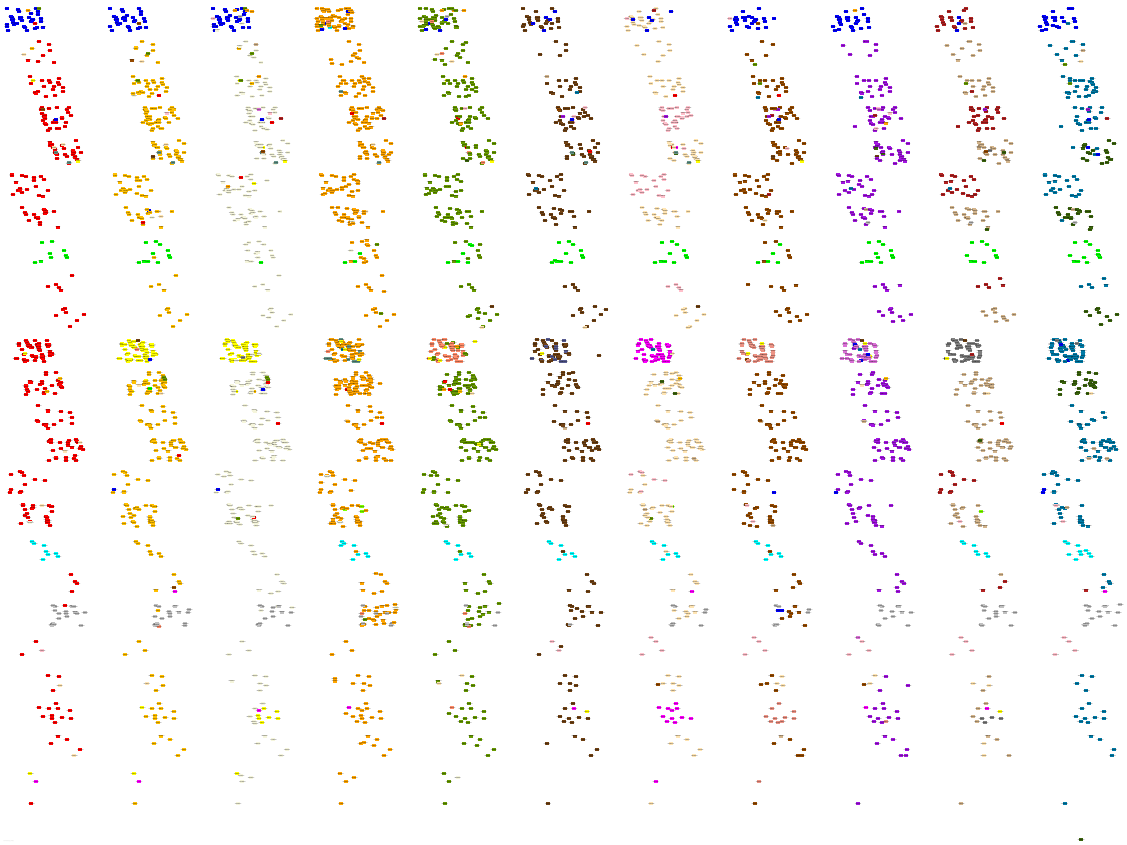
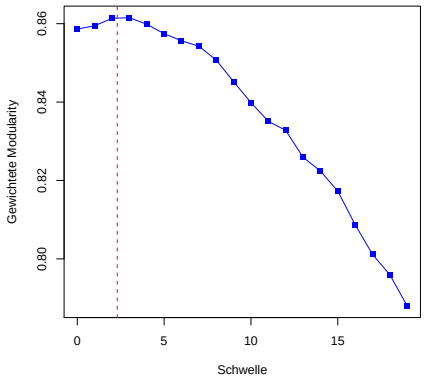
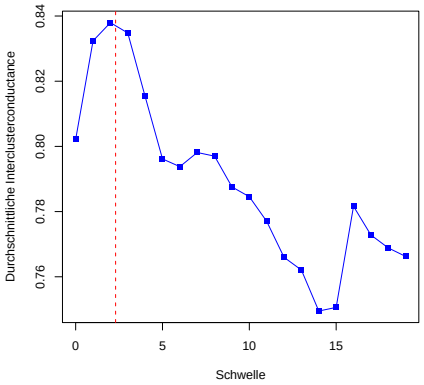


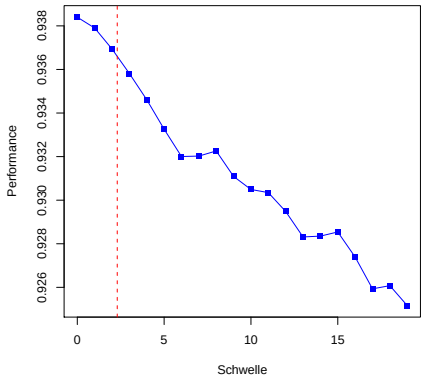
Abbildung 40: *Significance-Clusterung eines zeitexpandierten Graphen der Normal-Testreihe mit Reichweite* $\mathrm{k} = 7$, *Schwelle* $\mathrm{p} = 0$ *und Interzeitkantengewicht* $\alpha = 1$. *Jede Farbe steht für einen Cluster.*



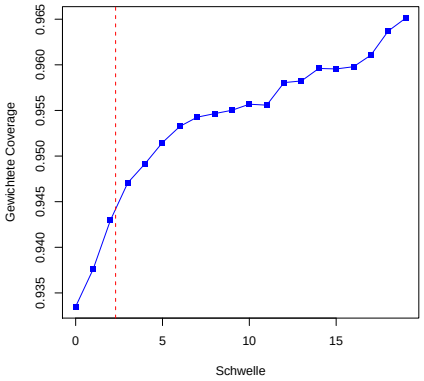
(a) Auswirkung der Schwelle auf die gewichtete Modularity.



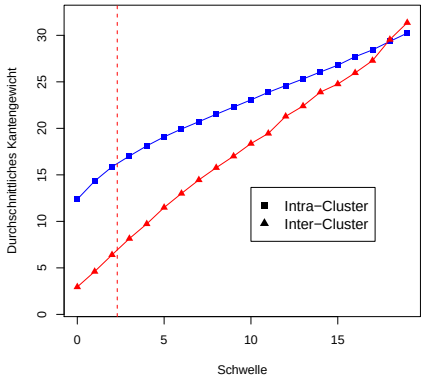
(b) Auswirkung der Schwelle auf die durchschnittliche Interclusterconductance.



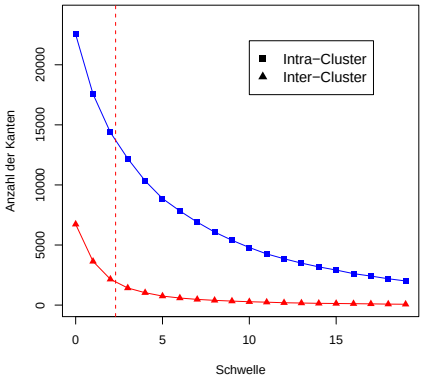
(c) Auswirkung der Schwelle auf die Performance.



(d) Auswirkung der Schwelle auf die gewichtete Coverage.

