

Rollup-Verfahren für komplexe dynamische Bayessche Netze

Diplomarbeit
Lehrstuhl Prof. Dr. Dr. h.c. W. Wahlster
Fakultät 6 – Informatik
Universität des Saarlandes

Boris Brandherm

Eidesstattliche Erklärung

Hiermit versichere ich an Eides Statt, daß ich diese Diplomarbeit nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.

Saarbrücken, den 5. September 2000

Vorwort

Bayessche Netze stellen einen Formalismus zur Verarbeitung und Repräsentation von unsicherem Wissen dar. Mit ihrer Hilfe lassen sich Domänen modellieren, in denen mit Unsicherheit behaftete Schlußfolgerungen gezogen werden müssen.

Für das Lösen (Evaluieren) von Bayesschen Netzen wurden exakte und näherungsweise Verfahren (Inferenzalgorithmen) entwickelt. In beiden Fällen ist die Berechnungskomplexität von großer Bedeutung (Das exakte oder auch näherungsweise Lösen von Bayesschen Netzen ist *NP-vollständig*).

Normalerweise beschreiben Bayessche Netze Kausalzusammenhänge, die entweder zeitunabhängig sind oder zu einem bestimmten Zeitpunkt gelten.

Dynamische Bayessche Netze sind eine Erweiterung von Bayesschen Netzen und ermöglichen die Modellierung von zeitabhängigen Prozessen, indem über die Zeit sogenannte *Zeitscheiben* (Das sind Bayessche Netze, die durch Kanten mit dem alten Netz verbunden werden.) an das bestehende Netz angehängt werden können. Dabei wächst das Netz, und für die Evaluation des Netzes werden die Laufzeiten immer länger und der Arbeitsspeicherbedarf immer umfangreicher. Damit die Evaluation eines Dynamischen Bayesschen Netzes in einem zeitlichen Rahmen bleibt und auch der Arbeitsspeicherbedarf nicht über die vorhandenen Ressourcen steigt, müssen alte Zeitscheiben abgehängt werden, was auch als *Rollup* bezeichnet wird.

Problemstellung

Das Problem des Rollups stellte sich konkret bei der Behandlung der Dynamischen Bayesschen Netze im READY-System¹. Für das Lösen der READY-Netze wird das Programm Netica™ [Net 96; Net 95] verwendet, das das Programm IDEAL² [Srinivas & Breese 92] ablöste.

Das Programm Netica™ war auch dazu vorgesehen, zur Lösung der Problemstellung beizutragen. In der Aufgabenstellung des Problems wurde zuerst davon ausgegangen, daß beim Rollup grundsätzlich ein *Informationsverlust* auftritt. Es war deshalb wichtig festzustellen, ab wann (d.h. wieviele Zeitscheiben vor der gegenwärtigen Zeitscheibe) der Informationsverlust durch den Rollup vernachlässigbar ist. Für die Berechnung des Informationsverlustes war es notwendig, die Knoten zu identifizieren, die die Hauptursache für den Informationsverlust sind. Mit der Einschätzung des Informationsverlustes

¹Die Dynamischen Bayesschen Netze im READY-System haben gegenüber Standard Bayesschen Netzen einige Besonderheiten, so daß im weiteren von READY-Netzen die Rede ist, wenn die Dynamischen Bayesschen Netze des READY-Systems gemeint sind.

²IDEAL ist das Akronym für Influence Diagram Evaluation and Analysis in Lisp.

und der Betrachtung der eigenen Ressourcen wie Arbeitsspeicher und Rechenzeit sollte vom System selbständig entschieden werden, wann ein Rollup durchzuführen ist.

Dazu muß das System Rechenzeit und Speicherbedarf vorhersagen, bevor sie wirklich auftreten. Hierbei entstand das Problem, daß man auf die interne Datenstruktur (den sogenannten *Junction Tree*) zugreifen muß. Die vorhandenen Hilfsmittel wie z.B. Netica™ erlauben jedoch keinen Zugriff von außen auf den Junction Tree.

Für die Evaluation von Bayesschen Netzen gibt es mehrere Verfahren, die auf einem Junction Tree arbeiten. Wie sich jedoch herausstellte, sind nur einige Verfahren davon geeignet, Dynamische Bayessche Netze effizient zu unterstützen.

So erweiterte sich die Aufgabenstellung um die Implementierung eines geeigneten Inferenzalgorithmus, der dem Benutzer einen Zugriff auf die internen Strukturen ermöglicht, und dabei die Dynamischen Bayesschen Netze effizient unterstützt. Zusammen mit der Implementierung des Inferenzalgorithmus und den Forschungsergebnissen von [Kjærulff 95] wurde klar, daß ein Rollup von Dynamischen Bayesschen Netzen ohne Informationsverlust machbar ist. Diese Verfahren konnten nicht ohne weiteres auf komplexe Netze wie die von READY angewendet werden.

In der vorliegenden Arbeit wird nun gezeigt, wie sich auch bei komplexeren Dynamischen Bayesschen Netzen ein Rollup ohne Informationsverlust durchführen läßt. Zugleich wird ein Rollup-Verfahren vorgestellt, daß nicht wie üblich nur den Rollup der ältesten Zeitscheibe zuläßt, sondern den Rollup jeder beliebigen Zeitscheibe ermöglicht.

Die Rollup-Verfahren setzen einige Bedingungen an das Dynamischen Bayesschen Netz voraus, das aufgerollt werden soll. Diese Bedingungen werden von den READY-Netzen nicht erfüllt. Es wird gezeigt, wie sich durch Umformungen der READY-Netze die Probleme beseitigen lassen.

Aufbau der Arbeit

Die Rollup-Verfahren, die in der vorliegenden Arbeit vorgestellt bzw. neu entwickelt werden, arbeiten auf dem sogenannten *Junction Tree* des Dynamischen Bayesschen Netzes oder nehmen direkte Veränderungen an den Wahrscheinlichkeitstabellen der Knoten und an der Struktur des Dynamischen Bayesschen Netzes vor.

Im **ersten Kapitel** wird nach der Motivation und Begriffsbildung für mehrfach verbundene Bayessche Netze an zwei Beispielnetzen die Bestimmung eines Junction Trees anschaulich erklärt. Dabei werden elementare Begriffe wie z.B. *Moralisierung*, *Eliminationsreihenfolge*, *Triangulierung*, *Junction Graph*, *Junction Tree Eigenschaft*, *Junction Tree*, *Knotenzuordnung* und ψ -*Tabellen* eingeführt und definiert. Eine Algebra von Wahrscheinlichkeitstabellen vereint die a-priori und bedingten Wahrscheinlichkeiten sowie die Belegung von Knoten mit Evidenzen in einer Schreibweise. Unter konsequenter Benutzung dieser Schreibweise werden mehrere exakte Inferenzverfahren vorgestellt, die auf dem Junction Tree arbeiten und von den Rollup-Verfahren in dieser Arbeit benötigt werden.

Im **zweiten Kapitel** werden die Dynamischen Bayesschen Netze auf der Grundlage der Bayesschen Netze eingeführt. Es wird gezeigt, wie sich die Schemata und die Zeitscheiben des Dynamischen Bayesschen Netzes effektiv und ressourcenschonend verwalten lassen. Für die Behandlung der Dynamischen Bayesschen Netze werden zwei verschiedene Verfahren systematisch miteinander verglichen. Mit einem der beiden

Verfahren werden dabei schon die Grundlagen für einige Rollup-Verfahren geschaffen. Ebenso wird auch der für die Dynamischen Bayesschen Netze und den Rollup wichtige Begriff der *eingeschränkten Eliminationsreihenfolge* eingeführt, der den Begriff der Eliminationsreihenfolge für Bayessche Netze erweitert.

Das **dritte Kapitel** ist ganz dem Rollup für Dynamische Bayessche Netze gewidmet. Es werden verschiedene exakte Verfahren vorgestellt, die den Rollup, außer beim “Approximativen Rollup”, ohne Informationsverlust durchführen. Als ein weiterer wichtiger Begriff wird das sogenannte *D-Separations-Kriterium* erwähnt. In einer großen Übersicht werden die Rollup-Verfahren abschließend gegenüber gestellt, ihre Vor- und Nachteile aufgezählt und ihre Anwendbarkeit kategorisiert, anhand derer ein Anwender sich für sein Problem ein richtiges Rollup-Verfahren aussuchen kann.

Im anschließenden **vierten Kapitel** wird gezeigt, wie die READY-Netze angepaßt werden können, damit auf sie ein Rollup angewendet werden kann. Dazu werden die verschiedenen Besonderheiten der READY-Netze sowohl an einem generischen Schema als auch an einem konkreten Beispielnetz einzeln erläutert. An diesem Schema und dem konkreten Beispielnetz wird dann auch gezeigt, warum die jeweilige Besonderheit der READY-Netze einen Rollup verhindert und wie man diese Eigenschaften umformen kann, so daß sich ein Rollup-Verfahren fehlerfrei anwenden läßt.

Zur Erhöhung der Lesbarkeit wurden technische Ausführungen in **Anhänge** ausgelagert. Als Anhang werden ebenfalls mehrere Programmdurchläufe vom Programm *RUBIN* beigefügt. Dieses Programm wurde entwickelt, um die theoretischen Aspekte beim Rollup nachvollziehen zu können.

Danksagung

Mein erster Dank gilt Herrn Prof. Dr. Dr. h.c. Wolfgang Wahlster, der mir die Gelegenheit zur Anfertigung dieser Diplomarbeit gab. Desweiteren möchte ich mich bei meinen beiden Betreuern Dr. Anthony Jameson und Dr. Ralph Schäfer für zahlreiche nützliche Anmerkungen und Kommentare sowie für die Hilfe bei der Konzeption und Gliederung der Arbeit bedanken.

Für unzählige Korrekturen und viele wertvolle Hinweise zur Arbeit bedanke ich mich bei Barbara Großmann-Hutter, Frank Wittig, Dirk Wagner und Thomas Weis.

Den Lehrstuhlbewohnern des Lehrstuhles von Prof. Wahlster möchte ich für das angenehme Arbeitsumfeld danken, auch wenn ich für die schriftliche Ausarbeitung der Diplomarbeit die Ruhe zuhause vorgezogen habe.

Mein abschließender Dank gilt meinen Eltern Irmgard und Bernhard Brandherm für die moralische und finanzielle Unterstützung des Studiums und meinen Geschwistern Marthe, Arno und Uwe Brandherm und meinem Schwager Ralf Gampfer. Jeder von ihnen weiß warum.

Übersicht der Fachbegriffe

Zur Zusammenfassung sind im folgenden die einzelnen Kapitel mit Überschrift und den Fachbegriffen aufgeführt, die dort vorkommen:

Kapitel 1 Mehrfach verbundene Bayessche Netze

Notation und Lösungsalgorithmen für Bayessche Netze, unsicheres Wissen, Abhängigkeiten, Wahrscheinlichkeiten, einfach verbundenes Bayessches Netz, mehrfach verbundenes Bayessches Netz, gerichteter Graph, Kausalzusammenhang, Proposition, Hypothese, Modell, Domäne, a-priori- und bedingte Wahrscheinlichkeiten, Knoten, (diskrete) Zufallsvariable, Elementarereignis, Wertebereich, Wurzelknoten, Graph \mathcal{G} , Knotenmenge \mathbf{V} , Kantenmenge \mathbf{E} , Quelle der Kante, Ziel der Kante, Elternknoten, Kinderknoten, Familie eines Knotens, benachbart (adjacent), Zyklus, gerichteter azyklischer Graph (DAG), $\text{pa}(V)$, $\text{ch}(V)$, $\text{fa}(V)$, $\text{adj}(V)$, gerichtete und ungerichtete Kante, Instantiierung eines Knotens, Lösungsverfahren, exakte und approximative Algorithmen, Junction Tree Verfahren, Moralisierung, moralisierter Graph \mathcal{G}^m , Eliminationsreihenfolge, triangulierter Graph \mathcal{G}^t , Junction Graph \mathcal{J} , Junction Tree \mathcal{T} , Wurzelknoten, Megaknoten, maximum cardinality search, Löschen, Eliminieren, Teilgraph, Junction Tree Eigenschaft, Gewicht einer Clique, Ordnung des Graphens, chord, (redundanter) fill-in, minimale Triangulierung, perfekte Eliminationsreihenfolge, (maximaler) Cluster, Clique, Separator, Knotenzuordnung, ψ -Tabelle, Algebra von Wahrscheinlichkeitentabellen, Knotenmenge, Konfiguration, Tabelle bedingter Wahrscheinlichkeiten, Tabelle gemeinsamer Wahrscheinlichkeiten, Normalisierungskonstante α , finding eines Knotens, $T_{\text{finding}(\cdot)}$, Multiplikation und Division von Tabellen, Marginalisation von Tabellen, Inferenzverfahren, Collect und Distribute Evidence, Initialisierung, Auffrischung, aufrufender Knoten, Absorption, BEL-Wert, Aalborg Architektur, Shafer-Shenoy Architektur, Shafer-Shenoy Skripte, Polynom-Propagierung, multivariates Polynom, Hypothesenparametrisierung, Query-DAG, Evidenz- und Query-Variable, Q-DAG Evaluator

Kapitel 2 Dynamische Bayessche Netze

Dynamisches Bayessches Netz (DBN), stochastischer temporaler Prozeß, Zeitscheibe, Zustandentwicklungsmodell, Rollup, dynamischer Knoten, Schema, $\mathbf{V}(t', n)$, $\mathbf{V}(t)$, temporale Kante, relationale Kante, zeitliche Kante, Markov Eigenschaft, Interfaceknoten, Strukturhaltung, eingeschränkte Eliminationsreihenfolge, Satz von *Rose*, *Tarjan und Lueker*, geordneter Graph, vorberechnete Zeitscheibe

Kapitel 3 Rollup für Dynamische Bayessche Netze

Ressource, Aufspalten eines Knotens, Löschen eines Knotens, Rollup-Verfahren, Gesamtwahrscheinlichkeit, Knotenabsorption, Größe einer Wahrscheinlichkeitentabelle, $\#(T_{\text{cpt}(\cdot)})$, D-Separations-Kriterium, Vorgängerknoten, Nachfolgerknoten, Prediction-Estimation Verfahren, Zustandsraum, dHUGIN, Modell, \mathcal{P}_i , Markov-Eigenschaft, Zeitfenster (time window), Backward smoothing model, Forecasting model, Stochastische Simulation, Vergrößerung des Zeitfensters,

Verkleinerung des Zeitfensters, Backward smoothing slice, Forecast slices, Time window slices, Polynom-Rollup, Zeitscheibenorientiert, Knotenorientiert, (faktorierte) Shafer-Shenoy Skripte, approximativer Rollup, kanonische Menge der Zufallsvariablen, Partition von Zufallsvariablen, schwache Interaktion, spärliche Interaktion, Dialogsteuerung

Kapitel 4 READY-Netze

READY-Netze, Dialogsystem, statischer Knoten, temporärer Knoten, Doppelschema, Nachinstanziierung von Netzstrukturen, zeitscheibenüberspringende Kante, Zustandsentwicklungsmodell

Kapitel 5 Zusammenfassung und Ausblick

Anhang A Mehrfach verbundene Bayessche Netze

Aalborg Architektur, Collect Evidence, Distribute Evidence, Absorption, Initialisierung, Auffrischung, ψ -Tabelle, Marginalisation, Separator, Clique, $T_{\text{finding}(\cdot)}$, Shafer-Shenoy Architektur, Polynom-Propagierung, a-priori-Parametrisierung, CPT-Parametrisierung, Gaußscher Algorithmus, Sensitivitätsanalyse, D-Separations-Kriterium, blockieren, aktivieren

Anhang B Mathematik

Algebra von Wahrscheinlichkeitstabellen, Tabelle über einer Knotenmenge, Hypothese, mehrdimensionaler Vektor, Vektorelement, Konfiguration, Tabellenelement, Tabelle bedingter Wahrscheinlichkeiten (CPT), Tabelle gemeinsamer Wahrscheinlichkeiten (JPT), Wahrscheinlichkeitentabelle, Gleichheit von Tabellen, Multiplikation und Division von Tabellen, Marginalisation, Produkt-Marginalisation, multivariate Polynome, Polynom, Multiindizes, Polynomaddition, Polynommultiplikation, Polynomring, (teilweise bzw. totale) Ordnung

Anhang C Das Programm

RUBIN, mehrfach verbundene Bayessche Netze, Visualisierungsprogramm, Hypothesenparametrisierung, Definition eines Knotens, a-priori Parametrisierung, Collect und Distribute Evidence, Dynamisches Bayessches Netz, Zeitscheibe, temporale Kanten, pruning, Shafer-Shenoy Skript

Inhaltsverzeichnis

1	Mehrfach verbundene Bayessche Netze	1
1.1	Motivation	2
1.2	Struktur	5
1.3	Junction Tree Verfahren	9
1.3.1	Bestimmung des Junction Trees	10
1.3.2	Algebra von Wahrscheinlichkeitentabellen	22
1.3.3	Inferenzverfahren	23
1.3.3.1	Aalborg Architektur	25
1.3.3.2	Shafer-Shenoy Architektur	26
1.3.3.3	Shafer-Shenoy-Skripte	29
1.3.3.4	Polynom-Propagierung	34
1.4	Zusammenfassung	38
2	Dynamische Bayessche Netze	41
2.1	Motivation	42
2.2	Struktur	46
2.3	Zusammenfassung	58
3	Rollup für Dynamische Bayessche Netze	61
3.1	Motivation	61
3.2	Algorithmen	65
3.2.1	Knotenabsorption	66
3.2.2	Prediction-Estimation Verfahren	69
3.2.3	dHugin	72
3.2.4	Polynom-Rollup	74
3.2.5	Shafer-Shenoy Skripte	79
3.2.6	Approximativer Rollup	83
3.3	Vergleich der Algorithmen	86
4	READY-Netze	89
4.1	Problembehandlung	90
4.1.1	Zeitscheibenüberspringende Kanten	91
4.1.2	Statische Knoten	104
4.1.3	Nachinstantiierung von Netzstrukturen	112
4.2	Zusammenfassung	118
5	Zusammenfassung und Ausblick	121

A	Mehrfach verbundene Bayessche Netze	123
A.1	Aalborg Architektur	123
A.2	Shafer-Shenoy Architektur	130
A.3	Weitere Anwendungen der Polynom-Propagierung	136
A.4	D-Separations-Kriterium	141
B	Mathematik	145
B.1	Algebra von Wahrscheinlichkeitentabellen	145
B.2	Multivariate Polynome	150
B.3	Nebenrechnungen	152
C	RUBIN	159
C.1	Mehrfach verbundene Bayessche Netze	159
C.2	Dynamische Bayessche Netze	171

Abbildungsverzeichnis

1.1	Beispielnetz Gehirntumor	2
1.2	Beispielnetz Asienbesuch	4
1.3	Beispiele für moralisierte Graphen	12
1.4	Beispiele für gut triangulierte Graphen	15
1.5	Beispiele für schlecht triangulierte Graphen	17
1.6	Beispiele für Junction-Graphen bei guter Triangulierung	18
1.7	Beispiele für Junction-Graphen bei schlechter Triangulierung	18
1.8	Beispiele für Junction-Trees bei guter Triangulierung	20
1.9	Beispiele für Junction-Trees bei schlechter Triangulierung	20
1.10	Die beiden Phasen Collect und Distribute Evidence	24
1.11	Aalborg: Absorption der Clique Clq_ℓ von der Clique Clq_i	26
1.12	Shafer-Shenoy: Berechnung der Cliqueninfo	27
1.13	Shafer-Shenoy: Absorption der Clique Clq_ℓ von der Clique Clq_i	28
1.14	Beispiel für einen Query-DAG	37
1.15	Generierung von Query-DAGs	38
2.1	Beispielnetz Prüfungsfrage statisch (statisches Bayessches Netz)	43
2.2	Beispielnetz Prüfungsfrage dynamisch (dynamisches Bayessches Netz)	44
2.3	DBN: Die Zeitscheiben sind Instanzen der Schemata für das DBN.	47
2.4	DBN: Fünf Zeitscheiben eines DBNs mit vier Schemata.	49
2.5	DBN: Temporale Kanten und Dynamische Knoten eines DBNs	49
2.6	DBN: Moralisierungskanten und Interfaceknoten eines DBNs	50
2.7	DBN: Triangulationskanten (Zeitscheiben unberücksichtigt)	51
2.8	DBN: Junction Tree (Zeitscheiben unberücksichtigt)	52
2.9	DBN: Triangulationskanten (Zeitscheiben berücksichtigt)	55
2.10	DBN: Junction Tree (Zeitscheiben berücksichtigt)	56
2.11	DBN: Anhängen von Zeitscheibe 1 im Junction Tree	58
2.12	DBN: Anhängen von Zeitscheibe 3 im Junction Tree (Erkennen von Regelmäßigkeiten)	59
3.1	Beispielnetz Ampelsteuerung	62
3.2	Beispielnetz Zebrastreifen	64
3.3	Beispielnetz Straßenverkehr	65
3.4	Knotenabsorption: Ausgangsnetz	67
3.5	Knotenabsorption: Absorption eines uninstantiierten Knotens	68
3.6	Knotenabsorption: Absorption eines instantiierten Knotens	69
3.7	Die drei Schritte beim Prediction-Estimation Verfahren	71
3.8	dHUGIN: Einteilung des DBNs in eine Reihe von Modellen	73

3.9	Polynom-Rollup: Prüfungsfrage dynamisch	75
3.10	Rollup: Shafer-Shenoy Skripte, Junction Tree zu Beispielnetz Prüfungsfrage dynamisch	80
4.1	Zeitscheibenüberspringende Kanten (allgemein) ohne Rollup	92
4.2	Zeitscheibenüberspringende Kanten (allgemein) mit Rollup	93
4.3	Zeitscheibenüberspringende Kanten (allgemein) korrigiert	93
4.4	Beispiel: Zeitscheibenüberspringende Kanten ohne Korrektur	95
4.5	Beispiel: Zeitscheibenüberspringende Kanten ohne Korrektur beim Rollup	97
4.6	Beispiel: Zeitscheibenüberspringende Kanten mit Korrektur	98
4.7	Beispiel: Zeitscheibenüberspringende Kanten mit Korrektur beim Rollup	99
4.8	Beispiel: Triangulierung bei zeitscheibenüberspringenden Kanten ohne Korrektur	100
4.9	Beispiel: Triangulierung bei zeitscheibenüberspringenden Kanten mit Korrektur	101
4.10	Zeitscheibenüberspringende Kanten (Doppelschema) ohne Rollup	102
4.11	Zeitscheibenüberspringende Kanten (Doppelschema) mit Rollup	103
4.12	Zeitscheibenüberspringende Kanten (Doppelschema) korrigiert	104
4.13	Statische Knoten ohne Rollup	106
4.14	Statische Knoten in den Schemata eingebettet	107
4.15	Beispiel: Statischer Knoten ohne Korrektur	108
4.16	Beispiel: Statischer Knoten ohne Korrektur beim Rollup	109
4.17	Beispiel: Statischer Knoten mit Korrektur	110
4.18	Beispiel: Statischer Knoten mit Korrektur beim Rollup	111
4.19	Nachinstantiierung ohne Rollup	113
4.20	Nachinstantiierung mit Rollup	114
4.21	Nachinstantiierung korrigiert	115
4.22	Beispiel: Nachinstantiierung ohne Korrektur	116
4.23	Beispiel: Nachinstantiierung ohne Korrektur beim Rollup	117
4.24	Beispiel: Nachinstantiierung mit Korrektur	118
4.25	Beispiel: Nachinstantiierung mit Korrektur beim Rollup	119
A.1	Aalborg: Beispiel eines Junction Trees für Beispielnetz Asienbesuch	124
A.2	Aalborg: Initialisierung eines Junction Trees	125
A.3	Aalborg: Auffrischung eines Junction Trees bei neuer Evidenz	129
A.4	Shafer-Shenoy: Initialisierung eines Junction Trees	132
A.5	Shafer-Shenoy: Auffrischung eines Junction Trees bei neuer Evidenz	136
A.6	Der Beliefwert in Abhängigkeit von A_1 beschreibt eine Gerade	138
A.7	Der Beliefwert in Abhängigkeit von X_1 und Y_1 beschreibt eine Ebene	140
A.8	D-Separations-Kriterium: Veranschaulichung	141
A.9	D-Separations-Kriterium: Beispiel 1	142
A.10	D-Separations-Kriterium: Beispiel 2	143
C.1	Bildschirmabzug des Beispielnetzes Gehirntumor (DAG)	163
C.2	Bildschirmabzug des Beispielnetzes Gehirntumor (trianguliert)	164
C.3	Bildschirmabzug des Beispielnetzes Gehirntumor (Junction Tree)	165
C.4	Bildschirmabzug des Beispielnetzes Gehirntumor (Infofenster)	165
C.5	Bildschirmabzug des Dynamischen Bayesschen Netzes (DAG)	176

C.6	Bildschirmabzug des Dynamischen Bayesschen Netzes (trianguliert) . .	177
C.7	Bildschirmabzug des Dynamischen Bayesschen Netzes (Junction Tree) .	178

Tabellenverzeichnis

1.1	Wahrscheinlichkeiten für das Beispielnetz Gehirntumor	3
1.2	Wahrscheinlichkeiten für das Beispielnetz Asienbesuch	5
1.3	Triangulierungsprozeß mit Cliques des Beispielnetzes Asienbesuch . . .	21
1.4	Shafer-Shenoy Skripte: Tabellen bei der Initialisierung	30
1.5	Shafer-Shenoy Skripte: Tabellen bei der Auffrischung	31
2.1	Wahrscheinlichkeiten für das Beispielnetz Prüfungsfrage statisch	43
2.2	Einschätzung des Wissensniveaus (Prüfungsfrage statisch)	44
2.3	Wahrscheinlichkeiten für das Beispielnetz Prüfungsfrage dynamisch . . .	45
2.4	Einschätzung des Wissensniveaus (Prüfungsfrage dynamisch)	45
2.5	Vergleich von Prüfungsfrage statisch mit Prüfungsfrage dynamisch	46
2.6	DBN: Knoteneliminierung (Zeitscheiben unberücksichtigt)	53
2.7	DBN: Knoteneliminierung (Zeitscheiben berücksichtigt)	57
3.1	Wahrscheinlichkeiten für das Beispielnetz Ampelsteuerung	63
3.2	Wahrscheinlichkeiten für das Beispielnetz Zebrastrreifen	63
3.3	Wahrscheinlichkeiten für das Beispielnetz Straßenverkehr	64
3.4	Vergleich der Rollup-Verfahren	87
A.1	Aalborg: ψ -Tabellen der Cliques nach der Konstruktion	124
A.2	Aalborg: ψ -Tabellen bei der Initialisierung	126
A.3	Aalborg: ψ -Tabellen bei der Auffrischung	127
A.4	Shafer-Shenoy: ψ -Tabellen der Cliques nach der Konstruktion	131
A.5	Shafer-Shenoy: ψ -Tabellen bei der Initialisierung	134
A.6	Shafer-Shenoy: ψ -Tabellen bei der Auffrischung	135

Kapitel 1

Mehrfach verbundene Bayessche Netze

Mehrfach verbundene Bayessche Netze sind eine spezielle Art von Bayesschen Netzen. Mit Bayesschen Netzen läßt sich unsicheres Wissen modellieren und handhaben. Die in den Bayesschen Netzen modellierten Abhängigkeiten zwischen Ereignissen sind dabei mit Wahrscheinlichkeiten behaftet. Für die Handhabung des unsicheren Wissens in Bayesschen Netzen existieren verschiedene Lösungsverfahren.

In diesem Kapitel wird zunächst anhand von zwei kleinen Beispielnetzen der Einsatz von Bayesschen Netzen motiviert und anschließend formal die Begriffe und Schreibweisen für mehrfach verbundene Bayessche Netze eingeführt. Daraufhin werden verschiedene Lösungsverfahren für mehrfach verbundene Bayessche Netze vorgestellt. Diese Lösungsverfahren werden in einem späteren Kapitel so angepaßt oder weiterentwickelt, daß ein sogenannter *Rollup von Dynamischen Bayesschen Netzen* (siehe Kapitel 3) ausgeführt werden kann.

Bayessche Netze lassen sich in *einfach* und *mehrfach verbundene* Bayessche Netze einteilen. Bei *einfach verbundenen* Bayesschen Netzen gibt es zwischen jedem Paar von Knoten jeweils nur einen Pfad. Existiert zwischen wenigstens einem Paar von Knoten mehrere Pfade, so ist es ein *mehrfach verbundenes* Bayessches Netz. Da sich die einfach verbundenen Bayesschen Netze auch mit den Verfahren für die mehrfach verbundenen Bayesschen Netze lösen lassen, werden die einfach verbundenen Bayesschen Netze in dieser Arbeit nicht weiter betrachtet. Dem geneigten Leser für einfach verbundene Bayessche Netze sei [Pearl 88] als Literatur empfohlen, in dem unter anderem auch einfach verbundene Bayessche Netze besprochen werden.

Nach einer informellen Einführung von mehrfach verbundenen Bayesschen Netzen und der notwendigen Begriffsbildung werden verschiedene Verfahren zum Lösen (Evaluieren) von mehrfach verbundenen Bayesschen Netzen vorgestellt. Es sind exakte Lösungsverfahren, die zur Kategorie der Junction-Tree-Verfahren zählen. Diese Verfahren berechnen auf der Struktur des Bayesschen Netzes (der gerichtete Graph) eine sekundäre Struktur, den *Junction Tree*, auf dem dann die Berechnungen¹ durch-

¹Die Berechnungen auf dem Junction Tree werden auch als *Inferenz* oder *Propagierung* bezeichnet.

geführt werden können². Die Bestimmung eines Junction Trees wird Schritt für Schritt präsentiert. Jeder Schritt wird dabei mit der notwendigen Begriffsbildung sowie weiterführendem Hintergrundwissen ebenso allgemeingültig wie auch an zwei kleinen Beispielnetzen dargestellt.

Nachdem der Junction Tree konstruiert ist, müssen die quantitativen Zusammenhänge aus dem Bayesschen Netz in den Junction Tree konsistent übertragen werden. Dazu wird eine Schreibweise (Notation) eingeführt, die die mathematischen Gesichtspunkte der quantitativen Zusammenhänge außer acht läßt³. Dadurch kann man sich auf die wesentlichen Gesichtspunkte der Berechnungen konzentrieren, die bei der Inferenz im Junction Tree durchgeführt werden. Es sind dies die Addition, die Multiplikation und die Division.

Anhand von zwei Beispielnetzen werden nun mehrfach verbundene Bayessche Netze vorgestellt und dann die entsprechende Notation eingeführt.

1.1 Motivation

Bayessche Netze dienen der Repräsentation von unsicherem Wissen. In solchen Netzen können kausale Zusammenhänge zwischen einzelnen Propositionen, sogenannten *Hypothesen*, dargestellt werden. Dieses wird in den beiden folgenden Beispielen veranschaulicht. Im Anschluß daran folgen die formalen Definitionen für Bayessche Netze.

Das nun folgende Beispiel 1.1.1 wurde aus [Pearl 88] entnommen (siehe dazu auch [Spiegelhalter 86], [Neapolitan 90] und [Cooper 84]).

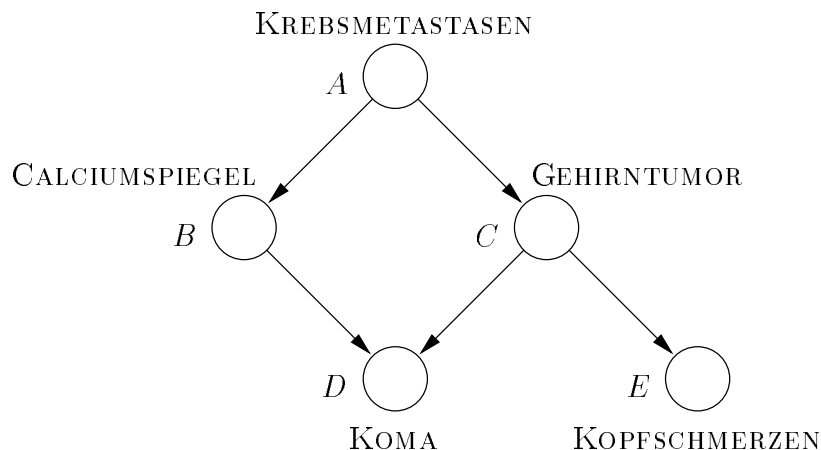


Abbildung 1.1: Beispielnetz *Gehirntumor* [$P(a, b, c, d, e) = P(a) \cdot P(b | a) \cdot P(c | a) \cdot P(d | b, c) \cdot P(e | c)$]

²In [Neapolitan 90; Pearl 88] werden auch approximative oder andere exakte Lösungsverfahren vorgestellt, auf die hier nicht eingegangen wird. Zu den approximativen Verfahren gehört zum Beispiel *stochastic simulation*. Andere exakte Verfahren sind beispielsweise *clustering* und *conditioning*.

³Es ist zum Beispiel zwischen bedingten Wahrscheinlichkeiten, a-priori Wahrscheinlichkeiten, und Instantiierungen von Knoten zu unterscheiden.

Beispiel 1.1.1 (Gehirntumor)

Dieses Beispiel ist ein einfaches Modell (einfaches Bayessches Netz) zur Darstellung des Sachverhaltes (der *Domäne*) ‘‘Gehirntumor’’, wobei nur die Zustände des Patienten bezüglich Krebsmetastasen, Gehirntumor, Calciumspiegel, Koma und Kopfschmerzen berucksichtigt werden können. Mißt ein behandelnder Arzt zum Beispiel Fieber, so kann es in diesem Modell nicht behandelt werden.

Es seien nun die folgenden kausalen Zusammenhänge bekannt: Krebsmetastasen sind die Ursache eines Gehirntumors und können auch den Calciumspiegel erhöhen. Ein Gehirntumor oder ein erhöhter Calciumspiegel können dazu führen, daß ein Patient ins Koma fällt. Ein Gehirntumor kann wiederum starke Kopfschmerzen verursachen. Andere direkte Zusammenhänge seien unbekannt. Die kausalen Zusammenhänge der Domäne werden als gerichteter Graph für das Bayessche Netz (siehe Abbildung 1.1) modelliert.

Jeder Knoten des Netzes symbolisiert eine diskrete Zufallsvariable, die als Wertemenge zwei Hypothesen besitzt, die zwei Ereignisse repräsentieren, wobei das eine Ereignis genau dann eintritt, wenn das andere Ereignis ausgeschlossen wird.

KREBSMETASTASEN ($A: a_1 = \text{nein}, a_2 = \text{ja}$)		
$P(a) :$	$P(a_1) = 0.80$	$P(a_2) = 0.20$
CALCIUMSPIEGEL ($B: b_1 = \text{normal}, b_2 = \text{erhöht}$)		
$P(b a) :$	$P(b_1 a_1) = 0.80$	$P(b_2 a_1) = 0.20$
	$P(b_1 a_2) = 0.20$	$P(b_2 a_2) = 0.80$
GEHIRNTUMOR ($C: c_1 = \text{nein}, c_2 = \text{ja}$)		
$P(c a) :$	$P(c_1 a_1) = 0.95$	$P(c_2 a_1) = 0.05$
	$P(c_1 a_2) = 0.80$	$P(c_2 a_2) = 0.20$
KOMA ($D: d_1 = \text{nicht im Koma}, d_2 = \text{komatös}$)		
$P(d b, c) :$	$P(d_1 b_1, c_1) = 0.95$	$P(d_2 b_1, c_1) = 0.05$
	$P(d_1 b_1, c_2) = 0.20$	$P(d_2 b_1, c_2) = 0.80$
	$P(d_1 b_2, c_1) = 0.20$	$P(d_2 b_2, c_1) = 0.80$
	$P(d_1 b_2, c_2) = 0.20$	$P(d_2 b_2, c_2) = 0.80$
KOPFSCHMERZEN ($E: e_1 = \text{keine}, e_2 = \text{starke}$)		
$P(e c) :$	$P(e_1 c_1) = 0.40$	$P(e_2 c_1) = 0.60$
	$P(e_1 c_2) = 0.20$	$P(e_2 c_2) = 0.80$

Tabelle 1.1: bedingte Wahrscheinlichkeiten bzw. a-priori Wahrscheinlichkeiten für die Knoten des Beispielnetzes **Gehirntumor** in Abbildung 1.1

Der Knoten A symbolisiert die Zufallsvariable KREBSMETASTASEN, die die beiden Ereignisse ‘‘Der Patient hat keine Krebsmetastasen.’’ und ‘‘Der Patient hat Krebsmetastasen.’’ zusammenfaßt. Der Wertebereich von KREBSMETASTASEN ist $\{\text{nein}, \text{ja}\}$. Die Hypothese ‘‘nein’’ der Zufallsvariablen KREBSMETASTASEN steht für ‘‘Der Patient hat keine Krebsmetastasen.’’, und die Hypothese ‘‘ja’’ steht für ‘‘Der Patient hat Krebsmetastasen.’’. Die zugehörigen Hypothesen der Zufallsvariablen CALCIUMSPIEGEL sind ‘‘normal’’ und ‘‘erhöht’’. Sie repräsentieren die beiden Ereignisse ‘‘Der Patient hat normalen Calciumspiegel.’’ und ‘‘Der Patient hat erhöhten Calciumspiegel.’’. Der Knoten C symbolisiert die Zufallsvariable GEHIRNTUMOR mit der Hypothese ‘‘nein’’ für das Ereignis ‘‘Der Patient hat keinen Gehirntumor.’’ und der Hypothese ‘‘ja’’ für das Er-

eignis “Der Patient hat einen Gehirntumor.”. Dem Knoten D ist die Zufallsvariable KOMA mit den Hypothesen “komatös” und “nicht im Koma” zugeordnet. Dabei steht die Hypothese “komatös” für das Ereignis “Der Patient befindet sich in tiefer Bewußtlosigkeit.” und die Hypothese “nicht im Koma” für das Ereignis “Der Patient ist bei Bewußtsein.”. Die Hypothesen für den Knoten E , der das Ereignis KOPFSCHMERZEN repräsentiert, sind “keine” und “starke”.

Die Hypothesen h der einzelnen Knoten sind mit h_1 für das Eintreffen des ersten Zustandes bzw. mit h_2 für das Eintreffen des zweiten Zustandes beschriftet, wobei $h \in \{a, b, c, d, e\}$.

In der Tabelle 1.1 sind die Wahrscheinlichkeiten der jeweiligen Knoten des Beispiels in Abhängigkeit vom Zustand ihrer Elternknoten festgelegt⁴. Aus dieser Tabelle geht zum Beispiel hervor, daß mit einer Wahrscheinlichkeit von 80% keine Krebsmetastasen zu befürchten sind, oder auch, daß ein Patient ohne Krebsmetastasen mit einer Wahrscheinlichkeit von 20% einen erhöhten Calciumspiegel hat. Bsp

Beispiel 1.1.2 (Asienbesuch)

Dieses Beispiel (entnommen aus [Neapolitan 90]) ist um einige Knoten größer als das vorhergehende Beispiel. Es handelt sich wiederum um einen kleinen medizinischen (Laien-)Sachverhalt.

Atembeschwerden können gleichermaßen von Lungenkrebs oder Tuberkulose herühren. Ein positiver Röntgenbefund steht zu gleichen Teilen für Lungenkrebs oder Tuberkulose. Ein kürzlicher Asienbesuch verstärkt dabei die Wahrscheinlichkeit von Tuberkulose. Auch Bronchitis ist eine Ursache für Atembeschwerden, wie Rauchen für Lungenkrebs oder Bronchitis.

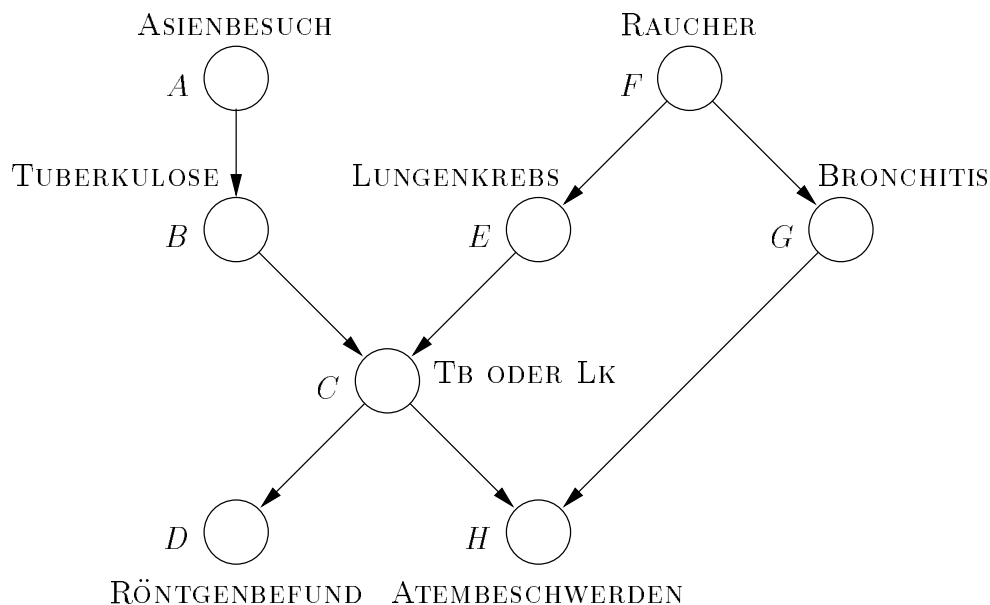


Abbildung 1.2: Beispielnetz *Asienbesuch* [$P(a, b, c, d, e, f, g, h) = P(a) \cdot P(b | a) \cdot P(c | b, e) \cdot P(d | c) \cdot P(e | f) \cdot P(f) \cdot P(g | h) \cdot P(h | c, g)$]

⁴Die Wahrscheinlichkeiten der Knoten mit Elternknoten werden als *bedingte Wahrscheinlichkeiten* (siehe Definition 1.2.4) bezeichnet. Die Wahrscheinlichkeiten der Wurzelknoten, d.h. der Knoten ohne Elternknoten, werden als *a-priori Wahrscheinlichkeiten* (siehe Seite 8) bezeichnet.

Der gerichtete Graph in Abbildung 1.2 stellt diese Zusammenhänge anschaulich dar. Jeder Knoten des Netzes ist binär, d.h. er hat zwei Hypothesen, die das zugehörige Ereignis eintreffen oder nicht eintreffen lassen. Die dazugehörigen bedingten Wahrscheinlichkeiten der Knoten bzw. die a-priori Wahrscheinlichkeiten der Wurzelknoten können dabei der Tabelle 1.2 entnommen werden. Bsp

ASIENBESUCH ($A: a_1 = \text{ja}, a_2 = \text{nein}$)		
$P(a) :$	$P(a_1) = 0.01$	$P(a_2) = 0.99$
TUBERKULOSE ($B: b_1 = \text{infiziert}, b_2 = \text{nein}$)		
$P(b a) :$	$P(b_1 a_1) = 0.05$	$P(b_2 a_1) = 0.95$
	$P(b_1 a_2) = 0.01$	$P(b_2 a_2) = 0.99$
TB ODER LK ($C: c_1 = \text{ja}, c_2 = \text{nein}$)		
$P(c b, e) :$	$P(c_1 b_1, e_1) = 1$	$P(c_2 b_1, e_1) = 0$
	$P(c_1 b_1, e_2) = 1$	$P(c_2 b_1, e_2) = 0$
	$P(c_1 b_2, e_1) = 1$	$P(c_2 b_2, e_1) = 0$
	$P(c_1 b_2, e_2) = 0$	$P(c_2 b_2, e_2) = 1$
RÖNTGENBEFUND ($D: d_1 = \text{positiv}, d_2 = \text{negativ}$)		
$P(d c) :$	$P(d_1 c_1) = 0.98$	$P(d_2 c_1) = 0.02$
	$P(d_1 c_2) = 0.05$	$P(d_2 c_2) = 0.95$
LUNGENKREBS ($E: e_1 = \text{positiv}, e_2 = \text{negativ}$)		
$P(e f) :$	$P(e_1 f_1) = 0.10$	$P(e_2 f_1) = 0.90$
	$P(e_1 f_2) = 0.01$	$P(e_2 f_2) = 0.99$
RAUCHER ($F: f_1 = \text{ja}, f_2 = \text{nein}$)		
$P(f) :$	$P(f_1) = 0.50$	$P(f_2) = 0.50$
BRONCHITIS ($G: g_1 = \text{positiv}, g_2 = \text{negativ}$)		
$P(g f) :$	$P(g_1 f_1) = 0.60$	$P(g_2 f_1) = 0.40$
	$P(g_1 f_2) = 0.30$	$P(g_2 f_2) = 0.70$
ATEMBESCHWERDEN ($H: h_1 = \text{ja}, h_2 = \text{nein}$)		
$P(h c, g) :$	$P(h_1 c_1, g_1) = 0.90$	$P(h_2 c_1, g_1) = 0.10$
	$P(h_1 c_1, g_2) = 0.70$	$P(h_2 c_1, g_2) = 0.30$
	$P(h_1 c_2, g_1) = 0.80$	$P(h_2 c_2, g_1) = 0.20$
	$P(h_1 c_2, g_2) = 0.10$	$P(h_2 c_2, g_2) = 0.90$

Tabelle 1.2: bedingte Wahrscheinlichkeiten bzw. a-priori Wahrscheinlichkeiten für die Knoten des Beispielnetzes *Asienbesuch* in Abbildung 1.2

1.2 Struktur

Nach der informalen Einführung der Bayesschen Netze durch zwei einfache Beispiele werden im folgenden die Begriffe und Schreibweisen formal definiert. Dazu wird zuerst die Struktur der Bayesschen Netze betrachtet, die dann mit Zufallsvariablen und Wahrscheinlichkeiten versehen wird.

Für die Bezeichnung von Graphen⁵ und Bayesschen Netzen werden im folgenden verschiedene Begriffe definiert.

Definition 1.2.1 (Graph)

Ein Graph \mathcal{G} ist ein Tupel (\mathbf{V}, \mathbf{E}) mit

- \mathbf{V} ist eine endliche Menge von *Knoten*, und
- \mathbf{E} ist eine Teilmenge von $\mathbf{V} \times \mathbf{V}$. Die Elemente von \mathbf{E} heißen *Kanten*.

Man schreibt: $\mathcal{G} = (\mathbf{V}, \mathbf{E})$.

Ein Graph $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ mit zwei Abbildungen

$$Q : \mathbf{E} \rightarrow \mathbf{V} \text{ (Quelle der Kante) und}$$

$$Z : \mathbf{E} \rightarrow \mathbf{V} \text{ (Ziel der Kante)}$$

heißt *gerichteter Graph*.

Bei einem gerichteten Graphen bezeichnet man den Knoten P einer Kante $E \in \mathbf{E}$ mit $P = Q(E)$ als *Elternknoten* vom Knoten C mit $C = Z(E)$. Der Knoten C ist umgekehrt ein *Kinderknoten* vom Knoten P .

Die Menge aller Eltern- bzw. Kinderknoten vom Knoten V bezeichnet man mit $\text{pa}(V)$ bzw. $\text{ch}(V)$.

Als *Familie* eines Knotens V in einem gerichteten Graphen bezeichnet man $\text{fa}(V) = \text{pa}(V) \cup \{V\}$.

Mit $\text{adj}(V)$ bezeichnet man die Knoten, die mit dem Knoten V eine gemeinsame Kante haben, also miteinander benachbart (*adjazent*) sind.

Eine Folge von Kanten ϕ eines gerichteten Graphen $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ mit $\phi = (\phi_1, \dots, \phi_t) \in \mathbf{E}^*$ und $\forall i \in \{1, \dots, t-1\}: Z(\phi_i) = Q(\phi_{i+1})$ heißt *gerichteter Pfad* in \mathcal{G} . Ein gerichteter Pfad $\phi = (\phi_1, \dots, \phi_t)$ mit $Q(\phi_1) = Z(\phi_t)$ heißt *Zyklus*. Enthält der Graph \mathcal{G} keine Zyklen, so heißt \mathcal{G} *azyklisch*. □

Bei Kanten unterscheidet man *gerichtete* und *ungerichtete* Kanten. Bei ungerichteten Kanten spielt die Anordnung der Knoten keine Rolle, man spricht von einer Kante *zwischen* zwei Knoten. Eine Kante zwischen den Knoten A und B schreibt man mit $\{A, B\}$. Durch eine gerichtete Kante vom Knoten A zum Knoten B wird eine *Eltern-Kind-Relation* beschrieben; A ist ein *Elternknoten* von B und B ist ein *Kinderknoten* von A . Man schreibt (A, B) . Nach der Definition 1.2.1 ist dann $Q((A, B)) = A$ und $Z((A, B)) = B$, wenn (A, B) die Kante eines gerichteten Graphens ist.

In einem gerichteten Graphen bezeichnet man Knoten, die keine Elternknoten haben, als *Wurzelknoten*.

Definition 1.2.2 (gerichteter azyklischer Graph)

Ein *gerichteter azyklischer Graph* (kurz: DAG (von *directed acyclic graph*)), ist ein Graph $\mathcal{G} = (\mathbf{V}, \mathbf{E})$, der nur gerichtete Kanten und keine Zyklen enthält. □

⁵Für die Definition von Graphen hat man sich an die Definition von *orientierten Graphen* in [Hotz 90] angelehnt. Die Bezeichnungen und Schreibweisen wurden so angepaßt, wie sie im Umgang mit Bayesschen Netzen üblich und notwendig sind.

Ein gerichteter azyklischer Graph kann durchaus mehrere gerichtete Pfade von A nach B enthalten. Man spricht dann von einem *mehrfach* ansonsten von einem *einfach* verknüpften Graphen.

Ein Bayessches Netz baut auf der Struktur eines gerichteten azyklischen Graphens auf. Jeder Knoten V des Graphen repräsentiert dabei jeweils eine Zufallsvariable X_V mit einem endlichen Wertebereich \mathbf{W}_{X_V} . Die Elemente der Menge \mathbf{W}_{X_V} sind die Elementarereignisse der Zufallsvariablen X_V . Sie sind paarweise disjunkt und decken den gesamten Ereignisraum der Zufallsvariablen ab. Jedes Elementarereignis wird durch eine Hypothese v_i des Knotens V symbolisiert.

Beispiel 1.2.3 (Hypothesen)

Soll ein Knoten beispielsweise die Geschwindigkeit eines Autos repräsentieren, so kann er Hypothesen wie “langsam”, “mittel” und “schnell” oder “0-50 km/h”, “50-100 km/h” und “100-150 km/h” enthalten. In diesem Fall kann das Auto nicht schneller als 150 km/h fahren. Der Knoten steht dann zum Beispiel für die Zufallsvariable GESCHWINDIGKEIT AUTO mit den Elementarereignissen “Das Auto fährt mit langsamer Geschwindigkeit.”, “Das Auto fährt mit mittlerer Geschwindigkeit.” und “Das Auto fährt mit schneller Geschwindigkeit.”. Bsp

Eine Hypothese kann die beiden Werte 0 für “tritt nicht ein” und 1 für “kann noch eintreten” annehmen. Der Zustand der Hypothesen eines Knotens V wird in einer Tabelle $T_{\text{finding}(V)}$ festgehalten. Hat ein Knoten 3 Hypothesen, so sieht die Tabelle im Anfang so aus: $T_{\text{finding}(V)} = \begin{bmatrix} v_1 & v_2 & v_3 \\ 1 & 1 & 1 \end{bmatrix}$, d.h., daß alle Hypothesen noch eintreten können. Soll markiert werden, daß das Elementarereignis, das durch die Hypothese v_2 des Knotens V symbolisiert wird, eingetreten ist, so sieht $T_{\text{finding}(V)}$ wie folgt aus: $T_{\text{finding}(V)} = \begin{bmatrix} v_1 & v_2 & v_3 \\ 0 & 1 & 0 \end{bmatrix}$. $T_{\text{finding}(V)}$ wird auch als *Instantiierung* des Knotens V bezeichnet. Der Belief-Wert eines Knotens V , $\text{BEL}(V)$, gibt die Wahrscheinlichkeit für das Eintreffen der Elementarereignisse der zugehörigen Zufallsvariablen von V bei gegebenen Instantiierungen der Knoten des Graphen an. $\text{BEL}(V) = \begin{bmatrix} v_1 & v_2 & v_3 \\ 0.50 & 0.30 & 0.20 \end{bmatrix}$ zum Beispiel gibt an, daß das Elementarereignis, für das die Hypothese v_2 steht, in der gegebenen Situation mit einer Wahrscheinlichkeit von 30% eintritt.

Definition 1.2.4 (Bayessches Netz)

Ein *Bayessches Netz* (BN) über einer Menge von Zufallsvariablen $\mathbf{X} = \{X_1, \dots, X_n\}$ ist ein Tupel $(\mathbf{V}, \mathbf{E}, \mathbf{X}, \mathbf{P})$ mit

- $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ ist ein *gerichteter azyklischer Graph*. Jeder Knoten $V \in \mathbf{V}$ ist mit einer Zufallsvariablen $X_V \in \mathbf{X}$ assoziiert, die den Wertebereich \mathbf{W}_{X_V} besitzt. Die Struktur des gerichteten azyklischen Graphen gibt dabei die kausalen Abhängigkeiten der Zufallsvariablen untereinander an.
- Jedem Knoten $V \in \mathbf{V}$ mit der Zufallsvariablen X_V sind bedingte Wahrscheinlichkeiten $P(V \mid \text{pa}(V))$ zugeordnet, die den Einfluß der Zufallsvariablen X_U der Knoten $U \in \text{pa}(V)$ auf die Zufallsvariable X_V des Knotens V quantifizieren.

Man schreibt $\mathcal{BN} = (\mathbf{V}, \mathbf{E}, \mathbf{X}, \mathbf{P})$. D

Die Struktur des Graphen \mathcal{G} repräsentiert also die Abhängigkeiten der Zufallsvariablen. Eine Kante von U zu V bedeutet intuitiv, daß X_U einen direkten Einfluß auf X_V besitzt.

Die Wahrscheinlichkeitsverteilung aller Zufallsvariablen läßt sich als Produkt ihrer bedingten Wahrscheinlichkeiten schreiben: $P(\mathbf{X}) = P(X_1, \dots, X_n) = \prod_{V \in \mathbf{X}} P(V \mid \text{pa}(V))$. Die bedingten Wahrscheinlichkeiten $P(V \mid \text{pa}(V))$ sind dabei dem Knoten V , der die Zufallsvariable V repräsentiert, als Tabelle bedingter Wahrscheinlichkeiten $T_{\text{cpt}(V)}$ zugeordnet (siehe Abschnitt 1.3.2).

Ist V ein Wurzelknoten, so reduzieren sich die bedingten Wahrscheinlichkeiten $P(V \mid \text{pa}(V))$ zu $P(V)$. Die den Ereignissen der Zufallsvariable eines Wurzelknotens zugeordneten Wahrscheinlichkeiten heißen *a-priori Wahrscheinlichkeiten*.

Beispiel 1.2.5 (Begriffe)

Am Beispiel 1.1.1 werden nun einige Begriffe verdeutlicht.

Der gerichtete azyklische Graph ist

$$\mathcal{G} = (\mathbf{V}, \mathbf{E}) \text{ mit } \mathbf{V} = \{A, B, C, D, E\} \text{ und} \\ \mathbf{E} = \{(A, B), (A, C), (B, D), (C, D), (C, E)\}.$$

Die Menge der diskreten Zufallsvariablen ist $\mathbf{X} = \{\text{KREBSMETASTASEN}, \text{CALCIUMSPIEGEL}, \text{GEHIRNTUMOR}, \text{KOMA}, \text{KOPFSCHMERZEN}\}$.

Die dem Knoten B zugeordnete diskrete Zufallsvariable ist

$$X_B = \text{CALCIUMSPIEGEL}.$$

Der Wertebereich von CALCIUMSPIEGEL ist

$$W_{X_B} = W_{\text{CALCIUMSPIEGEL}} = \{\text{normal}, \text{erhöht}\}.$$

Die bedingten Wahrscheinlichkeiten von CALCIUMSPIEGEL sind

$$\begin{aligned} P(\text{CALCIUMSPIEGEL} = \text{normal} \mid \text{KREBSMETASTASEN} = \text{nein}) &= 0.80, \\ P(\text{CALCIUMSPIEGEL} = \text{erhöht} \mid \text{KREBSMETASTASEN} = \text{nein}) &= 0.20, \\ P(\text{CALCIUMSPIEGEL} = \text{normal} \mid \text{KREBSMETASTASEN} = \text{ja}) &= 0.20 \text{ und} \\ P(\text{CALCIUMSPIEGEL} = \text{erhöht} \mid \text{KREBSMETASTASEN} = \text{ja}) &= 0.80. \end{aligned}$$

Die Tabelle bedingter Wahrscheinlichkeiten von B ist

$$T_{\text{cpt}(B)} = \left[\begin{array}{c|cc} & b_1 & b_2 \\ \hline a_1 & 0.80 & 0.20 \\ a_2 & 0.20 & 0.80 \end{array} \right].$$

Bsp

Zwischen den Knoten und ihren Zufallsvariablen wird im weiteren nicht mehr unterschieden, außer daß ausdrücklich darauf hingewiesen wird.

Es gibt verschiedene Lösungsverfahren für mehrfach verknüpfte Bayessche Netze, die sich zum Beispiel in exakte und approximative Algorithmen einteilen lassen.

Zu den *exakten* Algorithmen gehören zum Beispiel das *Conditioning*- und das *Clustering*-Verfahren. Beim *Conditioning* wird die Netzstruktur des mehrfach verbundenen Bayesschen Netzes aufgebrochen, um dann

auf einem einfach verbundenen Bayesschen Netz den Standardalgorithmus für einfach verbundene Bayessche Netze anzuwenden. Das *Clustering*-Verfahren von Jensen wandelt ein mehrfach verbundenes Bayessches Netz in ein einfach verbundenes Bayessches Netz um, indem Knoten zusammengeführt werden. Auf diesem einfach verbundenen Bayesschen Netz kann dann wieder der Standardalgorithmus angewendet werden.

Als *approximatives* Verfahren sei zum Beispiel die *Stochastische Simulation* erwähnt, die auf dem unveränderten mehrfach verbundenem Bayesschen Netz arbeitet. Hierbei werden mehrfach in einer Reihenfolge alle Knoten des Bayesschen Netzes nach den Wahrscheinlichkeiten für das Eintreffen der Hypothesen des aktuellen Knotens in Abhängigkeit der sie umgebenden Knoten (genauer: die Knoten der sogenannte *Markov-Überdeckung* des betrachteten Knotens) zufällig instantiiert. Der Durchschnitt über alle Instantiierungsdurchläufe ergibt dann die gesuchte Lösung. Generell kann gesagt werden, daß die approximative Lösung um so näher an die genaue Lösung herankommt, je mehr Instantiierungsdurchläufe durchgeführt werden.

Zu den exakten Algorithmen gehören ebenso die *Junction Tree* Verfahren, die sich im Vergleich zu den anderen Verfahren besser für die Behandlung von Dynamischen Bayesschen Netzen (siehe Kapitel 2 und Kapitel 3) eignen. In dieser Arbeit werden somit nur die Junction Tree Verfahren bearbeitet und die anderen Verfahren weitestgehend außer Betracht gelassen.

In den nachfolgenden Abschnitten werden nun verschiedene *Junction Tree* Verfahren vorgestellt, mit denen man in mehrfach verknüpften Bayesschen Netzen (und natürlich auch *einfach* verknüpften Bayesschen Netzen) propagieren kann. Die verschiedenen Lösungsalgorithmen werden dabei anhand der beiden oben eingeführten Beispielnetze illustriert.

1.3 Junction Tree Verfahren

Der Namensgeber der Junction Tree Verfahren ist der sogenannte *Junction Tree*. Der Junction Tree ist die einfach verbundene Struktur, die aus dem mehrfach verbundenen Bayesschen Netz bestimmt wird. Er wird benötigt, um die für die Bayesschen Netze notwendigen Berechnungen durchführen zu können. Im folgenden wird gezeigt, wie man einen Junction Tree für ein Bayessches Netz bestimmt. Im Anschluß daran wird eine neue Notation eingeführt, die z.B. die a-priori Wahrscheinlichkeiten und die bedingten Wahrscheinlichkeiten der Knoten in einer Schreibweise zusammenführt und sich somit alle Berechnungen bezüglich Bayesscher Netze durch diese Schreibweise einfacher darstellen lassen. Diese neue Notation wird im daran anschließenden Abschnitt dazu verwendet, zwei Architekturen vorzustellen, die auf dem Junction Tree arbeiten.

Bei einigen Rollup-Verfahren (siehe Kapitel 3) ist die Konstruktion eines geeigneten Junction Trees von zentraler Bedeutung. Zusammen mit der Bestimmung eines geeigneten Junction Trees und den beiden Architekturen kann der Rollup dann relativ einfach bestimmt werden.

Im Abschnitt 1.3.3 werden zwei verschiedene Junction Tree Verfahren zum exakten Lösen von Bayesschen Netzen vorgestellt. Es sind dies die *Aalborg* Architektur und die *Shafer-Shenoy* Architektur. Beide Verfahren arbeiten auf einem *Junction Tree*. Der Junction Tree ist die einfach verbundene Struktur, die aus dem mehrfach verbundenen Bayesschen Netz bestimmt wird. Für die Bestimmung des Junction Trees werden die Knoten des Bayesschen Netzes zu *Megaknoten* zusammengefaßt. Die gemeinsame Vorgehensweise beider Verfahren beim Bestimmen des Junction Trees wird im Abschnitt 1.3.1 präsentiert. Im Anschluß daran wird eine neue Notation eingeführt, die z.B. die a-priori Wahrscheinlichkeiten und die bedingten Wahrscheinlichkeiten der Knoten in einer Schreibweise zusammenführt, so daß sich alle Berechnungen auf dem Junction Tree durch diese Schreibweise darstellen lassen.

Die Shafer-Shenoy Architektur läßt sie sich im Gegensatz zur Aalborg Architektur einfach um eine besondere Funktionalität erweitern. Anstatt fester Zahlenwerte für Wahrscheinlichkeiten sind auch Variablen möglich. Dadurch kann das Bayessche Netz *parametrisiert* werden. Die Möglichkeiten, die sich durch eine Parametrisierung bieten, werden dabei an verschiedenen Beispielen anschaulich vorgeführt. Ebenso werden mit der Shafer-Shenoy Architektur Formeln für die Berechnung von Wahrscheinlichkeiten hergeleitet, so daß diese an verschiedene Algebra-Programme weitergegeben werden können.

1.3.1 Bestimmung des Junction Trees

Der Junction Tree wird auch als die sekundäre Struktur des Bayesschen Netzes bezeichnet (Der gerichtete azyklische Graph, der die Abhängigkeiten der Zufallsvariablen des Bayesschen Netzes modelliert, wird somit auch primäre Struktur genannt.). Wie die Bezeichnungen richtig vermuten lassen, wird der Junction Tree aus dem gerichteten azyklischen Graphen (auch DAG⁶ genannt) bestimmt. Für einen DAG existieren dabei mehrere Junction Trees als Lösungen. Vom Junction Tree selbst ist es abhängig, wie schnell ein Bayessches Netz gelöst werden kann und wie hoch die Speicherkomplexität liegt. Die unterschiedlichen Lösungen für einen Junction Tree eines DAGs sind dabei unterschiedlich gut.

Im folgenden wird nun die allgemeine Vorgehensweise zur Bestimmung eines Junction Trees für ein Bayessches Netz den Einzelschritten nach strukturiert dargestellt. Zu den verschiedenen Einzelschritten werden gelegentlich einzelne Effizienzaspekte für einen Junction Tree angesprochen. Zum besseren Verständnis werden die Einzelschritte zur Bestimmung des Junction Trees anhand der beiden Beispielnetze, die in diesem Kapitel eingeführt wurden, grafisch dargestellt.

Dieser Abschnitt gibt einen Überblick über die Bestimmung eines *Junction Trees* für ein Bayessches Netz $\mathcal{BN} = (\mathbf{V}, \mathbf{E}, \mathbf{X}, \mathbf{P})$ über den Zufallsvariablen \mathbf{X} . Bei der Umwandlung des Bayesschen Netzes in seine zweite Struktur, dem *Junction Tree*, werden Einzelschritte durchgeführt, die wie folgt zusammengefaßt werden können:

1. Konstruktion des *moralisierten Graphen* \mathcal{G}^m aus dem gerichteten azyklischen Graphen \mathcal{G}

⁶DAG ist ein Akronym für *directed acyclic graph*

2. Bestimmung einer *Eliminationsreihenfolge* elim und Konstruktion des *triangulierten Graphen* \mathcal{G}^t aus dem moralisierten Graphen \mathcal{G}^m
3. Bestimmung des ungerichteten *Junction Graphens* \mathcal{J} mit Gewichten. Die *Cliquen* von \mathcal{G}^t bilden die Knoten des Junction Graphens \mathcal{J} .
4. Bestimmung des *Junction Trees* \mathcal{T} aus dem Junction Graphen \mathcal{J} .
5. Jeder Knoten $V \in \mathbf{V}$ des Bayesschen Netzes \mathcal{BN} wird einer bestimmten Clique des Junction Trees \mathcal{T} zugeordnet.
6. Wähle einen beliebigen Knoten des *Junction Trees* \mathcal{T} als Wurzelknoten .

Die Schritte von 1 bis 4 sind dabei von graphentheoretischer Art, d.h. es werden nur Umformungen des Graphens des Bayesschen Netzes vorgenommen. Für Details und Hintergründe wird auf [Pearl 88; Neapolitan 90; Jensen 96; Huang & Darwiche 94; Brandherm et al. 97] verwiesen. Im folgenden wird für jeden Einzelschritt ein Überblick gegeben.

Moralisierung

Bei der Moralisierung werden die Elternknoten eines Knotens durch Kanten miteinander verbunden. Man sagt auch, daß die Elternknoten miteinander verheiratet werden, was den Begriff *Moralisierung* erklärt.

Die Moralisierung eines Graphen \mathcal{G} stellt sicher, daß ein Knoten zusammen mit seinen Elternknoten in einer Clique des Junction Trees zu liegen kommt. Dies spielt eine wichtige Rolle bei der Zuordnung von Knoten zu Cliques, die weiter hinter in diesem Abschnitt beschrieben wird.

Sei nun $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ der gerichtete azyklische Graph des Bayesschen Netzes $\mathcal{BN} = (\mathbf{V}, \mathbf{E}, \mathbf{X}, \mathbf{P})$. Den *moralisierten Graphen* \mathcal{G}^m erhält man aus dem Graphen \mathcal{G} wie folgt:

1. Es wird der Graph \mathcal{G} in $\mathcal{G}^m = (\mathbf{V}^m, \mathbf{E}^m)$ "kopiert".
2. Für alle Knoten $V \in \mathbf{V}^m$ werden die Elternknoten $\text{pa}(V)$ durch ungerichtete Kanten miteinander verbunden.
3. Es werden alle gerichteten Kanten $(A, B) \in \mathbf{E}^m$ durch ungerichtete Kanten $\{A, B\}$ ersetzt.

Der aus diesen Umformungen resultierende Graph $\mathcal{G}^m = (\mathbf{V}^m, \mathbf{E}^m)$ ist ein ungerichteter Graph. Es gilt: $\mathbf{V} = \mathbf{V}^m$ und $\mathbf{E} \subset \mathbf{E}^m$. In Abbildung 1.3 (a) und (b) sind die moralisierten Graphen der Beispielnetze aus den Beispielen 1.1.1 und 1.1.2 dargestellt. Die neu eingefügten Kanten sind gestrichelt dargestellt.

Eliminationsreihenfolge

Mit der Eliminationsreihenfolge ist die Struktur des Junction Trees für ein Bayessches Netz eindeutig bestimmt. Die Eliminationsreihenfolge ist eine Liste der Knoten des Bayesschen Netzes. Ändert man die Reihenfolge der Knoten in der Liste, so entsteht im allgemeinen auch ein anderer

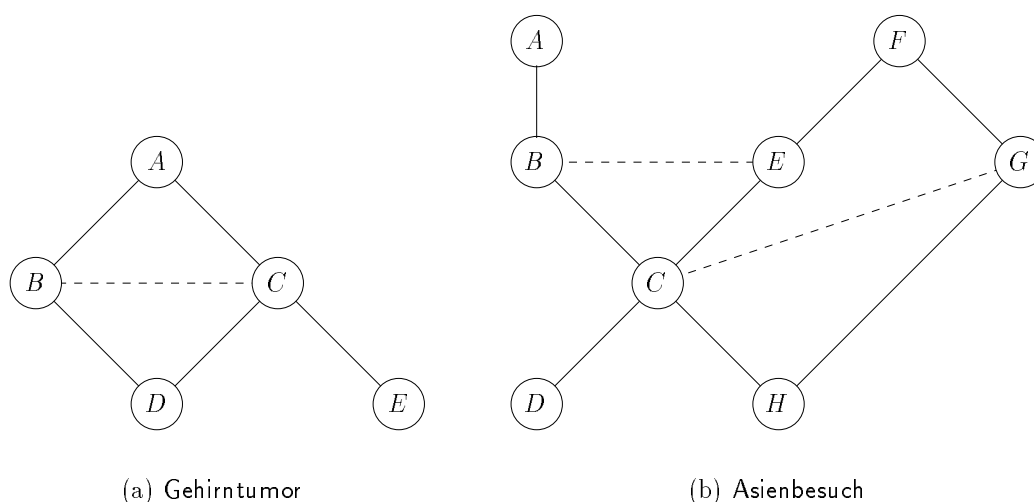


Abbildung 1.3: Beispiele für moralisierte Graphen

Junction Tree. Für die sogenannte Triangulierung, die im folgenden Abschnitt beschrieben wird, ist die Eliminationsreihenfolge der Fahrplan für die Bestimmung der Triangulationskanten. Wie man schon sieht, sind Eliminationsreihenfolge und die Triangulierung stark miteinander verbunden. Ist die Eliminationsreihenfolge bekannt, so kann der DAG des Bayesschen Netzes trianguliert werden.

In diesem Abschnitt werden nur die notwendigsten Begriffe eingeführt und auf Beweise verzichtet, da sie für diese Arbeit im weiteren nicht notwendig sind.

Um das mehrfach verbundene Bayessches Netz in eine einfach verbundene Struktur umzuwandeln, werden Knoten zu “Megaknoten” zusammengefaßt. Dazu werden die Knoten des Bayesschen Netzes in einer bestimmten Reihenfolge *eliminiert*. Bei der Elimination von Knoten werden Kanten im Graphen des Bayesschen Netzes eingefügt, so daß das Bayessche Netz zusammenhängend bleibt, wenn man den Knoten aus dem Netz entfernt. Dies ist wichtig für die Junction Tree Eigenschaft (siehe Definition 1.3.9), die vom *Junction Tree* erfüllt werden muß. Die Reihenfolge in der die Knoten des Bayesschen Netzes eliminiert werden, wird als *Eliminationsreihenfolge* bezeichnet. Die Eliminationsreihenfolge zusammen mit den Kanten (auch den neu eingefügten) des Netzes geben an, welche Knoten zusammen in einem “Megaknoten” zu liegen kommen. Es entsteht zunächst wieder ein mehrfach verbundenes Netz, der Junction Graph, der dann durch einfaches Löschen bestimmter Kanten in einen einfach verbundenen Graphen, den Junction Tree, umgewandelt wird. Dazu mehr in den zugehörigen Abschnitten. Die “Megaknoten” werden später *Cliquen* genannt. Bei unterschiedlichen Eliminationsreihenfolgen entstehen unterschiedlich viele und unterschiedlich große “Megaknoten”. Ziel ist es, möglichst kleine “Megaknoten” zu erhalten. Eine Eliminationsreihenfolge, die dies leistet, wird als *optimal* bezeichnet.

Die Bestimmung einer optimalen Eliminationsreihenfolge ist jedoch *NP*-vollständig [Cooper 87]. Deswegen empfehlen sich zur Berechnung einer Eliminationsreihenfolge Heuristiken, wie sie z.B. in [Kjærulff 90; Kjærulff 93] vorgestellt werden.

Das traditionelle Verfahren zur Bestimmung einer Eliminationsreihenfolge ist **maximum cardinality search**. Dabei wird zufällig ein beliebiger Knoten des Graphen als Startknoten ausgewählt und markiert. Der Knoten und alle seine unmarkierten Nachbarknoten bilden dabei einen *Cluster*. Alle Knoten des Clusters werden miteinander mit ungerichteten Kanten verbunden. Als nächstes wird der Knoten bearbeitet, der jetzt die meisten markierten Nachbarknoten besitzt. Gibt es mehrere solcher Knoten, dann wird ein Knoten zufällig ausgewählt. Das Verfahren terminiert, wenn alle Knoten des Graphen markiert sind. Die Knoten ergeben nun sortiert nach der Reihenfolge ihrer Abarbeitung die Eliminationsreihenfolge.

Das Markieren eines Knotens und das Verbinden aller Knoten eines Clusters miteinander bezeichnet man auch als *Elimination*.

Definition 1.3.1 (Löschen, Eliminieren)

Werden ein Knoten $V \in \mathbf{V}$ und seine Kanten aus $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ entfernt, so sagt man, daß V *gelöscht* (*delete*) wird.

Werden alle Nachbarknoten von $V \in \mathbf{V}$ ($\text{adj}(V)$) durch Kanten miteinander verbunden und dann V gelöscht, dann sagt man, daß der Knoten V *eliminiert* (*eliminate*) wird. □

Bei der Elimination eines Knotens bleibt der Teilgraph, der vom Knoten induziert wird, durch das Einfügen der Kanten zwischen den restlichen Knoten des Teilgraphen als Zusammenhangskomponente erhalten. Dieses kommt bei der Junction Tree Eigenschaft zum Tragen.

Bei den Clustering-Verfahren in [Pearl 88; Neapolitan 90] ist die Eliminationsreihenfolge durch **maximum cardinality search** vorbestimmt. Die Clustering-Verfahren von [Jensen 96; Shafer 96] akzeptieren beliebige Eliminationsreihenfolgen.

Andere Algorithmen zur Bestimmung der Eliminationsreihenfolge berücksichtigen das *Gewicht* der Clique, die bei der Elimination eines Knotens entstehen würde. Die Heuristik dabei ist, daß insgesamt Cliquen mit kleinen Gewichten entstehen. Als Gewicht einer Clique bezeichnet man die Anzahl der verschiedenen Zustände, die die Clique haben kann. Enthält die Clique die Knoten K_1, \dots, K_n , dann bestimmt sich die Anzahl der Zustände der Clique zu $|K_1| \cdot \dots \cdot |K_n|$. Für $1 \leq i \leq n$ gibt $|K_i|$ die Anzahl der Hypothesen des Knotens K_i an.

Durch die Eliminationsreihenfolge wird auf dem Graphen eine Ordnung der Knoten vorgenommen.

Definition 1.3.2 (Ordnung des Graphen)

Eine *Ordnung* auf $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ ist eine Bijektion $\# : \mathbf{V} \leftrightarrow \{1, 2, \dots, n\}$, mit $|\mathbf{V}| = n$. Den Graphen $\mathcal{G}_\# = (\mathbf{V}_\#, \mathbf{E})$ bezeichnet man als *geordneter Graph*. □

Es gilt: $\mathcal{G} = \mathcal{G}_\#$ und $\mathbf{V} = \mathbf{V}_\#$. Der Index $\#$ gibt lediglich an, daß dem Graphen bzw. der Knotenmenge eine Ordnung zugewiesen wurde.

Durch die Eliminationsreihenfolge *elim* wird der Graph \mathcal{G} zu einem geordneten Graphen.

Definition 1.3.3 (Eliminationsreihenfolge)

Die Ordnung des Graphen $\mathcal{G}_\#$ (siehe Definition 1.3.2) steht mit der Eliminationsreihenfolge *elim*_# in folgendem Zusammenhang: $\text{elim}_\# = (\#^{-1}(1), \dots, \#^{-1}(n)) = (V_{\#(1)}, \dots, V_{\#(n)})$. □

Die Eliminationsreihenfolge in Abbildung 1.4(a) wurde durch **maximum cardinality search** bestimmt: Der Knoten A wurde durch Zufall als erster Knoten ausgewählt. Als nächste Kandidaten stehen die beiden Knoten B und C zur Auswahl. Das Los fällt auf den Knoten B . Der Knoten C hat nun die meisten markierten Nachbarknoten und wird als nächster markiert. Dann kommt der Knoten D mit den meisten markierten Nachbarknoten an die Reihe und zum Schluß der Knoten E . Bei der Bestimmung der Eliminationsreihenfolge in Abbildung 1.4(b) wurde die folgende Heuristik angewandt: Eliminiere die Knoten des Graphen, die keine Kanten im Graphen induzieren. Gibt es keine solche Knoten, so wähle den Knoten, der die kleinste Clique entstehen läßt. Bei der Eliminierung der Knoten H , D , A und B in dieser Reihenfolge werden keine Kanten im Graphen eingefügt. Jetzt steht kein Knoten mehr zur Auswahl, der keine Kanten bei seiner Eliminierung im Graphen einfügen würde. Jeder der verbleibenden Knoten fügt bei seiner Eliminierung genau eine Kante ein. Die entstehenden Cliques wären somit alle gleich groß. Die zufällige Wahl fällt auf den Knoten G . Er induziert die Kante zwischen den Knoten C und F . Die Eliminierung der Knoten F , E und C in dieser Reihenfolge fügt keine weiteren Kanten mehr im Graphen ein.

Bei der vorhergehenden Bestimmung der Eliminationsreihenfolge wurden schon Kanten im Graphen eingefügt. Der resultierende Graph ist am Ende *trianguliert*. Bei der vorhergehenden Bestimmung der Eliminationsreihenfolge durch **maximum cardinality search** wurde der Graph aber noch nicht trianguliert. Wie man bei einer gegebenen Eliminationsreihenfolge die Triangulierung eines Graphen berechnet, und was Triangulierung bedeutet, wird im folgenden besprochen.

Triangulierung

Die Triangulierung von Bayesschen Netzen ist nicht zu verwechseln mit der Triangulierung in der Computergrafik, bei der die geometrischen Objekte in Dreiecke zerlegt werden, so daß man nur konkave Objekte (Dreiecke sind immer konkav) erhält, die dann vom Rechner sehr schnell zu zeichnen sind.

In diesem Abschnitt wird beschrieben, wann ein Graph trianguliert ist, wie man bei gegebener Eliminationsreihenfolge die Triangulierung eines Graphen bestimmt, und wie man bei bereits triangulierten Graphen die Triangulierung verbessern kann. An den beiden Beispielnetzen wird gezeigt, wie man eine gute Triangulierung von einer schlechten Triangulierung unterscheiden kann.

Eine gute Triangulierung ist die Voraussetzung für einen schlanken Junction Tree, der notwendig für schnelle Berechnungen ist.

Die *Triangulierung* wird auf dem Graphen \mathcal{G}^m ausgeführt. Der Graph \mathcal{G}^m ist dann trianguliert, wenn jeder Zyklus der Länge vier oder länger eine Kante besitzt, die zwei nichtbenachbarte Knoten im Zyklus im Graphen miteinander verbindet. Diese Kante wird auch als "*chord*" bezeichnet. Die Triangulierung kann zusammen mit der Bestimmung der Eliminationsreihenfolge durchgeführt werden. Da in manchen Fällen aber die Eliminationsreihenfolge durch den Benutzer vorgegeben wird, wurde eine getrennte Vorstellung von Eliminationsreihenfolge und Triangulierung bevorzugt.

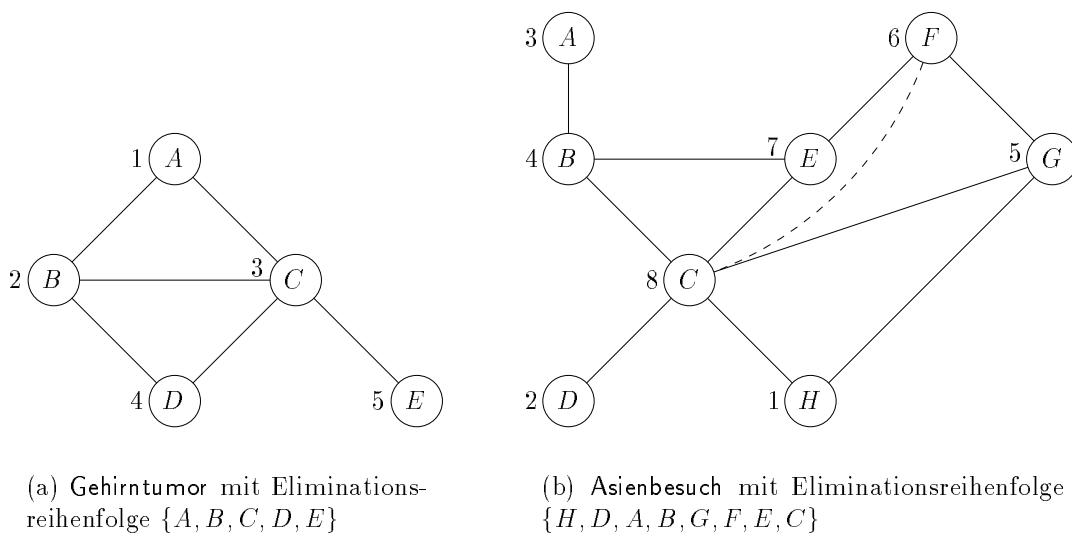


Abbildung 1.4: Beispiele für gut triangulierte Graphen

Sei $\mathcal{G}^m = (\mathbf{V}^m, \mathbf{E}^m)$ der moralisierte Graph des Bayesschen Netzes. Den *triangulierten Graphen* \mathcal{G}^t erhält man mithilfe der Eliminationsreihenfolge $elim$ wie folgt:

1. Es wird der Graph \mathcal{G}^m in $\mathcal{G}^t = (\mathbf{V}^t, \mathbf{E}^t)$ "kopiert".
Alle Knoten $V \in \mathbf{V}^t$ sind unmarkiert.
2. Für alle Knoten $V \in \mathbf{V}^t$ wird der Reihe nach, wie sie in der Eliminationsreihenfolge $elim$ stehen⁷, folgendes durchgeführt:
 - (a) Die Nachbarknoten von V , die noch nicht markiert sind, werden durch eine ungerichtete Kante miteinander verbunden. (Dadurch kann sich die Menge der Nachbarknoten eines noch nicht markierten Knotens V vergrößern.)
 - (b) Der Knoten V wird markiert.
3. \mathcal{G}^t ist nun der triangulierte Graph von \mathcal{G} .

In den Abbildungen 1.4 und 1.5 wird deutlich, wie die Qualität der Triangulierung von der Eliminationsreihenfolge abhängt. In Abbildung 1.4(a) wurde keine Kante und in Abbildung 1.4(b) nur eine Kante eingefügt, für die sich der Knoten G verantwortlich zeigt. In Abbildung 1.5(a) wurden fünf Kanten eingefügt, so daß jeder Knoten des Graphen \mathcal{G} mit jedem Knoten des Graphen verbunden ist. Schlechter hätte eine Triangulierung für diesen Graphen nicht ausfallen können. In Abbildung 1.5(b) kamen sieben Kanten hinzu. Für diesen Graphen gibt es allerdings noch schlechtere Triangulierungen.

⁷In [Pearl 88] wird z.B. die Eliminationsreihenfolge in umgekehrter Reihenfolge abgearbeitet. Je nachdem, ob die Eliminationsreihenfolge *vorwärts* oder *rückwärts* abgearbeitet wird, können zwei unterschiedliche Triangulierungen entstehen, die dann i.a. auch in der Qualität stark variieren. Deswegen ist es wichtig zu wissen, ob vorwärts oder rückwärts eliminiert wird. Wegen der Dynamischen Bayesschen Netze, die in späteren Kapiteln besprochen werden, wird hier die Vorwärtselimination vorgezogen.

Definition 1.3.4 (Triangulierung)

Die Menge \mathbf{T} der Kanten, die hinzugefügt werden, wenn alle Knoten in \mathbf{V} in der Reihenfolge eliminiert werden, wie sie in der Eliminationsreihenfolge stehen, nennt man *Triangulierung* von \mathcal{G}^m , da $(\mathbf{V}, \mathbf{E}^m \cup \mathbf{T})$ *trianguliert* ist. \square

Die Kanten \mathbf{T} bezeichnet man als *fill-edges* oder *fill-ins*. Die fill-ins sind in den Abbildungen 1.4 und 1.5 gestrichelt eingezeichnet. Der Vergleich von Abbildung 1.4(a) mit Abbildung 1.5(a) zeigt, daß die neu eingefügten Kanten in 1.5(a) überflüssig für eine Triangulierung sind. Man bezeichnet diese Kanten als *redundante fill-ins*.

Definition 1.3.5 (redundanter fill-in)

Sei \mathbf{T} eine Triangulierung von \mathcal{G}^m . Dann bezeichnet man eine Kante $f \in \mathbf{T}$ als *redundant*, wenn auch $\mathbf{T} \setminus \{f\}$ eine Triangulierung von \mathcal{G}^m ist. \square

Werden die redundanten fill-ins nacheinander entfernt, so daß keine redundanten fill-ins mehr vorhanden sind, dann erhält man eine *minimale Triangulierung*, d.h. wenn ein weiterer fill-in der Triangulierung \mathbf{T} entfernt würde, dann wäre \mathbf{T} keine Triangulierung mehr.

Definition 1.3.6 (minimale Triangulierung)

Eine Triangulierung \mathbf{T} von $\mathcal{G}^m = (\mathbf{V}, \mathbf{E}^m)$ heißt *minimal*, wenn $\forall \mathbf{T}' \subset \mathbf{T}$ gilt: $\mathcal{G}' = (\mathbf{V}, \mathbf{E}^m \cup \mathbf{T}')$ ist nicht trianguliert. \square

In [Kjærulff 93] wird der folgende Algorithmus vorgeschlagen, um redundante fill-ins einer nichtminimalen Triangulation \mathbf{T} eines Graphen \mathcal{G} zu entfernen. Die Komplexität liegt bei $O(c^2 |\mathbf{T}|^2)$. Der Algorithmus wird mit $\text{minT}(\mathbf{T}, \mathcal{G} = (\mathbf{V}, \mathbf{E} \cup \mathbf{T}), \mathbf{T})$ aufgerufen.

```

minT( $\mathbf{T}, \mathcal{G} = (\mathbf{V}, \mathbf{E} \cup \mathbf{T}), \mathbf{R}$ )
(1)  $\mathbf{R}' \leftarrow \{E_1 \in \mathbf{T} \mid \exists E_2 \in \mathbf{R} : E_1 \cap E_2 \neq \emptyset\}$ 
(2)  $\mathbf{T}' \leftarrow \{\{V, W\} \in \mathbf{R}' \mid \text{Graph}(\text{adj}(V, \mathcal{G}) \cap \text{adj}(W, \mathcal{G})) \text{ ist zusammenhängend}\}$ 
(3) if  $\mathbf{T}' \neq \emptyset$  then
(4)   return ( $\text{minT}(\mathbf{T} \setminus \mathbf{T}', \mathcal{G} = (\mathbf{V}, \mathbf{E} \cup \mathbf{T} \setminus \mathbf{T}'), \mathbf{T}')$ )
(5) else
(6)   return ( $\mathbf{T}$ )
(7) fi

```

In Programmzeile (1) wird die Menge \mathbf{R}' bestimmt, d.h. die Menge der Kanten aus \mathbf{T} , die einen Knoten gemeinsam haben mit einer Kante aus \mathbf{R} . Im Anfang ist $\mathbf{R} = \mathbf{T}$. Um \mathbf{R}' zu bestimmen, gehe alle Kanten aus \mathbf{T} durch und überprüfe, ob eine Kante aus \mathbf{R} wenigstens einen Knoten mit ihr gemeinsam hat. Ist dies der Fall, so gehört die Kante zu \mathbf{R}' . Im nächsten Schritt wird \mathbf{T}' bestimmt. Das ist die Menge der Kanten aus \mathbf{R}' , die entfernt werden kann, ohne daß der Graph nicht mehr trianguliert ist. Um \mathbf{T}' zu bestimmen, müssen alle ungerichteten Kanten $\{V, W\} \in \mathbf{R}'$ darauf überprüft werden, ob die Schnittmenge $\text{adj}(V, \mathcal{G}) \cap \text{adj}(W, \mathcal{G})$ einen zusammenhängenden Graphen induziert.

