

# Das Kausalnetz als Kern eines DSS

Reiner Borchert, Institut für Geoinformatik,  
Westf. Wilhelms-Universität Münster

## ZUSAMMENFASSUNG

Für ein *Decision Support System* (DSS) im Flussgebietsmanagement wird ein Kausalnetz aus verschiedenen Komponenten aufgebaut, das kausale und finale Zusammenhänge zwischen Zuständen (Ursachen- und Wirkungsbeziehungen, Ziele und Zwecke) modelliert.

Aufbauend auf einem modifizierten *Bayesian Network* werden verschiedene Knotentypen definiert, mit deren Hilfe die Ursachen von Defiziten festgestellt werden können. Für gefundene Ursachen werden Maßnahmen zur Beseitigung vorgeschlagen, deren Auswirkungen wiederum das Kausalnetz ermitteln kann.

Der Ablauf der Analysen und Inferenzen wird an einem einfachen Fallbeispiel dargestellt.

## EINLEITUNG

Eine Entscheidungs-Unterstützung (*decision support*) bedeutet für das Flussgebietsmanagement die

- Analyse des Ist-Zustands, um die **Ursachen** für den Status quo zu ermitteln, sowie die
- Abschätzung der **Folgen** und Auswirkungen eines Eingriffs.

In beiden Fällen haben wir es mit Ursache-Wirkungs-Zusammenhängen zu tun. Die Inferenzmaschine einer Wissensbasis muss also in der Lage sein, die kausalen Relationen „hat Ursache“ wie auch „hat Wirkung“ zu verfolgen, wenn sie brauchbare Aussagen über die Folgen von Eingriffen und Maßnahmen machen soll.

Für die Beschreibung von Kausalitäten liegt der Begriff „Kausalkette“ nahe, der eine eindimensionale Verkettung von Eingriffen und Zuständen suggeriert. Eine genauere Betrachtung technischer und ökologischer Zusammenhänge zeigt jedoch, dass eine Kette nicht ausreicht, um komplexe Kausalitäten zu erfassen. Ein Eingriff kann eine ganze Reihe von Folgen nach sich ziehen (erwünschte wie unerwünschte), so dass wir anstelle der Kausalkette eine als „Kausalnetz“ bezeichnete netzartige Verknüpfung von Eingriffen und Zuständen verwenden.

Das Netz wird aus Knoten (Zustände, Aktionen, Ziele) und den kausalen Relationen (Ursache, Wirkung) zwischen ihnen gebildet. Zusätzlich können finale Relationen (Mittel, Zweck) integriert werden.

Wie bereits angedeutet, verbergen sich hinter den Knoten des Kausalnetzes verschiedenartige Konzepte. Damit keine begriffliche Konfusion auftreten kann, sind wohl definierte und gut abgrenzbare semantische Konzepte zu erarbeiten.

Dieser Artikel stellt den Entwurf eines Kausalnetzes vor, das das Kernstück der Wissensbasis im FLUMAGIS-Projekt bildet. Auf der Grundlage eines *Bayesian Belief Network* wird ein Konzept entwickelt, in dem bewährte Ansätze wie *Coloured Petri Nets* (CPN) unter dem Dach einer ontologisch strukturierten Wissensbasis miteinander kombiniert werden.

## ELEMENTE DES KAUSALNETZES

Die Grundlage des Kausalnetzes bildet eine abgewandelte und erweiterte Form des *Bayesian Network*, in das an bestimmten Knotentypen prozessmodellierende *Coloured Petri Nets* gekoppelt werden können.

### Bayesian Belief Network

Da die Kausalität der Vorgänge in der Natur wegen ihrer Komplexität nie vollständig bekannt ist, sind auch exakte Prognosen über die Folgen einer Ursache a priori unmöglich. Darüber hinaus haben wir auch in den real verfügbaren Daten Lücken, Ungenauigkeiten und Fehler, die alle aus ihnen abgeleiteten Aussagen mit großen Unsicherheiten versehen.