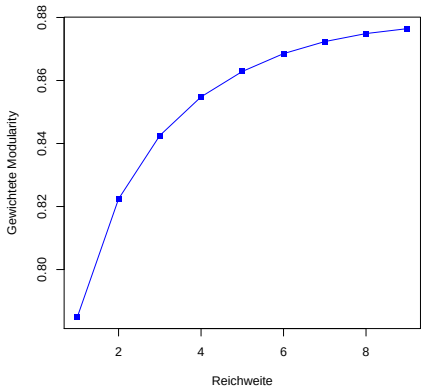


(e) Auswirkung der Schwelle auf die durchschnittlichen Gewichte der Intra- und Interclusterkanten.

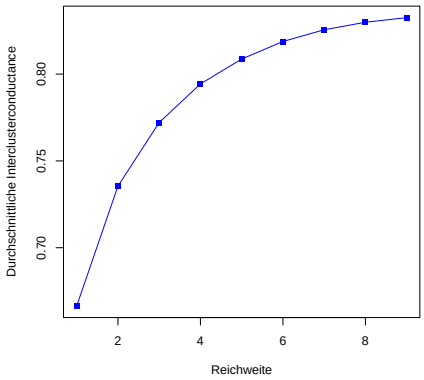


(f) Auswirkung der Schwelle auf die Anzahl der Intra- und Interclusterkanten.

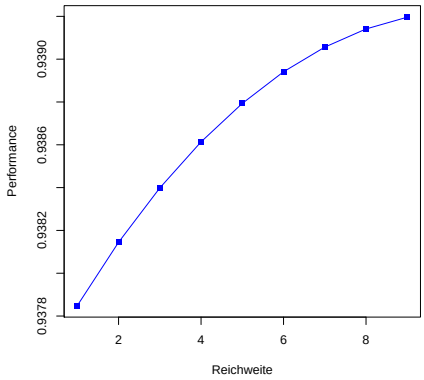
Abbildung 41: *Die Auswirkungen auf die Bewertung der Referenz-Clusterung durch die Erhöhung der Schwelle bei konstanter Reichweite* $\mathrm{k} = 3$ *und konstantem Interzeitkantengewicht* $\alpha = 20$.



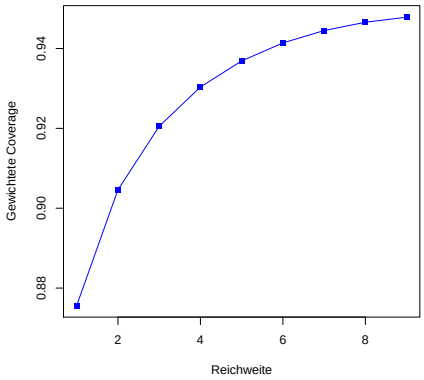
(a) Auswirkung der Reichweite auf die gewichtete Modularity.



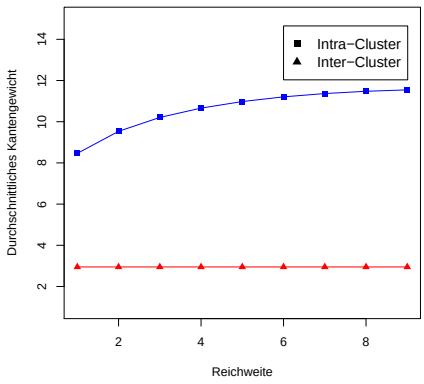
(b) Auswirkung der Reichweite auf die durchschnittliche Interclusterconductance.



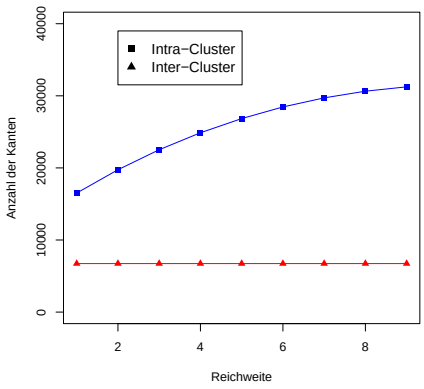
(c) Auswirkung der Reichweite auf die Performance.



(d) Auswirkung der Reichweite auf die gewichtete Coverage.

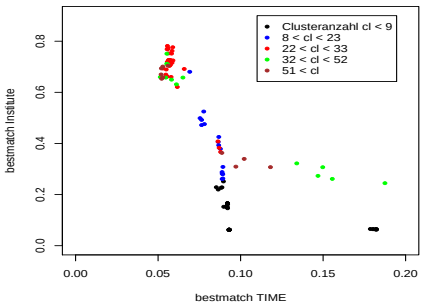


(e) Auswirkung der Reichweite auf die durchschnittlichen Gewichte der Intra- und Interclusterkanten.

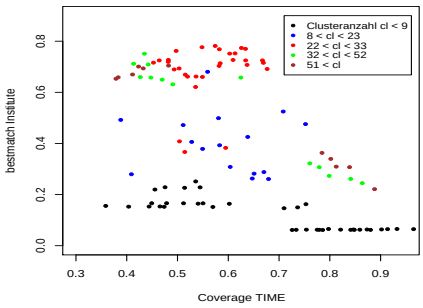


(f) Auswirkung der Reichweite auf die Anzahl der Intra- und Interclusterkanten.

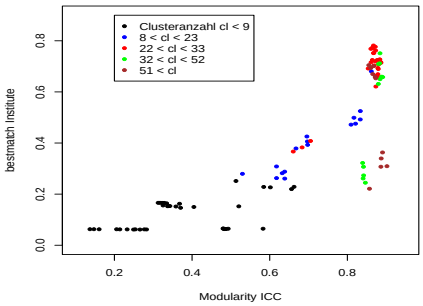
Abbildung 42: *Die Auswirkungen auf die Bewertung der Referenz-Clusterung durch die Erhöhung der Reichweite bei konstanter Schwelle* $\mathrm{p} = 0$ *und konstantem Interzeitkantengewicht* $\alpha = 15$.



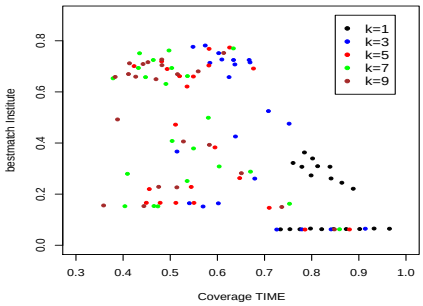
(a) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$. Die Farben stehen für die unterschiedlichen Clusterzahlen.



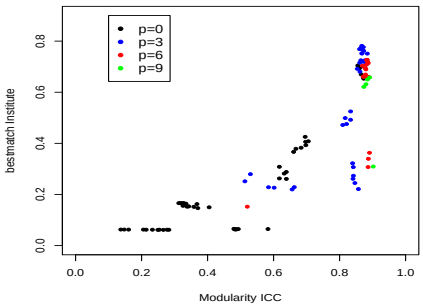
(b) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Coverage der Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Clusterzahlen.