Eine minimale Triangulierung ist aber nicht unbedingt die *absolut minimale* Triangulierung eines Graphen. Es gilt zwar, daß $\forall \mathbf{T}' \subset \mathbf{T}$ ist $\mathcal{G}' = (\mathbf{V}, \mathbf{E}^m \cup \mathbf{T}')$ nicht trianguliert, es kann aber Triangulierung $\mathbf{T}_<$ geben mit $\mathbf{T}_< \not\subset \mathbf{T}$ und $|\mathbf{T}_<| < |\mathbf{T}|$.

Definition 1.3.7 (perfekte Eliminationsreihenfolge)

Die Triangulierung $\mathbf{T}(\mathcal{G}_\#)$ ist die Menge der Kanten, die durch die Elimination der Knoten $V \in \mathbf{V}$ von \mathcal{G} in der Reihenfolge $elim_\#$ dem Graphen \mathcal{G} hinzugefügt wurden. Eine Eliminationsreihenfolge $elim_\#$ heißt *perfekt*, wenn $\mathbf{T}(\mathcal{G}_\#) = \emptyset$. \square

Es existiert nicht für jeden Graphen \mathcal{G} eine perfekte Eliminationsreihenfolge. Der Graph in Abbildung 1.4(b) ist ein Beispiel dafür, da sich keine Eliminationsreihenfolge gibt, die weniger Triangulationskanten einfügt.

Wie sich die Triangulierung des Graphen auf den Junction Tree auswirkt und deshalb von *guten* und *schlechten* Triangulierungen die Rede ist, wird in 1.3.1 beleuchtet.

Für effizientere Methoden zur Graphentriangulierung wird auf [Becker & Geiger 96] verwiesen.

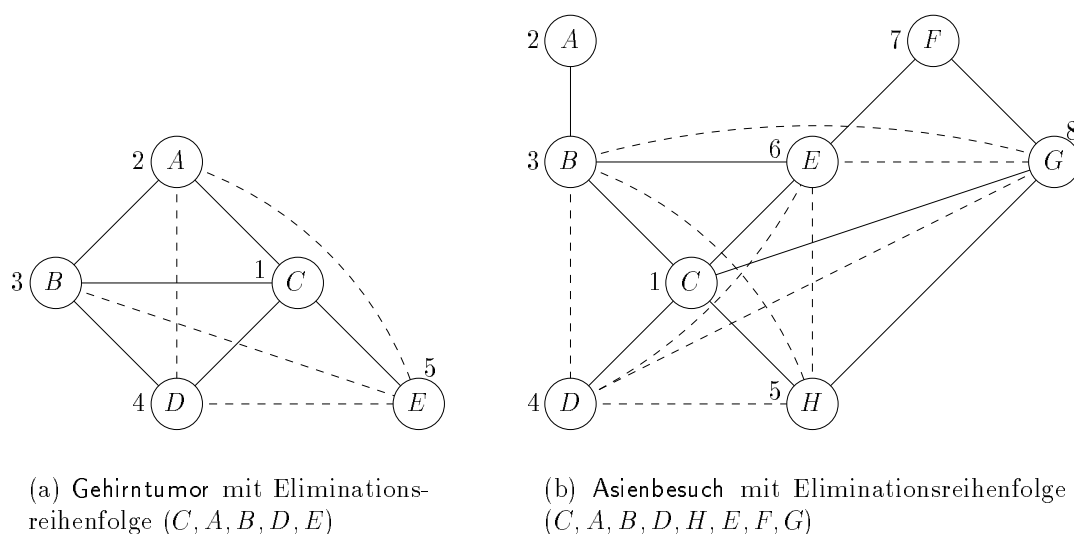


Abbildung 1.5: Beispiele für schlecht triangulierte Graphen

Im triangulierten Graphen können nun mit der Eliminationsreihenfolge die Cliques des Junction Graphen \mathcal{J} bestimmt werden.

Junction Graph \mathcal{J}

Der Junction Graph \mathcal{J} ist die Vorstufe vom Junction Tree \mathcal{T} . Die Knoten des Junction Graphens sind die Cliques des triangulierten Graphens $\mathcal{G}_\#^t$. Die Knoten des Junction Graphens bezeichnet man auch als Cliques. Die Cliques des Junction Graphens, die gemeinsame Knoten enthalten, sind dabei durch Kanten miteinander verbunden.

Die *maximalen Cluster* des Graphen $\mathcal{G}_\#^t$, auch *Cliques* genannt, bilden die Knoten des Junction Graphens \mathcal{J} .

Definition 1.3.8 (Cluster)

In einem triangulierten Graphen mit der Ordnung $\#$, bildet der Knoten $V \in \mathcal{G}_\#^t$ zusammen mit seinen Nachbarknoten $N \in \text{adj}(V)$ mit $\#(N) > \#(V)$, die also eine höhere Ordnung als V haben, einen *Cluster*. \square

Ein Cluster Clu_i heißt *maximal*, wenn sich kein weiterer Cluster Clu_ℓ des Graphen $\mathcal{G}_\#^t$ findet, so daß der Cluster Clu_i eine Teilmenge vom Cluster Clu_ℓ ist, d.h. $\exists \ell : Clu_i \subset Clu_\ell$.

Die Cliques des Junction Graphens \mathcal{J} , die Knoten des Graphen \mathcal{G} gemeinsam haben, werden durch eine Kante verbunden. Auf diese Weise erhält man einen mehrfach verbundenen Graphen \mathcal{J} . Den Kanten des Junction Graphens \mathcal{J} ist dabei in Abhängigkeit von den beiden durch die Kante verbundenen Cliques ein *Gewicht* zugeordnet. Das Gewicht der Kante zwischen Clq_i und Clq_ℓ bestimmt sich als $\gamma := |Clq_i \cap Clq_\ell|$. $Clq_i \cap Clq_\ell$ wird als *Separator* von den beiden Cliques Clq_i und Clq_ℓ bezeichnet.

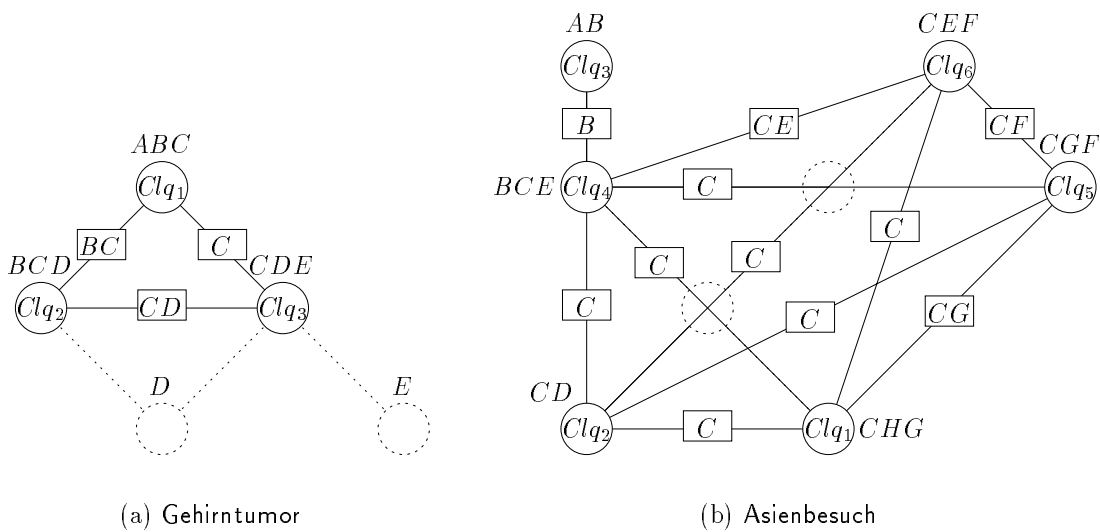


Abbildung 1.6: Beispiele für Junction-Graphen bei guter Triangulierung

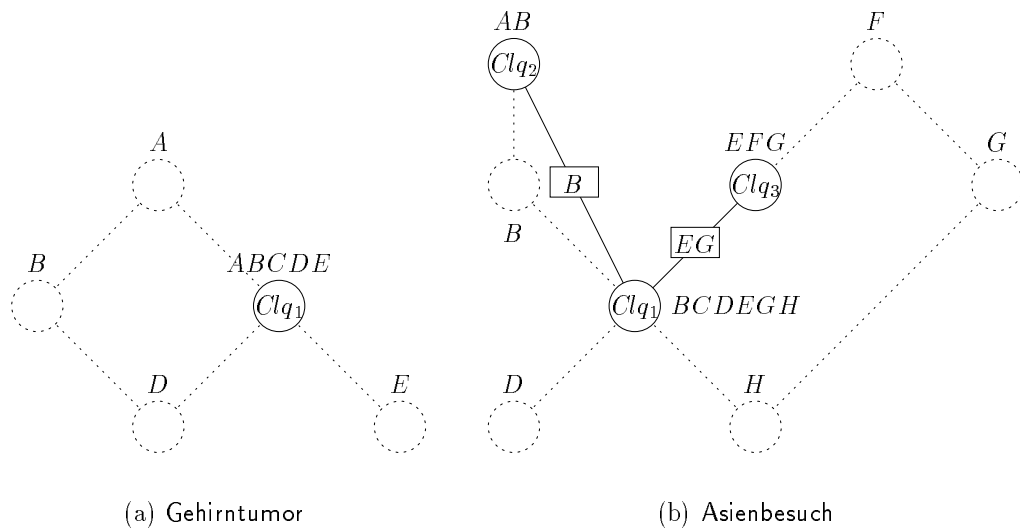


Abbildung 1.7: Beispiele für Junction-Graphen bei schlechter Triangulierung

Durch das Entfernen bestimmter Kanten im Junction Graphen $\mathcal{J} = (\mathbf{C}, \mathbf{E}_\gamma)$ erhält man einen Junction Tree \mathcal{T} , wobei mit \mathbf{C} die Cliques des Graphen und mit \mathbf{E}_γ die mit einem Gewicht versehenen Kanten des Graphen bezeichnet werden.

Junction Tree

Der Junction Graph \mathcal{J} ist eine mehrfach verbundene Struktur. Im folgenden wird gezeigt, wie aus dieser mehrfach verbundenen Struktur eine einfach verbundene Struktur, der sogenannte mphJunction Tree \mathcal{T} , bestimmt wird, der die sogenannte *Junction Tree Eigenschaft* erfüllt.

Der Junction Tree $\mathcal{T} = (\mathbf{C}, \mathbf{E}_{Sep}^{\mathcal{T}})$ des Junction Graphen \mathcal{J} ist ein minimal aufspannender Graph von \mathcal{J} (siehe auch [Jensen & Jensen 94]), der die *Junction Tree Eigenschaft* erfüllt. Es gilt: $\mathbf{E}_{Sep}^{\mathcal{T}} \subseteq \mathbf{E}_\gamma$. Ebenso sind den Kanten des Junction Trees nicht mehr wie beim Junction Graphen Gewichte γ sondern Separatoren *Sep* zugeordnet. Dieses wird durch die entsprechenden Indizes gekennzeichnet.

Definition 1.3.9 (Junction Tree Eigenschaft)

Ein minimal aufspannender Graph $\mathcal{M} = (\mathbf{C}, \mathbf{E}_{\mathcal{M}})$ des Junction Graphen $\mathcal{J} = (\mathbf{C}, \mathbf{E}_\gamma)$ erfüllt die *Junction Tree Eigenschaft*, wenn für alle Knoten V des Graphen \mathcal{G} gilt: Werden alle Cliques aus \mathcal{J} entfernt, die den Knoten V nicht enthalten, dann bleibt der entstandene Unterbaum zusammenhängend, d.h. zwei Cliques in \mathcal{M} , die den Knoten V enthalten, sind in \mathcal{M} adjazent (benachbart), oder über einen Pfad verbunden, der über Cliques geht, die den Knoten V enthalten.

Ein minimal aufspannender Graph \mathcal{M} von \mathcal{J} , der die Junction Tree Eigenschaft erfüllt, wird als *Junction Tree* \mathcal{T} bezeichnet. □

Im allgemeinen ist der Junction Tree eines Junction Graphens \mathcal{J} nicht eindeutig bestimmt. In Abbildung 1.8(b) hätte anstatt der Kante zwischen den Cliques Clq_2 und Clq_6 auch die Kante zwischen den Cliques Clq_2 und Clq_4 eingefügt werden können.

Der Algorithmus **MST-Kruskal**(\mathcal{J}, γ) (siehe [Cormen et al. 92]) bestimmt aus einem gegebenen Junction Graphen einen Junction Tree.

```

MST-Kruskal( $\mathcal{J}, \gamma$ )
(1) ordne jeden Knoten  $V$  des Graphen  $\mathcal{J}$  einem eigenen Baum  $\mathcal{B} = (\{V\}, \emptyset)$  zu,
    d.h.  $\mathcal{B}(V) = (\{V\}, \emptyset)$ 
(2) sortiere alle Kanten  $E = \{U, V\} \in \mathbf{E}_\gamma$  des Graphen  $\mathcal{J}$  absteigend bzgl.  $\gamma$ 
(3) for jede Kante  $E = \{U, V\}$  des Graphen  $\mathcal{J}$  nach Sortierung do
(4)   if  $\mathcal{B}(U) \neq \mathcal{B}(V)$  then
(5)     verbinde beide Bäume  $\mathcal{B}(U) = (\mathbf{N}_U, \mathbf{E}_U)$  und  $\mathcal{B}(V) = (\mathbf{N}_V, \mathbf{E}_V)$  durch
        die Kante  $E$  zu einem Baum  $\mathcal{B} = (\mathbf{N}_U \cup \mathbf{N}_V, \mathbf{E}_U \cup \mathbf{E}_V \cup E)$ 
(6)   fi
(7) od
(8) return  $\mathcal{B}$ 

```

Es wird der Junction Graph \mathcal{J} mit der Gewichtung γ übergeben. In Programmzeile (1) wird jedem Knoten V des Junction Graphen \mathcal{J} ein eigener Baum $\mathcal{B}(V)$ zugeordnet. Im Anfang enthält dieser Baum nur den jeweiligen Knoten V und noch keine

Kanten. In der Zeile (2) werden die Kanten des Graphen \mathcal{J} bzgl. ihres Gewichtes γ absteigend sortiert. In den Programmzeilen (3) bis (7) wird der minimal aufspannende Graph bestimmt. In Zeile (8) wird der minimal aufspannende Graph \mathcal{B} als Ergebnis zurückgegeben.

Wie man einen optimalen Junction Tree aus einem gegebenen Junction Graphen erhält, so daß später die benötigte Zeit für Inferenzen minimiert wird, wird z.B. in [Jensen & Jensen 94; Huang & Darwiche 94] beschrieben.

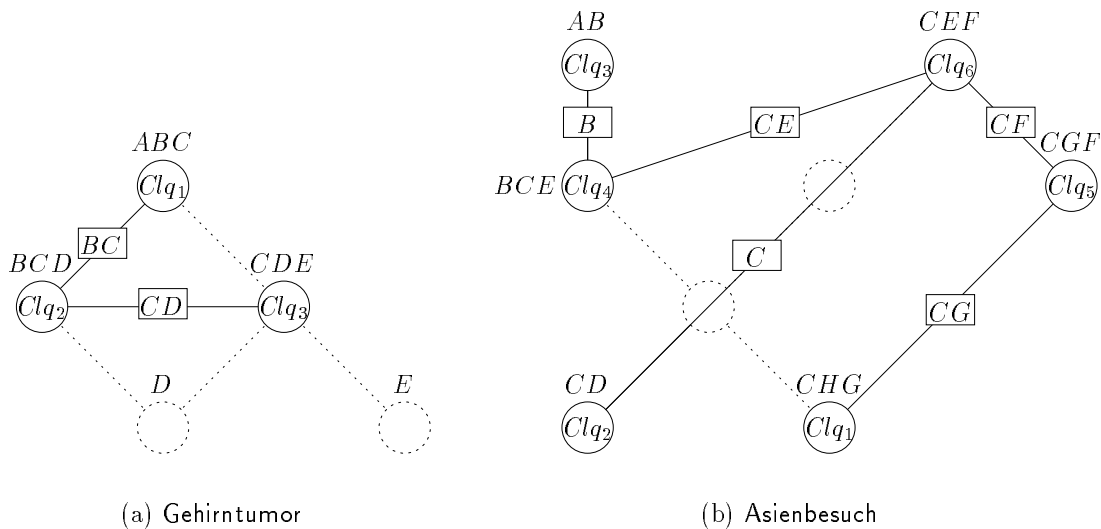


Abbildung 1.8: Beispiele für Junction-Trees bei guter Triangulierung

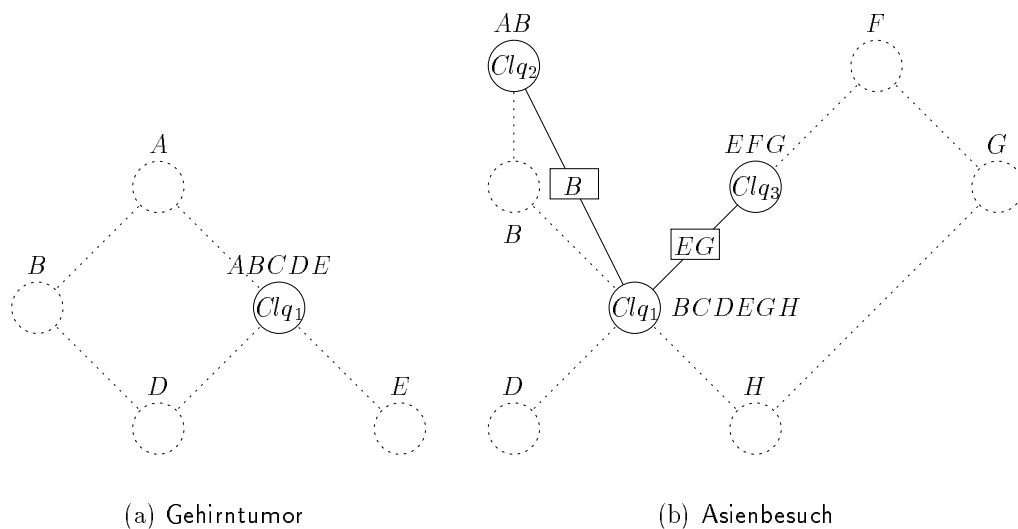


Abbildung 1.9: Beispiele für Junction-Trees bei schlechter Triangulierung

Jeder Kante (Clq_i, Clq_j) des *Junction Trees* \mathcal{T} wird ein Separatorknoten $Sep_{i,j} = Clq_i \cap Clq_j$ zugeordnet.

<i>triangulierter Graph</i>			<i>Junction-Graph</i>	
<i>eliminiertes Knoten</i>	<i>fills</i>	<i>induziertes Cluster</i>	<i>induzierte Clique</i>	<i>Kante zur Clique</i>
<i>H</i>	-	<i>CGH</i>	$Clq_1 = \{C, G, H\}$	-
<i>D</i>	-	<i>CD</i>	$Clq_2 = \{C, D\}$	<i>Clq₁</i>
<i>A</i>	-	<i>AB</i>	$Clq_3 = \{A, B\}$	-
<i>B</i>	-	<i>BCE</i>	$Clq_4 = \{B, C, E\}$	<i>Clq₁, Clq₂, Clq₃</i>
<i>G</i>	$\{C, F\}$	<i>CFG</i>	$Clq_5 = \{C, F, G\}$	<i>Clq₁, Clq₂, Clq₄</i>
<i>F</i>	-	<i>CEF</i>	$Clq_6 = \{C, E, F\}$	<i>Clq₁, Clq₂, Clq₄, Clq₅</i>
<i>E</i>	-	<i>CE</i>	-	-
<i>C</i>	-	<i>C</i>	-	-

Tabelle 1.3: Triangulierungsprozeß mit Cliques des Beispielnetzes *Asienbesuch*

Knotenzuordnung und ψ -Tabellen

Im folgenden werden die Zufallsvariablen des Bayesschen Netzes den Cliques des Junction Trees zugeordnet. Dazu werden die bedingten Wahrscheinlichkeiten, die mit einer Zufallsvariablen assoziiert sind, zu einer bestimmten Tabelle der Clique, der sogenannten ψ -Tabelle, hinzumultipliziert.

Jeder Knoten des Bayesschen Netzes wird eineindeutig einer Clique Clq seines Junction Trees \mathcal{T} zugeordnet. Ein Knoten kann in mehreren Cliques liegen aber nur einer Clique zugeordnet sein, d.h.

$$\begin{aligned} \exists f: \mathbf{V} &\rightarrow \mathbf{C} \quad \text{mit} \\ K &\mapsto f(K) = Clq_i \in \mathbf{C}. \end{aligned}$$

In Worten: Es gibt eine Funktion f , die für jeden Knoten $K \in \mathbf{V}$ des Graphen \mathcal{G} angibt, welcher Clique des zugehörigen *Junction-Trees* er zugeordnet ist.

In der Clique, die einem Knoten zugeordnet wird, müssen auch die Eltern des Knotens liegen, d.h. $\{K\} \cup pa(K) \subseteq f(K) = Clq_i$ (Die Elternknoten müssen der Clique aber nicht zugeordnet sein.). Durch die Moralisierung des Graphen des Bayesschen Netzes im Anfang der Konstruktion des Junction Trees wird sichergestellt, daß ein Knoten mit seinen Elternknoten in wenigstens einer Clique des Junction Trees zusammen zu liegen kommt.

Den einzelnen Cliques des Junction Trees sind bestimmte Tabellen zugeordnet, die im Anfang mit Einsen vorbelegt sind. Die Dimension einer Tabelle ist jeweils so bemessen, daß für jede Kombination von Knotenzuständen der Knoten, die in der Clique enthaltenen sind, ein Eintrag möglich ist. Diese Tabelle wird auch als ψ -Tabelle bezeichnet. Dieser Tabelle werden die bedingten Wahrscheinlichkeiten und die a-priori Wahrscheinlichkeiten der Knoten hinzumultipliziert, die der Clique durch die Funktion $f: \mathbf{V} \rightarrow \mathbf{C}$ zugeordnet sind.

Für diese Multiplikation werden im folgenden Abschnitt spezielle Tabellen über Knotenmengen eingeführt, die die bedingten Wahrscheinlichkeiten und a-priori Wahrscheinlichkeiten der Knoten repräsentieren. Auf ihnen werden dann die Multiplikation und weitere Operationen definiert, die für die folgenden Inferenzalgorithmen notwendig sind. Welche Tabellen den Separatorknoten zugeordnet werden und welche zusätzlichen

Tabellen den Cliques dann hinzumultipliziert werden, wird in den beiden Abschnitten 1.3.3.1 und 1.3.3.2 der entsprechenden Inferenzalgorithmen besprochen, da diese unterschiedlich vorgehen.

Der zugehörige Junction Tree \mathcal{T} eines Bayesschen Netzes $\mathcal{BN} = (\mathbf{V}, \mathbf{E}, \mathbf{X}, P)$ läßt sich nun als ein 6-Tupel $\mathcal{T} = (\mathbf{V}, \mathbf{C}, \mathbf{E}_{Sep}, \mathbf{X}, P, f)$ schreiben, mit der Bedeutung der Symbole wie sie eingeführt wurden.

1.3.2 Algebra von Wahrscheinlichkeitstabellen

Die a-priori und bedingten Wahrscheinlichkeiten der Zufallsvariablen müssen dem Knoten zugeordnet werden, der mit der Zufallsvariablen assoziiert ist. In dieser Arbeit geschieht dies durch *Wahrscheinlichkeitstabellen*, die die a-priori oder bedingten Wahrscheinlichkeiten jeweils einer Zufallsvariablen des Bayesschen Netzes zusammenfaßt. Wahrscheinlichkeitstabellen sind bestimmte Tabellen über Knotenmengen. Alle notwendigen Berechnungen können als Berechnungen von Tabellen über Knotenmengen angegeben werden. Im folgenden wird kurz beschrieben, welche Operationen mit Tabellen über Knotenmengen möglich sind und gebraucht werden, und welche besonderen Tabellen über Knotenmengen existieren. Genauere Informationen sind im Anhang B.1 nachzulesen. Hier wird zum Beispiel darauf eingegangen, wie die Operationen ausgeführt werden und was beachtet werden kann, um die Operationen effizient (bzgl. Speicherverbrauch und Rechengeschwindigkeit) auszuführen.

Eine Tabelle über einer Knotenmenge $\mathbf{V} = \{K_1, \dots, K_n\}$ ist eine Tabelle, die für jede Konfiguration (k_1, \dots, k_n) einen Eintrag enthält. Sie wird mit $T(K_1, \dots, K_n)$ bezeichnet. Mit $T(k_{1,j_1}, \dots, k_{n,j_n})$ oder $T(K_1, \dots, K_n)_{j_1, \dots, j_n}$ kann auf eine Konfiguration $(k_{1,j_1}, \dots, k_{n,j_n})$ zugegriffen werden. Ist es eindeutig, für welchen Knoten ein Hypothesenname steht, so können die Hypothesennamen für den Zugriff auf die Tabelle auch in beliebiger Reihenfolge angegeben werden.

Mit $T_{1(K_1, \dots, K_n)}$ wird die Tabelle über der Knotenmenge $\{K_1, \dots, K_n\}$ bezeichnet, die nur Einsen als Einträge in der Tabelle besitzt.

Eine Tabelle bedingter Wahrscheinlichkeiten des Knotens K ist eine Tabelle $T_{\text{cpt}(K)}$ über der Knotenmenge $\{K\} \cup \text{pa}(K)$.

Eine Tabelle gemeinsamer Wahrscheinlichkeiten der Knoten K_1, \dots, K_n ist eine Tabelle $T(K_1, \dots, K_n)$ über der Knotenmenge $\mathbf{V} = \{K_1, \dots, K_n\}$. Um bei einer Tabelle gemeinsamer Wahrscheinlichkeiten sicher zu stellen, daß sich die Tabelleneinträge insgesamt zu Eins addieren, wird die Tabelle mit einem Normalisierungsfaktor versehen. Dieser Normalisierungsfaktor wird auch als *Normalisierungskonstante* α bezeichnet.

Ein finding eines Knotens K ist eine eindimensionale Tabelle $T_{\text{finding}(K)}$ über dem Knoten K , die durch eine 0 an der entsprechenden Stelle angibt, daß diese Hypothese des Knotens K nicht mehr eintreten wird. Hypothesen, die noch eintreten können, sind durch eine 1 an der entsprechenden Stelle der Tabelle markiert. Zum Beispiel gibt die Tabelle $T_{\text{finding}(E)} = \begin{bmatrix} \epsilon_1 & \epsilon_2 & \epsilon_3 \\ 0 & 1 & 1 \end{bmatrix}$ an, daß nur noch die Hypothesen ϵ_2 und ϵ_3 des Knotens E eintreten können. Die Hypothese ϵ_1 des Knotens E wird nicht mehr eintreten.

Die oben vorgestellten Tabellen werden zusammenfassend auch als *Wahrscheinlichkeitentabellen* bezeichnet.

Jetzt folgen die verschiedenen Operationen, die auf den Tabellen und die mit den Tabellen notwendig sind. Dazu zählen die *Multiplikation*, die *Division* sowie die *Marginalisation* von Tabellen.

Die Multiplikation zweier Tabellen ist abelsch. Sind die beiden Tabellen $T(A, B)$ und $T(A, C)$ über der Knotenmenge $\{A, B\}$ bzw. $\{A, C\}$ gegeben, so ist das Produkt der beiden Tabellen eine Tabelle $T(A, B, C)$ über der Knotenmenge $\{A, B\} \cup \{A, C\} = \{A, B, C\}$. Für eine Konfiguration (a_i, b_j, c_k) berechnet sich der Tabelleneintrag von $T(A, B, C)$ zu: $T(A, B, C)_{a_i, b_j, c_k} = T(A, B)_{a_i, b_j} \cdot T(A, C)_{a_i, c_k}$.

Die Division der Tabellen $T(A, B)$ über der Knotenmenge $\{A, B\}$ durch die Tabelle $T(A, C)$ über der Knotenmenge $\{A, C\}$ ist wie bei der Multiplikation dieser beiden Tabellen eine Tabelle $T(A, B, C)$ über der Knotenmenge $\{A, B\} \cup \{A, C\} = \{A, B, C\}$. Für eine Konfiguration (a_i, b_j, c_k) berechnet sich der Tabelleneintrag von $T(A, B, C)$ zu: $T(A, B, C)_{a_i, b_j, c_k} = T(A, B)_{a_i, b_j} / T(A, C)_{a_i, c_k}$. Ist $T(A, C)_{a_i, c_k} = 0$, so muß auch $T(A, B)_{a_i, b_j}$ gleich 0 sein, und der Wert von $T(A, B, C)_{a_i, b_j, c_k}$ wird auf 0 gesetzt, d.h. es gilt: $\frac{0}{0} = 0$.

Um aus einer Tabelle $T(A, B, C, D)$ über der Knotenmenge $\{A, B, C, D\}$ Informationen über einer Teilmenge von Knoten –zum Beispiel $\{B, C\}$ – zu erhalten, so wird über die restlichen Knoten der Tabelle addiert. Man erhält eine Tabelle $T(B, C)$ über der Knotenmenge $\{B, C\}$: $T(B, C) = \sum_{\{A, D\}} T(A, B, C, D)$. Man bezeichnet dies auch als *Marginalisation*.

In den folgenden Abschnitten werden zwei verschiedene Architekturen vorgestellt, die die Algebra von Wahrscheinlichkeitentabellen für den Inferenzalgorithmus benötigen.

1.3.3 Inferenzverfahren

Berechnungen auf dem Junction Tree, um die Wahrscheinlichkeiten für das Eintreffen gewisser Hypothesen zu ermitteln, werden auch als *Inferenz* oder *Propagierung* bezeichnet. Es gibt verschiedene Inferenzverfahren, wie zum Beispiel die Aalborg Architektur oder die Shafer-Shenoy Architektur, die beide in den nächsten Abschnitten vorgestellt werden.

Für die Bestimmung der Cliques und des *Junction Trees* sind dieselben Techniken anzuwenden, wie sie in Abschnitt 1.3.1 dargestellt sind. Sei nun der *Junction Tree* für das Bayessche Netz gegeben, das berechnet werden soll. Der Inferenzmechanismus gliedert sich bei beiden Architekturen in die zwei Phasen

1. *Collect Evidence* und
2. *Distribute Evidence*,

die bei der *Initialisierung* des Junction Trees (*Init*) oder der *Auffrischung* des Junction Trees (*Update*) nach dem Eintragen neuer Beobachtungen hintereinander ausgeführt werden.

Bevor die 1. Phase *Collect Evidence* aufgerufen wird, wird noch der *aufzufende Knoten* bestimmt. Dazu wird ein Knoten R zufällig als Wurzelknoten gewählt. Der

Junction Tree \mathcal{T} wird somit vom Knoten R zu den Blattknoten gerichtet. Alle Knoten außer dem Knoten R haben damit genau einen Elternknoten.

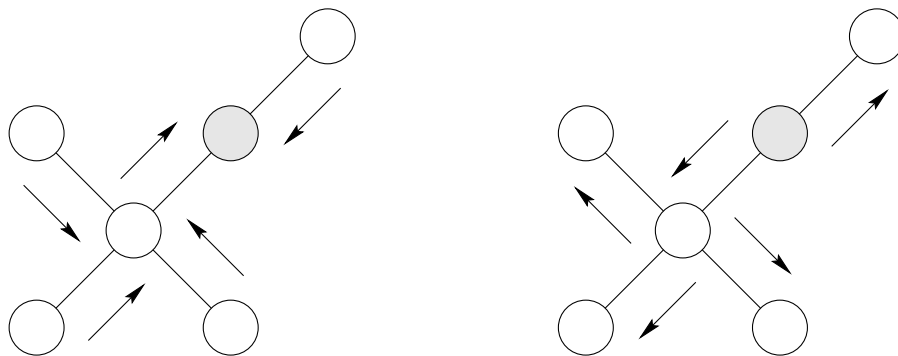
Die 1. Phase *Collect Evidence* läuft dabei wie folgt ab (siehe auch Abbildung 1.10(a)):

- Die Blattknoten schicken ihre Nachricht zum aufrufenden Knoten.
- Ein innerer Knoten schickt die Nachricht erst weiter, wenn er von allen Nachbarknoten bis auf einen die Nachricht erhalten hat. Die Nachricht wird an den Knoten⁸ geschickt, von dem er noch keine Nachricht erhalten hat.
- Durch *Absorption* wird die Nachricht bestimmt, die verschickt wird.

In der 2. Phase *Distribute Evidence* wird vom aufrufenden Knoten wie folgt eine Nachricht an die Blattknoten gesendet (siehe auch Abbildung 1.10(b)):

- Der aufrufende Knoten schickt seine Nachricht zu den Blattknoten des Baumes.
- Ein innerer Knoten erhält von einem Knoten die Nachricht. Die Nachricht wird an die Nachbarknoten weiter geschickt von denen er keine Nachricht erhalten hat⁹.
- Die Nachricht, die verschickt wird, wird durch *Absorption* bestimmt.

Die beiden Architekturen unterscheiden sich in der *Absorption*, die in den entsprechenden folgenden Abschnitten besprochen wird.



(a) Collect Evidence: Die Blattknoten schicken ihre Nachricht zum aufrufenden Knoten.

(b) Distribute Evidence: Der aufrufende Knoten schickt seine Nachricht an die Blattknoten.

Abbildung 1.10: Die beiden Phasen Collect und Distribute Evidence. Der aufrufende Knoten ist grau schraffiert dargestellt.

Nach einem vollen Durchlauf von Initialisierung oder Auffrischung des Junction Trees sind die Informationen konsistent, die in den Cliques des Junction Trees gespeichert sind: Ist man an der Einschätzung (BEL-Wert) über einer Knotenmenge \mathbf{K}

⁸Dieser Knoten ist der Elternknoten des inneren Knotens.

⁹Diese Knoten sind die Kinderknoten des inneren Knotens.

interessiert, so kann man diese aus jeder Clique des Junction Trees gewinnen, in der die Knotenmenge enthalten ist. Man erhält immer dasselbe Ergebnis. Man sagt auch: Die Cliques des Junction Trees sind zueinander *kalibriert*.

Ist man also an den genauen BEL-Werten für alle Knoten interessiert, so müssen in der zweiten Phase (Distribute Evidence) alle Äste des *Junction Trees* besucht werden. Hat man jedoch nur ein Interesse an BEL-Werten gewisser Knoten, so kann man bei Distribute Evidence die Äste des *Junction Trees* aus Performanz-Gründen auslassen, in denen sich nur noch Knoten befinden, deren BEL-Werte nicht benötigt werden. (Dieses wird später bewußt beim *Rollup* (siehe Kapitel 3) ausgenutzt, da hier die alten Cliques, die den alten Zeitscheiben zugeordnet sind, bei Distribute Evidence ausgelassen werden.).

1.3.3.1 Aalborg Architektur

Im folgenden Abschnitt wird gezeigt, wie sich die Informationen in der Aalborg Architektur berechnen, die bei Collect Evidence und Distribute Evidence von Clique zu Clique verschickt werden.

Damit jede Clique im Junction Tree konsistente Informationen besitzt, werden Nachrichten im Junction Tree verschickt. Das Versenden einer Nachricht von einer Clique zu einer anderen Clique wird *Absorption* genannt. In der *Aalborg Architektur* (siehe [Andersen et al. 89] und [Jensen 96]) wird die *Absorption* wie folgt durchgeführt.

Definition 1.3.10 (Absorption (Aalborg))

Seien Clq_i und Clq_ℓ benachbarte Cliques mit dem Separator $Sep_{i,\ell}$, und seien $T_{\psi(Clq_i)}$, $T_{\psi(Clq_\ell)}$ und $T_{\psi(Sep_{i,\ell})}$ die zugehörigen Wahrscheinlichkeitentabellen über den entsprechenden Knotenmengen.

- Berechne die Tabelle $T_{\psi(Sep_{i,\ell})}^* = \sum_{Clq_i \setminus Sep_{i,\ell}} T_{\psi(Clq_i)}$. (Marginalisation)
- Berechne die Tabelle $T_{\psi(Clq_\ell)}^* = T_{\psi(Clq_\ell)} \cdot \frac{T_{\psi(Sep_{i,\ell})}^*}{T_{\psi(Sep_{i,\ell})}}$.
- Weise dem Separator $Sep_{i,\ell}$ die Tabelle $T_{\psi(Sep_{i,\ell})}^*$ zu.
- Weise der Clique Clq_ℓ die Tabelle $T_{\psi(Clq_\ell)}^*$ zu.

Man sagt, daß die Clique Clq_ℓ von der Clique Clq_i *absorbiert* hat. □

Die Absorption der Clique Clq_ℓ von der benachbarten Clique Clq_i ist in Abbildung 1.11 grafisch dargestellt. Dieser Abbildung und auch der Definition 1.3.10 ist zu entnehmen, welche Tabellen den Cliques $Clq \in \mathbf{C}$ und Separatorknoten $Sep \in \mathbf{S}$ des Junction Trees $\mathcal{T} = (\mathbf{C}, \mathbf{S})$ zugeordnet sind. Der Clique Clq_i wird eine Tabelle über der Knotenmenge Clq_i zugewiesen, die im Anfang nur Einsen als Einträge besitzt. Dieser Tabelle werden nun die Tabellen, die die bedingten Wahrscheinlichkeiten bzw. a-priori Wahrscheinlichkeiten ihrer ihr zugeordneten Knoten repräsentieren, hinzumultipliziert. Die der Clique Clq_i zugeordnete Tabelle wird mit $T_{\psi(Clq_i)}$ bezeichnet. Dem Separatorknoten $Sep_{i,\ell}$, der die Knotenmenge $Clq_i \cap Clq_\ell$ enthält, wird eine Tabelle über der Knotenmenge $Clq_i \cap Clq_\ell$ zugeordnet, die wie vor im Anfang nur Einsen als Einträge enthält und mit $T_{\psi(Sep_{i,\ell})}$ bezeichnet wird. Die Einträge dieser Tabelle ändern

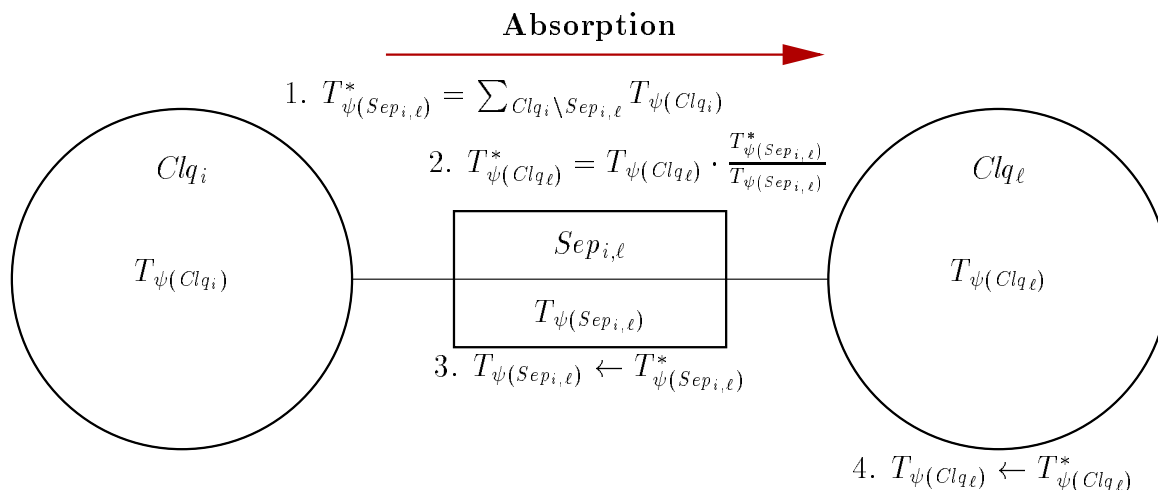


Abbildung 1.11: Aalborg: Absorption der Clique Clq_ℓ von der Clique Clq_i

sich nur durch Absorption. Liegen für Knoten *findings* vor, d.h. daß einige Zustände der dem Knoten zugehörige Zufallsvariable unmöglich geworden sind, werden diese *findings* in der Clique eingetragen, die dem Knoten zugeordnet ist. Ein finding eines Knotens K ist eine eindimensionale Tabelle $T_{\text{finding}(K)}$ über dem Knoten K , die durch eine 0 an der entsprechenden Stelle angibt, daß diese Hypothese des Knotens K nicht mehr eintreten wird. Hypothesen, die noch eintreten können, sind durch eine 1 an der entsprechenden Stelle der Tabelle markiert. Zum Beispiel gibt die Tabelle $T_{\text{finding}(E)} = \begin{bmatrix} e_1 & e_2 & e_3 \\ 0 & 1 & 1 \end{bmatrix}$ an, daß nur noch die Hypothesen e_2 und e_3 des Knotens E eintreten können. Die Hypothese e_1 des Knotens E wird nicht mehr eintreten.

Im Abschnitt A.1 werden an einem Beispiel die beiden Phasen Collect Evidence und Distribute Evidence beim Init und beim Update eines Junction Trees ausführlich vorgestellt.

1.3.3.2 Shafer-Shenoy Architektur

Bei der Shafer-Shenoy Architektur sind nicht wie bei der Aalborg Architektur die an die Clique verschickten Nachrichten in der Clique selbst sondern in den Separatoren gespeichert, die sich in der Nachbarschaft der Clique befinden. Die sich daraus ergebenden Änderungen für die Absorption werden im folgenden Abschnitt vorgestellt.

Bei der *Shafer-Shenoy Architektur* sind im Gegensatz zur *Aalborg Architektur* in jedem Separator zwei Tabellen gespeichert, nämlich jeweils eine für jede Richtung. In der einen Tabelle wird die Nachricht gespeichert, die die erste Clique zur zweiten Clique schickt, und in der anderen Tabelle wird die Nachricht gespeichert, die die zweite Clique zur ersten Clique zurückschickt. In jeder Clique ist eine Tabelle gespeichert.

Die hier vorgestellte *Shafer-Shenoy Architektur* wurde im Vergleich zur Shafer-Shenoy Architektur, wie sie in [Shafer 96] vorgestellt wird, vom Autor so abgewandelt, daß die Tabellen der Cliquen nicht explizit als Tabellen sondern nur als Berechnungsvorschriften (Skripte) gespeichert sind. Dies ist auch ein Unterschied zur Aalborg Architektur, bei der die Tabelle explizit ausgerechnet werden. Dadurch ist bei der Absorption in der Shafer-Shenoy Architektur im Gegensatz zur *Aalborg Architektur* keine

Division notwendig. Als weiterführende Literatur für die Shafer-Shenoy Architektur und den Vergleich mit anderen Algorithmen bieten sich unter anderem [Shafer 96; Madsen & Jensen 98; Xiang & Jensen 99] an.

Aus den Cliques und Separatorknoten lassen sich nach erfolgreicher Propagierung die BEL-Werte der interessierenden Knoten berechnen. Dieses ist in Abbildung 1.12 grafisch dargestellt.

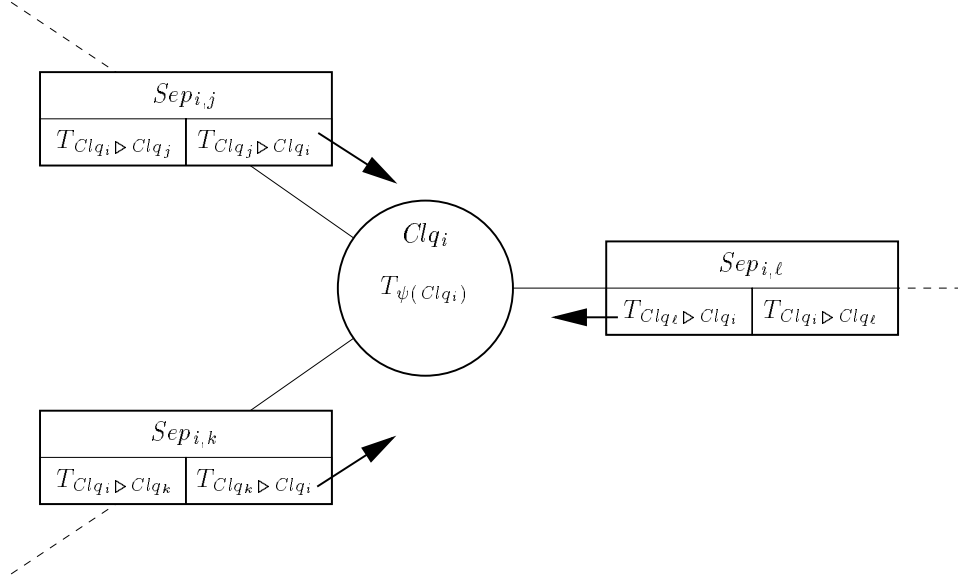


Abbildung 1.12: Shafer-Shenoy: Berechnung der Cliquesinfo nach erfolgreicher Propagierung.

In den Separatoren sind die Informationen ($T_{Clq \triangleright Clq_i}$) gespeichert, die die benachbarten Cliques an Clique Clq_i geschickt haben. Heißt der Separatorknoten $Sep_{i,\ell}$, so stehen in ihm die beiden Tabellen $T_{Clq_i \triangleright Clq_\ell}$ und $T_{Clq_\ell \triangleright Clq_i}$. Die Information, die die Clique Clq_i an die Clique Clq_ℓ schickt, ist eine Tabelle $T_{Clq_i \triangleright Clq_\ell}$ über der Knotenmenge $Clq_i \cap Clq_\ell$. Die Clique Clq_ℓ wird dabei von der Clique Clq_i über die Knoten informiert, die beide Cliques gemeinsam haben (nämlich $Clq_i \cap Clq_\ell$).

In der Clique Clq_i sind die Informationen ($T_{\text{cpt}(\cdot)}$ und $T_{\text{finding}(\cdot)}$) der Knoten gespeichert, die der Clique durch die Funktion $f : \mathbf{V} \rightarrow \mathbf{C}$ zugeordnet sind.

Somit ergibt sich als ψ -Tabelle für die Clique Clq_i :

$$T_{\psi(Clq_i)} = T_1(Clq_i) \cdot \prod_{K \in \{K | f(K) = Clq_i\}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \cdot \prod_{Clq \in \text{adj}(Clq_i)} T_{Clq \triangleright Clq_i}.$$

Für die ψ -Tabelle der Clique Clq_i ohne die Nachricht der benachbarten Clique Clq_ℓ schreibt man auch $T_{\psi(Clq_i) \setminus Clq_\ell}$ und meint damit:

$$T_{\psi(Clq_i) \setminus Clq_\ell} = T_1(Clq_i) \cdot \prod_{K \in \{K | f(K) = Clq_i\}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \cdot \prod_{Clq \in \text{adj}(Clq_i) \setminus Clq_\ell} T_{Clq \triangleright Clq_i}.$$

Ist der Clique Clq_i kein Knoten zugeordnet, so gilt:

$$\prod_{K \in \{K | f(K) = Clq_i\}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) = \prod_{K \in \{\}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) = 1.$$

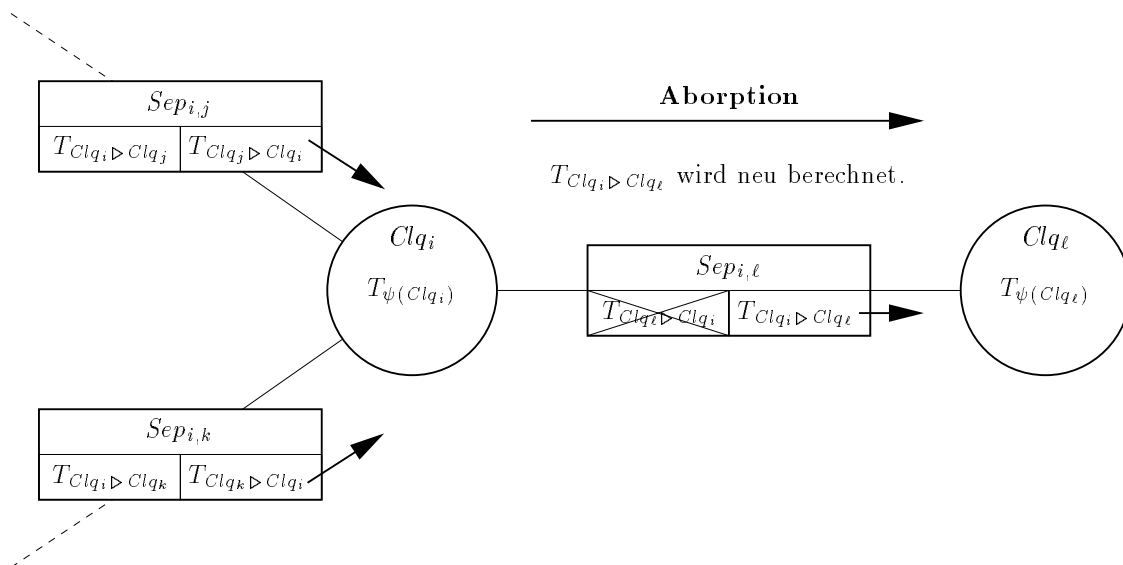


Abbildung 1.13: Shafer-Shenoy: Absorption der Clique Clq_ℓ von der Clique Clq_i

In der Shafer-Shenoy Architektur wird die Absorption von der Clique Clq_i zur Clique Clq_ℓ wie folgt durchgeführt.

Definition 1.3.11 (Absorption (Shafer-Shenoy))

Seien Clq_i und Clq_ℓ benachbarte Cliques mit dem Separator $Sep_{i,\ell}$. Die benachbarten Cliques von Clq_i seien mit $\text{adj}(Clq_i)$ bezeichnet. Die verschiedenen Wahrscheinlichkeitstabellen seien wie üblich definiert. Dann berechnet sich die Nachricht von Clq_i zu Clq_ℓ wie folgt:

$$\begin{aligned}
 T_{Clq_i \triangleright Clq_\ell} &= \\
 &= \sum_{Clq_i \setminus Clq_\ell} T_{\psi(Clq_i) \setminus Clq_\ell} = \\
 &= \sum_{Clq_i \setminus Clq_\ell} \left(T_{1(Clq_i)} \cdot \prod_{K \in \{K | f(K) = Clq_i\}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \cdot \prod_{Clq \in \text{adj}(Clq_i) \setminus Clq_\ell} T_{Clq \triangleright Clq_i} \right).
 \end{aligned}$$

□

Man beachte, daß im Vergleich zur Hugin-Propagierung keine Division erfolgt. Die Tabelle, durch die bei der Hugin-Propagierung dividiert wird, wird in der Shafer-Shenoy Architektur gar nicht erst hinzu multipliziert.

Dadurch, daß keine Division erforderlich ist, wird die Implementierung einer parametrisierten Propagierung (siehe Abschnitt 1.3.3.4) in Bayesschen Netzen sehr vereinfacht.

Im Anhang A.2 wird eine ausführliche Beispielrechnung für die Shafer-Shenoy Architektur durchgeführt. Um dabei die Shafer-Shenoy Architektur mit der Aalborg Architektur vergleichen zu können, wurde dasselbe Beispiel wie bei der Beispielrechnung für die Aalborg Architektur (siehe A.1) herangezogen.

Wie in [Shafer 96] ausgeführt wird, ist die Shafer-Shenoy Architektur bei der Multiplikation ziemlich verschwenderisch. Jeder Knoten des Junction Trees schickt eine Nachricht an seine Nachbarknoten. Diese Nachricht wird zwar nur einmal berechnet,

aber bei der Berechnung der Nachricht an einen Nachbarknoten werden einige Multiplikationen noch einmal ausgeführt, die schon bei der Berechnung der Nachricht für einen anderen Nachbarknoten angefallen sind. In der *Aalborg Architektur* wird hingegen die Ersparnis von Multiplikationen durch einen erhöhten Speicherplatzbedarf der Tabellen in den Cliques erkauft.

In den nachfolgenden Abschnitten werden Verfahren vorgestellt, die der Autor aus den gegebenen Inferenzverfahren entwickelt hat. Das erste Verfahren (die sog. *Shafer-Shenoy-Skripte*) verbindet die Vorzüge der Aalborg Architektur mit den Vorzügen der Shafer-Shenoy Architektur. Das zweite Verfahren (die sog. *Polynom-Propagierung*) erweitert sowohl die Shafer-Shenoy-Skripte als auch die Shafer-Shenoy Architektur um eine Funktionalität, die bei einigen Rollup-Verfahren in Kapitel 3 benötigt wird.

1.3.3.3 Shafer-Shenoy-Skripte

Im folgenden wird gezeigt, wie man für eine Clique eine Berechnungsvorschrift bestimmen kann, die sich als einfaches Skript darstellen läßt. Desweiteren wird gezeigt, wie sich Berechnungen einsparen lassen, wenn man die Skripte der gesamten Cliques eines Junction Trees auf intelligente Art und Weise gemeinsam löst.

Im Vergleich zur Shafer-Shenoy Architektur werden jetzt im folgenden auch die Tabellen in den Separatoren nicht mehr explizit ausgerechnet, sondern nur noch die Berechnungsvorschriften gespeichert. Nach Kenntnis des Autors ist dies das erste Mal, daß ein solches Vorgehen vorgeschlagen wird. Dabei erhält man Formeln, die zur Lösung an verschiedene Algebraprogramme weitergegeben werden können. Ist man nicht an den genauen Ergebnissen interessiert und erfüllen Terme der gewonnenen Formeln gewisse Voraussetzungen an eine Fehlerabschätzung, so können die Formeln, in denen diese Terme enthalten sind, vereinfacht werden.

Beispiel 1.3.12 (Asienbesuch (Shafer-Shenoy Skripte))

Wie im Beispiel A.2.1 wird der Junction Tree in Abbildung A.1 für das Bayessche Netz *Asienbesuch* in Abb. 1.2 genommen.

Die Initialisierung und der Update verlaufen wie im Beispiel A.2.1 und werden hier nicht mehr dargestellt. Was sich ändert ist die Darstellung der Wahrscheinlichkeitstabellen in den Cliques und Separatorknoten des Junction Trees. Es sind keine Zahlenwerte sondern nur noch Berechnungsvorschriften (Skripte) gespeichert.

Die sich daraus ergebenden Änderungen sind in den Tabellen 1.4 und 1.5 dargestellt. In Tabelle 1.4 sind die ψ -Tabellen der Cliques und \triangleright -Tabellen der Separatorknoten im Verlauf der Initialisierung dargestellt. Die Werte, die sich nach der Auffrischung nach dem Eintrag der *findings* ergeben, sind in der Tabelle 1.5 dargestellt. Die Tabellen, die in den verschiedenen Phasen nicht neu berechnet werden, sind wiederum aus Gründen der Übersichtlichkeit nicht dargestellt. Für die Reihenfolge der Berechnung wird auf die Phasen *Collect Evidence* und *Distribute Evidence* in Abbildung A.4 verwiesen. Die eingekreisten Zahlen erleichtern auch hier wieder die Zuordnung.

Wie sich nach der Initialisierung die Zahlenwerte der Tabellen berechnen lassen, wird in B.3.1 dargestellt:

An dieser Stelle sind die Ergebnisse der ψ -Tabellen der Cliques aufgeführt.

\triangleright -Tab., ψ -Tab.	Berechnungsvorschrift der \triangleright -Tabelle und der ψ -Tabelle bei		
	Zuordnung	Collect Evidence	Distribute Evidence
$T_{\psi}(Cl_{q_1})$	siehe Tab. A.4	① Absorption: $\sum_D T_{\psi}(Cl_{q_1}) \setminus Cl_{q_4}$	⑤ Absorption: $\sum_{EG} T_{\psi}(Cl_{q_4}) \setminus Cl_{q_1}$
$T_{Cl_{q_4} \triangleright Cl_{q_1}}$	-		
$T_{Cl_{q_1} \triangleright Cl_{q_4}}$	-		
$T_{\psi}(Cl_{q_2})$	siehe Tab. A.4	② Absorption: $\sum_H T_{\psi}(Cl_{q_2}) \setminus Cl_{q_4}$	④ Absorption: $\sum_E T_{\psi}(Cl_{q_4}) \setminus Cl_{q_2}$
$T_{Cl_{q_4} \triangleright Cl_{q_2}}$	-		
$T_{Cl_{q_2} \triangleright Cl_{q_4}}$	-		
$T_{\psi}(Cl_{q_3})$	siehe Tab. A.4	③ Absorption: $\sum_F T_{\psi}(Cl_{q_3}) \setminus Cl_{q_4}$	③ Absorption: $\sum_C T_{\psi}(Cl_{q_4}) \setminus Cl_{q_3}$
$T_{Cl_{q_4} \triangleright Cl_{q_3}}$	-		
$T_{Cl_{q_3} \triangleright Cl_{q_4}}$	-		
$T_{\psi}(Cl_{q_4})$	siehe Tab. A.4	④ Absorption: $\sum_G T_{\psi}(Cl_{q_4}) \setminus Cl_{q_5}$	② Absorption: $\sum_B T_{\psi}(Cl_{q_5}) \setminus Cl_{q_4}$
$T_{Cl_{q_5} \triangleright Cl_{q_4}}$	-		
$T_{Cl_{q_4} \triangleright Cl_{q_5}}$	-		
$T_{\psi}(Cl_{q_5})$	siehe Tab. A.4	⑤ Absorption: $\sum_{CE} T_{\psi}(Cl_{q_5}) \setminus Cl_{q_6}$	① Absorption: $\sum_A T_{\psi}(Cl_{q_6}) \setminus Cl_{q_5}$
$T_{Cl_{q_6} \triangleright Cl_{q_5}}$	-		
$T_{Cl_{q_5} \triangleright Cl_{q_6}}$	-		
$T_{\psi}(Cl_{q_6})$	siehe Tab. A.4		

Tabelle 1.4: Shafer-Shenoy Skripte: Tabellen bei der Initialisierung.

$$T_{\psi}(Cl_{q_1}) = T_{\text{cpt}(D)} \cdot \left(\sum_{EG} \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right)$$

$$T_{\psi}(Cl_{q_2}) = T_{\text{cpt}(H)} \cdot \left(\sum_E \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right)$$

$$T_{\psi}(Cl_{q_3}) = T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot \left(\sum_C \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right)$$

\triangleright -Tab., ψ -Tab.	Berechnungsvorschrift der \triangleright -Tabelle und der ψ -Tabelle bei Zuordnung		
		Collect Evidence	Distribute Evidence
$T_{\psi(Cl_{q_1})}$	Hypothese d_2 : $T_{\psi(Cl_{q_1})} \cdot T_{\text{finding}(D)}$	① Absorption: $\sum_D T_{\psi(Cl_{q_1}) \setminus Cl_{q_4}}$	⑤ Absorption: $\sum_{EG} T_{\psi(Cl_{q_4}) \setminus Cl_{q_1}}$
$T_{Cl_{q_4} \triangleright Cl_{q_1}}$ $T_{Cl_{q_1} \triangleright Cl_{q_4}}$	- -		
$T_{\psi(Cl_{q_2})}$	siehe Tab. A.4	② Absorption: $\sum_G T_{\psi(Cl_{q_4}) \setminus Cl_{q_5}}$	④ Absorption: $\sum_E T_{\psi(Cl_{q_4}) \setminus Cl_{q_2}}$
$T_{Cl_{q_4} \triangleright Cl_{q_2}}$ $T_{Cl_{q_2} \triangleright Cl_{q_4}}$	- $\sum_H T_{\psi(Cl_{q_2}) \setminus Cl_{q_4}}$		
$T_{\psi(Cl_{q_3})}$	siehe Tab. A.4	③ Absorption: $\sum_{CE} T_{\psi(Cl_{q_5}) \setminus Cl_{q_6}}$	③ Absorption: $\sum_C T_{\psi(Cl_{q_4}) \setminus Cl_{q_3}}$
$T_{Cl_{q_4} \triangleright Cl_{q_3}}$ $T_{Cl_{q_3} \triangleright Cl_{q_4}}$	- $\sum_F T_{\psi(Cl_{q_3}) \setminus Cl_{q_4}}$		
$T_{\psi(Cl_{q_4})}$	siehe Tab. A.4	① Absorption: $\sum_A T_{\psi(Cl_{q_6}) \setminus Cl_{q_5}}$	② Absorption: $\sum_B T_{\psi(Cl_{q_5}) \setminus Cl_{q_4}}$
$T_{Cl_{q_5} \triangleright Cl_{q_4}}$ $T_{Cl_{q_4} \triangleright Cl_{q_5}}$	- -		
$T_{\psi(Cl_{q_5})}$	siehe Tab. A.4	Hypothese a_1 : $T_{\psi(Cl_{q_1})} \cdot T_{\text{finding}(A)}$	
$T_{Cl_{q_6} \triangleright Cl_{q_5}}$ $T_{Cl_{q_5} \triangleright Cl_{q_6}}$	- -		
$T_{\psi(Cl_{q_6})}$			

Tabelle 1.5: Shafer-Shenoy Skripte: Tabellen bei der Auffrischung. Es wurden die Knoten A und D instantiiert.

$$T_{\psi(Cl_{q_4})} = \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right)$$

$$T_{\psi(Cl_{q_5})} = T_{\text{cpt}(C)} \cdot \left(\sum_G \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \right) \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right)$$

$$T_{\psi(Cl_{q_6})} = T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot \left(\sum_{CE} T_{\text{cpt}(C)} \cdot \left(\sum_G \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \right) \right)$$

Wenn jede Formel für sich alleine ausgewertet wird, ohne darauf zu achten, daß

Berechnungen mehrfach ausgeführt werden, so ist dies damit vergleichbar, daß für jede Clique *Collect Evidence* aufgerufen wird. Bei einer ganzheitlichen Betrachtung kann hingegen der Berechnungsaufwand erheblich reduziert werden.

Es läßt sich nun eine Reihenfolge der Berechnungen von Tabellen bestimmen, so daß so wenige Marginalisationen und Multiplikationen von Tabellen wie möglich ausgeführt werden. Zum Beispiel tritt der Ausdruck $T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)}$ in allen Formeln genau einmal auf. Bei einer geschickten Handhabung muß er insgesamt nur einmal und nicht in jeder Formel neu ausgerechnet werden. Wird nun $T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)}$ ausgerechnet und das Ergebnis mit $\mathbf{1}$ bezeichnet, so erhält man:

$$\begin{aligned} T_{\psi(Cl_{q_1})} &= T_{\text{cpt}(D)} \cdot \left(\sum_{EG} \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F \mathbf{1} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right) \\ T_{\psi(Cl_{q_2})} &= T_{\text{cpt}(H)} \cdot \left(\sum_E \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_F \mathbf{1} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right) \\ T_{\psi(Cl_{q_3})} &= \mathbf{1} \cdot \left(\sum_C \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right) \\ T_{\psi(Cl_{q_4})} &= \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F \mathbf{1} \right) \cdot \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \\ T_{\psi(Cl_{q_5})} &= T_{\text{cpt}(C)} \cdot \left(\sum_G \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F \mathbf{1} \right) \right) \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \\ T_{\psi(Cl_{q_6})} &= T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot \left(\sum_{CE} T_{\text{cpt}(C)} \cdot \left(\sum_G \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F \mathbf{1} \right) \right) \right) \end{aligned}$$

Ebenso lassen sich weitere Vereinfachungen erreichen, wie zum Beispiel mit:

$$\mathbf{2} = T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)}, \quad \mathbf{3} = T_{\text{cpt}(C)}, \quad \mathbf{4} = T_{\text{cpt}(D)} \quad \text{und} \quad \mathbf{5} = T_{\text{cpt}(H)}.$$

Man erhält:

$$\begin{aligned} T_{\psi(Cl_{q_1})} &= \mathbf{4} \cdot \left(\sum_{EG} \left(\sum_H \mathbf{5} \right) \cdot \left(\sum_F \mathbf{1} \right) \cdot \left(\sum_B \mathbf{3} \cdot \left(\sum_A \mathbf{2} \right) \right) \right) \\ T_{\psi(Cl_{q_2})} &= \mathbf{5} \cdot \left(\sum_E \left(\sum_D \mathbf{4} \right) \cdot \left(\sum_F \mathbf{1} \right) \cdot \left(\sum_B \mathbf{3} \cdot \left(\sum_A \mathbf{2} \right) \right) \right) \\ T_{\psi(Cl_{q_3})} &= \mathbf{1} \cdot \left(\sum_C \left(\sum_D \mathbf{4} \right) \cdot \left(\sum_H \mathbf{5} \right) \cdot \left(\sum_B \mathbf{3} \cdot \left(\sum_A \mathbf{2} \right) \right) \right) \\ T_{\psi(Cl_{q_4})} &= \left(\sum_D \mathbf{4} \right) \cdot \left(\sum_H \mathbf{5} \right) \cdot \left(\sum_F \mathbf{1} \right) \cdot \left(\sum_B \mathbf{3} \cdot \left(\sum_A \mathbf{2} \right) \right) \\ T_{\psi(Cl_{q_5})} &= \mathbf{3} \cdot \left(\sum_G \left(\sum_D \mathbf{4} \right) \cdot \left(\sum_H \mathbf{5} \right) \cdot \left(\sum_F \mathbf{1} \right) \right) \cdot \left(\sum_A \mathbf{2} \right) \\ T_{\psi(Cl_{q_6})} &= \mathbf{2} \cdot \left(\sum_{CE} \mathbf{3} \cdot \left(\sum_G \left(\sum_D \mathbf{4} \right) \cdot \left(\sum_H \mathbf{5} \right) \cdot \left(\sum_F \mathbf{1} \right) \right) \right) \end{aligned}$$

Wie man leicht sieht, werden manche Berechnungen in mehreren Formeln benötigt. Auch diese lassen sich wieder zentral berechnen, um dann eingetragen zu werden. Ein schönes Beispiel ist die Berechnung $(\sum_A \mathbf{2})$, die mit $\mathbf{6}$ bezeichnet wird. Insgesamt

ergeben sich folgende Berechnungen, die sich lohnen, zentral ausgerechnet zu werden:

$$\boxed{6} = \left(\sum_A \boxed{2} \right), \quad \boxed{7} = \left(\sum_D \boxed{4} \right), \quad \boxed{8} = \left(\sum_H \boxed{5} \right) \quad \text{und} \quad \boxed{9} = \left(\sum_F \boxed{11} \right).$$

Dieses liefert:

$$\begin{aligned} T_{\psi(Cl_{q_1})} &= \boxed{4} \cdot \left(\sum_{EG} \boxed{8} \cdot \boxed{9} \cdot \left(\sum_B \boxed{3} \cdot \boxed{6} \right) \right) & T_{\psi(Cl_{q_4})} &= \boxed{7} \cdot \boxed{8} \cdot \boxed{9} \cdot \left(\sum_B \boxed{3} \cdot \boxed{6} \right) \\ T_{\psi(Cl_{q_2})} &= \boxed{5} \cdot \left(\sum_E \boxed{7} \cdot \boxed{9} \cdot \left(\sum_B \boxed{3} \cdot \boxed{6} \right) \right) & T_{\psi(Cl_{q_5})} &= \boxed{3} \cdot \left(\sum_G \boxed{7} \cdot \boxed{8} \cdot \boxed{9} \right) \cdot \boxed{6} \\ T_{\psi(Cl_{q_3})} &= \boxed{1} \cdot \left(\sum_C \boxed{7} \cdot \boxed{8} \cdot \left(\sum_B \boxed{3} \cdot \boxed{6} \right) \right) & T_{\psi(Cl_{q_6})} &= \boxed{2} \cdot \left(\sum_{CE} \boxed{3} \cdot \left(\sum_G \boxed{7} \cdot \boxed{8} \cdot \boxed{9} \right) \right) \end{aligned}$$

Aus diesen Formeln lassen sich zum Beispiel die folgenden Einsparnisse erzielen:

$$\boxed{10} = \boxed{3} \cdot \boxed{6} \qquad \boxed{11} = \boxed{7} \cdot \boxed{8}$$

Anstatt $\boxed{7} \cdot \boxed{8}$ hätte auch $\boxed{7} \cdot \boxed{9}$ oder $\boxed{8} \cdot \boxed{9}$ genommen werden können. Als Ergebnis erhält man:

$$\begin{aligned} T_{\psi(Cl_{q_1})} &= \boxed{4} \cdot \left(\sum_{EG} \boxed{8} \cdot \boxed{9} \cdot \left(\sum_B \boxed{10} \right) \right) & T_{\psi(Cl_{q_4})} &= \boxed{11} \cdot \boxed{9} \cdot \left(\sum_B \boxed{10} \right) \\ T_{\psi(Cl_{q_2})} &= \boxed{5} \cdot \left(\sum_E \boxed{7} \cdot \boxed{9} \cdot \left(\sum_B \boxed{10} \right) \right) & T_{\psi(Cl_{q_5})} &= \boxed{10} \cdot \left(\sum_G \boxed{11} \cdot \boxed{9} \right) \\ T_{\psi(Cl_{q_3})} &= \boxed{1} \cdot \left(\sum_C \boxed{11} \cdot \left(\sum_B \boxed{10} \right) \right) & T_{\psi(Cl_{q_6})} &= \boxed{2} \cdot \left(\sum_{CE} \boxed{3} \cdot \left(\sum_G \boxed{11} \cdot \boxed{9} \right) \right) \end{aligned}$$

Mit

$$\boxed{12} = \left(\sum_B \boxed{10} \right) \qquad \boxed{13} = \boxed{11} \cdot \boxed{9}$$

erhält man:

$$\begin{aligned} T_{\psi(Cl_{q_1})} &= \boxed{4} \cdot \left(\sum_{EG} \boxed{8} \cdot \boxed{9} \cdot \boxed{12} \right) & T_{\psi(Cl_{q_4})} &= \boxed{13} \cdot \boxed{12} \\ T_{\psi(Cl_{q_2})} &= \boxed{5} \cdot \left(\sum_E \boxed{7} \cdot \boxed{9} \cdot \boxed{12} \right) & T_{\psi(Cl_{q_5})} &= \boxed{10} \cdot \left(\sum_G \boxed{13} \right) \\ T_{\psi(Cl_{q_3})} &= \boxed{1} \cdot \left(\sum_C \boxed{11} \cdot \boxed{12} \right) & T_{\psi(Cl_{q_6})} &= \boxed{2} \cdot \left(\sum_{CE} \boxed{3} \cdot \left(\sum_G \boxed{13} \right) \right) \end{aligned}$$

Mit

$$\boxed{14} = \boxed{9} \cdot \boxed{12} \qquad \boxed{15} = \left(\sum_G \boxed{13} \right)$$

erhält man:

$$T_{\psi(Cl_{q_1})} = \boxed{4} \cdot \left(\sum_{EG} \boxed{8} \cdot \boxed{14} \right) \qquad T_{\psi(Cl_{q_4})} = \boxed{13} \cdot \boxed{12}$$

$$T_{\psi}(Cl_{q_2}) = \boxed{5} \cdot \left(\sum_E \boxed{7} \cdot \boxed{14} \right)$$

$$T_{\psi}(Cl_{q_5}) = \boxed{10} \cdot \boxed{15}$$

$$T_{\psi}(Cl_{q_3}) = \boxed{1} \cdot \left(\sum_C \boxed{11} \cdot \boxed{12} \right)$$

$$T_{\psi}(Cl_{q_6}) = \boxed{2} \cdot \left(\sum_{CE} \boxed{3} \cdot \boxed{15} \right)$$

Die Berechnungen von $\boxed{1}$ bis $\boxed{5}$ können parallel ausgeführt werden. Dasselbe gilt für $\boxed{6}$ bis $\boxed{9}$, $\boxed{10}$ und $\boxed{11}$, $\boxed{12}$ und $\boxed{13}$ bzw. $\boxed{14}$ und $\boxed{15}$.

Die Shafer-Shenoy-Skripte vereinigen die Vorteile der Aalborg Architektur und der Shafer-Shenoy Architektur. Im Gegensatz zur Aalborg Architektur wird keine Division durchgeführt. Im Vergleich mit der Shafer-Shenoy Architektur werden einige Berechnungen nicht mehrfach ausgeführt.

Ist man nun an den BEL-Werten der Knoten des Bayesschen Netzes interessiert, so betrachtet man die ψ -Tabelle der Clique, in der derjenige Knoten vorkommt, und marginalisiert alle anderen Knoten aus dieser Tabelle aus. Im Beispiel sähe das wie folgt aus:

$$\text{BEL}(A) = \sum_B T_{\psi}(Cl_{q_6})$$

$$\text{BEL}(E) = \sum_{FG} T_{\psi}(Cl_{q_3})$$

$$\text{BEL}(B) = \sum_A T_{\psi}(Cl_{q_6})$$

$$\text{BEL}(F) = \sum_{EG} T_{\psi}(Cl_{q_3})$$

$$\text{BEL}(C) = \sum_{BE} T_{\psi}(Cl_{q_5})$$

$$\text{BEL}(G) = \sum_{EF} T_{\psi}(Cl_{q_3})$$

$$\text{BEL}(D) = \sum_C T_{\psi}(Cl_{q_1})$$

$$\text{BEL}(H) = \sum_{CG} T_{\psi}(Cl_{q_2})$$

Man erkennt, daß die ψ -Tabelle von der Clique Cl_{q_4} nicht benötigt wird. Ebenso hätte man sich die Berechnung der ψ -Tabelle von Clique Cl_{q_5} ersparen können, denn der BEL-Wert von C läßt sich auch mit der ψ -Tabelle der Clique Cl_{q_1} bestimmen: $\text{BEL}(C) = \sum_D T_{\psi}(Cl_{q_1})$.

Bsp

1.3.3.4 Polynom-Propagierung

Die Funktionalität der *Shafer-Shenoy Architektur* wird nun dadurch erweitert, daß als Einträge in den Wahrscheinlichkeitstabellen ($T_{\text{cpt}(K)}$, $T_{\text{finding}(K)}$) nicht mehr nur Zahlen sondern allgemeiner sogenannte *multivariate Polynome*¹⁰ erlaubt sind.

In [Skaanning et al. 98] wird ein Bayessche Netz für das System SACSO¹¹ parametrisiert. Um eine Inferenz in parametrisierten Bayesschen Netzen zu ermöglichen, wurde dazu der Hugin Inferenz Algorithmus in Maple V[®] implementiert. Die betrachteten Tabellen der bedingten Wahrscheinlichkeiten für das Bayessche Netz enthalten nur Nullen und Einsen. Indem nur Nullen und Einsen verwendet werden, umgeht man das Problem von Rundungsfehlern.

¹⁰Multivariate Polynome sind Polynome in mehreren Unbestimmten (siehe auch Abschnitt A.3).

¹¹SACSO ist das Akronym für Systems for Automatic Customer Support Operations

Diese Arbeit geht einen anderen Weg. Wie man gesehen hat, benötigt die Shafer-Shenoy Architektur im Vergleich zur Aalborg Architektur keine Division. Da bei Nullen und Einsen keine Rundungsfehler zu erwarten sind, läßt sich die Division von zwei Polynomen in dem Fall ohne große Probleme durchführen. Treten jedoch Rundungsfehler bei Zahlenwerten ungleich Null oder Eins auf, so ist manchmal eine Division unmöglich, da sich das Divisor-Polynom nicht mehr aus dem Dividend-Polynom herausfaktorisieren läßt. Unproblematisch dagegen ist die Polynommultiplikation. Im Rahmen dieser Arbeit wurde eine eigene Polynommultiplikation implementiert, so daß nicht auf externe Mathematikprogramme zurückgegriffen werden muß. Bei der Polynompropagierung auf der Shafer-Shenoy Architektur bereiten kleine Rundungsfehler keine Probleme mehr.

Im Kapitel B.2 werden die multivariaten Polynome formal eingeführt und auch formal die Addition und Multiplikation zweier multivariater Polynome sowie die Skalarmultiplikation einer Konstanten mit einem multivariaten Polynom definiert.

Ein multivariates Polynom ist eine endliche Summe von *Termen*. Ein Term ist ein endliches Produkt von Variablen und einem konstanten Faktor. Zwei multivariate Polynome werden *addiert*, indem die Summe von Termen des ersten multivariaten Polynoms und die Summe von Termen des zweiten multivariaten Polynoms zu einer Summe zusammengefaßt werden. Dabei kann der Ausdruck verkürzt werden, indem man Terme über dieselben Variablen durch Beibehaltung des Produktes der Variablen und Addition der Konstanten zusammenfaßt. Dabei kann sich der konstante Faktor eines Termes zu Null bestimmen. Diese Terme werden aus der Summe gelöscht. Zwei multivariate Polynome werden *multipliziert*, indem man die Produkte aus jedem Term des ersten Polynoms und jedes Termes des zweiten Polynoms aufsummiert.

Zur Verdeutlichung, was man bei der Propagierung von Polynomen als Ergebnis zu erwarten hat, wird im folgenden das bekannte Beispiel 1.1.1 aufgegriffen.

Beispiel 1.3.13 (Hypothesenparametrisierung)

Sei das Bayessche Netz **Gehirntumor** gegeben (siehe den Graph in Abbildung 1.1 und die dazugehörigen a-priori- und bedingten Wahrscheinlichkeiten in Tabelle 1.1). Der Knoten A werde mit $(A1, A2)$ instantiiert, d.h. die Hypothese a_1 des Knotens A wird mit der Variablen $A1$ und die Hypothese a_2 des Knotens A wird mit der Variablen $A2$ belegt. Die Propagierung liefert als Ergebnis die folgenden Belief-Werte der verschiedenen Knoten des Graphen:

$$\begin{aligned} \text{BEL}(A) &= \alpha \begin{pmatrix} 0.20 * A1 \\ 0.80 * A2 \end{pmatrix} \\ \text{BEL}(B) &= \alpha \begin{pmatrix} 0.16 * A1 + 0.16 * A2 \\ 0.04 * A1 + 0.64 * A2 \end{pmatrix} \\ \text{BEL}(C) &= \alpha \begin{pmatrix} 0.04 * A1 + 0.04 * A2 \\ 0.76 * A1 + 0.16 * A2 \end{pmatrix} \\ \text{BEL}(D) &= \alpha \begin{pmatrix} 0.148 * A1 + 0.196 * A2 \\ 0.052 * A1 + 0.604 * A2 \end{pmatrix} \\ \text{BEL}(E) &= \alpha \begin{pmatrix} 0.128 * A1 + 0.488 * A2 \\ 0.072 * A1 + 0.312 * A2 \end{pmatrix} \end{aligned}$$

Die zusätzliche Instantiierung des Knotens C mit $(C1, C2)$ und die anschließende Propagierung liefert als Ergebnis die folgenden Belief-Werte für die verschiedenen

Knoten des Graphen:

$$\begin{aligned} \text{BEL}(A) &= \alpha \begin{pmatrix} 0.04 A1 C1 + 0.16 A1 C2 \\ 0.04 A2 C1 + 0.76 A2 C2 \end{pmatrix} \\ \text{BEL}(B) &= \alpha \begin{pmatrix} 0.032 A1 C1 + 0.128 A1 C2 + 0.008 A2 C1 + 0.152 A2 C2 \\ 0.008 A1 C1 + 0.032 A1 C2 + 0.032 A2 C1 + 0.608 A2 C2 \end{pmatrix} \\ \text{BEL}(C) &= \alpha \begin{pmatrix} 0.04 A1 C1 + 0.04 A2 C1 \\ 0.16 A1 C2 + 0.76 A2 C2 \end{pmatrix} \\ \text{BEL}(D) &= \alpha \begin{pmatrix} 0.0312 A1 C1 + 0.1168 A1 C2 + 0.0288 A2 C1 + 0.1672 A2 C2 \\ 0.0088 A1 C1 + 0.0432 A1 C2 + 0.0112 A2 C1 + 0.5928 A2 C2 \end{pmatrix} \\ \text{BEL}(E) &= \alpha \begin{pmatrix} 0.032 A1 C1 + 0.096 A1 C2 + 0.032 A2 C1 + 0.456 A2 C2 \\ 0.008 A1 C1 + 0.064 A1 C2 + 0.008 A2 C1 + 0.304 A2 C2 \end{pmatrix} \end{aligned}$$

In den Belief-Werten der Knoten stehen nun Polynome $p \in R[A1, A2, C1, C2]$. Die Variablen $A1, A2, C1$ und $C2$ können die Werte 0 und 1 annehmen, dabei muß aber immer wenigstens eine Hypothese eines Knotens eintreffen können, d.h. $\exists i, j \in \{1, 2\} : Ai = 1 \wedge Cj = 1$.

Sei jetzt $C1 = 0$ und $C2 = 1$, d.h. die zweite Hypothese des Knotens C wird instantiiert. Damit ergeben sich die folgenden Belief-Werte:

$$\begin{aligned} \text{BEL}(A) &= \alpha \begin{pmatrix} 0.16 A1 \\ 0.76 A2 \end{pmatrix} \\ \text{BEL}(B) &= \alpha \begin{pmatrix} 0.128 A1 + 0.152 A2 \\ 0.032 A1 + 0.608 A2 \end{pmatrix} \\ \text{BEL}(C) &= \alpha \begin{pmatrix} 0 \\ 0.16 A1 + 0.76 A2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \text{BEL}(D) &= \alpha \begin{pmatrix} 0.1168 A1 + 0.1672 A2 \\ 0.0432 A1 + 0.5928 A2 \end{pmatrix} \\ \text{BEL}(E) &= \alpha \begin{pmatrix} 0.096 A1 + 0.456 A2 \\ 0.064 A1 + 0.304 A2 \end{pmatrix} \end{aligned}$$

Durch die Instantiierung des Knotens C wird der Knoten E vom Knoten A d-separiert, d.h. der Knoten A beeinflusst den Knoten E nicht mehr. Dieses ist aber dem Belief-Wert des Knotens E nicht so ohne weiteres zu entnehmen. Eine genauere Betrachtung liefert:

$$\begin{aligned} \text{BEL}(E) &= \alpha \begin{pmatrix} 0.096 A1 + 0.456 A2 \\ 0.064 A1 + 0.304 A2 \end{pmatrix} = \\ &= \frac{1}{0.096 A1 + 0.456 A2 + 0.064 A1 + 0.304 A2} \begin{pmatrix} 0.096 A1 + 0.456 A2 \\ 0.064 A1 + 0.304 A2 \end{pmatrix} = \\ &= \frac{1}{0.16 A1 + 0.76 A2} \begin{pmatrix} 0.6 * (0.16 A1 + 0.76 A2) \\ 0.4 * (0.16 A1 + 0.76 A2) \end{pmatrix} = \begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix}. \end{aligned}$$

Abschließend wird die Variable $A1$ auf 1 und die Variable $A2$ auf 0 gesetzt. Als Ergebnis erhält man:

$$\text{BEL}(A) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \text{BEL}(B) = \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix},$$

$$\text{BEL}(C) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \text{BEL}(D) = \begin{pmatrix} 0.73 \\ 0.27 \end{pmatrix} \quad \text{und} \quad \text{BEL}(E) = \begin{pmatrix} 0.60 \\ 0.40 \end{pmatrix}.$$

Bsp

Ein vergleichbares Ergebnis erhält man mit *Query DAGs*, die in [Darwiche & Provan 96] beschrieben werden.

Beispiel 1.3.14 (Hypothesenparametrisierung (Query DAGs))

Query-DAGs sind gerichtete azyklische Graphen, deren innere Knoten Operationen (Addition und Multiplikation) und die Blattknoten Zahlenwerte von 0 bis 1 oder Evidenz-Variablen sind, die den Wert 0 oder 1 annehmen können. In Abbildung 1.14 ist ein Bayessches Netz mit seinem zugehörigen Query-DAG (*C* als Evidenz-Variable und *B* als Query-Variable) als Beispiel gegeben.

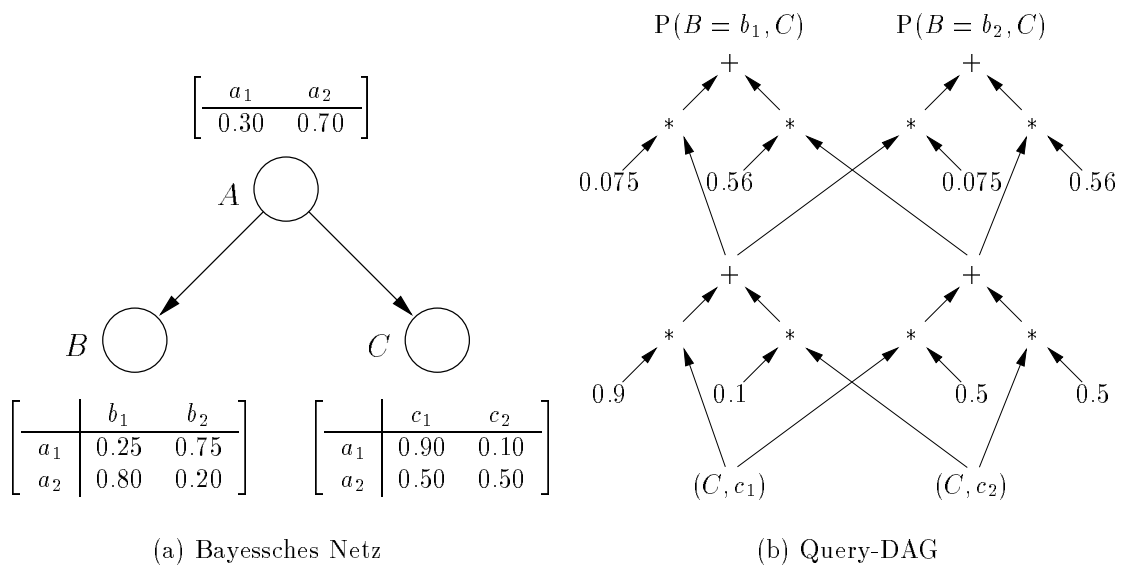


Abbildung 1.14: Beispiel für einen Query-DAG

Ein Query-DAG wird offline bei gegebenem Bayesschem Netz und gegebenen *Evidenz-* und *Query-Variablen* generiert. D.h. die Evidenz- und Query-Variablen sind bekannt aber noch nicht ihre Belegung. Mit dem *Q-DAG Evaluator* kann dann bei gegebener Belegung der Evidenz-Variablen der Query-DAG online ausgewertet werden, und man erhält die Werte für die Query-Variablen. In Abbildung 1.15 ist dieser Zusammenhang grafisch dargestellt. Wie in [Darwiche & Provan 96] beschrieben, werden nur die Wertebelegungen der Zufallsvariablen parametrisiert. Die bedingten Wahrscheinlichkeiten oder die a-priori Wahrscheinlichkeiten werden nicht parametrisiert. Im Gegensatz dazu kann bei der Polynom-Propagierung jeder beliebige Tabelleneintrag einer Wahrscheinlichkeitentabelle parametrisiert werden. Als Ergebnis erhält man Polynome in den Hypothesen der Knoten, deren Zufallsvariable von Interesse ist.

Die Polynompropagierung liefert für den BEL-Wert des Knotens *B* des Bayesschen Netzes in Abbildung 1.14 das folgende Ergebnis:

$$\text{BEL}(B) = \alpha \begin{pmatrix} 0.3475 C1 + 0.2875 C2 \\ 0.2725 C1 + 0.0925 C2 \end{pmatrix}.$$

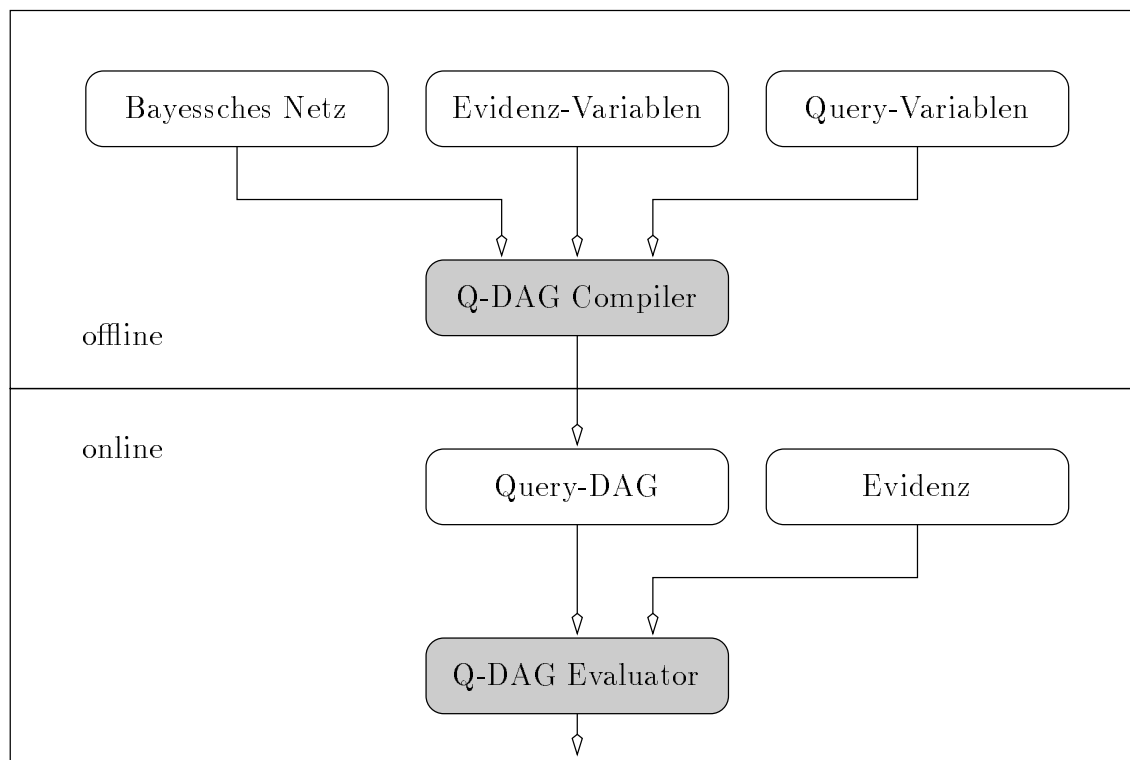


Abbildung 1.15: Generierung von Query-DAGs

Gleichzeitig liegen bei der Polynom-Propagierung auch die Ergebnisse der BEL-Werte der Knoten A und C vor, d.h. für sie müssen keine zusätzlichen Propagierungen durchgeführt werden.