Eine Möglichkeit, mit unsicherem, lückenhaftem Wissen auf exakt definierte Weise umzugehen, ist die Verwendung der Bayes'schen Regel für die Berechnung der Wahrscheinlichkeiten von Zuständen und Ereignissen (Bayes 1764). In *Bayesian Belief Networks* können Zustände auch dann mathematisch nachvollziehbar kalkuliert werden, wenn nicht alle Fakten bekannt sind bzw. wenn die verfügbare Datenbasis unsicher oder veraltet ist (Marcot, Holthausen et al. 2001), (Hahn, Palmer et al. 1999).

Das *Bayesian Belief Network* wird aus Knoten (*nodes*) und gerichteten Kanten (*arcs*) gebildet. Jeder Knoten symbolisiert logische Aussagen oder Attribute, die jeweils verschiedene Zustände (*states*) einnehmen können. Die Kanten verbinden einen Knoten mit anderen Knoten, die er bei einer Zustandsänderung beeinflusst (Sowa 2000).

### Coloured Petri Nets (CPN)

Ein durch vielseitigen Einsatz und Weiterentwicklungen bewährter Ansatz für die Beschreibung, Darstellung, Simulation und mathematische Überprüfung von Prozessabläufen sind Petri-Netze (Peleg, Yeh et al. 2002), (Seppelt 2003).

In der allgemeinen Grundform, den *Low Level Petri Nets*, sind vier Elementtypen zu unterscheiden, aus denen das Netz konstruiert wird (Sowa 2000):

- „Stellen“ (*places*): mögliche Systemzustände, passive Elemente
- „Transitionen“ (*transitions*): Übergänge zwischen Stellen, aktive Elemente. Verfügen über Eingangs-Stellen (*input places*), in denen die Voraussetzungen für die Transition definiert sind, und über Ausgangs-Stellen (*output places*), auf die sich die Transition auswirkt.
- Kanten (Pfeile, *arcs*): verbinden Stellen mit Transitionen (*input places*) und Transitionen (*output places*) mit Stellen
- „Marken“ (*tokens*): bewegliche Elemente, in Stellen lokalisiert, werden durch Transitionen bewegt, neu erzeugt oder gelöscht.

Im *Low Level Petri Net* wird nur die Anzahl von Marken (*tokens*) berücksichtigt. In sog. *Coloured Petri Nets* hingegen können Datentypen deklariert und den Marken zugeordnet werden. Analog zu prozeduralen Programmiersprachen werden Stellen mit typisierten Aufrufparametern versehen, für die Marken der passenden Datentypen verwendet werden können (Kristensen, Christensen et al. 1998).

Hierarchische Petrinetze setzen sich aus „Unternetzen“ (*subnets*) zusammen, die ähnlich wie Prozeduren in Programmiersprachen verwendet werden (Desel, Freytag et al. 1997).

## DER KAUSALNETZ-KNOTEN

Der im Kausalnetz verwendete Knotentyp entspricht im Wesentlichen dem des *Bayesian Network*. Der Grundtyp des Kausalknotens (*CausalNode*) definiert Eigenschaften, die eine Verknüpfung mit anderen Kausalknoten über die Ursachen- oder über die Wirkungsrelation erlaubt (Eigenschaften *causingNodes* und *effectedNodes*).

Class	CausalNode of Node
Properties	causingNodes: CausalNode (multiple) effectedNodes: CausalNode (multiple)

Im Folgenden werden die vom Kausalknoten abgeleiteten Knotentypen vorgestellt, die verschiedene Rollen im Kausalnetz übernehmen und zum Teil über zusätzliche Eigenschaften verfügen, während sie die hier genannten Eigenschaften vom Basis-Knotentyp erben.

### Aktionsknoten

Aktionsknoten behandeln von Menschenhand durchgeführte Aktionen (Eingriffe, Maßnahmen). Auch solche Knoten haben Wirkungen auf andere Knoten (Eigenschaft *effectedNodes*), sind aber vor allem durch die von der Aktion verfolgte Finalität (Zweck, Intention und Zielsetzung) gekennzeichnet.