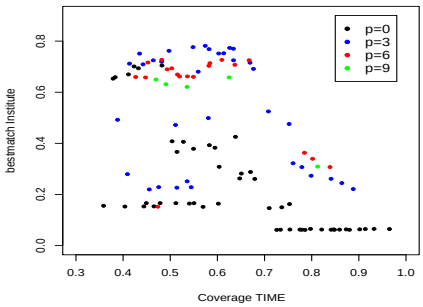
(c) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Modularity der ICC-Clusterung. Die Farben stehen für die unterschiedlichen Clusterzahlen.



(d) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Coverage der Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Reichweiten der Graphen.

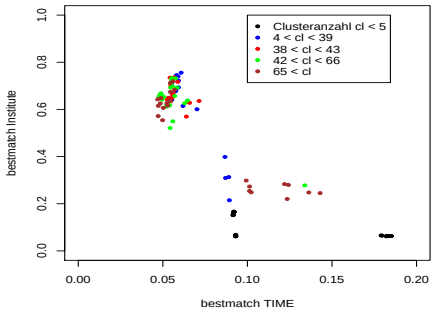


(e) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Modularity der ICC-Clusterung. Die Farben stehen für die unterschiedlichen Schwellen.

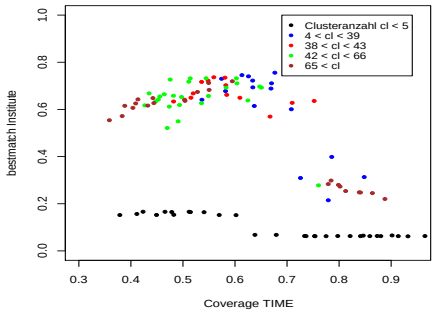


(f) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Coverage der Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Schwellen der Graphen.

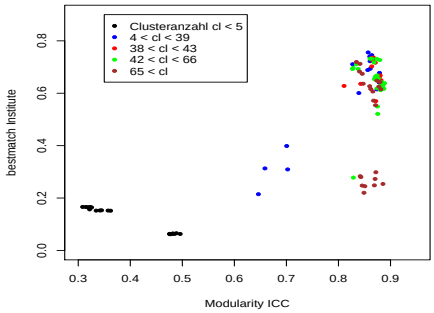
Abbildung 43: *Plots zu den ICC-Clusterungen mit* $\mathrm{a}^* = 0,05$. *Die Plots zeigen die Auswirkungen der Parameter und der Clusterzahlen auf den* $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ *der ICC-Clusterung* $\mathcal{C}'$ *bezüglich der Referenz-Clusterung* $\mathcal{C}$.



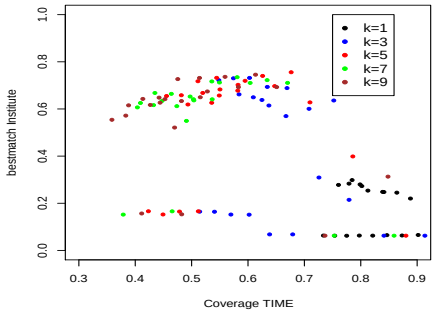
(a) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$. Die Farben stehen für die unterschiedlichen Clusterzahlen.



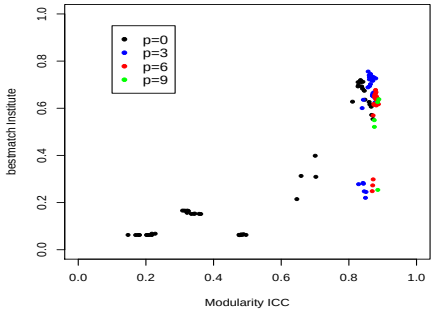
(b) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Coverage der Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Clusterzahlen.



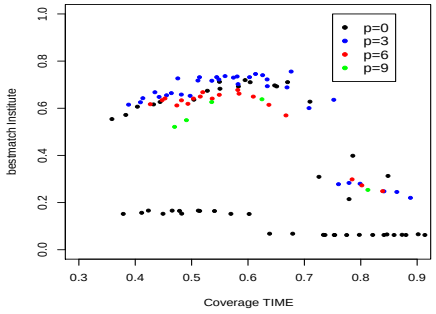
(c) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Modularity der ICC-Clusterung. Die Farben stehen für die unterschiedlichen Clusterzahlen.



(d) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Coverage der Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Reichweiten der Graphen.

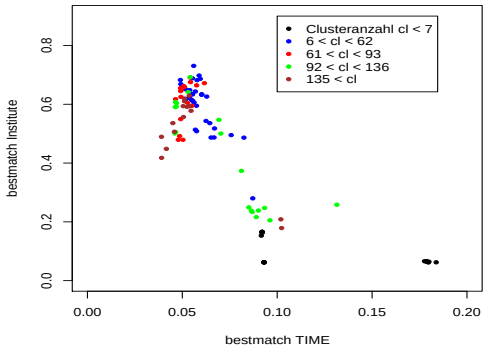


(e) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Modularity der ICC-Clusterung. Die Farben stehen für die unterschiedlichen Schwellen.

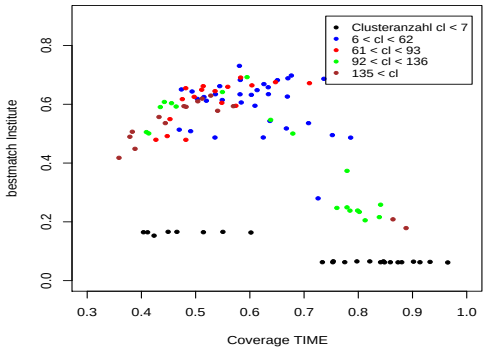


(f) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Coverage der Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Schwellen der Graphen.

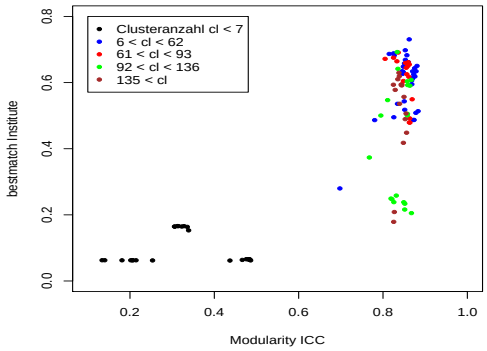
Abbildung 44: *Plots zu den ICC-Clusterungen mit* $\mathrm{a}^* = 0,075$. *Die Plots zeigen die Auswirkungen der Parameter und der Clusterzahlen auf den* $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ *der ICC-Clusterung* $\mathcal{C}'$ *bezüglich der Referenz-Clusterung* $\mathcal{C}$.



(a) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Ähnlichkeit der ICC-Clusterung $\mathcal{C}'$ zur Zeit-Clusterung $\mathcal{C}_{\mathrm{time}}$ anhand des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}_{\mathrm{time}}, \mathcal{C}')$. Die Farben stehen für die unterschiedlichen Clusterzahlen.



(b) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Coverage der Zeit-Clusterung. Die Farben stehen für die unterschiedlichen Clusterzahlen.



(c) Vergleich des $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ und der Modularity der ICC-Clusterung. Die Farben stehen für die unterschiedlichen Clusterzahlen.

Abbildung 45: *Plots zu den ICC-Clusterungen mit* $\mathrm{a}^* = 0,10$. *Die Plots zeigen die Auswirkungen der Clusterzahlen auf den* $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ *der ICC-Clusterung* $\mathcal{C}'$ *bezüglich der Referenz-Clusterung* $\mathcal{C}$.