Bei der Vereinfachung oder Evaluierung der Polynome kann auf vorhandene Softwarepakete zurückgegriffen werden. Im Falle der Hypothesenparametrisierung werden die Zahlen sogar nur addiert, je nach dem ob eine 0 als Faktor vorhanden ist oder nicht, so daß eine sehr einfache Auswertung möglich ist.

Sollen andere Zufallsvariablen des Bayesschen Netzes zusätzlich als Evidenz-Variablen parametrisiert werden, so wird bei der Polynom-Propagierung so weit wie möglich auf vorhandene Berechnungen zurückgegriffen. Bei den Query-DAGs wird darüber in [Darwiche & Provan 96] nichts ausgesagt. Bsp

Für weitere Anwendungsbeispiele der Polynom-Propagierung wird auf das Kapitel A.3 verwiesen.

1.4 Zusammenfassung

Anhand von zwei kleinen Beispielen werden die mehrfach verbundenen Bayesschen Netze und einige Begriffe informal eingeführt. Nach dieser Motivation werden die notwendigen Notationen und Begriffe für Bayessche Netze definiert. Dann wird der für die Inferenzverfahren notwendige Junction Tree eingeführt. Dabei wird anhand der in der Motivation vorgestellten Beispielnetze gezeigt, welche Elementarschritte für die Bestimmung eines Junction Trees ausgeführt werden müssen. Man erkennt, wie wichtig dabei die Bestimmung einer optimalen Eliminationsreihenfolge ist. Eine schlechte

Eliminationsreihenfolge induziert eine große Anzahl von Triangulationskanten im moralisierten Graphen, was zur Folge einen Junction Tree haben kann, von dem einige Clique unhabbar groß bzgl. Speicherplatzverbrauch und Berechnungsaufwand sein können. Desweiteren wird eine Algebra von Wahrscheinlichkeitstabellen eingeführt, die eine Schreibweise benutzt, die sich auf die wesentlichen Zusammenhänge bezieht. So wird nicht zwischen bedingten und a-priori Wahrscheinlichkeiten unterschieden und auch für die Instantiierung von Knoten mit bestimmten Werten werden Wahrscheinlichkeitstabellen benutzt. Als Inferenzverfahren werden die Aalborg Architektur und die Shafer-Shenoy Architektur vorgestellt, die beide auf einem Junction Tree arbeiten und exakte Verfahren sind. Die Shafer-Shenoy Architektur wird noch zu Shenoy-Shafer Skripte umgeformt und zur Polynom-Propagierung erweitert. Die Polynom-Propagierung wird dabei durch ein Beispiel ergänzt.

Kapitel 2

Dynamische Bayessche Netze

In diesem Kapitel werden die Dynamischen Bayesschen Netze (DBN) behandelt und die Begriffe und Schreibweisen für Bayessche Netze um die notwendigen Begriffe und Schreibweisen für Dynamische Bayessche Netze erweitert. Obwohl sich der Rollup hier schon andeutet, wird erst in einem späteren Kapitel auf ihn eingegangen werden.

Dynamische Bayessche Netze sind spezielle Bayessche Netze, die sich aus mehreren Bayesschen Netzen (sogenannten Zeitscheiben) aufbauen. Als neue Zeitscheibe wird ein Bayessches Netz aus einer vorgegebenen Menge von Bayesschen Netzen an die letzte Zeitscheibe des Dynamischen Bayesschen Netzes angehängt. Zu den unterschiedlichsten Zeitpunkten lassen sich die verschiedensten Bayesschen Netze an der letzten Zeitscheibe des Dynamischen Bayesschen Netzes anhängen. Dabei lassen sich beim Anhängen von Zeitscheiben zwei Verfahren unterscheiden. Diese zwei Möglichkeiten werden an einem Beispielnetz gegenübergestellt.

Eines der beiden Verfahren für das Anhängen von Zeitscheiben bildet die Grundlagen für einige Rollup-Verfahren, die in 3 vorgestellt werden. Die Idee bei diesem Verfahren ist, daß sich die Struktur des Dynamischen Bayesschen Netzes (der Aufbau des Dynamischen Bayesschen Netzes aus Zeitscheiben) im Junction Tree (siehe Abschnitt 1.3) widerspiegelt.

Ein *dynamisches Bayessches Netz* ist der Spezialfall eines dynamischen probabilistischen Netzes (DPN). Einige Autoren sprechen statt von “dynamischen Netzen” auch von “temporalen Netzen”. Dynamische Probabilistische Netze sind Netze, die in der Lage sind, stochastische temporale Prozesse zu modellieren (vgl. [Dagum et al. 92; Kjærulff 92; Nicholson & Brady 92]). Beispielsweise wird in [Dagum et al. 92] ein dynamisches Netz dazu verwendet, um den Verkauf amerikanischer Wagen auf dem japanischen Markt vorherzusagen. Im Projekt BAT¹ (siehe [Forbes et al. 95]) werden zur Lokalisierung eines Fahrzeuges dynamische Netze eingesetzt.

In einem Dynamischen Bayesschen Netz repräsentiert eine *Zeitscheibe* typischerweise einen Schnappschuß des temporalen Prozesses. Die bedingten Wahrscheinlichkeiten, die die Abhängigkeiten zwischen den Zeitscheiben beschreiben, werden auch als das *Zustandsentwicklungsmodell* bezeichnet.

¹BAT ist das Akronym für Bayesian Automated Taxi.

Im allgemeinen bleibt die Struktur des Teilnetzes in einer Zeitscheibe und die Abhängigkeiten zwischen den Knoten dieser Teilnetze gleich. In [Dagum et al. 92] wird jedoch darauf hingewiesen, daß in einigen Domänen nicht modellierte externe Umstände zu einem Verfall des Modells führen können und eine Aktualisierung der bedingten Wahrscheinlichkeiten oder sogar der Struktur des Netzes verlangen. Ähnlich wird in [Tawfik & Neufeld 94] auf Probleme hingewiesen, die aufgrund von zeitlichen Abläufen entstehen können. So kann beispielsweise die gleiche Evidenz zu verschiedenen Zeitpunkten verschiedene Bedeutungen haben. In diesem Fall ist es notwendig, die bedingten Wahrscheinlichkeiten oder die Struktur der Teilnetze in den entsprechenden Zeitscheiben neu zu definieren.

In einigen Fällen sind die bedingten Wahrscheinlichkeiten sogar zeitabhängig (vergleiche [Tawfik & Neufeld 94]). In [Horvitz & Barry 95] wird ein Beispiel in der Domäne der Weltraumfähre der NASA beschrieben. Wenn es einen Fehler in der Treibstoffversorgung gibt, während das Triebwerk feuert, so ist die Wahrscheinlichkeit, daß die Maschine beschädigt wird, um so höher, je länger das Triebwerk feuert. Gerade bei Krankheiten kann die Wahrscheinlichkeit des Auftretens von Symptomen stark mit dem Fortschreiten der Krankheit schwanken. Zur Behandlung dieses Falls kann man die bedingten Zeitscheiben zeitabhängig definieren, beispielsweise indem man sie durch eine Funktion ausdrückt, die von der Zeit abhängt.

Ein Dynamisches Bayessches Netz kann durch *Aufrollen* (Rollup) vereinfacht werden. Einige Rollup-Verfahren werden dazu in Kapitel 3 vorgestellt. Durch den Rollup kann das Netz so vereinfacht werden, daß höchstens zwei Zeitscheiben gleichzeitig existieren. Auf diese Weise kann die Vorhersage und Interpretation beliebig vieler Beobachtungen erfolgen.

2.1 Motivation

Im folgenden Abschnitt wird an einem Beispiel eine Schranke für Statische Bayessche Netze aufgezeigt. In einem weiteren Beispiel wird man sehen, wie diese Schranke durch Dynamische Bayessche Netze aufgehoben wird.

Für gewisse Anwendungen sind Bayessche Netze, wie sie bisher vorgestellt wurden, nicht sinnvoll. Ihr Einsatz wird durch ihre statische Natur stark eingeschränkt. Dazu soll das folgende Beispiel betrachtet werden, das aus [Kipper et al. 95] entnommen wurde.

Beispiel 2.1.1 (Prüfungssituation statisch)

Dieses Beispiel, das durch die Überlegungen in [Jameson 90] angeregt wurde, ist durch eine Prüfungssituation gegeben. Dem Prüfling wird eine Frage gestellt, die er je nach seinem Wissensniveau und der Schwierigkeit der Frage beantworten kann oder nicht. Je höher das Wissensniveau des Prüflings ist bzw. je leichter die Frage ist, die ihm gestellt wurde, desto größer ist die Wahrscheinlichkeit, daß er die Antwort auf die Frage kennt. Ob er die Antwort auf die Frage kennt, läßt sich anhand seiner Gestik oder Mimik erahnen. Kennt er die Antwort, so wird seine Gestik viel eher ruhig und seine Mimik viel eher gelassen sein, als wenn er sie nicht kennen würde. (Die Gestik und Mimik werden aber erst in späteren Beispielen eine Rolle spielen.)

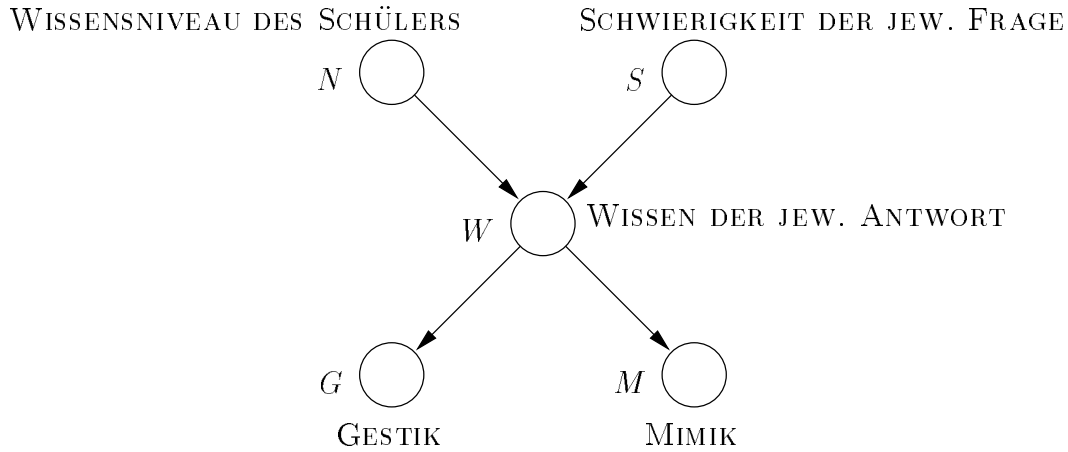


Abbildung 2.1: Beispielnetz Prüfungsfrage statisch (statisches Bayessches Netz)

Die quantitativen Abhängigkeiten sind aus Tabelle 2.1 zu entnehmen.

Das Bayessche Netz eignet sich dazu, um das Wissensniveau eines Prüflings bezüglich einer Frage einzuschätzen.

WISSENSNIVEAU DES SCHÜLERS ($n_1 = \text{hoch}$, $n_2 = \text{niedrig}$)	
$P(n) :$	$P(n_1) = 0.8$ $P(n_2) = 0.2$
SCHWIERIGKEIT DER JEW. FRAGE ($s_1 = \text{schwer}$, $s_2 = \text{mittel}$, $s_3 = \text{leicht}$)	
$P(s) :$	$P(s_1) = 0.6$ $P(s_2) = 0.3$ $P(s_3) = 0.1$
WISSEN DER JEW. ANTWORT ($w_1 = \text{richtig}$, $w_2 = \text{falsch}$)	
$P(w n, s) :$	$P(w_1 n_1, s_1) = 0.5$ $P(w_2 n_1, s_1) = 0.5$ $P(w_1 n_1, s_2) = 0.6$ $P(w_2 n_1, s_2) = 0.4$ $P(w_1 n_1, s_3) = 0.8$ $P(w_2 n_1, s_3) = 0.2$ $P(w_1 n_2, s_1) = 0.3$ $P(w_2 n_2, s_1) = 0.7$ $P(w_1 n_2, s_2) = 0.4$ $P(w_2 n_2, s_2) = 0.6$ $P(w_1 n_2, s_3) = 0.5$ $P(w_2 n_2, s_3) = 0.5$
GESTIK ($g_1 = \text{hektisch}$, $g_2 = \text{ruhig}$)	
$P(g w) :$	$P(g_1 w_1) = 0.3$ $P(g_2 w_1) = 0.7$ $P(g_1 w_2) = 0.9$ $P(g_2 w_2) = 0.1$
MIMIK ($m_1 = \text{entsetzt}$, $m_2 = \text{gelassen}$)	
$P(m w) :$	$P(m_1 w_1) = 0.2$ $P(m_2 w_1) = 0.8$ $P(m_1 w_2) = 0.6$ $P(m_2 w_2) = 0.4$

Tabelle 2.1: bedingte Wahrscheinlichkeiten bzw. a-priori Wahrscheinlichkeiten für die Knoten des Beispielnetzes in Abbildung 2.1

Werden dem Prüfling allerdings mehrere Prüfungsfragen gestellt, um das Wissensniveau des Prüflings differenzierter einzuschätzen, so muß das vorhandene Bayessche Netz um Knoten erweitert werden. Wird dies nicht getan, so stellt sich die Situation wie folgt dar: Um das Wissensniveau des Prüflings einschätzen zu können, werden ihm nacheinander mehrere Prüfungsfragen gestellt, deren Schwierigkeitsgrade bekannt sind. Die Antwort des Prüflings wird abgewartet, so daß sie als richtig oder falsch eingestuft werden kann. Die Gestik oder die Mimik werden in diesem Fall also nicht

benötigt. Im selben Bayesschen Netz werden nun sukzessive die Hypothesen der Knoten S (SCHWIERIGKEIT DER JEW. FRAGE) und W (WISSEN DER JEW. ANTWORT) instantiiert, die den eingetroffenen Ereignissen entsprechen.

Prüfungsfrage	Schwierigkeit	Antwort	Einschätzung des Wissensniveaus
-	-	-	$BEL(N) = (0.800, 0.200)$
1	leicht	falsch	$BEL(N) = (0.615, 0.385)$
2	leicht	falsch	$BEL(N) = (0.615, 0.385)$
3	leicht	falsch	$BEL(N) = (0.615, 0.385)$
4	schwer	richtig	$BEL(N) = (0.870, 0.130)$
5	mittel	richtig	$BEL(N) = (0.857, 0.143)$
6	schwer	richtig	$BEL(N) = (0.870, 0.130)$
7	schwer	falsch	$BEL(N) = (0.741, 0.259)$
8	leicht	richtig	$BEL(N) = (0.865, 0.135)$
9	leicht	richtig	$BEL(N) = (0.865, 0.135)$
10	leicht	richtig	$BEL(N) = (0.865, 0.135)$

Tabelle 2.2: Einschätzung des Wissensniveaus eines Prüflings im Verlauf einer Prüfungssituation mit mehreren Prüfungsfragen durch das statische Beispielnetz **Prüfungsfrage** statisch in Abbildung 2.1

Wie man der Tabelle 2.2 entnehmen kann, wird der Verlauf der Prüfungssituation bei der Einschätzung des Wissensniveaus des Prüflings nicht berücksichtigt. Selbst mehrere falsche Antworten auf leichte Fragen belassen die Einschätzung des Wissensniveaus des Prüflings unverändert. **Bsp**

Für solche Domänen ist ein statisches Modell nicht ausreichend oder sinnvoll. Es wird ein Modell benötigt, das eine Historie berücksichtigt. Das bestehende Bayessche Netz soll also bei Bedarf um notwendige Knoten erweitert werden können. Dieses wird im folgenden Beispiel illustriert.

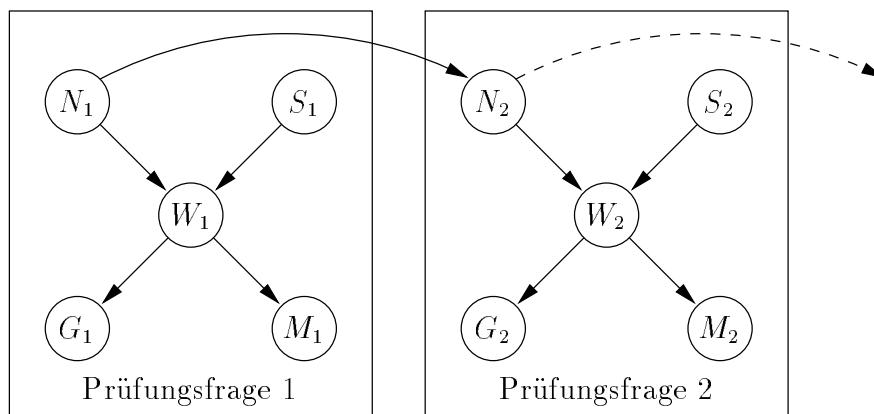


Abbildung 2.2: Beispielnetz **Prüfungsfrage** dynamisch (dynamisches Bayessches Netz)

Beispiel 2.1.2 (Prüfungssituation dynamisch)

Das Bayessche Netz aus dem vorhergehenden Beispiel wird nun so erweitert, daß frühere Prüfungsfragen mit berücksichtigt werden. Es wird für jede Prüfungsfrage

ein Bayessches Netz angelegt. Die Knoten, die die Zufallsvariable WISSENSNIVEAU DES SCHÜLERS repräsentieren, sind voneinander abhängig. Die Einschätzung über das Wissensniveau des Prüflings wird an die nächste Instanz weitergegeben.

WISSENSNIVEAU DES SCHÜLERS ($n_{0,1} = \text{hoch}$, $n_{0,2} = \text{niedrig}$) für Zeitscheibe 0			
$P(n_0) :$	$P(n_{0,1}) = 0.8$	$P(n_{0,2}) = 0.2$	
WISSENSNIVEAU DES SCHÜLERS ($n_{t,1} = \text{hoch}$, $n_{t,2} = \text{niedrig}$) für Zeitscheibe t ($t > 0$)			
$P(n_t n_{t-1}) :$	$P(n_{t,1} n_{t-1,1}) = 1$	$P(n_{t,2} n_{t-1,1}) = 0$	
	$P(n_{t,1} n_{t-1,2}) = 0$	$P(n_{t,2} n_{t-1,2}) = 1$	

Tabelle 2.3: bedingte Wahrscheinlichkeiten bzw. a-priori Wahrscheinlichkeiten für die Zufallsvariable WISSENSNIVEAU DES SCHÜLERS des Beispielnetzes Prüfungsfrage dynamisch in Abbildung 2.2

Die a-priori und bedingten Wahrscheinlichkeiten für die Zufallsvariable WISSENSNIVEAU DES SCHÜLERS sind aus Tabelle 2.3 zu entnehmen. Die bedingten Wahrscheinlichkeiten der anderen Zufallsvariablen bleiben unverändert und können aus Tabelle 2.1 entnommen werden.

Prüfungsfrage	Schwierigkeit	Antwort	Einschätzung des Wissensniveaus
-	-	-	$BEL(N_0) = (0.800, 0.200)$
1	leicht	falsch	$BEL(N_0) = (0.615, 0.385)$
2	leicht	falsch	$BEL(N_1) = (0.390, 0.610)$
3	leicht	falsch	$BEL(N_2) = (0.204, 0.796)$
4	schwer	richtig	$BEL(N_3) = (0.299, 0.701)$
5	mittel	richtig	$BEL(N_4) = (0.390, 0.610)$
6	schwer	richtig	$BEL(N_5) = (0.516, 0.484)$
7	schwer	falsch	$BEL(N_6) = (0.432, 0.568)$
8	leicht	richtig	$BEL(N_7) = (0.549, 0.451)$
9	leicht	richtig	$BEL(N_8) = (0.661, 0.339)$
10	leicht	richtig	$BEL(N_9) = (0.757, 0.243)$

Tabelle 2.4: Einschätzung des Wissensniveaus eines Prüflings im Verlauf einer Prüfungssituation mit mehreren Prüfungsfragen durch das dynamische Beispielnetz Prüfungsfrage dynamisch in Abbildung 2.2

Mit jeder Prüfungsfrage wächst das Bayessche Netz um eine sogenannte *Zeitscheibe*. Mit jeder Zeitscheibe wächst dabei nicht nur der Speicherbedarf für das Bayessche Netz sondern auch der Speicher- und Zeitbedarf für den Propagierungsalgorithmus. Sind die Ressourcen für Speicher oder Zeit beschränkt, so müssen nach einer bestimmten Anzahl von Zeitscheiben alte Zeitscheiben abgehängt werden. Man bezeichnet dies auch als *Rollup*. Im vorliegenden Beispiel gestaltet sich der Rollup relativ einfach. Der BEL-Wert des vorhergehenden Knotens der Zufallsvariablen WISSENSNIVEAU DES SCHÜLERS wird als a-priori Wahrscheinlichkeit für den aktuellen Knoten der Zufallsvariablen WISSENSNIVEAU DES SCHÜLERS eingetragen. Die alten Zeitscheiben können somit ohne Informationsverlust abgeschnitten werden. Knoten wie der Knoten der Zufallsvariablen WISSENSNIVEAU DES SCHÜLERS bezeichnet man auch als dynamische Knoten. Treten in einer Zeitscheibe mehrere solcher dynamischen Knoten auf, so

Prüfungsfrage	Schwierigkeit	Antwort	Einschätzung des Wissensniveaus	
			statisch	dynamisch
-	-	-	(0.800, 0.200)	(0.800, 0.200)
1	leicht	falsch	(0.615, 0.385)	(0.615, 0.385)
2	leicht	falsch	(0.615, 0.385)	(0.390, 0.610)
3	leicht	falsch	(0.615, 0.385)	(0.204, 0.796)
4	schwer	richtig	(0.870, 0.130)	(0.299, 0.701)
5	mittel	richtig	(0.857, 0.143)	(0.390, 0.610)
6	schwer	richtig	(0.870, 0.130)	(0.516, 0.484)
7	schwer	falsch	(0.741, 0.259)	(0.432, 0.568)
8	leicht	richtig	(0.865, 0.135)	(0.549, 0.451)
9	leicht	richtig	(0.865, 0.135)	(0.661, 0.339)
10	leicht	richtig	(0.865, 0.135)	(0.757, 0.243)

Tabelle 2.5: Vergleich der Einschätzungen des Wissensniveaus durch das Beispielnetz **Prüfungsfrage statisch** mit den Einschätzungen des Wissensniveaus durch das Beispielnetz **Prüfungsfrage dynamisch**

ist der Rollup nicht so einfach durchzuführen. Diese Problematik wird in Kapitel 3 beschrieben.

Der Vergleich von Tabelle 2.4 mit Tabelle 2.2 in Tabelle 2.5 zeigt, wie wichtig es ist, den Verlauf der Prüfungssituation bei der Einschätzung des Wissensniveaus des Prüflings mit zu berücksichtigen. Der BEL-Wert für die Zufallsvariable WISSENSNIVEAU DES SCHÜLERS im statischen Bayesschen Netz unterscheidet sich im Verlauf der Prüfungssituation ganz erheblich vom entsprechenden BEL-Wert für die Zufallsvariable WISSENSNIVEAU DES SCHÜLERS im Dynamischen Bayesschen Netz.

Bsp

Dynamische Bayessche Netze verändern sich also über die Zeit. Bei Bedarf werden neue Zeitscheiben angehängt und alte Zeitscheiben abgeschnitten. Im nachfolgenden Abschnitt werden dazu die notwendigen Begriffe eingeführt und die genaue Vorgehensweise beim Anhängen und Abschneiden von Zeitscheiben erläutert.

2.2 Struktur

Im folgenden werden einige allgemeine Sachzusammenhänge für Dynamische Bayessche Netze beschrieben und die entsprechenden Begriffe eingeführt. Es wird gezeigt, wie sich Speicherplatz einsparen läßt, indem die Schemata für die Zeitscheiben entsprechend verwaltet werden.

Für das Lösen von Dynamischen Bayesschen Netzen werden zwei verschiedene Strategien vorgestellt. Die eine Strategie kann als global angesehen werden, bei der das gesamte Dynamische Bayessche Netz beim Lösen als ein Bayessches Netz betrachtet wird, das sich nicht aus Zeitscheiben aufbaut. Bei der zweiten Strategie hingegen soll sich der Aufbau des Bayesschen Netzes aus Zeitscheiben auch in der Struktur des Junction Tree widerspiegeln. Jede der beiden Strategien hat seine Vor- und Nachteile, die jeweils in den einzelnen Abschnitten angesprochen werden.

Ein Dynamisches Bayessches Netz besteht aus einer variablen endlichen Anzahl n von Zeitscheiben. Sei $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ der DAG (*gerichtete azyklische Graph*), der die Struktur des dynamischen Modells zu einem bestimmten Zeitpunkt beschreibt. Wenn t' die erste Zeitscheibe des DAGs ist, dann besteht \mathbf{V} aus den disjunkten Teilmengen $\mathbf{V}(t'), \dots, \mathbf{V}(t' + n - 1)$, d.h.

$$\mathbf{V} = \mathbf{V}(t', n) = \bigcup_{t=t'}^{t'+n-1} \mathbf{V}(t).$$

Als Vorlage für die Instantiierung von Zeitscheiben dienen Bayessche Netze, die auch als *Schema* bezeichnet werden. Dabei übernehmen einige Knoten eines Schemas eine doppelte Rolle. Je nach Instantiierung sind die Zufallsvariablen dieser Knoten unabhängig von irgendwelchen Zufallsvariablen, die durch Knoten in vorhergehenden Zeitscheiben repräsentiert werden, oder von Zufallsvariablen abhängig, die mit Knoten in der vorhergehenden Zeitscheibe assoziiert sind (In Kapitel 4 werden Abhängigkeiten über mehrere Zeitscheiben hinweg und andere Besonderheiten vorgestellt.). Diese Abhängigkeiten werden durch Kanten von Zeitscheibe $t - 1$ zu Zeitscheibe t modelliert, sogenannten *temporale Kanten* oder auch *relationale Kanten*. Jede neue Zeitscheibe wird also durch temporale Kanten mit der letzten Zeitscheibe des alten Netzes verbunden.

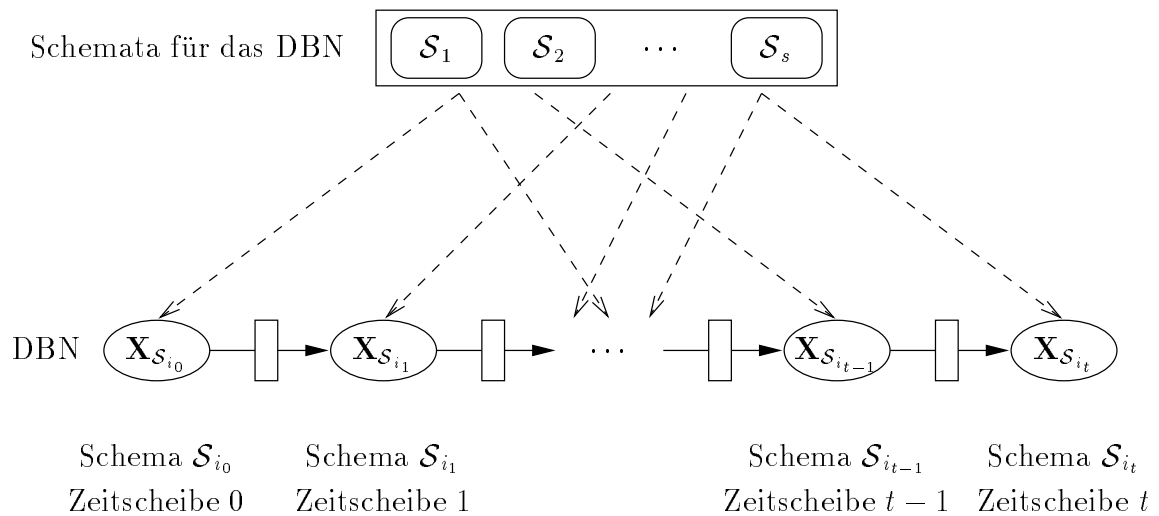


Abbildung 2.3: DBN: Die Zeitscheiben sind Instanzen der Schemata für das DBN.

Ein Dynamisches Bayessches Netz ist somit eine endliche Aneinanderreihung von Schemata, die als Zeitscheiben instantiiert wurden (siehe Abbildung 2.3). In Abbildung 2.3 sind die Schemata, die als Zeitscheiben für das DBN instantiiert werden können, mit $\mathcal{S}_1, \dots, \mathcal{S}_s$ bezeichnet. Mit $\mathbf{X}_{\mathcal{S}_{i_\ell}}$ sind die Mengen der Zufallsvariablen bezeichnet, aus denen sich das DBN zusammensetzt. Die Zufallsvariablen für das Schema \mathcal{S}_i ($1 \leq i \leq s$) werden durch $\mathbf{X}_{\mathcal{S}_i}$ symbolisiert. Als Zeitscheibe k ($0 \leq k \leq t$) ist also somit das Schema \mathcal{S}_{i_k} ($1 \leq i_k \leq s$) mit der Menge der Zufallsvariablen $\mathbf{X}_{\mathcal{S}_{i_k}}$ instantiiert. Die mit den Zufallsvariablen eines Schemas assoziierten bedingten Wahrscheinlichkeiten, die jeweils in einer Tabelle bedingter Wahrscheinlichkeiten organisiert sind, sind zu allen Zeitpunkten gleich, an denen das Schema instantiiert wird. Bei einer

entsprechenden Verwaltung muß für die verschiedenen zeitlichen Instantiierungen einer Zufallsvariablen insgesamt nur eine Tabelle bedingter Wahrscheinlichkeiten vorliegen, nämlich in einem Schema, in dem die Zufallsvariable vorkommt. D.h. die verschiedenen zeitlichen Instantiierungen von Zufallsvariablen in den Zeitscheiben des Dynamischen Bayesschen Netzes enthalten keine Wahrscheinlichkeitentabellen, sondern nur einen Verweis auf das Schema in dem die Zufallsvariable mit ihrer Wahrscheinlichkeitentabelle modelliert ist. Dadurch läßt sich der Speicherplatzverbrauch des Dynamischen Bayesschen Netzes in der Repräsentation als gerichteter azyklischer Graph enorm verringern. Sind von einem Schema zum Beispiel zehn Zeitscheiben instantiiert worden, so sind die Wahrscheinlichkeitentabellen nicht insgesamt elfmal (im Schema und jeweils in jeder Zeitscheibe) sondern nur einmal (nämlich im Schema der Zeitscheibe) vorhanden. Diese Speicherplatzersparnis läßt sich zusammen mit der Shafer-Shenoy Architektur, die in 1.3.3.2 vorgestellt wird, auch in der Repräsentation des Dynamischen Bayesschen Netzes als Junction Tree erzielen. Anstatt nämlich die Wahrscheinlichkeitentabellen in den Cliques des Junction Trees zu speichern, kann man in den Cliques einen Verweis auf das Schema eintragen, in dem die entsprechende Zufallsvariable mit ihrer Wahrscheinlichkeitentabelle gespeichert ist. Dies geht aber nur bei der Shafer-Shenoy Architektur, wie sie in 1.3.3.2 vorgestellt wird, da die Wahrscheinlichkeitentabellen hier noch als Faktoren eines Produkts vorliegen. Bei der Aalborg Architektur läßt sich die Einsparung nicht erzielen, da das Produkt der Wahrscheinlichkeitentabellen ausgerechnet wird.

In der Abbildung 2.3 sind die sogenannten *Zustandsentwicklungsmodelle* dargestellt durch rechteckige Kästen zwischen zwei jeweils benachbarten Zeitscheiben. Ein Zustandsentwicklungsmodell beschreibt die Abhängigkeiten der Zustandsvariablen einer Zeitscheibe k von den Zustandsvariablen der Vorgängerzeitscheibe $k - 1$ ($1 \leq k \leq t$). Analog zur Wahrscheinlichkeitsverteilung aller Zufallsvariablen eines Bayesschen Netzes läßt sich die Wahrscheinlichkeitsverteilung für Dynamische Bayessche Netze als Produkt ihrer bedingten Wahrscheinlichkeiten schreiben:

$$P(\mathbf{X}_{S_{i_0}}, \dots, \mathbf{X}_{S_{i_t}}) = \prod_{V \in \bigcup_{k=0}^t \mathbf{X}_{S_{i_k}}} P(V \mid \text{pa}(V)).$$

Ist V ein Wurzelknoten, so reduzieren sich die bedingten Wahrscheinlichkeiten $P(V \mid \text{pa}(V))$ zu $P(V)$.

In Abbildung 2.4 ist ein Dynamisches Bayessches Netz dargestellt, das sich aus den vier verschiedenen Schemata \mathcal{A} , \mathcal{B} , \mathcal{C} und \mathcal{D} aufbaut. Es sind dabei fünf Zeitscheiben (Zeitscheibe 0 bis Zeitscheibe 4) instantiiert. Wie in der Abbildung zu sehen ist, sind im Schema \mathcal{B} Knoten enthalten, die Elternknoten im Schema \mathcal{A} und Elternknoten im Schema \mathcal{D} haben. Diese Elternknoten der beiden Schemata müssen dieselben Namen besitzen oder über denselben *Alias* referenziert werden können. Ein Knoten in einem Dynamischen Bayesschen Netz kann also neben seinem *eindeutigen* Namen, den kein weiterer Knoten des Dynamischen Bayesschen Netzes besitzt, noch mit mehreren anderen Namen (*Aliase*) benannt sein, die er sich mit anderen Knoten des Dynamischen Bayesschen Netzes teilt. Zwei Knoten eines Schemas haben aber nie denselben Alias. Ein Alias kommt also in einem Schema höchstens einmal vor.

Ein Schema hat keine Elternknoten in vorhergehenden Zeitscheiben, wenn es als Zeitscheibe 0 instantiiert wird. Würde das Schema \mathcal{A} zum Zeitpunkt $t \neq 0$ als Zeit-

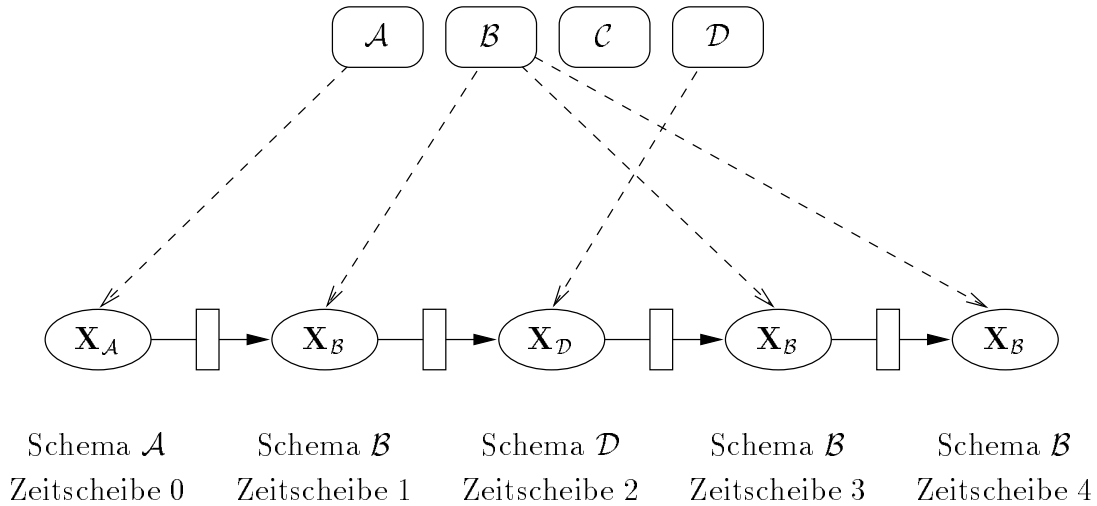


Abbildung 2.4: DBN: Fünf Zeitscheiben eines DBNs mit vier Schemata.

scheibe t instantiiert werden, so hätten einige Knoten der Zeitscheibe t Elternknoten in Zeitscheibe $t - 1$. Diese Knoten bezeichnet man als *Dynamische Knoten*. Die gerichteten Kanten zwischen den Zeitscheiben werden *temporale Kanten* genannt. In Abbildung 2.5 sind die *temporalen Kanten* gestrichelt und die *Dynamischen Knoten* grau-schraffiert eingezeichnet.

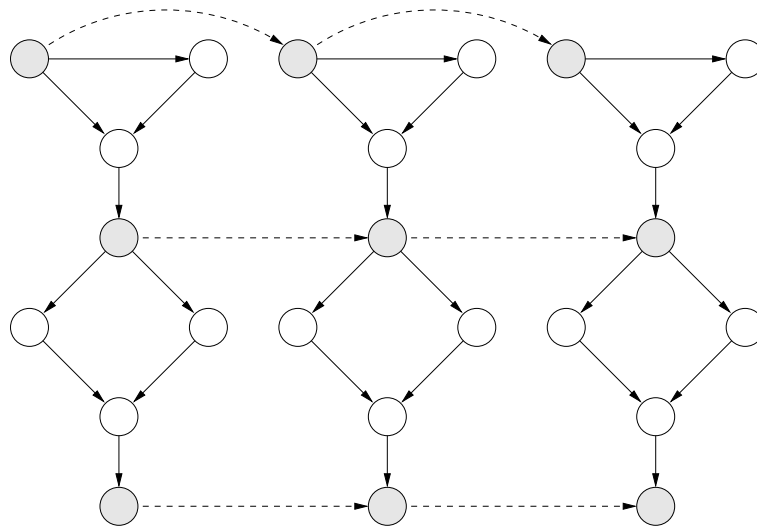


Abbildung 2.5: DBN: Temporale Kanten und Dynamische Knoten eines DBNs.

Mit den temporalen Kanten sind bedingte Wahrscheinlichkeiten verbunden, die beschreiben, wie sich die Wahrscheinlichkeitsverteilung der Zufallsvariablen in der Zeitscheibe $t - 1$ auf die Wahrscheinlichkeitsverteilung der Zufallsvariablen in der Zeitscheibe t auswirkt.

Definition 2.2.1 (zeitliche Kanten)

Die Menge der gerichteten Kanten

$$\mathbf{E}^{tmp}(t) = \{(A, B) \in \mathbf{E} \mid A \in \mathbf{V}(t - 1), B \in \mathbf{V}(t)\}, \quad t' \leq t \leq t' + n - 1,$$

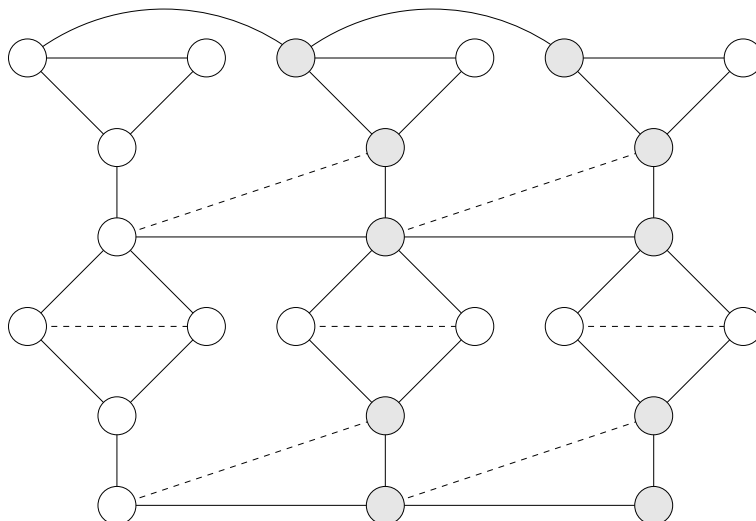


Abbildung 2.6: DBN: Moralisierungskanten und Interfaceknoten eines DBNs.

wird als Menge der *zeitlichen Kanten* (oder *zeitlichen Relationen*) der Zeitscheibe t bezeichnet. \square

Durch die obige Definition für die *zeitlichen Kanten* wird implizit vorausgesetzt, daß die Zeitscheiben des Dynamischen Bayesschen Netzes die *Markov Eigenschaft* einhalten, was informal damit beschrieben werden kann, daß die Zukunft unabhängig von der Vergangenheit ist, wenn die Gegenwart bekannt ist. Formal wird dies u.a. wie folgt geschrieben:

$$\mathbf{X}_{\mathbf{V}(0)}, \dots, \mathbf{X}_{\mathbf{V}(t-1)} \perp\!\!\!\perp \mathbf{X}_{\mathbf{V}(t+1)}, \dots, \mathbf{X}_{\mathbf{V}(t+k)} \mid \mathbf{X}_{\mathbf{V}(t)}, \quad \forall t, k > 0.$$

Dies bedeutet, daß eine Zufallsvariable X_K , die einem Knoten $K \in \mathbf{V}(\ell)$ zugeordnet ist, von keiner Zufallsvariablen X_A direkt beeinflusst wird, die mit einem beliebigen Knoten $A \in \mathbf{V}(i)$ assoziiert ist ($i \leq \ell - 2$). Graphentheoretisch bedeutet dies, daß keine Kanten erlaubt sind, die zwei nichtbenachbarte Zeitscheiben miteinander verbinden.

Nachfolgend werden zwei verschiedene Möglichkeiten für die Bearbeitung von Dynamischen Bayesschen Netzen an einem Beispielnetz miteinander verglichen. Bei der ersten Möglichkeit wird das Dynamische Bayessche Netz nur als Bayessches Netz betrachtet. Der Aufbau des Dynamischen Bayesschen Netzes aus Zeitscheiben spiegelt sich dabei nicht im Junction Tree wider. Bei der zweiten Möglichkeit werden die Zeitscheiben des Dynamischen Bayesschen Netzes bei der Evaluierung berücksichtigt. Hier deutet sich schon an, wie ein Rollup ohne Informationsverlust durchgeführt werden kann.

Behandlung des DBNs als normales BN

Im folgenden wird für das Lösen ein Dynamisches Bayessches Netz als ein normales Bayessches Netz zu einem bestimmten Zeitpunkt aufgefaßt. Der Aufbau des Dynamischen Bayesschen Netzes aus Zeitscheiben wird

nicht beachtet. Werden nun neue Zeitscheiben angehängt, so wird man sehen, daß das neu entstehende Dynamische Bayessche Netz insgesamt neu berechnet werden muß, und man keine alten Berechnungen wieder verwenden kann.

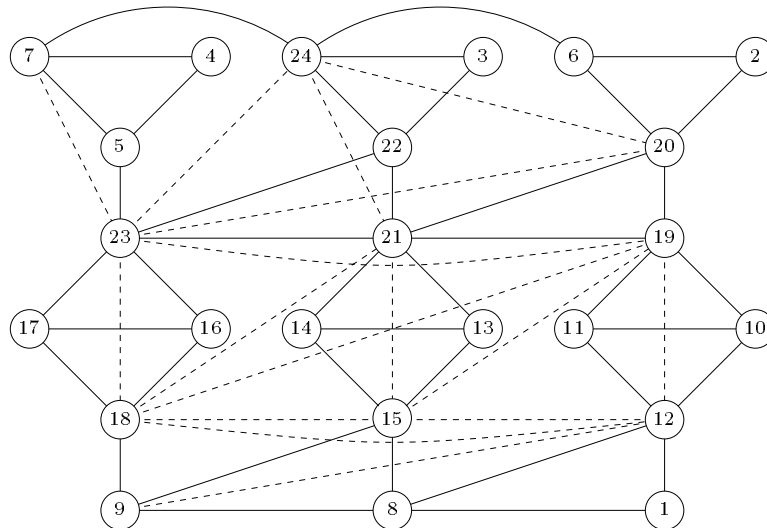


Abbildung 2.7: DBN: Triangulationskanten eines DBNs ohne Berücksichtigung von Zeitscheiben. Die Eliminationsreihenfolge ist der Knotennummerierung zu entnehmen.

Zu jedem Zeitpunkt t ist das Dynamische Bayessche Netz genau bekannt. Zu einem Zeitpunkt t kann das Dynamische Bayessche Netz damit auch als statisches Bayessches Netz aufgefaßt werden, und zur Lösung des Dynamischen Bayesschen Netzes können die in 1.3 vorgestellten Propagierungsalgorithmen ohne Anpassung zur Anwendung kommen. Dabei wird für das Dynamische Bayessche Netz die sekundäre Repräsentation² berechnet, ohne dabei zu berücksichtigen, daß das Netz aus Zeitscheiben aufgebaut ist. Zu allererst wird das Dynamische Bayessche Netz moralisiert. In Abbildung 2.6 sind die *Moralisierungskanten* gestrichelt eingezeichnet. Die grau-schraffierten Knoten bezeichnet man als *Interfaceknoten*. Der Grund für diese Bezeichnung wird klar, wenn bei der Triangulierung des Dynamischen Bayesschen Netzes die Zeitscheiben berücksichtigt werden. Dann nämlich bilden die Interfaceknoten das *Zustandsentwicklungsmodell* der Zeitscheibe.

Zunächst wird nun aber die globale Betrachtung vorgestellt, in der die Zeitscheiben des Dynamischen Bayesschen Netzes unberücksichtigt bleiben sollen. Die optimale Eliminationsreihenfolge wird über alle Knoten des gesamten Dynamischen Bayesschen Netzes bestimmt. In Abbildung 2.7 sind dazu die *Triangulationskanten* einer Eliminationsreihenfolge von Knoten ohne Berücksichtigung der Zeitscheiben gestrichelt eingezeichnet.

Man erkennt, wie durch Triangulationskanten Knoten über mehrere Zeitscheiben hinweg miteinander verbunden werden. Die hieraus resultierenden Cliques vereinen Knoten aus mehreren Zeitscheiben. Wird nun zusätzlich eine neue Zeitscheibe angefügt, und nun eine optimale Eliminationsreihenfolge über alle Knoten gesucht, so

²Die sekundäre Repräsentation wird auch als *Junction Tree* bezeichnet.

muß der *Junction Tree* im allgemeinen vollständig neu berechnet werden. Dabei können alte Berechnungen nicht wiederverwendet werden.

In Abbildung 2.8 ist der aus der Abbildung 2.7 resultierende Junction Tree dargestellt.

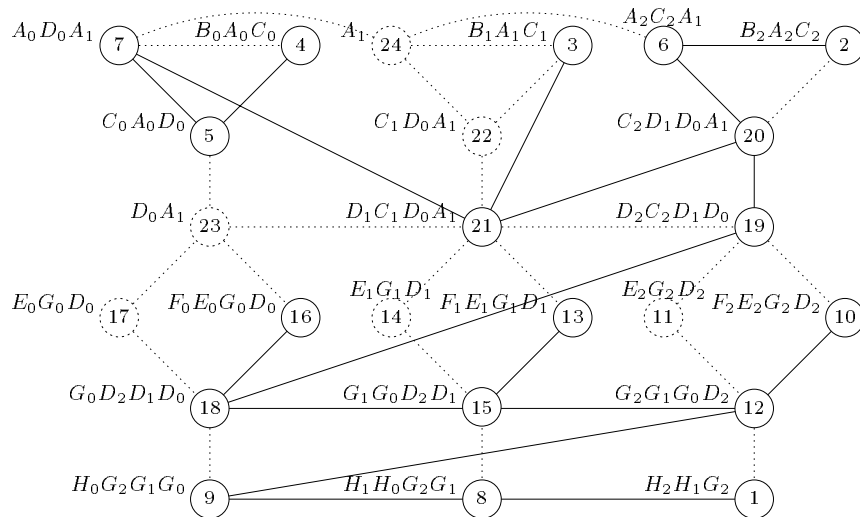


Abbildung 2.8: DBN: Junction Tree eines DBNs ohne Berücksichtigung von Zeitscheiben.

Insgesamt müssen

- die Eliminationsreihenfolge,
- der Junction Tree des Bayesschen Netzes, der auf der Eliminationsreihenfolge beruht, und
- die einzelnen Clques-Nachrichten, die bei vorhergehenden Inferenzen verschickt wurden,

neu berechnet werden.

In Tabelle 2.6 sind die Eliminationsreihenfolge der Knoten und die dabei induzierten Cluster und Clques dargestellt. Die Cluster und Clques umfassen Knoten aus mehreren Zeitscheiben. Eine maximale Clque enthält vier Knoten.

Strukturerhaltung im Junction Tree des DBNs

Jetzt wird das Dynamische Bayessche Netz beim Lösen so behandelt, daß sich der Aufbau des Dynamischen Bayesschen Netzes aus Zeitscheiben auch in der Struktur des Junction Trees widerspiegelt. In diesem Zusammenhang wird der Begriff der *eingeschränkten Eliminationsreihenfolge* eingeführt. Werden neue Zeitscheiben angehängt, so wird man sehen, daß beim Lösen des neu entstandenen Dynamischen Bayesschen Netzes alte Berechnungen wiederverwendet werden können. Leider ist im allgemeinen die eingeschränkte Eliminationsreihenfolge im Vergleich zu einer Eliminationsreihenfolge, die das gesamte Netz berücksichtigt, schlechter im

<i>eliminiertes Knoten</i>	<i>induzierter Cluster</i>	<i>induzierte Clique</i>
H_2	$H_2H_1G_2$	$Clq_1 = \{H_2, H_1, G_2\}$
B_2	$B_2A_2C_2$	$Clq_2 = \{B_2, A_2, C_2\}$
B_1	$B_1A_1C_1$	$Clq_3 = \{B_1, A_1, C_1\}$
B_0	$B_0A_0C_0$	$Clq_4 = \{B_0, A_0, C_0\}$
C_0	$C_0A_0D_0$	$Clq_5 = \{C_0, A_0, D_0\}$
A_2	$A_2C_2A_1$	$Clq_6 = \{A_2, C_2, A_1\}$
A_0	$A_0D_0A_1$	$Clq_7 = \{A_0, D_0, A_1\}$
H_1	$H_1H_0G_2G_1$	$Clq_8 = \{H_1, H_0, G_2, G_1\}$
H_0	$H_0G_2G_1G_0$	$Clq_9 = \{H_0, G_2, G_1, G_0\}$
F_2	$F_2E_2G_2D_2$	$Clq_{10} = \{F_2, E_2, G_2, D_2\}$
E_2	$E_2G_2D_2$	-
G_2	$G_2G_1G_0D_2$	$Clq_{11} = \{G_2, G_1, G_0, D_2\}$
F_1	$F_1E_1G_1D_1$	$Clq_{12} = \{F_1, E_1, G_1, D_1\}$
E_1	$E_1G_1D_1$	-
G_1	$G_1G_0D_2D_1$	$Clq_{13} = \{G_1, G_0, D_2, D_1\}$
F_0	$F_0E_0G_0D_0$	$Clq_{14} = \{F_0, E_0, G_0, D_0\}$
E_0	$E_0G_0D_0$	-
G_0	$G_0D_2D_1D_0$	$Clq_{15} = \{G_0, D_2, D_1, D_0\}$
D_2	$D_2C_2D_1D_0$	$Clq_{16} = \{D_2, C_2, D_1, D_0\}$
C_2	$C_2D_1D_0A_1$	$Clq_{17} = \{C_2, D_1, D_0, A_1\}$
D_1	$D_1C_1D_0A_1$	$Clq_{16} = \{D_1, C_1, D_0, A_1\}$
C_1	$C_1D_0A_1$	-
D_0	D_0A_1	-
A_1	A_1	-

Tabelle 2.6: Eliminierung der Knoten: Zeitscheiben werden nicht berücksichtigt

Triangulationsergebnis. Vor allem die Cliques, die an der Nahtstelle zwischen den Zeitscheiben entstehen, werden besonders groß. Dies ist jedoch vom entsprechenden Zustandsentwicklungsmodell zwischen den Zeitscheiben abhängig.

Sollen also alte Berechnungen wiederverwendet werden, so muß das Junction Tree Verfahren angepaßt werden. Ideal wäre es, wenn man beim Junction Tree neue Strukturen auch so anhängen könnte, wie dies bei den Dynamischen Bayesschen Netzen mit den Zeitscheiben möglich ist. Wie man aber gesehen hat, ist dies bei einer beliebigen Eliminationsreihenfolge im allgemeinen nicht machbar. Da man weiß, daß die Cliques eines Junction Trees des Bayesschen Netzes alleine schon durch die Eliminationsreihenfolge festgelegt sind, wird man die Eliminationsreihenfolge entsprechend anpassen müssen. Durch die Eliminationsreihenfolge werden auch die Triangulationskanten bestimmt. Will man Cliques, die nur Knoten aus der jeweiligen Zeitscheibe und maximal aus einer benachbarten Zeitscheibe enthalten, so dürfen die Triangulationskanten nur innerhalb einer Zeitscheibe liegen oder maximal zu einer benachbarten Zeitscheibe reichen. Den Zusammenhang zwischen der Eliminationsreihenfolge und den Triangulationskanten (und somit den Cliques, die entstehen) liefert uns der folgende

Satz von Rose, Tarjan und Lueker (entnommen aus [Kjærulff 93]) der allgemeinen Graphentheorie:

Satz 2.2.2 (Rose, Tarjan und Lueker)

Sei $\mathcal{G}_\# = (\mathbf{V}, \mathbf{E})$ ein Graph, dessen Knoten $V \in \mathbf{V}$ durch die bijektive Funktion $\#: \mathbf{V} \rightarrow \{1, \dots, |\mathbf{V}|\}, V \mapsto \#(V)$ mit einer Nummer versehen werden. Ein solcher Graph wird auch als *geordneter Graph* bezeichnet.

Sei also $\mathcal{G}_\# = (\mathbf{V}, \mathbf{E})$ ein geordneter Graph, dann wird die Kante $\{U, V\}$ im triangulierten Graphen $\mathcal{G}_\#^t$ genau dann und nur dann enthalten sein, wenn es einen Pfad $\langle V, V_1, \dots, V_k, U \rangle$ im geordneten Graphen $\mathcal{G}_\# = (\mathbf{V}, \mathbf{E})$ gibt, so daß $\#(V_i) < \min\{\#(V), \#(U)\} \forall i = 1, \dots, k$. □

Wie schon im Satz erwähnt, ist $\#$ eine bijektive Funktion von der Menge der Knoten \mathbf{V} in die Menge $\{1, \dots, |\mathbf{V}|\}$, d.h.

$$\#: \mathbf{V} \rightarrow \mathbb{N} \text{ mit } V \mapsto \#(V) \in \{1, \dots, |\mathbf{V}|\}.$$

Für einen Knoten $V \in \mathbf{V}$ gibt $\#(V)$ die Position von V in der Eliminationsreihenfolge an. Mit dem Satz 2.2.2 kann nun zusammen mit einer Eliminationsreihenfolge, die sich auf die Zeitscheiben einschränkt (Die Knoten einer Zeitscheibe t werden vor den Knoten einer Zeitscheibe $t + 1$ abgearbeitet und so fort.), gezeigt werden, daß bei der Eliminierung eines Knotens V der Zeitscheibe t maximal Triangulationskanten eingefügt werden, die Knoten der Zeitscheibe t oder die Interfaceknoten der Zeitscheibe $t + 1$ miteinander verbinden. Die entsprechenden Aussagen und dazugehörigen Beweise können in [Kjærulff 93] nachgelesen werden.

Dabei ist eine wichtige Schlußfolgerung aus dem Satz, daß bei der Eliminierung eines Knotens nur zwischen nicht eliminierten Knoten Triangulationskanten eingefügt werden.

Dieses wird wie folgt ausgenutzt: Beim Anhängen einer neuen Zeitscheibe an das Dynamische Bayessche Netz behält man die Eliminationsreihenfolge bei, in der die Knoten der alten Zeitscheiben abgearbeitet wurden, und fügt daran die Eliminationsreihenfolge für die Abarbeitung der Knoten der neuen Zeitscheibe an. So werden zwischen den alten Knoten keine neuen Triangulationskanten eingefügt, und die alten Cliques bleiben zum großen Teil erhalten. Der *Junction Tree* muß nicht komplett neu berechnet werden. An den alten Junction Tree werden die neuen Cliques angehängt und eventuell alte Cliques an der Schnittstelle zwischen alten und neuen Cliques um ein paar Knoten erweitert. Es können alte Berechnungen wiederverwendet werden. In Abbildung 2.9 sind die *Triangulationskanten* einer Eliminationsreihenfolge gestrichelt eingezeichnet, die berücksichtigt, daß das Netz aus Zeitscheiben aufgebaut ist. Man erkennt, wie die Triangulationskanten Knoten einer Zeitscheibe oder zweier benachbarten Zeitscheiben miteinander verbinden. Die hieraus resultierenden Cliques vereinen somit nur Knoten einer Zeitscheibe oder zweier benachbarter Zeitscheiben. Wird nun eine neue Zeitscheibe angefügt, und die alte Eliminationsreihenfolge für die alten Knoten beibehalten, so muß der *Junction Tree* nicht komplett neu berechnet werden. Hier können alten Berechnungen wiederverwendet werden.

In Abbildung 2.10 ist der aus Abbildung 2.9 resultierende Junction Tree dargestellt. Damit die Übersicht nicht noch mehr leidet, wurden die Separatorknoten nicht eingezeichnet, da sich diese für jede Kante des Junction Trees leicht bestimmen lassen.

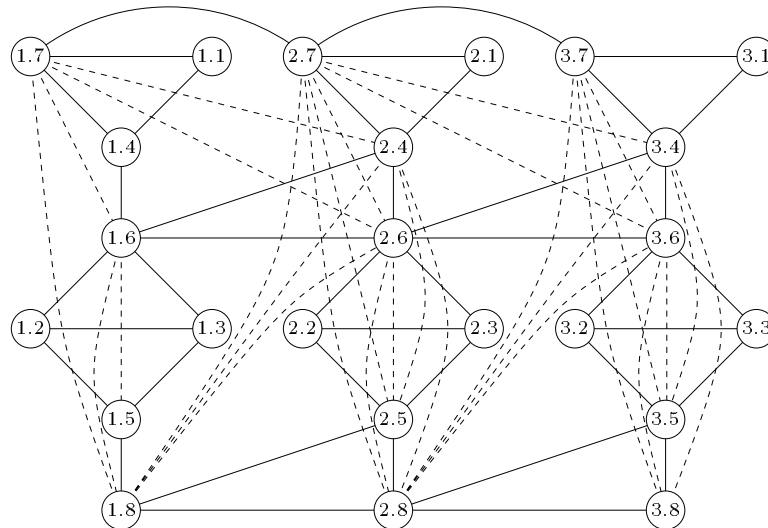


Abbildung 2.9: DBN: Triangulationskanten eines DBNs mit Berücksichtigung von Zeitscheiben. Die Eliminationsreihenfolge ist der Knotennumerierung zu entnehmen.

Die durchgezogenen Linien und Kreise geben die Kanten bzw. Cliques des Junction Trees an. Die gestrichelten Linien und Kreise stehen für die Kanten bzw. Cliques des Junction Trees, die beim Anhängen von Zeitscheiben entstehen. In Klammern ist angegeben, um welche Knoten sich die Cluster erweitern. Wie schon in 1.3.1 erwähnt wurde, wird nicht jeder Cluster zu einer Clique des Junction Trees. In der Abbildung sind die Cluster gepunktet dargestellt, die keine Cliques des Junction Trees sind.

In Tabelle 2.7 ist die auf die Zeitscheiben eingeschränkte Eliminierung noch einmal übersichtlich dargestellt. Die Knoten der verschiedenen Zeitscheiben sind durch einen waagrechten Strich in der Tabelle voneinander getrennt. Man erkennt, daß die Knoten in den verschiedenen Zeitscheiben in der gleichen Reihenfolge abgearbeitet werden. Dabei werden die dynamischen Knoten in logischer Folgerung aus Satz 2.2.2 als letzte Knoten der Zeitscheibe abgearbeitet. Im Vergleich von Tabelle 2.6 zu Tabelle 2.7 erkennt man leider auch, daß die Einschränkung der Eliminationsreihenfolge auf die Zeitscheiben in diesem Beispiel sehr viel größere Cliques erzeugt. In Kapitel 3 werden wir ein Rollup-Verfahren kennen lernen, daß sich nur bei Junction Trees einsetzen läßt, die mittels einer eingeschränkten Eliminationsreihenfolge bestimmt wurden. Dabei müssen vom Dynamischen Bayesschen Netz maximal immer nur zwei Zeitscheiben repräsentiert sein. Andere Junction Trees, die mittels einer Eliminationsreihenfolge bestimmt wurden, die die Zeitscheiben nicht berücksichtigen, lassen sich mit diesem Rollup-Verfahren nicht behandeln. Die Anzahl der Zeitscheiben des Dynamischen Bayesschen Netzes wächst mit der Zeit, und damit auch die Komplexität (Verbrauch von Speicherplatz und Rechenzeit) des Dynamischen Bayesschen Netzes. Irgendwann übersteigt die Komplexität des Dynamischen Bayesschen Netzes, das mit einer Eliminationsreihenfolge behandelt wird, die die Zeitscheiben nicht berücksichtigt, die Komplexität des Dynamischen Bayesschen Netzes, von dem immer maximal nur zwei Zeitscheiben repräsentiert werden. Dann relativiert sich das Verfahren, das die Zeitscheiben berücksichtigt, gegenüber dem Verfahren, welches eine globale Strategie verfolgt.

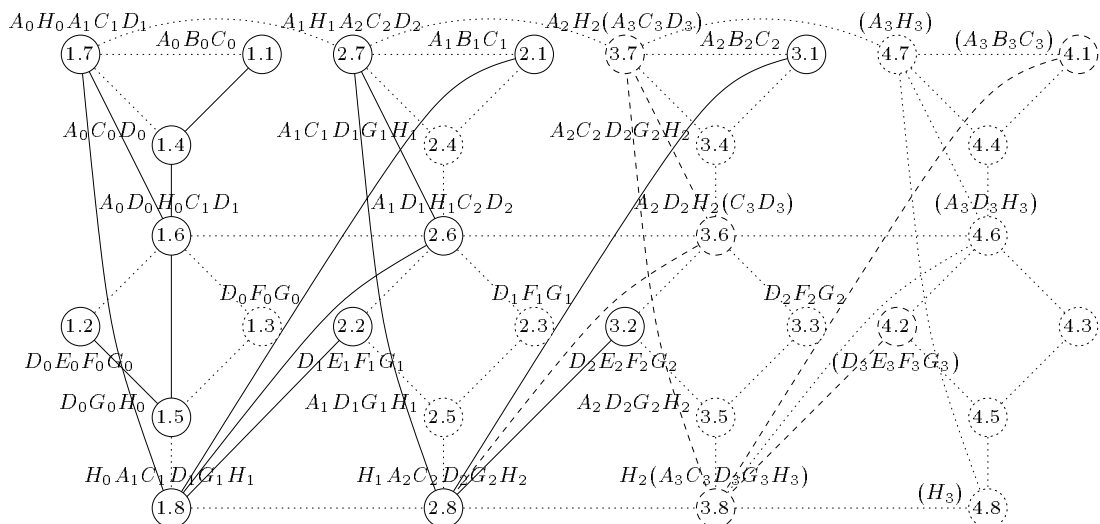


Abbildung 2.10: DBN: der Junction Tree eines DBNs mit Berücksichtigung von Zeitscheiben. Die Separatorknoten wurden aus Gründen der Übersichtlichkeit nicht eingezeichnet.

Man erkennt, wie sich der Junction Tree durch das Anhängen von neuen Zeitscheiben entwickelt, wenn die Zeitscheiben bei der Berechnung der Eliminationsreihenfolge berücksichtigt werden.

Um eine neue Zeitscheibe anzuhängen, wird also wie folgt vorgegangen:

1. Moralisiere den zusammengesetzten Graphen aus *Junction Tree* und DAG.
2. Trianguliere (eingeschränkt auf die Zeitscheiben) den neu entstandenen Graphen und bestimme die neuen Cliques.
3. Konstruiere den neuen (expandierten) *Junction Tree*.
4. Berechne die neuen Potentiale der Cliques und berücksichtige die alten Potentiale dabei.

Vom alten *Junction Tree* $\mathcal{T} = (\mathbf{C}, \mathbf{S})$ soll so viel wie möglich wiederverwendet werden, um den Konstruktionsaufwand für den neuen *Junction Tree* $\mathcal{T}' = (\mathbf{C}', \mathbf{S}')$ zu minimieren. Als direkte Konsequenz der eingeschränkten Eliminationsreihenfolge existiert für jede alte Clique $Clq \in \mathbf{C}$ eine neue Clique $Clq' \in \mathbf{C}'$, so daß gilt: $Clq \subseteq Clq'$. Für einige Cliques ist diese Relation strikt. Bei einer beliebigen Eliminationsreihenfolge ist eine Zuordnung alte zu neue Clique im allgemeinen nicht möglich. Der neue *Junction Tree* \mathcal{T} wird nun wie folgt erzeugt:

1. Bestimme die Menge der Cliques \mathbf{C}' von \mathcal{T}' .
2. Erzeuge \mathcal{T}' :
 - (a) Erzeuge die ψ -Tabellen der Cliques $Clq \in \mathbf{C}' \setminus \mathbf{C}$ und der Separatoren $Sep \in \mathbf{S}' \setminus \mathbf{S}$.
 - (b) Initialisiere alle neuen Tabellen mit 1. (Alle anderen Tabellen bleiben unverändert.)

<i>eliminiertes Knoten</i>	<i>induzierter Cluster</i>	<i>induzierte Clique</i>
B_0	$A_0B_0C_0$	$Clq_1 = \{A_0, B_0, C_0\}$
E_0	$D_0E_0F_0G_0$	$Clq_2 = \{D_0, E_0, F_0, G_0\}$
F_0	$D_0F_0G_0$	-
C_0	$A_0C_0D_0$	$Clq_3 = \{A_0, C_0, D_0\}$
G_0	$D_0G_0H_0$	$Clq_4 = \{D_0, G_0, H_0\}$
D_0	$A_0D_0H_0C_1D_1$	$Clq_5 = \{A_0, D_0, H_0, C_1, D_1\}$
A_0	$A_0H_0A_1C_1D_1$	$Clq_6 = \{A_0, H_0, A_1, C_1, D_1\}$
H_0	$H_0A_1C_1D_1G_1H_1$	$Clq_7 = \{H_0, A_1, C_1, D_1, G_1, H_1\}$
B_1	$A_1B_1C_1$	$Clq_8 = \{A_1, B_1, C_1\}$
E_1	$D_1E_1F_1G_1$	$Clq_9 = \{D_1, E_1, F_1, G_1\}$
F_1	$D_1F_1G_1$	-
C_1	$A_1C_1D_1G_1H_1$	-
G_1	$A_1D_1G_1H_1$	-
D_1	$A_1D_1H_1C_2D_2$	$Clq_{10} = \{A_1, D_1, H_1, C_2, D_2\}$
A_1	$A_1H_1A_2C_2D_2$	$Clq_{11} = \{A_1, H_1, A_2, C_2, D_2\}$
H_1	$H_1A_2C_2D_2G_2H_2$	$Clq_{12} = \{H_1, A_2, C_2, D_2, G_2, H_2\}$
B_2	$A_2B_2C_2$	$Clq_{13} = \{A_2, B_2, C_2\}$
E_2	$D_2E_2F_2G_2$	$Clq_{14} = \{D_2, E_2, F_2, G_2\}$
F_2	$D_2F_2G_2$	-
C_2	$A_2C_2D_2G_2H_2$	-
G_2	$A_2D_2G_2H_2$	-
D_2	$A_2D_2H_2(C_3D_3)$	$(Clq_{15} = \{A_2, D_2, H_2, C_3, D_3\})$
A_2	$A_2H_2(A_3C_3D_3)$	$(Clq_{16} = \{A_2, H_2, A_3, C_3, D_3\})$
H_2	$H_2(A_3C_3D_3G_3H_3)$	$(Clq_{17} = \{H_2, A_3, C_3, D_3, G_3, H_3\})$

Tabelle 2.7: Eliminierung der Knoten: die Zeitscheiben werden berücksichtigt

3. Für jede Clique $Clq \in \mathbf{C} \setminus \mathbf{C}'$ und jeden Separator $Sep \in \mathbf{S} \setminus \mathbf{S}'$ multipliziere die entsprechenden Belief-Potentiale zu den zugehörigen Belief-Potentialen der Cliques und Separatoren.
4. Ordne die Knoten der neuen Zeitscheibe einer bestimmten Clique zu. Es muß dabei gelten, daß die Eltern des Knotens auch in der Clique vorhanden sind.

In den beiden Abbildungen 2.11 und 2.12 ist das Anhängen von Zeitscheiben in der Repräsentation des Junction Trees dargestellt. Dabei ist in Abbildung 2.11 zu sehen, wie vorhandene Cliques um Knoten erweitert werden. In Abbildung 2.12 wird der Junction Tree nur um neue Cliques ergänzt. Alte Cliques werden nicht um Knoten erweitert.

Die eingeschränkte Eliminationsreihenfolge ermöglicht somit auch die Verwaltung von *vorberechneten Zeitscheiben*. Diese vorberechneten Zeitscheiben sind Junction Trees, deren Cliques und Kanten berechnet sind und für die schon Collect Evidence in Richtung der Nahtstelle an die Vorgängerzeitscheibe durchgeführt wurde. Kann ein Schema \mathcal{A} an Zeitscheiben angehängt werden, die von verschiedenen Schemata erzeugt werden, so wird je nach Vorgängerzeitscheibe eine andere vorberechnete Zeitscheibe für die Instantiierung des Schemas \mathcal{A} benötigt. Damit der gesamte Junction Tree kalibriert

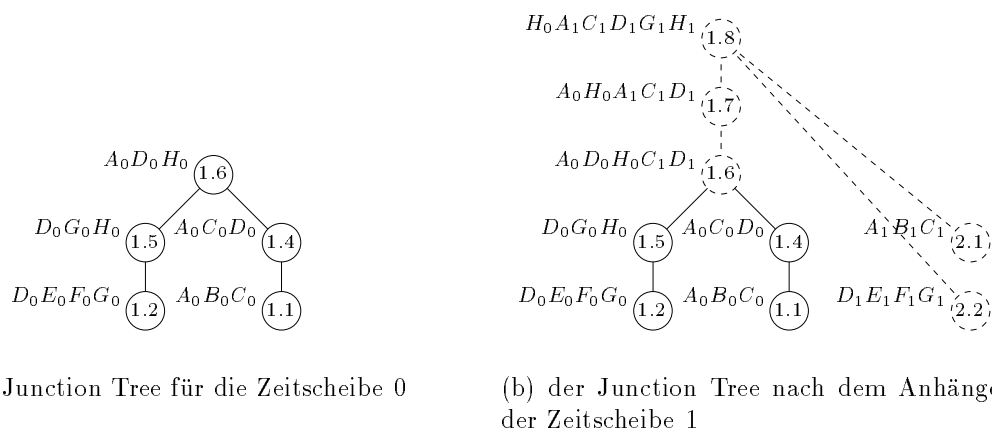


Abbildung 2.11: In der Repräsentation des Junction Trees wird an die Zeitscheibe 0 die Zeitscheibe 1 angehängt. Veränderte und neue Cliques sind gestrichelt eingezeichnet.

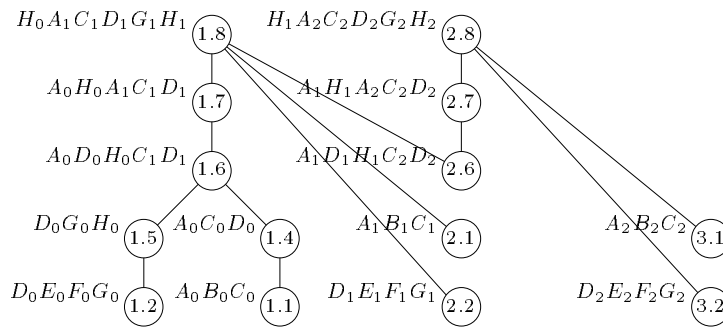
ist, muß, wenn keine Evidenzen im neuen Junction Tree eingetragen wurden und der alten Junction Tree vor dem Anhängen des neuen Junction Tree kalibriert war, nur noch ein Distribute Evidence an der Nahtstelle zwischen altem und neu angehängten Junction Tree in Richtung des neu angehängten Junction Trees aufgerufen werden. Wurden jedoch im neu angehängten Junction Tree Evidenzen eingetragen, so muß in den Ästen des neu angehängten Junction Trees, in denen wenigstens eine Evidenz eingetragen wurde, auch ein Collect Evidence bis zur Nahtstelle durchgeführt werden. Von der Nahtstelle aus muß dann auch ein Distribute Evidence in den alten Junction Tree erfolgen.

2.3 Zusammenfassung

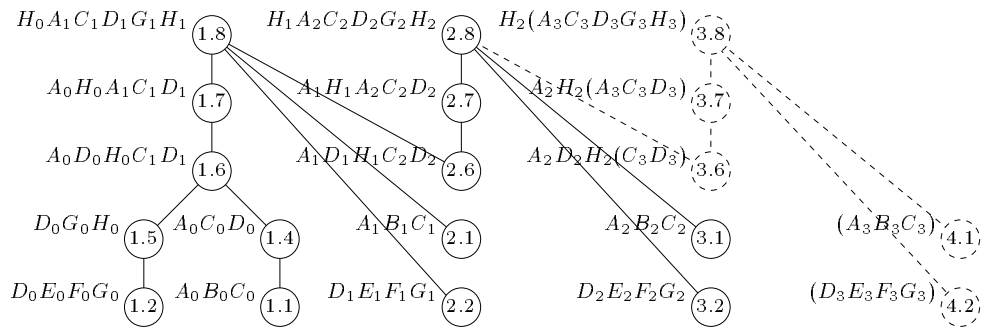
In diesem Kapitel vergleiche ich systematisch die Eigenschaften (zum Beispiel Komplexität und Berechnungersparnis) der beiden Verfahren “Behandlung des Dynamischen Bayesschen Netzes als normales Bayessches Netz” und “Strukturerhaltung im Junction Tree des Dynamischen Bayesschen Netzes” miteinander. Mit dem zweiten Verfahren, das auf einem von Kjærulff in [Kjærulff 93] vorgestellten Verfahren beruht, werden die Grundlagen für einige Rollup-Verfahren geschaffen. Diese Rollup-Verfahren (darunter auch das Rollup-Verfahren von Kjærulff) werden im folgenden Kapitel 3 vorgestellt. Dadurch daß sich die Struktur des Dynamischen Bayesschen Netzes im Junction Tree widerspiegelt, wird ein Rollup von Dynamischen Bayesschen Netzen ermöglicht, der ohne Informationsverlust durchgeführt werden kann.

Wie sich aber schon im Beispielvergleich zeigte, benötigt der Junction Tree, der strukturerhaltend erzeugt wurde, vorübergehend mehr Rechenleistung als der Junction Tree, bei dem eine globale Strategie zum Einsatz kommt.

Es wäre ideal, ein Verfahren zu haben, daß die Vorzüge der globalen und der strukturerhaltenden Strategie miteinander verbindet. Man wird sehen, daß dies mithilfe der sogenannten *Polynompropagierung* möglich ist. Im folgenden Kapitel 3 werden nun einige Rollup-Verfahren vorgestellt, die je nach Gegebenheit und Anwendung ihre Schwächen und Stärken haben. Eine Gegenüberstellung und Zusammenfassung von



(b) der Junction Tree für die Zeitscheiben 0 bis 2



(c) der Junction Tree nach dem Anhängen der Zeitscheibe 3

Abbildung 2.12: In der Repräsentation des Junction Trees wird die Zeitscheibe 3 angehängt. Die neuen Cliques sind gestrichelt eingezeichnet. Es wurden keine alten Cliques verändert.

Kriterien erleichtern die Wahl, welches Rollup-Verfahren für die Anwendung am Besten zum Einsatz kommt. Gegebenenfalls lassen sich einige Rollup-Verfahren sogar miteinander kombinieren.

Lösungsverfahren, die bei der Bestimmung der Eliminationsreihenfolge der Knoten den Algorithmus MAXIMUM CARDINALITY SEARCH oder ähnliche zwingend vorschreiben, sind für dieses Rollup-Verfahren nicht geeignet.

Kapitel 3

Rollup für Dynamische Bayessche Netze

Durch das Anhängen von Zeitscheiben wächst bei Dynamischen Bayesschen Netzen der Bedarf an *Ressourcen* wie Rechenzeit und Arbeitsspeicher. Zu gewissen Zeitpunkten wird es zwingend erforderlich, Dynamische Bayessche Netz an die vorhandene Rechenzeit und/oder den vorhandenen Arbeitsspeicher anzupassen. Diese Anpassung kann durch ein Rollup-Verfahren geschehen. Im allgemeinen werden dabei ganze Zeitscheiben des Dynamischen Bayesschen Netzes aufgerollt¹ und maximal zwei Zeitscheiben des Dynamischen Bayesschen Netzes verwaltet.

Im folgenden werden für die verschiedensten Bedürfnisse Rollup-Verfahren vorgestellt, die die Inferenzverfahren nutzen, die in dieser Arbeit vorgestellt werden (siehe Kapitel 1). Durch diese Rollup-Verfahren werden dann Vorhersagen und Interpretationen beliebig vieler Beobachtungen ermöglicht.

Wie man sehen wird, läßt sich der Rollup von Dynamischen Bayesschen Netzen nicht so einfach durch das Abschneiden von Knoten (oder ganzen Zeitscheiben) realisieren. Der hierdurch verursachte Fehler könnte so groß sein oder durch Akkumulation (beim Abschneiden mehrerer Zeitscheiben nacheinander) so groß werden, daß die durch die Propagierung gewonnenen Einschätzungen unbrauchbar sind.

3.1 Motivation

In diesem Abschnitt wird gezeigt, wie sich schon in einem kleinen Bayesschen Netz die BEL-Werte der Knoten ändern, wenn eine Netzstruktur aufgebrochen oder ein Knoten des Bayesschen Netzes gelöscht wird.

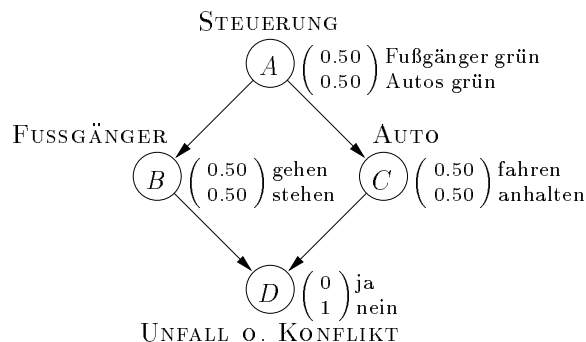
Das Abschneiden eines Knotens K in einem Bayesschen Netz kann als Aufspalten des Netzes an der Stelle des Knotens K mit anschließendem Löschen des Knotens K betrachtet werden.

¹Eine aufgerollte Zeitscheibe bezeichnet Thomas Weis in seiner Dissertation (siehe [Weis 00]) auch als *abgeschlossen*. Dieser Ausdruck verdeutlicht sehr anschaulich, daß keine Einschätzungen für die Knoten dieser Zeitscheibe mehr vorgenommen werden können.

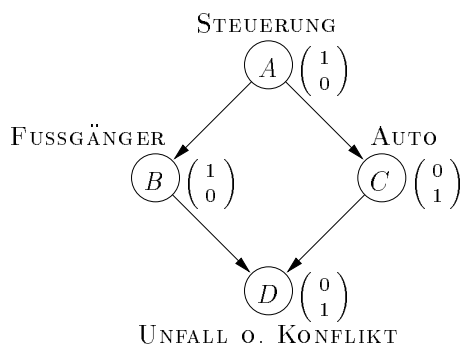
Im folgenden Beispiel wird man sehen, wie aber schon ein einfaches Aufspalten der Netzstruktur eines Bayesschen Netzes die BEL-Werte der Knoten des Bayesschen Netzes verändert.

Beispiel 3.1.1 (Aufspalten bzw. Löschen eines Knotens)

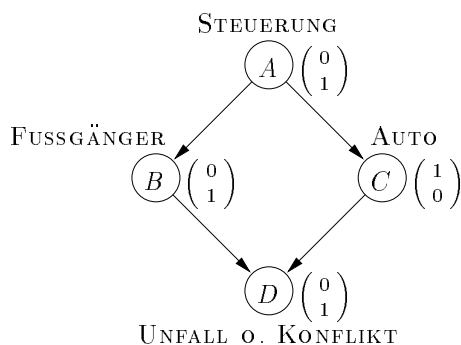
Sei der Graph aus Abbildung 3.1 mit den a-priori und bedingten Wahrscheinlichkeiten



(a) Anfangswahrscheinlichkeiten



(b) Wahrscheinlichkeiten bei der Instantiierung des Knotens A mit a_1



(c) Wahrscheinlichkeiten bei der Instantiierung des Knotens A mit a_2

Abbildung 3.1: Beispielnetz Ampelsteuerung $[P(A) \cdot P(B | A) \cdot P(C | A) \cdot P(D | B, C)]$

in Tabelle 3.1 gegeben.

Es ist ein einfaches Bayessches Netz zur Darstellung des Sachverhaltes ‘Ampelsteuerung’ an einer einfachen Straße zur Straßenüberquerung. Eine zentrale STEUERUNG gibt an, ob die AUTOS fahren oder die FUSSGÄNGER die Straße überqueren dürfen. Dürfen die Autos fahren, so zeigt die STEUERUNG für die Autos grün und die Fußgänger rot an. Geht man davon aus, daß sich alle Verkehrsteilnehmer an die Straßenverkehrsordnung halten, so ist die Wahrscheinlichkeit für einen UNFALL gleich null, selbst wenn man nicht weiß, ob die Ampel jetzt für die Fußgänger oder die Autos grün anzeigt (siehe Abbildung 3.1(a)). Für die verschiedenen Instantiierungen des Knotens A sind die BEL-Werte der anderen Knoten in den Abbildungen 3.1(a), (b) und (c) dargestellt.

Jetzt wird das obige Beispiel leicht abgeändert (siehe Abbildung 3.2). Anstatt einer zentralen Steuerung sind zwei separate Steuerungen (Knoten A_1 und Knoten A_2) modelliert. Die a-priori Wahrscheinlichkeiten der neuen Knoten sind in Tabelle 3.2 gegeben. Die Wahrscheinlichkeiten der anderen Knoten bleiben erhalten. Man erhält ein

STEUERUNG (A : $a_1 = \text{Fußgänger grün}$, $a_2 = \text{Autos grün}$)		
$P(a) :$	$P(a_1) = 0.50$	$P(a_2) = 0.50$
FUSSGÄNGER (B : $b_1 = \text{gehen}$, $b_2 = \text{stehen}$)		
$P(b a) :$	$P(b_1 a_1) = 1.00$	$P(b_2 a_1) = 0.00$
	$P(b_1 a_2) = 0.00$	$P(b_2 a_2) = 1.00$
AUTO (C : $c_1 = \text{fahren}$, $c_2 = \text{anhalten}$)		
$P(c a) :$	$P(c_1 a_1) = 0.00$	$P(c_2 a_1) = 1.00$
	$P(c_1 a_2) = 1.00$	$P(c_2 a_2) = 0.00$
UNFALL O. KONFLIKT (D : $d_1 = \text{ja}$, $d_2 = \text{nein}$)		
$P(d b, c) :$	$P(d_1 b_1, c_1) = 1.00$	$P(d_2 b_1, c_1) = 0.00$
	$P(d_1 b_1, c_2) = 0.00$	$P(d_2 b_1, c_2) = 1.00$
	$P(d_1 b_2, c_1) = 0.00$	$P(d_2 b_2, c_1) = 1.00$
	$P(d_1 b_2, c_2) = 1.00$	$P(d_2 b_2, c_2) = 0.00$

Tabelle 3.1: bedingte Wahrscheinlichkeiten bzw. a-priori Wahrscheinlichkeiten für die Knoten des Bayesschen Netzes **Ampelsteuerung** (siehe Abbildung 3.1)

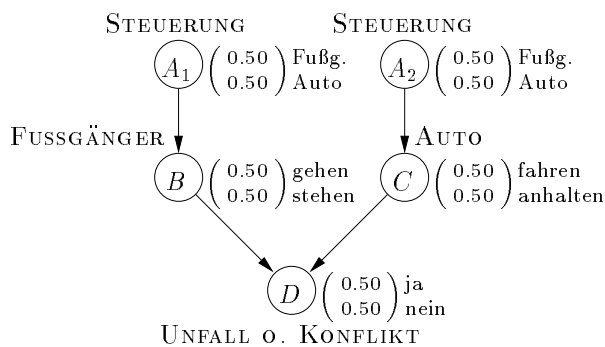
STEUERUNG (A_1, A_2 : $a_1 = \text{Fußgänger}$, $a_2 = \text{Auto}$)		
$P(a) :$	$P(a_1) = 0.50$	$P(a_2) = 0.50$

Tabelle 3.2: a-priori Wahrscheinlichkeiten für die beiden Knoten A_1 und A_2 des aufgespaltenen Knotens A des Bayesschen Netzes **Zebrastrreifen** (siehe Abbildung 3.2)

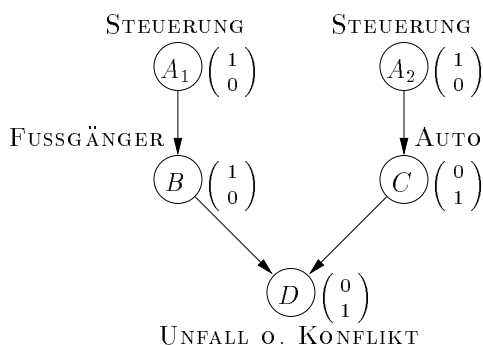
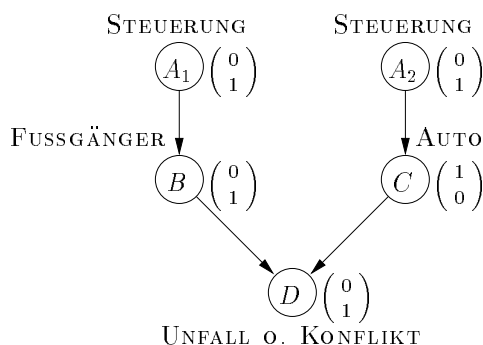
einfaches Bayessches Netz zur Darstellung des Sachverhaltes “Zebrastrreifen” an einer einfachen Straße zur Straßenüberquerung. Die beiden Verkehrsteilnehmer übernehmen die Ampelsteuerung. Signalisiert ein Fußgänger, daß er die Straße überqueren möchte, so wird der Autofahrer am Zebrastrreifen halten und ihn auf die andere Straßenseite lassen. Geht man davon aus, daß sich alle Verkehrsteilnehmer an die Straßenverkehrsordnung halten, so ist die Wahrscheinlichkeit für einen UNFALL gleich null, wenn sich die Verkehrsteilnehmer miteinander (z.B. durch Handzeichen) verständigt haben (siehe Abbildungen 3.2(b) und (c)). Hat diese Kommunikation versagt (z.B. tritt der Fußgänger unerwartet für den Autofahrer an den Zebrastrreifen heran und überquert die Straße, ohne auf die Autos zu achten.), so steigt die Wahrscheinlichkeit in diesem Beispiel auf 50% (siehe Abbildung 3.2(a)). Man kann in diesem Beispiel davon ausgehen, daß die beiden Knoten A_1 und A_2 immer gleich instantiiert sind (bei Kommunikation beide auf die erste oder beide auf die zweite Hypothese der beiden Knoten oder bei keiner Kommunikation keine von beiden Hypothesen).

Das neue Bayessche Netz **Zebrastrreifen** kann man damit vergleichen, daß der Knoten A des Bayesschen Netzes **Ampelsteuerung** aufgespalten wird. Für verschiedene Instantiiierungen der Knoten A_1 und A_2 sind die BEL-Werte der anderen Knoten in den Abbildungen 3.2(a), (b) und (c) dargestellt.