Aktionsknoten bekommen also eine zusätzliche Eigenschaft zur Herstellung von Mittel-Zweck-Relationen zu Zweckknoten (Eigenschaft *aimedNodes*).

Zur Spezifizierung der repräsentierten Aktion kann ein *Coloured Petri Net* konstruiert und mit dem Aktionsknoten verbunden werden. Das CPN nimmt die Effekt- und Zweck-Knoten als Stellen (*places*) auf, während die betroffenen Objekte und ihre Attribute als Marken (*tokens*) durch das Netz „wandern“. Die Einzelschritte der Aktion bilden die Transitionen (*transitions*) zwischen den Stellen.

Class	<b>ActionNode</b> of CausalNode
Properties	aimedNodes: AimNode (multiple) petriNet: ColouredPetriNet

### Zustandsknoten

Der Zustandsknoten hat Eigenschaften zur Spezifikation des von ihm repräsentierten Zustands (*objectClasses*: betroffene Objektklassen, *attribute*: verwendetes Attribut, *allowedValues*: erlaubte/mögliche Attributwerte).

Class	<b>StatusNode</b> of CausalNode
Properties	objectClasses: ObjectClass (multiple) attribute: ObjectAttribute allowedValues: AnyType (multiple) certainty: (unknown, likelihood, certain)

Die Eigenschaft *certainty* definiert den Gewissheitsgrad der Kenntnis über den Zustand dieses Knotens. Bei Gewissheitsgrad *unknown* werden alle möglichen Attributwerte a priori als gleich wahrscheinlich angesehen. *Likelihood* steht für unterschiedliche Wahrscheinlichkeiten, die für jeden Attributwert angenommen werden können. Bei Gewissheit (*certain*) bekommt der als sicher angesehene Attributwert die Wahrscheinlichkeit 100%, alle anderen Werte dagegen die Wahrscheinlichkeit 0%.

### Zweckknoten

Das Ziel oder der Zweck einer Aktion ist die Herstellung eines erwünschten Zustands von Objektattributen. Zweckknoten bilden somit das Gegenstück zu Aktionsknoten, die das Mittel zur Erreichung des Ziels beschreiben. Dementsprechend haben Zweckknoten zusätzlich die Eigenschaft *aimingNodes*:

Class	<b>AimNode</b> of StatusNode
Properties	aimingNodes: ActionNode (multiple)

### Indexknoten

Attribute von Geoobjekten oder Messwerte werden häufig zu Index- oder Indikatorwerten zusammengefasst, damit eine allgemeinere Aussage über den

Zustand eines Gewässers gemacht werden kann (Beispiel: Gewässergüte). In die Berechnung eines Indexes gehen also eine Reihe von Einzelfaktoren ein. Im strengen logischen Sinn kann man hier zwar nicht von Kausalität reden, es macht aber praktischen Sinn, diese Einzelfaktoren als Ursachenknoten für den Indexknoten aufzuführen. Ein Indexwert zeigt einen Zustand an, hat aber keine kausalen Auswirkungen auf andere Knoten, außer auf Indexknoten, in deren Berechnung er eingeht.

Class	<b>IndexNode</b> of StatusNode
-------	--------------------------------

### Handlungsbedarfsknoten

Bei der Analyse des Datenbestandes kann festgestellt werden, dass „Handlungsbedarf“ besteht, sei es, dass die realen Werte auf ein Defizit hindeuten, sei es, dass sie ein ausbaufähiges Potenzial aufweisen. In beiden Fällen sind Maßnahmen notwendig, um entweder das Defizit zu beseitigen oder das Potenzial zu nutzen.

Dieser Knotentyp ist der abstrakte Grundtyp für die folgenden beiden Knotentypen und enthält als zusätzliche Eigenschaft eine Relation zu einem Aktionsknoten (*requiredActionNode*). Auf diesem Wege können geeignete Maßnahmen mit einem Handlungsbedarfsknoten verknüpft werden.