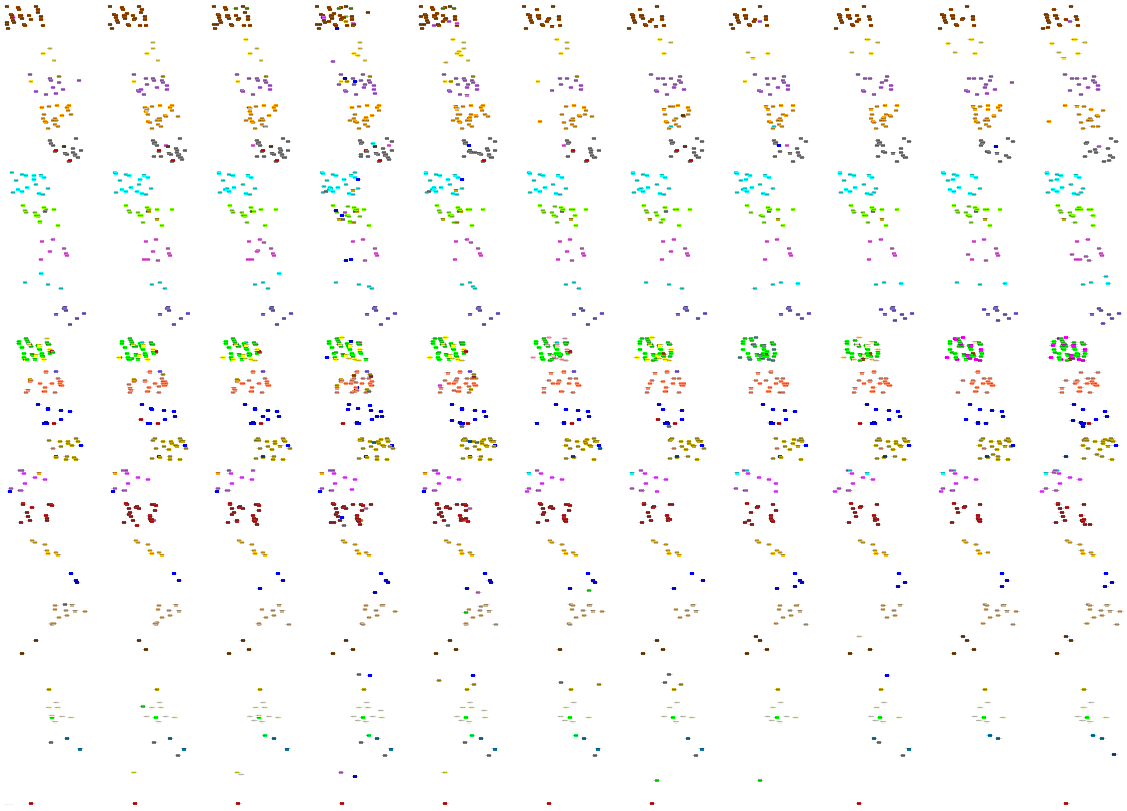
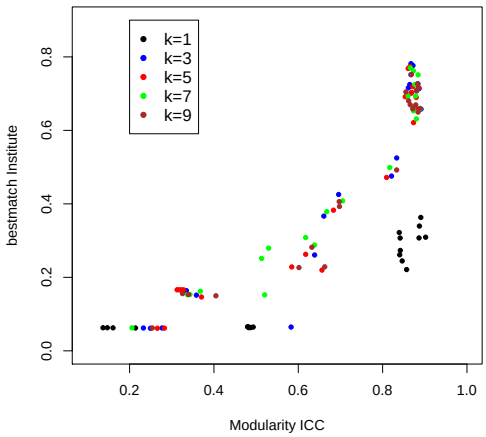
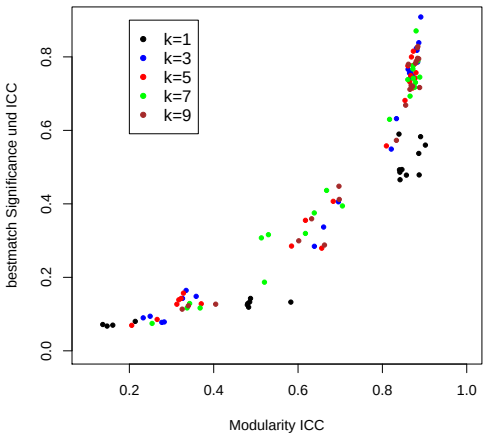


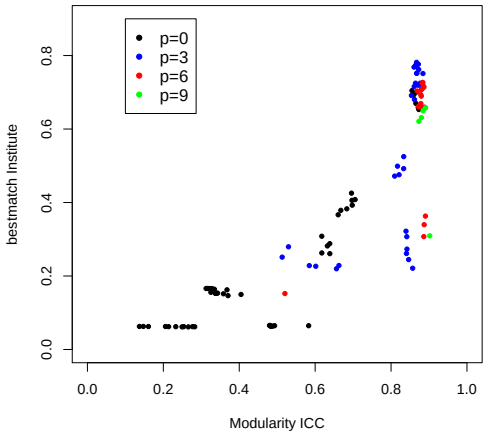
Abbildung 46: *ICC-Clusterung mit* $\mathrm{a}^* = 0,075$ *eines zeitexpandierter Graph mit Reichweite* $\mathrm{k} = 3$*, Schwelle* $\mathrm{p} = 3$ *und starrem Interzeitkantengewicht* $\alpha = 9$*. Jede Farbe steht für einen Cluster. Diese Clusterung ergab einen* $\mathrm{bestmatch}_{\mathrm{no}}$ *von* $0,78$ *bezüglich der Referenz-Clusterung.*



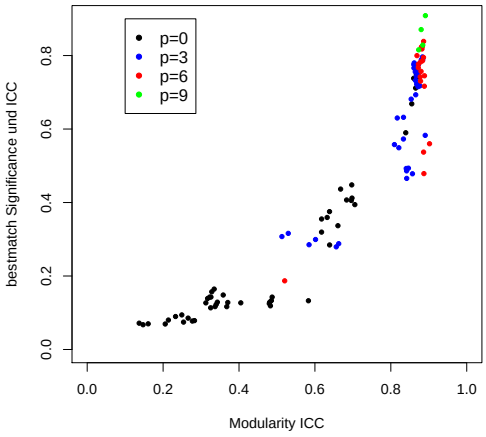
(a) Vergleich der Ähnlichkeit $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}')$ der ICC-Clusterung $\mathcal{C}'$ zur Referenz-Clusterung $\mathcal{C}$ anhand der gewichteten Modularity $\mathrm{mod}_{\mathrm{w}}(\mathcal{C}')$ der ICC-Clusterung. Die Farben stehen für die unterschiedlichen Reichweiten der Graphen.



(b) Vergleich der Ähnlichkeit $\mathrm{bestmatch}_{\mathrm{no}}(\mathcal{C}, \mathcal{C}_{\mathrm{sig}})$ der Significance-Clusterung $\mathcal{C}_{\mathrm{sig}}$ zur ICC-Clusterung $\mathcal{C}'$ anhand der gewichteten Modularity $\mathrm{mod}_{\mathrm{w}}(\mathcal{C}')$ der ICC-Clusterung. Die Farben stehen für die unterschiedlichen Reichweiten der Graphen.