Im nächsten Bayesschen Netz **Straßenverkehr** (siehe Abbildung 3.3) ist der Sachverhalt einer “Straßenüberquerung” mit schlechten Verhältnissen modelliert, so daß es weder eine Ampel oder ein Zebrastrreifen noch eine andere Möglichkeit zur Mitteilung an einen Autofahrer gibt (z.B. durch eine starke Kurve oder verminderte Sichtverhältnisse wie Nebel oder Dunkelheit), daß man als Fußgänger die Straße überqueren möchte. Die Knoten A_1 und A_2 wie im Beispielnetz **Zebrastrreifen** oder der Knoten A



(a) Anfangswahrscheinlichkeiten

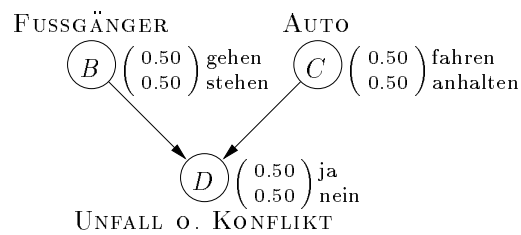
(b) Wahrscheinlichkeiten bei der Instantiierung des Knotens A mit a_1 (c) Wahrscheinlichkeiten bei der Instantiierung des Knotens A mit a_2 Abbildung 3.2: Beispielnetz **Zebrastrreifen** [$P(A_1) \cdot P(A_2) \cdot P(B | A_1) \cdot P(C | A_2) \cdot P(D | B, C)$]

wie im Beispielnetz **Ampelsteuerung** existieren im Bayesschen Netz **Straßenverkehr** nicht mehr. Die Knoten B und C erhalten a-priori Wahrscheinlichkeiten (siehe Tabelle 3.3), die Wahrscheinlichkeiten für den Knoten D bleiben aus dem Vorgängernetz erhalten.

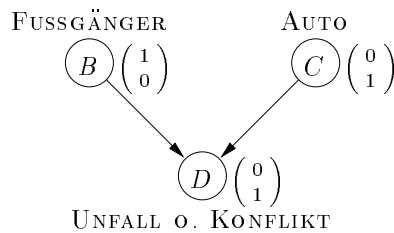
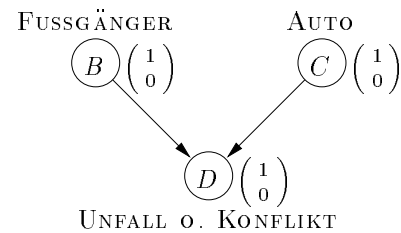
FUSSGÄNGER (B : $b_1 = \text{gehen}$, $b_2 = \text{stehen}$)	
$P(b a)$:	$P(b_1) = 0.50 \quad P(b_2) = 0.50$
AUTO (C : $c_1 = \text{fahren}$, $c_2 = \text{anhalten}$)	
$P(c a)$:	$P(c_1) = 0.50 \quad P(c_2) = 0.50$

Tabelle 3.3: a-priori Wahrscheinlichkeiten für die beiden Knoten B und C des Bayesschen Netzes **Straßenverkehr** (siehe Abbildung 3.3)

Wie man aus den BEL-Werten, die in den Abbildungen 3.3(a), (b) und (c) ablesen kann, ist die Überquerung der Fahrbahn für den Fußgänger ein reines Glücksspiel. Wir der Passant bei der Überquerung der Fahrbahn rechtzeitig von einem ankommenden Fahrzeug bemerkt und kann dieses anhalten, so kommt es zu keinem Unfall (siehe Abbildung 3.3(b)). Reagiert ein Autofahrer jedoch zu spät, so kommt es zu einem Zusammenstoß (siehe Abbildung 3.3(c)).



(a) Anfangswahrscheinlichkeiten

(b) Wahrscheinlichkeiten bei der Instantiierung des Knotens B mit b_1 und des Knotens C mit c_2 (c) Wahrscheinlichkeiten bei der Instantiierung des Knotens B mit b_1 und des Knotens C mit c_1 Abbildung 3.3: Beispielnetz Straßenverkehr $[P(B) \cdot P(C) \cdot P(D | B, C)]$

Das Bayessche Netz **Straßenverkehr** kann man mit dem Bayesschen Netz **Ampelsteuerung** damit vergleichen, daß der Knoten A des Bayesschen Netzes **Ampelsteuerung** gelöscht wird und die beiden Nachfolgerknoten B und C , den BEL-Wert des Knotens A nach der Initialisierung des Bayesschen Netzes als a-priori Wahrscheinlichkeiten eingetragen bekommen.

Bsp

Wie man im Beispiel gesehen hat, entsteht beim Aufspalten oder Löschen eines Knotens in einem Bayesschen Netz ein Fehler. Dieser Fehler kann dabei so groß sein, daß die durch Inferenz gewonnenen BEL-Werte der Knoten nur noch bedingt brauchbar sind. Wie sich ein fehlerfreier Rollup realisieren läßt, das wird nun im folgenden Abschnitt vorgestellt.

3.2 Algorithmen

Wie man gesehen hat, ändert sich beim Aufspalten oder Löschen von Knoten im Bayesschen Netz die Gesamtwahrscheinlichkeit der verbleibenden Knoten. Es entsteht ein Fehler. Im folgenden werden verschiedene Algorithmen vorgestellt, die beim Entfernen von Zeitscheiben (Es werden also alle Knoten einer Zeitscheibe entfernt.) die Gesamtwahrscheinlichkeit der verbleibenden Zeitscheiben erhalten.

Die folgenden Algorithmen erhalten die Gesamtwahrscheinlichkeit der verbleibenden Zeitscheiben, indem entweder die Struktur des Dynamischen Bayesschen Netzes

beim Rollup verändert wird, oder der Rollup auf dem zugehörigen Junction Tree des Dynamischen Bayesschen Netzes ausgeführt wird. Als nächstes wird ein Algorithmus vorgestellt, der die Struktur des Bayesschen Netzes verändert.

3.2.1 Knotenabsorption

Das Verfahren der *Knotenabsorption* dient dazu, uninteressante Knoten aus dem Bayesschen Netz zu entfernen, und dadurch gegebenenfalls die Inferenz im neuen Bayesschen Netz zu vereinfachen. Wie man aber sehen wird, kann bei der Absorption eines Knotens oder einer Knotenmenge die Komplexität des Bayesschen Netzes im Gegenteil auch steigen.

Es wird nun gezeigt, ob und wie sich das Verfahren der Knotenabsorption als Rollup-Verfahren einsetzen läßt.

Das Verfahren der *Knotenabsorption* formt ein Bayessches Netz um, indem Knoten des Netzes so entfernt werden, daß jede Inferenz des resultierenden neuen Bayesschen Netzes dasselbe Ergebnis liefert wie das alte unveränderte Bayessche Netz. Für die absorbierten Knoten können aber keine neuen Evidenzen eingetragen oder die BEL-Werte abgelesen werden. Die gesamte gemeinsame Wahrscheinlichkeitsverteilung der verbleibenden Knoten bleibt unverändert. Das umgewandelte Bayessche Netz muß wieder in einen Junction Tree compiliert werden.

Bei der Absorption eines Knotens $A \in \mathbf{X}$ werden die Tabellen der bedingten Wahrscheinlichkeiten der verbleibenden Knoten $V \in \mathbf{X}$ so verändert, daß die Gesamtwahrscheinlichkeit des Bayesschen Netzes ohne Berücksichtigung des absorbierten Knotens gleich bleibt. Formal bedeutet dies:

$$\forall \mathbf{K} \subset \mathbf{X} \setminus \{A\} : \sum_{\mathbf{X} \setminus \mathbf{K}} \left(\prod_{V \in \mathbf{X}} T_{\text{cpt}(V)} \cdot T_{\text{finding}(V)} \right) = \sum_{\mathbf{X} \setminus \mathbf{K}} \left(\prod_{V \in \mathbf{X} \setminus \{A\}} \tilde{T}_{\text{cpt}(V)} \cdot T_{\text{finding}(V)} \right)$$

Die Absorption eines Knotens A läßt sich zu einer Absorption einer Knotenmenge $\mathbf{A} \subset \mathbf{X}$ fortsetzen. Je nachdem in welcher Reihenfolge die Knoten aus dem Bayesschen Netz absorbiert werden, ändert sich dabei das Ergebnis der Knotenabsorption. Bei der Veränderung der Tabellen der bedingten Wahrscheinlichkeiten eines Knotens des Bayesschen Netzes, sind nicht nur die Zahlenwerte der bedingten Wahrscheinlichkeiten betroffen, sondern auch die Abhängigkeiten der Knoten untereinander ändern sich. D.h. die Struktur des Bayesschen Netzes ändert sich. Es werden Kanten gelöscht und eingefügt.

In der Abbildung 3.4 ist das Beispielnetz **Asienbesuch** aus dem Beispiel 1.1.2 zu sehen, an dem nun strukturell die Absorption zweier Knoten dargestellt wird. (Für die Berechnungen wurde das Programm NeticaTM herangezogen.) Die Anzahl der Einträge in der Wahrscheinlichkeitentabelle eines Knotens V wird auch als Größe der Wahrscheinlichkeitentabelle oder mit $\#(T_{\text{cpt}(V)})$ bezeichnet. Die Größen der Wahrscheinlichkeitentabellen aller Knoten des Bayesschen Netzes addieren sich zu $\#(T_{\text{cpt}(A)}) + \#(T_{\text{cpt}(B)}) + \#(T_{\text{cpt}(C)}) + \#(T_{\text{cpt}(D)}) + \#(T_{\text{cpt}(E)}) + \#(T_{\text{cpt}(F)}) + \#(T_{\text{cpt}(G)}) + \#(T_{\text{cpt}(H)}) = 2 + 4 + 8 + 4 + 4 + 2 + 4 + 8 = 36$.

Jetzt wird der uninstanzierte Knoten C des Bayesschen Netzes absorbiert. Wie in der Abbildung 3.5 zu sehen ist, werden Kanten vom Knoten B zu den Knoten D und H sowie Kanten vom Knoten E zu den Knoten D und H eingefügt. Dabei werden

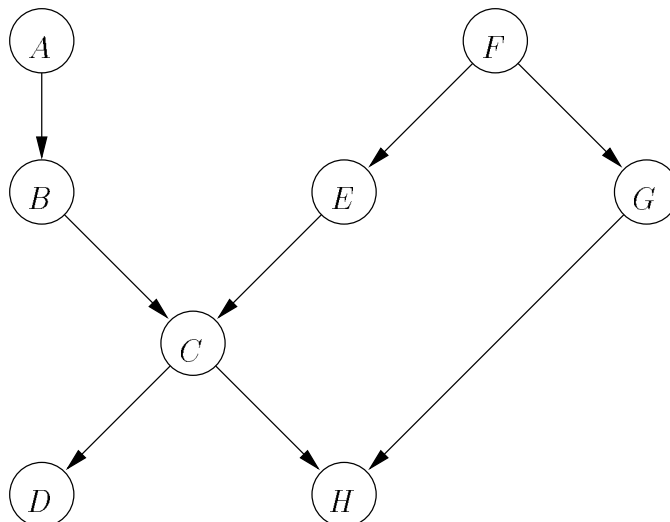


Abbildung 3.4: Knotenabsorption: Das Beispielnetz **Asienbesuch** (siehe Beispiel 1.1.2) dient als Ausgangsnetz. Die Größen der Wahrscheinlichkeitstabellen der Knoten addieren sich zu $\#(T_{\text{cpt}(A)}) + \#(T_{\text{cpt}(B)}) + \#(T_{\text{cpt}(C)}) + \#(T_{\text{cpt}(D)}) + \#(T_{\text{cpt}(E)}) + \#(T_{\text{cpt}(F)}) + \#(T_{\text{cpt}(G)}) + \#(T_{\text{cpt}(H)}) = 2 + 4 + 8 + 4 + 4 + 2 + 4 + 8 = 36$.

die Einträge der Wahrscheinlichkeitstabellen und die Größe der Wahrscheinlichkeitstabellen der beiden Knoten D und H vergrößert. Es gilt: $\#(\tilde{T}_{\text{cpt}(D)}) = 8$ und $\#(\tilde{T}_{\text{cpt}(H)}) = 16$. Die Größen der Wahrscheinlichkeitstabellen aller Knoten des Bayesschen Netzes addieren sich zu $\#(T_{\text{cpt}(A)}) + \#(T_{\text{cpt}(B)}) + 0 + \#(\tilde{T}_{\text{cpt}(D)}) + \#(T_{\text{cpt}(E)}) + \#(T_{\text{cpt}(F)}) + \#(T_{\text{cpt}(G)}) + \#(\tilde{T}_{\text{cpt}(H)}) = 2 + 4 + 0 + 8 + 4 + 2 + 4 + 16 = 40$.

Nun wird der Knoten H instantiiert. Dadurch werden unter anderem die Knoten G und A voneinander abhängig. Die Abhängigkeiten von Knoten(mengen) untereinander lassen sich mit dem folgenden Kriterium bestimmen:

Definition 3.2.1 (D-Separations-Kriterium)

Seien \mathbf{X} , \mathbf{Y} und \mathbf{Z} drei disjunkte Teilmengen von Knoten des gerichteten azyklischen Graphen \mathcal{G} des Bayesschen Netzes \mathcal{BN} . Dabei muß die Vereinigung von \mathbf{X} , \mathbf{Y} und \mathbf{Z} nicht die gesamte Knotenmenge des Graphen \mathcal{G} ergeben. Dann d -separiert die Menge \mathbf{Z} die beiden Mengen \mathbf{X} und \mathbf{Y} voneinander (Schreibweise: $\langle \mathbf{X} \mid \mathbf{Z} \mid \mathbf{Y} \rangle_{\mathcal{G}}$), wenn es entlang jeden Pfades zwischen einem Knoten aus \mathbf{X} und einem Knoten aus \mathbf{Y} ein Knoten w existiert, der eine der beiden nachfolgenden Eigenschaften erfüllt:

1. w hat Vorgängerknoten und weder w noch seine Nachfolgerknoten sind in \mathbf{Z} , oder
2. w hat keine Vorgängerknoten, und w ist in \mathbf{Z} .

□

Weitere Begriffe und ein Beispiel im Zusammenhang mit dem D -Separations-Kriterium sind für den interessierten Leser im Anhang A.4 zusammengefaßt.

Nun aber wieder zurück vom D -Separations-Kriterium zur Absorption des Knotens H . Es werden Kanten vom Knoten G zu den Knoten A , B und E sowie vom Knoten E zu den Knoten A und B eingefügt. Dabei werden die Einträge der Wahrscheinlichkeitstabellen und die Größe der Wahrscheinlichkeitstabellen der Knoten A , B und E

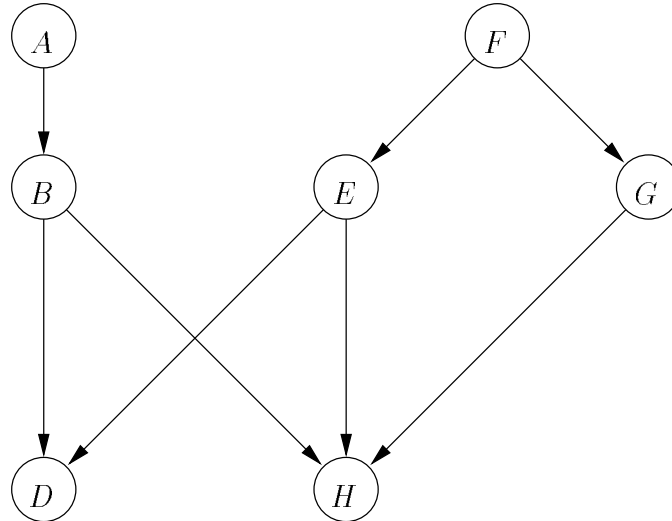


Abbildung 3.5: Knotenabsorption: Der uninstanzierte Knoten C wurde absorbiert. Die Größen der Wahrscheinlichkeitentabellen der Knoten addieren sich zu $\#(T_{\text{cpt}(A)}) + \#(T_{\text{cpt}(B)}) + 0 + \#(\tilde{T}_{\text{cpt}(D)}) + \#(T_{\text{cpt}(E)}) + \#(T_{\text{cpt}(F)}) + \#(T_{\text{cpt}(G)}) + \#(\tilde{T}_{\text{cpt}(H)}) = 2 + 4 + 0 + 8 + 4 + 2 + 4 + 16 = 40$.

vergrößert. Es gilt: $\#(\tilde{T}_{\text{cpt}(A)}) = 16$, $\#(\tilde{T}_{\text{cpt}(B)}) = 16$ und $\#(\tilde{T}_{\text{cpt}(E)}) = 8$. Die Größen der Wahrscheinlichkeitentabellen aller Knoten des Bayesschen Netzes addieren sich zu $\#(\tilde{T}_{\text{cpt}(A)}) + \#(\tilde{T}_{\text{cpt}(B)}) + 0 + \#(\tilde{T}_{\text{cpt}(D)}) + \#(\tilde{T}_{\text{cpt}(E)}) + \#(T_{\text{cpt}(F)}) + \#(T_{\text{cpt}(G)}) + 0 = 8 + 16 + 0 + 8 + 8 + 2 + 4 + 0 = 46$.

Um mit der Absorption von Knotenmengen einen Rollup durchführen zu können, geht man nun wie folgt vor:

- Das Bayessche Netz besteht aus zwei Zeitscheiben $t - 1$ und t . Die Knoten der Zeitscheiben vor der Zeitscheibe $t - 1$ wurden alle absorbiert. Liegen Evidenzen für Knoten der Zeitscheibe $t - 1$ vor, so werden sie eingetragen.
- Die Knoten der Zeitscheibe $t - 1$ werden absorbiert.
- Es wird die Zeitscheibe $t + 1$ angehängt, und das Bayessche Netz ist für den nächsten Berechnungsvorgang bereit.

Obwohl ein Bayessches Netz, von dem einige Knoten absorbiert wurden, weniger Knoten als das vorhergehende Bayessche Netz enthält, kann es, wenn Kanten durch die Absorption eingefügt wurden, sehr viel komplexer sein als das vorhergehende Bayessche Netz. Es werden also Kanten ins Bayessche Netz eingefügt bei der Absorption

- eines Knotens mit Evidenz, der viele Vorgängerknoten besitzt, oder
- eines Knotens ohne Evidenz, der viele Nachfolgerknoten besitzt.

Es werden keine neuen Kanten im Netz eingefügt bei der Absorption von

- Knoten mit Evidenz, die keine Vorgängerknoten besitzen, oder
- Knoten ohne Evidenz, die keine Nachfolgerknoten besitzen.

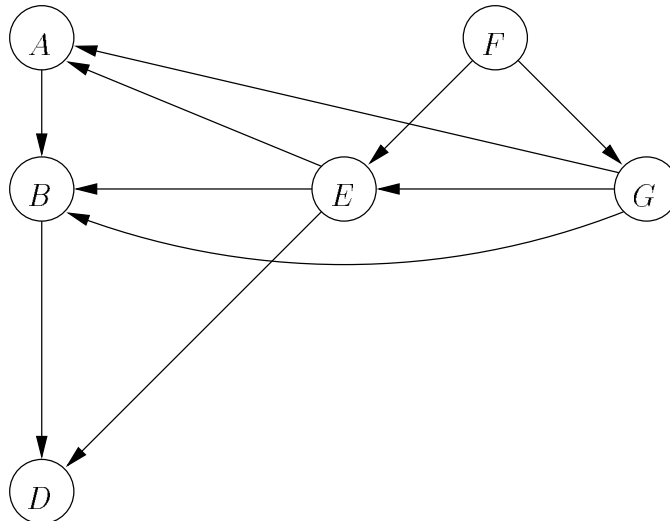


Abbildung 3.6: Knotenabsorption: Der instantiierte Knoten H wird absorbiert. Die Größen der Wahrscheinlichkeitstabellen der Knoten addieren sich zu $\#(\tilde{T}_{\text{cpt}(A)}) + \#(\tilde{T}_{\text{cpt}(B)}) + 0 + \#(\tilde{T}_{\text{cpt}(D)}) + \#(\tilde{T}_{\text{cpt}(E)}) + \#(T_{\text{cpt}(F)}) + \#(T_{\text{cpt}(G)}) + 0 = 8 + 16 + 0 + 8 + 8 + 2 + 4 + 0 = 46$.

Mit Hilfe des D-Separations-Kriteriums weiß man zum Beispiel, daß durch die Instantiierung eines Knotens mit Evidenz seine Vorgängerknoten voneinander abhängig werden.

Ist die Anzahl der Knoten gering, für die Evidenzen eingetragen werden sollen oder an deren BEL-Werten man interessiert ist, und haben diese Knoten auch nur eine geringe Anzahl an Hypothesen, so ist auch das resultierende Bayessche Netz nach der Absorption der uninteressanten Knoten von geringer Größe. Der Berechnungsvorgang kann aber vorübergehend so enorme Speicherplatzprobleme aufwerfen, daß er wegen Speicherplatzmangel gegebenenfalls sogar abbricht.

3.2.2 Prediction-Estimation Verfahren

Beim *Prediction-Estimation Verfahren* (siehe auch [Russell & Norvig 95]) sind immer maximal nur zwei Zeitscheiben des Dynamischen Bayesschen Netzes repräsentiert. Dabei wird in dem Dynamischen Bayesschen Netz eine neue Zeitscheibe etabliert und dann eine alte Zeitscheibe aufgerollt.

Das Anhängen einer Zeitscheibe t an ein Dynamisches Bayessches Netz kann mathematisch so betrachtet werden, daß an das Produkt von bedingten Wahrscheinlichkeitstabellen und findings der Knoten der alten Zeitscheiben, die bedingten Wahrscheinlichkeitstabellen und findings der Knoten der neuen Zeitscheibe t hinzumultipliziert werden²:

²Die Schreibweise $K \in \mathbf{X}$ ist ein wenig nachlässig. Ein Knoten K kann nie das Element einer Menge von Zufallsvariablen sein. Vielmehr ist mit diesem Ausdruck gemeint, daß K ein Knoten ist, der mit einer Zufallsvariablen X_K assoziiert ist, die in \mathbf{X} enthalten ist. Streng mathematisch müßte man also für $K \in \mathbf{X}$ schreiben: $K \in \{V : V \text{ ist assoziiert mit } X_V \in \mathbf{X}\}$.

$$\begin{aligned}
P(\mathbf{X}_t, \dots, \mathbf{X}_0 \mid \mathbf{E}_t, \dots, \mathbf{E}_0) &= P(\mathbf{X}_{t-1}, \dots, \mathbf{X}_0 \mid \mathbf{E}_{t-1}, \dots, \mathbf{E}_0) \cdot P(\mathbf{X}_t \mid \mathbf{E}_t) = \\
&= \left(\prod_{K \in \mathbf{X}_0} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \right) \cdot \dots \cdot \prod_{K \in \mathbf{X}_{t-1}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \cdot \\
&\quad \cdot \prod_{K \in \mathbf{X}_t} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}).
\end{aligned}$$

Dabei stehen die \mathbf{X}_k und \mathbf{E}_k , mit $0 \leq k \leq t$, für die Menge der Zufallsvariablen bzw. die Menge der Evidenzen einer Zeitscheibe k . Die alten Zeitscheiben werden aus dem Produkt entfernt, indem sie herausmarginalisiert werden:

$$\begin{aligned}
\sum_{K \in \mathbf{X}_0 \cup \dots \cup \mathbf{X}_{t-1}} P(\mathbf{X}_t, \dots, \mathbf{X}_0 \mid \mathbf{E}_t, \dots, \mathbf{E}_0) &= \\
&= \sum_{K \in \mathbf{X}_0 \cup \dots \cup \mathbf{X}_{t-1}} \left(\prod_{K \in \mathbf{X}_0} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \right) \cdot \dots \cdot \prod_{K \in \mathbf{X}_{t-1}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \cdot \\
&\quad \cdot \prod_{K \in \mathbf{X}_t} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}).
\end{aligned}$$

In Abbildung 3.7 sind die drei Schritte eines Zyklus des Prediction-Estimation Verfahrens grafisch dargestellt. Jeder Zyklus des Verfahrens arbeitet dabei wie folgt:

- *Prediction*: Das Dynamische Bayessche Netz bestehe aus den beiden Zeitscheiben $t-1$ und t . Weiterhin werde angenommen, daß schon der BEL-Wert $\text{BEL}(\mathbf{X}_{t-1})^3$ berechnet worden sei mit der Berücksichtigung aller Evidenzen bis einschließlich \mathbf{E}_{t-1} . Die Zeitscheibe $t-1$ besitzt keine Verbindungen mehr zu vorhergehenden Zeitscheiben. Den Zustandsvariablen der Zeitscheibe $t-1$ wurden a-priori-Wahrscheinlichkeiten zugeordnet (siehe auch den nächsten Schritt). Dann wird der BEL-Wert $\widehat{\text{BEL}}(\mathbf{X}_t)$ wie folgt neu berechnet:

$$\widehat{\text{BEL}}(\mathbf{X}_t) = \sum_{\mathbf{X}_{t-1}} P(\mathbf{X}_t \mid \mathbf{X}_{t-1} = \mathbf{x}_{t-1}) \text{BEL}(\mathbf{X}_{t-1} = \mathbf{x}_{t-1}).$$

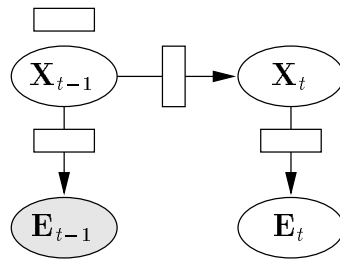
Die \mathbf{x}_{t-1} gehen dabei alle möglichen Werte von \mathbf{X}_{t-1} durch. Die Berechnung läßt sich durch einen Update des aktuellen Bayesschen Netzes (Zeitscheiben $t-1$ und t mit den a-priori Wahrscheinlichkeiten für die Zustandsvariablen der Zeitscheibe $t-1$) mit der Evidenz \mathbf{E}_{t-1} erreichen.

- *Rollup*: Die Zeitscheibe $t-1$ wird entfernt. Dazu muß eine a-priori Wahrscheinlichkeitentabelle für die Zustandsvariablen der Zeitscheibe t hinzugefügt werden. Diese a-priori Wahrscheinlichkeitentabelle ist gerade $\widehat{\text{BEL}}(\mathbf{X}_t)$.

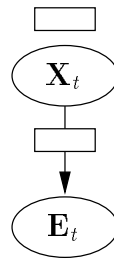
³ $\text{BEL}(\mathbf{X}_{t-1})$ steht für

$$\sum_{K \in \mathbf{X}_0 \cup \dots \cup \mathbf{X}_{t-2}} \left(\prod_{K \in \mathbf{X}_0} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}) \right) \cdot \dots \cdot \prod_{K \in \mathbf{X}_{t-1}} (T_{\text{cpt}(K)} \cdot T_{\text{finding}(K)}).$$

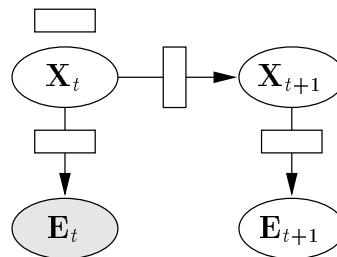
- *Estimation*: Jetzt werden die neuen Evidenzen \mathbf{E}_t eingetragen, und dann der BEL-Wert $\text{BEL}(\mathbf{X}_t)$, d.h. die Wahrscheinlichkeitsverteilung des aktuellen Zustandes, durch ein Update des jetzigen Dynamischen Bayesschen Netzes berechnet. Anschließend wird die Zeitscheibe $t + 1$ angehängt, und das Dynamische Bayesche Netz ist für den nächsten Berechnungsvorgang bereit.



(a) Prediction: Berechnung von $\widehat{\text{BEL}}(\mathbf{X}_t)$ unter Berücksichtigung von $\text{BEL}(\mathbf{X}_{t-1})$ und \mathbf{E}_{t-1} .



(b) Rollup: Die neue a-priori Wahrscheinlichkeitentabelle ist $\widehat{\text{BEL}}(\mathbf{X}_t)$.



(c) Estimation: Berechnung von $\text{BEL}(\mathbf{X}_t)$ bei gegebenen neuen Evidenzen \mathbf{E}_t .

Abbildung 3.7: Die drei Schritte beim Prediction-Estimation Verfahren

Ein Problem bei diesem Verfahren, ist die Speicherung von $\text{BEL}(\mathbf{X}_t)$, des *Zustandsraumes*. Die Tabelle kann unhandlich bis unhabbar groß werden, welches von den involvierten Knoten $X \in \mathbf{X}_t$ und ihrer Hypothesenanzahl abhängig ist. Ebenso findet bei diesem Verfahren keine Neuberechnung alter Zeitscheiben statt.

Soll eine alte Zeitscheibe, die abgeschnitten wurde, neu berechnet werden, kann das Verfahren von Kjærulff (siehe auch [Kjærulff 95]) angewendet werden, das im nachfolgenden Abschnitt vorgestellt wird.

3.2.3 dHugin

In [Kjærulff 92] wird ein Schema zur Handhabung Dynamischer Bayescher Netze vorgestellt. Es heißt dHUGIN und stellt eine Erweiterung von HUGIN dar. Mit dHUGIN werden die Zeitscheiben des Dynamischen Bayesschen Netzes nicht aufgerollt sondern nur aus dem Inferenzprozeß herausgenommen. Die aus dem Inferenzprozeß herausgenommenen Zeitscheiben des Dynamischen Bayesschen Netzes können bei Bedarf wieder in den Inferenzprozeß aufgenommen werden. D.h. werden in einer neuen Zeitscheibe Evidenzen eingetragen, so lassen sich herausgenommene Zeitscheiben wieder in den Inferenzprozeß aufnehmen und somit auffrischen. Dazu wird das Dynamische Bayessche Netz in verschiedene Modelle aufgeteilt.

Da die Zeitscheiben des Dynamischen Bayesschen Netzes die Markov-Eigenschaft erfüllen, kann das Dynamische Bayessche Netz dargestellt werden als eine Reihe von Modellen $\mathcal{P}_1, \dots, \mathcal{P}_{N+1}$. Die Markov-Eigenschaft wird schon in Abschnitt 2.2 dargestellt und hier nur zur Erinnerung noch einmal wiederholt.

Definition 3.2.2 (Markov-Eigenschaft)

Die Zukunft ist unabhängig von der Vergangenheit, wenn die Gegenwart bekannt ist. Formal kann dies auch wie folgt geschrieben werden:

$$\mathbf{X}_{\mathbf{V}(0), \dots, \mathbf{V}(t-1)} \perp\!\!\!\perp \mathbf{X}_{\mathbf{V}(t+1), \dots, \mathbf{V}(t+k)} \mid \mathbf{X}_{\mathbf{V}(t)}, \quad \forall t, k > 0,$$

oder auch mit:

$$P(\mathbf{X}_t \mid \mathbf{X}_1, \dots, \mathbf{X}_{t-1}) = P(\mathbf{X}_t \mid \mathbf{X}_{t-1}).$$

□

Das Modell \mathcal{P}_N wird auch als *Zeitfenster (time window)* bezeichnet, die Modelle $\mathcal{P}_1, \dots, \mathcal{P}_{N-1}$ als *Backward smoothing models* und das Modell \mathcal{P}_{N+1} als *Forecasting model*.

Im Zeitfenster des Dynamischen Bayesschen Netzes sind die Zeitscheiben vertreten, die im Inferenzprozeß enthalten sind. Für die Knoten kann man die BEL-Werte abfragen und Evidenzen eintragen. Das Zeitfenster kann vergrößert, verkleinert und verschoben werden. Das Zeitfenster ist als Junction Tree repräsentiert, der nach dem Verfahren “Strukturerhaltung im Junction Tree des DBNs”, das im Kapitel 2 beschrieben wird, konstruiert ist.

Die Backward smoothing models entsprechen jeweils einer Zeitscheibe, die aus dem Inferenzprozeß herausgenommen wurden. Die einzelnen Modelle $\mathcal{P}_1, \dots, \mathcal{P}_{N-1}$ sind als Junction Tree repräsentiert.

Dem Forecasting model \mathcal{P}_{N+1} entsprechen mehrere Zeitscheiben, die als gerichteter azyklischer Graph repräsentiert sind. Das Modell dient dazu eine schnelle Voraussage über die Entwicklung zu erhalten. Die Inferenz auf diesem gerichteten azyklischen Graphen erfolgt durch *Stochastische Simulation*, da man an einem schnellen Ergebnis interessiert ist, die Genauigkeit aber nicht so wichtig ist. Evidenzen können in diesem Modell nicht eingetragen werden.

Das Zeitfenster wird wie folgt vorwärts geschoben:

- *Vergrößerung des Zeitfensters*: An den Junction Tree des Zeitfensters wird ein neuer Teilbaum angehängt.

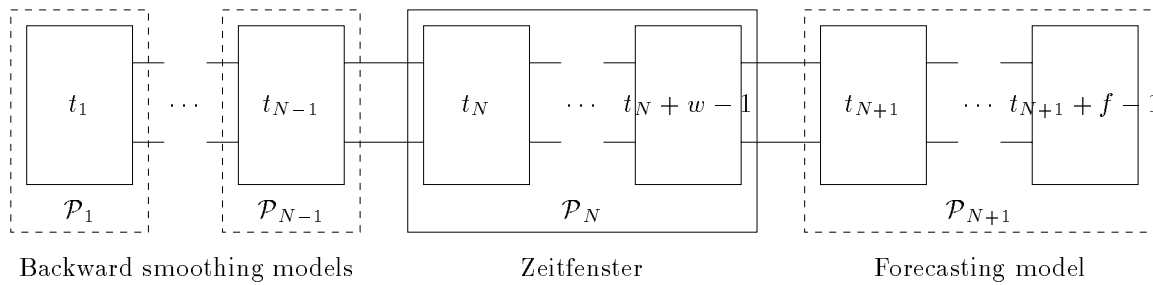


Abbildung 3.8: dHUGIN: Einteilung des DBNs in eine Reihe von Modellen

- *Verkleinerung des Zeitfensters:* Vom Junction Tree des Zeitfensters werden Teilbäume aus dem Inferenzprozeß herausgenommen (sogenanntes *emphpruning*). Dabei wird in diesen Teilbäumen kein Collect Evidenz aufgerufen und auch Distribute Evidenz nicht mehr durchgeführt.

Wie in [Kjærulff 93] beschrieben wird, wird bei der *Vergrößerung des Zeitfensters* um k Zeitscheiben wie folgt vorgegangen:

1. Es werden k neue konsekutive Zeitscheiben dem *Forecasting model* hinzugefügt.
2. Die k ältesten Zeitscheiben des *Forecasting models* werden zum *Zeitfenster* geschoben.
3. Der Graph, der aus dem triangulierten Graphen des *Zeitfensters* und dem gerichteten Graphen der k neuen Zeitscheiben besteht, wird *moralisiert*.
4. Der *moralisierte Graph* wird trianguliert nach dem Verfahren “Strukturerhaltung im Junction Tree des DBNs”, und die neuen Cliques werden bestimmt.
5. Der neue (expandierte) Junction Tree wird erzeugt.
6. Unter Berücksichtigung der alten Cliques werden die Werte für die neuen Cliques berechnet.

Bei der *Verkleinerung des Zeitfensters* um k Zeitscheiben wird wie folgt vorgegangen:

1. Es werden k Teilbäume vom Junction Tree des *Zeitfensters* abgeschnitten. Die k Teilbäume entsprechen dabei den k Zeitscheiben, um die das *Zeitfenster* verkleinert werden soll. (Ein Teilbaum besteht nur aus Cliques, in denen ein Knoten der Zeitscheibe enthalten ist, für den der Teilbaum stehen soll. Ein Teilbaum einer Zeitscheibe enthält in seinen Cliques alle Knoten der Zeitscheibe.)
2. Diese Teilbäume werden ihren entsprechenden *Backward smoothing models* zugeordnet. (Über die Schnittstellen an denen die Teilbäume voneinander getrennt wurden, kann der Informationsaustausch stattfinden.)

Dies ist deswegen so einfach möglich, da der Junction Tree nach dem Verfahren “Strukturerhaltung im Junction Tree des DBNs”, das im Kapitel 2 beschrieben wird, konstruiert ist.

Jedem *Backward smoothing model* \mathcal{P}_n ($1 \leq n \leq N - 1 = b$) ist eine Zeitscheibe t_n zugeordnet, die auch als *Backward smoothing slice* bezeichnet wird.

Das *Forecast model* \mathcal{P}_{N+1} enthält $f = w_{N+1}$ Zeitscheiben, nämlich die Zeitscheiben t_{N+1} bis $t_{N+1} + f - 1$. Diese Zeitscheiben werden auch als *Forecast slices* bezeichnet.

Das Zeitfenster \mathcal{P}_N enthält die w Zeitscheiben $t_N, \dots, t_N + w - 1$, die auch als *Time window slices* bezeichnet werden.

Das aktuelle Dynamische Bayessche Netz besteht somit aus den folgenden Zeitscheiben:

- b Backward smoothing slices,
- w Time window slices und
- f Forecasting slices.

Beschränkt man die maximale Größe des Zeitfensters auf zwei Zeitscheiben (im Zeitfenster muß wenigstens immer eine Zeitscheibe vertreten sein), und erlaubt somit auch nur die Vergrößerung des Zeitfensters um eine Zeitscheibe, so läßt sich das Prediction-Estimation Verfahren simulieren, wenn das Zeitfenster nicht rückwärts geschoben wird. Man erkennt, daß bei dHUGIN ähnliche Probleme wie beim Prediction-Estimation Verfahren auftreten. Jeder Teilbaum des Junction Tree enthält eine Clique mit allen Interfaceknoten der zugehörigen Zeitscheibe. Dynamische Bayessche Netze mit einer großen Anzahl von Interfaceknoten oder Interfaceknoten mit einer großen Hypothesenanzahl (so daß die Clique mit den Interfaceknoten aus Speicherplatzproblemen und/oder Rechenzeitbedarf unhabbar groß wird) werden mit dHUGIN nicht ohne weiteres behandelt werden können.

3.2.4 Polynom-Rollup

Im folgenden wird der Rollup eines Dynamischen Bayesschen Netzes mit Hilfe der Polynom-Propagierung präsentiert. Der Rollup mittels der Polynom-Propagierung ist im Vergleich zu den vorhergehenden Rollup-Verfahren nicht Zeitscheiben- sondern Knotenorientiert. Im Gegensatz zu den Rollup-Verfahren, die vorstehend vorgestellt werden, werden nicht ganze Zeitscheiben berechnet oder aufgerollt, sondern nur gezielt die BEL-Werte von Knoten berechnet, die für den Anwender von Interesse sind. Dazu wird für einen interessierenden Knoten ein multivariates Polynom bestimmt, das durch die Belegung seiner Variablen mit Zahlenwerten vereinfacht werden kann. Man hat es also mit zwei Arten des Rollups zu tun. Der erste Rollup, daß für die uninteressanten Knoten keine Polynome aufgestellt werden und der zweite Rollup, wenn die Polynome der interessanten Knoten vereinfacht werden. Sind die Polynome erst einmal bestimmt, so wird kein Inferenzverfahren für Bayessche Netze mehr benötigt. So läßt sich z.B. ein Rollup auf einem PalmPilot durchführen, für den sich wegen Leistungsbeschränkungen bzgl. Speicherplatz und Rechenleistung keine leistungsfähigen Inferenzverfahren für Bayessche Netze implementieren lassen.

Um zunächst ein Gespür für diese Art des Rollups zu entwickeln und die dahinterstehende Idee zu verstehen, wird dieses Rollup-Verfahren als erstes an einem Beispiel

veranschaulicht, bei dem nur eine Zufallsvariable als Evidenz-Variable dient, dies aber als Instanz in jeder Zeitscheibe des Bayesschen Netzes.

Beispiel 3.2.3 (Polynom-Rollup, eine Zufallsvariable)

Sei das Bayessche Netz *Prüfungsfrage* dynamisch (Zur Erinnerung ist in Abbildung 3.9 der Graph des Bayesschen Netzes dargestellt.) aus dem Beispiel 2.1.2 mit seinem Graphen in Abbildung 2.2 und seinen zugehörigen Wahrscheinlichkeiten in Tabelle 2.1 mit der entsprechenden Anpassung aus Tabelle 2.3 gegeben.

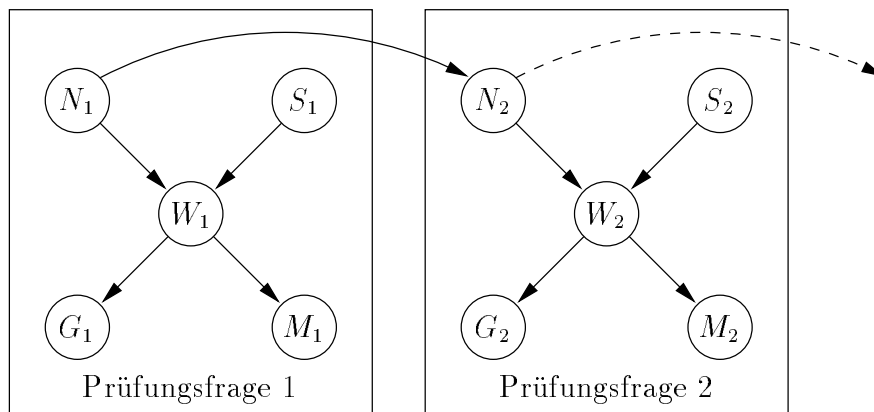


Abbildung 3.9: Polynom-Rollup: Prüfungsfrage dynamisch

Die Zufallsvariable WISSEN DER JEW. ANTWORT sei die einzige Zufallsvariable für die in den einzelnen Zeitscheiben t Evidenzen vorliegen sollen, d.h. der für die Zeitscheibe t zu WISSEN DER JEW. ANTWORT gehörige Knoten W_t wird mit $(W_{t,1}, W_{t,2})$ instantiiert. Man ist nur für den BEL-Wert des Knoten N_t , der für die Zufallsvariable WISSENSNIVEAU DES SCHÜLERS in Zeitscheibe t steht, interessiert. Ist man auch an den BEL-Werten der anderen Knoten interessiert, so geht man analog vor.

Nach der Initialisierung des Netzes ergibt sich für den BEL-Wert von N_0 :

$$\text{BEL}(N_0) = \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}.$$

Nun wird der Knoten W_0 mit $(W_{0,1}, W_{0,2})$ instantiiert, und man erhält für den BEL-Wert des Knotens N_0 das folgende Ergebnis:

$$\begin{aligned} \text{BEL}(N_0) &= \alpha \begin{pmatrix} 0.448 W_{0,1} + 0.352 W_{0,2} \\ 0.130 W_{0,1} + 0.070 W_{0,2} \end{pmatrix} = \\ &= \alpha \begin{pmatrix} 0.560 W_{0,1} + 0.440 W_{0,2} \\ 0.350 W_{0,1} + 0.650 W_{0,2} \end{pmatrix} \cdot \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}. \end{aligned}$$

Jetzt ist noch nicht zu erkennen, weshalb in der obigen Formel der Faktor $\begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}$ herausfaktoriert wird. Der Grund für die Faktorisierung zeigt sich in der nachfolgenden Rechnung. Zusätzlich zum Knoten W_0 wird nun der Knoten W_1 mit $(W_{1,1}, W_{1,2})$ instantiiert. Nach dem Update des Netzes erhält man für den BEL-Wert des Knotens

N_1 :

$$\begin{aligned}
 \text{BEL}(N_1) &= \alpha \begin{pmatrix} (0.25088 W_{0,1} W_{1,1} + 0.19712 W_{0,2} W_{1,1} + \\ 0.19712 W_{0,1} W_{1,2} + 0.15488 W_{0,2} W_{1,2}) \\ (0.02450 W_{0,1} W_{1,1} + 0.04550 W_{0,2} W_{1,1} + \\ 0.04550 W_{0,1} W_{1,2} + 0.08450 W_{0,2} W_{1,2}) \end{pmatrix} = \\
 &= \alpha \begin{pmatrix} 0.560 W_{1,1} + 0.440 W_{1,2} \\ 0.350 W_{1,1} + 0.650 W_{1,2} \end{pmatrix} \cdot \begin{pmatrix} 0.448 W_{0,1} + 0.352 W_{0,2} \\ 0.130 W_{0,1} + 0.070 W_{0,2} \end{pmatrix} = \\
 &= \alpha \underbrace{\begin{pmatrix} 0.560 W_{1,1} + 0.440 W_{1,2} \\ 0.350 W_{1,1} + 0.650 W_{1,2} \end{pmatrix}}_{\text{Zeitscheibe 1}} \cdot \underbrace{\begin{pmatrix} 0.560 W_{0,1} + 0.440 W_{0,2} \\ 0.350 W_{0,1} + 0.650 W_{0,2} \end{pmatrix}}_{\text{BEL}(N_0)} \cdot \underbrace{\begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}}_{\text{Init}}.
 \end{aligned}$$

Man erkennt, wie sich der BEL-Wert des Knotens N_t in der Zeitscheibe t berechnet. Der Polynom-Rollup sieht nun in diesem Beispiel wie folgt aus:

1. Der BEL-Wert von N_0 bestimmt sich zum Zeitpunkt 0 zu

$$\text{BEL}(N_0) = \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}.$$

2. Der BEL-Wert von N_t bestimmt sich für $t > 0$ zum Zeitpunkt t zu

$$\text{BEL}(N_t) = \alpha \begin{pmatrix} 0.560 W_{t,1} + 0.440 W_{t,2} \\ 0.350 W_{t,1} + 0.650 W_{t,2} \end{pmatrix} \cdot \text{BEL}(N_{t-1}).$$

Mit $\text{BEL}(N_{t-1})$ ist der BEL-Wert von N_{t-1} zum Zeitpunkt $t - 1$ gemeint. Zum Zeitpunkt t ist in diesem Fall der BEL-Wert von N_{t-1} identisch mit dem BEL-Wert von N_t .

Das gesammelte Wissen, das für die nachfolgenden Zeitscheiben erforderlich ist, ist bis zum Zeitpunkt $t - 1$ bekannt und im BEL-Wert von N_{t-1} gespeichert.

Bsp

Für jeden Knoten des Dynamischen Bayesschen Netzes läßt sich ein solches Polynom berechnen. Wie man sieht, ist die Komplexität des Polynoms alleine davon abhängig, welche Knoten parametrisiert werden. Diese Komplexität des Polynoms läßt sich dadurch verringern, indem das Polynom faktorisiert dargestellt wird. Allerdings ist die Faktorisierung eines Polynoms nicht immer so leicht durchzuführen, wie im vorigen Beispiel. Werden nämlich mehrere Knoten parametrisiert, so steigt die Komplexität der Polynome enorm, wie man im folgenden Beispiel 3.2.4 sehen wird.

Beim Polynom-Rollup sind zwei Arten von Knoten zu unterscheiden:

1. Die Knoten, die parametrisiert werden, d.h. für die Evidenzen eingetragen werden können.
2. Die Knoten, für die die Polynome aufgestellt werden, d.h. deren BEL-Werte für den Anwender von Interesse sind.

Ein Knoten kann zu beiden Arten gleichzeitig zählen, d.h. er wird parametrisiert und sein BEL-Wert ist für den Anwender interessant. Ebenso kann ein Knoten des Dynamischen Bayesschen Netzes zu keiner von beiden Arten zählen, d.h. für ihn wird weder ein Polynom berechnet noch werden Evidenzen für ihn eingetragen.

Mittels des Polynom-Rollups werden nur noch die BEL-Werte von interessanten Knoten berechnet. Hier werden somit schon die uninteressanten Knoten aufgerollt. Die Polynome für die interessanten Knoten werden faktorisiert (Dies ist nur notwendig, um Speicherplatz zu sparen oder um Regelmäßigkeiten festzustellen.). Ein Polynom eines Knotens wird aufgerollt, indem die Variablen des Polynoms mit Zahlenwerten belegt werden. Dadurch fallen diese Variablen aus dem Polynom heraus, und das Polynom kann vereinfacht werden. Dabei können Variablen neuerer Zeitscheiben auch schon vor Variablen älterer Zeitscheiben mit Zahlenwerten belegt werden. D.h. eine neuere Zeitscheibe wird vor einer älteren Zeitscheibe aufgerollt. Betrachten wir aber den Fall, daß immer nur zwei Zeitscheiben repräsentiert sein sollen, so sieht der Rollup für das Polynom eines Knotens K wie folgt aus:

- Das Polynom besteht aus drei Faktoren. Der erste Faktor ist konstant und enthält das gesammelte Wissen der Vorgängerzeitscheiben, die bis jetzt aufgerollt wurden. Der zweite Faktor steht für die Zeitscheibe $t - 1$, der dritte Faktor für die Zeitscheibe t . Die Variablen im Faktor für die Zeitscheibe $t - 1$ werden mit Zahlenwerten (Je nachdem welche Hypothesen instantiiert werden.) belegt. Der Faktor wird konstant. Der erste und der zweite Faktor werden zusammen multipliziert. Es liegen jetzt nur noch zwei Faktoren des Polynoms vor.
- Es wird ein Faktor für die Zeitscheibe $t + 1$ bestimmt, der dem Polynom als Faktor angehängt wird.

Die Bestimmung des Faktors für die Zeitscheibe $t + 1$ ist leider nicht so einfach und wird im folgenden Abschnitt besprochen.

Jetzt noch ein Beispiel dazu, wie die Komplexität der Polynome steigt, wenn mehrere Knoten parametrisiert werden.

Beispiel 3.2.4 (Polynom-Rollup, zwei Zufallsvariablen)

Sei nun wiederum das Bayessche Netz **Prüfungsfrage** dynamisch aus dem Beispiel 2.1.2 mit seinem Graphen in Abbildung 2.2 und den zugehörigen Wahrscheinlichkeiten in Tabelle 2.1 und der entsprechenden Anpassung aus Tabelle 2.3 gegeben.

Neben der Zufallsvariablen WISSEN DER JEW. ANTWORT sollen nun auch für die Zufallsvariable SCHWIERIGKEIT DER JEW. FRAGE Evidenzen vorliegen, d.h. der für die Zeitscheibe t zu WISSEN DER JEW. ANTWORT gehörige Knoten W_t wird mit $(W_{t,1}, W_{t,2})$ und der für die Zeitscheibe t zu SCHWIERIGKEIT DER JEW. FRAGE gehörige Knoten S_t wird mit $(S_{t,1}, S_{t,2}, S_{t,3})$ instantiiert. Man ist ebenfalls nur für den BEL-Wert des Knoten N_t , der für die Zufallsvariable WISSENSNIVEAU DES SCHÜLERS in Zeitscheibe t steht, interessiert.

Nach der Initialisierung des Netzes ergibt sich für den BEL-Wert von N_0 :

$$\text{BEL}(N_0) = \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}.$$

Nun wird der Knoten W_0 mit $(W_{0,1}, W_{0,2})$ und der Knoten S_0 mit $(S_{0,1}, S_{0,2}, S_{0,3})$ instantiiert, und man erhält für den BEL-Wert des Knotens N_0 :

$$\begin{aligned} \text{BEL}(N_0) &= \alpha \begin{pmatrix} (0.240 S_{0,1} W_{0,1} + 0.240 S_{0,1} W_{0,2} + \\ 0.144 S_{0,2} W_{0,1} + 0.096 S_{0,2} W_{0,2} + \\ 0.064 S_{0,3} W_{0,1} + 0.016 S_{0,3} W_{0,2}) \\ (0.036 S_{0,1} W_{0,1} + 0.084 S_{0,1} W_{0,2} + \\ 0.024 S_{0,2} W_{0,1} + 0.036 S_{0,2} W_{0,2} + \\ 0.010 S_{0,3} W_{0,1} + 0.010 S_{0,3} W_{0,2}) \end{pmatrix} = \\ &= \alpha \begin{pmatrix} (0.30 S_{0,1} W_{0,1} + 0.30 S_{0,1} W_{0,2} + \\ 0.18 S_{0,2} W_{0,1} + 0.12 S_{0,2} W_{0,2} + \\ 0.08 S_{0,3} W_{0,1} + 0.02 S_{0,3} W_{0,2}) \\ (0.18 S_{0,1} W_{0,1} + 0.42 S_{0,1} W_{0,2} + \\ 0.12 S_{0,2} W_{0,1} + 0.18 S_{0,2} W_{0,2} + \\ 0.05 S_{0,3} W_{0,1} + 0.05 S_{0,3} W_{0,2}) \end{pmatrix} \cdot \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}. \end{aligned}$$

Aus bekannten Gründen wird in der obigen Formel der Faktor $\begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}$ herausfaktoriert. Zusätzlich zu den Knoten W_0 und S_0 werden nun die Knoten W_1 bzw. S_1 mit $(W_{1,1}, W_{1,2})$ bzw. $(S_{1,1}, S_{1,2}, S_{1,3})$ instantiiert. Nach dem Update des Netzes erhält man für den BEL-Wert des Knotens N_1 (Die nicht faktorisierte Darstellung, die fast über eine ganze Seite geht, ist in B.3.2 zu sehen.):

$$\begin{aligned} \text{BEL}(N_1) &= \\ &= \alpha(\text{siehe B.3.2}) = \\ &= \alpha \begin{pmatrix} (0.30 S_{1,1} W_{1,1} + 0.30 S_{1,1} W_{1,2} + \\ 0.18 S_{1,2} W_{1,1} + 0.12 S_{1,2} W_{1,2} + \\ 0.08 S_{1,3} W_{1,1} + 0.02 S_{1,3} W_{1,2}) \\ (0.18 S_{1,1} W_{1,1} + 0.42 S_{1,1} W_{1,2} + \\ 0.12 S_{1,2} W_{1,1} + 0.18 S_{1,2} W_{1,2} + \\ 0.05 S_{1,3} W_{1,1} + 0.05 S_{1,3} W_{1,2}) \end{pmatrix} \cdot \begin{pmatrix} (0.240 S_{0,1} W_{0,1} + 0.240 S_{0,1} W_{0,2} + \\ 0.144 S_{0,2} W_{0,1} + 0.096 S_{0,2} W_{0,2} + \\ 0.064 S_{0,3} W_{0,1} + 0.016 S_{0,3} W_{0,2}) \\ (0.036 S_{0,1} W_{0,1} + 0.084 S_{0,1} W_{0,2} + \\ 0.024 S_{0,2} W_{0,1} + 0.036 S_{0,2} W_{0,2} + \\ 0.010 S_{0,3} W_{0,1} + 0.010 S_{0,3} W_{0,2}) \end{pmatrix} = \\ &= \alpha \underbrace{\begin{pmatrix} (0.30 S_{1,1} W_{1,1} + 0.30 S_{1,1} W_{1,2} + \\ 0.18 S_{1,2} W_{1,1} + 0.12 S_{1,2} W_{1,2} + \\ 0.08 S_{1,3} W_{1,1} + 0.02 S_{1,3} W_{1,2}) \\ (0.18 S_{1,1} W_{1,1} + 0.42 S_{1,1} W_{1,2} + \\ 0.12 S_{1,2} W_{1,1} + 0.18 S_{1,2} W_{1,2} + \\ 0.05 S_{1,3} W_{1,1} + 0.05 S_{1,3} W_{1,2}) \end{pmatrix}}_{\text{Zeitscheibe 1}} \cdot \underbrace{\begin{pmatrix} (0.30 S_{0,1} W_{0,1} + 0.30 S_{0,1} W_{0,2} + \\ 0.18 S_{0,2} W_{0,1} + 0.12 S_{0,2} W_{0,2} + \\ 0.08 S_{0,3} W_{0,1} + 0.02 S_{0,3} W_{0,2}) \\ (0.18 S_{0,1} W_{0,1} + 0.42 S_{0,1} W_{0,2} + \\ 0.12 S_{0,2} W_{0,1} + 0.18 S_{0,2} W_{0,2} + \\ 0.05 S_{0,3} W_{0,1} + 0.05 S_{0,3} W_{0,2}) \end{pmatrix}}_{\text{Zeitscheibe 0}} \cdot \underbrace{\begin{pmatrix} 0.8 \\ 0.2 \end{pmatrix}}_{\text{Init}}. \end{aligned}$$

Man erkennt, wie sich der BEL-Wert des Knotens N_t in der Zeitscheibe t berechnet. Der Polynom-Rollup sieht in diesem Beispiel nun wie folgt aus:

1. Der BEL-Wert von N_0 bestimmt sich zum Zeitpunkt 0 zu

$$\text{BEL}(N_0) = \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}.$$

2. Der BEL-Wert von N_t bestimmt sich für $t > 0$ zum Zeitpunkt t zu

$$\text{BEL}(N_t) = \alpha \begin{pmatrix} (0.30 S_{t,1} W_{t,1} + 0.30 S_{t,1} W_{t,2} + \\ 0.18 S_{t,2} W_{t,1} + 0.12 S_{t,2} W_{t,2} + \\ 0.08 S_{t,3} W_{t,1} + 0.02 S_{t,3} W_{t,2}) \\ (0.18 S_{t,1} W_{t,1} + 0.42 S_{t,1} W_{t,2} + \\ 0.12 S_{t,2} W_{t,1} + 0.18 S_{t,2} W_{t,2} + \\ 0.05 S_{t,3} W_{t,1} + 0.05 S_{t,3} W_{t,2}) \end{pmatrix} \cdot \text{BEL}(N_{t-1}).$$

Mit $\text{BEL}(N_{t-1})$ ist der BEL-Wert von N_{t-1} zum Zeitpunkt $t - 1$ gemeint. Zum Zeitpunkt t ist in diesem Fall der BEL-Wert von N_{t-1} identisch mit dem BEL-Wert von N_t .

Es ist also das gesammelte Wissen, das für die nachfolgenden Zeitscheiben erforderlich ist, bis zum Zeitpunkt $t - 1$ bekannt und im BEL-Wert von N_{t-1} gespeichert. Wie in diesem Beispiel zu erkennen war, ist die Komplexität von den beteiligten Knoten (und hier die Anzahl der Hypothesen eines Knotens) abhängig.

Bsp

Die Komplexität dieses Verfahrens ist im Ende, d.h. wenn die Polynome vorliegen, von der Anzahl der parametrisierten Knoten und ihren Hypothesen abhängig. Die Komplexität bei der Bestimmung der Polynome ist abhängig vom Inferenzverfahren, das eingesetzt wird.

3.2.5 Shafer-Shenoy Skripte

Im folgenden wird der Rollup eines Dynamischen Bayesschen Netzes mit Hilfe von Shafer-Shenoy Skripten präsentiert. Die Shafer-Shenoy Skripte dienen dazu, die Faktorisierung eines Polynoms, wie es in der Polynom-Propagierung berechnet wird, einfacher bestimmen zu können. Es wird beispielhaft vorgeführt, wie ein faktorisiertes Polynom direkt aus einem Shafer-Shenoy Skript bestimmt werden kann.

Der Rollup basiert auf der Struktur des zugehörigen Junction Trees \mathcal{T} des Bayesschen Netzes \mathcal{BN} , das aufgerollt werden soll. Für jeden Knoten einer Zeitscheibe wird ein Shafer-Shenoy Skript berechnet. Wird nun eine Zeitscheibe abgeschlossen, so werden die Shafer-Shenoy Skripte der Knoten dieser Zeitscheibe gelöscht. Bei den Shafer-Shenoy Skripten der Knoten einer neuen Zeitscheibe erkennt man, daß sie auf den Berechnungen für die Shafer-Shenoy Skripte der Knoten der alten Zeitscheiben aufbauen. Diese Berechnungen werden "recycled". Somit erfolgt der Rollup ohne Informationsverlust.

Dieser Rollup wird nun an einem Beispielnetz veranschaulicht, das auch schon beim Polynom-Rollup herangezogen wurde.

Beispiel 3.2.5 (Shafer-Shenoy Skripte)

Sei nun das Bayes'sche Netz **Prüfungsfrage dynamisch** aus dem Beispiel 2.1.2 mit seinem Graphen in Abbildung 2.2 und seinen zugehörigen Wahrscheinlichkeiten in Tabelle 2.1 mit der entsprechenden Anpassung aus Tabelle 2.3 gegeben.

Der zugehörige Junction Tree ist in Abbildung 3.10 dargestellt. Die Shafer-Shenoy

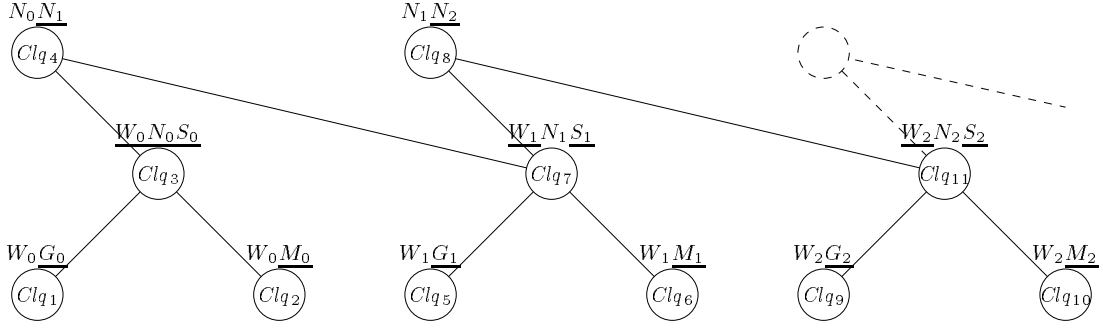


Abbildung 3.10: Rollup: Shafer-Shenoy Skripte, Junction Tree zu Beispielnetz Prüfungsfrage dynamisch

Skripte der Zeitscheibe 0 bestimmen sich ohne Berücksichtigung der Zeitscheibe 1 zu:

$$\begin{aligned}
 T_{\psi}(Clq_1) &= T_{\text{cpt}(G_0)} \cdot \left(\sum_{S_0, N_0} T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot \left(\sum_{M_0} T_{\text{cpt}(M_0)} \right) \right) \\
 T_{\psi}(Clq_2) &= T_{\text{cpt}(M_0)} \cdot \left(\sum_{S_0, N_0} T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot \left(\sum_{G_0} T_{\text{cpt}(G_0)} \right) \right) \\
 T_{\psi}(Clq_3) &= T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot \left(\sum_{G_0} T_{\text{cpt}(G_0)} \right) \cdot \left(\sum_{M_0} T_{\text{cpt}(M_0)} \right)
 \end{aligned}$$

Ohne die Berücksichtigung der Zeitscheibe 2 bestimmen sich die Shafer-Shenoy Skripte der Zeitscheibe 1 zu:

$$\begin{aligned}
 T_{\psi}(Clq_4) &= T_{\text{cpt}(N_1)} \cdot (T_{Clq_3 \triangleright Clq_4}) \cdot (T_{Clq_7 \triangleright Clq_4}) = \\
 &= T_{\text{cpt}(N_1)} \cdot \left(\sum_{W_0, S_0} T_{\psi}(Clq_3) \right) \cdot \left(\sum_{W_1, S_1} T_{\psi}(Clq_7) \right) = \\
 &= T_{\text{cpt}(N_1)} \cdot \left(\sum_{W_0, S_0} T_{\psi}(Clq_3) \right) \cdot \left(\sum_{W_1, S_1} T_{\text{cpt}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot \left(\sum_{G_1} T_{\text{cpt}(G_1)} \right) \cdot \left(\sum_{M_1} T_{\text{cpt}(M_1)} \right) \right) \\
 T_{\psi}(Clq_5) &= T_{\text{cpt}(G_1)} \cdot \left(\sum_{S_1, N_1} T_{\text{cpt}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot \left(\sum_{M_1} T_{\text{cpt}(M_1)} \right) \cdot (T_{Clq_4 \triangleright Clq_7}) \right) = \\
 &= T_{\text{cpt}(G_1)} \cdot \left(\sum_{S_1, N_1} T_{\text{cpt}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot \left(\sum_{M_1} T_{\text{cpt}(M_1)} \right) \cdot \left(\sum_{N_0} T_{\text{cpt}(N_0)} \cdot \sum_{W_0, S_0} T_{\psi}(Clq_3) \right) \right) \\
 T_{\psi}(Clq_6) &= T_{\text{cpt}(M_1)} \cdot \left(\sum_{S_1, N_1} T_{\text{cpt}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot \left(\sum_{G_1} T_{\text{cpt}(G_1)} \right) \cdot (T_{Clq_4 \triangleright Clq_7}) \right) = \\
 &= T_{\text{cpt}(M_1)} \cdot \left(\sum_{S_1, N_1} T_{\text{cpt}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot \left(\sum_{G_1} T_{\text{cpt}(G_1)} \right) \cdot \left(\sum_{N_0} T_{\text{cpt}(N_0)} \cdot \sum_{W_0, S_0} T_{\psi}(Clq_3) \right) \right) \\
 T_{\psi}(Clq_7) &= T_{\text{cpt}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot \left(\sum_{G_1} T_{\text{cpt}(G_1)} \right) \cdot \left(\sum_{M_1} T_{\text{cpt}(M_1)} \right) \cdot (T_{Clq_4 \triangleright Clq_7}) = \\
 &= T_{\text{cpt}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot \left(\sum_{G_1} T_{\text{cpt}(G_1)} \right) \cdot \left(\sum_{M_1} T_{\text{cpt}(M_1)} \right) \cdot \left(\sum_{N_0} T_{\text{cpt}(N_0)} \cdot \sum_{W_0, S_0} T_{\psi}(Clq_3) \right)
 \end{aligned}$$

Der Wert von $T_{\psi}(Clq_3)$ in den vorstehenden Formeln ist ohne die Berücksichtigung der Zeitscheibe 1. Würde die Tabelle $T_{\psi}(Clq_3)$ die Informationen aus Zeitscheibe 1 mitberücksichtigen, so hätte man $T_{Clq_3 \triangleright Clq_4}$ schreiben müssen anstatt $T_{\psi}(Clq_3)$. $T_{Clq_3 \triangleright Clq_4}$ bezeichnet die Informationen, die von Zeitscheibe 0 zu Zeitscheibe 1 fließen.

Für die Bestimmung von $T_{\psi(Cl_{q4})}$ bis $T_{\psi(Cl_{q7})}$ in der Zeitscheibe 1 wird nur noch $T_{\psi(Cl_{q3})}$ aus der Zeitscheibe 0 benötigt.

Ist man an der neuen Einschätzung der Cliques der Zeitscheibe 0 interessiert, so bestimmen sich die Shafer-Shenoy Skripte der Cliques der Zeitscheibe 0 zu:

$$\begin{aligned} T_{\psi(Cl_{q1})} &= T_{\text{cpt}(G_0)} \cdot \left(\sum_{S_0 N_0} T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot \left(\sum_{M_0} T_{\text{cpt}(M_0)} \right) \cdot (T_{Cl_{q4} \triangleright Cl_{q3}}) \right) \\ T_{\psi(Cl_{q2})} &= T_{\text{cpt}(M_0)} \cdot \left(\sum_{S_0 N_0} T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot \left(\sum_{G_0} T_{\text{cpt}(G_0)} \right) \cdot (T_{Cl_{q4} \triangleright Cl_{q3}}) \right) \\ T_{\psi(Cl_{q3})} &= T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot \left(\sum_{G_0} T_{\text{cpt}(G_0)} \right) \cdot \left(\sum_{M_0} T_{\text{cpt}(M_0)} \right) \cdot (T_{Cl_{q4} \triangleright Cl_{q3}}) \end{aligned}$$

In den vorstehenden Formeln bezeichnet $T_{Cl_{q3} \triangleright Cl_{q4}}$ die Informationen, die von Zeitscheibe 1 zu Zeitscheibe 0 fließen. **Bsp**

Mit den Shafer-Shenoy-Skripten läßt sich sowohl ein zielgerichteter Rollup als auch eine Neuinterpretation von Cliques in alten Zeitscheiben durchführen. Beim “normalen” Rollup kann eine Zeitscheibe nur aufgerollt werden, wenn keine Vorgängerzeitscheibe existiert (schon aufgerollt wurde) und alle der Zeitscheibe zugeordneten Cliques für den Benutzer nicht mehr von Interesse sind. Existiert noch eine Vorgängerzeitscheibe oder eine Clique, die noch benötigt wird, so ist kein Rollup möglich. Beim zielgerichteten Rollup werden nun keine ganzen Zeitscheiben mehr abgeschnitten, sondern nur noch die Berechnungsvorschriften für Cliques, die für den Benutzer ohne Interesse sind. Ist man sogar nur an gewissen Knoten in einer Clique interessiert, so läßt sich die Berechnungsvorschrift im allgemeinen weiter kürzer.

Im folgenden werden jetzt die Vorzüge des Polynom-Rollups mit den Vorzügen des Shafer-Shenoy-Skripte-Rollups verbunden.

Beim Polynom-Rollup ist ein großes Problem die faktorisierte Darstellung der Polynome. Im Beispiel waren die Zahlenwerte in den Polynomen exakt. Dies kann im allgemeinen nicht vorausgesetzt werden. Dementsprechend lassen sich die Polynome nicht immer richtig faktorisiert darstellen.

Durch die Shafer-Shenoy Skripte wird zuerst die Faktorisierung für einen Knoten bestimmt und dann das faktorisierte Polynom berechnet. Dies wird nun im folgenden Beispiel vorgeführt.

Beispiel 3.2.6 (faktorisierte Shafer-Shenoy Skripte)

Zu Vergleichszwecken wird wieder dasselbe Beispielnetz wie im vorigen Beispiel herangezogen. Sei nun das Bayes'sche Netz **Prüfungsfrage dynamisch** aus dem Beispiel 2.1.2 mit seinem Graphen in Abbildung 2.2 und seinen zugehörigen Wahrscheinlichkeiten in Tabelle 2.1 mit der entsprechenden Anpassung aus Tabelle 2.3 gegeben. Der zugehörige Junction Tree ist in Abbildung 3.10 dargestellt.

Nach der Initialisierung des Netzes ergibt sich für den BEL-Wert von N_0 :

$$\begin{aligned} \text{BEL}(N_0) &= \\ &= \alpha \cdot \sum_{W_0 S_0} T_{\psi(Cl_{q3})} = \\ &= \alpha \cdot \left(\sum_{W_0 S_0} T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot \left(\sum_{G_0} T_{\text{cpt}(G_0)} \right) \cdot \left(\sum_{M_0} T_{\text{cpt}(M_0)} \right) \right) = \end{aligned}$$

$$\begin{aligned}
&= \alpha \cdot T_{\text{cpt}(N_0)} \cdot \left(\sum_{W_0 S_0} T_{\text{cpt}(W_0)} \cdot T_{\text{cpt}(S_0)} \cdot T_{1(W_0)} \cdot T_{1(W_0)} \right) = \\
&= \alpha \cdot T_{\text{cpt}(N_0)} = \begin{pmatrix} 0.80 \\ 0.20 \end{pmatrix}.
\end{aligned}$$

Nun wird der Knoten W_0 mit $(W_{0,1}, W_{0,2}) = T_{\text{finding}(W_0)}$ und der Knoten S_0 mit $(S_{0,1}, S_{0,2}, S_{0,3}) = T_{\text{finding}(S_0)}$ instantiiert, und man erhält für den BEL-Wert des Knotens N_0 :

$$\begin{aligned}
\text{BEL}(N_0) &= \\
&= \alpha \cdot \sum_{W_0 S_0} T_{\psi(Cl_{q_3})} = \\
&= \alpha \cdot \left(\sum_{W_0 S_0} T_{\text{cpt}(W_0)} \cdot T_{\text{finding}(W_0)} \cdot T_{\text{cpt}(N_0)} \cdot T_{\text{cpt}(S_0)} \cdot T_{\text{finding}(S_0)} \cdot \right. \\
&\quad \cdot \left. \left(\sum_{G_0} T_{\text{cpt}(G_0)} \right) \cdot \left(\sum_{M_0} T_{\text{cpt}(M_0)} \right) \right) = \\
&= \alpha \cdot \left(\sum_{W_0 S_0} T_{\text{cpt}(W_0)} \cdot T_{\text{finding}(W_0)} \cdot T_{\text{cpt}(S_0)} \cdot T_{\text{finding}(S_0)} \cdot T_{1(W_0)} \cdot T_{1(W_0)} \right) \cdot T_{\text{cpt}(N_0)} = \\
&= \alpha \cdot \left(\sum_{W_0 S_0} T_{\text{cpt}(W_0)} \cdot T_{\text{finding}(W_0)} \cdot T_{\text{cpt}(S_0)} \cdot T_{\text{finding}(S_0)} \right) \cdot T_{\text{cpt}(N_0)} = \\
&= \alpha \cdot \left(\sum_{S_0} T_{\text{cpt}(S_0)} \cdot T_{\text{finding}(S_0)} \cdot \left(\sum_{W_0} T_{\text{cpt}(W_0)} \cdot T_{\text{finding}(W_0)} \right) \right) \cdot T_{\text{cpt}(N_0)} = \\
&= \alpha \cdot \begin{bmatrix} n_{0,1} \\ n_{0,2} \end{bmatrix} \begin{bmatrix} (0.30 S_{0,1} W_{0,1} + 0.30 S_{0,1} W_{0,2} \\ 0.18 S_{0,2} W_{0,1} + 0.12 S_{0,2} W_{0,2} \\ 0.08 S_{0,3} W_{0,1} + 0.02 S_{0,3} W_{0,2}) \\ (0.18 S_{0,1} W_{0,1} + 0.42 S_{0,1} W_{0,2} \\ 0.12 S_{0,2} W_{0,1} + 0.18 S_{0,2} W_{0,2} \\ 0.05 S_{0,3} W_{0,1} + 0.05 S_{0,3} W_{0,2}) \end{bmatrix} \cdot \begin{bmatrix} n_{0,1} & | & 0.80 \\ n_{0,2} & | & 0.20 \end{bmatrix}
\end{aligned}$$

Wie man im Verlauf der Rechnung gesehen hat, wurde die Faktorisierung direkt bestimmt.

Zusätzlich zu den Knoten W_0 und S_0 werden nun die Knoten W_1 und S_1 mit $(W_{1,1}, W_{1,2})$ bzw. $(S_{1,1}, S_{1,2}, S_{1,3})$ instantiiert. Nach dem Update des Netzes erhält man für den BEL-Wert des Knotens N_1 :

$$\begin{aligned}
\text{BEL}(N_1) &= \\
&= \alpha \cdot \sum_{W_1 S_1} T_{\psi(Cl_{q_7})} = \\
&= \alpha \cdot \left(\sum_{W_1 S_1} T_{\text{cpt}(W_1)} \cdot T_{\text{finding}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot T_{\text{finding}(S_1)} \cdot \right. \\
&\quad \cdot \left. \left(\sum_{G_1} T_{\text{cpt}(G_1)} \right) \cdot \left(\sum_{M_1} T_{\text{cpt}(M_1)} \right) \cdot \left(\sum_{N_0} T_{\text{cpt}(N_1)} \cdot \left(\sum_{W_0 S_0} T_{\psi(Cl_{q_3})} \right) \right) \right) = \\
&= \alpha \cdot \left(\sum_{W_1 S_1} T_{\text{cpt}(W_1)} \cdot T_{\text{finding}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot T_{\text{finding}(S_1)} \cdot \right. \\
&\quad \cdot T_{1(W_1)} \cdot T_{1(W_1)} \cdot \left. \left(\sum_{N_0} T_{\text{cpt}(N_1)} \cdot \left(\sum_{W_0 S_0} T_{\psi(Cl_{q_3})} \right) \right) \right) =
\end{aligned}$$

$$\begin{aligned}
&= \alpha \cdot \left(\sum_{W_1 S_1} T_{\text{cpt}(W_1)} \cdot T_{\text{finding}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot T_{\text{finding}(S_1)} \cdot \right. \\
&\quad \left. \cdot \left(\sum_{N_0} T_{\text{cpt}(N_1)} \cdot \left(\sum_{W_0 S_0} T_{\psi(Cl_{q_3})} \right) \right) \right) = \\
&= \alpha \cdot \left(\sum_{W_1 S_1} T_{\text{cpt}(W_1)} \cdot T_{\text{finding}(W_1)} \cdot T_{\text{cpt}(S_1)} \cdot T_{\text{finding}(S_1)} \right) \cdot \\
&\quad \cdot \left(\sum_{N_0} T_{\text{cpt}(N_1)} \cdot \left(\sum_{W_0 S_0} T_{\psi(Cl_{q_3})} \right) \right) = \\
&= \alpha \cdot \left[\begin{array}{c|c} n_{1,1} & \begin{array}{l} (0.30 S_{1,1} W_{1,1} + 0.30 S_{1,1} W_{1,2} + \\ 0.18 S_{1,2} W_{1,1} + 0.12 S_{1,2} W_{1,2} + \\ 0.08 S_{1,3} W_{1,1} + 0.02 S_{1,3} W_{1,2}) \end{array} \\ \hline n_{1,2} & \begin{array}{l} (0.18 S_{1,1} W_{1,1} + 0.42 S_{1,1} W_{1,2} + \\ 0.12 S_{1,2} W_{1,1} + 0.18 S_{1,2} W_{1,2} + \\ 0.05 S_{1,3} W_{1,1} + 0.05 S_{1,3} W_{1,2}) \end{array} \end{array} \right] \cdot \\
&\quad \cdot \left(\sum_{N_0} T_{\text{cpt}(N_1)} \cdot \left[\begin{array}{c|c} n_{0,1} & \begin{array}{l} (0.30 S_{0,1} W_{0,1} + 0.30 S_{0,1} W_{0,2} + \\ 0.18 S_{0,2} W_{0,1} + 0.12 S_{0,2} W_{0,2} + \\ 0.08 S_{0,3} W_{0,1} + 0.02 S_{0,3} W_{0,2}) \end{array} \\ \hline n_{0,2} & \begin{array}{l} (0.18 S_{0,1} W_{0,1} + 0.42 S_{0,1} W_{0,2} + \\ 0.12 S_{0,2} W_{0,1} + 0.18 S_{0,2} W_{0,2} + \\ 0.05 S_{0,3} W_{0,1} + 0.05 S_{0,3} W_{0,2}) \end{array} \end{array} \right] \cdot \left[\begin{array}{c|c} n_{0,1} & 0.80 \\ \hline n_{0,2} & 0.20 \end{array} \right] \right)
\end{aligned}$$

Bsp

3.2.6 Approximativer Rollup

Wie man bei den vorigen Rollup-Verfahren gesehen hat, kann die Repräsentation eines Zustandsraumes (der BEL-Wert über allen relevanten Knoten der Zeitscheibe) aus Speicherplatzproblemen und/oder Rechenzeitbeschränkungen unmöglich werden, da er ohne Informationsverlust nicht faktorisiert dargestellt werden kann. Beim approximativen Rollup wird dieser Informationsverlust hingenommen und der genaue BEL-Wert durch einen approximativen BEL-Wert angenähert, indem der genaue BEL-Wert faktorisiert wird und als approximativer BEL-Wert kompakter gespeichert werden kann. Genauer gesagt, der approximative BEL-Wert wird direkt ausgerechnet, ohne daß der genaue BEL-Wert benötigt wird. Bei einer geeigneten Wahl der Approximation des BEL-Wertes akkumuliert sich der Fehler über die Zeit nicht.

Im folgenden wird nun der approximative Rollup vorgestellt, wie er in [Boyen & Koller 98a] oder [Boyen & Koller 99] beschrieben wird. Die Schreibweise, die in den beiden Artikeln benutzt wird, wurde dabei an die Schreibweise dieser Arbeit angepaßt.

Die Menge der Dynamischen Knoten der Zeitscheibe t ist die Menge $\mathbf{D}_t := \{V: V \in \text{pa}(\mathbf{V}(t+1))\}$ (In [Boyen & Koller 98a] wird die Menge der Zufallsvariablen, die mit den Dynamischen Knoten assoziiert sind, als die *kanonische Menge der Zufallsvariablen* bezeichnet.). Die Menge der Dynamischen Knoten ist also die Menge der Knoten, die Elternknoten eines Knotens der darauffolgenden Zeitscheibe sind. Ein Dynamisches Bayessches Netz, das aus zwei Zeitscheiben besteht, ist in *kanonischer Form*, wenn zum Zeitpunkt $t+1$ in der Zeitscheibe t nur noch die Menge der Dynamischen Knoten

vorhanden sind. Die Menge der Dynamischen Knoten wird in disjunkte Teilmengen $\mathbf{K}_{1,t}, \dots, \mathbf{K}_{L,t}$ partitioniert, d.h. $\dot{\cup}_{\ell} \mathbf{K}_{\ell,t} = \mathbf{D}_t$. Zusätzlich muß die Partition die folgende Forderung erfüllen: Keine disjunkte Teilmenge $\mathbf{K}_{\ell,t}$ ($1 \leq \ell \leq L$) darf durch eine andere Teilmenge $\mathbf{K}_{\ell',t}$ ($1 \leq \ell' \leq L, \ell' \neq \ell$) innerhalb derselben Zeitscheibe t beeinflusst werden, d.h., wenn $U \in \mathbf{K}_{\ell,t}$ in der Zeitscheibe t , dann kann U keinen Vorgängerknoten V in der Zeitscheibe t besitzen mit $V \in \mathbf{K}_{\ell',t}, \mathbf{K}_{\ell,t} \neq \mathbf{K}_{\ell',t}$.

Als erstes wird nun ein Junction Tree konstruiert, in dem für jedes $\ell, 1 \leq \ell \leq L$, einige Cliques des Junction Trees die Partitionen von \mathbf{D}_t und einige Cliques des Junction Trees die Partitionen von \mathbf{D}_{t+1} enthalten. Es wird die Evidenz \mathbf{E}_{t+1} eingetragen und dann eine Propagierung auf dem Junction Tree durchgeführt. Aus den entsprechenden Cliques können nun die Werte für die Zufallsvariablen der Zeitscheibe $t+1$, die mit den Partitionen assoziiert sind, herausgezogen werden.

Weitere Einsparungen können erzielt werden, wenn immer dasselbe Schema zur Instantiierung einer Zeitscheibe herangezogen und immer diesselbe Approximation vorgenommen wird. Dann ist es möglich, die Struktur des Junction Trees vorzuberechnen, und die Knoten des Dynamischen Bayesschen Netzes den entsprechenden Cliques des Junction Trees zuzuordnen. Dieses resultiert in einer Vorlage eines Junction Trees, die als Ausgangspunkt für jede Propagierung herangezogen werden kann.

Das Vorgehen beim approximativen Rollup kann man nun wie folgt zusammenfassen:

- Eingabe:
 1. zwei Zeitscheiben eines Dynamischen Bayesschen Netzes in kanonischer Form
 2. eine Partition der Dynamischen Knoten $\mathbf{D} := \cup_{\ell=1}^L \mathbf{K}_{\ell}$
 3. ein approximativer BEL-Wert $\widetilde{\text{BEL}}(\mathbf{D}_0)$ als Initialisierung
 4. eine Reihe von Evidenzen $\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3, \dots$

- Ausgabe: Eine Reihe approximativer BEL-Werte

$$\widetilde{\text{BEL}}(\mathbf{D}_1), \widetilde{\text{BEL}}(\mathbf{D}_2), \widetilde{\text{BEL}}(\mathbf{D}_3), \dots$$

- Vorgehen:
 1. Konstruktion eines Junction Trees für das gegebene Dynamische Bayessche Netz aus zwei Zeitscheiben. Jedes $\mathbf{K}_{\ell,t}$ und $\mathbf{K}_{\ell,t+1}$ muß wenigstens in einer Clique des Junction Trees vollständig enthalten sein.
 2. Initialisiere jeden Cliques-Faktor mit der konstanten Funktion = 1.
 3. Ordne jeden Knoten des Dynamischen Bayesschen Netzes einer Clique des Junction Trees zu. Sei \mathcal{T}_0 die daraus resultierende (unkalibrierte) "Vorlage eines Junction Trees".
 4. Für $t = 0, 1, 2, \dots$:
 - (a) Lege eine Kopie \mathcal{T} des Junction Trees \mathcal{T}_0 an, auf der gearbeitet wird. Erzeuge $\widetilde{\text{BEL}}(\mathbf{D}_{t+1})$ als neuen approximativen BEL-Wert, der noch nicht belegt ist.

- (b) Trage jeden Faktor $\text{BEL}(\mathbf{K}_{\ell,t})$ von $\widetilde{\text{BEL}}(\mathbf{D}_t)$ in die entsprechende Clique des Junction Trees \mathcal{T} ein.
- (c) Trage die Evidenz \mathbf{E}_{t+1} im Junction Tree \mathcal{T} ein.
- (d) Kalibriere die Cliquen des Junction Trees \mathcal{T} zueinander, d.h. führe auf dem Junction Tree \mathcal{T} eine Propagierung durch.
- (e) Für jedes ℓ berechne $\text{BEL}(\mathbf{K}_{\ell,t+1})$ auf dem Junction Tree \mathcal{T} und speichere es im approximativen BEL-Wert $\widetilde{\text{BEL}}(\mathbf{D}_{t+1})$.
- (f) Lösche den Junction Tree \mathcal{T} und gebe den approximativen BEL-Wert $\widetilde{\text{BEL}}(\mathbf{D}_{t+1})$ aus.

Um dieses Verfahren auf einem bestimmten Dynamischen Bayesschen Netz anwenden zu können, muß eine Partition für die Dynamischen Knoten bestimmt werden. Dabei ist zu beachten, daß kleine Partitionen mit einer geringen Anzahl von Knoten eine schnellere Propagierung ermöglichen. Jedoch müssen die Partitionen groß genug sein, damit keine Kanten zwischen Partitionen innerhalb einer Zeitscheibe vorhanden sind. Weiterhin muß beachtet werden, wenn die Partitionen klein gewählt werden, daß der Fehler größer ist durch die Annahme, daß die Partitionen unabhängig voneinander sind. Hat man insbesondere zwei (Mengen von) Knoten, die sich untereinander stark beeinflussen, so ist es sehr wahrscheinlich eine schlechte Idee diese beiden in zwei verschiedene Partitionen aufzuteilen.

In [Boyen & Koller 98a] wird informal argumentiert, daß der Fehler durch die Approximation des BEL-Wertes der Dynamischen Knoten gering ist, wenn die Interaktion zwischen den Partitionen der Variablen gering ist. Dieses wird in [Boyen & Koller 99] genauer ausgeführt. Hier werden informationstheoretische Definitionen für die Begriffe *schwache Interaktion* und *spärliche Interaktion* eingeführt, um die Bedingungen zu analysieren, unter denen der Fehler durch die Approximation des BEL-Wertes der Dynamischen Knoten klein ist. Wie in [Boyen & Koller 98b] bewiesen wird, schaukelt sich der Fehler über die Zeit nicht auf, d.h. er ist beschränkt. Allerdings ist die Schranke für den Fehler nicht bekannt.

Bei Dynamischen Bayesschen Netzen, bei denen zu jedem Zeitpunkt schon im Voraus bekannt ist, welches Schema zu welchem Zeitpunkt als Zeitscheibe instantiiert wird, wird ein kleiner Fehler, der zudem noch beschränkt ist, keine großen Auswirkungen auf das Dynamische Bayessche Netz haben. Hingegen sieht dies bei Dynamischen Bayesschen Netzen anders aus, bei denen erst bei der Instantiierung einer Zeitscheibe entschieden wird, welches Schema zu verwenden ist. Zum Beispiel wird aufgrund der BEL-Werte bestimmter Knoten(-mengen) bei READY-Netzen in der Dialogsteuerung entschieden, welches Schema bei der Instantiierung einer Zeitscheibe zu tragen kommt (siehe [Schäfer 98; Weis 00]). Je nachdem wie sensibel die Dialogsteuerung ist, kann schon ein kleiner Fehler die Dialogentwicklung in eine andere Richtung lenken. Die Dialogentwicklung kann aber auch in die andere Richtung gehen, wenn schon im exakten Fall eine knappe Entscheidung getroffen wird, die im approximativen Fall umschwenkt. Hier wird bewußt von “andere Richtung” und nicht von “falscher Richtung” geschrieben. Man muß sich eben klar machen, daß man durch eine Approximation nicht mehr dieselben Ergebnisse erhält. Frei nach dem Motto: “*It’s not a bug, it’s a feature.*”

3.3 Vergleich der Algorithmen

Im folgenden werden die wichtigsten Eigenschaften der Rollup-Verfahren kurz zusammengefaßt. In einer Tabelle sind dazu die Vor- und Nachteile der verschiedenen Verfahren und ihre Anwendbarkeit aufgelistet.