Class	<b>ActionCallNode</b> of StatusNode
Properties	requiredActionNode: ActionNode

### Defizitknoten

Defizite in der Landschaft werden dort lokalisiert, wo Abweichungen vom Leitbild festgestellt werden können. Ein Defizitknoten definiert einen Zustand, der vom Idealzustand um mehr als einen tolerierbaren Wert abweicht. Welche Attributwerte auf ein Defizit hindeuten, wird durch die Eigenschaft *allowedValues* festgelegt. Diese Eigenschaft listet Attributwerte auf, die auf ein Defizit hinweisen.

Defizitknoten enthalten eine Relation zu Aktionsknoten (*requiredActionNode*). Auf diesem Wege können Maßnahmen mit dem Defizit verknüpft werden, die geeignet sind, es zu beseitigen.

Class	<b>DeficitNode</b> of ActionCallNode
-------	--------------------------------------

### Potenzialknoten

Als positives Gegenstück zu Defiziten lassen sich auch Potenziale in der Landschaft feststellen, die die Möglichkeit bieten, einem angestrebten Ziel (Leitbild) näher zu kommen. Ein Potenzialknoten unterscheidet sich vom Defizitknoten lediglich in der Interpretation der Eigenschaft *allowedValues*: sie enthält hier erwünschte, positiv gedeutete Werte. Entsprechen die realen Attributwerte den „erlaubten“, wird ein Potenzial erkannt und eine Maßnahme

vorgeschlagen (Eigenschaft *requiredActionNode*), die geeignet ist, das Potenzial auszuschöpfen bzw. zur Entfaltung zu bringen.

Class	<b>PotentialNode</b> of ActionCallNode
-------	--

## INFERENZEN

Der Nutzen eines Kausalnetzes besteht darin, dass einfache Inferenz-Methoden angewandt werden können, um die vorhandenen Daten zu analysieren und Schlüsse daraus zu ziehen. Die grundlegende Methode beruht darauf, dass prinzipiell jeder Knoten des Netzes als vom Nutzer gewählter Ausgangspunkt für Analysen und Prognosen dienen kann. In einem iterativen Prozess werden die miteinander verknüpften Knoten entweder (bei Analysen) über die Ursachen-Relation (Eigenschaft *causingNodes*), oder (bei Prognosen) über die Wirkungs-Relation (Eigenschaft *effectedNodes*) verfolgt - so lange, bis keine Relationen zu weiteren Knoten mehr bekannt sind. Je nach Knotentyp sind weitere Inferenzfragen möglich:

Knotentyp	Inferenzfrage	Eigenschaft
Aktionsknoten	Welche Folgen hat eine Maßnahme? Welche Ziele verfolgt diese Maßnahme?	<i>effectedNodes</i> <i>aimedNodes</i>
Indexknoten	Welches sind die Ursachen für diesen Zustand? Welches sind die Wirkungen dieses Zustands?	<i>causingNodes</i> <i>effectedNodes</i>
- zusätzlich: Zweckknoten	Mit welchen Maßnahmen kann dieses Ziel erreicht werden?	<i>aimingNodes</i>
Defizitknoten	Mit welchen Maßnahmen kann dieses Defizit beseitigt werden?	<i>requiredActionNode</i>
Potenzialknoten	Mit welchen Maßnahmen kann dieses Potenzial umgesetzt werden?	<i>requiredActionNode</i>

## TECHNISCHE UMSETZUNG

Für den Aufbau einer Wissensbasis existiert eine ganze Reihe mehr oder weniger brauchbarer Tools. Im FLUMAGIS-Projekt hat sich der Einsatz des Programms „*Protégé-2000*“ bewährt. Es handelt sich dabei um ein *Open-Source*-Projekt, das an der Stanford University entwickelt wird. *Protégé* ist ein auf *Frame Logic* basierender Ontologie-Editor, der über weit reichende Java-Schnittstellen verfügt und dadurch beliebig durch eigene Module (*Plug-ins*) erweiterbar ist (Noy, Fergerson et al. 2000).