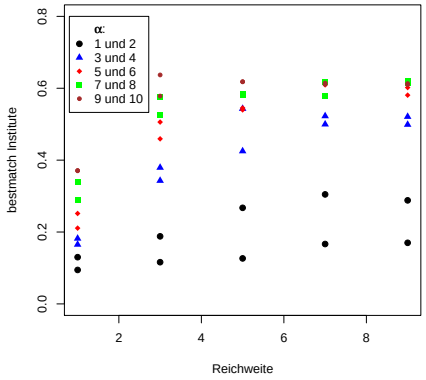


(c) Abbildung ist identisch zu 47a, bis auf die Tatsache, dass die Farben für die unterschiedlichen Schwellen der Graphen stehen.

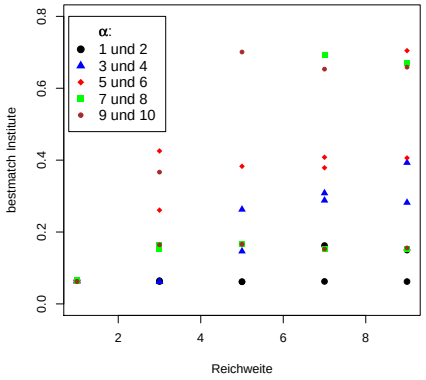


(d) Abbildung ist identisch zu 47b, bis auf die Tatsache, dass die Farben für die unterschiedlichen Schwellen der Graphen stehen.

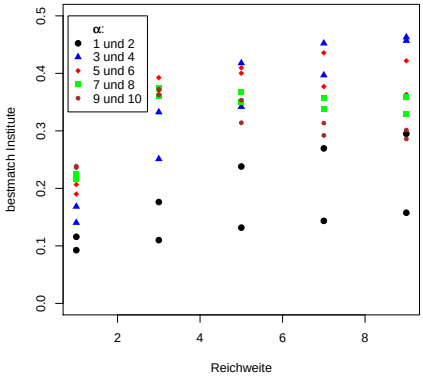
Abbildung 47: *Vergleich der Significance- und ICC-Clusterungen mit* $\mathrm{a}^* = 0,05$ *der Normal-Testreihe. Die Ähnlichkeit der ICC-Clusterung bezüglich der gefundenen Significance-Clusterung wird in den beiden rechten Bildern dargestellt. Auf der linken Seite steht dem, die Ähnlichkeit zur Institute-Clusterung gegenüber. Man erkennt eine starke Zunahme der Ähnlichkeit beider Clusterungen bei der Erhöhung der Schwelle.*



(a) Ähnlichkeit der Significance-Clusterungen zur Referenz-Clusterung.



(b) Ähnlichkeit der ICC-Clusterungen (mit $a^* = 0,05$) zur Referenz-Clusterung.



(c) Ähnlichkeit der MCL-Clusterungen (mit $e = 4$, $r = 2$ und $\kappa = 120$) zur Referenz-Clusterung.

Abbildung 48: *Einfluss der statischen Interzeitkantengewichte α auf die Clusterung der Graphen mit Schwelle $p{=}0$ für alle drei Cluster-Verfahren. Die einzelnen Punkte stehen für die verschiedenen Graphen, deren Farbgebung durch die Interzeitkantengewichte bestimmt ist. Bei der Clusterung mit dem Significance-Verfahren brachte eine Erhöhung von α generell eine höhere Ähnlichkeit zur Referenz-Clusterung. Für die Reichweite $k = 1$ fallen die Unterschiede am geringsten aus. Für die ICC-Clusterungen sind die Ergebnisse sehr unterschiedlich, was wir auf die stark schwankende Granularität der Clusterungen zurückführen. Eine sehr hohe Ähnlichkeit zur Referenz-Clusterung wird nur bei hohen Werten für α und k erreicht. Das Markov-Clustering zeigt ein anderes Bild. Die Ähnlichkeit zur Referenz-Clusterung ist, abhängig von der Reichweite, für unterschiedliche Werte von α am größten. Je höher die Reichweite, desto kleiner ist der Wert von α, für den die Clusterung des entsprechenden Graphen eine möglichst hohe Ähnlichkeit zur Referenz-Clusterung hat.*

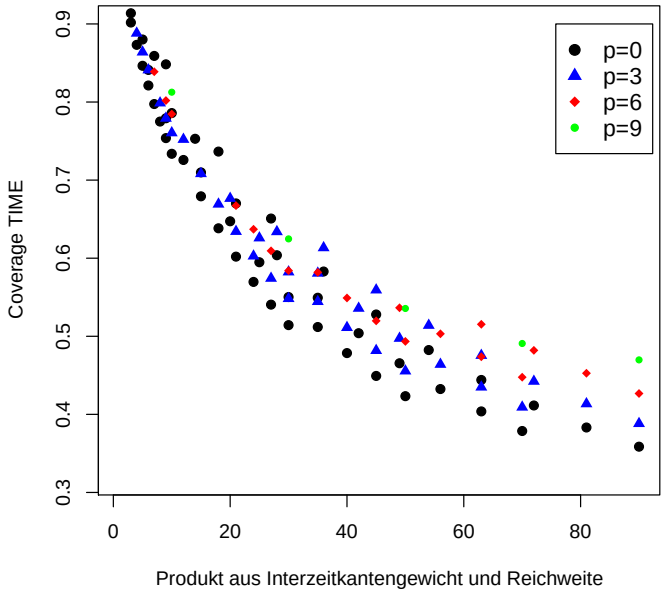


Abbildung 49: *Vergleich des Produktes von Reichweite* k *und Interzeitkantengewicht* α *und der Coverage der Zeit-Clusterung. Die einzelnen Punkte stehen für die 108 zeitexpandierten Graphen. Dabei stehen die Farben für die unterschiedlichen Schwellen der Graphen. Es ist eine starke Korrelation zwischen den beiden Parametern Reichweite* k *und Interzeitkantengewicht* α *und der Coverage der Zeit-Clusterung zu beobachten. Die Schwelle hat hingegen keine starke Auswirkung auf die Coverage der Zeit-Clusterung.*