Die drei Verfahren Knotenabsorption, Prediction-Estimation und dHUGIN benötigen beim Rollup den aktuellen BEL-Wert der Zeitscheibe, die aufgerollt werden soll. In diesem BEL-Wert ist das gesamte Wissen enthalten, das für die folgende Zeitscheibe notwendig ist, um eine genaue Inferenz durchführen zu können. Dieser BEL-Wert kann leider nicht faktorisiert dargestellt werden. Dies kann bei sehr komplexen Dynamischen Bayesschen Netzen zu Problemen bei der Berechnung und Speicherung des BEL-Wertes führen.

Bei der Polynom-Propagierung und den Shafer-Shenoy Skripten wird der BEL-Wert des gesamten Netzes nicht explizit benötigt. Für einen Knoten ist nur ein Teil des BEL-Wertes der Zeitscheibe interessant und notwendig.

Beim approximativen Rollup wird der BEL-Wert einer Zeitscheibe faktorisiert. Dadurch entsteht ein Fehler, der aber bei einer geeigneten Faktorisierung eine unbekannte Fehlerschranke nicht überschreitet.

In Tabelle 3.4 sind die Vor- und Nachteile der verschiedenen Verfahren und ihre Anwendbarkeit aufgelistet.

<i>Verfahren</i>	<i>Vorteile</i>	<i>Nachteile</i>	<i>Anwendbarkeit</i>
Knotenabsorption	Exaktes Verfahren Erzeugt im allgemeinen nichtkomplexe Netze bzgl. Inferenzverfahren, wenn nur wenige Knoten des ursprünglichen Netzes erhalten bleiben	Mögliche Speicherplatzprobleme während der Berechnung, die zum Abbruch führen können (auch wenn kleine Netze erhalten bleiben)	Für Dynamische Bayesschen Netze, die nicht komplex bzgl. Inferenzverfahren sind
Prediction-Estimation	Exaktes Verfahren Gut für das Verständnis der Rollup-Verfahren im allgemeinen	Keine faktorisierte Darstellung des BEL-Wertes einer Zeitscheibe möglich	Für Dynamische Bayesschen Netze, die nicht komplex bzgl. Inferenzverfahren sind
dHugin	Exaktes Verfahren Rückwirkende Einschätzung von Zeitscheiben möglich	Die Interface-Cliquen der Zeitscheiben des Zeitfensters und der Backward smoothing models können bzgl. Speicherplatzbedarf und Rechenzeitverbrauch unhabbar groß werden	Für Dynamische Bayesschen Netze, die nicht komplex bzgl. Inferenzverfahren sind
Polynom-Propagierung	Exaktes Verfahren Keine eingeschränkte Eliminationsreihenfolge notwendig Nicht Zeitscheiben- sondern Knotenorientiert Eine Zeitscheibe t_{neu} kann vor einer Zeitscheibe t_{alt} ($t_{\text{alt}} < t_{\text{neu}}$) aufgerollt werden	Komplexität der Polynome ist abhängig von der Anzahl der Evidenzvariablen und der Anzahl ihrer Hypothesen Schwierige Faktorisierung der Polynome	Nur wenn wenige Evidenzvariablen im Dynamischen Bayesschen Netz vorhanden sind Für Geräte mit geringer Rechenleistung geeignet, auf denen sich kein effizienter Inferenzalgorithmus implementieren läßt
Shafer-Shenoy Skripte	Keine Speicherplatzprobleme, da die ψ -Tabellen nicht explizit ausgerechnet werden Möglichkeit der Vereinfachung bzw. Umformung eines Shafer-Shenoy Skriptes	Die Vereinfachung bzw. Umformung eines Shafer-Shenoy Skriptes ist aufwendig und erfordert ein Mathematik-Programm	Zusammen mit Polynom-Propagierung zur Faktorisierung der Polynome
approximativer Rollup	Schnellere Inferenz Geringerer Speicherverbrauch Fehlertolerant	BEL-Werte der Knoten sind mit einem Fehler behaftet Faktorisierung nicht beliebig erlaubt	Dynamische Bayessche Netze, die eine Faktorisierung erlauben und keine exakten BEL-Werte benötigen

Tabelle 3.4: Vergleich der Rollup-Verfahren

Kapitel 4

READY-Netze

Im Projekt READY¹ (siehe [Wahlster et al. 95; Jameson et al. 99]) wird ein Dialogsystem entwickelt, das seine Dialogbeiträge an die Ressourcen des Benutzers (wie z.B. dessen verfügbare Arbeitsgedächtniskapazität) adaptiert. Für die Modellierung werden Dynamische Bayessche Netze mit besonderen Modellierungseigenschaften verwendet, die als READY-Netze bezeichnet werden.

Im folgenden wird überprüft, wie sich die Rollup-Verfahren aus dem Kapitel 3 auf die READY-Netze anwenden lassen. Dazu werden die Besonderheiten eingeführt und erläutert, die die READY-Netze gegenüber Standard Dynamischen Bayesschen Netzen besitzen. Wie man sehen wird, verhindern diese Besonderheiten die direkte Anwendung der Rollup-Verfahren aus Kapitel 3. Es wird gezeigt, wie die READY-Netze in Standard Dynamische Bayessche Netze umgewandelt werden können, und somit wieder eine direkte Anwendung der Rollup-Verfahren möglich ist.

In READY-Netzen werden gegenüber Standard Dynamischen Bayesschen Netzen zusätzlich zu den dynamischen Knoten in Dynamischen Bayesschen Netzen weitere Knoten unterschieden und eingeführt: Wie in Dynamischen Bayesschen Netzen modellieren *Dynamische Knoten* in READY-Netzen Sachverhalte, die sich über die Zeit entwickeln. Unveränderliche Sachverhalte werden durch *statische Knoten* repräsentiert. Statische Knoten beeinflussen mehrere Zeitscheiben, ohne daß sie einer gewissen Zeitscheibe zugeordnet sind. Knoten, die Sachverhalten entsprechen, die nur innerhalb einer Zeitscheibe existieren, werden *temporäre Knoten* genannt.

Zusätzlich werden für die Modellierung von READY-Netzen (siehe [Schäfer & Weyrath 97]) einige Besonderheiten² eingeführt, um gewisse Sachzusammenhänge für den Anwender in der Darstellung und für den Entwickler in der Modellierung von Bayesschen Netze zu vereinfachen. Dabei werden bestimmte Schemata zu einem *Doppelschema* zusammengefaßt, wie zum Beispiel die Schemata **Frage** und **Antwort** zum Doppelschema **Frage-Antwort** zusammengefaßt werden, so daß direkt das Schema **Frage-Antwort** instantiiert werden kann, da man nach einer Frage eine Antwort³ erwarten kann. Dazu müssen dann nicht nacheinander zuerst das Schema **Frage** und dann das

¹READY ist das Akronym für Ressourcenadvaptives Dialogsystem

²Auf diese Besonderheiten wird in der bisherigen Literatur noch nicht hingewiesen.

³Keine Antwort ist auch eine.

Schema **Antwort** instantiiert werden. Auf diese Weise werden Sinneinheiten gebildet und das Dynamische Bayessche Netz weiter gegliedert. Eine weitere Sinneinheit ist zum Beispiel das Doppelschema **Instruktion-Handlung**, da nach einer Anweisung eine Ausführung⁴ erfolgt.

Bei der Instantiierung von Zeitscheiben werden in READY nicht immer alle Teile eines Schemas erzeugt, sondern so lange mit der Instantiierung gewartet, bis sie wirklich benötigt werden. Dann kommt es zu einer *Nachinstantiierung von Netzstrukturen*. Findet eine Nachinstantiierung nicht statt, so bleibt das zu lösende Netz kleiner. Die folgenden Besonderheiten der READY-Netze wie

- Doppelschemata und im allgemeinen zeitscheibenüberspringende Kanten,
- Statische Knoten und
- die Nachinstantiierung von Netzstrukturen

sind problematisch beim Aufrollen der Netze, wie es sich in den nachfolgenden Betrachtungen zeigen wird. In den folgenden Abschnitten werden die verschiedenen “Erleichterungen” anhand eines generischen Schemas und eines Beispiels vorgestellt, die aus den Erleichterungen resultierenden Schwierigkeiten für den Rollup diskutiert und abschließend gezeigt, welche Umformungen getätigt werden müssen, damit wieder ein Rollup mit den Mitteln aus Kapitel 3 möglich ist.

Im Kapitel 3 haben wir gesehen, daß es ausreicht immer nur maximal zwei Zeitscheiben des DBNs in Bearbeitung zu haben. Die READY-Netze werden jetzt auf dieses Vorgehen hin geprüft.

4.1 Problembehandlung

In den folgenden Abschnitten wird jede der oben aufgeführten Besonderheiten der READY-Netze isoliert für sich betrachtet und bzgl. des Rollups eine Lösung vorgestellt. Dazu wird mittels generischer Schemata die jeweilige Besonderheit der READY-Netze eingeführt und erklärt. Auf diesen generischen Schemata wird gezeigt, welche Probleme auftreten würden, wenn ein unangepaßtes READY-Netze aufgerollt werden würde. Dann wird das unaufgerollte generische Schema so abgewandelt, daß die Besonderheit des READY-Netzes verschwindet, und wieder ein Rollup mit den Mitteln aus Kapitel 3 möglich ist.

Ein anschließendes Beispiel mit einem Dynamischen Bayesschen Netz, bei dem die behandelte Besonderheit herausgearbeitet wurde, veranschaulicht dabei die Problemstellung und ihre Lösung weiter.

In diesem Abschnitt wird das Problem des Rollups an einem Beispiel für ein Dynamisches Bayessches Netz gelöst, das die gleiche Struktur wie die READY-Netze hat aber wesentlich kleiner ist. Somit wird das Augenmerk ganz auf die jeweilige Problemstellung gelenkt.

Die Besonderheiten der READY-Netze wie zum Beispiel *Statische Knoten*, *zeitscheibenüberspringende Kanten* oder *Doppelzeitscheiben*, die den Rollup-Verfahren aus

⁴Keine Ausführung ist ... (dito)

Kapitel 3 Schwierigkeiten bereiten, werden im folgenden in Situationen aufsteigender Komplexität bezüglich dieser Besonderheiten vorgestellt. Ziel ist es, Umformungen vorzunehmen, die die verschiedenen Besonderheiten in ein Dynamisches Bayessches Netz übersetzen, das den Spezifikationen aus Kapitel 2 entspricht.

Dazu wird zuerst das entsprechende Problem umfassend vorgestellt, und dann am angepaßten Beispielnetz verdeutlicht.

Im folgenden Abschnitt wird nun das Problem von *zeitscheibenüberspringenden Kanten* angegangen.

4.1.1 Zeitscheibenüberspringende Kanten

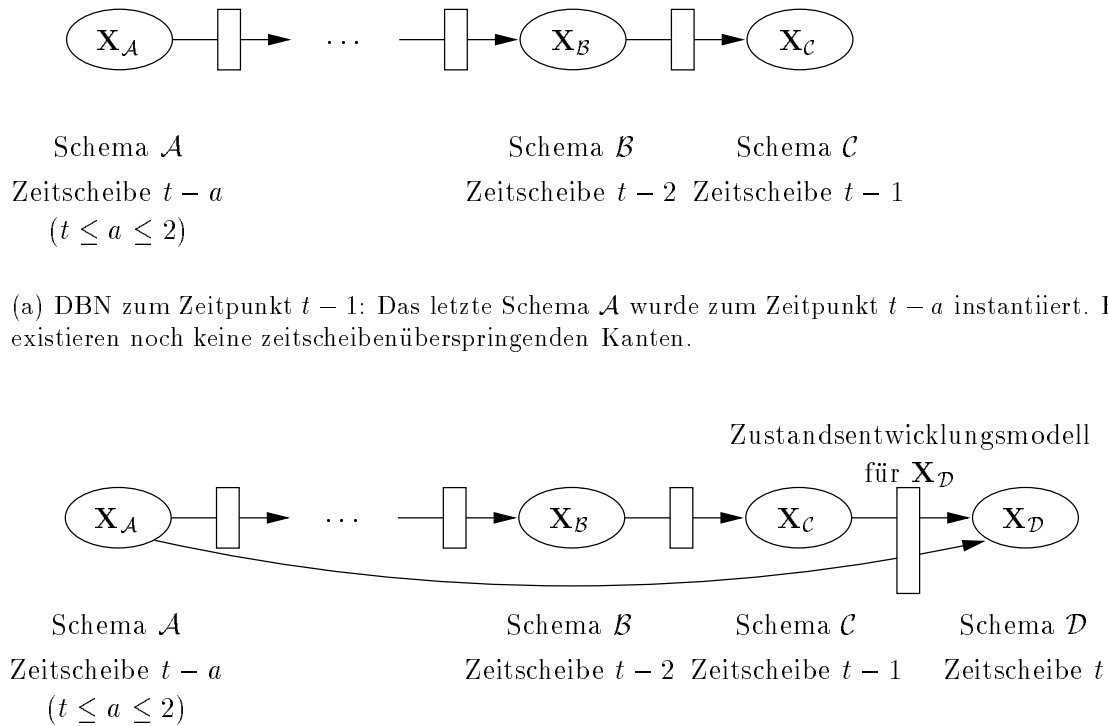
Allgemeines Problem

Im folgenden wird beschrieben, wie zeitscheibenüberspringende Kanten entstehen können und warum sie ein Problem für Dynamische Bayessche Netze und insbesondere für die Rollup-Verfahren aus Kapitel 3 darstellen. Man wird sehen, daß sich die Rollup-Verfahren aus Kapitel 3 nicht auf das Dynamische Bayessche Netz anwenden lassen, wenn zeitscheibenüberspringende Kanten vorkommen. Desweiteren wird gezeigt, wie sich das Problem der zeitscheibenüberspringenden Kanten durch das Einfügen von bestimmten Knoten an entsprechenden Stellen der zeitscheibenüberspringenden Kanten lösen läßt und die Rollup-Verfahren dadurch wieder anwendbar sind.

Es sei ein Dynamisches Bayessches Netz \mathcal{DBN} zum Zeitpunkt $t - a - 1$ gegeben, das nicht aufgerollt wird. Als Zeitscheibe $t - a$ werde jetzt das Schema \mathcal{A} instantiiert. Bis zum Zeitpunkt $t - 1$ werden dann sukzessive die einzelnen Zeitscheiben erzeugt. Für die Zeitscheiben kommen dabei verschiedene Schemata zum Tragen. Zu jedem Zeitpunkt h ($1 \leq h \leq t - 1$) seien die Schemata des Dynamischen Bayesschen Netzes so instantiiert, daß das jeweilige Zustandsentwicklungsmodell der Zeitscheibe h nur von der vorhergehenden Zeitscheibe $h - 1$ bestimmt ist. Dieses wird im generischen Schema durch die gerichteten Kanten, die durch einen Kasten gehen, angedeutet. Die Zeitscheiben erfüllen somit die Markov-Eigenschaft für Dynamische Bayessche Netze.

In Abbildung 4.1(a) ist ein Schnappschuß des resultierenden Dynamischen Bayesschen Netzes zum Zeitpunkt $t - 1$ zu sehen. Es werden die Zeitscheiben $t - a$ bis $t - 1$ mit den dazwischenliegenden Zustandsentwicklungsmodellen schematisch dargestellt. Die Zeitscheiben bis vor dem Zeitpunkt $t - a$ sind ausgeblendet, da sie nichts zum Sachverhalt beitragen. Zum Zeitpunkt $t - a$ sei das Schema \mathcal{A} zum letzten Mal instantiiert worden (Dies ist wichtig für die Instantiierung der Zeitscheibe t). Danach seien bis zum Zeitpunkt $t - 1$ nur noch Schemata $\mathcal{B} \neq \mathcal{A}$ ($\mathcal{B} \in \{\mathcal{S} \mid \mathcal{S} \text{ ist ein Schema}\}$) als Zeitscheiben eingefügt worden. Exemplarisch sind die Schemata \mathcal{B} und \mathcal{C} angegeben, wobei die Schemata \mathcal{B} und \mathcal{C} auch identisch sein können. Zum Zeitpunkt $t - 1$ besteht das Dynamische Bayessche Netz somit aus t Zeitscheiben.

Zum Zeitpunkt t wird nun ein Schema als Zeitscheibe t instantiiert, das ab dem Zeitpunkt $t - a$ nicht mehr instantiiert wurde. Dieses Schema wird in Abbildung 4.1(b) mit \mathcal{D} bezeichnet. Im Schema \mathcal{D} ist ein Knoten enthalten, der eine Zufallsvariable repräsentiert, die auch oder nur von einer Zufallsvariablen beeinflusst wird, deren entsprechender Knoten im Schema \mathcal{A} enthalten ist. Die Schemata \mathcal{A} und \mathcal{D} können



(a) DBN zum Zeitpunkt $t-1$: Das letzte Schema \mathcal{A} wurde zum Zeitpunkt $t-a$ instantiiert. Es existieren noch keine zeitscheibenüberspringenden Kanten.

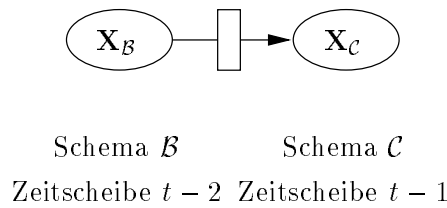
(b) DBN zum Zeitpunkt t : Das Schema \mathcal{D} wird instantiiert. Dadurch werden Kanten von Zeitscheibe $t-a$ zu Zeitscheibe t induziert.

Abbildung 4.1: Zeitscheibenüberspringende Kanten: Im Vorgängerschema sind nicht alle Zufallsvariablen enthalten, von denen das neu instantiierte Schema abhängig ist.

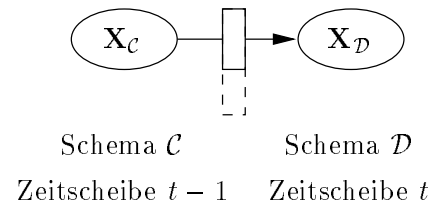
dabei identisch sein. Das Zustandsentwicklungsmodell für $\mathbf{X}_{\mathcal{D}}$ zum Zeitpunkt t ist nicht mehr nur von der Zeitscheibe $t-1$ abhängig, sondern auch von der Zeitscheibe $t-a$, in der das Schema \mathcal{A} als Zeitscheibe instantiiert wurde. Es wird eine *zeitscheibenüberspringende Kante* von der Zeitscheibe $t-a$ zur Zeitscheibe t induziert. Dies ist in der Abbildung 4.1(b) durch einen gerichteten Pfeil von Zeitscheibe $t-a$ zu Zeitscheibe t dargestellt. Die Zustandsentwicklungsmodelle und die Schemata dazwischen ändern sich dabei nicht.

Werden keine Zeitscheiben aufgerollt, so muß zwar der Junction Tree des Dynamischen Bayesschen Netzes vollständig neu berechnet werden (Dieses erkennt man z.B. an den Triangulationskanten in den Abbildungen 4.8(a) und (b)). Wie man erkennen kann, werden Knoten aus der Zeitscheibe t durch Kanten mit Knoten aus den Zeitscheiben $t-3$ und $t-2$ verbunden. Dadurch ändern sich z.B. die Cliques in den entsprechenden Teilbäumen.), es ergeben sich aber keine weiteren Probleme (vorausgesetzt, daß sich das Dynamische Bayessche Netz noch auswerten läßt bzgl. Speicherplatzverbrauch und Rechenzeitbedarf.).

Probleme treten allerdings auf, wenn ein Rollup immer dann ausgeführt wird, wenn eine neue Zeitscheibe etabliert wurde (siehe Kapitel 3). In Abbildung 4.2(a) ist das Dynamische Bayessche Netz zum Zeitpunkt $t-1$ zu sehen, bevor die Zeitscheibe $t-2$ aufgerollt wird. Bis zum Zeitpunkt $t-2$ wurden die Zeitscheiben h ($0 \leq h < t-2$) jeweils dann aufgerollt, wenn die neue Zeitscheibe $h+1$ etabliert war. Zu jedem



(a) DBN mit Rollup zum Zeitpunkt $t-1$: Nur das Schema \mathcal{B} und das Schema \mathcal{C} sind als Zeitscheibe $t-2$ bzw. Zeitscheibe $t-1$ vorhanden.



(b) DBN mit Rollup zum Zeitpunkt t : Das Schema \mathcal{D} wird instantiiert. Es können keine zeitscheibenüberspringende Kanten induziert werden. Das Zustandsentwicklungsmodell für \mathbf{X}_D fällt kleiner aus als in Abbildung 4.1(b).

Abbildung 4.2: Zeitscheibenüberspringende Kanten mit Rollup ohne Korrektur: Es werden nicht mehr alle Abhängigkeiten berücksichtigt.

Zeitpunkt h ($0 \leq h \leq t-2$) waren somit immer maximal zwei Zeitscheiben des Dynamischen Bayesschen Netzes präsent.

Wird nun zum Zeitpunkt t das Schema \mathcal{D} als Zeitscheibe t instantiiert, so läßt sich die Abhängigkeit von Zeitscheibe $t-a$, in der das Schema \mathcal{A} instantiiert war, nicht mehr modellieren. In Abbildung 4.2(a) und (b) sind die aufgerollten Zeitscheiben gestrichelt eingezeichnet.

In Abbildung 4.2(b) erkennt man zusätzlich, daß die zeitscheibenüberspringende Kante nicht etabliert werden konnte. Insbesondere wird die Abhängigkeit der Zeitscheibe t von der Zeitscheibe $t-a$ nicht mehr modelliert.

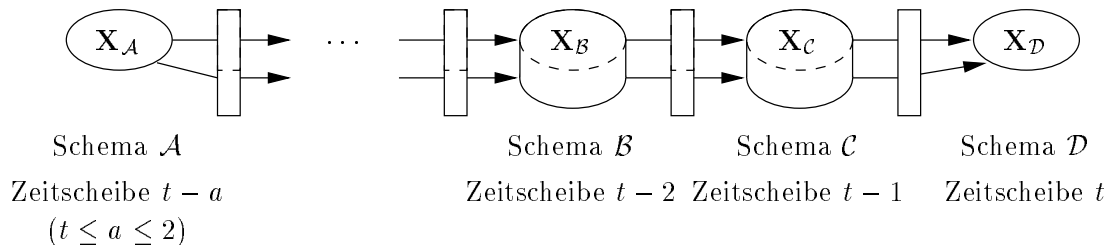


Abbildung 4.3: Zeitscheibenüberspringende Kanten mit Korrektur: Die verschiedenen Modelle werden um Zwischenknoten erweitert, so daß eine zeitscheibenüberspringende Kante in den Zeitscheiben zu liegen kommt.

Damit ein Rollup mit den Mitteln aus Kapitel 3 möglich ist, muß also eine mögliche

Beeinflussung einer Zeitscheibe auf eine nachfolgende Zeitscheibe durch alle Zeitscheiben, die zwischen den beiden Zeitscheiben liegen, durchgereicht werden. In Abbildung 4.3 ist dargestellt, daß sich die Schemata für die Zeitscheiben und die Zustandsentwicklungsmodelle zwischen den Zeitscheiben erweitern. Man sieht, daß temporale Kanten hinzukommen. Es existiere zum Beispiel vom Knoten Z_{t-a} aus der Zeitscheibe $t-a$ eine zeitscheibenüberspringende Kante zum Knoten Z_t aus der Zeitscheibe t . Damit die Beeinflussung durch die dazwischenliegenden Zeitscheiben durchgereicht werden kann, wird für den Knoten Z_{t-a} in jeder Zeitscheibe h ($t-a < h \leq t-1$) ein Knoten Z_h instantiiert. Die bedingten Wahrscheinlichkeiten der Knoten werden wie folgt angepaßt, wenn i die Anzahl der Hypothesen von Z_{t-a} bezeichnet:

- Für die Knoten Z_h mit $h \in \{(t-a) + 1, \dots, t-1\}$:

$$T_{\text{cpt}(Z_h)} = T(Z_h, Z_{h-1}) = \left[\begin{array}{c|ccc} & z_{h,1} & \cdots & z_{h,i} \\ \hline z_{h-1,1} & 1 & & 0 \\ \vdots & & \ddots & \\ z_{h-1,i} & 0 & & 1 \end{array} \right].$$

Den Knoten Z_h wird somit eine 2-dimensionale Tabelle über den beiden Knoten Z_{h-1} und Z_h der Größe $i \times i$ zugeordnet, die in der Diagonale von $(z_{h-1,1}, z_{h,1})$ bis $(z_{h-1,i}, z_{h,i})$ nur Einsen und ansonsten nur Nullen als Eintrag enthält. Man kann $T_{\text{cpt}(Z_h)}$ auch als Identitätstabelle bedingter Wahrscheinlichkeiten bezeichnen, da durch sie der Knoten Z_h denselben BEL-Wert wie sein Vorgängerknoten Z_{h-1} erhält.

- Für den Knoten Z_t :

Anstatt der Tabelle $T_{\text{cpt}(Z_t)} = T(Z_t, Z_{t-a}, E_1, \dots, E_p)$ erhält der Knoten Z_t nun die Tabelle $T_{\text{cpt}(Z_t)} = T(Z_t, Z_{t-1}, E_1, \dots, E_p)$, wobei mit E_1, \dots, E_p bis auf Z_{t-a} oder Z_{t-1} die Elternknoten von Z_t gemeint sind. Die bedingten Wahrscheinlichkeiten bleiben unverändert. In der Tabelle der bedingten Wahrscheinlichkeiten des Knoten Z_t , wird der Elternknoten Z_{t-a} durch den Knoten Z_{t-1} ersetzt.

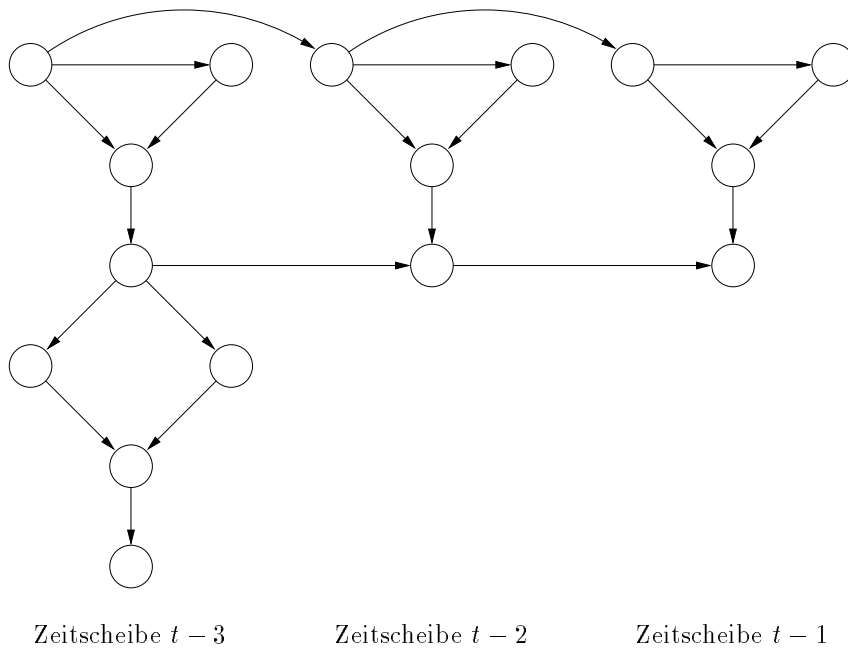
Dadurch daß die bedingten Wahrscheinlichkeiten wie vor definiert sind, wird die Beeinflussung des Knotens Z_{t-a} an die Zeitscheibe t unverändert weitergegeben. Jetzt kann der Rollup wie gewohnt mit den Mitteln aus dem Kapitel 3 durchgeführt werden.

Im folgenden Beispiel wird nun an einem konkreten Beispielnetz gezeigt, wie zeitscheibenüberspringende Kanten entstehen, die sich daraus entwickelnde Problemstellung erläutert und im Anschluß daran die Problemlösung vorgeführt. Als Beispielnetz wird das Dynamische Bayessche Netz aus Kapitel 2 wiederaufgegriffen und die entsprechende Problemstellung herausgearbeitet.

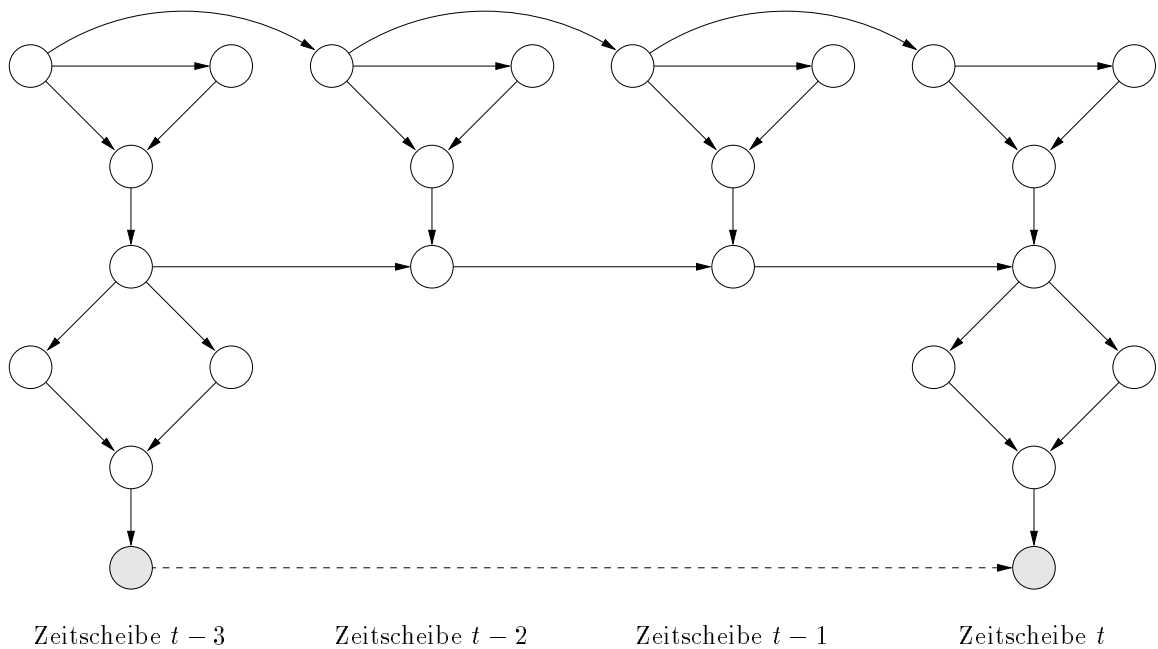
Beispiel 4.1.1 (Zeitscheibenüberspringende Kanten (allgemeiner Fall))

Wie den obigen Ausführungen zu entnehmen ist, sind zeitscheibenüberspringende Kanten vor allem ein graphentheoretisches Problem. Dazu sei als Beispiel das Dynamische Bayessche Netz in Abbildung 4.4(a) gegeben, von dem die Zeitscheiben von $t-3$ bis $t-1$ dargestellt sind.

Zu diesem Zeitpunkt erfüllt das Dynamische Bayessche Netz noch alle Anforderungen, die an ein Dynamisches Bayessches Netz gestellt werden. In Abbildung 4.8(a) wird



(a) DBN zum Zeitpunkt $t - 1$: Als letztes wurde die Zeitscheibe t instantiiert. Es existieren noch keine zeitscheibenüberspringenden Kanten.



(b) DBN zum Zeitpunkt t : Die Zeitscheibe t induziert eine zeitscheibenüberspringenden Kanten von Zeitscheibe $t - 3$ zu Zeitscheibe t .

Abbildung 4.4: Beispiel: Zeitscheibenüberspringende Kanten ohne Korrektur

gezeigt, wie sich das Dynamische Bayessche Netz triangulieren läßt, so daß ein Junction Tree entsteht, dessen Struktur den Aufbau des Dynamischen Bayesschen Netzes aus Zeitscheiben widerspiegelt. Auf die Triangulierung wird weiter hinten im Beispiel noch näher darauf eingegangen.

Jetzt wird ein Schema als Zeitscheibe t instantiiert, das eine zeitscheibenüberspringende Kante von Zeitscheibe $t - 3$ zur Zeitscheibe t induziert (siehe Abbildung 4.4(b)). Die zeitscheibenüberspringende Kante ist gestrichelt eingezeichnet und die beteiligten Knoten sind grau schraffiert dargestellt. In Abbildung 4.8(b) ist die Triangulierung bei einer Eliminationsreihenfolge, die die Zeitscheiben berücksichtigt, dargestellt. Man erkennt durch die zeitscheibenüberspringenden Triangulationskanten, daß der aus dieser Triangulierung resultierende Junction Tree nicht den Aufbau des Dynamischen Bayesschen Netzes aus Zeitscheiben widerspiegelt. Weiter soll auf die Triangulierung im Moment nicht eingegangen werden. Dazu im Beispiel später noch mehr und tiefergehend.

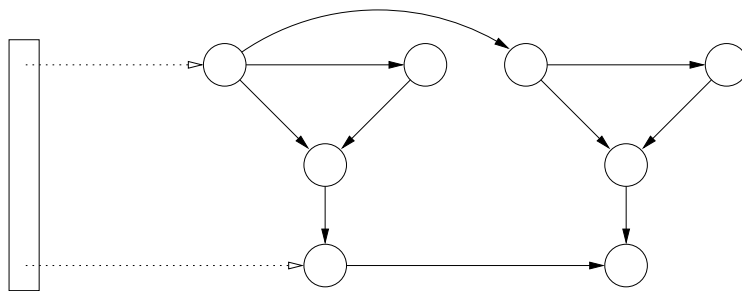
Führt man den Rollup mit den Mitteln aus dem Kapitel 3 regelmäßig durch, so daß maximal immer nur zwei Zeitscheiben des Dynamischen Bayesschen Netzes repräsentiert sind, so wird eine zeitscheibenüberspringende Kante nie induziert werden. Im Beispiel ist dies die zeitscheibenüberspringende Kante von der Zeitscheibe $t - 3$ zu der Zeitscheibe t , die beim Rollup nicht induziert wird, wie in den beiden Abbildungen 4.5(a) und 4.5(b) zu sehen ist. Die Zustandsentwicklungsmodelle sind in den Abbildungen 4.5(a) und (b) als Rechtecke eingezeichnet. Man erkennt, daß sie die zeitscheibenüberspringende Kante nicht umfassen.

Bei der Problemlösung muß nun versucht werden, daß die Zustandsentwicklungsmodelle die zeitscheibenüberspringenden Kanten umfassen, oder anders ausgedrückt, daß sich das Dynamische Bayessche Netz wieder eindeutig in Zeitscheiben einteilen läßt (und dabei klar ist, welches die Interfacekanten sind).

In der Abbildung 4.6 wird gezeigt, wie das Problem der zeitscheibenüberspringenden Kante behandelt wird. Durch neue Knoten wird die zeitscheibenüberspringende Kante in die Zeitscheiben eingebettet, die sie überspringt. In Abbildung 4.9 ist die Triangulierung mit der eingeschränkten Eliminationsreihenfolge dargestellt. Man erkennt, daß es nur innerhalb einer Zeitscheibe oder maximal zwischen benachbarten Zeitscheiben Triangulationskanten existieren. Dieses erzeugt einen Junction Tree, dessen Struktur den Aufbau des Dynamischen Bayesschen Netzes widerspiegelt. Es ist jetzt wieder ein Rollup mit den Mitteln aus Kapitel 3 möglich. Dieser Rollup wird in den Abbildungen 4.7(a) und (b) dargestellt. Wie man in diesen Abbildungen erkennt, umfassen die Zustandsentwicklungsmodelle die zeitscheibenüberspringenden Kanten.

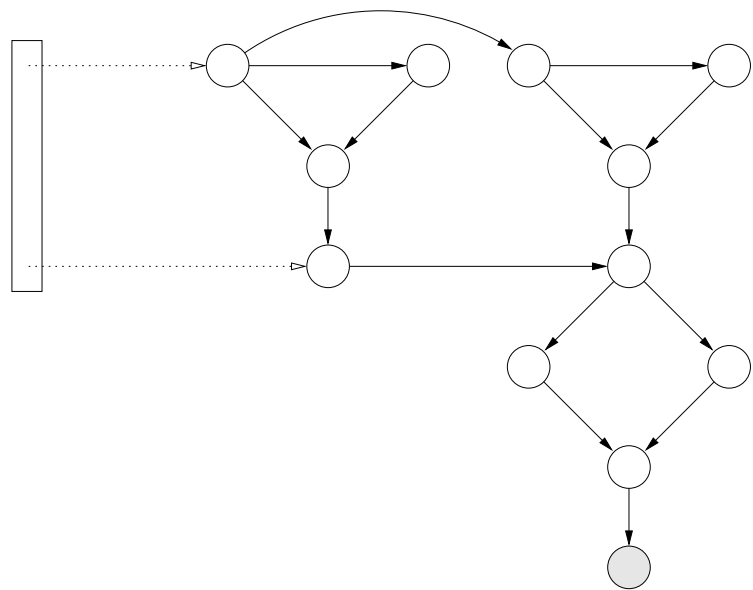
Jetzt sollen die triangulierten Graphen in den Abbildungen 4.8(a), 4.8(b) und 4.9 noch einmal genauer betrachtet werden. Beim triangulierten Graphen in Abbildung 4.8(a) existieren nur Triangulationskanten innerhalb einer Zeitscheibe oder zwischen zwei benachbarten Zeitscheiben. Der Junction Tree, der aus diesem triangulierten Graphen entsteht, enthält nur Cliques, die Knoten aus einer Zeitscheibe oder maximal aus zwei benachbarten Zeitscheiben enthalten. Ein solcher Junction Tree baut sich also ähnlich wie das Dynamische Bayessche Netz aus Zeitscheiben auf. Dabei lassen sich die Cliques des Junction Trees, die Knoten aus zwei verschiedenen Zeitscheiben enthalten, mit den Dynamischen Knoten des Dynamischen Bayesschen Netzes vergleichen. Beide übernehmen die Vermittlerrolle von einer zur nächsten Zeitscheibe.

Beim triangulierten Graphen in Abbildung 4.8(b) ist zu erkennen, daß die Trian-



Zustandsentwicklungsmodell Zeitscheibe $t - 2$ Zeitscheibe $t - 1$

(a) DBN zum Zeitpunkt $t - 1$: Die Zeitscheibe $t - 3$ ist aufgerollt.



Zustandsentwicklungsmodell Zeitscheibe $t - 1$ Zeitscheibe t

(b) DBN zum Zeitpunkt t : Die Zeitscheibe t wurde etabliert und die Zeitscheibe $t - 2$ aufgerollt.

Abbildung 4.5: Beispiel: Zeitscheibenüberspringende Kanten ohne Korrektur beim Rollup

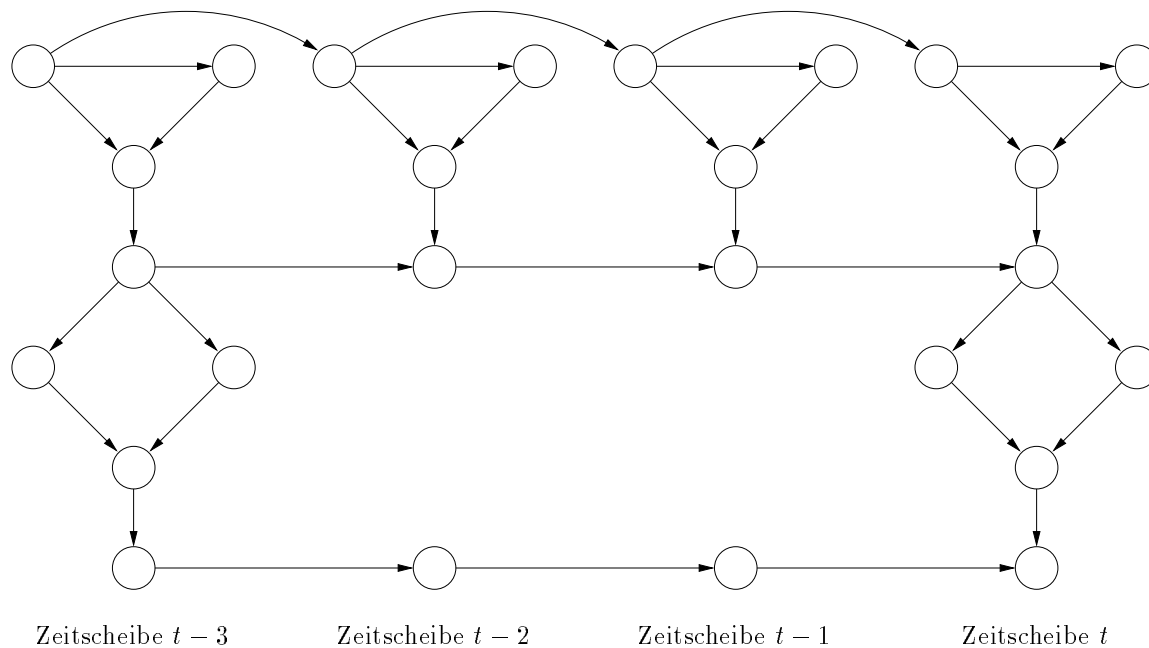


Abbildung 4.6: Beispiel: Zeitscheibenüberspringende Kanten mit Korrektur

gulationskanten des triangulierten Graphen aus Abbildung 4.8(a) enthalten sind und weitere Triangulationskanten hinzu gekommen sind. Einige dieser neuen Triangulationskanten überspringen Zeitscheiben und lassen somit die involvierten Zeitscheiben zu einer Zeitscheibe zusammenfallen. Der aus dem triangulierten Graphen in Abbildung 4.8(a) resultierende Junction Tree kann bei der Bestimmung für den Junction Tree des triangulierten Graphen in Abbildung 4.8(b) nur wenig wiederverwendet werden. Desweiteren sind die vielen Triangulationskanten, die in einem Knoten enden, ein Indiz dafür, daß große Cliques entstehen. Die Cliques dieses Junction Trees enthalten Knoten aus mehreren Zeitscheiben. Ein solcher Junction Tree ist nicht für die Rollup-Verfahren geeignet, wie sie in Kapitel 3 vorgestellt werden.

Beim triangulierten Graphen in Abbildung 4.9 existieren wieder nur Triangulationskanten innerhalb einer Zeitscheibe oder zwischen zwei benachbarten Zeitscheiben. Der Junction Tree, der aus diesem triangulierten Graphen entsteht, enthält nur Cliques, die Knoten aus einer Zeitscheibe oder maximal aus zwei benachbarten Zeitscheiben enthalten. Ein solcher Junction Tree baut sich also ähnlich wie das Dynamische Bayessche Netz aus Zeitscheiben auf und ist wieder für die Rollup-Verfahren geeignet, wie sie im Kapitel 3 vorgestellt werden. Desweiteren erkennt man, daß die Cliques des zugehörigen Junction Trees im Vergleich zum zugehörigen Junction Trees des triangulierten Graphen in Abbildung 4.8(b) kleiner sind.

Bsp

Eine zeitscheibenüberspringende Kante kann sich über beliebig viele Zeitscheiben erstrecken. Für jede zeitscheibenüberspringende Kante ist pro Zeitscheibe, die sie überspringt, ein Zwischenknoten einzufügen. Die Komplexität des daraus entstandenen Netzes läßt sich allerdings erst feststellen, wenn ein Rollup-Verfahren oder Inferenzalgorithmus angewendet wird. Die Komplexität des Dynamischen Bayesschen Netzes wird im dazugehörigen Junction Tree an der Größe der einzelnen Cliques festgestellt.

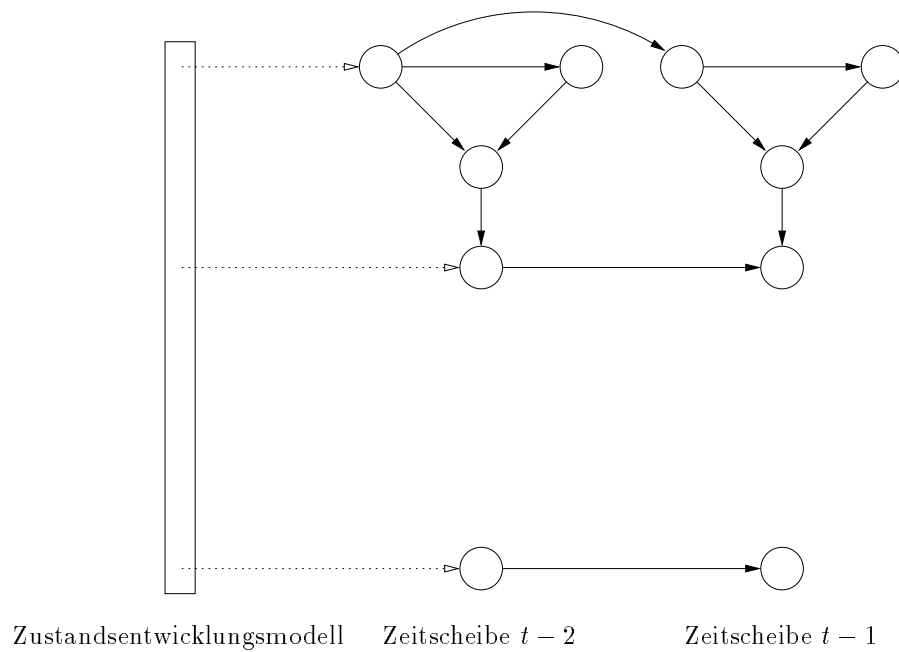
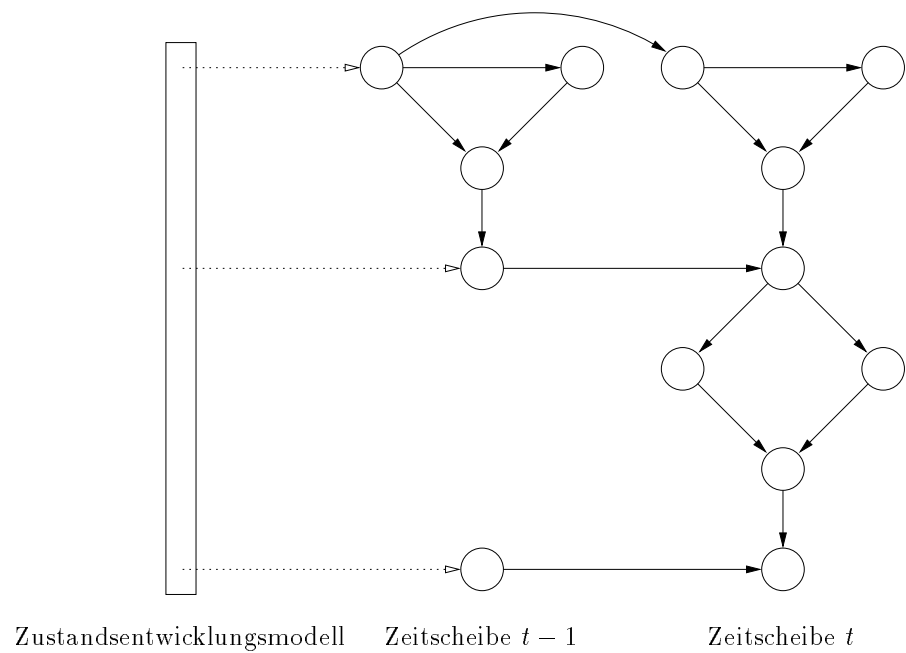
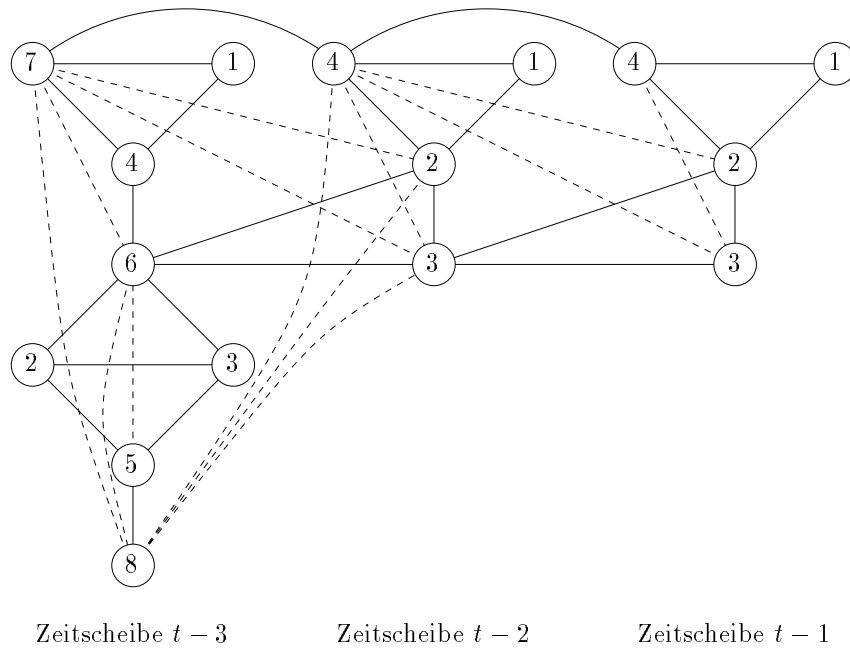
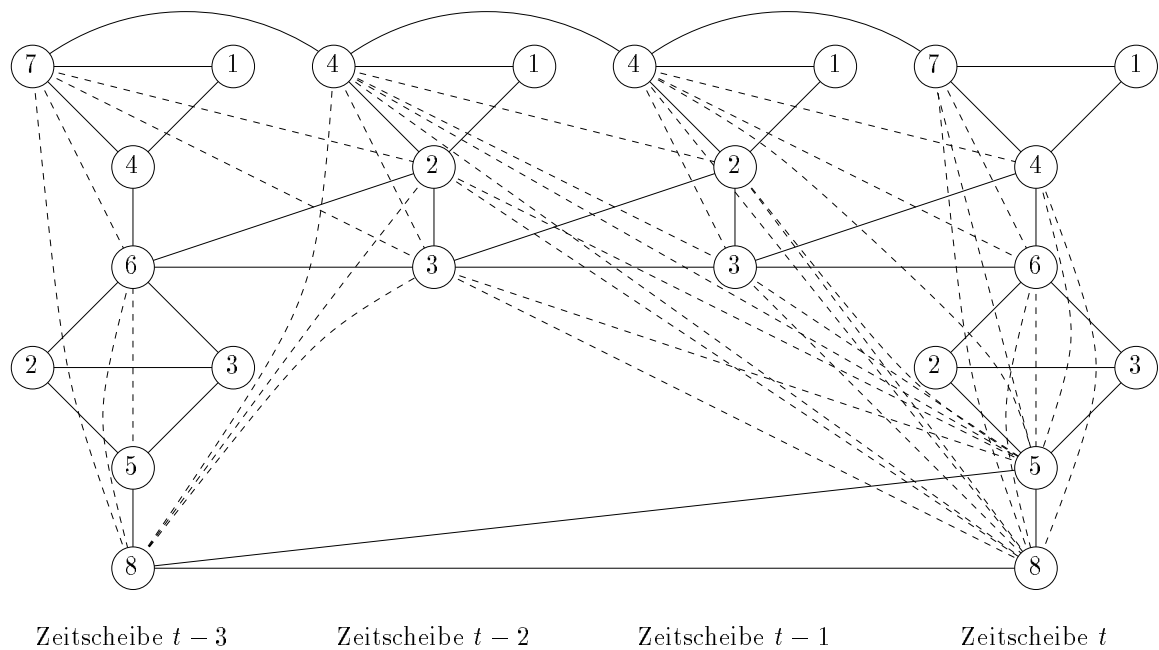
(a) DBN zum Zeitpunkt $t - 1$: Die Zeitscheibe $t - 3$ ist aufgerollt.(b) DBN zum Zeitpunkt t : Die Zeitscheibe t wurde etabliert und die Zeitscheibe $t - 2$ aufgerollt.

Abbildung 4.7: Beispiel: Zeitscheibenüberspringende Kanten mit Korrektur beim Rollup



(a) DBN zum Zeitpunkt $t - 1$: Es existieren nur Triangulationskanten innerhalb einer Zeitscheibe oder zwischen benachbarten Zeitscheiben.



(b) DBN zum Zeitpunkt t : Es existieren Triangulationskanten über mehrere Zeitscheiben hinweg. Die Zeitscheiben von $t - 3$ bis $t - 1$ fallen zusammen.

Abbildung 4.8: Beispiel: Triangulierung bei zeitscheibenüberspringenden Kanten ohne Korrektur

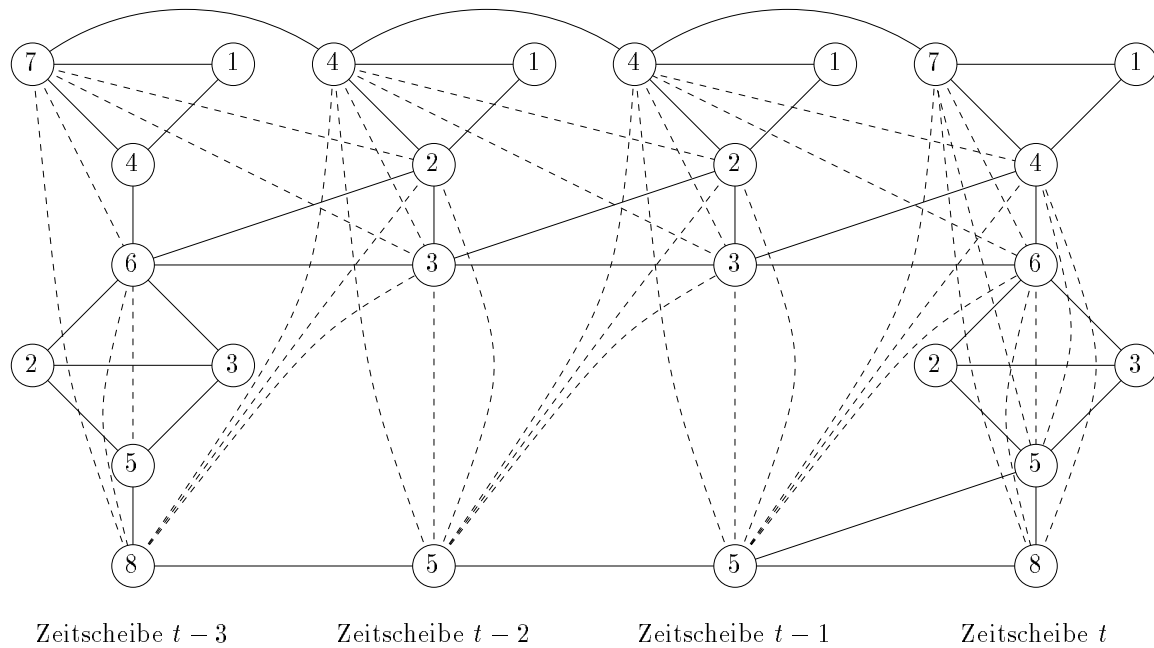


Abbildung 4.9: Beispiel: Triangulierung bei zeitscheibenüberspringenden Kanten mit Korrektur

Die Bestimmung eines Junction Trees ist entscheidend von der Eliminationsreihenfolge abhängig. Da die Berechnung eines optimalen Junction Trees *NP*-vollständig ist, kann es durchaus vorkommen, daß der neue Junction Tree nach dem Hinzufügen von neuen Knoten weniger komplex ist als der alte Junction Tree, was einigen Lesern kurios⁵ vorkommen mag.

Ein Spezialfall von zeitscheibenüberspringenden Kanten sind in den READY-Netzen die *Doppelschemata*.

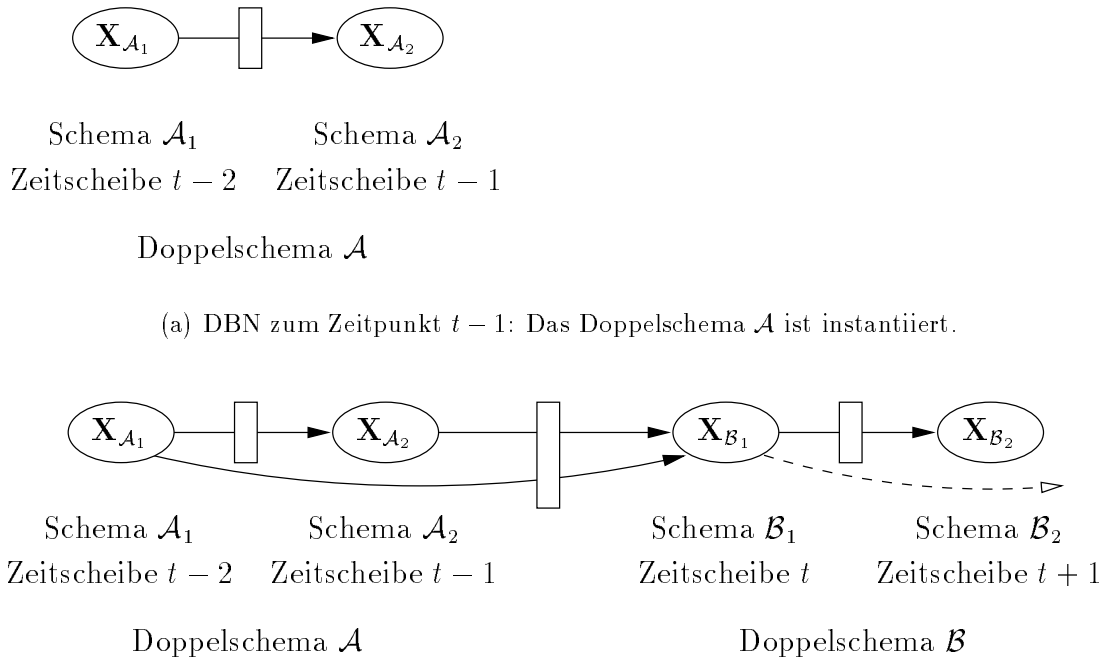
Doppelschemata

Bei den READY-Netzen werden Sinneinheiten von zwei Schemata zu einem Doppelschema zusammengefaßt. Auf eine Frage wird eine Antwort erwartet. Auf eine Instruktion soll eine Handlung erfolgen. Wird als Zeitscheibe das Schema **Frage** instantiiert, so wird gleichzeitig das Schema **Antwort** als darauffolgende Zeitscheibe erzeugt. In den einzelnen Schemata seien ausschließlich die Zusammenhänge modelliert, die für den Sachverhalt relevant sind, für den das Schema entworfen wurde.

Im folgenden wird beschrieben, wie bei den Doppelschemata zeitscheibenüberspringende Kanten entstehen können. Das Problem der Doppelschemata wird dann auf das Problem der zeitscheibenüberspringenden Kanten zurückgeführt. Somit kann das Problem der Doppelschemata wie das Problem der zeitscheibenüberspringenden Kanten gelöst werden, indem Zwischenknoten an den entsprechenden Stellen ins Bayessche Netz eingefügt werden.

⁵Dem Leser, dem dies auch beim wiederholten Lesen des Satzes immer noch kurios vorkommt, wird empfohlen, sich noch einmal Abschnitt 1.3.1 anzuschauen.

Es sei ein Dynamisches Bayessches Netz \mathcal{DBN} zum Zeitpunkt $t - 3$ gegeben, das nicht aufgerollt wird. Es werde nun das Doppelschema \mathcal{A} mit den Schemata \mathcal{A}_1 und \mathcal{A}_2 instantiiert. Dabei entstehen die beiden Zeitscheiben $t - 2$ und $t - 1$. In Abbildung 4.10(a) ist ein Schnappschuß des resultierenden Dynamischen Bayesschen Netzes zum Zeitpunkt $t - 1$ zu sehen. Es werden die Zeitscheiben $t - 2$ und $t - 1$ mit dem dazwi-



(a) DBN zum Zeitpunkt $t - 1$: Das Doppelschema \mathcal{A} ist instantiiert.

(b) DBN zum Zeitpunkt t : Es wird das Doppelschemata \mathcal{B} instantiiert. Dadurch werden zeitscheibenüberspringende Kanten von Zeitscheibe $t - 2$ zu Zeitscheibe t induziert.

Abbildung 4.10: Zeitscheibenüberspringende Kanten: Die Zufallsvariablen in Modell \mathcal{B}_1 sind von Zufallsvariablen abhängig, die weiter zurückliegen als die direkt vorangegangene Zeitscheibe.

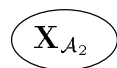
schenliegenden Zustandsentwicklungsmodell schematisch dargestellt. Die Zeitscheiben bis vor dem Zeitpunkt $t - 2$ sind ausgeblendet. Zum Zeitpunkt $t - 1$ besteht das Dynamische Bayessche Netz somit aus t Zeitscheiben. Jetzt wird das Doppelschema \mathcal{B} mit den beiden Schemata \mathcal{B}_1 und \mathcal{B}_2 als Zeitscheiben t und $t + 1$ instantiiert. Die beiden Doppelschemata \mathcal{A} und \mathcal{B} können auch identisch sein. Beispielsweise sei im Schema \mathcal{B}_1 ein Knoten enthalten, der eine Zufallsvariable repräsentiert, die von einer Zufallsvariablen beeinflusst wird, deren entsprechender Knoten im Schema \mathcal{A}_1 enthalten sei. Das Zustandsentwicklungsmodell für $\mathbf{X}_{\mathcal{B}_1}$ zum Zeitpunkt t ist nicht mehr nur von der Zeitscheibe $t - 1$ abhängig, sondern auch von der Zeitscheibe $t - 2$, in der das Schema \mathcal{A}_1 als Zeitscheibe instantiiert wurde. Es wird eine *zeitscheibenüberspringende Kante* von der Zeitscheibe $t - a$ zur Zeitscheibe t induziert⁶. Dies ist in der Abbildung 4.10(b) durch einen gerichteten Pfeil von Zeitscheibe $t - 2$ zu Zeitscheibe t dargestellt. Das Schema \mathcal{A}_2 und das Zustandsentwicklungsmodelle zwischen \mathcal{A}_1 und \mathcal{A}_2 ändern sich

⁶Um eine zeitscheibenüberspringende Kante zu induzieren, könnten auch die Schemata \mathcal{B}_2 und \mathcal{A}_1 bzw. \mathcal{B}_2 und \mathcal{A}_2 involviert sein.

dabei nicht.

Werden keine Zeitscheiben aufgerollt, so muß zwar der Junction Tree für das Dynamische Bayessche Netz insgesamt neu berechnet werden, ohne daß auch alten Berechnungen wiederzuverwenden sind, aber es ergeben sich keine weiteren Probleme. Wenn jedoch der Rollup immer dann durchgeführt wird, bevor ein neues Doppelschema instantiiert wird, so stellt sich die Situation allerdings anders dar: In Abbildung 4.11(a) ist das Dynamische Bayessche Netz zum Zeitpunkt $t - 1$ zu sehen, bevor das Doppelschema \mathcal{B} etabliert wird. Bis zum Zeitpunkt $t - 2$ wurden die Zeitscheiben h ($0 \leq h \leq t - 2$) jeweils dann aufgerollt, wenn mehr als zwei Zeitscheiben etabliert war. Zu jedem Zeitpunkt h ($0 \leq h \leq t - 2$) waren somit immer maximal nur drei Zeitscheiben des Dynamischen Bayesschen Netzes repräsentiert.

Wird nun zum Zeitpunkt t das Doppelschema \mathcal{B} mit den Schemata \mathcal{B}_1 und \mathcal{B}_2 als Zeitscheiben t und $t + 1$ instantiiert, so läßt sich die Abhängigkeit der Zeitscheibe t von Zeitscheibe $t - 2$, in der das Schema \mathcal{A}_1 instantiiert wurde, nicht mehr modellieren. In Abbildung 4.11(a) und (b) sind die aufgerollten Zeitscheiben gestrichelt eingezeichnet. In Abbildung 4.11(b) erkennt man zusätzlich, daß die zeitscheibenüberspringende

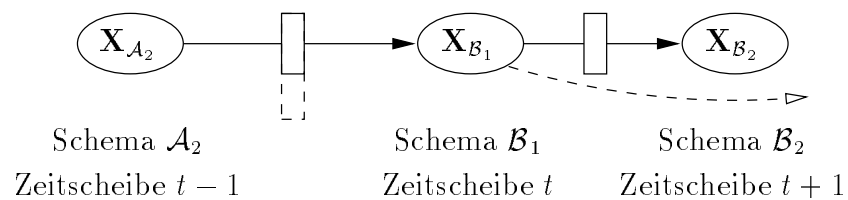


Schema \mathcal{A}_2

Zeitscheibe $t - 1$

Doppelschema \mathcal{A}

(a) DBN nach dem Rollup zum Zeitpunkt $t - 1$: Vom Doppelschema \mathcal{A} ist noch das Schema \mathcal{A}_2 vorhanden.



Schema \mathcal{A}_2

Zeitscheibe $t - 1$

Schema \mathcal{B}_1

Zeitscheibe t

Schema \mathcal{B}_2

Zeitscheibe $t + 1$

Doppelschema \mathcal{B}

(b) DBN zum Zeitpunkt t : Es wird das Doppelschema \mathcal{B} instantiiert. Es werden keine zeitscheibenüberspringende Kanten induziert.

Abbildung 4.11: Zeitscheibenüberspringende Kanten beim Rollup ohne Korrektur: Es werden nicht mehr alle Abhängigkeiten berücksichtigt.

Kante nicht mehr etabliert werden kann. Insbesondere wird die Abhängigkeit der Zeitscheibe t von der Zeitscheibe $t - 2$ nicht mehr modelliert.

Damit ein Rollup mit den Mitteln aus Kapitel 3 möglich ist, muß also eine mögliche Beeinflussung einer Zeitscheibe auf eine andere Zeitscheibe durch alle Zeitscheiben, die dazwischen liegen, durchgereicht werden. In Abbildung 4.12 ist dargestellt, daß sich

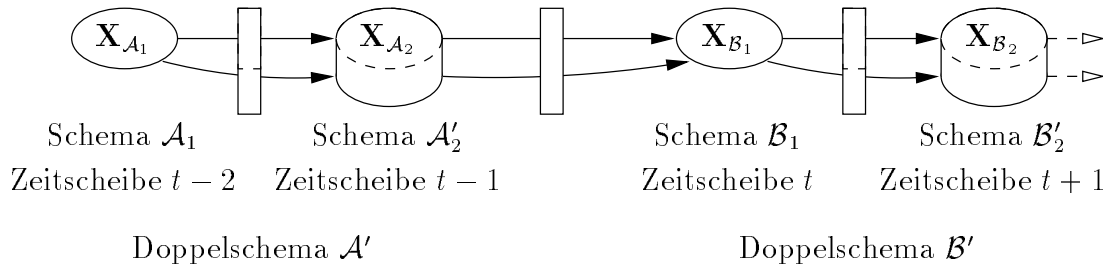


Abbildung 4.12: Zeitscheibenüberspringende Kanten mit Korrektur: Die verschiedenen Schema der Doppelschemata werden um Zwischenknoten ergänzt, so daß eine zeitscheibenüberspringende Kante im jeweiligen Teilschema zu liegen kommt.

die Schemata und die Zustandsentwicklungsmodelle der Zeitscheiben erweitern. Man sieht, daß temporale Kanten hinzukommen.

Im Fall der Doppelschemata muß nun in vergleichbarer Weise wie im allgemeinen Fall der zeitscheibenüberspringenden Kanten vorgegangen werden. Entsprechende Knoten sind in den verschiedenen Schemata neu zu modellieren. Wie die bedingten Wahrscheinlichkeiten dieser entsprechenden Knoten zu wählen sind, ist ebenfalls im allgemeinen Fall der zeitscheibenüberspringenden Kanten behandelt und kann unverändert übernommen werden. Jetzt kann der Rollup wie gewohnt mit den Mitteln aus dem Kapitel 3 durchgeführt werden.

4.1.2 Statische Knoten

Statische Knoten modellieren Eigenschaften, die über die Zeit unverändert bleiben, in der das Dynamische Bayessche Netz gültig ist. Der BEL-Wert eines Statischen Knotens, d.h. seine Einschätzung, kann sich jedoch sehr wohl ändern. Statische Knoten beeinflussen verschiedene Zeitscheiben, sind aber keiner Zeitscheibe zugeordnet und im gesamten READY-Netz nur einmal instantiiert.

In diesem Abschnitt wird beschrieben, warum die Statischen Knoten für Dynamische Bayessche Netze und insbesondere für die Rollup-Verfahren aus Kapitel 3 problematisch sind. Desweiteren wird gezeigt, wie sich das Problem der Statischen Knoten auf das Problem der zeitscheibenüberspringenden Kanten zurückführen und somit lösen läßt.

Der Knoten WISSENSNIVEAU DES SCHÜLERS aus Beispiel 2.1.2 ist ein typisches Beispiel für einen statischen Knoten, mit dem Unterschied, daß der Knoten WISSENSNIVEAU DES SCHÜLERS in jeder Zeitscheibe modelliert ist.

Es sei nun ein Dynamisches Bayessches Netz \mathcal{DBN} zum Zeitpunkt $t-3$ gegeben, das nicht aufgerollt wird. Werde nun als Zeitscheibe $t-2$ das Schema \mathcal{A} und als Zeitscheibe $t-1$ das Schema \mathcal{B} instantiiert. Diese beiden Schemata seien von den Statischen Knoten X_1, \dots, X_a abhängig. Zu jedem Zeitpunkt $h \leq t-1$ seien die Schemata des Dynamischen Bayesschen Netzes so instantiiert, daß keine zeitscheibenüberspringende Kanten induziert werden⁷. Dieses wird im generischen Schema durch die gerichteten

⁷Diese Annahme kann vorausgesetzt werden, da ansonsten das Verfahren aus dem Abschnitt 4.1.1 zur Anwendung kommt.

Kanten zwischen den konsekutiven Zeitscheiben angedeutet. Die Zeitscheiben erfüllen damit jedoch nicht die Markov-Eigenschaft für Dynamische Bayessche Netze, da die Zeitscheiben auch von den Statischen Knoten abhängig sind. Die Problematik wird im folgenden dargestellt.

In Abbildung 4.13(a) ist ein Schnappschuß des resultierenden Dynamischen Bayesschen Netzes zum Zeitpunkt $t - 1$ zu sehen.

Es werden die Zeitscheiben $t - 2$ und $t - 1$ schematisch dargestellt. Die Zeitscheiben bis vor dem Zeitpunkt $t - 2$ sind ausgeblendet. Nicht alle dargestellten Statischen Knoten beeinflussen alle Schemata. So beeinflusst der Statische Knoten X_1 nur direkt das Schema \mathcal{A} . Zum Zeitpunkt $t - 1$ besteht das Dynamische Bayessche Netz somit aus t Zeitscheiben.

Beispielhaft werden noch als Zeitscheiben t und $t + 1$ die Schemata \mathcal{C} und \mathcal{D} instantiiert. Wie in Abbildung 4.13(b) zu sehen ist, induzieren manche Schemata neue Statische Knoten, die noch nicht instantiiert wurden, andere Schemata greifen auf vorhandene Statische Knoten zurück. Ebenso beeinflussen manche Statische Knoten immer nur Zeitscheiben in denen ein bestimmtes Schema instantiiert wurde (z.B. der Statische Knoten X_1 das Schema \mathcal{A}). Andere Statische Knoten beeinflussen jede Zeitscheibe.

Wie im Fall der zeitscheibenüberspringenden Kanten und der Doppelschemata ergeben sich die Probleme mit den Statischen Knoten erst, wenn der Rollup (siehe Kapitel 3) immer dann durchgeführt wird, wenn eine neue Zeitscheibe etabliert wurde.

Werden keine Zeitscheiben aufgerollt, so muß der zugehörige Junction Tree des Dynamischen Bayesschen Netzes zwar insgesamt neu berechnet werden, aber es ergeben sich keine weiteren Probleme (vorausgesetzt, daß sich der Junction Tree des Dynamischen Bayesschen Netzes noch berechnen läßt.). Die Situation stellt sich allerdings anders dar, wenn der Rollup immer dann durchgeführt wird, wenn eine neue Zeitscheibe etabliert wurde (siehe Kapitel 3). Wird die Zeitscheibe $t - 2$ des Bayesschen Netzes in Abbildung 4.13(b) aufgerollt, so stellt sich die Frage, ob die Statischen Knoten X_1, \dots, X_a auch zur Zeitscheibe $t - 2$ gehören und mitaufgerollt werden. Zumindest der Knoten X_a kann nicht ohne weiteres aufgerollt werden, da er auch noch andere Zeitscheiben beeinflusst. Auch der Knoten X_1 darf nicht aufgerollt werden, denn in ferner Zukunft könnte wieder eine Zeitscheibe mit dem Schema \mathcal{A} instantiiert werden. Ebenso könnte man an dem BEL-Wert des statischen Knotens X_1 , der sich durch das Eintragen von Beobachtungen in den anderen Zeitscheiben auch ändert, die ganze Zeit über interessiert sein.

Die Statischen Knoten dürfen also nie aufgerollt werden, und sie benötigen immer eine Verbindung zu den Zeitscheiben, damit sie ihren BEL-Wert jeweils an die entsprechenden Situationen anpassen können.

Betrachtet man die Menge aller möglichen Statischen Knoten als besondere Zeitscheibe -1 (siehe Abbildung 4.14(a)), dann läßt sich das Problem der Statischen Knoten auf das Problem von zeitscheibenüberspringenden Kanten reduzieren. Jedes Schema wird dabei um alle Statischen Knoten ergänzt (siehe Abbildung 4.14(b)) und die bedingten Wahrscheinlichkeiten der ergänzten Statischen Knoten werden wie schon im Abschnitt 4.1.1 definiert. In Abbildung 4.14 ist auch dargestellt, daß sich die Schemata und die Zustandsentwicklungsmodelle zwischen den Zeitscheiben erweitern. Man sieht, daß temporale Kanten hinzukommen. Jetzt kann der Rollup wie gewohnt mit den Mitteln aus dem Kapitel 3 durchgeführt werden, und es müssen maximal immer

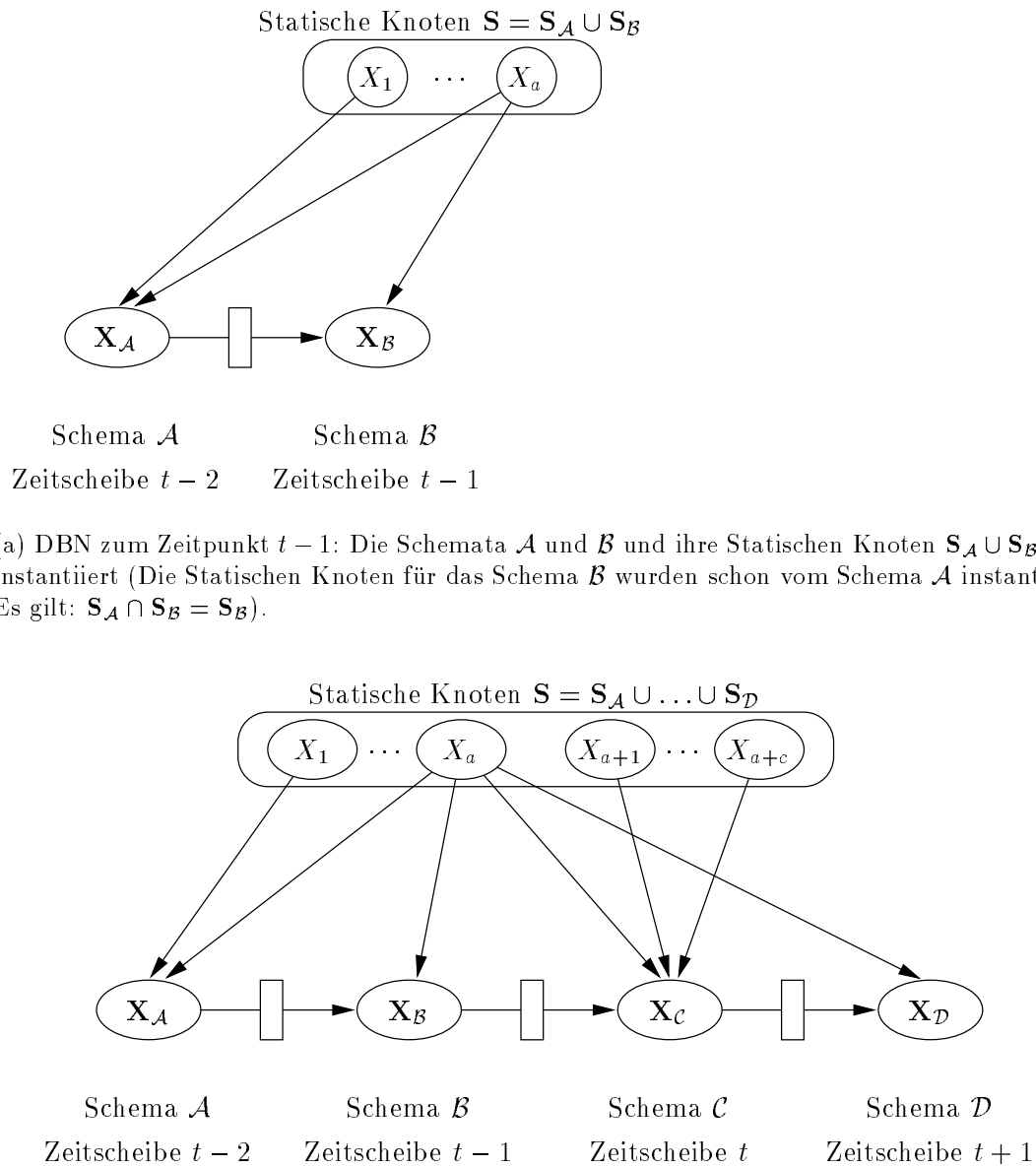
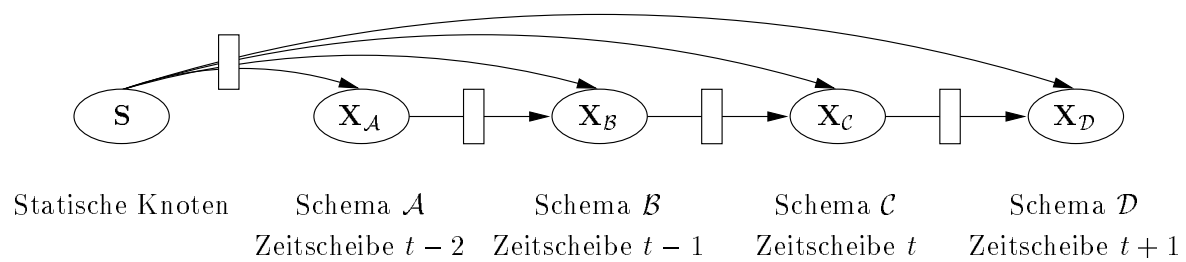
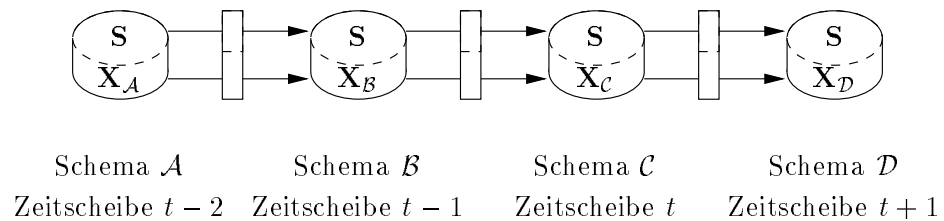


Abbildung 4.13: Statische Knoten: Die Knoten der Schemata sind nur von gewissen Statische Knoten abhängig.



(a) DBN zum Zeitpunkt $t + 1$: Die Statischen Knoten werden als eigene Zeitscheibe betrachtet. Man erhält den allgemeinen Fall von zeitscheibenüberspringenden Kanten.



(b) DBN zum Zeitpunkt $t + 1$: Die Statischen Knoten werden in die einzelnen Schemata eingebettet, so daß es keine zeitscheibenüberspringende Kanten mehr gibt.

Abbildung 4.14: Statische Knoten: Die Statischen Knoten werden in die Schemata eingebettet.

nur zwei Zeitscheiben vorhanden sein.

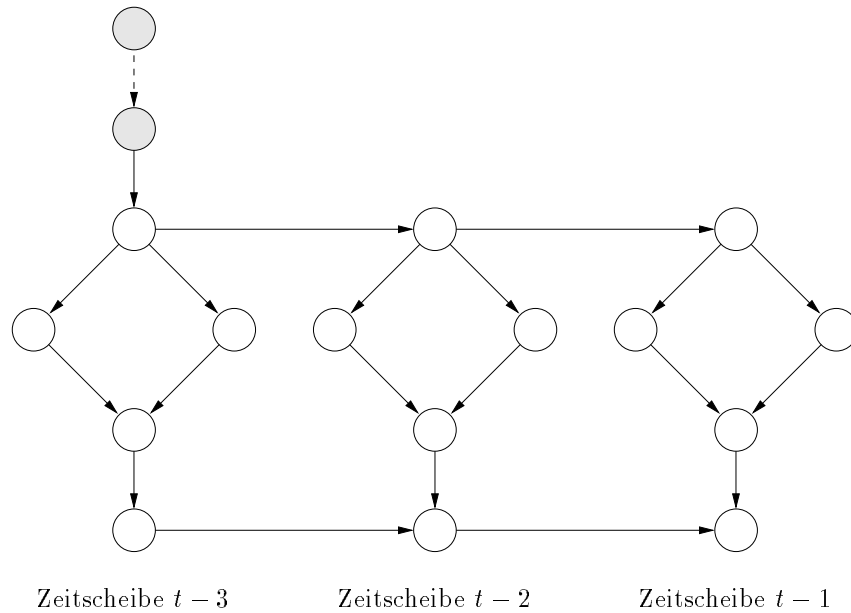
Beispiel 4.1.2 (Statische Knoten)

Im folgenden wird an einem kleinen Beispielnetz konkret gezeigt, welche Fragestellungen bei Statischen Knoten aufkommen und wie die Probleme gelöst werden. Als Beispiel sei dazu das Netz in Abbildung 4.15(a) gegeben, von dem die Zeitscheiben von $t - 3$ bis $t - 1$ dargestellt sind. Bis jetzt wirke nur ein statischer Knoten auf die Zeitscheiben. Diesen statischen Knoten kann man zum jetzigen Zeitpunkt auch als zur Zeitscheibe $t - 3$ zugehörig ansehen.

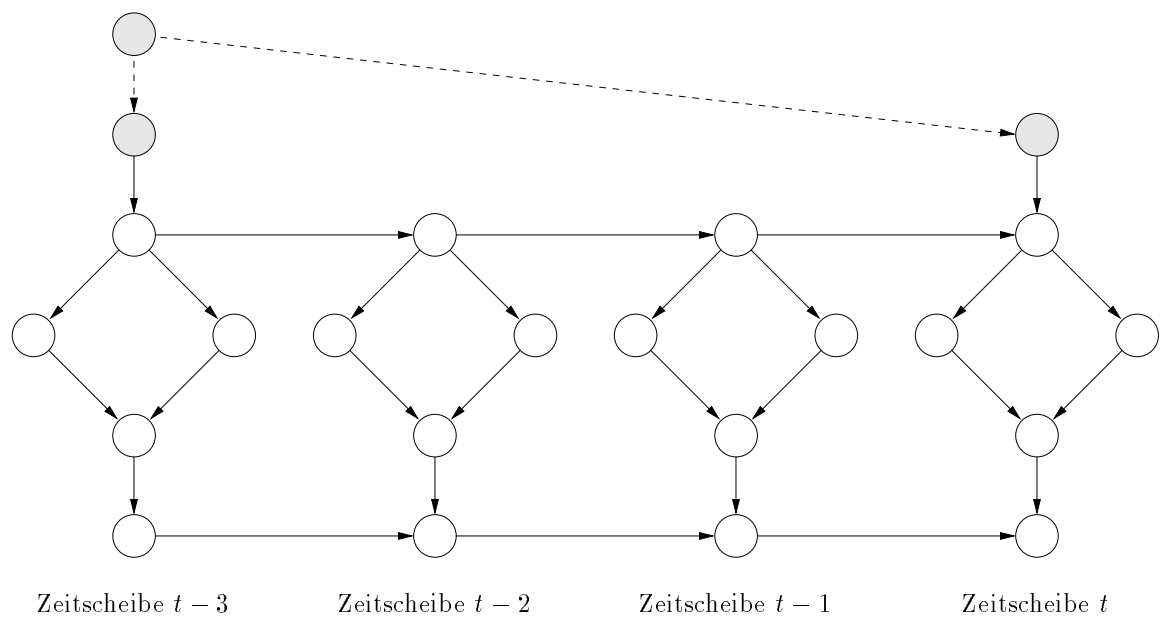
Jetzt wird ein Schema als Zeitscheibe t instantiiert, das vom selben statischen Knoten wie die Zeitscheibe $t - 3$ beeinflusst wird. Es wird von der Zeitscheibe t eine Kante induziert, die vom statischen Knoten zur Zeitscheibe t geht (siehe Abbildung 4.15(b)). Würde der statische Knoten nicht existieren, so würde er jetzt instantiiert werden.

Betrachtet man nun einen statischen Knoten zu der Zeitscheibe zugehörig, von der er induziert wurde, so können die Kanten, die von diesem statischen Knoten aus zu anderen Zeitscheiben gehen, als zeitscheibenüberspringende Kanten angesehen werden. Auf diese Art läßt sich das Problem der statischen Knoten auf das Problem der zeitscheibenüberspringenden Kanten übertragen. Die vom statischen Knoten ausgehenden Kanten sind gestrichelt eingezeichnet und der statische Knoten sowie die anderen beteiligten Knoten sind grau schraffiert dargestellt.

Die Probleme, die beim Rollup auftreten, wenn er immer durchgeführt wird, so daß maximal zu jedem Zeitpunkt nur zwei Zeitscheiben des Dynamischen Bayesschen Netzes repräsentiert sind, sind in den beiden Abbildungen 4.16(a) und 4.16(b) dar-



(a) DBN zum Zeitpunkt $t - 1$: Als letztes wurde die Zeitscheibe $t - 1$ instantiiert. Bis jetzt wirkt nur auf die Zeitscheibe $t - 3$ ein Statischer Knoten.



(b) DBN zum Zeitpunkt t : Die Zeitscheibe t wird ebenfalls vom statischen Knoten beeinflusst.

Abbildung 4.15: Beispiel: Statischer Knoten ohne Korrektur

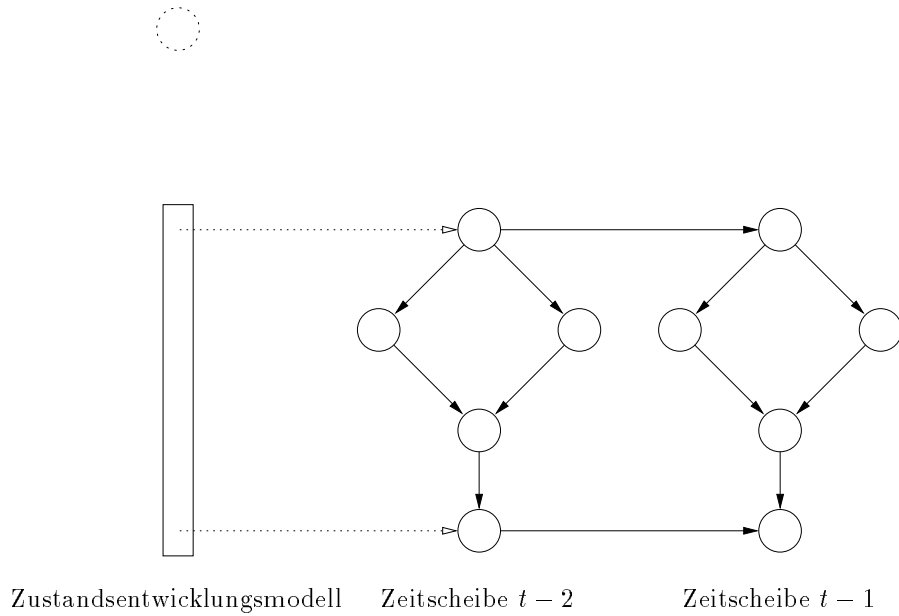
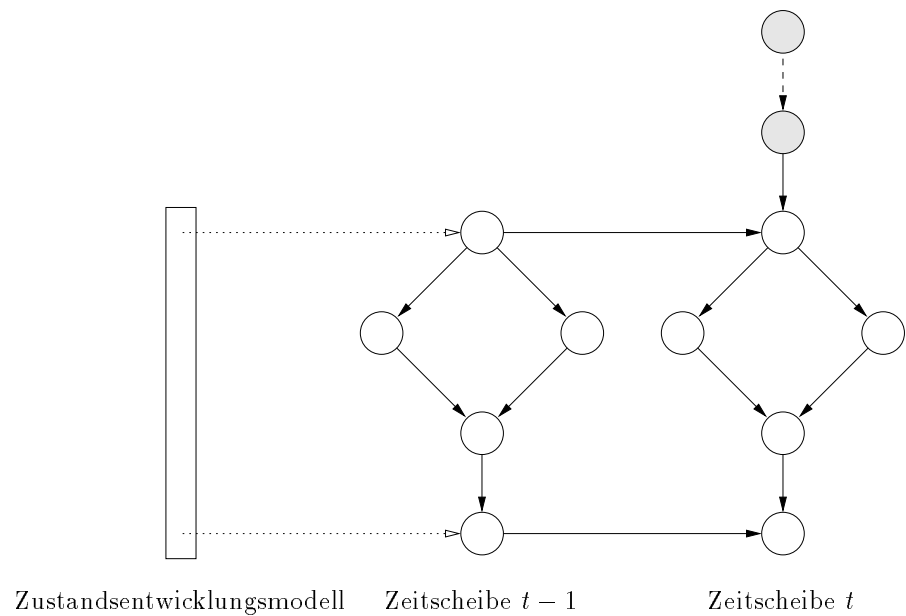
(a) DBN zum Zeitpunkt $t - 1$: Die Zeitscheibe $t - 3$ ist aufgerollt.(b) DBN zum Zeitpunkt t : Die Zeitscheibe t wurde etabliert und die Zeitscheibe $t - 2$ aufgerollt.