Das Kausalnetz lässt sich in seiner statischen Form leicht als Ontologie realisieren, indem die verschiedenen Knotentypen als Klassen und ihre Eigenschaften als *Slots* modelliert werden. Da *Protégé* nicht allein der Erstellung von Ontologien dient, sondern auch Daten in Form von Instanzen

verwalten kann, werden alle bekannten Netzknoten mit ihren Relationen in die Wissensbasis eingegeben.

Für die Inferenzen im Kausalnetz (Feststellung von Defiziten, Ermittlung von geeigneten Maßnahmen zur Beseitigung derselben, Prognosen der Auswirkungen) ist die Entwicklung eines speziellen Plug-ins notwendig, ebenso für die Kalkulation der Zustands-Wahrscheinlichkeiten auf Grund des *Bayes'schen* Gesetzes und die Abläufe in *Coloured Petri Nets*.

Für den Einsatz in der Praxis ist außerdem eine Anbindung an die Geodaten erforderlich, die es der Wissensbasis ermöglicht, diese auszuwerten. Der Inhalt der Datenbanken ist letztlich der limitierende Faktor für die Analyse- und Prognose-Fähigkeiten der Wissensbasis.

## **EIN FALLBEISPIEL**

Die konkrete Ausgestaltung des Kausalnetzes ist von den Fragestellungen abhängig, die es beantworten soll, und natürlich von den verfügbaren Daten. In unserem Fallbeispiel soll die Gewässerstrukturgüte-Kartierung analysiert werden, damit vorhandene Defizite in Hinblick auf signifikante Abweichungen vom Leitbild. Die Gewässerstrukturgüte-Kartierung besteht aus einer Reihe von Einzelparametern, Gewässersohle, Ufer und Umland betreffend, die zu Hauptparametern als Index zusammengefasst werden. Aus allen Hauptparametern kann wiederum ein Gesamt-Index berechnet werden. Die Einzelparameter werden jeweils für 100-Meter-Abschnitte erhoben.

Für die Analyse der Strukturgüte wählt der Anwender einen Defizit- oder Index-Knoten des Kausalnetzes aus – in diesem Falle den Knoten, der sich auf den Gesamt-Index der Strukturgüte bezieht. Bei der Überprüfung der Strukturgütedaten werden alle Gewässerabschnitte, deren Indexwert im Defizitbereich liegen, erkannt und aufgelistet (Defizitfeststellung).

Die Ursachenanalyse eines defizitären Gewässerabschnitts betrachtet zunächst die Ursachen-Knoten des Gesamtindex-Knotens. Dabei handelt es sich wiederum um Index-Knoten, die jeweils einen der Hauptparameter repräsentieren. Für alle diese Ursachen-Knoten wird nun ihrerseits durch einen Abgleich mit den Kartierungsdaten eine Defizitfeststellung durchgeführt.

Dabei zeigt sich, dass der Indexknoten zum Parameter „Gewässersohle“ die schlechte Gesamtbewertung des Abschnitts „verursacht“ hat. Somit lässt sich das Defizit bereits auf die Gewässersohle eingrenzen. Mit den Ursachen-Knoten des „Gewässersohlen“-Indexknotens erfolgt entsprechend der nächste Iterationsschritt. Als defizitäre Einzelparameter treten „Rückstau vorhanden“ und „Laufstrukturen fehlen“ in Erscheinung. Für das erste Defizit kommen als Ursachen-Knoten „Querbauwerk vorhanden“ und/oder „starke Sedimentierung“ in Betracht, für letzteres „Längsverbau“ und/oder „Sohlenverbau vorhanden“.

Eine Suche nach entsprechenden Objekten (Wehren, Verbauen etc.) des untersuchten Gewässerabschnitts im Geodatenbestand kann nun die verursachenden Objekte zu Tage fördern, sofern entsprechende Daten verfügbar sind. Die Information, nach welchen Objektklassen gefahndet werden muss, findet sich bei den passenden Zustandsknoten in der Eigenschaft *objectClasses*.