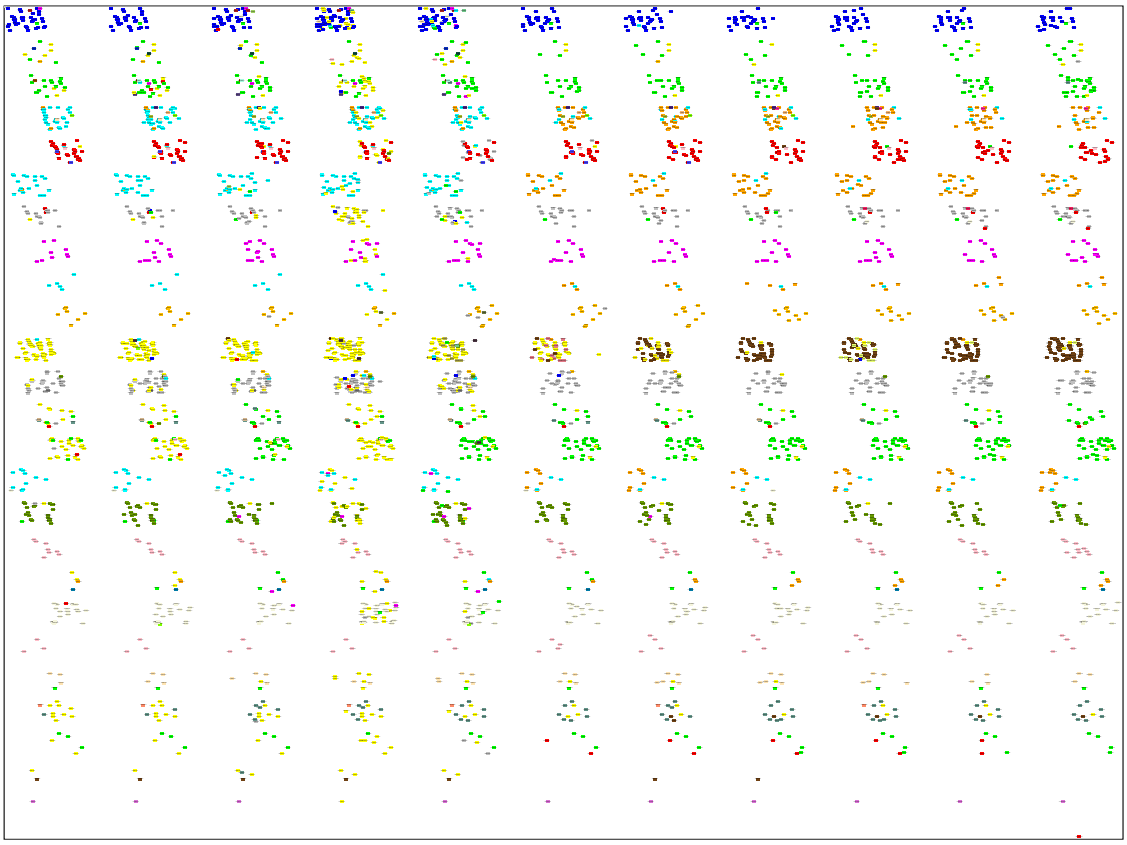


Abbildung 50: *Zeitexpandierter Graph mit Reichweite* $\mathrm{k} = 3$*, Schwelle* $\mathrm{p} = 0$ *und Interzeitkantengewicht* $\alpha = 5$*. Die Clusterung entspricht der Significance-Clusterung des Graphen. Jede Farbe steht für einen Cluster. Der* $\mathrm{bestmatch}_{\mathrm{no}}$ *bezüglich der Referenz-Clusterung beträgt* $0,46$*.*

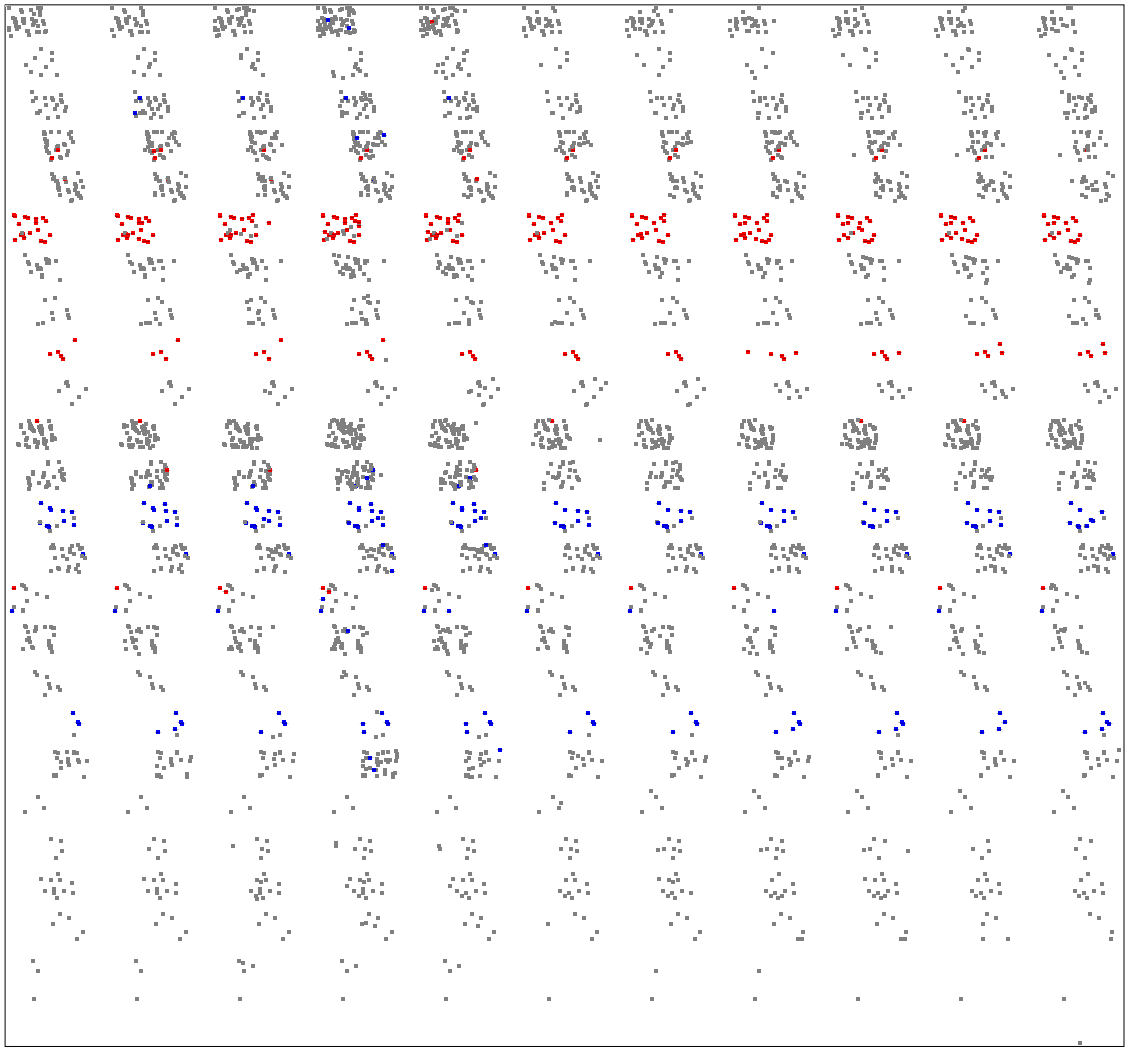


Abbildung 51: *Ergebnis der Clusterung eines zeitexpandierten Graphen der ersten Testreihe mit den Parametern* $\mathrm{k} = 9$, $\alpha = 6$ *und* $\mathrm{p} = 0$ *mit dem Iterative-Conductance-Cutting. Dabei wurden Lehrstühle die häufig gemeinsam geclustert wurden farblich hervorgehoben, während die restlichen Knoten grau gefärbt sind. Der rote Cluster enthält die Lehrstühle 5 und 8. Im blauen Cluster sind die Lerhstühle 12 und 17 enthalten.*