Abbildung 4.16: Beispiel: Statischer Knoten ohne Korrektur beim Rollup

gestellt. In der Abbildung 4.16(a) ist der statische Knoten gestrichelt eingezeichnet. Berücksichtigt man, daß der statische Knoten zu keiner Zeitscheibe gehört, dann wird er durch den Rollup von den anderen Zeitscheiben abgetrennt. Somit wird er auch nicht mehr aufgefrischt, wenn neue Evidenzen in den noch vorhandenen Zeitscheiben eingetragen werden. Nimmt man hingegen an, daß der statische Knoten zur entsprechenden Zeitscheibe gehört, so wird er einfach beim Rollup mit der Zeitscheibe gelöscht. In beiden Fällen jedoch nimmt er nicht mehr seine Funktion als statischer Knoten wahr. Dies wird durch das Zustandsentwicklungsmodell, das als Rechteck eingezeichnet ist, dargestellt. Wie man sieht, umfaßt es den statischen Knoten nicht. In Abbildung 4.16(b) wurde die Zeitscheibe $t - 2$ aufgerollt und dann die Zeitscheibe t etabliert. Bei der Instantiierung von Zeitscheibe t wurde vom statischen Knoten eine Kante auf die Zeitscheibe t induziert. Dieser statische Knoten berücksichtigt aber nicht, daß er auch die Zeitscheibe $t - 3$ beeinflusst hat. Vereinfachend kann man annehmen, daß der statische Knoten neu instantiiert wurde.

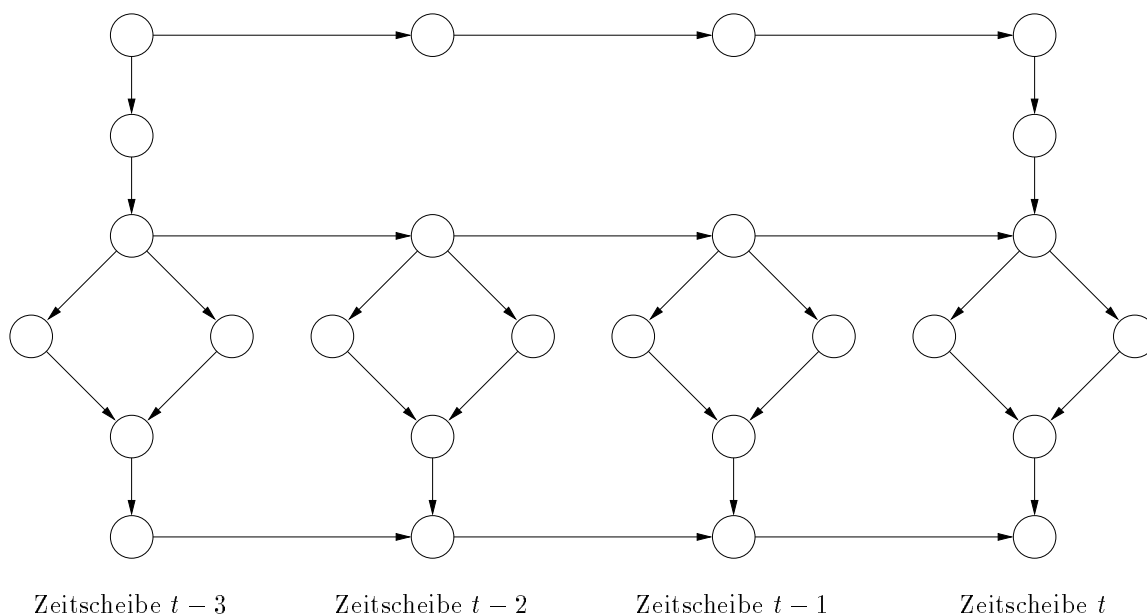


Abbildung 4.17: Beispiel: Statischer Knoten mit Korrektur

Das Problem der statischen Knoten muß nun so bearbeitet werden, daß man zu jedem Zeitpunkt des Rollups auch die aktuellen BEL-Werte der statischen Knoten abfragen kann. Dazu müssen die statischen Knoten zu jedem Zeitpunkt des Rollups auch vorhanden und in die Zeitscheiben eingebettet sein.

In der Abbildung 4.17 ist zu sehen, wie das Problem der statischen Knoten gelöst wird. Durch neue Knoten wird ein statischer Knoten in jeder Zeitscheibe des dynamischen Bayesschen Netzes eingebettet. Somit ist jetzt wieder ein Rollup mit den Mitteln aus Kapitel 3 möglich. Dieser Rollup wird in den Abbildungen 4.18(a) und (b) dargestellt. Wie man sieht, umfassen die Zustandsentwicklungsmodelle jetzt auch den statischen Knoten. Über diese Zustandsentwicklungsmodelle wird der statische Knoten auch aufgefrischt, wenn in Zeitscheiben neue Evidenzen eingetragen werden, auf die der Knoten keine direkte Beeinflussung ausübt.

Die Grafiken für die Triangulierungen des verschiedenen Bayesschen Netzes sehen

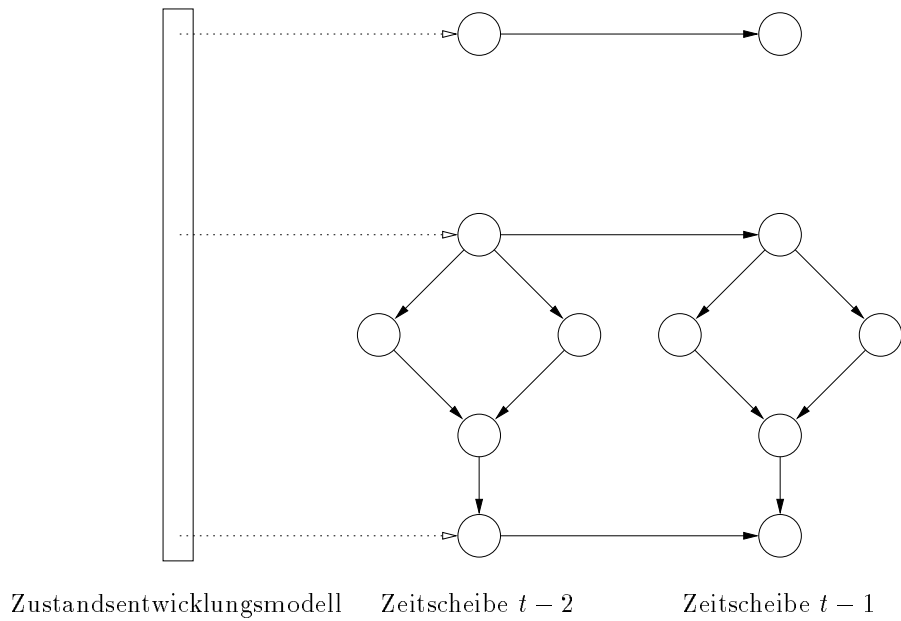
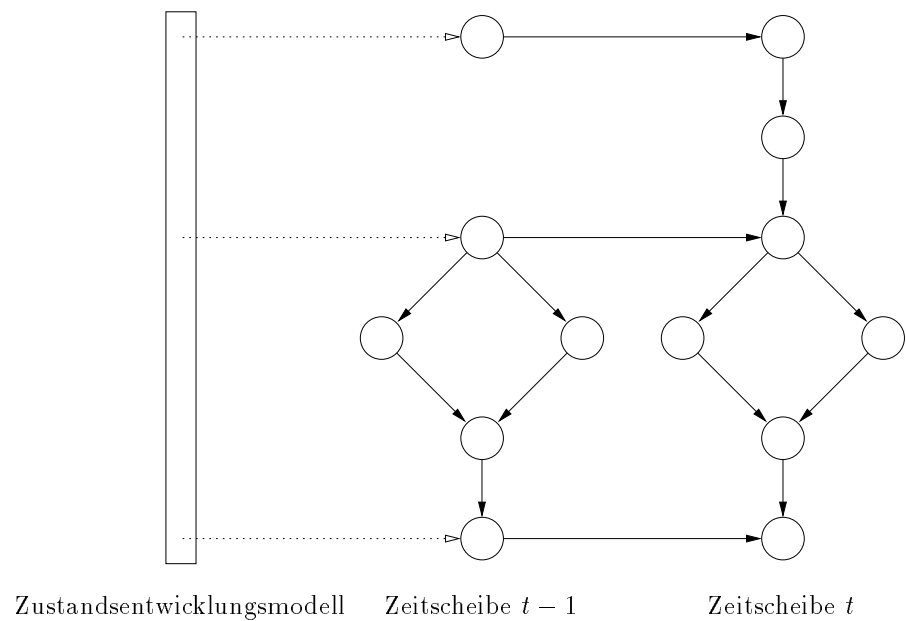
(a) DBN zum Zeitpunkt $t - 1$: Die Zeitscheibe $t - 3$ ist aufgerollt.(b) DBN zum Zeitpunkt t : Die Zeitscheibe t wurde etabliert und die Zeitscheibe $t - 2$ aufgerollt.

Abbildung 4.18: Beispiel: Statischer Knoten mit Korrektur beim Rollup

ähnlich aus wie im Fall der zeitscheibenüberspringenden Kanten, die in die Zeitscheiben eingebettet wurden, so daß auf sie verzichtet wird. Bsp

4.1.3 Nachinstantiierung von Netzstrukturen

In READY-Netzen werden manche Netzstrukturen von Zeitscheiben erst im Nachhinein etabliert. Zum einen, weil man nicht weiß, welche Netzstruktur instantiiert werden soll, d.h. man hat ein gewisses Grundschema \mathcal{A} , das sich je nach Nachinstantiierung zum Schema $\mathcal{A}_1, \dots, \mathcal{A}_a$ entwickelt. Zum anderen, weil die entsprechende Netzstruktur vorläufig nicht benötigt wird, und man hofft, daß man sie im weiteren Zeitverlauf nicht brauchen wird. Dadurch läßt sich dann eventuell Rechenzeit und Speicherplatz einsparen.

In diesem Abschnitt wird nun beschrieben, warum die Nachinstantiierung von Netzstrukturen für Dynamische Bayessche Netze und insbesondere für die Rollup-Verfahren aus Kapitel 3 problematisch sind. Desweiteren wird gezeigt, wie sich das Problem der Nachinstantiierung von Netzstrukturen lösen läßt, indem alle möglichen Netzstrukturen, die nachinstantiiert werden könnten, bei der Etablierung der Zeitscheibe gleichfalls erzeugt werden.

Wie im Fall der zeitscheibenüberspringenden Kanten sei wiederum ein Dynamisches Bayessches Netz \mathcal{DBN} zum Zeitpunkt $t - 1$ gegeben, das bisher noch nicht aufgerollt wurde.

In Abbildung 4.19(a) ist ein Schnappschuß des Dynamischen Bayesschen Netzes zum Zeitpunkt $t - 1$ zu sehen (Die für den Sachverhalt uninteressanten Zeitscheiben sind nicht dargestellt.).

Ein Vergleich der Abbildung 4.19(a) mit der Abbildung 4.1(a) zeigt, daß sowohl bei den zeitscheibenüberspringenden Kanten wie auch bei der Nachinstantiierung von Netzstrukturen von derselben Ausgangssituation ausgegangen wird.

Zum Zeitpunkt t wird nun ein Schema als Zeitscheibe t instantiiert, das ab dem Zeitpunkt $t - a$ nicht mehr instantiiert wurde. Dieses Schema wird in Abbildung 4.19(b) mit \mathcal{D} bezeichnet. Im Schema \mathcal{D} wird nun auch ein Knoten instantiiert, der eine Zufallsvariable repräsentiert, die auch oder nur von einer Zufallsvariablen beeinflusst wird, deren entsprechender Knoten eigentlich im Schema \mathcal{C} enthalten ist, aber noch nicht instantiiert wurde, weil das Teilnetz noch nicht notwendig war. Es findet eine rekursive Nachinstantiierung von Teilnetzen der einzelnen Zeitscheiben statt, bis es keine nachinstantiierten Knoten mehr gibt, die eine Nachinstantiierung von Netzstrukturen in Vorgängerzeitscheiben notwendig machen. Es wird vorausgesetzt, daß durch die Nachinstantiierung keine zeitscheibenüberspringenden Kanten induziert werden⁸. Sei nun die Zeitscheibe $t - a$, in der das Schema \mathcal{A} instantiiert wurde, die Zeitscheibe, in der keine Nachinstantiierung mehr notwendig wird. Die Zeitscheiben von $t - a + 1$ bis $t - 1$ werden um Netzstrukturen erweitert und die dazwischenliegenden Zustandsentwicklungsmodell ändern sich entsprechend. Dies ist in der Abbildung 4.19(b) jeweils durch einen gerichteten Pfeil von Zeitscheibe h zu Zeitscheibe $h + 1$ ($t - a \leq h \leq t - 1$) dargestellt. Die veränderten Zustandsentwicklungsmodelle sind gestrichelt eingezeichnet.

⁸Dies kann vorausgesetzt werden, da ansonsten das Lösungsverfahren aus 4.1.1 zum Einsatz kommt.

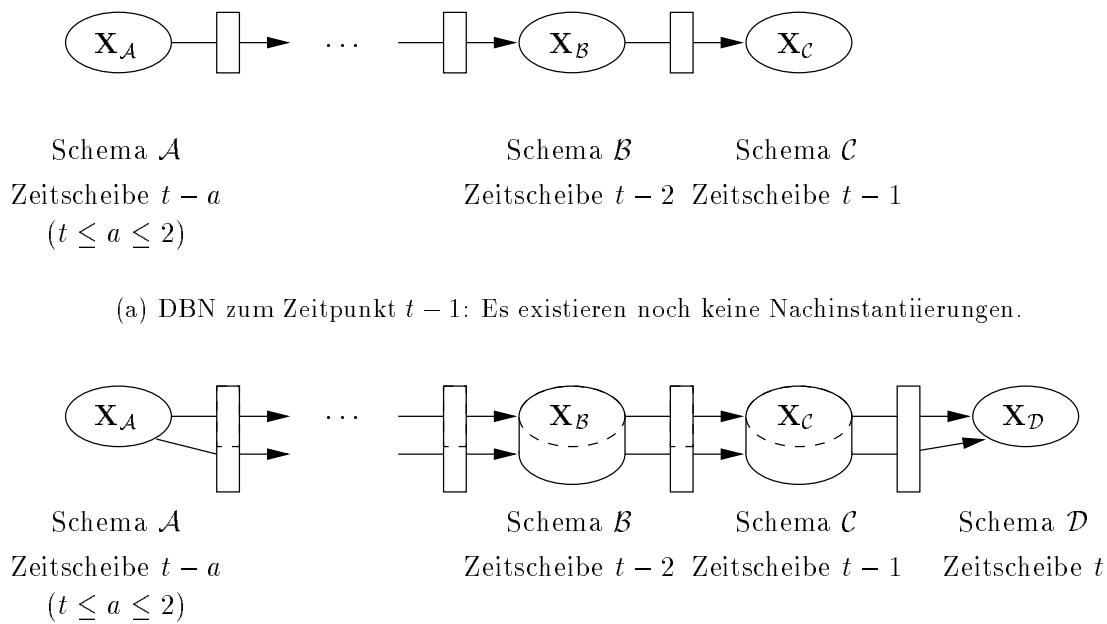
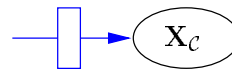


Abbildung 4.19: Nachinstantiierung von Netzstrukturen in den Vorgängerzeitscheiben

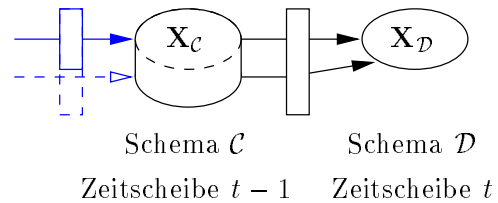
Werden keine Zeitscheiben aufgerollt, so muß für das Dynamische Bayessche Netz der zugehörige Junction Tree neu berechnet werden, aber es ergeben sich keine weiteren Probleme (vorausgesetzt, daß sich der zugehörige Junction Tree des Dynamischen Bayesschen Netzes mit den ψ -Tabellen aller Cliques noch bestimmen läßt.). Die Situation stellt sich wie bei den zeitscheibenüberspringenden Kanten oder Statischen Knoten auch schon allerdings anders dar, wenn der Rollup immer dann durchgeführt wird, wenn eine neue Zeitscheibe etabliert wurde (siehe Kapitel 3). In Abbildung 4.20(a) ist das Dynamische Bayessche Netz zum Zeitpunkt $t - 1$ zu sehen, nachdem die Zeitscheibe $t - 2$ aufgerollt wurde. Bis zum Zeitpunkt $t - 2$ wurden die anderen alten Zeitscheiben $h \leq t - 2$ jeweils dann aufgerollt, wenn die neue Zeitscheibe $h + 1$ etabliert war. Zu jedem Zeitpunkt h , $0 \leq h \leq t - 2$ waren somit immer maximal nur zwei Zeitscheiben des Dynamischen Bayesschen Netzes repräsentiert.

Wird nun zum Zeitpunkt t das Schema \mathcal{D} als Zeitscheibe t instantiiert, so wird in der Zeitscheibe $t - 1$ eine Netzstruktur nachinstantiiert. Diese würde eigentlich eine Netzstruktur in Zeitscheibe $t - 2$ nachinstantiiieren, die aber nicht mehr vorhanden ist. Die Nachinstantiierung bricht also schon in der Zeitscheibe $t - 1$ ab. Dies ist in der Abbildung 4.20(b) durch eine gestrichelte Kante dargestellt. Die Nachinstantiierungen in den Zeitscheiben h ($t - a < h < t - 2$) lassen sich nicht mehr modellieren. In Abbildung 4.20(a) und (b) sind die aufgerollten Zeitscheiben gestrichelt eingezeichnet.

Damit ein Rollup mit den Mitteln aus Kapitel 3 möglich ist, muß also eine mögliche Nachinstantiierung in einer Zeitscheibe verhindert werden, indem bei der Instantiierung der Zeitscheibe die gesamte Netzstruktur instantiiert wird. In Abbildung 4.21 ist dargestellt, daß die Schemata und die Zustandsentwicklungsmodelle der Zeitscheiben

Schema \mathcal{C} Zeitscheibe $t - 1$

(a) DBN nach dem Rollup zum Zeitpunkt $t - 1$: Es existieren noch keine Nachinstantiierungen.



(b) DBN zum Zeitpunkt t : Rückwirkend wird in der vorhergehenden Zeitscheibe eine Netzstruktur nachinstantiiert. Da die Zeitscheibe $t - 2$ schon aufgerollt ist, bricht hier die Nachinstantiierung ab.

Abbildung 4.20: Nachinstantiierung von Netzstrukturen beim Rollup ohne Korrektur: Es wird eine Netzstruktur in der Vorgängerzeitscheibe nachinstantiiert. Trotzdem werden nicht alle Abhängigkeiten modelliert.

immer vollständig vorhanden sind und nicht mehr nachinstantiiert werden müssen. Man sieht, daß temporale Kanten hinzukommen. Jetzt kann der Rollup wie gewohnt mit den Mitteln aus dem Kapitel 3 durchgeführt werden.

Beispiel 4.1.3 (Nachinstantiierung)

Wie den obigen Ausführungen zu entnehmen ist, muß dafür gesorgt werden, daß es keine Nachinstantiierungen gibt. Im Zweifelsfall instantiiert man also lieber mehr als weniger, so daß die Rollupverfahren aus dem Kapitel 3 angewendet werden können.

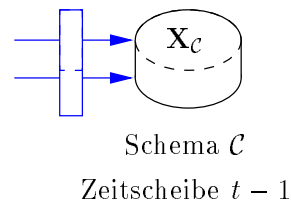
Sei dazu als Beispiel das Netz in Abbildung 4.22(a) gegeben, von dem die Zeitscheiben von $t - 3$ bis $t - 1$ dargestellt sind.

Zum Zeitpunkt $t - 1$ sind die Zeitscheiben $t - 1$ und $t - 2$ ausreichend instantiiert.

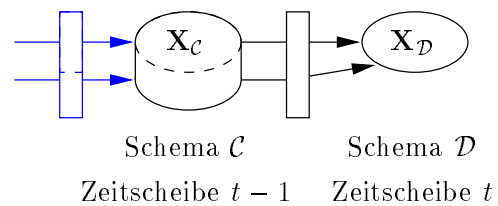
Jetzt wird ein Schema als Zeitscheibe t instantiiert, das Nachinstantiierungen in Zeitscheiben induziert. In der Abbildung 4.22(b) sind dies Nachinstantiierungen in den Zeitscheiben $t - 2$ und $t - 1$. Die involvierten Knoten sind in der Abbildung 4.22(b) grau-schraffiert und die Kanten gestrichelt eingezeichnet.

Die Instantiierung eines Bayesschen Netzes passiert in READY rekursiv. Es wird eine gewisse Menge von Knoten instantiiert, die die Instantiierung ihrer Elternknoten nach sich ziehen, die wiederum die Instantiierung ihrer Elternknoten veranlassen und so fort. Eine Nachinstantiierung wird zum Beispiel dann angeworfen, wenn ein Knoten instantiiert wird, der durch einen Knoten beeinflusst wird, der in der Vorgängerzeitscheibe modelliert ist, aber noch nicht instantiiert wurde.

Wird der Rollup regelmäßig durchgeführt, so daß immer nur maximal zwei Zeit-



(a) DBN nach dem Rollup zum Zeitpunkt $t - 1$: Die potentielle Nachinstantiierung ist schon vorhanden.



(b) DBN zum Zeitpunkt t : Es werden anders wie in Abbildung 4.20(b) keine Netzstrukturen nachinstantiiert, da die entsprechenden Netzstrukturen schon im Schema enthalten sind.

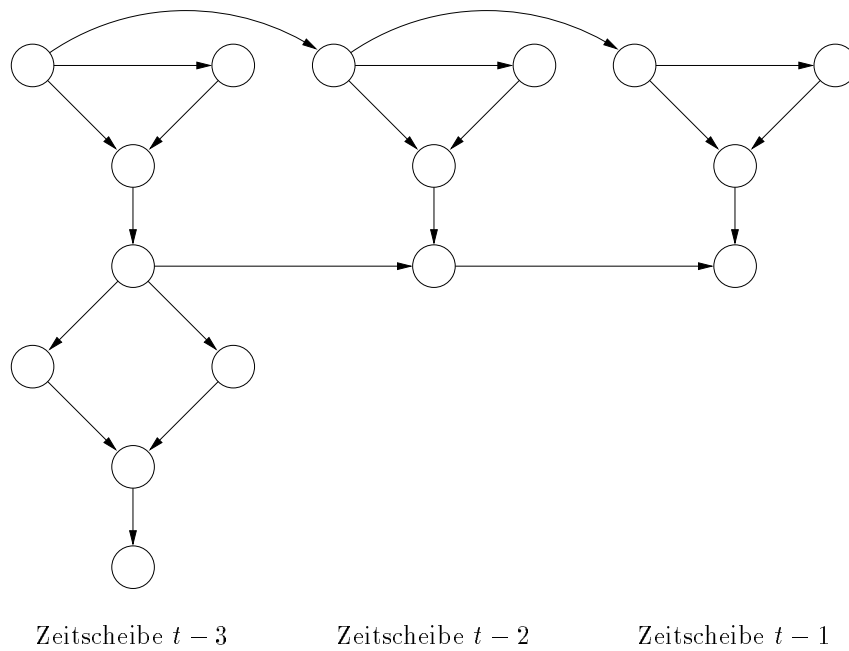
Abbildung 4.21: Nachinstantiierung von Netzstrukturen mit Korrektur: Obwohl keine Netzstrukturen nachinstantiiert werden, sind alle Abhängigkeiten modelliert.

scheiben des Dynamischen Bayesschen Netzes repräsentiert sind, so treten Probleme auf.

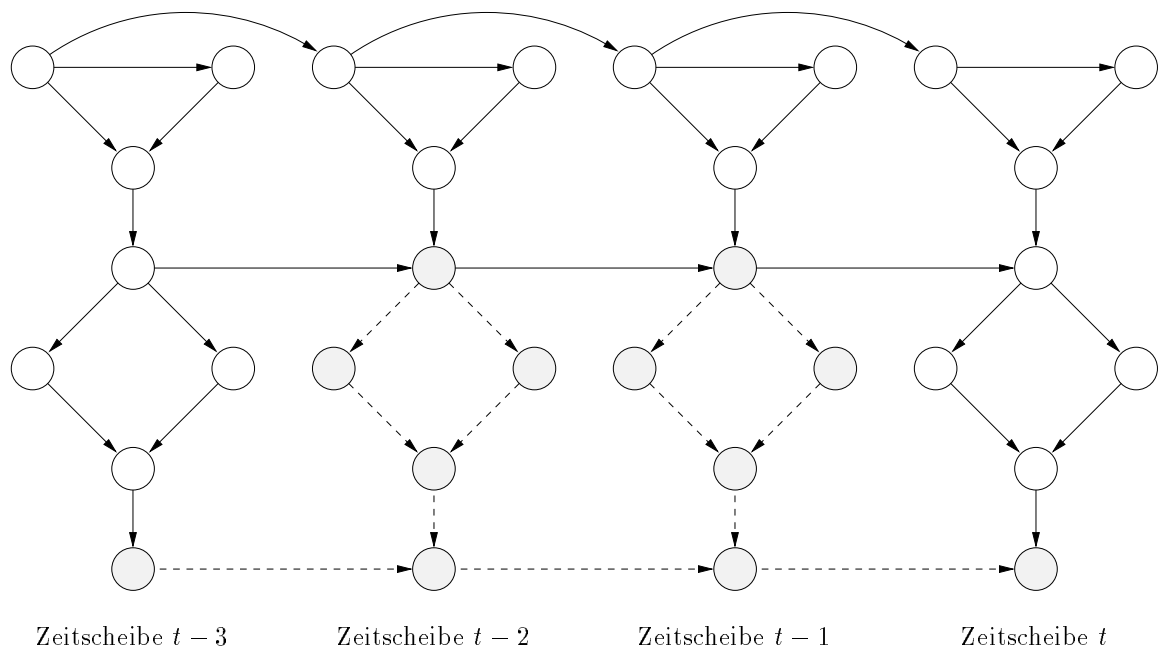
In den beiden Abbildungen 4.23(a) und (b) sieht man, wie ein Rollup bei Nachinstantiierungen ohne Korrektur aussieht. In Abbildung 4.23(a) wurde die Zeitscheibe $t - 3$ aufgerollt und die neue Zeitscheibe $t - 1$ etabliert. Das Zustandsentwicklungsmodell berücksichtigt nicht die Nachinstantiierung, wie es später die korrigierte Version tut (siehe Abbildungen 4.24 und 4.25(a) und (b)). Jetzt wird die Zeitscheibe $t - 2$ aufgerollt und die Zeitscheibe t etabliert. Durch den Knoten, der in der Zeitscheibe t grau schraffiert dargestellt ist, wird eine Nachinstantiierung in Zeitscheibe $t - 1$ ausgeführt. Die involvierten Knoten und Kanten sind grau schraffiert bzw. gestrichelt dargestellt. In Zeitscheibe $t - 2$ kann keine Nachinstantiierung mehr durchgeführt werden, da diese Zeitscheibe schon aufgerollt ist. Das Zustandsentwicklungsmodell berücksichtigt wiederum die Nachinstantiierung nicht.

Zum Zeitpunkt des Rollups müssen also alle Netzstrukturen vorhanden sein. Irgendwelche Nachinstantiierungen können nicht berücksichtigt werden. Die entsprechenden Zustandsentwicklungsmodelle enthalten somit nie alle notwendigen Informationen.

In der Abbildung 4.24 wird gezeigt, wie das Problem der Nachinstantiierungen behandelt wird. Durch die gesamte Instantiierung eines Schemas als Zeitscheibe (auch der Netzstrukturen, die optional zur Nachinstantiierung vorgesehen sind) kann der Rollup zu jedem Zeitpunkt problemlos ausgeführt werden. Ist jedoch definitiv klar, daß eine Netzstruktur eines Schemas, die optional zur Nachinstantiierung vorgesehen ist, nicht nachinstantiiert wird, so kann auf die Instantiierung dieser Netzstruktur verzichtet

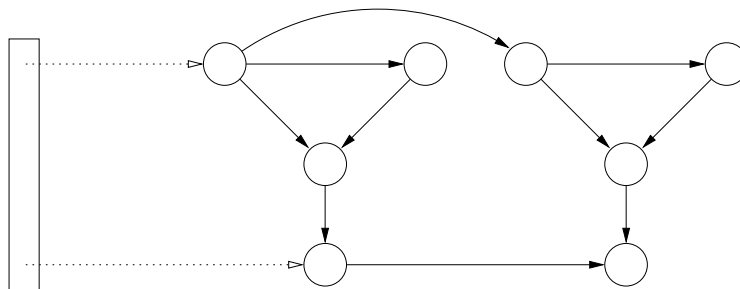


(a) DBN zum Zeitpunkt $t - 1$: Als letztes wurde die Zeitscheibe t instantiiert. Es existieren noch keine Nachinstantiierungen von Netzstrukturen.



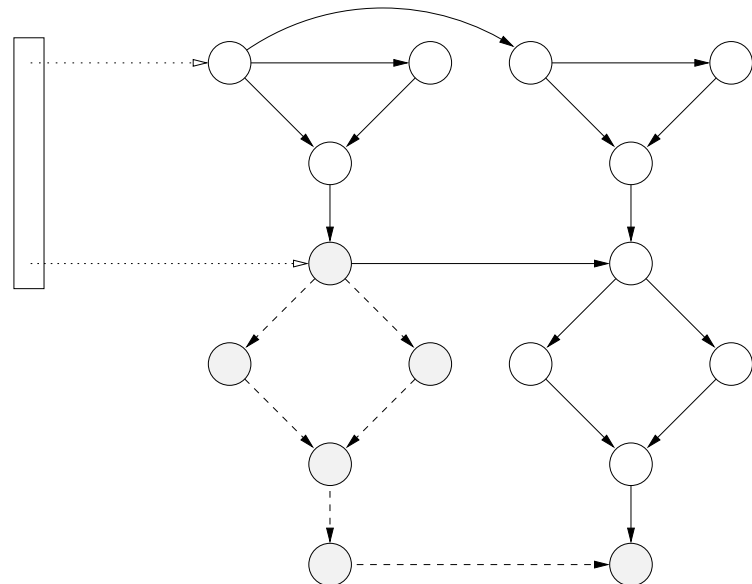
(b) DBN zum Zeitpunkt t : Die Zeitscheibe t induziert eine Reihe von Nachinstantiierungen von Netzstrukturen.

Abbildung 4.22: Beispiel: Nachinstantiierung ohne Korrektur



Zustandsentwicklungsmodell Zeitscheibe $t - 2$ Zeitscheibe $t - 1$

(a) DBN zum Zeitpunkt $t - 1$: Die Zeitscheibe $t - 3$ ist aufgerollt.



Zustandsentwicklungsmodell Zeitscheibe $t - 1$ Zeitscheibe t

(b) DBN zum Zeitpunkt t : Die Zeitscheibe t wurde etabliert und die Zeitscheibe $t - 2$ aufgerollt.

Abbildung 4.23: Beispiel: Nachinstantiierung ohne Korrektur beim Rollup

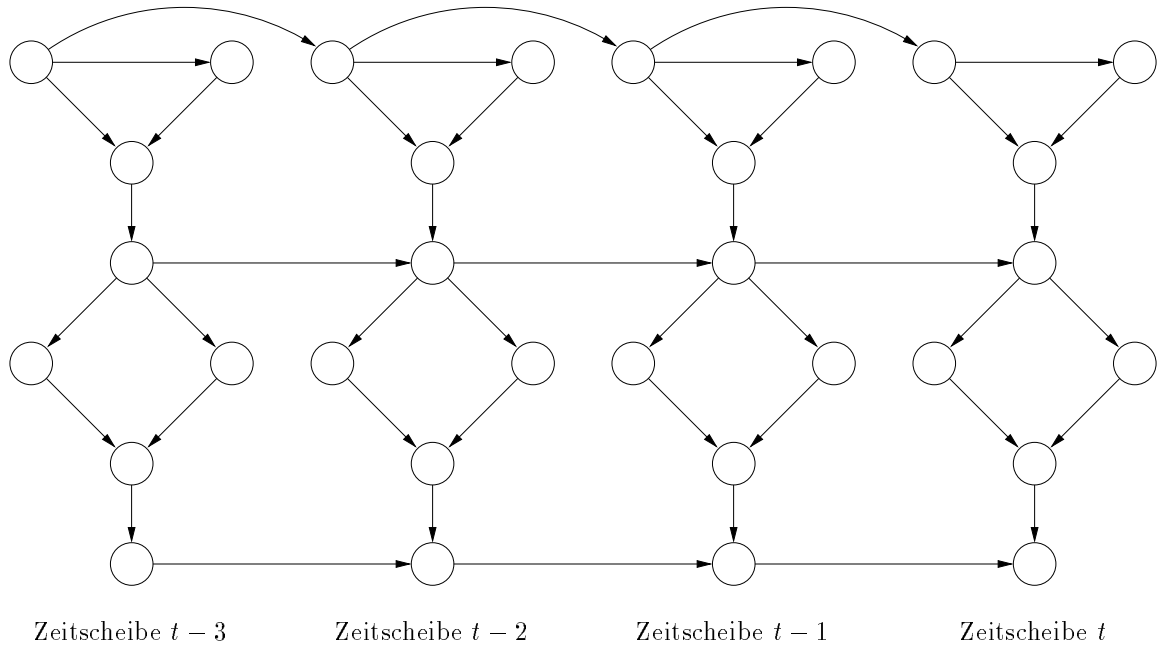


Abbildung 4.24: Beispiel: Nachinstantiierung mit Korrektur

werden.

Somit ist jetzt wieder ein Rollup mit den Mitteln aus Kapitel 3 möglich. Dieser Rollup wird in den Abbildungen 4.25(a) und (b) dargestellt.

In den Abbildungen 4.25(a) und (b) ist der Rollup in der korrigierten Version dargestellt. In Abbildung 4.25(a) wurde die Zeitscheibe $t - 2$ aufgerollt und dann die Zeitscheibe $t - 1$ etabliert. Die Netzstrukturen, die in der unkorrigierten Version erst durch die Zeitscheibe t nachinstantiiert werden, sind schon vorhanden. Das Zustandsentwicklungsmodell berücksichtigt jetzt die gesamte Netzstruktur, was vorher nicht möglich war. Abbildung 4.25(b) wurde die Zeitscheibe $t - 1$ aufgerollt und dann die Zeitscheibe t etabliert. Auch hier berücksichtigt das Zustandsentwicklungsmodell wieder die gesamte Netzstruktur, was im unkorrigierten Fall unmöglich war. **Bsp**

4.2 Zusammenfassung

Es werden die Neuerungen von READY-Netzen gegenüber den Standard Dynamischen Bayesschen Netzen vorgestellt. Diese Neuerungen stellen eine Vereinfachung für den Inferenzalgorithmus oder den Modellierer von Dynamischen Bayesschen Netzen dar. Wenn jedoch ein Rollup durchgeführt wird, so treten massive Probleme auf. Diese Probleme werden jeweils an dem generischen Schema und einem konkreten Beispielnetz dargestellt. Sowohl am generischen Schema als auch am konkreten Beispielnetz wird gezeigt, wie das READY-Netz umgewandelt werden muß, so daß beim Rollup keine Probleme mehr auftreten.

Das Problem der zeitscheibenüberspringenden Kanten wird dadurch behoben, indem solange Zwischenknoten in der betroffenen Kante eingefügt werden, bis die Kante in jeder der Zeitscheiben zu liegen kommt, die sie übersprungen hat. Durch das Einfügen von Zwischenknoten läßt sich auch das Problem der Statischen Knoten lösen,

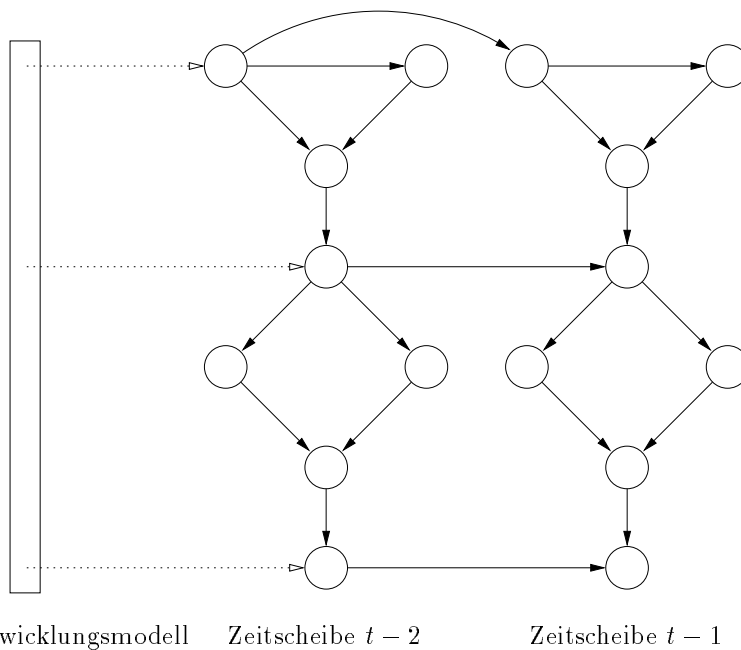
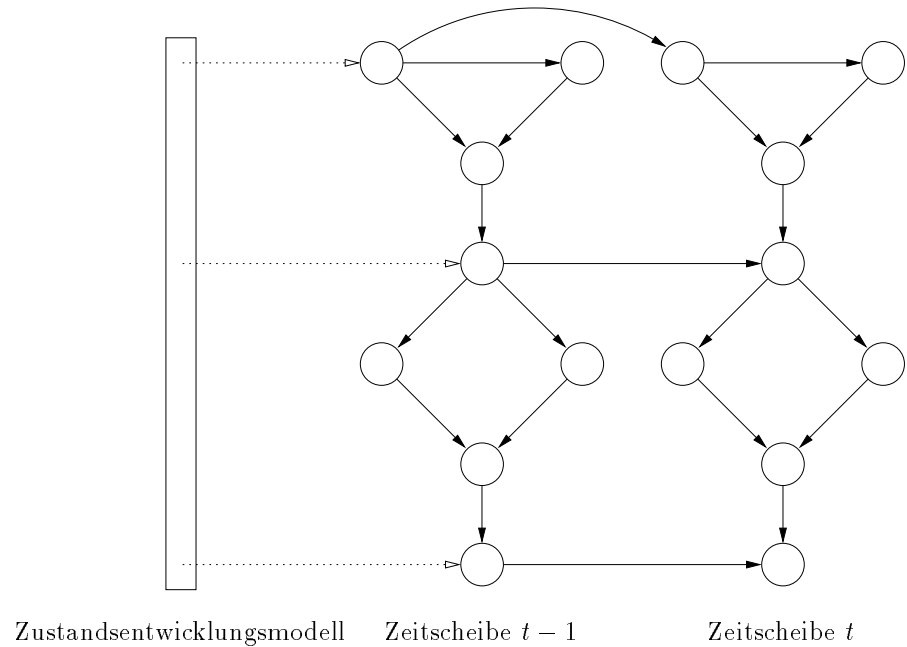
(a) DBN zum Zeitpunkt $t - 1$: Die Zeitscheibe $t - 3$ ist aufgerollt.(b) DBN zum Zeitpunkt t : Die Zeitscheibe t wurde etabliert und die Zeitscheibe $t - 2$ aufgerollt.

Abbildung 4.25: Beispiel: Nachinstantiierung mit Korrektur beim Rollup

das sich auf das Problem der zeitscheibenüberspringenden Kanten zurückführen läßt. Die Nachinstantiierung von Netzstrukturen läßt sich verhindern, indem die entsprechenden Strukturen bei der Etablierung der Zeitscheibe mitinstantiiert werden. Stehen dabei mehrere verschiedene Netzstrukturen zur Auswahl, so sind schlimmstenfalls alle zu erzeugen.

Da erst jetzt der Rollup fehlerfrei anwendbar ist, kann man nicht behaupten, daß durch das Einfügen von Knoten die Komplexität des Dynamischen Bayesschen Netzes steigt. Die Komplexität läßt sich vielmehr erst feststellen, wenn ein Rollup-Verfahren oder Inferenzalgorithmus angewendet wird. Dabei hängt die Komplexität vom verwendeten Rollup-Verfahren oder Inferenzalgorithmus ab. Beim Inferenzalgorithmus wird die Komplexität entscheidend von der Eliminationsreihenfolge der Knoten bestimmt.

Kapitel 5

Zusammenfassung und Ausblick

Zusammenfassung

In dieser Arbeit wird eine kohärente Besprechung verschiedener exakter Rollup-Verfahren sowie ihrer theoretischen Grundlagen vorgenommen. Diese Besprechung kann als Leitfaden für diejenigen verwendet werden, die sich eingehend mit Rollup-Verfahren beschäftigen wollen.

Für Dynamische Bayessche Netze werden mehrere Rollup-Verfahren neu entwickelt bzw. vorgestellt, die auf exakten Inferenzverfahren basieren. Dabei werden für die verschiedenen Verfahren neben der Erklärung der Funktionsweise, d.h. wie der Rollup der einzelnen Verfahren durchgeführt wird, auch die Vor- und Nachteile sowie die Einsatzbereiche behandelt. In einer Tabelle werden die wesentlichen Ergebnisse kurz und übersichtlich aufgelistet.

Es wird gezeigt, wie mit dHUGIN und der Polynom-Propagierung Zeitscheiben konsekutiv, mit der ältesten Zeitscheibe beginnend, aus dem Inferenzprozeß herausgenommen und, beginnend mit der jüngst aus dem Inferenzprozeß herausgenommenen Zeitscheibe, konsekutiv Zeitscheiben wieder in den Inferenzprozeß eingefügt werden können. Dabei wird die Benutzung der Polynom-Propagierung als Grundlage für ein Rollup-Verfahren mit faktorisierten Polynomen in dieser Arbeit erstmalig vorgeschlagen. Es wird zusätzlich gezeigt, wie es mit der Polynom-Propagierung ohne weitere Aufwendungen möglich ist, eine Zeitscheibe t_{neu} vor einer Zeitscheibe t_{alt} , mit $t_{\text{alt}} < t_{\text{neu}}$, aufzurollen, im Gegensatz zum normalen Rollup, bei dem immer nur die älteste Zeitscheibe aufgerollt werden kann.

Für Dynamische Bayessche Netze, die die Bedingungen nicht erfüllen, die die Rollup-Verfahren an sie stellen, um angewendet werden zu können, wird am Beispiel der READY-Netze gezeigt, wie sie erfolgreich in Standard Dynamische Bayessche Netze umgeformt werden können. Dazu gehört auch zu erkennen, welche Eigenschaften einen fehlerfreien Rollup verhindern.

Ausblick

Momentan muß sich der Anwender das passende Rollup-Verfahren für ein Dynamisches Bayessches Netz noch selber aussuchen. Es wäre interessant, ein System zu haben, das dem Benutzer hilft, aus einer Reihe von Rollup-Verfahren das jeweils geeignete Verfah-

ren für die Situation zu wählen. Hierfür müßte eine grafische Darstellung von Junction Trees, wie sie auch schon im Programm *RUBIN* erfolgt, realisiert werden. Noch einen Schritt weiter als das System, das dem Benutzer hilft, das richtige Rollup-Verfahren auszuwählen, wäre ein vollautomatisches System, das alle möglichen Rollup-Verfahren zur Auswahl hat, manche von ihnen bei Bedarf miteinander vereint und automatisch immer das jeweils beste Rollup-Verfahren für die jeweilige Situation anwendet. So könnte man sich vorstellen, daß man automatisch vom approximativen Rollup zum Rollup mit der Polynom-Propagierung wechselt und umgekehrt. Eine offene Frage dabei ist, wie sich ein solcher Wechsel effizient organisieren läßt.

Desgleichen ist es aber auch wichtig, Inferenzverfahren zu entwickeln, die Junction Trees mit großen Cliques effizient handhaben können. Für effiziente Rollup-Verfahren sind effiziente Inferenzverfahren unabdingbar. So ist es wichtig, Inferenzverfahren zu entwickeln, die die momentan bzgl. Speicherplatzverbrauch und Rechenzeitbedarf komplexen Bayesschen Netze effizient handhaben können, so daß diese Bayesschen Netze ihre Komplexität verlieren. Einen Weg in diese Richtung geht mit Einschränkungen die Polynom-Propagierung. Bei der Polynom-Propagierung ist zu untersuchen, wie die Polynome direkt faktorisiert dargestellt werden können, ohne den Zwischenweg zu gehen, daß zuerst das Polynom unfaktorisiert berechnet wird. Eine Hilfe zur direkten Bestimmung von faktorisierten Polynomen sind die Shafer-Shenoy Skripte. Aber auch hier sind noch Erfahrungswerte bzgl. Effizienz und Anwendbarkeit zu sammeln.

Es ist zu erwarten, daß innerhalb der nächsten Jahre die großen Software-Pakete zur Verarbeitung Bayesscher Netze Verfahren zum Rollup von Dynamischen Bayesschen Netzen anbieten werden.

Anhang A

Mehrfach verbundene Bayessche Netze

Im folgenden wird für die Aalborg Architektur und die Shafer-Shenoy Architektur jeweils dasselbe Beispielnetz durchgerechnet. Im Anschluß daran werden für den interessierten Leser weitere Anwendungen der Polynom-Propagierung vorgestellt. Der Abschnitt über das D-Separations-Kriterium rundet diesen Anhang ab.

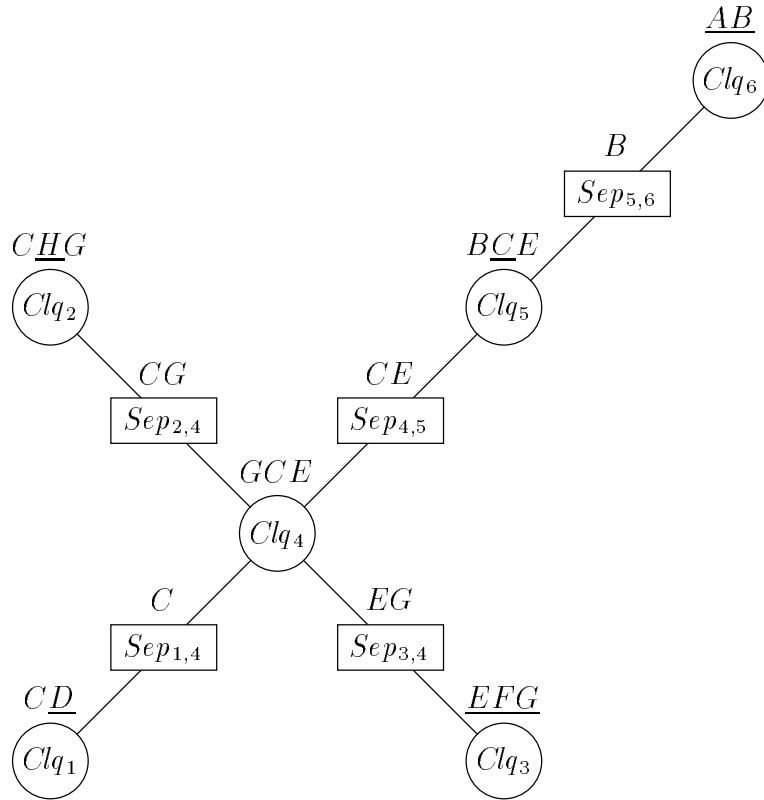
A.1 Aalborg Architektur

In diesem Anhang wird ein Beispiel für die Aalborg Architektur umfassend vorgerechnet. Die einzelnen Schritte der beiden Phasen Collect Evidence und Distribute Evidence werden sowohl bei der Initialisierung wie auch bei der Auffrischung des Junction Trees einmal in einer Liste und auch graphisch dargestellt. Wichtige Zwischenergebnisse der Berechnungen, die bei der Initialisierung und der Auffrischung durchgeführt werden, sind in zwei Tabellen übersichtlich angeordnet. Konkrete Berechnungen mit Zahlenwerten, wie zum Beispiel die Absorption von einer Clique durch eine andere Clique, runden das Beispiel ab.

Beispiel A.1.1 (Asienbesuch (Aalborg))

In Abbildung A.1 ist ein weiterer Junction Tree für das Bayesschen Netz **Asienbesuch** in Abb. 1.2 angegeben.

Bevor am Junction Tree aus Abbildung A.1 nun die beiden Phasen Collect Evidence und Distribute Evidence mit Absorption vorgestellt werden, müssen noch die Knoten des Bayesschen Netzes den Cliques des Junction Trees eindeutig zugeordnet werden. Die Knoten *A* und *B* werden der Clique Clq_6 , der Knoten *C* der Clique Clq_5 , die Knoten *E*, *F* und *G* der Clique Clq_3 , der Knoten *D* der Clique Clq_1 zugeordnet. In Abbildung A.1 sind die Knoten der Clique unterstrichen, die der Clique zugeordnet sind. In den Cliques des Junction Trees stehen nun, wie folgt und auch in Tabelle A.1 dargestellt ist, die folgenden ψ -Tabellen: Die Clique Clq_1 enthält die ψ -Tabelle $T_{\psi(Clq_1)} = T_{1(CD)} \cdot T_{\text{cpt}(D)}$. $T_{\text{cpt}(D)}$ repräsentiert dabei die bedingten Wahrscheinlichkeiten des Knotens *D*, nämlich $P(D | C)$. In Clique Clq_2 steht die ψ -Tabelle $T_{\psi(Clq_2)} = T_{1(CG H)} \cdot T_{\text{cpt}(H)}$, in Clique Clq_3 die ψ -Tabelle $T_{\psi(Clq_3)} = T_{1(EFG)} \cdot T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)}$, in der Clique Clq_4 –ihr werden keine Knoten zugeteilt– die ψ -Tabelle $T_{\psi(Clq_4)} = T_{1(CEG)}$, in Clique Clq_5

Abbildung A.1: weiteres Beispiel eines Junction Trees für das Beispielnetz **Asienbesuch**

$T_{\psi(Clq_5)} = T_{1(BCE)} \cdot T_{\text{cpt}(C)}$ und in Clique Clq_6 $T_{\psi(Clq_6)} = T_{1(AB)} \cdot T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)}$. In den Separatoren des Junction Trees stehen Wahrscheinlichkeitentabellen über den Knoten, die in den Separatoren enthalten sind. Alle Einträge in den Tabellen sind dabei 1.

Clique	zugeordnete Knoten	Berechnungsvorschrift der ψ -Tabelle
$Clq_1 = \{\underline{D}, C\}$	D	$T_{\psi(Clq_1)} = T_{1(CD)} \cdot T_{\text{cpt}(D)}$
$Clq_2 = \{C, G, \underline{H}\}$	H	$T_{\psi(Clq_2)} = T_{1(CG H)} \cdot T_{\text{cpt}(H)}$
$Clq_3 = \{\underline{E}, \underline{F}, \underline{G}\}$	E, F, G	$T_{\psi(Clq_3)} = T_{1(EFG)} \cdot T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)}$
$Clq_4 = \{E, C, G\}$	-	$T_{\psi(Clq_4)} = T_{1(CEG)} = 1^{ E \times C \times G }$
$Clq_5 = \{B, \underline{C}, E\}$	C	$T_{\psi(Clq_5)} = T_{1(BCE)} \cdot T_{\text{cpt}(C)}$
$Clq_6 = \{\underline{A}, \underline{B}\}$	A, B	$T_{\psi(Clq_6)} = T_{1(AB)} \cdot T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)}$

Tabelle A.1: Aalborg: ψ -Tabellen der Cliques nach der Konstruktion des Junction Trees

Initialisierung. Wählt man als Wurzelknoten die Clique Clq_6 , dann sieht *Collect Evidence* bei der Initialisierung des Junction Trees wie folgt aus (siehe auch Abbildung A.2(a)):

1. Die Cliques Clq_1 , Clq_2 und Clq_3 schicken ihre Nachricht an die Clique Clq_4 (Man sagt auch: Die Clique Clq_4 absorbiert von den Cliques Clq_1 , Clq_2 und Clq_3).
2. Jetzt schickt Clique Clq_4 ihre Nachricht an Clique Clq_5 .

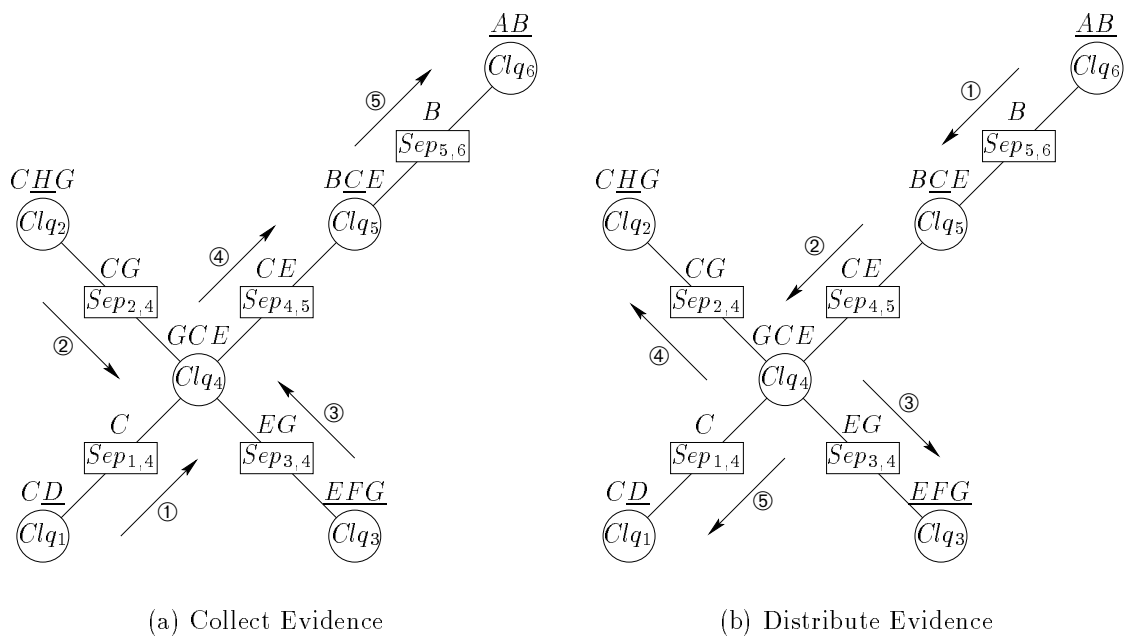


Abbildung A.2: Aalborg: Initialisierung eines Junction Trees

3. Darauf schickt Clique Clq_5 ihre Nachricht an Clique Clq_6 und Collect Evidence terminiert.

Daraufhin erfolgt *Distribute Evidence* (siehe auch Abbildung A.2(b)):

1. Die Clique Clq_6 schickt nun ihrerseits ihre Nachricht an Clique Clq_5 .
2. Die Clique Clq_5 schickt ihre Nachricht an Clique Clq_4 .
3. Zum Abschluß verschickt Clique Clq_4 ihre Nachrichten an die Cliques Clq_1 , Clq_2 und Clq_3 . Die Nachrichten an die verschiedenen Cliques sind dabei unterschiedlich. Distribute Evidence terminiert.

In Tabelle A.2 sind die ψ -Tabellen¹ der Cliques und Separatoren im Verlauf der Initialisierung dargestellt. Die Tabellen, die in den verschiedenen Phasen nicht neu berechnet werden, sind aus Gründen der Übersichtlichkeit nicht dargestellt. Für die Reihenfolge der Berechnung wird auf die Phasen *Collect Evidence* und *Distribute Evidence* in Abbildung A.2 verwiesen. Die eingekreisten Zahlen erleichtern dabei die Zuordnung.

Exemplarisch wird nun gezeigt, wie sich die ψ -Tabelle der Clique Clq_4 in der Phase *Distribute Evidence* berechnet:

1. Im Anfang² bestimmt sich die ψ -Tabelle der Clique Clq_4 zu $T_{\psi(Clq_4)}$:

$$T_{\psi(Clq_4)} = \begin{bmatrix} & & c_1 & c_2 \\ e_1 & g_1 & 0.0315 & 0.0315 \\ & g_2 & 0.0235 & 0.0235 \\ e_2 & g_1 & 0.4185 & 0.4185 \\ & g_2 & 0.5265 & 0.5265 \end{bmatrix} .$$

¹Damit eine ψ -Tabelle die gemeinsamen Wahrscheinlichkeiten der Knoten in dieser Tabelle angibt, muß die Tabelle normalisiert werden.

²nach Collect Evidence

ψ -Tab.	Wert der ψ -Tabelle bei		
	Zuordnung	Collect Evidence	Distribute Evidence
$T_{\psi}(Clq_1)$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline d_1 & 0.98 & 0.05 \\ d_2 & 0.02 & 0.95 \end{array} \right]$		⑤ Absorption: $\left[\begin{array}{cc} 0.0635 & 0.0468 \\ 0.0013 & 0.8884 \end{array} \right]$
$T_{\psi}(Sep_{1,4})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline 1 & 1 & 1 \end{array} \right]$	① Marginalisation: $\left[\begin{array}{cc} 1 & 1 \end{array} \right]$	⑤ Marginalisation: $\left[\begin{array}{cc} 0.0648 & 0.9352 \end{array} \right]$
$T_{\psi}(Clq_2)$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline h_1 & g_1 & 0.9 & 0.8 \\ & g_2 & 0.7 & 0.1 \\ h_2 & g_1 & 0.1 & 0.2 \\ & g_2 & 0.3 & 0.9 \end{array} \right]$		④ Absorption: $\left[\begin{array}{cc} 0.0323 & 0.3313 \\ 0.0203 & 0.0521 \\ 0.0036 & 0.0828 \\ 0.0087 & 0.4689 \end{array} \right]$
$T_{\psi}(Sep_{2,4})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline g_1 & 1 & 1 \\ g_2 & 1 & 1 \end{array} \right]$	② Marginalisation: $\left[\begin{array}{cc} 1 & 1 \\ 1 & 1 \end{array} \right]$	④ Marginalisation: $\left[\begin{array}{cc} 0.0359 & 0.4141 \\ 0.0290 & 0.5210 \end{array} \right]$
$T_{\psi}(Clq_3)$	$\left[\begin{array}{c cc} & e_1 & e_2 \\ \hline f_1 & g_1 & 0.03 & 0.27 \\ & g_2 & 0.02 & 0.18 \\ f_2 & g_1 & 0.0015 & 0.1485 \\ & g_2 & 0.0035 & 0.3465 \end{array} \right]$		③ Absorption: $\left[\begin{array}{cc} 0.03 & 0.27 \\ 0.02 & 0.18 \\ 0.0015 & 0.1485 \\ 0.0035 & 0.3465 \end{array} \right]$
$T_{\psi}(Sep_{3,4})$	$\left[\begin{array}{c cc} & e_1 & e_2 \\ \hline g_1 & 1 & 1 \\ g_2 & 1 & 1 \end{array} \right]$	③ Marginalisation: $\left[\begin{array}{cc} 0.0315 & 0.4185 \\ 0.0235 & 0.5265 \end{array} \right]$	③ Marginalisation: $\left[\begin{array}{cc} 0.0315 & 0.4185 \\ 0.0235 & 0.5265 \end{array} \right]$
$T_{\psi}(Clq_4)$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline e_1 & g_1 & 1 & 1 \\ & g_2 & 1 & 1 \\ e_2 & g_1 & 1 & 1 \\ & g_2 & 1 & 1 \end{array} \right]$	① ② ③ Absorpt.: $\left[\begin{array}{cc} 0.0315 & 0.0315 \\ 0.0235 & 0.0235 \\ 0.4185 & 0.4185 \\ 0.5265 & 0.5265 \end{array} \right]$	② Absorption: $\left[\begin{array}{cc} 0.0315 & 0 \\ 0.0235 & 0 \\ 0.0044 & 0.4141 \\ 0.0055 & 0.5210 \end{array} \right]$
$T_{\psi}(Sep_{4,5})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline e_1 & 1 & 1 \\ e_2 & 1 & 1 \end{array} \right]$	④ Marginalisation: $\left[\begin{array}{cc} 0.055 & 0.055 \\ 0.945 & 0.945 \end{array} \right]$	② Marginalisation: $\left[\begin{array}{cc} 0.055 & 0 \\ 0.0098 & 0.9352 \end{array} \right]$
$T_{\psi}(Clq_5)$	$\left[\begin{array}{c cc} & b_1 & b_2 \\ \hline c_1 & e_1 & 1 & 1 \\ & e_2 & 1 & 0 \\ c_2 & e_1 & 0 & 0 \\ & e_2 & 0 & 1 \end{array} \right]$	④ Absorption: $\left[\begin{array}{cc} 0.055 & 0.055 \\ 0.945 & 0 \\ 0 & 0 \\ 0 & 0.945 \end{array} \right]$	① Absorption: $\left[\begin{array}{cc} 0.0006 & 0.0544 \\ 0.0098 & 0 \\ 0 & 0 \\ 0 & 0.9352 \end{array} \right]$
$T_{\psi}(Sep_{5,6})$	$\left[\begin{array}{c cc} & b_1 & b_2 \\ \hline 1 & 1 & 1 \end{array} \right]$	⑤ Marginalisation: $\left[\begin{array}{cc} 1 & 1 \end{array} \right]$	① Marginalisation: $\left[\begin{array}{cc} 0.0104 & 0.9896 \end{array} \right]$
$T_{\psi}(Clq_6)$	$\left[\begin{array}{c cc} & a_1 & a_2 \\ \hline b_1 & 0.0005 & 0.0099 \\ b_2 & 0.0095 & 0.9801 \end{array} \right]$	⑤ Absorption: $\left[\begin{array}{cc} a_1 & a_2 \\ 0.0005 & 0.0099 \\ 0.0095 & 0.9801 \end{array} \right]$	

Tabelle A.2: Aalborg: ψ -Tabellen bei der Initialisierung. Es sind noch keine Knoten instantiiert.

ψ -Tab.	Wert der ψ -Tabelle bei		
	Instantiierung	Collect Evidence	Distribute Evidence
$T_{\psi}(Cl_{q_1})$	Hypothese d_2 : $\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline d_1 & 0 & 0 \\ d_2 & 0.0013 & 0.8884 \end{array} \right]$		⑤ Absorption: $\left[\begin{array}{cc} 0 & 0 \\ \hline 2.045-05 & 8.528-03 \end{array} \right]$
$T_{\psi}(Sep_{1,4})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline & 0.0648 & 0.9352 \end{array} \right]$	① Marginalisation: $\left[\begin{array}{cc} & \\ \hline 1.296-03 & 8.884-01 \end{array} \right]$	⑤ Marginalisation: $\left[\begin{array}{cc} & \\ \hline 2.045-05 & 8.528-03 \end{array} \right]$
$T_{\psi}(Cl_{q_2})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline h_1 & g_1 & 0.0323 & 0.3313 \\ & g_2 & 0.0203 & 0.0521 \\ h_2 & g_1 & 0.0036 & 0.0828 \\ & g_2 & 0.0087 & 0.4689 \end{array} \right]$		④ Absorption: $\left[\begin{array}{cc} 9.436-06 & 3.021-03 \\ \hline 6.975-06 & 4.751-03 \\ 1.048-06 & 7.553-03 \\ \hline 2.989-06 & 4.276-03 \end{array} \right]$
$T_{\psi}(Sep_{2,4})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline g_1 & 0.0359 & 0.4141 \\ g_2 & 0.0290 & 0.5210 \end{array} \right]$	nicht neu berechnet: $\left[\begin{array}{cc} 3.585-02 & 4.141-01 \\ \hline 2.897-02 & 5.210-01 \end{array} \right]$	④ Marginalisation: $\left[\begin{array}{cc} 1.048-05 & 3.776-03 \\ \hline 9.965-06 & 4.751-03 \end{array} \right]$
$T_{\psi}(Cl_{q_3})$	$\left[\begin{array}{c cc} & e_1 & e_2 \\ \hline f_1 & g_1 & 0.03 & 0.27 \\ & g_2 & 0.02 & 0.18 \\ f_2 & g_1 & 0.0015 & 0.1485 \\ & g_2 & 0.0035 & 0.3465 \end{array} \right]$		③ Absorption: $\left[\begin{array}{cc} 0.006-03 & 2.439-03 \\ \hline 0.004-03 & 1.626-03 \\ 0.003-04 & 1.341-03 \\ \hline 0.007-04 & 3.130-03 \end{array} \right]$
$T_{\psi}(Sep_{3,4})$	$\left[\begin{array}{c cc} & e_1 & e_2 \\ \hline g_1 & 0.0315 & 0.4185 \\ g_2 & 0.0235 & 0.5265 \end{array} \right]$	nicht neu berechnet: $\left[\begin{array}{cc} 0.0315 & 0.4185 \\ \hline 0.0235 & 0.5265 \end{array} \right]$	③ Marginalisation: $\left[\begin{array}{cc} 0.063-04 & 3.781-03 \\ \hline 0.047-04 & 4.756-03 \end{array} \right]$
$T_{\psi}(Cl_{q_4})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline e_1 & g_1 & 0.0315 & 0 \\ & g_2 & 0.0235 & 0 \\ e_2 & g_1 & 0.0044 & 0.4141 \\ & g_2 & 0.0055 & 0.5210 \end{array} \right]$	① Absorption: $\left[\begin{array}{cc} 0.00063 & 0 \\ \hline 0.00047 & 0 \\ 8.705-05 & 3.934-01 \\ \hline 1.095-05 & 4.950-01 \end{array} \right]$	② Absorption: $\left[\begin{array}{cc} 0.063-04 & 0 \\ \hline 0.047-04 & 0 \\ 4.185-06 & 3.777-03 \\ \hline 5.265-06 & 4.752-03 \end{array} \right]$
$T_{\psi}(Sep_{4,5})$	$\left[\begin{array}{c cc} & c_1 & c_2 \\ \hline e_1 & 0.055 & 0 \\ e_2 & 0.0098 & 0.9352 \end{array} \right]$	② Marginalisation: $\left[\begin{array}{cc} 0.0011 & 0 \\ \hline 1.966-04 & 8.884-01 \end{array} \right]$	② Marginalisation: $\left[\begin{array}{cc} 0.011-03 & 0 \\ \hline 0.945-05 & 8.529-03 \end{array} \right]$
$T_{\psi}(Cl_{q_5})$	$\left[\begin{array}{c cc} & b_1 & b_2 \\ \hline c_1 & e_1 & 0.0006 & 0.0544 \\ & e_2 & 0.0098 & 0 \\ c_2 & e_1 & 0 & 0 \\ & e_2 & 0 & 0.9352 \end{array} \right]$	② Absorption: $\left[\begin{array}{cc} 1.144-05 & 1.089-03 \\ \hline 1.966-04 & 0 \\ 0 & 0 \\ \hline 0 & 8.884-01 \end{array} \right]$	① Absorption: $\left[\begin{array}{cc} 0.055-05 & 1.045-05 \\ \hline 0.945-05 & 0 \\ 0 & 0 \\ \hline 0 & 8.529-03 \end{array} \right]$
$T_{\psi}(Sep_{5,6})$	$\left[\begin{array}{c cc} & b_1 & b_2 \\ \hline & 0.0104 & 0.9896 \end{array} \right]$	③ Marginalisation: $\left[\begin{array}{cc} & \\ \hline 0.208-03 & 8.895-01 \end{array} \right]$	① Marginalisation: $\left[\begin{array}{cc} & \\ \hline 0.001-02 & 8.539-03 \end{array} \right]$
$T_{\psi}(Cl_{q_6})$	Hypothese a_1 : $\left[\begin{array}{c cc} & a_1 & a_2 \\ \hline b_1 & 0.0005 & 0 \\ b_2 & 0.0095 & 0 \end{array} \right]$	③ Absorption: $\left[\begin{array}{cc} 0.001-02 & 0 \\ \hline 8.539-03 & 0 \end{array} \right]$	

Tabelle A.3: Aalborg: ψ -Tabellen bei der Auffrischung. Es wurden die Knoten A und D instantiiert. (Der Ausdruck $0.006 - 03$ steht zum Beispiel für die Zahl $0.006 \cdot 10^{-3}$.)

2. Die Clique Clq_4 erhält nun durch Absorption die Informationen der Clique Clq_5 wie folgt:

(a) Die Clique Clq_5 enthält die Tabelle $T_{\psi(Clq_5)}$:

$$T_{\psi(Clq_5)} = \left[\begin{array}{c|cc} & b_1 & b_2 \\ \hline c_1 & e_1 & 0.000572 & 0.054428 \\ & e_2 & 0.009828 & 0 \\ c_2 & e_1 & 0 & 0 \\ & e_2 & 0 & 0.935172 \end{array} \right].$$

(b) Es wird die Tabelle $T_{\psi(Sep_{4,5})}^*$ berechnet (⊗ Marginalisation):

$$\begin{aligned} T_{\psi(Sep_{4,5})}^* &= \sum_B \left[\begin{array}{c|cc} & b_1 & b_2 \\ \hline c_1 & e_1 & 0.000572 & 0.054428 \\ & e_2 & 0.009828 & 0 \\ c_2 & e_1 & 0 & 0 \\ & e_2 & 0 & 0.935172 \end{array} \right] = \\ &= \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & 0.000572 + 0.054428 & 0 + 0 \\ e_2 & 0.009828 + 0 & 0 + 0.935172 \end{array} \right] = \\ &= \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & 0.055 & 0 \\ e_2 & 0.009828 & 0.935172 \end{array} \right] = \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & 0.055 & 0 \\ e_2 & 0.0098 & 0.9352 \end{array} \right] \end{aligned}$$

(c) Nun wird die neue ψ -Tabelle des Separators durch die alte ψ -Tabelle des Separators dividiert und der ψ -Tabelle der Clique Clq_4 hinzumultipliziert (⊗ Absorption):

$$\begin{aligned} &\left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & g_1 & 0.0315 & 0.0315 \\ & g_2 & 0.0235 & 0.0235 \\ e_2 & g_1 & 0.4185 & 0.4185 \\ & g_2 & 0.5265 & 0.5265 \end{array} \right] \cdot \frac{\left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & 0.055 & 0 \\ e_2 & 0.009828 & 0.935172 \end{array} \right]}{\left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & 0.055 & 0.055 \\ e_2 & 0.945 & 0.945 \end{array} \right]} = \\ &= \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & g_1 & 0.0315 & 0.0315 \\ & g_2 & 0.0235 & 0.0235 \\ e_2 & g_1 & 0.4185 & 0.4185 \\ & g_2 & 0.5265 & 0.5265 \end{array} \right] \cdot \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & 1 & 0 \\ e_2 & 0.0104 & 0.9896 \end{array} \right] = \\ &= \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & g_1 & 0.0315 & 0 \\ & g_2 & 0.0235 & 0 \\ e_2 & g_1 & 0.0043524 & 0.4141476 \\ & g_2 & 0.0054756 & 0.5210244 \end{array} \right]. \end{aligned}$$

3. Die neue ψ -Tabelle des Separators $Sep_{4,5}$ bestimmt sich zu:

$$T_{\psi(Sep_{4,5})} = \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline e_1 & 0.055 & 0 \\ e_2 & 0.009828 & 0.935172 \end{array} \right].$$

4. Die neue ψ -Tabelle der Clique Clq_4 bestimmt sich zu:

$$T_{\psi(Clq_4)} = \begin{bmatrix} & & c_1 & c_2 \\ e_1 & g_1 & 0.0315 & 0 \\ & g_2 & 0.0235 & 0 \\ e_2 & g_1 & 0.0043524 & 0.4141476 \\ & g_2 & 0.0054756 & 0.5210244 \end{bmatrix}.$$

Auffrischung. Es liegen nun die folgenden Beobachtungen vor: Der Patient hat vor kurzem Asien besucht aber sein Röntgenbefund ist negativ. Als *findings* ergeben sich die folgenden Tabellen $T_{\text{finding}(A)} = \begin{bmatrix} a_1 & a_2 \\ 1 & 0 \end{bmatrix}$ und $T_{\text{finding}(D)} = \begin{bmatrix} d_1 & d_2 \\ 0 & 1 \end{bmatrix}$, die den ψ -Tabellen der Cliquen Clq_6 und Clq_1 hinzu multipliziert werden. In der Spalte *Instanziierung* der Tabelle A.3 sind die neuen Werte der ψ -Tabellen dargestellt. Die Cliquen des Junction Trees sind jetzt nicht mehr *konsistent* zueinander, d.h. die Informationen, die eine Clique über einen Knoten enthält, stimmen nicht mehr mit den Informationen einer anderen Clique überein, in der sich ebenfalls dieser Knoten befindet. Man sagt auch, daß der Junction Tree nicht mehr *kalibriert* ist. Es erfolgt eine *Auffrischung* auf dem Junction Tree. *Collect Evidence* sieht nun wie folgt aus (siehe auch Abbildung A.3(a)):

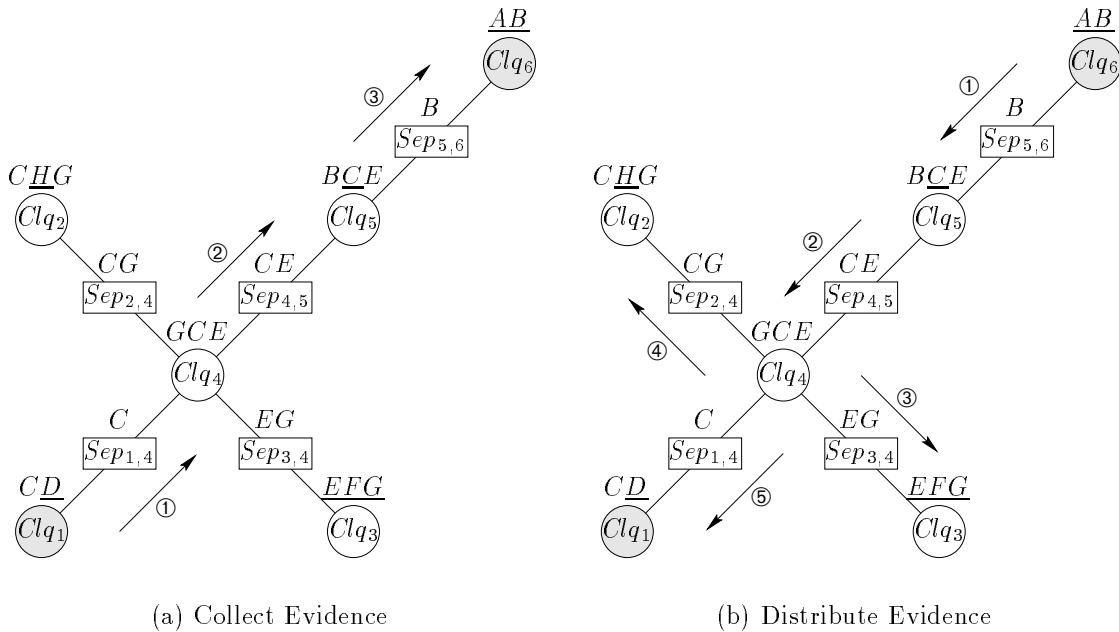


Abbildung A.3: Aalborg: Auffrischung eines Junction Trees nach der Eingabe von neuen Beobachtungen (durch grau schraffierte Kreise gekennzeichnet)

1. Die Cliquen Clq_1 schickt ihre Nachricht an die Clique Clq_4 . Von den Cliquen Clq_2 und Clq_3 wird keine Nachricht an Clique Clq_4 geschickt, da der Junction Tree vor dem Eintragen der Evidenzen schon kalibriert war. Von den Cliquen Clq_2 und Clq_3 würde an die Clique Clq_4 keine veränderte Nachricht geschickt werden, wie in der Tabelle A.3 zu sehen ist.
2. Jetzt schickt Clique Clq_4 ihre Nachricht an Clique Clq_5 .

3. Darauf schickt Clique Clq_5 ihre Nachricht an Clique Clq_6 und Collect Evidence terminiert.

Daraufhin erfolgt *Distribute Evidence* (siehe auch Abbildung A.3(b)):

1. Die Clique Clq_6 schickt nun ihrerseits ihre Nachricht an Clique Clq_5 .
2. Die Clique Clq_5 schickt ihre Nachricht an Clique Clq_4 .
3. Zum Abschluß verschickt Clique Clq_4 ihre Nachrichten an die Cliquen Clq_1 , Clq_2 und Clq_3 . Die Nachrichten an die verschiedenen Cliquen sind dabei unterschiedlich. *Distribute Evidence* terminiert.

Der Junction Tree ist nun wieder kalibriert. Die Ergebnisse für die ψ -Tabellen in den einzelnen Phasen sind der Tabelle A.3 zu entnehmen.

Die BEL-Werte für die einzelnen Knoten des Bayesschen Netzes können durch Marginalisation aus jeder ψ -Tabelle bestimmt werden, in der sich der Knoten befindet. Für die Knoten C und H wird dies hier exemplarisch gemacht:

$$\begin{aligned}
 \text{BEL}(C) &= \alpha \sum_{Clq_1 \setminus \{C\}} T_{\psi(Clq_1)} = \alpha \sum_{Clq_1 \setminus \{C\}} \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline d_1 & 0 & 0 \\ d_2 & 0.00002045 & 0.008528622 \end{array} \right] \\
 &= \alpha \sum_{Sep_{1,4} \setminus \{C\}} T_{\psi(Sep_{1,4})} = \alpha \sum_{Sep_{1,4} \setminus \{C\}} \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline & 0.00002045 & 0.008528622 \end{array} \right] \\
 &= \alpha \begin{pmatrix} 0.00002045 \\ 0.008528622 \end{pmatrix} \\
 \\
 \text{BEL}(H) &= \alpha \sum_{Clq_2 \setminus \{H\}} T_{\psi(Clq_2)} = \alpha \sum_{Clq_2 \setminus \{H\}} \left[\begin{array}{cc|cc} & & c_1 & c_2 \\ \hline h_1 & g_1 & 0.000009436 & 0.003021569 \\ & g_2 & 0.000006975 & 0.000475166 \\ h_2 & g_1 & 0.000001048 & 0.000755392 \\ & g_2 & 0.000002989 & 0.004276495 \end{array} \right] \\
 &= \alpha \begin{pmatrix} 0.000009436 + 0.003021569 + 0.000006975 + 0.000475166 \\ 0.000001048 + 0.000755392 + 0.000002989 + 0.004276495 \end{pmatrix} \\
 &= \alpha \begin{pmatrix} 0.0035131457 \\ 0.005035924 \end{pmatrix}
 \end{aligned}$$

Wie man sieht, müssen die durch die Marginalisation gewonnenen Ergebnisse noch normalisiert werden, damit sich die Werte für die verschiedenen Hypothesen zu Eins summieren. Daß der Vektor noch normalisiert werden muß, wird durch den Normalisierungsfaktor α angezeigt. Damit ergeben sich die BEL-Werte $\text{BEL}(C) = \begin{pmatrix} 0.0024 \\ 0.9976 \end{pmatrix}$ und $\text{BEL}(H) = \begin{pmatrix} 0.4109 \\ 0.5891 \end{pmatrix}$. **Bsp**

A.2 Shafer-Shenoy Architektur

In diesem Anhang wird ein Beispiel für die Shafer-Shenoy Architektur umfassend vorgechnet. Die einzelnen Schritte der beiden Phasen Collect Evidence und Distribute Evidence werden sowohl bei der Initialisierung wie auch bei der Auffrischung des Junction

Trees einmal in einer Liste und auch graphisch dargestellt. Wichtige Zwischenergebnisse der Berechnungen, die bei der Initialisierung und der Auffrischung durchgeführt werden, sind in zwei Tabellen übersichtlich angeordnet. Konkrete Berechnungen mit Zahlenwerten, wie zum Beispiel die Absorption von einer Clique durch eine andere Clique, runden das Beispiel ab.

Damit die Beispielrechnungen mit den Beispielrechnungen in A.1 vergleichbar sind, wurde wieder dasselbe Beispiel gewählt.

Beispiel A.2.1 (Asienbesuch (Shafer-Shenoy))

Für die Inferenz wird wieder der Junction Tree in Abbildung A.1 für das Bayesschen Netz *Asienbesuch* in Abb. 1.2 genommen.

Die Knoten des Bayesschen Netzes seien wie im Beispiel A.1.1 den Cliques des Junction Trees zugeordnet.

In den Cliques des Junction Trees stehen nun die Berechnungsvorschriften für die ψ -Tabellen: Die Clique Clq_1 enthält die Berechnungsvorschrift für die ψ -Tabelle $T_{\psi(Clq_1)} = T_{1(CD)} \cdot T_{\text{cpt}(D)} \cdot T_{Clq_4 \triangleright Clq_1} \cdot T_{1(CD)}$ stellt sicher, daß $T_{\psi(Clq_1)}$ Tabelle über den Knoten C und D ist. $T_{\text{cpt}(D)}$ repräsentiert dabei die bedingten Wahrscheinlichkeiten des Knotens D , nämlich $P(D | C)$. Die Tabelle $T_{Clq_4 \triangleright Clq_1}$ enthält die Informationen, die die Clique Clq_4 an die Clique Clq_1 schickt. In Clique Clq_2 steht die Berechnungsvorschrift für die ψ -Tabelle $T_{\psi(Clq_2)} = T_{1(CG H)} \cdot T_{\text{cpt}(H)} \cdot T_{Clq_4 \triangleright Clq_2}$, in Clique Clq_3 die Berechnungsvorschrift für die ψ -Tabelle $T_{\psi(Clq_3)} = T_{1(EFG)} \cdot T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot T_{Clq_4 \triangleright Clq_3}$, in der Clique Clq_4 , der keine Knoten zugeordnet sind, die Berechnungsvorschrift für die ψ -Tabelle $T_{\psi(Clq_4)} = T_{1(CEG)} \cdot T_{Clq_1 \triangleright Clq_4} \cdot T_{Clq_2 \triangleright Clq_4} \cdot T_{Clq_3 \triangleright Clq_4} \cdot T_{Clq_5 \triangleright Clq_4}$, in Clique Clq_5 $T_{\psi(Clq_5)} = T_{1(BCE)} \cdot T_{\text{cpt}(C)} \cdot T_{Clq_4 \triangleright Clq_5} \cdot T_{Clq_6 \triangleright Clq_5}$ und in Clique Clq_6 $T_{\psi(Clq_6)} = T_{1(AB)} \cdot T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot T_{Clq_5 \triangleright Clq_6}$. In den Separatoren des Junction Trees stehen jeweils zwei Wahrscheinlichkeitentabellen über den Knoten, die in den Separatoren enthalten sind. Alle Einträge in den Tabellen sind dabei 1.

Clique	Berechnungsvorschrift für die ψ -Tabelle
$Clq_1 = \{\underline{D}, C\}$	$T_{\psi(Clq_1)} = T_{1(CD)} \cdot T_{\text{cpt}(D)} \cdot T_{Clq_4 \triangleright Clq_1}$
$Clq_2 = \{C, G, \underline{H}\}$	$T_{\psi(Clq_2)} = T_{1(CG H)} \cdot T_{\text{cpt}(H)} \cdot T_{Clq_4 \triangleright Clq_2}$
$Clq_3 = \{\underline{E}, \underline{F}, \underline{G}\}$	$T_{\psi(Clq_3)} = T_{1(EFG)} \cdot T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot T_{Clq_4 \triangleright Clq_3}$
$Clq_4 = \{E, C, G\}$	$T_{\psi(Clq_4)} = T_{1(CEG)} \cdot T_{Clq_1 \triangleright Clq_4} \cdot T_{Clq_2 \triangleright Clq_4} \cdot T_{Clq_3 \triangleright Clq_4} \cdot T_{Clq_5 \triangleright Clq_4}$
$Clq_5 = \{B, \underline{C}, E\}$	$T_{\psi(Clq_5)} = T_{1(BCE)} \cdot T_{\text{cpt}(C)} \cdot T_{Clq_4 \triangleright Clq_5} \cdot T_{Clq_6 \triangleright Clq_5}$
$Clq_6 = \{\underline{A}, \underline{B}\}$	$T_{\psi(Clq_6)} = T_{1(AB)} \cdot T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot T_{Clq_5 \triangleright Clq_6}$

Tabelle A.4: Shafer-Shenoy: ψ -Tabellen der Cliques nach der Konstruktion des Junction Trees

Initialisierung. Wählt man als Wurzel die Clique Clq_6 , dann sieht *Collect Evidence* bei der Initialisierung des Junction Trees wie folgt aus (siehe auch Abbildung A.4(a)):

1. Die Cliques Clq_1 , Clq_2 und Clq_3 schicken ihre Nachricht an die Clique Clq_4 .
2. Jetzt schickt Clique Clq_4 ihre Nachricht an Clique Clq_5 .
3. Darauf schickt Clique Clq_5 ihre Nachricht an Clique Clq_6 und *Collect Evidence* terminiert.

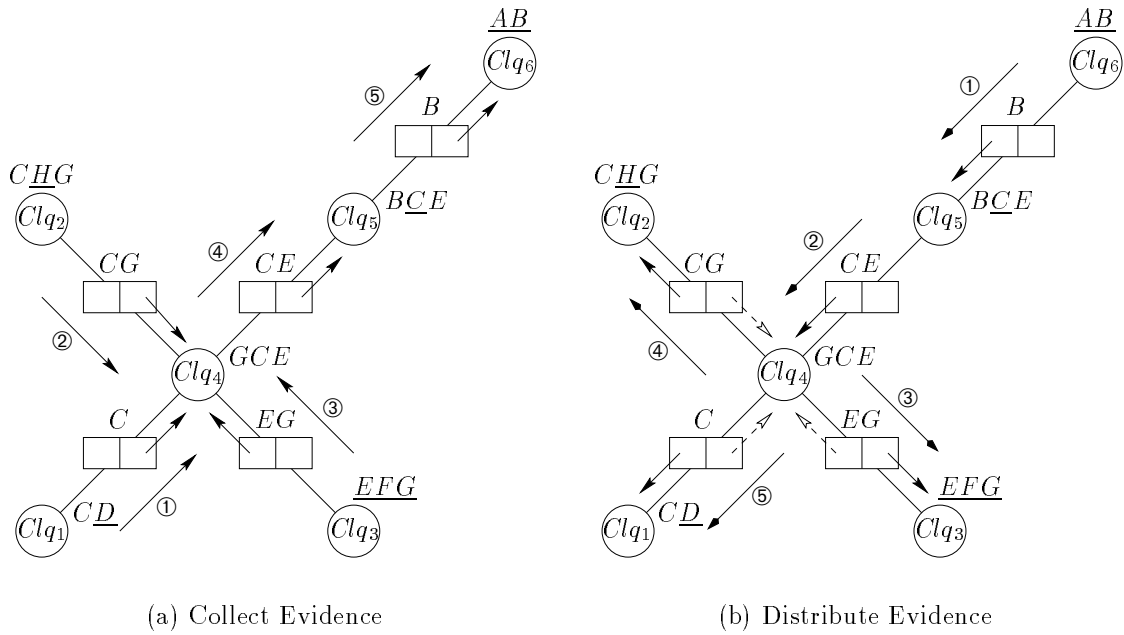


Abbildung A.4: Shafer-Shenoy: Initialisierung eines Junction Trees

Daraufhin erfolgt *Distribute Evidence* (siehe auch Abbildung A.4(b)):

1. Die Clique Clq_6 schickt nun ihrerseits ihre Nachricht an Clique Clq_5 .
2. Die Clique Clq_5 schickt ihre Nachricht an Clique Clq_4 .
3. Zum Abschluß verschickt Clique Clq_4 ihre Nachrichten an die Clques Clq_1 , Clq_2 und Clq_3 . *Distribute Evidence* terminiert.