Die verursachenden Objekte lassen sich finden, wenn die Kausalität von Knoten zu Knoten über die Eigenschaft *causingNodes* verfolgt werden kann. Möglicherweise liegen aber keine gesicherten Informationen beispielsweise über einzelne Bauwerke oder Einleiter vor. Dann lässt sich „nur“ eine allgemeine Aussage über wahrscheinliche bzw. mögliche Objekte machen, die sich nicht konkret verifizieren lässt. Bei einem lückenhaften Datenbestand ist damit natürlich relativ häufig zu rechnen. Ist dagegen der Datenbestand vollständig, könnte ein Nicht-Auffinden von verursachenden Objekten auf ein tatsächliches Nicht-Vorhandensein schließen.

Die hier aufgezeigte iterative Inferenzkette lässt sich ebenso in der umgekehrten Richtung durch Verfolgen der Eigenschaft *effectedNodes* gehen. Dadurch lassen sich Auswirkungen beispielsweise der Neuanlage eines Wehres bzw. der Entfernung eines solchen prognostizieren.

Da ein Defizitknoten mit der Eigenschaft *requiredActionNodes* über Verbindungen zu Maßnahmenvorschlägen (Aktionsknoten) verfügt, kann man immer, wenn solche Relationen vorliegen, zu einem Maßnahmenvorschlag gelangen, der aufzeigt, wie das entsprechende Defizit beseitigt werden könnte. Verfügt der Aktionsknoten über ein passendes Petri-Netz, dann kann diese Maßnahme „durchgeführt“ werden – das bedeutet, dass das Petri-Netz alle Veränderungen am Datenbestand vornimmt, die mit der Umsetzung dieser Maßnahme verbunden sind. Anschließend kann eine Prognose über den Erfolg der Maßnahme über die Verfolgung der Eigenschaft *effectedNodes* durchführen.

## AUSBLICK

Der hier vorgestellte Ansatz ist noch sehr grob-schematisch und muss im Praxiseinsatz noch weiter verfeinert werden. Viele technische Fragen sind noch nicht abschließend beantwortet. Insbesondere die Modellierung von kontinuierlichen und/oder parallelen Prozessen muss noch praxistauglich umgesetzt werden. Aufbauend auf den beschriebenen Komponenten soll ein funktionsfähiger Prototyp entstehen, der in der Lage ist, einige typische Fallbeispiele zu bearbeiten.

## LITERATUR

Bayes, T. (1764). "Essay towards solving a problem in the doctrine of chances." Philosophical Transactions of the Royal Society of London.

- Desel, J., T. Freytag, et al. (1997). Causal-semantic-based simulation and validation of high-level Petri nets. Proceedings of the European Simulation Multiconference, Istanbul.
- Hahn, M., R. N. Palmer, et al. (1999). Prioritizing Sewer Line Inspection with an Expert System. 26th Annual Water Resources Planning and Management Conference (ASCE), Tempe, Arizona.
- Kristensen, L. M., S. Christensen, et al. (1998). "The practitioner's guide to coloured Petri nets." International Journal on Software Tools for Technology Transfer **2**(2): 98-132.
- Marcot, B. G., R. S. Holthausen, et al. (2001). "Using Bayesian belief networks to evaluate fish and wildlife population viability under land management alternatives from an environmental impact statement." Forest Ecology and Management **153**(1-3): 29-42.
- Noy, N. F., R. W. Ferguson, et al. (2000). The knowledge model of Protégé-2000: combining interoperability and flexibility. 2th International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000), Juan-les-Pins, France.
- Peleg, M., I. Yeh, et al. (2002). "Modeling biological processes using Workflow and Petri Net models." Bioinformatics **18**(6): 825-837.
- Seppelt, R. (2003). Computer-Based Environmental Management. Weinheim, Wiley-VCH Verlag.
- Sowa, J. F. (2000). Knowledge representation: logical, philosophical, and computational foundations. Pacific Grove, CA, Brooks/Cole.