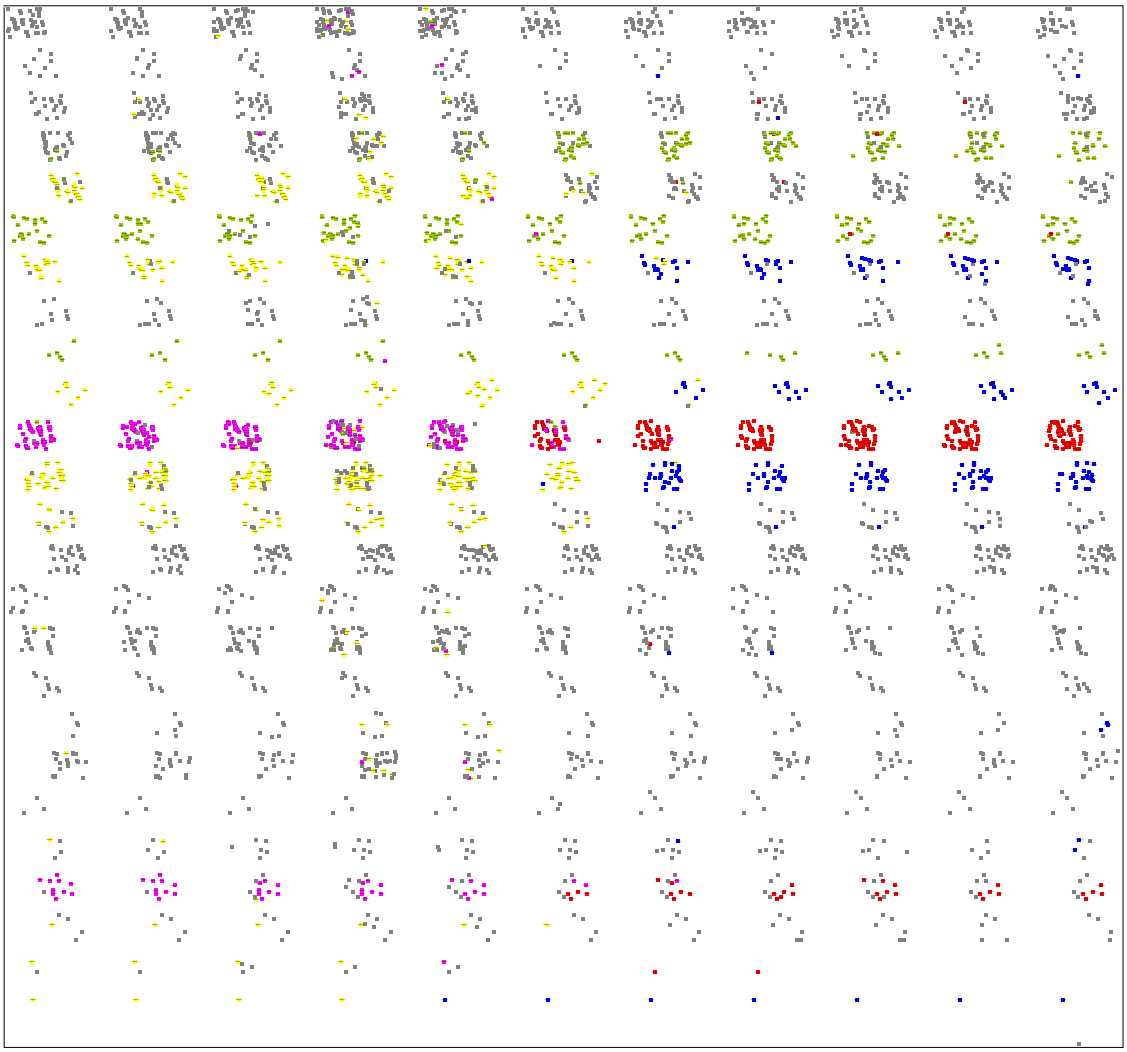
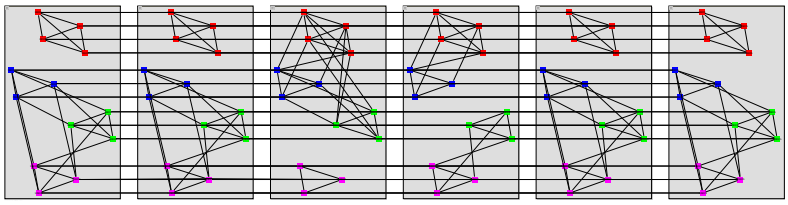
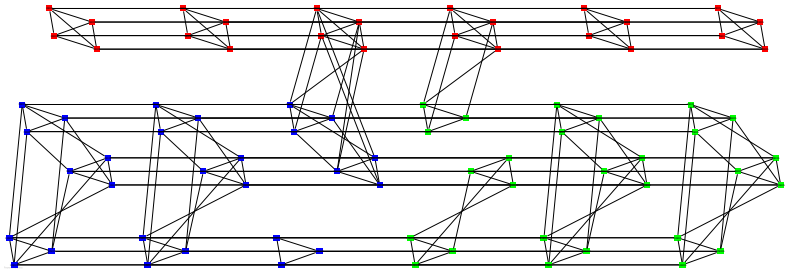


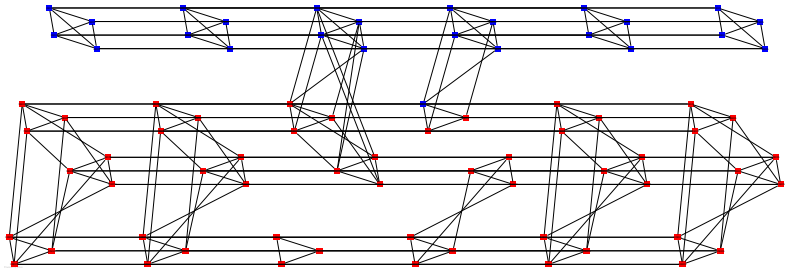
Abbildung 52: *Ergebnis der Clusterung eines zeitexpandierten Graphen mit der Methode Cosine-Time mit den Parametern* $\mathrm{k} = 1$ *und* $\mathrm{p} = 0$ *mit dem Greedy-Significance-Clustering. Dabei wurden Lehrstühle die häufig gemeinsam geclustert wurden farblich hervorgehoben, während die restlichen Knoten grau gefärbt sind. Siehe dazu Abschnitt 7.1.*



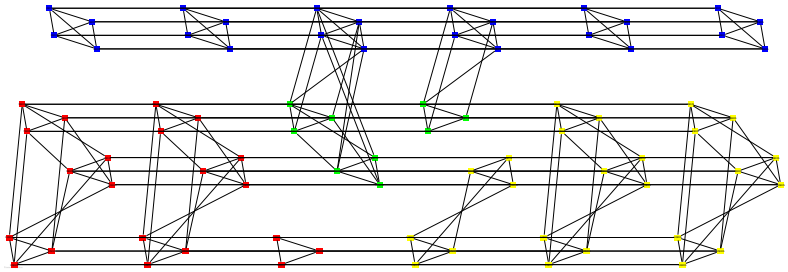
(a) Hier ist der Ausgangsgraph abgebildet. Die Farben stehen für die vorhandenen Cliquen und die Kästen für die einzelnen Zeitschritte.