In Tabelle A.5 sind die ψ -Tabellen der Clques und \triangleright -Tabellen der Separatoren im Verlauf der Initialisierung dargestellt. Die Tabellen, die in den verschiedenen Phasen nicht neu berechnet werden, sind aus Gründen der Übersichtlichkeit nicht dargestellt. Für die Reihenfolge der Berechnung wird auf die Phasen *Collect Evidence* und *Distribute Evidence* in Abbildung A.4 verwiesen. Die eingekreisten Zahlen erleichtern auch hier wieder die Zuordnung.

Exemplarisch wird nun gezeigt, wie sich die ψ -Tabelle der Clique Clq_4 in der Phase *Distribute Evidence* berechnet, wobei es hier vor allem darauf ankommt, die Tabelle $T_{Clq_5 \triangleright Clq_4}$ zu bestimmen:

1. Im Anfang bestimmt sich die Tabelle $T_{Clq_6 \triangleright Clq_5}$ zu:

$$T_{Clq_6 \triangleright Clq_5} = \begin{bmatrix} b_1 & b_2 \\ 0.0104 & 0.9896 \end{bmatrix}.$$

2. Die Clique Clq_4 erhält nun durch Absorption die Informationen ($T_{Clq_5 \triangleright Clq_4}$) der Clique Clq_5 wie folgt:

$$T_{Clq_5 \triangleright Clq_4} = \sum_{Clq_5 \setminus Clq_4} T_{\psi(Clq_5 \setminus Clq_4)} =$$

$$\begin{aligned}
&= \sum_{Clq_5 \setminus Clq_4} T_{\text{cpt}(C)} \cdot T_{Clq_6 \triangleright Clq_5} = \\
&= \sum_B \left[\begin{array}{cc|cc} & & c_1 & c_2 \\ b_1 & e_1 & 1 & 0 \\ & e_2 & 1 & 0 \\ b_2 & e_1 & 1 & 0 \\ & e_2 & 0 & 1 \end{array} \right] \cdot \left[\begin{array}{cc} b_1 & b_2 \\ \hline 0.0104 & 0.9896 \end{array} \right] = \\
&= \sum_B \left[\begin{array}{cc|cc} & & c_1 & c_2 \\ b_1 & e_1 & 0.0104 & 0 \\ & e_2 & 0.0104 & 0 \\ b_2 & e_1 & 0.9896 & 0 \\ & e_2 & 0 & 0.9896 \end{array} \right] = \left[\begin{array}{cc|cc} & & c_1 & c_2 \\ e_1 & 1 & 0 & \\ e_2 & 0.0104 & 0.9896 & \end{array} \right]
\end{aligned}$$

3. Der Wert der ψ -Tabelle bestimmt sich damit zu (Man beachte, daß der Clique Clq_4 keine Knoten zugeordnet sind.):

$$\begin{aligned}
T_{\psi(Clq_4)} &= \\
&= T_{Clq_1 \triangleright Clq_4} \cdot T_{Clq_2 \triangleright Clq_4} \cdot T_{Clq_3 \triangleright Clq_4} \cdot T_{Clq_5 \triangleright Clq_4} = \\
&= \left[\begin{array}{cc} c_1 & c_2 \\ \hline 1 & 1 \end{array} \right] \cdot \left[\begin{array}{cc|cc} & & c_1 & c_2 \\ g_1 & 1 & 1 & \\ g_2 & 1 & 1 & \end{array} \right] \cdot \left[\begin{array}{cc|cc} & & e_1 & e_2 \\ g_1 & 0.0315 & 0.4185 & \\ g_2 & 0.0235 & 0.5265 & \end{array} \right] \cdot \left[\begin{array}{cc|cc} & & c_1 & c_2 \\ e_1 & 1 & 0 & \\ e_2 & 0.0104 & 0.9896 & \end{array} \right] = \\
&= \left[\begin{array}{cc|cc} & & c_1 & c_2 \\ e_1 & g_1 & 0.0315 & 0 \\ & g_2 & 0.0235 & 0 \\ e_2 & g_1 & 0.0043524 & 0.4141476 \\ & g_2 & 0.0054756 & 0.5210244 \end{array} \right]
\end{aligned}$$

Auffrischung. Es liegen nun die folgenden Beobachtungen vor: Der Patient hat vor kurzem Asien besucht aber sein Röntgenbefund ist negativ. Als *findings* ergeben sich die folgenden Tabellen $T_{\text{finding}(A)} = \left[\begin{array}{cc} a_1 & a_2 \\ \hline 1 & 0 \end{array} \right]$ und $T_{\text{finding}(D)} = \left[\begin{array}{cc} d_1 & d_2 \\ \hline 0 & 1 \end{array} \right]$, die den ψ -Tabellen der Cliquen Clq_6 und Clq_1 hinzu multipliziert werden. In Tabelle A.6 sind die neuen Werte der ψ -Tabellen dargestellt. Die Cliquen des Junction Trees sind jetzt nicht mehr zueinander kalibriert. Es erfolgt eine *Auffrischung* auf dem Junction Tree. Collect Evidence sieht nun wie folgt aus (siehe auch Abbildung A.5(a)):

1. Die Cliquen Clq_1 schickt ihre Nachricht an die Clique Clq_4 . Von den Cliquen Clq_2 und Clq_3 wird keine Nachricht an Clique Clq_4 geschickt, da der Junction Tree vor dem Eintragen der Evidenzen schon kalibriert war. Von den Cliquen Clq_2 und Clq_3 würde an die Clique Clq_4 keine veränderte Nachricht geschickt werden, wie in der Tabelle A.6 zu sehen ist.
2. Jetzt schickt Clique Clq_4 ihre Nachricht an Clique Clq_5 .
3. Darauf schickt Clique Clq_5 ihre Nachricht an Clique Clq_6 und Collect Evidence terminiert.

Daraufhin erfolgt Distribute Evidence (siehe auch Abbildung A.5(b)):

1. Die Clique Clq_6 schickt nun ihrerseits ihre Nachricht an Clique Clq_5 .
2. Die Clique Clq_5 schickt ihre Nachricht an Clique Clq_4 .

\triangleright -Tab., ψ -Tab.	Wert der \triangleright -Tabelle und Berechnungsvorschrift der ψ -Tabelle bei Zuordnung	Collect Evidence	Distribute Evidence
$T_{\psi(Cl_{q_1})}$	siehe Tab. A.4		
$T_{Cl_{q_4} \triangleright Cl_{q_1}}$	$\left[\begin{array}{cc cc} & c_1 & c_2 & & \\ & 1 & 1 & & \\ \hline & c_1 & c_2 & & \\ & 1 & 1 & & \end{array} \right]$	① Absorption:	⑤ Absorption:
$T_{Cl_{q_1} \triangleright Cl_{q_4}}$		$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$	$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$
$T_{\psi(Cl_{q_2})}$	siehe Tab. A.4		
$T_{Cl_{q_4} \triangleright Cl_{q_2}}$	$\left[\begin{array}{cc cc} & c_1 & c_2 & & \\ g_1 & 1 & 1 & & \\ g_2 & 1 & 1 & & \\ \hline & c_1 & c_2 & & \\ g_1 & 1 & 1 & & \\ g_2 & 1 & 1 & & \end{array} \right]$	② Absorption:	④ Absorption:
$T_{Cl_{q_2} \triangleright Cl_{q_4}}$		$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$	$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$
$T_{\psi(Cl_{q_3})}$	siehe Tab. A.4		
$T_{Cl_{q_4} \triangleright Cl_{q_3}}$	$\left[\begin{array}{cc cc} & e_1 & e_2 & & \\ g_1 & 1 & 1 & & \\ g_2 & 1 & 1 & & \\ \hline & e_1 & e_2 & & \\ g_1 & 1 & 1 & & \\ g_2 & 1 & 1 & & \end{array} \right]$	③ Absorption:	③ Absorption:
$T_{Cl_{q_3} \triangleright Cl_{q_4}}$		$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$	$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$
$T_{\psi(Cl_{q_4})}$	siehe Tab. A.4		
$T_{Cl_{q_5} \triangleright Cl_{q_4}}$	$\left[\begin{array}{cc cc} & c_1 & c_2 & & \\ e_1 & 1 & 1 & & \\ e_2 & 1 & 1 & & \\ \hline & c_1 & c_2 & & \\ e_1 & 1 & 1 & & \\ e_2 & 1 & 1 & & \end{array} \right]$	④ Absorption:	② Absorption:
$T_{Cl_{q_4} \triangleright Cl_{q_5}}$		$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$	$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$
$T_{\psi(Cl_{q_5})}$	siehe Tab. A.4		
$T_{Cl_{q_6} \triangleright Cl_{q_5}}$	$\left[\begin{array}{cc cc} & b_1 & b_2 & & \\ & 1 & 1 & & \\ \hline & b_1 & b_2 & & \\ & 1 & 1 & & \end{array} \right]$	⑤ Absorption:	① Absorption:
$T_{Cl_{q_5} \triangleright Cl_{q_6}}$		$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$	$\left[\begin{array}{cc cc} & & & & \\ & & & & \\ \hline & & & & \\ & & & & \end{array} \right]$
$T_{\psi(Cl_{q_6})}$	siehe Tab. A.4		

Tabelle A.5: Shafer-Shenoy: ψ -Tabellen bei der Initialisierung. Es sind noch keine Knoten instantiiert.

\triangleright -Tab., ψ -Tab.	Wert der \triangleright -Tabelle und Berechnungsvorschrift der ψ -Tabelle bei Zuordnung	Collect Evidence	Distribute Evidence
$T_{\psi(Cl_{q_1})}$	Hypothese d_2 : $T_{\psi(Cl_{q_1})} \cdot T_{\text{finding}(D)}$		
$T_{Cl_{q_4} \triangleright Cl_{q_1}}$	$\begin{bmatrix} c_1 & c_2 \\ 1 & 1 \end{bmatrix}$	① Absorption: $\begin{bmatrix} & \end{bmatrix}$	⑤ Absorption: $\begin{bmatrix} & \end{bmatrix}$
$T_{Cl_{q_1} \triangleright Cl_{q_4}}$	$\begin{bmatrix} c_1 & c_2 \\ 1 & 1 \end{bmatrix}$	$\begin{bmatrix} & \end{bmatrix}$	
$T_{\psi(Cl_{q_2})}$	siehe Tab. A.4	nicht neu berechnet:	④ Absorption:
$T_{Cl_{q_4} \triangleright Cl_{q_2}}$	$\begin{bmatrix} & c_1 & c_2 \\ g_1 & 1 & 1 \\ g_2 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} & \\ 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1.049-05 & 3.777-03 \\ 9.965-06 & 4.752-03 \end{bmatrix}$
$T_{Cl_{q_2} \triangleright Cl_{q_4}}$	$\begin{bmatrix} & c_1 & c_2 \\ g_1 & 1 & 1 \\ g_2 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} & \\ 1 & 1 \end{bmatrix}$	
$T_{\psi(Cl_{q_3})}$	siehe Tab. A.4	nicht neu berechnet:	③ Absorption:
$T_{Cl_{q_4} \triangleright Cl_{q_3}}$	$\begin{bmatrix} & e_1 & e_2 \\ g_1 & 1 & 1 \\ g_2 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} & \\ 0.0235 & 0.5265 \end{bmatrix}$	$\begin{bmatrix} 0.0002 & 9.035-03 \\ 0.0002 & 9.035-03 \end{bmatrix}$
$T_{Cl_{q_3} \triangleright Cl_{q_4}}$	$\begin{bmatrix} & e_1 & e_2 \\ g_1 & 1 & 1 \\ g_2 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.0315 & 0.4185 \\ 0.0235 & 0.5265 \end{bmatrix}$	
$T_{\psi(Cl_{q_4})}$	siehe Tab. A.4	② Absorption:	② Absorption:
$T_{Cl_{q_5} \triangleright Cl_{q_4}}$	$\begin{bmatrix} & c_1 & c_2 \\ e_1 & 1 & 1 \\ e_2 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} & \\ 1.966-04 & 8.884-01 \end{bmatrix}$	$\begin{bmatrix} 0.0002 & 0 \\ 0.001-02 & 9.025-03 \end{bmatrix}$
$T_{Cl_{q_4} \triangleright Cl_{q_5}}$	$\begin{bmatrix} & c_1 & c_2 \\ e_1 & 1 & 1 \\ e_2 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.0011 & 0 \\ 1.966-04 & 8.884-01 \end{bmatrix}$	
$T_{\psi(Cl_{q_5})}$	siehe Tab. A.4	③ Absorption:	① Absorption:
$T_{Cl_{q_6} \triangleright Cl_{q_5}}$	$\begin{bmatrix} b_1 & b_2 \\ 1 & 1 \end{bmatrix}$	$\begin{bmatrix} & \end{bmatrix}$	$\begin{bmatrix} & \end{bmatrix}$
$T_{Cl_{q_5} \triangleright Cl_{q_6}}$	$\begin{bmatrix} b_1 & b_2 \\ 1 & 1 \end{bmatrix}$	$\begin{bmatrix} & \end{bmatrix}$	
$T_{\psi(Cl_{q_6})}$	Hypothese a_1 : $T_{\psi(Cl_{q_1})} \cdot T_{\text{finding}(A)}$		

Tabelle A.6: Shafer-Shenoy: ψ -Tabellen bei der Auffrischung. Es wurden die Knoten A und D instantiiert.

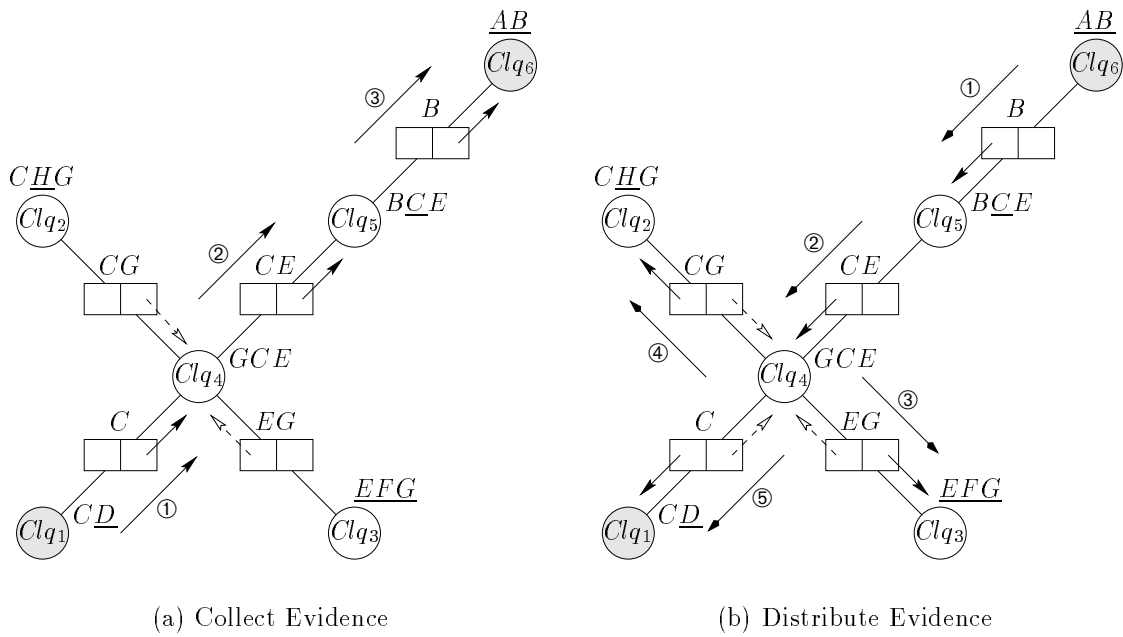


Abbildung A.5: Shafer-Shenoy: Auffrischung eines Junction Trees nach der Eingabe von neuen Beobachtungen (durch grau schraffierte Kreise gekennzeichnet)

3. Zum Abschluß verschickt Clique Clq_4 ihre Nachrichten an die Cliquen Clq_1 , Clq_2 und Clq_3 . Distribute Evidence terminiert.

Der Junction Tree ist nun wieder kalibriert. Die Ergebnisse für die ψ -Tabellen sind der Tabelle A.6 zu entnehmen.

Die BEL-Werte für die einzelnen Knoten des Bayesschen Netzes können durch Marginalisation aus jeder ψ -Tabelle bestimmt werden, in der sich der Knoten befindet. Für den Knoten C wird dies hier exemplarisch gemacht:

$$\begin{aligned}
 \text{BEL}(C) &= \alpha \sum_{Clq_1 \setminus \{C\}} T_{\psi(Clq_1)} = \alpha \sum_{Clq_1 \setminus \{C\}} T_{1(CD)} \cdot T_{\text{cpt}(D)} \cdot T_{Clq_4 \triangleright Clq_1} \cdot T_{\text{finding}(D)} \\
 &= \alpha \sum_{Clq_1 \setminus \{C\}} \left[\begin{array}{c|cc} & d_1 & d_2 \\ \hline c_1 & 0.98 & 0.02 \\ c_2 & 0.05 & 0.95 \end{array} \right] \cdot \left[\begin{array}{cc} c_1 & c_2 \\ \hline 0.0010225 & 0.0089775 \end{array} \right] \cdot \left[\begin{array}{cc} d_1 & d_2 \\ \hline 0 & 1 \end{array} \right] \\
 &= \alpha \sum_{Clq_1 \setminus \{C\}} \left[\begin{array}{c|cc} & d_1 & d_2 \\ \hline c_1 & 0 & 2.045e-5 \\ c_2 & 0 & 0.008528625 \end{array} \right] = \alpha \begin{pmatrix} 2.045e-5 \\ 0.008528625 \end{pmatrix} = \begin{pmatrix} 0.0024 \\ 0.9976 \end{pmatrix}
 \end{aligned}$$

Bsp

A.3 Weitere Anwendungen der Polynom-Propagierung

In diesem Abschnitt werden nun noch einige Anwendungsmöglichkeiten der Polynom-Propagierung vorgestellt.

In den nächsten Beispielen werden die Wahrscheinlichkeitstabellen der Knoten parametrisiert.

Beispiel A.3.1 (apriori-Parametrisierung)

Sei der Graph wie in Abbildung 1.1 und die dazugehörigen bedingten Wahrscheinlichkeiten wie in Tabelle 1.1 gegeben. Die apriori-Wahrscheinlichkeit des Knotens A sei nun nicht mehr $P(A) = (0.20, 0.80)$ sondern $P(A) = (A1, A2)$. Nach der Initialisierung des Bayesschen Netzes kann man nun anhand der Beliefwerte den Einfluß der a-priori-Wahrscheinlichkeit ablesen:

$$\begin{aligned} \text{BEL}(A) &= \alpha \begin{pmatrix} A1 \\ A2 \end{pmatrix} \\ \text{BEL}(B) &= \alpha \begin{pmatrix} 0.8 A1 + 0.2 A2 \\ 0.2 A1 + 0.8 A2 \end{pmatrix} \\ \text{BEL}(C) &= \alpha \begin{pmatrix} 0.2 A1 + 0.05 A2 \\ 0.8 A1 + 0.95 A2 \end{pmatrix} \\ \text{BEL}(D) &= \alpha \begin{pmatrix} 0.74 A1 + 0.245 A2 \\ 0.26 A1 + 0.755 A2 \end{pmatrix} \\ \text{BEL}(E) &= \alpha \begin{pmatrix} 0.64 A1 + 0.61 A2 \\ 0.36 A1 + 0.39 A2 \end{pmatrix}. \end{aligned}$$

Unter der Voraussetzung von $A2 = 1 - A1$ erhält man:

$$\begin{aligned} \text{BEL}(A) &= \begin{pmatrix} A1 \\ 1 - A1 \end{pmatrix} \\ \text{BEL}(B) &= \begin{pmatrix} 0.8 A1 + 0.2 (1 - A1) \\ 0.2 A1 + 0.8 (1 - A1) \end{pmatrix} = \begin{pmatrix} 0.2 + 0.6 A1 \\ 0.8 - 0.6 A1 \end{pmatrix} \\ \text{BEL}(C) &= \begin{pmatrix} 0.2 A1 + 0.05 (1 - A1) \\ 0.8 A1 + 0.95 (1 - A1) \end{pmatrix} = \begin{pmatrix} 0.05 + 0.15 A1 \\ 0.95 - 0.15 A1 \end{pmatrix} \\ \text{BEL}(D) &= \begin{pmatrix} 0.74 A1 + 0.245 (1 - A1) \\ 0.26 A1 + 0.755 (1 - A1) \end{pmatrix} = \begin{pmatrix} 0.245 + 0.495 A1 \\ 0.755 - 0.495 A1 \end{pmatrix} \\ \text{BEL}(E) &= \begin{pmatrix} 0.64 A1 + 0.61 (1 - A1) \\ 0.36 A1 + 0.39 (1 - A1) \end{pmatrix} = \begin{pmatrix} 0.61 + 0.03 A1 \\ 0.39 - 0.03 A1 \end{pmatrix}. \end{aligned}$$

Die Beliefwerte der einzelnen Knoten sind also linear von $A1$ abhängig. Dieses ist in der Abbildung A.6 dargestellt.

Die Knoten C und E werden also am wenigsten von der apriori-Wahrscheinlichkeit $A1$ des Knotens A beeinflusst. Beim Knoten E beträgt sie sogar maximal nur den Wert 0.03. Die Winkelhalbierende markiert die maximale Abhängigkeit, d.h. eine Gerade mit größerer Steigung ist nicht möglich. **Bsp**

Beispiel A.3.2 (CPT-Parametrisierung)

Sei der Graph wie in Abbildung 1.1 und die dazugehörigen a-priori- und bedingten Wahrscheinlichkeiten wie in Tabelle 1.1 gegeben. Die CPT des Knotens C wird parametrisiert und bestimmt sich nun zu

$$T_{\text{cpt}(C)} = \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline a_1 & X1 & X2 \\ a_2 & Y1 & Y2 \end{array} \right].$$

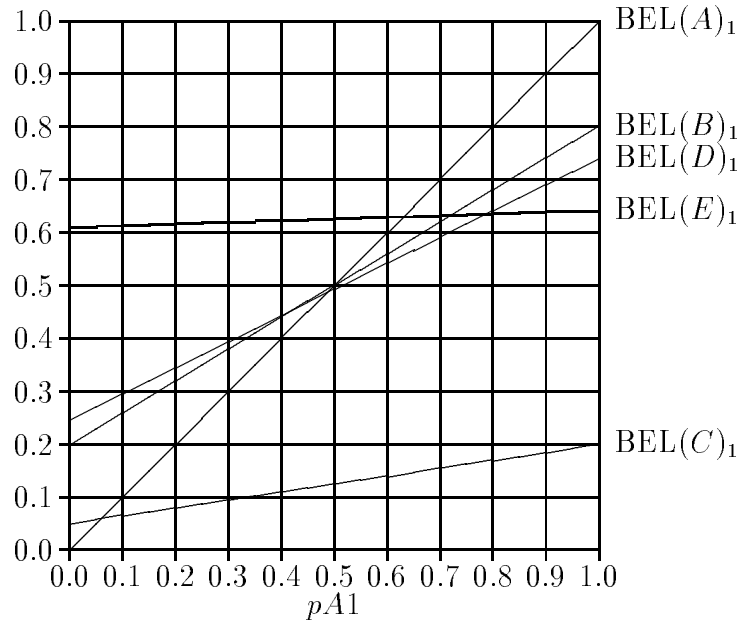


Abbildung A.6: Der Beliefwert in Abhängigkeit von A_1 beschreibt eine Gerade

Es gilt: $X_2 = 1 - X_1$ und $Y_2 = 1 - Y_1$. Dieses soll aber noch nicht ausgenutzt werden, um zu sehen, wie sich Knoten auf das Netz auswirken könnten, die mehr als zwei Hypothesen besitzen.

Nach der Initialisierung des Bayesschen Netzes kann man nun anhand der Beliefwerte den Einfluß der bedingten Wahrscheinlichkeiten ablesen:

$$\begin{aligned} \text{BEL}(A) &= \alpha \begin{pmatrix} 0.20 X_1 + 0.20 X_2 \\ 0.80 Y_1 + 0.80 Y_2 \end{pmatrix} \\ \text{BEL}(B) &= \alpha \begin{pmatrix} 0.16 X_1 + 0.16 X_2 + 0.16 Y_1 + 0.16 Y_2 \\ 0.04 X_1 + 0.04 X_2 + 0.64 Y_1 + 0.64 Y_2 \end{pmatrix} \\ \text{BEL}(C) &= \alpha \begin{pmatrix} 0.20 X_1 + 0.80 Y_1 \\ 0.20 X_2 + 0.80 Y_2 \end{pmatrix} \\ \text{BEL}(D) &= \alpha \begin{pmatrix} 0.156 X_1 + 0.146 X_2 + 0.576 Y_1 + 0.176 Y_2 \\ 0.044 X_1 + 0.054 X_2 + 0.224 Y_1 + 0.624 Y_2 \end{pmatrix} \\ \text{BEL}(E) &= \alpha \begin{pmatrix} 0.16 X_1 + 0.12 X_2 + 0.64 Y_1 + 0.48 Y_2 \\ 0.04 X_1 + 0.08 X_2 + 0.16 Y_1 + 0.32 Y_2 \end{pmatrix}. \end{aligned}$$

Unter der Voraussetzung von $X_1 + X_2 = 1$ und $Y_1 + Y_2 = 1$ erhält man:

$$\begin{aligned} \text{BEL}(A) &= \alpha \begin{pmatrix} 0.20 \\ 0.80 \end{pmatrix} = \begin{pmatrix} 0.20 \\ 0.80 \end{pmatrix} \\ \text{BEL}(B) &= \alpha \begin{pmatrix} 0.16 + 0.16 \\ 0.04 + 0.64 \end{pmatrix} = \begin{pmatrix} 0.32 \\ 0.68 \end{pmatrix} \\ \text{BEL}(C) &= \alpha \begin{pmatrix} 0.20 X_1 + 0.80 Y_1 \\ 0.20 X_2 + 0.80 Y_2 \end{pmatrix} = \begin{pmatrix} 0.20 X_1 + 0.80 Y_1 \\ 0.20 X_2 + 0.80 Y_2 \end{pmatrix} \\ \text{BEL}(D) &= \alpha \begin{pmatrix} 0.156 X_1 + 0.146 X_2 + 0.576 Y_1 + 0.176 Y_2 \\ 0.044 X_1 + 0.054 X_2 + 0.224 Y_1 + 0.624 Y_2 \end{pmatrix} = \\ &= \begin{pmatrix} 0.156 X_1 + 0.146 X_2 + 0.576 Y_1 + 0.176 Y_2 \\ 0.044 X_1 + 0.054 X_2 + 0.224 Y_1 + 0.624 Y_2 \end{pmatrix} \end{aligned}$$

$$\begin{aligned} \text{BEL}(E) &= \alpha \begin{pmatrix} 0.16 X1 + 0.12 X2 + 0.64 Y1 + 0.48 Y2 \\ 0.04 X1 + 0.08 X2 + 0.16 Y1 + 0.32 Y2 \end{pmatrix} = \\ &= \begin{pmatrix} 0.16 X1 + 0.12 X2 + 0.64 Y1 + 0.48 Y2 \\ 0.04 X1 + 0.08 X2 + 0.16 Y1 + 0.32 Y2 \end{pmatrix}. \end{aligned}$$

Jetzt wird ausgenutzt, daß $X2 = 1 - X1$ und $Y2 = 1 - Y1$ gilt. Man erhält:

$$\begin{aligned} \text{BEL}(A) &= \begin{pmatrix} 0.20 \\ 0.80 \end{pmatrix} \\ \text{BEL}(B) &= \begin{pmatrix} 0.32 \\ 0.68 \end{pmatrix} \\ \text{BEL}(C) &= \begin{pmatrix} 0.20 X1 + 0.80 Y1 \\ 0.20 (1 - X1) + 0.80 (1 - Y1) \end{pmatrix} = \begin{pmatrix} 0 + 0.20 X1 + 0.80 Y1 \\ 1 - 0.20 X1 - 0.80 Y1 \end{pmatrix} \\ \text{BEL}(D) &= \begin{pmatrix} 0.156 X1 + 0.146 (1 - X1) + 0.576 Y1 + 0.176 (1 - Y1) \\ 0.044 X1 + 0.054 (1 - X1) + 0.224 Y1 + 0.624 (1 - Y1) \end{pmatrix} = \\ &= \begin{pmatrix} 0.146 + 0.01 X1 + 0.176 + 0.4 Y1 \\ 0.054 - 0.01 X1 + 0.624 - 0.4 Y1 \end{pmatrix} = \begin{pmatrix} 0.322 + 0.01 X1 + 0.4 Y1 \\ 0.678 - 0.01 X1 - 0.4 Y1 \end{pmatrix} \\ \text{BEL}(E) &= \begin{pmatrix} 0.16 X1 + 0.12 (1 - X1) + 0.64 Y1 + 0.48 (1 - Y1) \\ 0.04 X1 + 0.08 (1 - X1) + 0.16 Y1 + 0.32 (1 - Y1) \end{pmatrix} = \\ &= \begin{pmatrix} 0.12 + 0.04 X1 + 0.48 + 0.16 Y1 \\ 0.08 - 0.04 X1 + 0.32 - 0.16 Y1 \end{pmatrix} = \begin{pmatrix} 0.60 + 0.04 X1 + 0.16 Y1 \\ 0.40 - 0.04 X1 - 0.16 Y1 \end{pmatrix}. \end{aligned}$$

Die Beliefwerte der einzelnen Knoten sind also linear von $X1$ und $Y1$ abhängig. Dieses ist in der Abbildung A.7 dargestellt.

Die Knoten A und B werden also von $X1$ und $Y1$ gar nicht beeinflusst. $X1$ hat im Vergleich zu $Y1$ auf den Beliefwert der Knoten C , D und E nur einen sehr geringen Einfluß. Bei den Knoten D und E ist dieser sogar vernachlässigbar klein.

Will man nun eine bestimmte Wahrscheinlichkeitentabelle für den Knoten C modellieren, so läßt sich ein lineares Gleichungssystem aufstellen. Kennt man die BEL-Werte der Knoten D und E (z.B. seien $\text{BEL}(D) = \begin{pmatrix} 0.344 \\ 0.656 \end{pmatrix}$ und $\text{BEL}(E) = \begin{pmatrix} 0.616 \\ 0.384 \end{pmatrix}$), so erhält man:

$$\begin{aligned} 0.344 &= 0.322 + 0.01 X1 + 0.40 Y1 \Leftrightarrow 0.022 = 0.01 X1 + 0.40 Y1 \quad (\text{I}) \\ 0.656 &= 0.678 - 0.01 X1 - 0.40 Y1 \Leftrightarrow 0.022 = 0.01 X1 + 0.40 Y1 \\ 0.616 &= 0.600 + 0.04 X1 + 0.16 Y1 \Leftrightarrow 0.016 = 0.04 X1 + 0.16 Y1 \quad (\text{II}) \\ 0.384 &= 0.400 - 0.04 X1 - 0.16 Y1 \Leftrightarrow 0.016 = 0.04 X1 + 0.16 Y1. \end{aligned}$$

Dies läßt sich mit dem Gaußschen Algorithmus (siehe [Lamprecht 78]) lösen, und man erhält:

mit (II) := (II) - 4 * (I):

$$\begin{aligned} 0.022 &= 0.01 X1 + 0.40 Y1 \quad (\text{I}) \\ -0.072 &= \quad \quad \quad -1.44 Y1 \quad (\text{II}) \end{aligned}$$

und (II) := (II)/(-1.44):

$$\begin{aligned} 0.022 &= 0.01 X1 + 0.40 Y1 \quad (\text{I}) \\ 0.05 &= \quad \quad \quad Y1 \quad (\text{II}) \end{aligned}$$

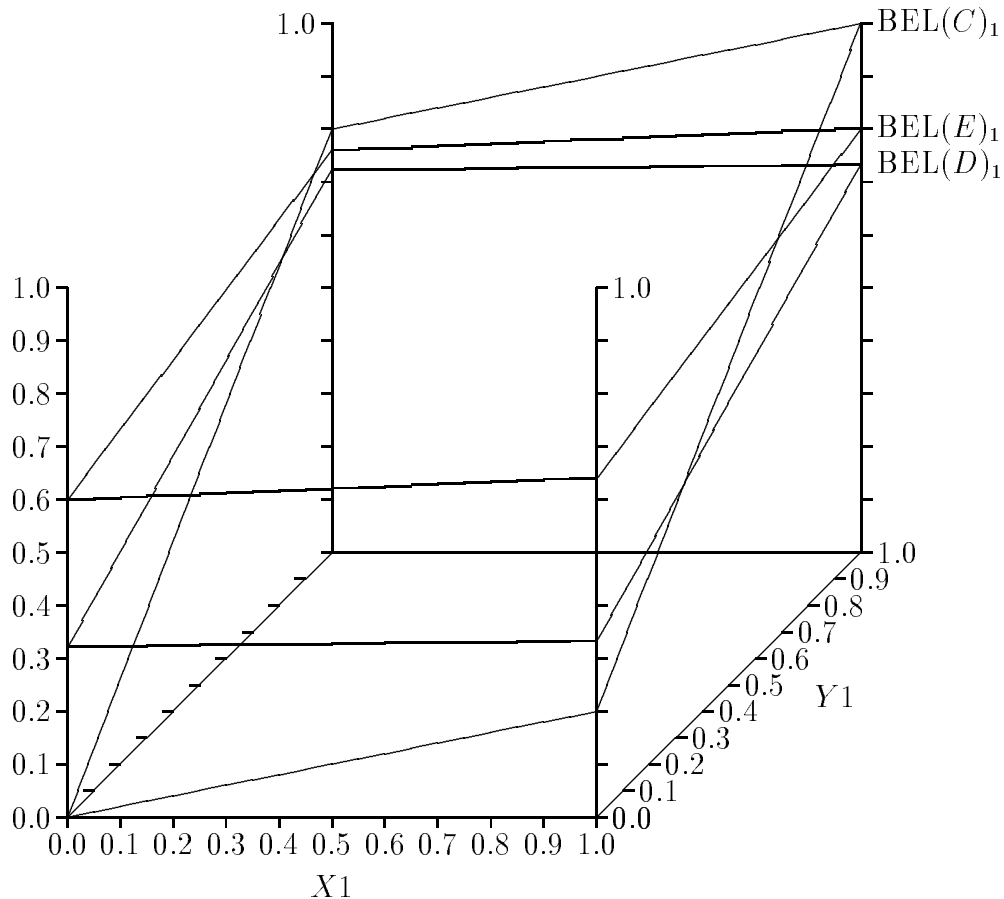


Abbildung A.7: Der Beliefwert in Abhängigkeit von X_1 und Y_1 beschreibt eine Ebene

Einsetzen von (II) in (I) und Umformen von (I) liefert:

$$0.20 = X_1 \quad (\text{I})$$

$$0.05 = Y_1 \quad (\text{II}).$$

Die CPT des Knotens C bestimmt sich nun zu

$$T_{\text{cpt}(C)} = \left[\begin{array}{c|cc} & c_1 & c_2 \\ \hline a_1 & 0.20 & 0.80 \\ a_2 & 0.05 & 0.95 \end{array} \right].$$

Bsp

Beispiel A.3.3 (Sensitivitätsanalyse)

Bei der Sensitivitätsanalyse wird ein Eintrag einer Wahrscheinlichkeitentabelle systematisch verändert. Die anderen Einträge der Wahrscheinlichkeitentabelle werden an diese Veränderung angepaßt, so daß sich die entsprechenden Einträge immer noch zu eins addieren. Wie in [van der Gaag & Coupé 98; van der Gaag & Kjærulff 00] erhält man Ausdrücke, die linear abhängig sind, in Abhängigkeit der parametrisierten Einträge der Wahrscheinlichkeitentabellen.

Bsp

A.4 D-Separations-Kriterium

Aus Gründen der Übersichtlichkeit ist an dieser Stelle das D-Separations Kriterium, wie es schon im Abschnitt 3.2.1 definiert ist, noch einmal aufgeführt.

Definition A.4.1 (D-Separations-Kriterium I)

Seien \mathbf{X} , \mathbf{Y} und \mathbf{Z} drei disjunkte Teilmengen von Knoten des gerichteten azyklischen Graphen \mathcal{G} des Bayesschen Netzes \mathcal{BN} . Dabei muß die Vereinigung von \mathbf{X} , \mathbf{Y} und \mathbf{Z} nicht die gesamte Knotenmenge des Graphen \mathcal{G} ergeben. Dann *d-separiert* die Menge \mathbf{Z} die beiden Mengen \mathbf{X} und \mathbf{Y} voneinander (Schreibweise: $\langle \mathbf{X} \mid \mathbf{Z} \mid \mathbf{Y} \rangle_{\mathcal{G}}$), wenn es entlang jeden Pfades zwischen einem Knoten aus \mathbf{X} und einem Knoten aus \mathbf{Y} ein Knoten w existiert, der eine der beiden nachfolgenden Eigenschaften erfüllt:

1. w hat Vorgängerknoten und weder w noch seine Nachfolgerknoten sind in \mathbf{Z} , oder
2. w hat keine Vorgängerknoten, und w ist in \mathbf{Z} .

□

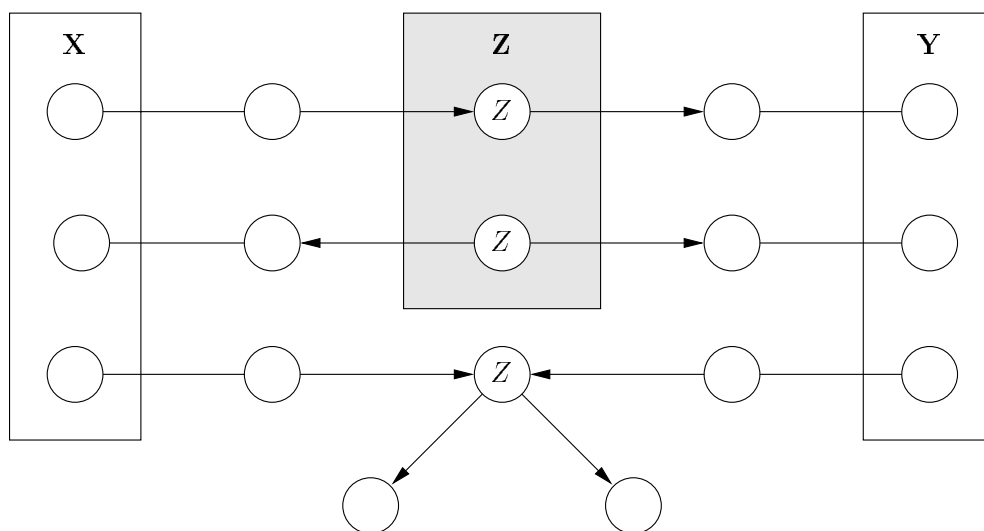


Abbildung A.8: D-Separations-Kriterium: Drei Arten wie ein Pfad von \mathbf{X} nach \mathbf{Y} durch die Menge \mathbf{Z} blockiert werden kann. Wird jeder Pfad von \mathbf{X} nach \mathbf{Y} blockiert, so sagt man, daß die Menge \mathbf{Z} die beiden Menge \mathbf{X} und \mathbf{Y} voneinander d-separiert (entnommen aus [Russell & Norvig 95]).

Die folgende Definition besagt dasselbe bzgl. des D-Separations Kriteriums wie die obige Definition. Beide Definitionen zusammen tragen jedoch gut zum Verständnis des D-Separations Kriteriums bei.

Definition A.4.2 (D-Separations-Kriterium II)

Eine Menge \mathbf{Z} von Knoten *d-separiert* zwei Mengen von Knoten \mathbf{X} und \mathbf{Y} , wenn jeder ungerichtete Pfad von einem Knoten in \mathbf{X} zu einem Knoten aus \mathbf{Y} durch \mathbf{Z} blockiert wird. Ein Pfad wird durch eine Menge \mathbf{Z} von Knoten blockiert, wenn es einen Knoten Z auf dem Pfad gibt, für den eine der drei folgenden Bedingungen gilt:

1. $Z \in \mathbf{Z}$ und Z hat auf dem Pfad eine eingehende Kante und eine austretende Kante.
2. $Z \in \mathbf{Z}$ und beide Kanten von Z auf dem Pfad zeigen von Z weg.
3. Weder Z noch einer seiner Nachfolgerknoten sind in der Knotenmenge \mathbf{Z} , und beide Kanten auf dem Pfad zeigen zu Z hin.

□

Die obige Definition ist in Abbildung A.8 grafisch veranschaulicht.

Abschließend jetzt noch zwei Beispiele bzgl. des D-Separations Kriteriums.

Das folgende Beispiel ist aus [Pearl 88] entnommen.

Beispiel A.4.3 (blockieren, aktivieren)

In Abbildung A.9 d-separiert die Menge $\mathbf{Z} = \{A\}$ die beiden Mengen $\mathbf{X} = \{B\}$ und $\mathbf{Y} = \{C\}$ voneinander, denn der Knoten $A \in \mathbf{Z}$ blockiert den Pfad $B \leftarrow A \rightarrow C$, und der Pfad $B \rightarrow D \leftarrow C$ wird blockiert, da der Knoten D und seine Nachfolger nicht in der Menge \mathbf{Z} enthalten sind.

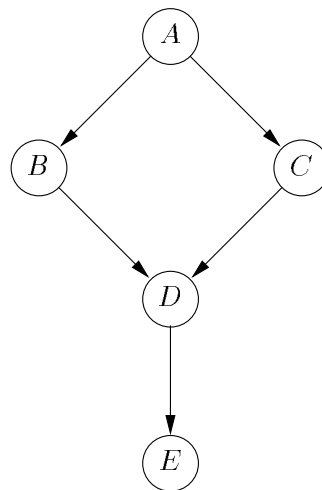


Abbildung A.9: D-Separations-Kriterium: Verdeutlichung zur Definition von *blockieren* und *aktivieren*

Andererseits separiert die Menge $\mathbf{Z}' = \{A, E\}$ nicht die beiden anderen Mengen $\mathbf{X} = \{B\}$ und $\mathbf{Y} = \{C\}$, denn der Pfad $B \rightarrow D \leftarrow C$ wird durch den Knoten E aktiviert.

□

Das nun folgende Beispiel stammt aus [Russell & Norvig 95].

Beispiel A.4.4 (D-Separations-Kriterium)

Sei das Bayessche Netz in Abbildung A.10 gegeben. Es ist ein einfaches Bayessches Netz zur Darstellung des Sachverhaltes "Auto", wobei nur die Zusammenhänge zwischen Batterie, Radio, Zündung, Benzin, Starten und Fahren betrachtet werden.

Aus dem Graphen lassen sich mit dem D-Separations Kriterium z.B. die folgenden Abhängigkeiten folgern:

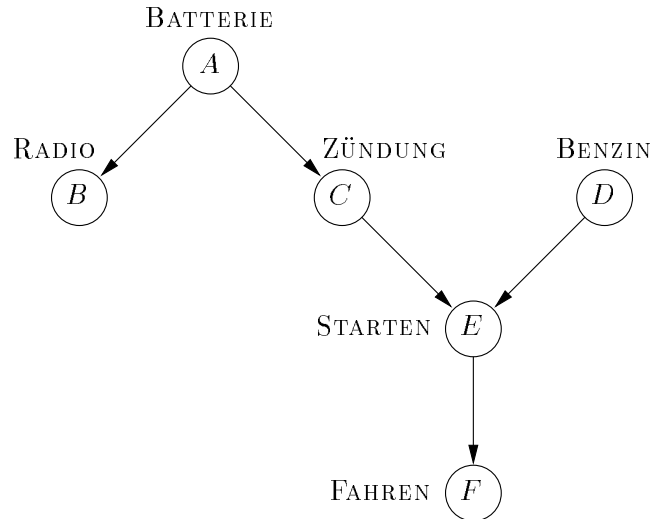


Abbildung A.10: D-Separations-Kriterium: Beispielnetz Auto

1. BENZIN und RADIO sind unabhängig voneinander, wenn BATTERIE bekannt ist.
2. BENZIN und RADIO sind unabhängig voneinander, wenn gar keine Evidenz vorliegt.
3. BENZIN und RADIO sind abhängig voneinander, wenn Evidenz über STARTEN vorliegt. Läßt sich das Auto nicht starten, dann ist das spielende Radio ein Indiz dafür, daß die Batterie in Ordnung ist und das Benzin alle sein muß.
4. BENZIN und RADIO sind abhängig voneinander, wenn Evidenz über FAHREN vorliegt. FAHREN ist ein Nachfolger von STARTEN.

Bsp

Anhang B

Mathematik

In diesem Anhang wird die *Algebra von Wahrscheinlichkeitstabellen* formal und tiefergehend wie bisher vorgestellt sowie die sogenannten *Multivariaten Polynome* formal eingeführt. Ebenso finden sich hier in einem eigenen Abschnitt wichtige Nebenrechnungen aus verschiedenen Kapiteln dieser Arbeit.

B.1 Algebra von Wahrscheinlichkeitstabellen

Die a-priori und bedingten Wahrscheinlichkeiten der Zufallsvariablen müssen dem Knoten zugeordnet werden, der mit der Zufallsvariablen assoziiert ist. In dieser Arbeit geschieht dies durch *Wahrscheinlichkeitstabellen*, die die a-priori oder bedingten Wahrscheinlichkeiten jeweils einer Zufallsvariablen des Bayesschen Netzes zusammenfaßt. Wahrscheinlichkeitstabellen sind bestimmte Tabellen über Knotenmengen. Alle notwendigen Berechnungen können als Berechnungen von Tabellen über Knotenmengen angegeben werden. Im folgenden wird beschrieben, wie mit Tabellen über Knotenmengen gerechnet wird, und welche besonderen Tabellen über Knotenmengen existieren.

Definition B.1.1 (Tabelle über einer Knotenmenge)

Seien K_1, \dots, K_n Knoten und bezeichne k_{ℓ, j_ℓ} die j_ℓ -te Hypothese des Knotens K_ℓ . Weiter habe der Knoten K_ℓ genau $|K_\ell|$ Hypothesen. Ein mehrdimensionaler Vektor A für den gilt:

- (i) Das Vektorelement $A[j_1, \dots, j_n]$ steht für die Konfiguration $(k_{1, j_1}, \dots, k_{n, j_n})$ der Knoten K_1, \dots, K_n .
- (ii) Für jede Konfiguration $(k_{1, j_1}, \dots, k_{n, j_n})$ der Knoten K_1, \dots, K_n existiert ein Vektorelement $A[j_1, \dots, j_n]$ des mehrdimensionalen Vektors A .

wird als *Tabelle über der Knotenmenge* $\mathbf{V} = \{K_1, \dots, K_n\}$ bezeichnet. Sie wird mit $T_{\mathbf{V}}$ bezeichnet. Die Menge aller möglichen Tabellen über der Knotenmenge $\mathbf{V} = \{K_1, \dots, K_n\}$ wird mit $\mathbf{K}^{\mathbf{V}}$ bezeichnet, wobei \mathbf{K} eine Menge ist, deren Elemente als Einträge in den Tabellen erlaubt sind. Ein Zugriff auf die Elemente der Tabelle ist durch Angabe eines Indizes möglich: $T(K_1, \dots, K_n)_{j_1, \dots, j_n}$ bezeichnet das Tabellenelement, das für die Konfiguration $(k_{1, j_1}, \dots, k_{n, j_n})$ der Knoten K_1, \dots, K_n steht. Man schreibt

auch $T(k_{1,j_1}, \dots, k_{n,j_n})$ anstatt $T(K_1, \dots, K_n)_{j_1, \dots, j_n}$. Ist es eindeutig, für welchen Knoten ein Hypothesenname steht, so können die Hypothesennamen für den Zugriff auf die Tabelle auch in beliebiger Reihenfolge angegeben werden.

Ist $(k_{1,j_1}, \dots, k_{n,j_n}, \ell_{1,i_1}, \dots, \ell_{m,i_m})$ eine Konfiguration der Knoten K_1, \dots, K_n und L_1, \dots, L_m , dann gilt:

$$T(K_1, \dots, K_n)_{k_{1,j_1}, \dots, k_{n,j_n}, \ell_{1,i_1}, \dots, \ell_{m,i_m}} = \\ T(K_1, \dots, K_n)_{k_{1,j_1}, \dots, k_{n,j_n}, \widehat{\ell_{1,i_1}}, \dots, \widehat{\ell_{m,i_m}}} = T(K_1, \dots, K_n)_{k_{1,j_1}, \dots, k_{n,j_n}}.$$

Mit $\widehat{\ell_{1,i_1}}$ wird angegeben, daß der Index ℓ_{1,i_1} weggelassen werden kann.

Mit $T_{1(K_1, \dots, K_n)}$ wird die Tabelle über der Knotenmenge $\{K_1, \dots, K_n\}$ bezeichnet, die nur Einsen als Einträge in der Tabelle besitzt. □

Mit diesen Bezeichnungen wird nun die *Tabelle bedingter Wahrscheinlichkeiten* und die *Tabelle gemeinsamer Wahrscheinlichkeiten* eingeführt.

Definition B.1.2 (Tabelle bedingter Wahrscheinlichkeiten)

Seien V_1, \dots, V_n die Vorgänger (Elternknoten) des Knotens K , bezeichne v_{ℓ, j_ℓ} die j_ℓ -te Hypothese des Vorgängerknotens V_ℓ ($1 \leq j_\ell \leq |V_\ell|$) und k_i ($i = 1, \dots, r$) die Hypothesen von K , ($|K| = r$). Eine Tabelle $T(K, V_1, \dots, V_n) \in \mathbf{K}^{\mathbf{V}}$ mit $\mathbf{V} = \{V_1, \dots, V_n\}$ und \mathbf{K} ist die Menge der reellen Zahlen von 0 bis 1 ($\mathbf{K} = [0..1]$) für die gilt:

- (i) $T(K, V_1, \dots, V_n)_{i, j_1, \dots, j_n}$ ist das i, j_1, \dots, j_n -te Element mit

$$T(K, V_1, \dots, V_n)_{i, j_1, \dots, j_n} = P(k_i \mid v_{1, j_1}, \dots, v_{n, j_n}).$$

- (ii) $\sum_{i=1}^r T(K, V_1, \dots, V_n)_{i, j_1, \dots, j_n} = 1$.

heißt *Tabelle bedingter Wahrscheinlichkeiten*, die $P(K \mid V_1, \dots, V_n)$ repräsentiert. Sind die Elternknoten von K bekannt, so schreibt man für die *Tabelle bedingter Wahrscheinlichkeiten von K* auch $T_{\text{cpt}(K)}$. Sie gibt die Wahrscheinlichkeit für das Eintreffen einer Hypothese k_i des Knotens K in Abhängigkeit von den Zuständen $v_{1, j_1}, \dots, v_{n, j_n}$ der Elternknoten V_1, \dots, V_n an ($1 \leq j_\ell \leq |V_\ell|$, ($\ell = 1, \dots, n$)). □

Definition B.1.3 (Tabelle gemeinsamer Wahrscheinlichkeiten)

Seien die Knoten K_1, \dots, K_n gegeben und bezeichne k_{ℓ, j_ℓ} die j_ℓ -te Hypothese des Knotens K_ℓ ($1 \leq j_\ell \leq |K_\ell|$). Eine Tabelle $T(K_1, \dots, K_n) \in \mathbf{K}^{\mathbf{V}}$ mit $\mathbf{V} = \{K_1, \dots, K_n\}$ und \mathbf{K} ist die Menge der reellen Zahlen von 0 bis 1 ($\mathbf{K} = [0..1]$) für die gilt:

- (i) $T(K_1, \dots, K_n)_{j_1, \dots, j_n}$ ist das j_1, \dots, j_n -te Element mit

$$T(K_1, \dots, K_n)_{j_1, \dots, j_n} = P(k_{1, j_1}, \dots, k_{n, j_n}).$$

- (ii) $\sum_{1 \leq j_1 \leq |K_1|, \dots, 1 \leq j_n \leq |K_n|} T(K_1, \dots, K_n)_{j_1, \dots, j_n} = 1$.

heißt *Tabelle gemeinsamer Wahrscheinlichkeiten (joint probability table (JPT))*, die $P(K_1, \dots, K_n)$ repräsentiert. Sie gibt die Wahrscheinlichkeit für das Eintreffen einer Hypothesenkombination $(k_{1, j_1}, \dots, k_{n, j_n})$ der Knoten K_1, \dots, K_n an ($1 \leq j_\ell \leq |K_\ell|$, ($\ell = 1, \dots, n$)). □

Um bei einer Tabelle gemeinsamer Wahrscheinlichkeiten sicher zu stellen, daß sich die Tabelleneinträge insgesamt zu Eins addieren, kann die Tabelle mit einem Normalisierungsfaktor versehen werden. Dieser Normalisierungsfaktor wird auch als *Normalisierungskonstante* α bezeichnet.

Definition B.1.4 (Wahrscheinlichkeitentabelle)

Die Tabellen bedingter Wahrscheinlichkeiten und die Tabellen gemeinsamer Wahrscheinlichkeiten werden zusammenfassend auch als *Wahrscheinlichkeitentabellen* bezeichnet. □

Beispiel B.1.5

Die *Wahrscheinlichkeitsverteilung* $P(A)$ des Knotens A mit den *Zuständen* (*Hypothesen*) a_1, \dots, a_n ist eine eindimensionale Wahrscheinlichkeitentabelle über dem Knoten A : Sei $P(A)$ gegeben durch:

$$P(a_1) = x_1, \dots, P(a_n) = x_n$$

Dann ergibt sich für die Wahrscheinlichkeitentabelle des Knotens A mit den Hypothesen a_1, \dots, a_n :

$$T(A) = \left[\begin{array}{ccc} a_1 & \dots & a_n \\ x_1 & \dots & x_n \end{array} \right], \text{ d.h.}$$

der Zustand a_i des Knotens A trifft mit der Wahrscheinlichkeit x_i ein. □

Definition B.1.6 (Gleichheit von Tabellen)

Zwei Tabellen $T(K_1, \dots, K_n)$ und $T(V_1, \dots, V_n)$ heißen gleich, wenn sie über denselben Knotenmengen gebildet wurden und die sich entsprechenden Tabelleneinträge in beiden Tabellen identisch sind. Man schreibt: $T(K_1, \dots, K_n) = T(V_1, \dots, V_n)$. □

Jetzt folgen die verschiedenen Operationen, die auf den Tabellen und die mit den Tabellen notwendig sind. Dazu zählen die *Multiplikation*, die *Division* sowie die *Marginalisation* von Tabellen.

Definition B.1.7 (Multiplikation von Tabellen)

Seien $T_{\mathbf{V}}$ und $T_{\mathbf{W}}$ zwei beliebige Tabellen über den Knotenmengen $\mathbf{V} = \{A_1, \dots, A_a, C_1, \dots, C_c\}$ und $\mathbf{W} = \{B_1, \dots, B_b, C_1, \dots, C_c\}$ mit $a + c = |\mathbf{V}|$, $b + c = |\mathbf{W}|$ und $\mathbf{V} \cap \mathbf{W} = \{C_1, \dots, C_c\}$, so nennt man die gemäß

$$\begin{aligned} K^{\mathbf{V}} \times K^{\mathbf{W}} &\rightarrow K^{\mathbf{V} \cup \mathbf{W}}, \\ (T_{\mathbf{V}}, T_{\mathbf{W}}) &\mapsto T_{\mathbf{V} \cup \mathbf{W}} = T_{\mathbf{V}} \cdot T_{\mathbf{W}} \end{aligned}$$

mit

$$\begin{aligned} T(A_1, \dots, A_a, B_1, \dots, B_b, C_1, \dots, C_c)_{r_1, \dots, r_a, s_1, \dots, s_b, t_1, \dots, t_c} = \\ = T(A_1, \dots, A_a, C_1, \dots, C_c)_{r_1, \dots, r_a, t_1, \dots, t_c} \cdot T(B_1, \dots, B_b, C_1, \dots, C_c)_{s_1, \dots, s_b, t_1, \dots, t_c} \end{aligned}$$

gebildete Tabelle über der Knotenmenge $\mathbf{V} \cup \mathbf{W}$ das Produkt von $T_{\mathbf{V}}$ und $T_{\mathbf{W}}$. □

Bei der Multiplikation von zwei Tabellen entsteht somit eine Tabelle, die um die Dimension der Knoten wächst, die nicht in beiden Tabellen enthalten sind.

Im Folgenden ist hierzu ein Beispiel angegeben:

Beispiel B.1.8 (Tabellenmultiplikation)

$$\text{Sei } T(K_1, K_2) = \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} & x_{2,1} \\ k_{2,2} & x_{1,2} & x_{2,2} \end{array} \right] \text{ und } T(K_1, K_3) = \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & y_{1,1} & y_{2,1} \\ k_{3,2} & y_{1,2} & y_{2,2} \end{array} \right].$$

Dann gilt:

$$\begin{aligned} T(K_1, K_2) \cdot T(K_1, K_3) &= \\ &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} & x_{2,1} \\ k_{2,2} & x_{1,2} & x_{2,2} \end{array} \right] \cdot \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & y_{1,1} & y_{2,1} \\ k_{3,2} & y_{1,2} & y_{2,2} \end{array} \right] = \\ &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & k_{2,1} & \begin{array}{l} z_{1,1,1} = x_{1,1} \cdot y_{1,1} \\ z_{1,1,2} = x_{1,2} \cdot y_{1,1} \end{array} \\ k_{3,2} & k_{2,2} & \begin{array}{l} z_{2,1,1} = x_{2,1} \cdot y_{1,1} \\ z_{2,1,2} = x_{2,1} \cdot y_{1,2} \\ z_{2,2,1} = x_{2,2} \cdot y_{1,1} \\ z_{2,2,2} = x_{2,2} \cdot y_{1,2} \end{array} \end{array} \right] = T(K_1, K_2, K_3). \end{aligned}$$

Bsp

Die *Division* einer beliebigen Tabelle $T_{\mathbf{V}}$ durch eine andere beliebige Tabelle $T_{\mathbf{W}}$ verläuft ähnlich wie die Multiplikation zweier Tabellen, wobei man Vorsicht walten lassen muß, wenn der Divisor Nullen als Tabelleneinträge enthält:

Definition B.1.9 (Division von Tabellen)

Seien $T_{\mathbf{V}}$ und $T_{\mathbf{W}}$ zwei beliebige Tabellen über den Knotenmengen $\mathbf{V} = \{A_1, \dots, A_a, C_1, \dots, C_c\}$ und $\mathbf{W} = \{B_1, \dots, B_b, C_1, \dots, C_c\}$ mit $a + c = |\mathbf{V}|$, $b + c = |\mathbf{W}|$ und $\mathbf{V} \cap \mathbf{W} = \{C_1, \dots, C_c\}$, so nennt man die gemäß

$$\begin{aligned} K^{\mathbf{V}} \times K^{\mathbf{W}} &\rightarrow K^{\mathbf{V} \cup \mathbf{W}}, \\ (T_{\mathbf{V}}, T_{\mathbf{W}}) &\mapsto T_{\mathbf{V} \cup \mathbf{W}} = \frac{T_{\mathbf{V}}}{T_{\mathbf{W}}} \end{aligned}$$

mit

$$\begin{aligned} T(A_1, \dots, A_a, B_1, \dots, B_b, C_1, \dots, C_c)_{r_1, \dots, r_a, s_1, \dots, s_b, t_1, \dots, t_c} &= \\ &= \frac{T(A_1, \dots, A_a, C_1, \dots, C_c)_{r_1, \dots, r_a, t_1, \dots, t_c}}{T(B_1, \dots, B_b, C_1, \dots, C_c)_{s_1, \dots, s_b, t_1, \dots, t_c}} = \frac{\text{Dividend}}{\text{Divisor}} = \\ &= \begin{cases} 0 & \text{Dividend} = 0 \wedge \text{Divisor} = 0, \\ \frac{\text{Dividend}}{\text{Divisor}} & \text{Divisor} \neq 0, \\ * \text{ (nicht definiert) } & \text{sonst} \end{cases} \end{aligned}$$

gebildete Tabelle über der Knotenmenge $\mathbf{V} \cup \mathbf{W}$ den Quotienten von $T_{\mathbf{V}}$ und $T_{\mathbf{W}}$. **D**

Dieses soll an einem Beispiel verdeutlicht werden:

Beispiel B.1.10 (Tabellendivision)

$$\begin{aligned} \text{Sei } T(K_1, K_2) &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} & x_{2,1} \\ k_{2,2} & x_{1,2} & x_{2,2} \end{array} \right] = \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & 0 & 4 \\ k_{2,2} & 2 & 8 \end{array} \right] \\ \text{und } T(K_1, K_3) &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & y_{1,1} & y_{2,1} \\ k_{3,2} & y_{1,2} & y_{2,2} \end{array} \right] = \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & 0 & 2 \\ k_{3,2} & 1 & 4 \end{array} \right]. \end{aligned}$$

Dann gilt:

$$\begin{aligned}
 T(K_1, K_2)/T(K_1, K_3) &= \frac{T(K_1, K_2)}{T(K_1, K_3)} = \\
 &= \frac{\left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} & x_{2,1} \\ k_{2,2} & x_{1,2} & x_{2,2} \end{array} \right]}{\left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & y_{1,1} & y_{2,1} \\ k_{3,2} & y_{1,2} & y_{2,2} \end{array} \right]} = \frac{\left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & 0 & 4 \\ k_{2,2} & 2 & 8 \end{array} \right]}{\left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & 0 & 2 \\ k_{3,2} & 1 & 4 \end{array} \right]} = \\
 &= \left[\begin{array}{cc|cc} & & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & k_{2,1} & z_{1,1,1} = x_{1,1}/y_{1,1} & z_{2,1,1} = x_{2,1}/y_{2,1} \\ & k_{2,2} & z_{1,2,1} = x_{1,2}/y_{1,1} & z_{2,2,1} = x_{2,2}/y_{2,1} \\ k_{3,2} & k_{2,1} & z_{1,1,2} = x_{1,1}/y_{1,2} & z_{2,1,2} = x_{2,1}/y_{2,2} \\ & k_{2,2} & z_{1,2,2} = x_{1,2}/y_{1,2} & z_{2,2,2} = x_{2,2}/y_{2,2} \end{array} \right] = T(K_1, K_2, K_3) = \\
 &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & k_{2,1} & 0 & 2 \\ & k_{2,2} & * & 4 \\ k_{3,2} & k_{2,1} & 0 & 1 \\ & k_{2,2} & 2 & 2 \end{array} \right].
 \end{aligned}$$

Die Tabelle ist insgesamt nicht definiert, da ein Tabelleneintrag (mit * markiert) nicht definiert ist. **Bsp**

Definition B.1.11 (Marginalisation)

Sei $T_{\mathbf{V}}$ eine Tabelle über der Knotenmenge $\mathbf{V} = \{A_1, \dots, A_a, B_1, \dots, B_b\}$ ($a \in \mathbb{N}^*, b \in \mathbb{N}$) und $\mathbf{W} = \{A_1, \dots, A_a\} \subseteq \mathbf{V}$ eine Teilmenge von \mathbf{V} , so nennt man die gemäß

$$\begin{aligned}
 K^{\mathbf{V}} &\rightarrow K^{\mathbf{W}}, \\
 T_{\mathbf{V}} &\mapsto T_{\mathbf{W}} = \sum_{\mathbf{V} \setminus \mathbf{W}} T_{\mathbf{V}}
 \end{aligned}$$

mit

$$T(A_1, \dots, A_a)_{r_1, \dots, r_a} = \sum_{(s_1, \dots, s_b) = (1, \dots, 1)}^{(|B_1|, \dots, |B_b|)} T(A_1, \dots, A_a, B_1, \dots, B_b)_{r_1, \dots, r_a, s_1, \dots, s_b}$$

gebildete Tabelle $T_{\mathbf{W}}$ die *Marginalisation* von $T_{\mathbf{V}}$. **D**

Beispiel B.1.12 (Marginalisation)

Wenn $P(A, B_1, \dots, B_n)$ gegeben ist, dann berechnet sich die Wahrscheinlichkeitsverteilung $P(A)$ wie folgt:

$$P(a_i) = \sum_{(j_1, \dots, j_n) = (0, \dots, 0)}^{(|B_1|, \dots, |B_n|)} P(a_i, b_{1j_1}, \dots, b_{nj_n}) \quad , \quad i = 1, \dots, |A|.$$

Die gesamten Einträge einer Tabelle gemeinsamer Wahrscheinlichkeiten summieren sich zu 1. So kann man bei der Wahrscheinlichkeitsverteilung $P(A)$ eines Knotens A auch von einer Tabelle gemeinsamer Wahrscheinlichkeiten reden. **Bsp**

Eine wichtige Eigenschaft der Marginalisation wird im folgenden Satz beschrieben:

Satz B.1.13

Seien \mathbf{V} und \mathbf{W} disjunkte Knotenmengen, d.h. $\mathbf{V} \cap \mathbf{W} = \emptyset$, $\mathbf{U} \subseteq \mathbf{V}$ und seien $T_{\mathbf{V}}$ und $T_{\mathbf{W}}$ Tabellen über diesen Knotenmengen. Dann gilt:

$$\sum_{\mathbf{U}} (T_{\mathbf{W}} \cdot T_{\mathbf{V}}) = T_{\mathbf{W}} \cdot \sum_{\mathbf{U}} T_{\mathbf{V}}$$

□

Ein Beweis zu diesem Satz läßt sich leicht herleiten und bleibt dem interessierten Leser überlassen. Tabellen können also aus der Marginalisation herausgeholt werden, wenn sie keine Knoten enthalten, über die marginalisiert wird. Dies wird im folgenden Beispiel B.1.14 illustriert.

Beispiel B.1.14 (Produkt-Marginalisation)

$$\text{Sei } T(K_1, K_2) = \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} & x_{2,1} \\ k_{2,2} & x_{1,2} & x_{2,2} \end{array} \right] \text{ und } T(K_1, K_3) = \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & y_{1,1} & y_{2,1} \\ k_{3,2} & y_{1,2} & y_{2,2} \end{array} \right].$$

Das Produkt beider Tabellen berechnet sich zu

$$T(K_1, K_2) \cdot T(K_1, K_3) = \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{3,1} & k_{2,1} & x_{1,1} \cdot y_{1,1} & x_{2,1} \cdot y_{2,1} \\ & k_{2,2} & x_{1,2} \cdot y_{1,1} & x_{2,2} \cdot y_{2,1} \\ k_{3,2} & k_{2,1} & x_{1,1} \cdot y_{1,2} & x_{2,1} \cdot y_{2,2} \\ & k_{2,2} & x_{1,2} \cdot y_{1,2} & x_{2,2} \cdot y_{2,2} \end{array} \right] = T(K_1, K_2, K_3).$$

Es gilt:

$$\begin{aligned} \sum_{K_3} T(K_1, K_2) \cdot T(K_1, K_3) &= \sum_{K_3} T(K_1, K_2, K_3) = \\ &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} \cdot y_{1,1} + x_{1,1} \cdot y_{1,2} & x_{2,1} \cdot y_{2,1} + x_{2,1} \cdot y_{2,2} \\ k_{2,2} & x_{1,2} \cdot y_{1,1} + x_{1,2} \cdot y_{1,2} & x_{2,2} \cdot y_{2,1} + x_{2,2} \cdot y_{2,2} \end{array} \right] = \\ &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} \cdot (y_{1,1} + y_{1,2}) & x_{2,1} \cdot (y_{2,1} + y_{2,2}) \\ k_{2,2} & x_{1,2} \cdot (y_{1,1} + y_{1,2}) & x_{2,2} \cdot (y_{2,1} + y_{2,2}) \end{array} \right] = \\ &= \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline k_{2,1} & x_{1,1} & x_{2,1} \\ k_{2,2} & x_{1,2} & x_{2,2} \end{array} \right] \cdot \left[\begin{array}{c|cc} & k_{1,1} & k_{1,2} \\ \hline & y_{1,1} + y_{1,2} & y_{2,1} + y_{2,2} \end{array} \right] = T(K_1, K_2) \cdot \sum_{K_3} T(K_1, K_3). \end{aligned}$$

Bei der Berechnung von $\sum_{K_3} T(K_1, K_2) \cdot T(K_1, K_3)$ sind 8 Multiplikationen und 4 Additionen notwendig, wohingegen bei der Berechnung des Termes von $T(K_1, K_2) \cdot \sum_{K_3} T(K_1, K_3)$ nur 4 Multiplikationen und 2 Additionen anfallen. Durch geschicktes Herausziehen von Faktoren aus der Marginalisation wird also der Rechenaufwand enorm verringert. □

B.2 Multivariate Polynome

Für den geneigten Leser werden im Folgenden die *multivariaten Polynome* formal definiert und eine Kurzschreibweise eingeführt.

Dann werden mit der Kurzschreibweise notwendige Operationen zwischen multivariaten Polynomen definiert. Dies sind die Addition und die Multiplikation zweier multivariater Polynome miteinander und die Skalarmultiplikation von einer Konstanten mit einem multivariaten Polynom. Für das Verständnis des Lesers trägt die Kurzschreibweise bei, wenn er sich im Anfang bei der Definition der Operationen auf die Betrachtung von "normalen" Polynomen mit einer Variablen (den sogenannten *univariaten Polynomen*) einschränkt. Für die Handhabung der Polynome in einem Rechner wird der Begriff der *Ordnung* eingeführt. Die Polynome können dadurch so in einer Form im Rechner dargestellt werden, daß schnell Vereinfachungen möglich sind. Zum Beispiel läßt sich das Polynom $x \cdot y \cdot z + z \cdot x \cdot y$ vereinfacht als $2 \cdot x \cdot y \cdot z$ darstellen.

Dazu im folgenden ein paar Definitionen und Schreibweisen für Polynome.

Definition B.2.1 (Polynom)

Sei R ein kommutativer Ring mit 1. Dann ist

$$f = f(x_1, \dots, x_n) = \sum_{\kappa_1 \geq 0, \dots, \kappa_n \geq 0} a_{\kappa_1, \dots, \kappa_n} \cdot x_1^{\kappa_1} \cdot \dots \cdot x_n^{\kappa_n}$$

ein Polynom in den Unbestimmten x_1, \dots, x_n mit Koeffizienten a_κ in R .

Man schreibt auch:

$$f = \sum_{\kappa, a_\kappa \neq 0} a_\kappa \cdot X^\kappa.$$

Ist $n = 1$, so redet man von *univariaten* Polynomen. Im Falle $n > 1$ heißen die Polynome *multivariat*. **D**

Für die Kurzschreibweise von Polynomen werden *Multiindizes* eingeführt.

Bemerkung B.2.2 (Multiindizes)

In der vorhergehenden Definition wird κ als ein *Multiindex* verwendet. Es gilt:

- (i) $\kappa = (\kappa_1, \dots, \kappa_n) \in \mathbb{N}^n$,
- (ii) $\kappa + \lambda = (\kappa_1 + \lambda_1, \dots, \kappa_n + \lambda_n)$,
- (iii) $|\kappa| = \kappa_1 + \dots + \kappa_n$,
- (iv) $a_\kappa = a_{\kappa_1, \dots, \kappa_n}$ und
- (v) $X^\kappa = x_1^{\kappa_1} \cdot \dots \cdot x_n^{\kappa_n}$.

Bem

Jetzt muß noch geklärt werden, wie zwei Polynome miteinander addiert bzw. multipliziert werden. Bezüglich der Multiplikation von Polynomen ist zu beachten, daß es zu einem nichtkonstanten Polynom kein Inverses existiert. Die Menge der Polynome zusammen mit den Operationen $+$ und \cdot bildet also keinen Körper; man spricht daher von einem *Polynomring* $(R[x_1, \dots, x_n], +, \cdot)$.

Definition B.2.3 (Polynomring)

Sei R kommutativer Ring mit 1. Die Menge aller Polynome in x_1, \dots, x_n mit Koeffizienten in R ($R[x_1, \dots, x_n]$) bildet zusammen mit

$$\begin{aligned} \sum_{\kappa} a_{\kappa} X^{\kappa} + \sum_{\kappa} b_{\kappa} X^{\kappa} &:= \sum_{\kappa} (a_{\kappa} + b_{\kappa}) X^{\kappa} \\ \left(\sum_{\kappa} a_{\kappa} X^{\kappa} \right) \cdot \left(\sum_{\lambda} b_{\lambda} X^{\lambda} \right) &:= \sum_{\mu} \left(\sum_{\kappa+\lambda=\mu} a_{\kappa} b_{\lambda} \right) X^{\mu} \\ \nu \left(\sum_{\kappa} a_{\kappa} X^{\kappa} \right) &:= \sum_{\kappa} \nu a_{\kappa} X^{\kappa} \quad , \quad \nu \in R \end{aligned}$$

einen *Polynomring* (kommutative R -Algebra mit 1). □

Für den Rechner sind die beiden Polynome $2 * x + 6 * y$ und $2 * (3 * y + x)$ verschieden, obwohl sie dieselben Ergebnisse liefern. Es ist also notwendig, daß eine eindeutige Repräsentation der Polynome gewährleistet wird. Wenn Elemente einer Menge miteinander vergleichbar sind, so spricht man von einer *teilweisen Ordnung*.

Definition B.2.4 (teilweise Ordnung)

Sei X eine Menge. Eine Relation $R = X \times X$ heißt *teilweise Ordnung* auf X , falls gilt:

- (i) $(x, x) \in R \quad \forall x \in X$
- (ii) $(x, y) \in R \wedge (y, x) \in R \Rightarrow y = x$
- (iii) $(x, y) \in R \wedge (y, z) \in R \Rightarrow (x, z) \in R$

Wir sprechen dann auch von einer *teilweisen Ordnung* \succ (abkürzend für \succ_R) auf R und schreiben $x \preceq y$ oder $y \succeq x$, falls $(x, y) \in R$ bzw. $x \prec y$ oder $y \succ x$, falls $x \preceq y$ und $x \neq y$. □

Für den besonderen Fall, daß alle Elemente einer Menge untereinander vergleichbar sind, so bezeichnet man die *teilweise Ordnung* auch als *totale Ordnung*.

Definition B.2.5 (totale Ordnung)

Eine *teilweise Ordnung* \succ auf X heißt *totale Ordnung*, falls sich je zwei Elemente von X bzgl. \succ vergleichen lassen:

$$x, y \in X \Rightarrow x \prec y \text{ oder } x = y \text{ oder } x \succ y.$$

□

B.3 Nebenrechnungen

Beispiel B.3.1 (Asienbesuch (Shafer-Shenoy Skripte))

Hier sind die Nebenrechnungen zum Beispiel 1.3.12 aufgeführt.

Ist man an der ψ -Tabelle der Clique Cl_{q_1} interessiert, so erhält man:

$$T_{\psi(Cl_{q_1})} = T_{\text{cpt}(D)} \cdot T_{Cl_{q_4} \triangleright Cl_{q_1}}$$

$$\begin{aligned}
&= T_{\text{cpt}(D)} \cdot \left(\sum_{EG} T_{\psi(Cl_{q_4}) \setminus Cl_{q_1}} \right) \\
&= T_{\text{cpt}(D)} \cdot \left(\sum_{EG} T_{Cl_{q_2} \triangleright Cl_{q_4}} \cdot T_{Cl_{q_3} \triangleright Cl_{q_4}} \cdot T_{Cl_{q_5} \triangleright Cl_{q_4}} \right) \\
&= T_{\text{cpt}(D)} \cdot \left(\sum_{EG} \left(\sum_H T_{\psi(Cl_{q_2}) \setminus Cl_{q_4}} \right) \cdot \left(\sum_F T_{\psi(Cl_{q_3}) \setminus Cl_{q_4}} \right) \cdot \right. \\
&\quad \left. \left(\sum_B T_{\psi(Cl_{q_5}) \setminus Cl_{q_4}} \right) \right) \\
&= T_{\text{cpt}(D)} \cdot \left(\sum_{EG} \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \right. \\
&\quad \left. \left(\sum_B T_{\text{cpt}(C)} \cdot T_{Cl_{q_6} \triangleright Cl_{q_5}} \right) \right) \\
&= T_{\text{cpt}(D)} \cdot \left(\sum_{EG} \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \right. \\
&\quad \left. \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\psi(Cl_{q_6}) \setminus Cl_{q_5}} \right) \right) \right) \\
&= T_{\text{cpt}(D)} \cdot \left(\sum_{EG} \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \right. \\
&\quad \left. \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right)
\end{aligned}$$

Die ψ -Tabelle der Clique Cl_{q_2} berechnet sich wie folgt:

$$\begin{aligned}
T_{\psi(Cl_{q_2})} &= T_{\text{cpt}(H)} \cdot T_{Cl_{q_4} \triangleright Cl_{q_2}} \\
&= T_{\text{cpt}(H)} \cdot \left(\sum_E T_{\psi(Cl_{q_4}) \setminus Cl_{q_2}} \right) \\
&= T_{\text{cpt}(H)} \cdot \left(\sum_E T_{Cl_{q_1} \triangleright Cl_{q_4}} \cdot T_{Cl_{q_3} \triangleright Cl_{q_4}} \cdot T_{Cl_{q_5} \triangleright Cl_{q_4}} \right) \\
&= T_{\text{cpt}(H)} \cdot \left(\sum_E \left(\sum_D T_{\psi(Cl_{q_1}) \setminus Cl_{q_4}} \right) \cdot \right. \\
&\quad \left(\sum_F T_{\psi(Cl_{q_3}) \setminus Cl_{q_4}} \right) \cdot \\
&\quad \left. \left(\sum_B T_{\psi(Cl_{q_5}) \setminus Cl_{q_4}} \right) \right) \\
&= T_{\text{cpt}(H)} \cdot \left(\sum_E \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \right. \\
&\quad \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \\
&\quad \left. \left(\sum_B T_{\text{cpt}(C)} \cdot T_{Cl_{q_6} \triangleright Cl_{q_5}} \right) \right) \\
&= T_{\text{cpt}(H)} \cdot \left(\sum_E \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \right. \\
&\quad \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \\
&\quad \left. \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\psi(Cl_{q_6}) \setminus Cl_{q_5}} \right) \right) \right)
\end{aligned}$$

$$\begin{aligned}
&= T_{\text{cpt}(H)} \cdot \left(\sum_E \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \right. \\
&\quad \left. \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \right. \\
&\quad \left. \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right)
\end{aligned}$$

Die ψ -Tabelle der Clique Clq_3 berechnet sich zu:

$$\begin{aligned}
T_{\psi(Clq_3)} &= T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot T_{Clq_4 \triangleright Clq_3} \\
&= T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot \left(\sum_C T_{\psi(Clq_4) \setminus Clq_3} \right) \\
&= T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot \left(\sum_C T_{Clq_1 \triangleright Clq_4} \cdot T_{Clq_2 \triangleright Clq_4} \cdot T_{Clq_5 \triangleright Clq_4} \right) \\
&= T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot \left(\sum_C \left(\sum_D T_{\psi(Clq_1) \setminus Clq_4} \right) \cdot \right. \\
&\quad \left. \left(\sum_H T_{\psi(Clq_2) \setminus Clq_4} \right) \cdot \right. \\
&\quad \left. \left(\sum_B T_{\psi(Clq_5) \setminus Clq_4} \right) \right) \\
&= T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot \left(\sum_C \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \right. \\
&\quad \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left. \left(\sum_B T_{\text{cpt}(C)} \cdot T_{Clq_6 \triangleright Clq_5} \right) \right) \\
&= T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \cdot \left(\sum_C \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \right. \\
&\quad \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left. \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right) \right)
\end{aligned}$$

Die ψ -Tabelle der Clique Clq_4 ist:

$$\begin{aligned}
T_{\psi(Clq_4)} &= T_{Clq_1 \triangleright Clq_4} \cdot T_{Clq_2 \triangleright Clq_4} \cdot T_{Clq_3 \triangleright Clq_4} \cdot T_{Clq_5 \triangleright Clq_4} \\
&= \left(\sum_D T_{\psi(Clq_1) \setminus Clq_4} \right) \cdot \left(\sum_H T_{\psi(Clq_2) \setminus Clq_4} \right) \cdot \left(\sum_F T_{\psi(Clq_3) \setminus Clq_4} \right) \cdot \\
&\quad \left(\sum_B T_{\psi(Clq_5) \setminus Clq_4} \right) \\
&= \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \\
&\quad \left(\sum_B T_{\text{cpt}(C)} \cdot T_{Clq_6 \triangleright Clq_5} \right) \\
&= \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \\
&\quad \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\psi(Clq_6) \setminus Clq_5} \right) \right)
\end{aligned}$$

$$\begin{aligned}
&= \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \cdot \\
&\quad \left(\sum_B T_{\text{cpt}(C)} \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \right)
\end{aligned}$$

Die ψ -Tabelle der Clique Clq_5 ist:

$$\begin{aligned}
T_{\psi(Clq_5)} &= T_{\text{cpt}(C)} \cdot T_{Clq_4 \triangleright Clq_5} \cdot T_{Clq_6 \triangleright Clq_5} \\
&= T_{\text{cpt}(C)} \cdot \left(\sum_G T_{\psi(Clq_4) \setminus Clq_5} \right) \cdot \left(\sum_A T_{\psi(Clq_6) \setminus Clq_5} \right) \\
&= T_{\text{cpt}(C)} \cdot \left(\sum_G T_{\psi(Clq_4) \setminus Clq_5} \right) \cdot \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \\
&= T_{\text{cpt}(C)} \cdot \left(\sum_G T_{Clq_1 \triangleright Clq_4} \cdot T_{Clq_2 \triangleright Clq_4} \cdot T_{Clq_3 \triangleright Clq_4} \right) \cdot \\
&\quad \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \\
&= T_{\text{cpt}(C)} \cdot \left(\sum_G \left(\sum_D T_{Clq_1 \setminus Clq_4} \right) \cdot \left(\sum_H T_{Clq_2 \setminus Clq_4} \right) \cdot \left(\sum_F T_{Clq_3 \setminus Clq_4} \right) \right) \cdot \\
&\quad \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right) \\
&= T_{\text{cpt}(C)} \cdot \left(\sum_G \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \right) \cdot \\
&\quad \left(\sum_A T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \right)
\end{aligned}$$

Ist man an der ψ -Tabelle der Clique Clq_6 interessiert, so erhält man:

$$\begin{aligned}
T_{\psi(Clq_6)} &= T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot T_{Clq_5 \triangleright Clq_6} \\
&= T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot \left(\sum_{CE} T_{\psi(Clq_5) \setminus Clq_6} \right) \\
&= T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot \left(\sum_{CE} T_{\text{cpt}(C)} \cdot T_{Clq_4 \triangleright Clq_5} \right) \\
&= T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot \left(\sum_{CE} T_{\text{cpt}(C)} \cdot \left(\sum_G T_{\psi(Clq_4) \setminus Clq_5} \right) \right) \\
&= T_{\text{cpt}(A)} \cdot T_{\text{cpt}(B)} \cdot \left(\sum_{CE} T_{\text{cpt}(C)} \cdot \left(\sum_G \left(\sum_D T_{\text{cpt}(D)} \right) \cdot \right. \right. \\
&\quad \left. \left. \left(\sum_H T_{\text{cpt}(H)} \right) \cdot \right. \right. \\
&\quad \left. \left. \left(\sum_F T_{\text{cpt}(E)} \cdot T_{\text{cpt}(F)} \cdot T_{\text{cpt}(G)} \right) \right) \right)
\end{aligned}$$

Beispiel B.3.2 (Polynom-Rollup)

Hier ist der BEL-Wert vom Knoten N_1 noch nicht faktorisiert dargestellt.

$$\begin{aligned}
& \text{BEL}(N_1) = \\
& \left(\begin{aligned}
& (0.07200 S_{0,1} W_{0,1} S_{1,1} W_{1,1} + 0.04320 S_{0,2} W_{0,1} S_{1,1} W_{1,1} + \\
& 0.01920 S_{0,3} W_{0,1} S_{1,1} W_{1,1} + 0.07200 S_{0,1} W_{0,2} S_{1,1} W_{1,1} + \\
& 0.02880 S_{0,2} W_{0,2} S_{1,1} W_{1,1} + 0.00480 S_{0,3} W_{0,2} S_{1,1} W_{1,1} + \\
& 0.04320 S_{0,1} W_{0,1} S_{1,2} W_{1,1} + 0.02592 S_{0,2} W_{0,1} S_{1,2} W_{1,1} + \\
& 0.01152 S_{0,3} W_{0,1} S_{1,2} W_{1,1} + 0.04320 S_{0,1} W_{0,2} S_{1,2} W_{1,1} + \\
& 0.01728 S_{0,2} W_{0,2} S_{1,2} W_{1,1} + 0.00288 S_{0,3} W_{0,2} S_{1,2} W_{1,1} + \\
& 0.01920 S_{0,1} W_{0,1} S_{1,3} W_{1,1} + 0.01152 S_{0,2} W_{0,1} S_{1,3} W_{1,1} + \\
& 0.00512 S_{0,3} W_{0,1} S_{1,3} W_{1,1} + 0.01920 S_{0,1} W_{0,2} S_{1,3} W_{1,1} + \\
& 0.00768 S_{0,2} W_{0,2} S_{1,3} W_{1,1} + 0.00128 S_{0,3} W_{0,2} S_{1,3} W_{1,1} + \\
& 0.07200 S_{0,1} W_{0,1} S_{1,1} W_{1,2} + 0.04320 S_{0,2} W_{0,1} S_{1,1} W_{1,2} + \\
& 0.01920 S_{0,3} W_{0,1} S_{1,1} W_{1,2} + 0.07200 S_{0,1} W_{0,2} S_{1,1} W_{1,2} + \\
& 0.02880 S_{0,2} W_{0,2} S_{1,1} W_{1,2} + 0.00480 S_{0,3} W_{0,2} S_{1,1} W_{1,2} + \\
& 0.02880 S_{0,1} W_{0,1} S_{1,2} W_{1,2} + 0.01728 S_{0,2} W_{0,1} S_{1,2} W_{1,2} + \\
& 0.00768 S_{0,3} W_{0,1} S_{1,2} W_{1,2} + 0.02880 S_{0,1} W_{0,2} S_{1,2} W_{1,2} + \\
& 0.01152 S_{0,2} W_{0,2} S_{1,2} W_{1,2} + 0.00192 S_{0,3} W_{0,2} S_{1,2} W_{1,2} + \\
& 0.00480 S_{0,1} W_{0,1} S_{1,3} W_{1,2} + 0.00288 S_{0,2} W_{0,1} S_{1,3} W_{1,2} + \\
& 0.00128 S_{0,3} W_{0,1} S_{1,3} W_{1,2} + 0.00480 S_{0,1} W_{0,2} S_{1,3} W_{1,2} + \\
& 0.00192 S_{0,2} W_{0,2} S_{1,3} W_{1,2} + 0.00032 S_{0,3} W_{0,2} S_{1,3} W_{1,2}) \\
& (0.00648 S_{0,1} W_{0,1} S_{1,1} W_{1,1} + 0.00432 S_{0,2} W_{0,1} S_{1,1} W_{1,1} + \\
& 0.00180 S_{0,3} W_{0,1} S_{1,1} W_{1,1} + 0.01512 S_{0,1} W_{0,2} S_{1,1} W_{1,1} + \\
& 0.00648 S_{0,2} W_{0,2} S_{1,1} W_{1,1} + 0.00180 S_{0,3} W_{0,2} S_{1,1} W_{1,1} + \\
& 0.00432 S_{0,1} W_{0,1} S_{1,2} W_{1,1} + 0.00288 S_{0,2} W_{0,1} S_{1,2} W_{1,1} + \\
& 0.00120 S_{0,3} W_{0,1} S_{1,2} W_{1,1} + 0.01008 S_{0,1} W_{0,2} S_{1,2} W_{1,1} + \\
& 0.00432 S_{0,2} W_{0,2} S_{1,2} W_{1,1} + 0.00120 S_{0,3} W_{0,2} S_{1,2} W_{1,1} + \\
& 0.00180 S_{0,1} W_{0,1} S_{1,3} W_{1,1} + 0.00120 S_{0,2} W_{0,1} S_{1,3} W_{1,1} + \\
& 0.00050 S_{0,3} W_{0,1} S_{1,3} W_{1,1} + 0.00420 S_{0,1} W_{0,2} S_{1,3} W_{1,1} + \\
& 0.00180 S_{0,2} W_{0,2} S_{1,3} W_{1,1} + 0.00050 S_{0,3} W_{0,2} S_{1,3} W_{1,1} + \\
& 0.01512 S_{0,1} W_{0,1} S_{1,1} W_{1,2} + 0.01008 S_{0,2} W_{0,1} S_{1,1} W_{1,2} + \\
& 0.00420 S_{0,3} W_{0,1} S_{1,1} W_{1,2} + 0.03528 S_{0,1} W_{0,2} S_{1,1} W_{1,2} + \\
& 0.01512 S_{0,2} W_{0,2} S_{1,1} W_{1,2} + 0.00420 S_{0,3} W_{0,2} S_{1,1} W_{1,2} + \\
& 0.00648 S_{0,1} W_{0,1} S_{1,2} W_{1,2} + 0.00432 S_{0,2} W_{0,1} S_{1,2} W_{1,2} + \\
& 0.00180 S_{0,3} W_{0,1} S_{1,2} W_{1,2} + 0.01512 S_{0,1} W_{0,2} S_{1,2} W_{1,2} + \\
& 0.00648 S_{0,2} W_{0,2} S_{1,2} W_{1,2} + 0.00180 S_{0,3} W_{0,2} S_{1,2} W_{1,2} + \\
& 0.00180 S_{0,1} W_{0,1} S_{1,3} W_{1,2} + 0.00120 S_{0,2} W_{0,1} S_{1,3} W_{1,2} + \\
& 0.00050 S_{0,3} W_{0,1} S_{1,3} W_{1,2} + 0.00420 S_{0,1} W_{0,2} S_{1,3} W_{1,2} + \\
& 0.00180 S_{0,2} W_{0,2} S_{1,3} W_{1,2} + 0.00050 S_{0,3} W_{0,2} S_{1,3} W_{1,2})
\end{aligned} \right) = \alpha \cdot \left(\begin{aligned}
& (0.30 S_{1,1} W_{1,1} + 0.30 S_{1,1} W_{1,2} + \\
& 0.18 S_{1,2} W_{1,1} + 0.12 S_{1,2} W_{1,2} + \\
& 0.08 S_{1,3} W_{1,1} + 0.02 S_{1,3} W_{1,2}) \\
& (0.18 S_{1,1} W_{1,1} + 0.42 S_{1,1} W_{1,2} + \\
& 0.12 S_{1,2} W_{1,1} + 0.18 S_{1,2} W_{1,2} + \\
& 0.05 S_{1,3} W_{1,1} + 0.05 S_{1,3} W_{1,2})
\end{aligned} \right) \cdot \left(\begin{aligned}
& (0.240 S_{0,1} W_{0,1} + 0.240 S_{0,1} W_{0,2} + \\
& 0.144 S_{0,2} W_{0,1} + 0.096 S_{0,2} W_{0,2} + \\
& 0.064 S_{0,3} W_{0,1} + 0.016 S_{0,3} W_{0,2}) \\
& (0.036 S_{0,1} W_{0,1} + 0.084 S_{0,1} W_{0,2} + \\
& 0.024 S_{0,2} W_{0,1} + 0.036 S_{0,2} W_{0,2} + \\
& 0.010 S_{0,3} W_{0,1} + 0.010 S_{0,3} W_{0,2})
\end{aligned} \right) =
\end{aligned}$$

$$= \alpha \underbrace{\begin{pmatrix} (0.30 S_{1,1} W_{1,1} + 0.30 S_{1,1} W_{1,2} + \\ 0.18 S_{1,2} W_{1,1} + 0.12 S_{1,2} W_{1,2} + \\ 0.08 S_{1,3} W_{1,1} + 0.02 S_{1,3} W_{1,2}) \\ (0.18 S_{1,1} W_{1,1} + 0.42 S_{1,1} W_{1,2} + \\ 0.12 S_{1,2} W_{1,1} + 0.18 S_{1,2} W_{1,2} + \\ 0.05 S_{1,3} W_{1,1} + 0.05 S_{1,3} W_{1,2}) \end{pmatrix}}_{\text{Zeitscheibe 1}} \cdot \underbrace{\begin{pmatrix} (0.30 S_{0,1} W_{0,1} + 0.30 S_{0,1} W_{0,2} + \\ 0.18 S_{0,2} W_{0,1} + 0.12 S_{0,2} W_{0,2} + \\ 0.08 S_{0,3} W_{0,1} + 0.02 S_{0,3} W_{0,2}) \\ (0.18 S_{0,1} W_{0,1} + 0.42 S_{0,1} W_{0,2} + \\ 0.12 S_{0,2} W_{0,1} + 0.18 S_{0,2} W_{0,2} + \\ 0.05 S_{0,3} W_{0,1} + 0.05 S_{0,3} W_{0,2}) \end{pmatrix}}_{\text{Zeitscheibe 0}} \cdot \underbrace{\begin{pmatrix} 0.8 \\ 0.2 \end{pmatrix}}_{\text{Init}}.$$

BEL(N_0)

Bsp

Anhang C

RUBIN

Im folgenden werden einige Berechnungsvorgänge des implementierten Programmes *RUBIN* dokumentiert. Diese Dokumentation geschieht mittels kommentierten Befehlsaufrufen und Befehlsausgaben und Bildschirmabzügen an ausgewählten Beispielen, die in dieser Arbeit in vorangegangenen Abschnitten vorgestellt wurden.

Die Inferenzalgorithmen (wie zum Beispiel die Shafer-Shenoy Architektur oder die Polynom-Propagierung) wurden in LISP [Steele 90; Norvig 92; Graham 94; Norvig & Pitman 93] implementiert und die Oberfläche in tcl/tk [Welch 97; Raines 98; Johnson 97] geschrieben.

C.1 Mehrfach verbundene Bayessche Netze

In diesem Abschnitt werden die Berechnungsvorgänge an mehrfach verbundenen Bayesschen Netzen vorgeführt.

Beispiel C.1.1 (Hypothesenparametrisierung)

Für die Hypothesenparametrisierung (siehe das Beispiel 1.3.13) wird das Bayessche Netz *Gehirntumor* als Beispielnetz herangezogen (siehe Beispiel 1.1.1, Abbildung 1.1 und Tabelle 1.1). Für die Definition eines Knotens K in einem Schema werden die folgenden Angaben benötigt:

- sein eindeutiger Name, der im Schema nur einmal vorkommt
- die Anzahl der Hypothesen des Knotens
- die Namen der Hypothesen des Knotens
- die Elternknoten, die er besitzt, wenn sein Schema als erste Zeitscheibe ($t = 0$) instantiiert wird
- die Wahrscheinlichkeitentabelle (a-priori oder bedingte Wahrscheinlichkeiten), die er besitzt, wenn sein Schema als erste Zeitscheibe ($t = 0$) instantiiert wird
- die Elternknoten, die er besitzt, wenn sein Schema als Zeitscheibe $t > 0$ instantiiert wird

- die Wahrscheinlichkeitentabelle (a-priori oder bedingte Wahrscheinlichkeiten), die er besitzt, wenn sein Schema als Zeitscheibe $t > 0$ instantiiert wird

Diese Angaben sind in der folgenden Datei enthalten, die die Knoten des Bayesschen Netzes **Gehirntumor** definiert. In dieser Datei ist also nicht nur die Struktur des Bayesschen Netzes enthalten, sondern auch alle a-priori und bedingten Wahrscheinlichkeiten der Knoten. Exemplarisch sind nur die beiden Knoten *A* und *D* aufgeführt.

```

1   ;;; Modell fuer mehrfach verknuepfte Bayessche Netze
2   ;;; Boris Brandherm
   ;;;
4   ;;; Gehirntumor (Neapolitan / Pearl)
   ;;;
6   ;;;
   ;;;
8   ;;;
   ;;;
10  ;;;
   ;;;
12  ;;;
   ;;;
14  ;;;
   ;;;
16  ;;;
   ;;;
18  ;;; Definition der Knoten:
   ;;;
20  ;;; *****
   ;;;
22  ;;; Knoten A
   ;;;
24  ;;;
   (Def-Netica-Node
26   ;; eindeutiger Name des Knotens im Schema
      "A"
28   ;; kind: NATURE_NODE DECISION_NODE UTILITY_NODE ASSUME_NODE
      NATURE_NODE
30   ;; chance or determin
      :chance
32   ;; Anzahl der Hypothesen
      2
34   ;; Namen der Hypothesen
      '("a1" "a2")
36   ;; Elternknoten des Knotens
      ;; (Hat sowohl als Zeitscheibe t = 0 oder
38   ;; als Zeitscheibe t > 0 keine Elternknoten.)
      '(()
40   :start ())
      ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t > 0
42   NIL
      ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t = 0
44   #'(lambda (node state-number) (float (nth state-number '(0.2 0.8))))
      ;; state
46   :dynamic
      ;; Aliasnamen des Knotens nicht definiert
48   ;; print-name optional

```

```

)
50 [...]
   ;; *****
52   ;;
   ;; Knoten D
54   ;;
   ;;
56 (Def-Netica-Node
   ;; eindeutiger Name des Knotens im Schema
58   "D"
   ;; kind
60   NATURE_NODE
   ;; chance or determin
62   :chance
   ;; Anzahl der Hypothesen
64   2
   ;; Namen der Hypothesen
66   '( "d1" "d2" )
   ;; Elternknoten des Knotens
68   '( ( )
       :start ("B" "C" ) )
70   ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t > 0
   NIL
72   ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t = 0
   #'(lambda (node state-number states-array)
74     (float
76       (cpt-d-under-bc state-number ; ; D
                        (foreign-aref states-array 0) ; ; B
                        (foreign-aref states-array 1) ; ; C
78       )
       ) ; float
80     ) ; lambda
   ;; state
82   :dynamic
   ;; Aliasnamen des Knotens nicht definiert
84   ;; print-name optional
   )
86   ;;
   ;; CPT fuer D (Zeitscheibe = 0)
88   ;;
   ;;      |           C           |
90   ;;      |      c1      |      c2      |
   ;;      +-----+-----+
92   ;;      |      B      |      B      |
   ;;      |  b1  |  b2  |  b1  |  b2  |
94   ;;  -----+-----+-----+-----+
   ;;      d1 | 0.80 | 0.70 | 0.90 | 0.05 |
96   ;; D  ---+-----+-----+-----+
   ;;      d2 | 0.20 | 0.30 | 0.10 | 0.95 |
98   ;;  ---+-----+-----+-----+
   ;;
100 (defun cpt-d-under-bc (d b c)
    (nth d
102      (nth b
104        (nth c

```

```

                                (0.80 0.20) ; P(D | b1, c1)
106                                (0.70 0.30)) ; P(D | b2, c1)
                                (
108                                (0.90 0.10) ; P(D | b1, c2)
                                (0.05 0.95)) ; P(D | b2, c2)
110                                )
                                ); nth c
112                                ); nth b
                                ); nth d
114    ); defun cpt-d-under-bc
    [...]
```

Die folgende Datei lädt das Modell des Bayesschen Netzes ein und erzeugt eine Instanz davon. Z.B. wird durch den Befehl `(Get-Netica-Node-from-spec "E" :time 0)` der Knoten *E* und alle seine Vorgängerknoten des Modells erzeugt.

```

1  (Make-NeticaEnvironment)
2  (Make-NewNet "cancer")
   ;;
4  ;; Modell des Netzes einladen
   ;;
6  (load "/WW/ai-gate/home/borisbra/DIPL0M2/rubin/modelle2/modell-cancer")
   ;;
8  ;; eine Instanz des Netzes erzeugen, ...
   ;;
10 (Get-Netica-Node-from-spec "E" :time 0)
    (Get-Netica-Node-from-spec "D" :time 0)
```

Im folgenden wird gezeigt, wie die Instanz des Bayesschen Netzes *Gehirntumor* und der dazugehörige Junction Tree erzeugt werden.

Als erstes wird das Hauptprogramm geladen:

```

USER(1): (load "rubin")
; Loading ./rubin.lisp
```

```

      ### # # ### # # # - - - - +++++#####
      # # # # # # # ## # - - - - +++++#####
      ### # # ### # # # # - - - - +++++#####
      # # # # # # # ## - - - - +++++#####
      # # ### ### # # # - - - - +++++#####
Rollup Benutzer-Interface
```

```

** Laedt alle benoetigten Module fuer Rollup ein.
[...]
```

```

T
USER(2): (in-package :clustering)
#<The CLUSTERING package>
CLUSTERING(3):
```

Jetzt wird das Modell des Bayesschen Netzes eingeladen und die Instanz in Netica™ erzeugt:

```

CLUSTERING(3): (load "/WW/ai-gate/home/borisbra/DIPL0M2/rubin/modelle2/demo-cancer")
; Loading /WW/ai-gate/home/borisbra/DIPL0M2/rubin/modelle2/demo-cancer.lisp
; Loading /WW/ai-gate/home/borisbra/DIPL0M2/rubin/modelle2/modell-cancer.lisp
```

```

Creating new Netica node named "E_0"
Creating new Netica node named "C_0"
Creating new Netica node named "A_0"
Creating new Netica node named "D_0"
Creating new Netica node named "B_0"
T
CLUSTERING(4):

```

Jetzt werden die Knoten aus Netica™ nach *RUBIN* übertragen:

```

CLUSTERING(4): (SETF KNOTENLISTE
                (MAPCAR #'GETNODENAME_BN
                        (NODELIST_BN-TO-LIST (GETNETNODES_BN *NET*))))
("A_0" "B_0" "C_0" "D_0" "E_0")
CLUSTERING(5): (SETF HASH (MAKE-RUBIN-GRAPH-FROM-NETICA KNOTENLISTE *NET*))
Habe die Netzstruktur. Erzeuge die Matrizen.
#<EQUAL hash-table with 5 entries @ #x839996a>
CLUSTERING(6):

```

Das Bayessche Netz läßt sich nun mit dem Visualisierungsprogramm von *RUBIN* bildlich darstellen. In Abbildung C.1 ist das Beispielnetzes Gehirntumor, wie es mit *RUBIN* angezeigt wird, als Bildschirmabzug zu sehen.

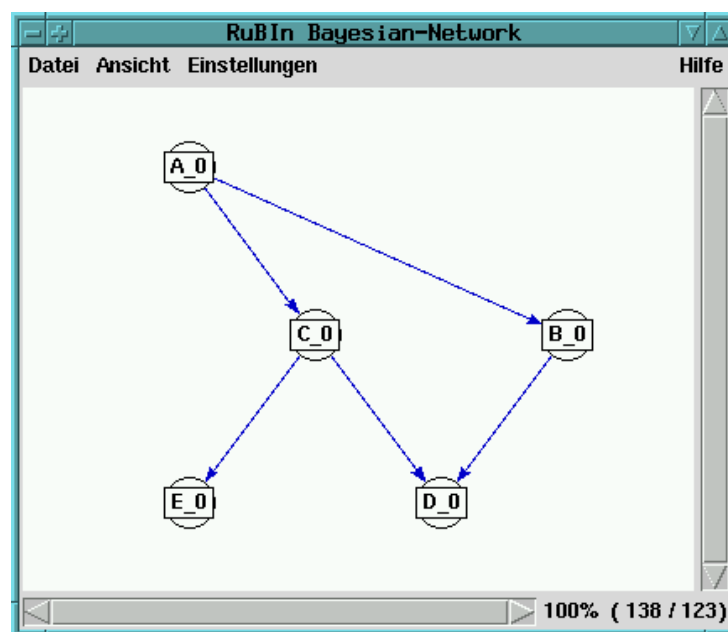


Abbildung C.1: Bildschirmabzug des Beispielnetzes Gehirntumor (DAG)

Der Graph wird nun moralisiert und trianguliert (Das Ergebnis der Triangulierung ist in Abbildung C.2 als Bildschirmabzug von *RUBIN* zu sehen.). Dazu wird für die Triangulierung die Eliminationsreihenfolge bestimmt.

```

CLUSTERING(6): (MAKE-MARKOV-GRAPH HASH)
(("B_0" "C_0"))
CLUSTERING(7): (MULTIPLE-VALUE-SETQ (ELIM TRIANGULATION)
                (GREEDY-ELIM HASH KNOTENLISTE))
;; Anstatt (MULTIPLE-VALUE-SETQ (ELIM TRIANG...)) koennen die beiden

```

```
;; nachfolgenden aukommentierten Funktionen aufgerufen werden
;;(setf elim (enumber-nodes hash Knotenliste))
;;(setf Triangulation (fill-in elim hash))
("E_0" "D_0" "C_0" "B_0" "A_0")
CLUSTERING(8):
```

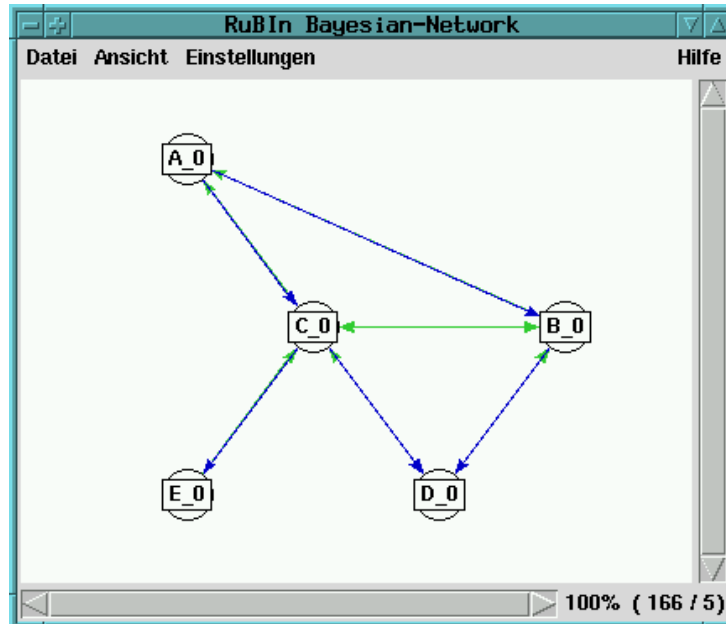


Abbildung C.2: Bildschirmabzug des Beispielnetzes Gehirntumor (trianguliert)

Aus dem triangulierten Graphen wird der Junction Graph bestimmt und jeder Knoten des Graphen einer Clique zugeordnet.

```
CLUSTERING(8): (SETF JOIN-HASH (FORM-JUNCTION-GRAPH ELIM HASH))
clique:1 (2 2)
clique:2 (2 2 2)
clique:3 (2 2 2)
**** Berechne Cliques-Zugehoerigkeit des Knotens A_0: 3 ((A_0))
**** Berechne Cliques-Zugehoerigkeit des Knotens B_0: 3 ((A_0) (B_0 A_0))
**** Berechne Cliques-Zugehoerigkeit des Knotens C_0: 3 ((A_0) (B_0 A_0) (C_0 A_0))
**** Berechne Cliques-Zugehoerigkeit des Knotens D_0: 2 ((D_0 B_0 C_0))
**** Berechne Cliques-Zugehoerigkeit des Knotens E_0: 1 ((E_0 C_0))
#<EQUAL hash-table with 3 entries @ #x83d34fa>
CLUSTERING(9):
```

Wie man oben ablesen kann, wird der Knoten D der Clique Clq_2 zugeordnet, und die Clique Clq_1 enthält eine 2×2 -Tabelle.

Durch Löschen von gewissen Kanten im Junction Graph erhält man den Junction Tree des Bayesschen Netzes Gehirntumor.

```
CLUSTERING(9): (FORM-JUNCTION-TREE JOIN-HASH)
((2 3) (1 3))
CLUSTERING(10):
```

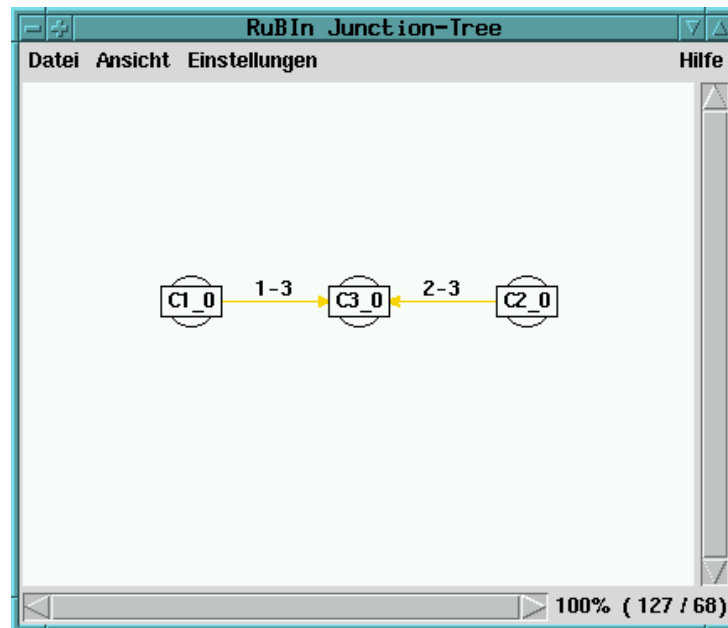


Abbildung C.3: Bildschirmabzug des Beispielnetzes Gehirntumor (Junction Tree)

Als Ausgabe erhält man die Kanten, die im Junction Tree die Cliques miteinander verbinden. In diesem Beispiel sind es die Kanten (2 3) (Clq_2 wird mit Clq_3 verbunden) und (1 3) (Clq_1 wird mit Clq_3 verbunden), wie man in Abbildung C.3 vergleichen kann.

Durch einen Doppelklick im Visualisierungsprogramm von *RUBIN* auf eine Clique des Junction Trees wird ein Infofenster geöffnet und zusätzliche Daten über die Clique dem Benutzer zur Verfügung gestellt. In Abbildung C.4 ist das Infofenster für die Clique Clq_3 abgebildet.

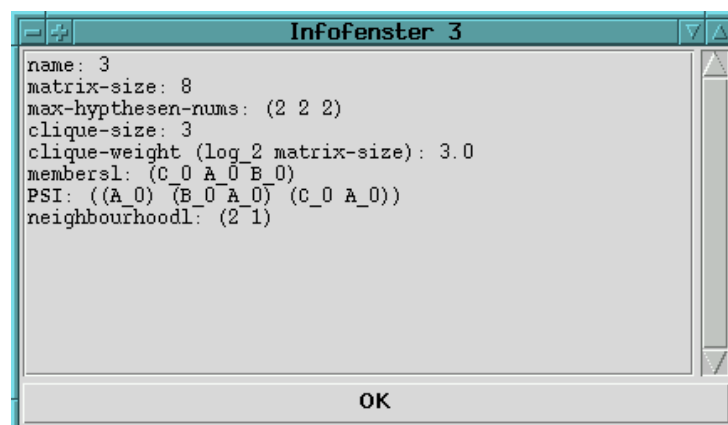


Abbildung C.4: Bildschirmabzug des Beispielnetzes Gehirntumor (Infofenster)

Auf dem Junction Tree kann nun der Inferenzalgorithmus gestartet werden. Als erstes wird der Junction Tree initialisiert. Man erkennt leicht die beiden Phasen *Collect Evidence* und *Distribute Evidence*:

```
CLUSTERING(10): (INITIALIZE-JUNCTION-TREE-SYM JOIN-HASH)
```

```
#####
COLLECT EVIDENCE (SYM)

Status von Clique 1 ist (collect: COLLECT) (pruning: NO).
Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
Status von Clique 2 ist (collect: COLLECT) (pruning: NO).
  *** Status von Clique 2 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 2 (Groesse: 8) nach Clique 3 (Groesse: 8)
  *** Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 3 (Groesse: 8) nach Clique 1 (Groesse: 4)
```

```
#####
DISTRIBUTE EVIDENCE (SYM)

Status von Clique 1 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 1 (Groesse: 4) zu Clique 3 (Groesse: 8)
Status von Clique 3 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 3 (Groesse: 8) zu Clique 2 (Groesse: 8)
Status von Clique 2 ist (distribute: DISTRIBUTE) (pruning: NO).
NIL
CLUSTERING(11):
```

Die Clique Clq_1 ist in diesem Beispiel die Wurzelclique.
Jetzt werden die Hypothesen des Knotens A parametrisiert.

```
CLUSTERING(11): (UPDATE-JUNCTION-TREE-SYM JOIN-HASH '(("A_0" ("A1" "A2"))))

**** Neue Evidenz in Clique 3 eingetragen.

#####
COLLECT EVIDENCE (SYM)

Status von Clique 1 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
Status von Clique 2 ist (collect: COLLECTED) (pruning: NO).
  *** Keine neuen Informationen in der Clique 2.
  *** Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 3 (Groesse: 8) nach Clique 1 (Groesse: 4)
```

```
#####
DISTRIBUTE EVIDENCE (SYM)

Status von Clique 1 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 1 (Groesse: 4) zu Clique 3 (Groesse: 8)
Status von Clique 3 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 3 (Groesse: 8) zu Clique 2 (Groesse: 8)
Status von Clique 2 ist (distribute: DISTRIBUTE) (pruning: NO).
NIL
CLUSTERING(12):
```

Wie man erkennt, werden nur von der Clique Clq_3 die Informationen gesammelt, da in der Clique Clq_2 keine neuen Informationen stehen.

Durch die Parametrisierung der Hypothesen des Knotens A ist jetzt der BEL-Wert jeden Knotens des Bayesschen Netzes ein Polynom in Abhängigkeit der Parameter.

```

CLUSTERING(12): (SHOW-BELIEF-SYM '"A_0" HASH JOIN-HASH)
a1: (/
(+ (* A1 0.20000002) 0.0)
(+ (* A1 0.20000002) (* A2 0.80000001) 0.0))
a2: (/
(+ (* A2 0.80000001) 0.0)
(+ (* A1 0.20000002) (* A2 0.80000001) 0.0))
NIL
CLUSTERING(13): (SHOW-BELIEF-SYM '"B_0" HASH JOIN-HASH)
b1: (/
(+ (* A2 0.16000001) (* A1 0.16000001) 0.0)
(+ (* A2 0.80000001) (* A1 0.20000002) 0.0))
b2: (/
(+ (* A2 0.64000005) (* A1 0.040000003) 0.0)
(+ (* A2 0.80000001) (* A1 0.20000002) 0.0))
NIL
CLUSTERING(14): (SHOW-BELIEF-SYM '"C_0" HASH JOIN-HASH)
c1: (/
(+ (* A2 0.040000003) (* A1 0.040000003) 0.0)
(+ (* A2 0.80000001) (* A1 0.20000002) 0.0))
c2: (/
(+ (* A2 0.76000005) (* A1 0.16000001) 0.0)
(+ (* A2 0.80000001) (* A1 0.20000002) 0.0))
NIL
CLUSTERING(15): (SHOW-BELIEF-SYM '"D_0" HASH JOIN-HASH)
d1: (/
(+ (* A2 0.19600001) (* A1 0.148) 0.0)
(+ (* A2 0.80000001) (* A1 0.2) 0.0))
d2: (/
(+ (* A2 0.60400003) (* A1 0.052) 0.0)
(+ (* A2 0.80000001) (* A1 0.2) 0.0))
NIL
CLUSTERING(16): (SHOW-BELIEF-SYM '"E_0" HASH JOIN-HASH)
e1: (/
(+ (* A2 0.48800007) (* A1 0.128) 0.0)
(+ (* A2 0.80000001) (* A1 0.20000002) 0.0))
e2: (/
(+ (* A2 0.312) (* A1 0.072000004) 0.0)
(+ (* A2 0.80000001) (* A1 0.20000002) 0.0))
NIL
CLUSTERING(17):

```

Im folgenden werden jetzt auch die Hypothesen des Knotens C parametrisiert.

```

CLUSTERING(17): (UPDATE-JUNCTION-TREE-SYM JOIN-HASH '(("C_0" ("C1" "C2"))))

**** Neue Evidenz in Clique 3 eingetragen.

#####
COLLECT EVIDENZ (SYM)

Status von Clique 1 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
Status von Clique 2 ist (collect: COLLECTED) (pruning: NO).
*** Keine neuen Informationen in der Clique 2.
*** Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 3 (Groesse: 8) nach Clique 1 (Groesse: 4)

```

#####

DISTRIBUTE EVIDENZ (SYM)

```
Status von Clique 1 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 1 (Groesse: 4) zu Clique 3 (Groesse: 8)
Status von Clique 3 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 3 (Groesse: 8) zu Clique 2 (Groesse: 8)
Status von Clique 2 ist (distribute: DISTRIBUTE) (pruning: NO).
NIL
CLUSTERING(18):
```

Die BEL-Werte der Knoten des Bayesschen Netzes sehen dann wie folgt aus:

```
CLUSTERING(18): (SHOW-BELIEF-SYM '"A_0" HASH JOIN-HASH)
a1: (/
(+ (* A1 C2 0.16000001) (* A1 C1 0.040000003) 0.0)
(+ (* A1 C1 0.040000003) (* A1 C2 0.16000001) (* A2 C2 0.76000005)
  (* A2 C1 0.040000003) 0.0))
a2: (/
(+ (* A2 C2 0.76000005) (* A2 C1 0.040000003) 0.0)
(+ (* A1 C1 0.040000003) (* A1 C2 0.16000001) (* A2 C2 0.76000005)
  (* A2 C1 0.040000003) 0.0))
NIL
CLUSTERING(19): (SHOW-BELIEF-SYM '"B_0" HASH JOIN-HASH)
b1: (/
(+ (* A2 C2 0.15200001) (* A1 C2 0.128) (* A2 C1 0.008) (* A1 C1 0.032) 0.0)
(+ (* A2 C2 0.76000005) (* A1 C2 0.16000001) (* A2 C1 0.040000003)
  (* A1 C1 0.040000003) 0.0))
b2: (/
(+ (* A2 C2 0.60800004) (* A1 C2 0.032) (* A2 C1 0.032) (* A1 C1 0.008) 0.0)
(+ (* A2 C2 0.76000005) (* A1 C2 0.16000001) (* A2 C1 0.040000003)
  (* A1 C1 0.040000003) 0.0))
NIL
CLUSTERING(20): (SHOW-BELIEF-SYM '"C_0" HASH JOIN-HASH)
c1: (/
(+ (* A2 C1 0.040000003) (* A1 C1 0.040000003) 0.0)
(+ (* A1 C1 0.040000003) (* A2 C1 0.040000003) (* A2 C2 0.76000005)
  (* A1 C2 0.16000001) 0.0))
c2: (/
(+ (* A2 C2 0.76000005) (* A1 C2 0.16000001) 0.0)
(+ (* A1 C1 0.040000003) (* A2 C1 0.040000003) (* A2 C2 0.76000005)
  (* A1 C2 0.16000001) 0.0))
NIL
CLUSTERING(21): (SHOW-BELIEF-SYM '"D_0" HASH JOIN-HASH)
d1: (/
(+ (* A1 C2 0.1168) (* A2 C2 0.16720001) (* A2 C1 0.028800001)
  (* A1 C1 0.031200001) 0.0)
(+ (* A1 C2 0.16) (* A2 C2 0.76000005) (* A2 C1 0.040000003) (* A1 C1 0.04)
  0.0))
d2: (/
(+ (* A1 C2 0.0432) (* A2 C2 0.5928) (* A2 C1 0.0112000005) (* A1 C1 0.0088)
  0.0)
(+ (* A1 C2 0.16) (* A2 C2 0.76000005) (* A2 C1 0.040000003) (* A1 C1 0.04)
  0.0))
NIL
CLUSTERING(22): (SHOW-BELIEF-SYM '"E_0" HASH JOIN-HASH)
```

```

e1: (/
(+ (* A1 C2 0.09600001) (* A2 C2 0.45600006) (* A2 C1 0.032) (* A1 C1 0.032)
  0.0)
(+ (* A1 C2 0.16000001) (* A2 C2 0.7600001) (* A2 C1 0.040000003)
  (* A1 C1 0.040000003) 0.0))
e2: (/
(+ (* A1 C2 0.064) (* A2 C2 0.30400002) (* A2 C1 0.008) (* A1 C1 0.008) 0.0)
(+ (* A1 C2 0.16000001) (* A2 C2 0.7600001) (* A2 C1 0.040000003)
  (* A1 C1 0.040000003) 0.0))
NIL
CLUSTERING(23):

```

Bsp

Im folgenden Beispiel werden nun anstatt der Hypothesen die Wahrscheinlichkeitstabellen der Knoten parametrisiert.

Beispiel C.1.2 (a-priori Parametrisierung)

Für die a-priori Parametrisierung (siehe auch Beispiel A.3.1) wird wie im vorangegangenen Beispiel wiederum das Bayessche Netz *Gehirntumor* als Beispielnetz herangezogen (siehe Beispiel 1.1.1, Abbildung 1.1 und Tabelle 1.1).

Sei das Bayessche Netz schon als Instanz erzeugt und die Knoten von Netica™ nach *RUBIN* übertragen worden. Jetzt werden die a-priori Wahrscheinlichkeiten des Knotens *A* parametrisiert. Dazu werden die entsprechenden Polynome erzeugt:

```

CLUSTERING(5): (setf a1 (make-polynom :variablen '((A1)) :koeffizienten '(1)))
#S(POLYNOM :VARIABLEN ((A1)) :KOEFFIZIENTEN (1) :KONSTANTE 0)
CLUSTERING(6): (setf a2 (make-polynom :variablen '((A2)) :koeffizienten '(1)))
#S(POLYNOM :VARIABLEN ((A2)) :KOEFFIZIENTEN (1) :KONSTANTE 0)
CLUSTERING(7):

```

Und dann im Knoten *A* eingetragen:

```

CLUSTERING(7): (setf (aref (psi-matrix (gethash "A_0" hash)) 0) a1)
#S(POLYNOM :VARIABLEN ((A1)) :KOEFFIZIENTEN (1) :KONSTANTE 0)
CLUSTERING(8): (setf (aref (psi-matrix (gethash "A_0" hash)) 1) a2)
#S(POLYNOM :VARIABLEN ((A2)) :KOEFFIZIENTEN (1) :KONSTANTE 0)
CLUSTERING(9):

```

Die Bestimmung des Junction Trees und die Inferenz laufen analog wie im oberen Beispiel ab:

```

CLUSTERING(10): (MAKE-MARKOV-GRAPH HASH)
(("B_0" "C_0"))
CLUSTERING(11): (MULTIPLE-VALUE-SETQ (ELIM TRIANGULATION)
                                     (GREEDY-ELIM HASH KNOTENLISTE))
("E_0" "D_0" "C_0" "B_0" "A_0")
CLUSTERING(12): (SETF JOIN-HASH (FORM-JUNCTION-GRAPH ELIM HASH))
[...]
CLUSTERING(13): (FORM-JUNCTION-TREE JOIN-HASH)
((2 3) (1 3))
CLUSTERING(14): (INITIALIZE-JUNCTION-TREE-SYM JOIN-HASH)

#####

```

```
COLLECT EVIDENZ (SYM)
[...]
```

```
#####
DISTRIBUTE EVIDENZ (SYM)
[...]
```

```
CLUSTERING(15):
```

Als BEL-Werte für die Knoten des Bayesschen Netzes erhält man:

```
CLUSTERING(15): (SHOW-BELIEF-SYM '"A_0" HASH JOIN-HASH)
a1: (/
(+ (* A1 1.0) 0.0)
(+ (* A1 1.0) (* A2 1.0) 0.0))
a2: (/
(+ (* A2 1.0) 0.0)
(+ (* A1 1.0) (* A2 1.0) 0.0))
NIL
CLUSTERING(16): (SHOW-BELIEF-SYM '"B_0" HASH JOIN-HASH)
b1: (/
(+ (* A2 0.2) (* A1 0.8000001) 0.0)
(+ (* A2 1.0) (* A1 1.0000001) 0.0))
b2: (/
(+ (* A2 0.8) (* A1 0.2000002) 0.0)
(+ (* A2 1.0) (* A1 1.0000001) 0.0))
NIL
CLUSTERING(17): (SHOW-BELIEF-SYM '"C_0" HASH JOIN-HASH)
c1: (/
(+ (* A2 0.050000004) (* A1 0.2000002) 0.0)
(+ (* A2 1.0) (* A1 1.0000001) 0.0))
c2: (/
(+ (* A2 0.95) (* A1 0.8000001) 0.0)
(+ (* A2 1.0) (* A1 1.0000001) 0.0))
NIL
CLUSTERING(18): (SHOW-BELIEF-SYM '"D_0" HASH JOIN-HASH)
d1: (/
(+ (* A2 0.24499999) (* A1 0.74000007) 0.0)
(+ (* A2 1.0) (* A1 1.0000001) 0.0))
d2: (/
(+ (* A2 0.755) (* A1 0.26000002) 0.0)
(+ (* A2 1.0) (* A1 1.0000001) 0.0))
NIL
CLUSTERING(19): (SHOW-BELIEF-SYM '"E_0" HASH JOIN-HASH)
e1: (/
(+ (* A2 0.61) (* A1 0.64000005) 0.0)
(+ (* A2 1.0) (* A1 1.0) 0.0))
e2: (/
(+ (* A2 0.39) (* A1 0.36) 0.0)
(+ (* A2 1.0) (* A1 1.0) 0.0))
NIL
CLUSTERING(20):
```

C.2 Dynamische Bayessche Netze

In diesem Abschnitt werden die Behandlungsmöglichkeiten für Dynamische Bayessche Netze vorgestellt. Es wird gezeigt, wie ein Modell eines Dynamischen Bayesschen Netzes definiert wird, wie sich die eingeschränkte Eliminationsreihenfolge bestimmen läßt und Äste im Junction Tree abgeschnitten werden (Rollup).

Beispiel C.2.1 (Dynamisches Bayessches Netz (3 Zeitscheiben))

Als Beispiel wird ein Dynamisches Bayessches Netz herangezogen, wie es schon im Kapitel 2 zu Erklärungszwecken verwendet wird. Für die Definition des Dynamischen Bayesschen Netzes mit seinen Zeitscheiben und Knoten wird die folgende Datei benötigt. In ihr sind auch alle a-priori und bedingten Wahrscheinlichkeiten der Knoten enthalten. Exemplarisch sind nur der Knoten *A* und der Knoten *D* präsentiert.

```

1   ;;; Modell fuer mehrfach verknuepfte Bayessche Netze
2   ;;; Boris Brandherm
3   ;;;
4   ;;; Kjaerulff
5   ;;;
6   ;;;           A----->A
7   ;;;           |\
8   ;;;           | \
9   ;;;           |  '
10  ;;;           |   B
11  ;;;           |  /
12  ;;;           | /
13  ;;;           v'
14  ;;;           C
15  ;;;           |
16  ;;;           v
17  ;;;           D----->D
18  ;;;           / \
19  ;;;          /   \
20  ;;;         ,     '
21  ;;;        E       F
22  ;;;       \       /
23  ;;;        \     /
24  ;;;         '   ,
25  ;;;           G
26  ;;;           |
27  ;;;           v
28  ;;;           H----->H
29  ;;;
30  ;;;
31  ;;; Definition der Knoten:
32  ;;;
33  ;;; *****
34  ;;;
35  ;;; Knoten A
36  ;;;
37  (Def-Netica-Node
38   ; Name des Knotens im Schema
39   "A"
40   ; kind: NATURE_NODE DECISION_NODE UTILITY_NODE ASSUME_NODE

```

```

NATURE_NODE
42  ;; chance or determin
    :chance
44  ;; Anzahl der Hypothesen des Knotens
    2
46  ;; Namen der Hypothesen
    '("a1" "a2")
48  ;; Elterknoten des Knotens
    ;; (Hat als Zeitscheibe 0 keine Elternknoten, ansonsten
50  ;; den Knoten A aus der Vorgaengerzeitscheibe.)
    '(
52  ((("A" -1))
     :start ())
54  )
    ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t > 0
56  ;; (bedingte Wahrscheinlichkeiten)
    #'(lambda (node state-number states-array)
58  (float
60  (cpt-a-under-a-1 state-number           ;; A
                    (foreign-aref states-array 0) ;; A-1
                    )
62  ); float
    ); lambda
64  ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t = 0
    ;; (a-priori Wahrscheinlichkeiten)
66  #'(lambda (node state-number)
68  (float (nth state-number '(0.2 0.8)))
    )
    ;; state
70  :dynamic
    ;; Aliasnamen des Knotens nicht definiert
72  ;; print-name optional
    )
74  ;;
    ;; CPT fuer A (Zeitscheibe t > 0)
76  ;;
    ;;      |      A-1      |
78  ;;      |  a1  |  a2  |
    ;;  -----+-----+-----+
80  ;;      a1 | 1.00 | 0.00 |
    ;;  A  -----+-----+-----+
82  ;;      a2 | 0.00 | 1.00 |
    ;;  -----+-----+-----+
84  ;;
    (defun cpt-a-under-a-1 (A A-1)
86  (nth A
        (nth A-1
            '((1.00 0.00) ; P(A | {a-1}1)
              (0.00 1.00)) ; P(A | {a-1}2)
            )))
90  [...]
92  ;;; *****
    ;;;
94  ;;; Knoten D
    ;;;
96  (Def-Netica-Node

```

```

    ;; eindeutiger Name des Knotens im Schema
98   "D"
    ;; kind
100  NATURE_NODE
    ;; chance or determin
102  :chance
    ;; Anzahl der Hypothesen
104  2
    ;; Namen der Hypothesen
106  '( "d1" "d2" )
    ;; Elternknoten des Knotens
108  '(
      ("C" ("D" -1))
110  :start ("C")
      )
112  ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t > 0
    ;; (bedingte Wahrscheinlichkeiten)
114  #'(lambda (node state-number states-array)
        (float
116      (cpt-d-under-cd-1 state-number           ;; D
                          (foreign-aref states-array 0) ;; C
118      (foreign-aref states-array 1) ;; D-1
                          )
        ); float
      ); lambda
120  ); float
    ;; Wahrscheinlichkeitentabelle fuer Zeitscheibe t = 0
    ;; (bedingte Wahrscheinlichkeiten)
124  #'(lambda (node state-number states-array)
        (float
126      (cpt-d-under-c state-number           ;; D
                          (foreign-aref states-array 0) ;; C
128      )
        ); float
      ); lambda
130  ); lambda
    ;; state
132  :dynamic
    ;; Aliasnamen des Knotens nicht definiert
134  ;; print-name optional
    )
136  ;; ***
    ;;
138  ;; CPT fuer D (Zeitscheibe = 0)
    ;;
140  ;;      |      C      |
    ;;      |  c1  |  c2  |
142  ;; -----+-----+-----+
    ;;      d1 | 0.80 | 0.20 |
144  ;; D ---+-----+-----+
    ;;      d2 | 0.20 | 0.80 |
146  ;; -----+-----+-----+
    ;;
148  (defun cpt-d-under-c (D C)
      (nth D
150        (nth C
              '((0.80 0.20) ; P(D=d1,d2|C=c1)
                (0.20 0.80)) ; P(D=d1,d2|C=c2)
152

```

```

    )))
154 ;; ***
    ;;
156 ;; CPT fuer D (Zeitscheibe t > 0)
    ;;
158 ;;      |           B           |
    ;;      |      b1      |      b2      |
160 ;;      +-----+-----+
    ;;      |      A      |      A      |
162 ;;      |  a1  |  a2  |  a1  |  a2  |
    ;;      +-----+-----+
164 ;;      c1 | 0.80 | 0.70 | 0.90 | 0.05 |
    ;;      C +-----+-----+
166 ;;      c2 | 0.20 | 0.30 | 0.10 | 0.95 |
    ;;      +-----+-----+
168 ;;
(defun cpt-d-under-cd-1 (d c d-1)
170   (nth d
      (nth c
          (nth d-1
              '((
174                 (0.80 0.20) ; P(D | c1, {d-1}1)
                  (0.70 0.30)) ; P(D | c2, {d-1}1)
176                 (
                  (0.90 0.10) ; P(D | c1, {d-1}2)
178                 (0.05 0.95)) ; P(D | c2, {d-1}2)
                  )
              ); nth d-1
          ); nth c
      ); nth d
    ); defun cpt-d-under-cd-1
184 [...]
```

Die folgende Datei lädt ein Modell eines Dynamischen Bayesschen Netzes ein und erzeugt davon drei Zeitscheiben (nämlich die Zeitscheiben 0 bis 2) durch den Befehl (`Get-Netica-Node-from-spec "H" :time 2`). Dadurch werden der Knoten *E* und rekursiv alle seine Vorgängerknoten erzeugt. In Zeitscheibe 0 besitzen die Knoten keine Vorgängerknoten in anderen Zeitscheiben und die Rekursion terminiert.

```

1 (Make-NeticaEnvironment)
2 (Make-NewNet "kjaerulff")
  ;;
4 ;; Modell des Netzes einladen
  ;;
6 (load "/WW/ai-gate/home/borisbra/DIPLOM2/rubin/modelle2/modell-kjaerulff")
  ;;
8 ;; eine Instanz des Netzes erzeugen, ...
  ;;
10 (Get-Netica-Node-from-spec "H" :time 2)
```

Wie im vorigen Beispiel wird wiederum das Hauptprogramm geladen:

```

USER(1): (load "rubin")
; Loading ./rubin.lisp
```

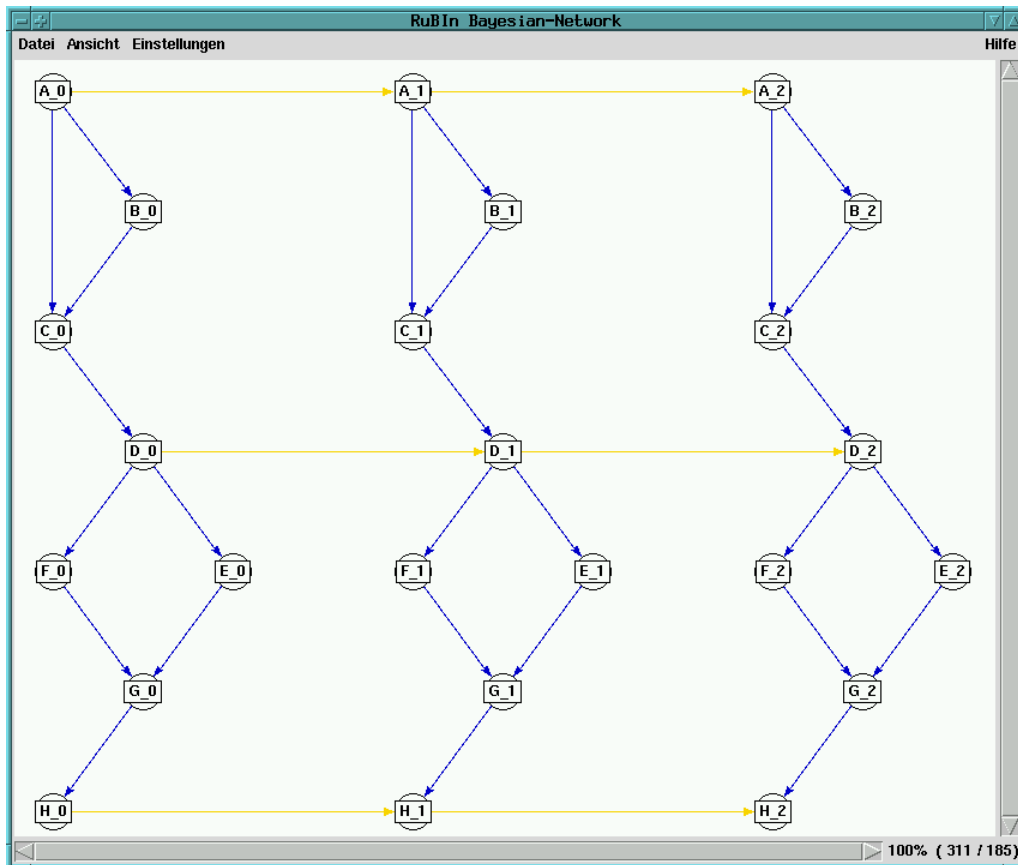



Abbildung C.5: Bildschirmabzug des Dynamischen Bayesschen Netzes (DAG)

```

(H_2 G_2 F_2 E_2 D_2 C_2 B_2 A_2) NIL)
(("G_0" "F_0" "E_0" "C_0" "B_0") ("A_0" "D_0" "H_0")
("G_1" "F_1" "E_1" "C_1" "B_1") ("A_1" "D_1" "H_1")
("H_2" "G_2" "F_2" "E_2" "D_2" "C_2" "B_2" "A_2") NIL)
CLUSTERING(8): (MULTIPLE-VALUE-SETQ (ELIM TRIANGULATION)
(GREEDY-TS-ELIM HASH TS-ELIM))
  Bearbeite folgende Liste: (G_0 F_0 E_0 C_0 B_0)
  Bearbeite folgende Liste: (A_0 D_0 H_0)
  Bearbeite folgende Liste: (G_1 F_1 E_1 C_1 B_1)
  Bearbeite folgende Liste: (A_1 D_1 H_1)
  Bearbeite folgende Liste: (H_2 G_2 F_2 E_2 D_2 C_2 B_2 A_2)
  Bearbeite folgende Liste: NIL
("B_0" "C_0" "E_0" "F_0" "G_0" "A_0" "H_0" "D_0" "B_1" "C_1" ...)
CLUSTERING(9):

```

Aus dem triangulierten Graphen wird der Junction Graph bestimmt und jeder Knoten des Graphen einer Clique zugeordnet. Durch Löschen von gewissen Kanten im Junction Graph erhält man dann den Junction Tree des Dynamischen Bayesschen Netzes.

```

CLUSTERING(9): (SETF JOIN-HASH (FORM-JUNCTION-GRAPH ELIM HASH))
clique:1 (2 2 2)
[...]
clique:14 (2 2 2 2)
**** Berechne Cliques-Zugehoerigkeit des Knotens A_0: 1 ((A_0))

```

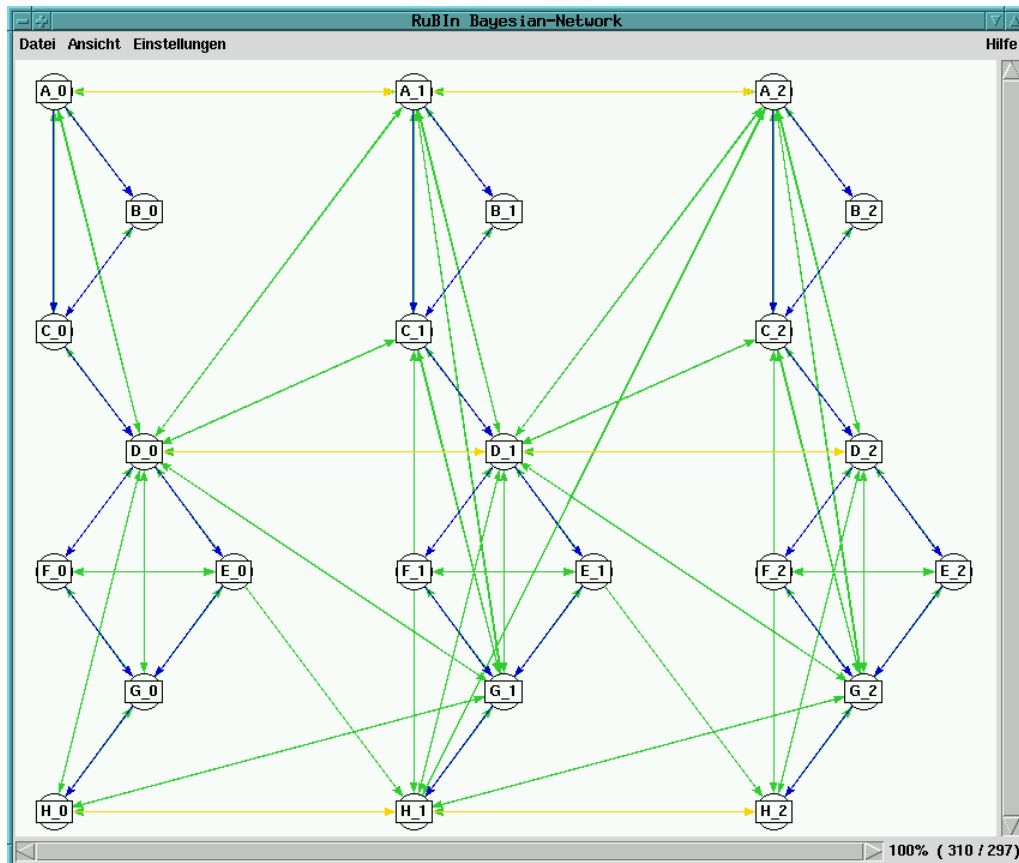


Abbildung C.6: Bildschirmabzug des Dynamischen Bayesschen Netzes (trianguliert)

```
[...]
**** Berechne Cliquen-Zugehoerigkeit des Knotens H_0: 4 ((H_0 G_0))
**** Berechne Cliquen-Zugehoerigkeit des Knotens H_1: 6 ((H_1 G_1 H_0))
[...]
**** Berechne Cliquen-Zugehoerigkeit des Knotens H_2: 11 ((H_2 G_2 H_1))
**** Berechne Cliquen-Zugehoerigkeit des Knotens G_2: 14 ((G_2 E_2 F_2))
[...]
**** Berechne Cliquen-Zugehoerigkeit des Knotens F_2: 14 ((G_2 E_2 F_2)
(E_2 D_2) (F_2 D_2))

#<EQUAL hash-table with 14 entries @ #x85ce782>
CLUSTERING(10): (FORM-JUNCTION-TREE JOIN-HASH)
((11 12) (6 7) (7 10) (10 11) (7 8) (12 14) (4 6) (7 9) (3 4) (1 2) ...)
CLUSTERING(11):
```

In Abbildung C.7 ist der Junction Tree des Dynamischen Bayesschen Netzes dargestellt.

Auf dem Junction Tree kann nun der Inferenzalgorithmus gestartet werden. Als erstes wird der Junction Tree initialisiert. Man erkennt wiederum leicht die beiden Phasen *Collect Evidence* und *Distribute Evidence*.

```
CLUSTERING(11): (INITIALIZE-JUNCTION-TREE-SYM JOIN-HASH)
```

```
#####
COLLECT EVIDENCE (SYM)
```

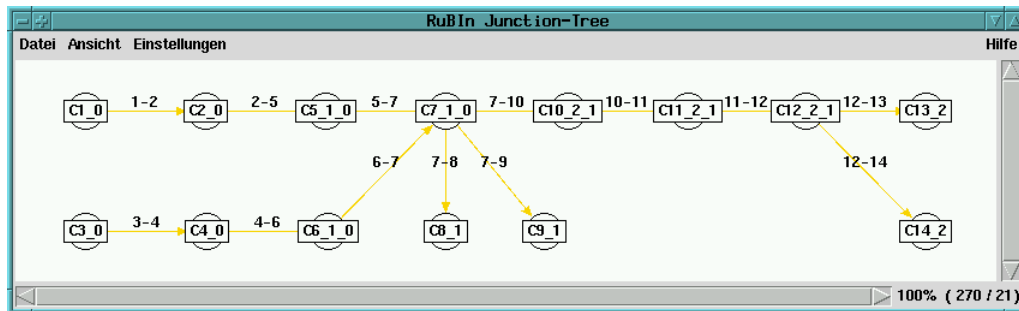


Abbildung C.7: Bildschirmabzug des Dynamischen Bayesschen Netzes (Junction Tree)

```

Status von Clique 1 ist (collect: COLLECT) (pruning: NO).
Status von Clique 2 ist (collect: COLLECT) (pruning: NO).
Status von Clique 5 ist (collect: COLLECT) (pruning: NO).
Status von Clique 7 ist (collect: COLLECT) (pruning: NO).
Status von Clique 9 ist (collect: COLLECT) (pruning: NO).
*** Status von Clique 9 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 9 nach Clique 7
Status von Clique 8 ist (collect: COLLECT) (pruning: NO).
*** Status von Clique 8 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 8 nach Clique 7
Status von Clique 10 ist (collect: COLLECT) (pruning: NO).
Status von Clique 11 ist (collect: COLLECT) (pruning: NO).
Status von Clique 12 ist (collect: COLLECT) (pruning: NO).
Status von Clique 13 ist (collect: COLLECT) (pruning: NO).
*** Status von Clique 13 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 13 nach Clique 12
Status von Clique 14 ist (collect: COLLECT) (pruning: NO).
*** Status von Clique 14 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 14 nach Clique 12
*** Status von Clique 12 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 12 nach Clique 11
*** Status von Clique 11 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 11 nach Clique 10
*** Status von Clique 10 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 10 nach Clique 7
Status von Clique 6 ist (collect: COLLECT) (pruning: NO).
Status von Clique 4 ist (collect: COLLECT) (pruning: NO).
Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
*** Status von Clique 3 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 3 nach Clique 4
*** Status von Clique 4 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 4 nach Clique 6
*** Status von Clique 6 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 6 nach Clique 7
*** Status von Clique 7 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 7 nach Clique 5
*** Status von Clique 5 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 5 nach Clique 2
*** Status von Clique 2 ist (collect: COLLECT) (pruning: NO).
**** Schicke die Nachricht von Clique 2 nach Clique 1

```

```
#####
```

```
DISTRIBUTE EVIDENCE (SYM)
```

```

Status von Clique 1 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 1 zu Clique 2
Status von Clique 2 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 2 zu Clique 5
Status von Clique 5 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 5 zu Clique 7
Status von Clique 7 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 7 zu Clique 9 Clique 8 Clique 10 Clique 6
Status von Clique 9 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 8 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 10 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 10 zu Clique 11
Status von Clique 11 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 11 zu Clique 12
Status von Clique 12 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 12 zu Clique 13 Clique 14
Status von Clique 13 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 14 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 6 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 6 zu Clique 4
Status von Clique 4 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 4 zu Clique 3
Status von Clique 3 ist (distribute: DISTRIBUTE) (pruning: NO).
NIL
CLUSTERING(12):

```

Wie obenstehend zu sehen ist, wird bei Collect Evidence vom aufrufenden Knoten in alle Blattknoten hinuntergestiegen und von allen eine Nachricht zum Aufrufenden Knoten geschickt. Bei Distribute Evidence wird bis zu allen Blattknoten hinunter die Nachricht vom aufrufenden Knoten versendet.

Im folgenden wird in der Clique Cl_{q_1} der Knoten A_0 mit einer neuen Evidenz instantiiert. Bei Collect Evidence wird zwar vom aufrufenden Knoten in alle Blattknoten hinuntergestiegen aber von keinem eine Nachricht zum aufrufenden Knoten geschickt, da keine Nachricht zu versenden ist. Bei Distribute Evidence wird bis zu allen Blattknoten hinunter die Nachricht vom aufrufenden Knoten versendet, da in Clique Cl_{q_1} eine neue Evidenz steht und eine neue Nachricht zu verschicken ist.

```
CLUSTERING(12): (UPDATE-JUNCTION-TREE-SYM JOIN-HASH '(("A_0" ("A1" "A2"))))
```

```
**** Neue Evidenz in Clique 1 eingetragen.
```

```
#####
```

```
COLLECT EVIDENCE (SYM)
```

```

Status von Clique 1 ist (collect: COLLECT) (pruning: NO).
Status von Clique 2 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 5 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 7 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 9 ist (collect: COLLECTED) (pruning: NO).
  *** Keine neuen Informationen in der Clique 9.
Status von Clique 8 ist (collect: COLLECTED) (pruning: NO).
  *** Keine neuen Informationen in der Clique 8.
Status von Clique 10 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 11 ist (collect: COLLECTED) (pruning: NO).

```

```

Status von Clique 12 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 13 ist (collect: COLLECTED) (pruning: NO).
  *** Keine neuen Informationen in der Clique 13.
Status von Clique 14 ist (collect: COLLECTED) (pruning: NO).
  *** Keine neuen Informationen in der Clique 14.
  *** Keine neuen Informationen in der Clique 12.
  *** Keine neuen Informationen in der Clique 11.
  *** Keine neuen Informationen in der Clique 10.
Status von Clique 6 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 4 ist (collect: COLLECTED) (pruning: NO).
Status von Clique 3 ist (collect: COLLECTED) (pruning: NO).
  *** Keine neuen Informationen in der Clique 3.
  *** Keine neuen Informationen in der Clique 4.
  *** Keine neuen Informationen in der Clique 6.
  *** Keine neuen Informationen in der Clique 7.
  *** Keine neuen Informationen in der Clique 5.
  *** Keine neuen Informationen in der Clique 2.

#####
DISTRIBUTE EVIDENCE (SYM)

Status von Clique 1 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 1 zu Clique 2
Status von Clique 2 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 2 zu Clique 5
Status von Clique 5 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 5 zu Clique 7
Status von Clique 7 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 7 zu Clique 9 Clique 8 Clique 10 Clique 6
Status von Clique 9 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 8 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 10 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 10 zu Clique 11
Status von Clique 11 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 11 zu Clique 12
Status von Clique 12 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 12 zu Clique 13 Clique 14
Status von Clique 13 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 14 ist (distribute: DISTRIBUTE) (pruning: NO).
Status von Clique 6 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 6 zu Clique 4
Status von Clique 4 ist (distribute: DISTRIBUTE) (pruning: NO).
**** Schicke die Nachricht von Clique 4 zu Clique 3
Status von Clique 3 ist (distribute: DISTRIBUTE) (pruning: NO).
NIL
CLUSTERING(13):

```

Durch den Befehl `(setf (status-pruning (gethash 7 join-hash)) 'pruning)` wird jetzt in der Clique Clq_7 eine Schranke für den Inferenzalgorithmus gesetzt. Wird als aufrufender Knoten z.B. die Clique Clq_{10} gewählt, so wird an der Clique Clq_7 mit Collect Evidence bzw. Distribute Evidence abgebrochen, wie im folgenden zu sehen ist. Um zu verstehen, in welchen Ästen Collect Evidence nach neuen Evidenzen sucht und in welche Äste die neue Information durch Distribute Evidence verschickt wird, sollte man sich auf Abbildung C.7 anschauen, in der der Junction Tree abgebildet ist.

```
CLUSTERING(13): (setf (status-pruning (gethash 7 join-hash)) 'pruning)
```

PRUNING

CLUSTERING(14): (UPDATE-JUNCTION-TREE-SYM JOIN-HASH '(("A_2" ("A1" "A2")))) 10)

**** Neue Evidenz in Clique 10 eingetragen.

#####

COLLECT EVIDENCE (SYM)

Status von Clique 10 ist (collect: COLLECT) (pruning: NO).

Status von Clique 11 ist (collect: COLLECTED) (pruning: NO).

Status von Clique 12 ist (collect: COLLECTED) (pruning: NO).

Status von Clique 13 ist (collect: COLLECTED) (pruning: NO).

*** Keine neuen Informationen in der Clique 13.

Status von Clique 14 ist (collect: COLLECTED) (pruning: NO).

*** Keine neuen Informationen in der Clique 14.

*** Keine neuen Informationen in der Clique 12.

*** Keine neuen Informationen in der Clique 11.

Ast hinter 7 wurde abgeschnitten: (collect: COLLECTED) (pruning: PRUNING)

#####

DISTRIBUTE EVIDENCE (SYM)

Status von Clique 10 ist (distribute: DISTRIBUTE) (pruning: NO).

**** Schicke die Nachricht von Clique 10 zu Clique 11 Clique 7

Status von Clique 11 ist (distribute: DISTRIBUTE) (pruning: NO).

**** Schicke die Nachricht von Clique 11 zu Clique 12

Status von Clique 12 ist (distribute: DISTRIBUTE) (pruning: NO).

**** Schicke die Nachricht von Clique 12 zu Clique 13 Clique 14

Status von Clique 13 ist (distribute: DISTRIBUTE) (pruning: NO).

Status von Clique 14 ist (distribute: DISTRIBUTE) (pruning: NO).

Status von Clique 7 ist (distribute: DISTRIBUTE) (pruning: PRUNING).

NIL

CLUSTERING(15):

Ist man an dem Shafer-Shenoy Skript einer Clique, z.B. Clique Clq_{10} , interessiert, so erhält man es auf die folgende Art und Weise:

CLUSTERING(15): (shafer-shenoy-skript join-hash '() 10)

T_{cpt(A_2)}

* (sum_{(G_2 H_2)} T_{cpt(H_2)})

* (sum_{(C_2 D_2)} T_{cpt(D_2)})

* (sum_{(B_2)} T_{cpt(C_2)} T_{cpt(B_2)}))

* (sum_{(F_2 E_2)} T_{cpt(G_2)} T_{cpt(E_2)} T_{cpt(F_2)}))))

* (sum_{(C_1 G_1 D_0)} T_{cpt(D_1)})

* (sum_{(A_0)} T_{cpt(A_1)})

* (sum_{(C_0)} T_{cpt(D_0)})

* (sum_{(B_0)} T_{cpt(A_0)} T_{cpt(B_0)} T_{cpt(C_0)}))))

* (sum_{(F_1 E_1)} T_{cpt(G_1)} T_{cpt(E_1)} T_{cpt(F_1)}))

* (sum_{(B_1)} T_{cpt(C_1)} T_{cpt(B_1)}))

* (sum_{(H_0)} T_{cpt(H_1)})

* (sum_{(G_0)} T_{cpt(H_0)})

* (sum_{(F_0 E_0)} T_{cpt(E_0)} T_{cpt(F_0)} T_{cpt(G_0)}))))

NIL

CLUSTERING(16):

Literaturverzeichnis

- [Andersen et al. 89] S. K. Andersen, K. G. Olesen, F. V. Jensen and F. Jensen. *A Shell for Building Bayesian Belief Universes for Expert Systems*. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pp. 1080–1085, 1989.
- [Bartsch 93] Hans-Jochen Bartsch. *Taschenbuch mathematischer Formeln*. Leipzig: Fachbuchverlag, 15., neubearbeitete Auflage, 1993.
- [Becker & Geiger 96] Ann Becker and Dan Geiger. *A Sufficiently Fast Algorithm for Finding Close to Optimal Junction Trees*. In *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI-96)*, pp. 81–89, Portland, Oregon, 1996.
- [Besnard & Hanks 95] Philippe Besnard and Steve Hanks (eds.). *Uncertainty in artificial intelligence (UAI-95): 11th conference, Montréal, Québec, Canada, August 18-20, 1995*, San Francisco, 1995. Morgan Kaufmann.
- [Boutilier & Goldszmidt 00] Craig Boutilier and Moisés Goldszmidt (eds.). *Proceedings of the Sixteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-2000)*, Stanford University, Stanford, California, June 30–July 3, 2000. Morgan Kaufmann Publishers.
- [Boyen & Koller 98a] Xavier Boyen and Daphne Koller. *Tractable Inference for Complex Stochastic Processes*. Technical report, Stanford University, Department of Computer Science, March 17 1998.
- [Boyen & Koller 98b] Xavier Boyen and Daphne Koller. *Tractable Inference for Complex Stochastic Processes*. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pp. 33–42, San Francisco, CA, 1998. Morgan Kaufmann Publishers.
- [Boyen & Koller 99] Xavier Boyen and Daphne Koller. *Exploiting the Architecture of Dynamic Systems*. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, pp. 1080–1085, Orlando, Florida, July 18–22 1999.
- [Brandherm et al. 97] Boris Brandherm, Dirk Wagner und Frank Wittig. *Lösungsalgorithmen in mehrfach verbundenen Bayes'schen Netzen*. Fortgeschrittenenpraktikum, Fachbereich 14 - Informatik, Universität des Saarlandes, 1997.

- [Bronstein & Semendjajew 89] Ilja N. Bronstein und Konstantin A. Semendjajew. *Taschenbuch der Mathematik*. Thun und Frankfurt/Main: Verlag Harri Deutsch, 24. Auflage, 1989.
- [Cooper & Moral 98] Gregory F. Cooper and Serafín Moral (eds.). *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, University of Wisconsin, Madison, Wisconsin, USA, July 24–26, 1998. Morgan Kaufmann Publishers.
- [Cooper 84] Gregory F. Cooper. *NESTOR: A Computer-Based Medical Diagnostic Aid that integrated Causal and Probabilistic Knowledge*. PhD thesis, Department of Computer Science, Stanford University, California, 1984.
- [Cooper 87] Gregory F. Cooper. *Probabilistic Inference using Belief Networks is NP-hard*. Technical Report KSL-87-27, Medical Computer Science Group, Stanford University, California, 1987.
- [Cormen et al. 92] Thomas H. Cormen, Charles E. Leiserson and Ronald L. Rivest. *Introduction to Algorithms*. The MIT electrical engineering and computer science series. Cambridge, MA: The MIT Press, 1992.
- [Dagum et al. 92] P. Dagum, A. Galper and E. Horvitz. *Dynamic Network Models for Forecasting*. In *Proceedings of the Eighth Annual Conference on Uncertainty in Artificial Intelligence (UAI-92)*, pp. 41–48, San Francisco, CA, 1992. Morgan Kaufmann Publishers.
- [Darwiche & Provan 96] Adnan Darwiche and Gregory M. Provan. *Query DAGs: A Practical Paradigm for Implementing Belief-Network Inference*. In *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI-96)*, pp. 203–210, Portland, Oregon, 1996.
- [Dubois et al. 92] Didier Dubois, Michael P. Wellman, Bruce D’Ambrosio and Phillippe Smets (eds.). *Proceedings of the Eighth Annual Conference on Uncertainty in Artificial Intelligence (UAI-92)*, Stanford University, Stanford, California, USA, July 17–19, 1992. Morgan Kaufmann Publishers.
- [Forbes et al. 95] Jeff Forbes, Tim Huang, Keiji Kanazawa and Stuart Russell. *The BATmobile: Towards a Bayesian Automated Taxi*. In Chris S. Mellish (ed.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pp. 1878–1885, San Mateo, CA, 1995. Morgan Kaufmann Publishers.
- [Goossens et al. 94] Michel Goossens, Frank Mittelbach und Alexander Samarin. *Der L^AT_EX-Begleiter*. Bonn; Paris; Reading, Mass. [u.a.]: Addison-Wesley, 1. Auflage, 1994. 1., korrigierter Nachdruck 1996.
- [Graham 94] Paul Graham. *On Lisp: Advanced Techniques for Common Lisp*. Englewood Cliffs, New Jersey: Prentice Hall, 1994.
- [Horvitz & Barry 95] Eric Horvitz and Matthew Barry. *Display of Information for Time-Critical Decision Making*. In *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence (UAI-95)*, pp. 296–305, San Francisco, CA, 1995. Morgan Kaufmann Publishers.

- [Horvitz & Jensen 96] Eric Horvitz and Finn Jensen (eds.). *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI-96)*, Reed College, Portland, Oregon, USA, August 1–4, 1996. Morgan Kaufmann Publishers.
- [Hotz 90] Günter Hotz. *Einführung in die Informatik*. Stuttgart: Teubner, 1990.
- [Huang & Darwiche 94] Cecil Huang and Adnan Darwiche. *Inference in Belief Networks: A Procedural Guide*. International Journal of Approximate Reasoning, 11, 1994.
- [Jameson et al. 99] Anthony Jameson, Ralph Schäfer, Thomas Weis, André Berthold and Thomas Weyrath. *Making Systems Sensitive to the User's Time and Working Memory Constraints*. In Mark T. Maybury (ed.), *IUI99: International Conference on Intelligent User Interfaces*, pp. 79–86. New York: ACM, 1999.
- [Jameson 90] Anthony Jameson. *Knowing What Others Know: Studies in Intuitive Psychometrics*. PhD thesis, Universität von Amsterdam,, Niederlande, 1990. Beim Autor erhältlich.
- [Jensen & Jensen 94] Finn V. Jensen and Frank Jensen. *Optimal Junction Trees*. In *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-94)*, pp. 360–366, Seattle, WA, 1994.
- [Jensen 96] Finn V. Jensen. *An Introduction to Bayesian Networks*. New York: Springer, 1996.
- [Johnson 97] Ron Johnson. *Tcl Style Guide*.
<http://www.scriptics.com/doc/styleGuide.pdf>, August 22, 1997. Sun Microsystems.
- [Kanazawa et al. 95] Keiji Kanazawa, Daphne Koller and Stuart Russell. *Stochastic Simulation Algorithm for Dynamic Probabilistic Networks*. In *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence (UAI-95)*, pp. 346–351, San Francisco, CA, 1995. Morgan Kaufmann Publishers.
- [Kipper et al. 95] Bernhard Kipper, Thorsten Brants, Marcus Plach und Ralph Schäfer. *Bayessche Netze: Ein einführendes Beispiel*. Technical Report 4, Graduiertenkolleg Kognitionswissenschaft, Saarbrücken, 1995.
- [Kjærulff 90] Uffe Kjærulff. *Triangulation of graphs – algorithms giving small total state space*. Research Report R 90-09, Department of Mathematics and Computer Science, Aalborg University, Denmark, 1990.
- [Kjærulff 92] Uffe Kjærulff. *A Computational Scheme for Reasoning in Dynamic Probabilistic Networks*. In *Proceedings of the Eighth Annual Conference on Uncertainty in Artificial Intelligence (UAI-92)*, pp. 121–129, San Francisco, CA, 1992. Morgan Kaufmann Publishers.
- [Kjærulff 93] Uffe Kjærulff. *Aspects of Efficiency Improvement in Bayesian Networks*. PhD thesis, Department of Mathematics and Computer Science, Aalborg University, Denmark, 1993.

- [Kjærulff 95] Uffe Kjærulff. *dHugin: A Computational System for Dynamic Time-Sliced Bayesian Networks*. In *International Journal of Forecasting, Special Issue on Probability Forecasting*, 11, pp. 89–111, 1995.
- [Knappen et al. 94] Jörg Knappen, Hubert Partl, Elisabeth Schlegl and Irene Hyna. *ÆTEX₂ ϵ -Kurzbeschreibung*, 20. Oktober 1994. Version 1.1.
- [Knuth 86] Donald E. Knuth. *The T_EXbook*, volume A: Computers and Typesetting. Reading, Mass.: Addison Wesley, 1986.
- [Lamprecht 78] Ernst Lamprecht. *Einführung in die Algebra*. Basel und Stuttgart: Birkhäuser Verlag, 1. Auflage, 1978.
- [Laskey & Prade 99] Kathryn Laskey and Henri Prade (eds.). *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, Stockholm, Sweden, July 30–August 1, 1999. Morgan Kaufmann Publishers.
- [Lopez de Mantaras & Poole 94] Ramon Lopez de Mantaras and David Poole (eds.). *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-94)*, University of Washington, Seattle, July 29–31, 1994. Morgan Kaufmann Publishers.
- [Madsen & Jensen 98] Anders L Madsen and Finn V Jensen. *Lazy Propagation in Junction Trees*. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pp. 362–369, San Francisco, CA, 1998. Morgan Kaufmann Publishers.
- [Neapolitan 90] Richard E. Neapolitan. *Probabilistic Reasoning in Expert Systems: Theory and Algorithms*. New York: Wiley, 1990.
- [Net 95] Norsys Software Corp., Vancouver, BC, Canada, <http://www.norsys.com>. *Netica™ API, Programmer's Library, Reference Manual, Version 1.01*, 1995.
- [Net 96] Norsys Software Corp., Vancouver, BC, Canada, <http://www.norsys.com>. *Netica™, Application for Belief Networks and Influence Diagrams, User's Guide, Version 1.03*, 1996.
- [Nicholson & Brady 92] A. E. Nicholson and J. M. Brady. *Sensor Validation Using Dynamic Belief Networks*. In *Proceedings of the Eighth Annual Conference on Uncertainty in Artificial Intelligence (UAI-92)*, pp. 207–214, San Francisco, CA, 1992. Morgan Kaufmann Publishers.
- [Norvig & Pitman 93] Peter Norvig and Kent Pitman. *Tutorial on Good Lisp Programming Style*. <http://www.norvig.com/luv-slides.ps>, August 10, 1993. Lisp Users and Vendors Conference.
- [Norvig 92] Peter Norvig. *Paradigms of Artificial Intelligence Programming: Case Studies in Common Lisp*. Artificial Intelligence Programming Languages. San Mateo, California: Morgan Kaufmann Publishers, 1992.

- [Pearl 88] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. The Morgan Kaufmann Series in Representation and Reasoning. San Mateo, CA: Morgan Kaufmann, 1988.
- [Raines 98] Paul Raines. *Tcl/Tk Pocket Reference*. Programming Tools. O'Reilly & Associates, 1st edition, October 1998.
- [Russell & Norvig 95] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall Series in Artificial Intelligence. Englewood Cliffs, New Jersey: Prentice Hall, 1995.
- [Schäfer & Weyrath 97] Ralph Schäfer and Thomas Weyrath. *Assessing Temporally Variable User Properties with Dynamic Bayesian Networks*. In Anthony Jameson, Cécile Paris and Carlo Tasso (eds.), *User Modeling: Proceedings of the Sixth International Conference, UM97*, pp. 377–388. Wien, New York: Springer Wien New York, 1997.
- [Schäfer 98] Ralph Schäfer. *Benutzmodellierung mit dynamischen Bayes'schen Netzen als Grundlage adaptiver Dialogsysteme*. Dissertation, Fachbereich Informatik, Universität des Saarlandes, 1998.
- [Shafer 96] Glenn Shafer. *Probabilistic Expert Systems*. CBMS-NSF regional conference series in applied mathematics. Philadelphia, PA: SIAM, Society for Industrial and Applied Mathematics, 1996.
- [Skaanning et al. 98] Claus Skaanning, Finn V. Jensen, Uffe Kjærulff and Anders L. Madsen. *Acquisition and Transformation of Likelihoods to Conditional Probabilities for Bayesian Networks*. In (AAAI-98), 1998.
- [Spiegelhalter 86] David J. Spiegelhalter. *Uncertainty in Artificial Intelligence*, chapter Probabilistic Reasoning in Predictive Expert Systems, pp. 47–68. Amsterdam: North Holland, 1986.
- [Srinivas & Breese 92] Sampath Srinivas and Jack Breese. *IDEAL: Influence Diagram Evaluation and Analysis in Lisp - Documentation and Users Guide*. Rockwell International Science Center, Palo Alto Laboratory, Palo Alto, CA 94301, 1992.
- [Steele 90] Guy L. Steele. *Common Lisp: The Language*. Digital Press, second edition, 1990.
- [Tawfik & Neufeld 94] A. Y. Tawfik and E. Neufeld. *Temporal Bayesian Network*. In S. D. Goodwin and H. J. Hamilton (eds.), *Proceedings of the TIME-94 – International Workshop on Temporal Representation and Reasoning*, pp. (Erhältlich von <http://www.cs.uregina.ca/~temporal/time94/time-94-papers.html>, keine Angabe der Seitenzahlen), Pensacola, FL, 1994.
- [van der Gaag & Coupé 98] Linda C. van der Gaag and Veerle M. H. Coupé. *Practical sensitivity analysis of Bayesian belief networks*. In M. Hušková, P. Lachout and J. A. Víšek (eds.), *Proceedings of the Joint Session of the 6th Prague Symposium of Asymptotic Statistics and the 13th Prague Conference on Information*

- Theory, Statistical Decision Functions and Random Processes*, pp. 81–86. Union of Czech Mathematicians and Physicists, 1998.
- [van der Gaag & Kjærulff 00] Linda C. van der Gaag and Uffe Kjærulff. *Making Sensitivity Analysis Computationally Efficient*. In *Proceedings of the Sixteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-2000)*, San Francisco, CA, 2000. Morgan Kaufmann Publishers.
- [Wahlster et al. 95] Wolfgang Wahlster, Anthony Jameson, Alassane Ndiaye, Ralph Schäfer und Thomas Weis. *Ressourcenadaptive Dialogführung: ein interdisziplinärer Forschungsansatz*. *Künstliche Intelligenz*, 9(6):17–21, 1995.
- [Weis 00] Thomas Weis. *Ressourcenadaptive Dialogführung auf entscheidungstheoretischer Basis*. Dissertation, Fachbereich Informatik, Universität des Saarlandes, 2000.
- [Welch 97] Brent B. Welch. *Practical Programming in Tcl and Tk*. Prentice Hall, second edition, July 1997.
- [Xiang & Jensen 99] Yang Xiang and Finn Jensen. *Inference in Multiply Sectioned Bayesian Networks with Extended Shafer-Shenoy and Lazy Propagation*. In *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pp. 680–687, San Francisco, CA, 1999. Morgan Kaufmann Publishers.

Index

Fettgedruckte Seitennummern verweisen auf Seiten, die wichtige Informationen wie z.B. eine Definition über den Eintrag enthalten. Seitennummern in kursiver Darstellung bezeichnen einen längeren Bereich. Normal gesetzte Seitennummern beziehen sich auf Erwähnungen im Text. Aus Gründen der Übersichtlichkeit sind im Index von den Einträgen nicht alle ihre Erwähnungen im Text vermerkt.

A

- Abhängigkeit
 - kausale *~ siehe* Kausalzusammenhang
 - ~* zwischen den Zeitscheiben.....41
- Absorption.....24
- Aalborg *~* **25**
- Shafer-Shenoy *~* **28**
- Algebra
 - ~*programm.....29
 - ~* von Wahrscheinlichkeitentabellen
22-23, 145-150
- Architektur
 - Aalborg *~* 25-26, 123-130
 - Shafer-Shenoy *~* 26-29, 130-136

B

- Bayessches Netz.....1, 3, **7**
- Dynamisches *~*.....41-59
 - Rollup-Verfahren..... *siehe* Rollup
- einfach verbundenes *~*.....1
- Inferenzverfahren..... 23-38
- mehrfach verbundenes *~* 1-39,
123-143
- parametrisiertes *~*.....34
- Propagierung . *siehe* Bayessches Netz,
Inferenzverfahren
- Berechnungsvorschrift *siehe* Skript
- BN..... *siehe* Bayessches Netz

C

- chord.....14
- Clique.....11, 25, 27
- Cluster.....13, **17**
 - maximaler *~*.....18
- Clustering-Verfahren.....13
- CPT..... *siehe* Tabelle bedingter
Wahrscheinlichkeiten

D

- D-Separations-Kriterium.....**67**, **141**,
141-143
- DAG. *siehe* Graph, gerichteter azyklischer
Query-*~*.....37
- DBN *siehe* Bayessches Netz, Dynamisches
- Domäne.....3

E

- Elementarereignis.....7
- Eliminationsreihenfolge.11, **13**, 11-14, 15
 - Berechnung einer *~*.....12
 - eingeschränkte *~*.....52
 - optimale *~*.....12
 - perfekte *~*.....**17**
- Elternknoten.....11
- Evidence
 - Collect *~*.....23, 29
 - Distribute *~*.....23, 29

Evidenz
 ~-Variable.....*siehe* Variable

F

Fehlerabschätzung.....29
 fill-edge.....16
 fill-in.....16
 redundanter ~.....16

G

Gewicht
 ~ einer Clique.....13
 ~ einer Kante.....11
 Graph.....6, 6
 azyklischer ~.....6
 einfach verknüpfter ~.....7
 geordneter ~.....13
 gerichteter ~.....3, 6
 gerichteter azyklischer ~..6, 8, 10, 11
 mehrfach verknüpfter ~.....7
 minimal aufspannender ~.....19
 moralisierter ~.....10, 11, 14
 Teilgraph.....13
 triangulierter ~.....11, 14, 15, 17
 ungerichteter ~.....11

H

Heuristik.....12, 14
 Hypothese.....2, 7, 22

I

Inferenzverfahren . *siehe* Bayessches Netz,
 Inferenzverfahren

J

JPT.....*siehe* Tabelle gemeinsamer
 Wahrscheinlichkeiten
 Junction Graph.....11, 12, 17–19
 Junction Tree.....11, 19, 19–20
 Auffrischung des ~s.....23
 Bestimmung des ~s.....10–22
 ~ Eigenschaft.....12, 13, 19
 expandierter ~.....56
 Initialisierung des ~s.....23

kalibrierter ~.....25
 optimaler ~.....20
 ~ Verfahren.....9–39

K

Kante.....6
 gerichtete ~.....6, 11
 Gewicht der ~.....18
 Quelle der ~.....6
 redundante ~.....16
 temporale ~.....49
 ungerichtete ~.....6, 11, 13
 zeitliche ~.....49
 zeitscheibenüberspringende ~.....90,
 91–104
 Ziel der ~.....6
 Kausalzusammenhang.....2, 3
 Knoten.....6
 Absorption eines ~s.....*siehe*
 Knotenabsorption
 adjazenter ~.....*siehe* Knoten,
 benachbarter
 Alias eines ~s.....48
 aufrufender ~.....23
 Aufspalten eines ~s.....62
 BEL-Wert eines ~s.....7
 Belief-Wert eines ~s.....7
 benachbarter ~.....6
 Blattknoten.....24
 ~ des Junction Graphens.....17
 dynamischer ~.....45, 49
 eindeutiger Name eines ~s.....48
 Eliminieren eines ~s.....12, 13
 Elternknoten.....6, 21
 Familie eines ~s.....6
 finding eines ~s.....22, 26
 Hypothese des ~s.....26
 Instantiierung des ~s.....7
 Interfaceknoten.....51
 Kinderknoten.....6
 Löschen eines ~s.....13, 62
 markierter ~.....13
 Megaknoten.....12
 Nachbarknoten.....17
 Separatorknoten.....18, 20, 25, 27
 Startknoten.....13

- statischer \sim 89, 104–112
 temporärer \sim 89
 unmarkierter \sim 13
 Wurzelknoten 6, 8, 11, 23
 Knotenabsorption *siehe* Rollup
 Knotenzuordnung 11, 21–22
- L**
- Lösungsverfahren *siehe*
 Bayessches Netz, Inferenzverfahren
- M**
- Markov-Eigenschaft 50, **72**
 maximum cardinality search 13
 minT *siehe* Triangulierung, minimale
 Modell 3
 dynamisches \sim 47
 Verfall des \sim s 42
 Moralisierung 11
 MST-Kruskal *siehe* Graph, minimal
 aufspannender
 Multiindex 151
- N**
- Nachinstantiierung 90, 112–118
 Nachricht 24
 Normalisierungskonstante 22
 NP-vollständig 12
- O**
- Ordnung
 \sim eines Graphens *siehe* Graph,
 geordneter
 teilweise \sim **152**
 totale \sim **152**
- P**
- Parametrisierung
 CPT- \sim 137
 \sim eines Bayesschen Netzes *siehe*
 Bayessches Netz, parametrisiertes
 Hypothesenparametrisierung 35
 Sensitivitätsanalyse *siehe*
 Sensitivitätsanalyse
- Pfad 19
 gerichteter \sim 6
- Polynom
- Addition von \sim en 35, **152**
 faktorisiertes \sim 81
 Multiplikation von \sim en 35, **152**
 multivariates \sim 34, 35, **151**
 \sim -Propagierung 34–38, 136–140
 \sim ring **152**
 Skalarmultiplikation 35, **152**
 univariates \sim **151**
- Propagierung *siehe* Bayessches Netz,
 Inferenzverfahren
 Proposition 2
- Q**
- Query
 \sim -DAG *siehe* DAG
 \sim -Variable *siehe* Variable
- R**
- READY-Netz 89–120
 Relation
 zeitliche \sim **49**
- Rollup 25, 61–87
 approximativer \sim 83–85, 86, 87
 dHUGIN 72–74, 86, 87
 Knotenabsorption 66–69, 86, 87
 Polynom-Rollup 74–79, 86, 87
 Prediction-Estimation .. 69–71, 86, 87
 Shafer-Shenoy Skripte .. 79–83, 86, 87
 faktorisierte \sim 81–83, 86, 87
 zielgerichteter \sim 81
 Rollup-Verfahren *siehe* Rollup
 RUBIN 122
 Rundungsfehler 35
- S**
- Satz
 \sim von Rose, Tarjan und Lueker ... 54
 Schema *siehe* Zeitscheibe
 Doppelschema 89, 101–104
 Sensitivitätsanalyse 140
 Skript 26, 29
 Shafer-Shenoy \sim e 29–34

T

- Tabelle
 ~ bedingter Wahrscheinlichkeiten .. 8, 22, **146**
 Division von \sim_n 23, **148**
 ~ gemeinsamer Wahrscheinlichkeiten 22, 146
 Gleichheit von \sim_n **147**
 Marginalisation einer \sim 23, **149**
 Multiplikation von \sim_n 23, **147**
 ψ -Tabelle 21
 ~ über einer Knotenmenge... 25, **145**
 Wahrscheinlichkeiten \sim 29, **147**
 Triangulierung **16**, 14–17
 absolut minimale \sim 16
 minimale \sim **16**

V

- Variable
 Evidenz- \sim 37
 Query- \sim 37

W

- Wahrscheinlichkeit
 a-priori \sim en 8, 21, 25
 bedingte \sim en 7, 8, 21, 25
 Gesamtwahrscheinlichkeit 65
 zeitabhängige bedingte \sim en 42
 Wertebereich 7
 ~ einer Zufallsvariablen 8
 endlicher \sim 7

Z

- Zeitscheibe 41
 Anhängen von \sim_n 41, 54
 Aufrollen einer \sim *siehe* Rollup
 Schema für \sim_n 46, 47
 vorberechnete \sim 57
 Vorgängerzeitscheibe 81
 Zufallsvariable 7
 diskrete \sim 3, 8
 Ereignisraum der \sim_n 7
 Menge der diskreten \sim_n 8
 Wertebereich einer \sim_n 8
 Zusammenhangskomponente 13
 Zustandsentwicklungsmodell 41, 48
 Zyklus 6, 14