(b) Das Ergebnis der Clusterung mit dem ICC-Verfahren mit $\mathrm{a}^* = 0,17$. Die Farben stehen für die verschiedenen Cluster. Die Clusterung mit dem ICC-Verfahren und $\mathrm{a}^* = 0,11$ ähnelt der MCL-Clusterung.



(c) Das Ergebnis der Clusterung mit dem MCL-Verfahren mit $\mathrm{e} = 3$ und $\mathrm{r} = 2$. Die Farben stehen für die verschiedenen Cluster.



(d) Das Ergebnis der Clusterung mit dem Significance-Verfahren. Die Farben stehen für die verschiedenen Cluster. Zu Beginn (Zeitschritte 0 und 1) und am Ende (Zeitschritte 4 und 5) bilden die drei 3-Cliquen eine Gruppe. In Zeitschritt 2 und 3 haben sie nur wenige Verbindungen untereinander, dafür teilweise starke Verbindungen zu der 4-Clique. Diese Verbindungen führen zu dem grünen Cluster, der die Repräsentanten der drei 3-Cliquen enthält die Verbindungen zu der 4-Clique haben.

Abbildung 53: *Die Abbildungen zeigen die Clusterungen des zeitexpandierten Graphen mit der Methode Cosine-Time und Reichweite* $\mathrm{k} = 2$ *für unser Beispiel aus 3.4.3, die kurzfristige Veränderung des Verhaltens einer Gruppe. Durch die erhöhte Reichweite fällt die kurzfristige Veränderung bei den Clusterungen nicht so stark ins Gewicht. Interessant ist vor allem die Significance-Clusterung, die die zeitlichen Entwicklung der Gruppen treffend aufzeigt.*

A.2. Tabellen

k in Abhängigkeit von d	$\mathrm{k} \geq \mathrm{d}-1$	$\mathrm{k} < \mathrm{d}-1$
Maximale Anzahl der Interzeitkanten für Variante 1	$\mathrm{nd}\,(\mathrm{d}-1)$	$\frac{1}{2}\left(2\mathrm{ndk} - \left(\mathrm{k}^2+\mathrm{k}\right)\right)$

Tabelle 17: *Wir gehen hier von einem zeitexpandierten Graphen mit* d *Zeitschritten aus. Weiter sei* k *die Reichweite und* $\mathcal{V}$ *die Knotenmenge des zugrundeliegenden dynamischen Graphen mit* $|\mathcal{V}| = \mathrm{n}$.

Index	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	$\mathrm{bestmatch_{no}}$ Institute
1. Quartil	0,7450	0,4889	0,5688	0,1612
3. Quartil	0,9764	0,7535	0,7445	0,2918
Minimum	0,6639	0,1339	0,3628	0,0616
Maximum	0,9969	0,7888	0,9893	0,4700
Mittelwert	0,8128	0,6300	0,6814	0,2244

Tabelle 18: *Indizes für die ICC-Clusterungen der 108 zeitexpandierten Graphen der Normal-Testreihe mit* $\mathrm{a}^* = 0,20$.

Index	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	$\mathrm{bestmatch_{no}}$ Institute
1. Quartil	0,8823	0,7912	0,6842	0,2142
3. Quartil	0,9257	0,8584	0,8332	0,6301
Minimum	0,7910	0,1339	0,4745	0,0618
Maximum	0,9969	0,8832	0,9877	0,7306
Mittelwert	0,9099	0,7310	0,7698	0,4456

Tabelle 19: *Indizes für die ICC-Clusterungen der 108 zeitexpandierten Graphen der Normal-Testreihe mit* $\mathrm{a}^* = 0,10$.

Index	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	$\mathrm{bestmatch_{no}}$ Institute
1. Quartil	0,9112	0,4844	0,7548	0,1656
3. Quartil	0,9824	0,8722	0,8915	0,6695
Minimum	0,8683	0,1472	0,4982	0,0618
Maximum	0,9970	0,8886	0,9894	0,7557
Mittelwert	0,9358	0,7074	0,8137	0,4571

Tabelle 20: *Indizes für die ICC-Clusterungen der 108 zeitexpandierten Graphen der Normal-Testreihe mit* $\mathrm{a}^* = 0,075$.

Index	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	$\mathrm{bestmatch_{no}}$ Institute
1. Quartil	0,9349	0,4823	0,8012	0,1636
3. Quartil	0,9869	0,8724	0,9344	0,6900
Minimum	0,9178	0,1369	0,5171	0,0615
Maximum	0,9970	0,9020	0,9911	0,7816
Mittelwert	0,9589	0,6683	0,8515	0,4082

Tabelle 21: *Indizes für die ICC-Clusterungen der 108 zeitexpandierten Graphen der Normal-Testreihe mit* $\mathrm{a}^* = 0,05$.

Index	$\mathrm{cov_w}$	$\mathrm{mod_w}$	δ_d	$\mathrm{bestmatch_{no}}$ Institute
1. Quartil	0,9386	0,0003	0,2791	0,3504
3. Quartil	1,0000	0,8710	0,9475	0,4762
Minimum	0,9338	0,0000	0,0000	0,1917
Maximum	1,0000	0,8851	1,0000	0,4826
Mittelwert	0,9724	0,4514	0,6470	0,3803

Tabelle 22: *Indizes für die MCL-Clusterungen der 108 zeitexpandierten Graphen der Normal-Testreihe mit* $\mathrm{e} = 5$, $\mathrm{r} = 1,5$ *und* $\kappa = 150$.

Schwelle	$\mathcal{NVD}(\mathcal{C},\mathcal{C}')$	$\mathrm{bestmatch_{no}}(\mathcal{C},\mathcal{C}')$	$\mathcal{NVI}(\mathcal{C},\mathcal{C}')$
$0,00 \leq \mathrm{p} \leq 0,05$	0,1995	0,5901	0,1584
$0,10 \leq \mathrm{p} \leq 0,15$	0,1957	0,5989	0,1567
$0,20 \leq \mathrm{p} \leq 0,25$	0,1949	0,6100	0,1562
$0,30 \leq \mathrm{p} \leq 0,35$	0,1757	0,6570	0,1252
$0,40 \leq \mathrm{p} \leq 0,45$	0,1284	0,7401	0,0792

Tabelle 23: *Die Vergleichsmaße van Dongen* $\mathcal{NVD}$*, Variation of Information* $\mathcal{NVI}$ *und* $\mathrm{bestmatch_{no}}$ *für die expandierten Clusterungen* $\mathcal{C}$ *und die Significance-Clusterungen* $\mathcal{C}'$ *der 550 Teilgraphen (siehe 7.2) in Abhängigkeit der Schwelle.*

Danksagung

Ich möchte mich bei Robert Görke für die gute Betreuung und die nützlichen Ratschläge bedanken. Außerdem bedanke ich mich bei meiner Familie für ihre jahrelange Unterstützung während meines Studiums.

Eidesstattliche Erklärung

Der Autor versichert, die vorliegende Arbeit selbständig erstellt zu haben. Er versichert weiterhin, nur die aufgeführten Hilfsmittel benutzt zu haben. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Karlsruhe, am 8. Februar 2008

Dieter Glaser