
MATHEMATIQ

Der Newsletter der MathSIG
(Interessensgruppe innerhalb der Mensa Österreich)

Ausgabe 25

<http://www.hugi.scene.org/adok/mensa/mathsig/>

Editorial

Liebe Leserinnen und Leser!

Dies ist die fünfundzwanzigste Ausgabe von MATHEMATIQ, dem Newsletter der MathSIG. Die MathSIG wurde gegründet, um die spezifischen Interessen mathematisch hochbegabter Menschen zu fördern. In erster Linie soll sie sich also den Themengebieten Mathematik, Informatik, Physik und Philosophie widmen. Beiträge von Lesern sind herzlich willkommen. Wenn in ihnen mathematische Sonderzeichen vorkommen, bitte ich aber, sie zwecks möglichst einfacher und fehlerfreier Formattierung im $\text{T}_{\text{E}}\text{X}$ -Format einzusenden. Als Vorlage ist eine Fassung des jeweils aktuellen Newsletters im $\text{T}_{\text{E}}\text{X}$ -Format auf Anfrage bei mir erhältlich. Außer Artikeln sind natürlich auch Illustrationen für das Titelblatt willkommen. Die Rechte an diesen müssen aber eindeutig bei euch selbst liegen, Kopieren von Bildern aus dem Internet ist nicht erlaubt.

Hinweis: Autoren sind für den Inhalt ihrer Artikel oder Werke selbst verantwortlich. Die in MATHEMATIQ veröffentlichten Beiträge widerspiegeln ausschließlich die Meinung ihrer Autoren und nicht jene des Vereins Mensa. Die Zusendung von Beiträgen gilt auch als Einverständnis zu deren Veröffentlichung in MATHEMATIQ.

Diese Ausgabe beschäftigt sich mit Constraint Propagation als Methode der Künstlichen Intelligenz zur Lösung von Logikaufgaben durch den Computer.

In diesem Sinne: Viel Spaß beim Lesen und Lernen!

Claus D. Volko, cdvolko@gmail.com

Logiktrainer mit Computern lösen

Anlässlich einer diesbezüglichen Anfrage beim Mensa-Café möchte ich darüber schreiben, wie man Aufgaben aus dem PM Logiktrainer mit Hilfe eines Computers lösen kann.

Zunächst einmal: Ja, man kann Computer programmieren, dass sie diese Aufgaben lösen können! Es ist richtig, dass Intelligenz nichts spezifisch Menschliches ist. An der TU Wien lernt man, dass intelligentes Verhalten wahrscheinlich nur eine Frage der Berechnung ist und man daher den Computer (der im Deutschen ja auch als "Rechner" bezeichnet wird) ohne weiteres so programmieren kann, dass er intelligentes Verhalten an den Tag legt, welches man bislang vielleicht nur Menschen zugetraut hätte.

Die einzige Schwierigkeit beim Lösen von Logiktrainer-Aufgaben durch einen Computer besteht darin, die Hinweise in eine für den Computer verständliche Sprache zu übersetzen. Die Verarbeitung natürlicher Sprache ist noch nicht so weit fortgeschritten. Aber: Einmal übersetzt, kann der Computer die Logiktrainer-Aufgaben problemlos lösen! Sehen wir uns eine typische Logiktrainer-Aufgabe an: Drei Mädchen, drei Früchte, drei Süßigkeiten. Bei diesen Aufgaben gilt immer, dass genau eine Frucht von genau einem Mädchen gegessen wird, genau eine Süßigkeit von genau einem Mädchen bevorzugt wird und auch genau eine Frucht mit genau einer Süßigkeit assoziiert ist. Im PM-Logiktrainer wird, um die Lösung zu erleichtern, in der Regel eine Tabelle wie die folgende vorgegeben:

| | Zitrone | Apfel | Birne | Bonbon | Zucker | Lolli |
|---------|---------|-------|-------|--------|--------|-------|
| Anna | | | | | | |
| Berta | | | | | | |
| Claudia | | | | | | |
| Bonbon | | | | | | |
| Zucker | | | | | | |
| Lolli | | | | | | |

Angenommen, der erste Hinweis lautet, dass Anna gern Zitronen isst. Das bedeutet, dass wir in diese Tabelle in der Zelle, die Anna und die Zitronen verbindet, ein Plus setzen. Da aber nur Anna die Zitrone isst und nicht auch die anderen Mädchen und Anna auch nur die Zitrone isst und nicht auch die anderen Früchte, bedeutet ein Plus in der Zelle "Anna - Zitrone", dass wir in demselben Kästchen in den Zellen, die in der gleichen Spalte und in der gleichen Zeile liegen, ein Minus setzen können.

Die Regel, dass ein Plus automatisch lauter Minus in der gleichen Zeile und der gleichen Spalte impliziert, kennt auch ein entsprechend programmierter Computer. Intern wird der Computer mit einer Datenstruktur arbeiten, die dieser Tabelle sehr ähnlich ist. Er wird wissen: Wenn ich irgendwo ein Plus setze, dann kann ich anderswo ein Minus setzen. Diese Methode nennt sich "Constraint Propagation". Gemeint ist: Eine "Constraint", also eine Einschränkung der Möglichkeiten, wo ich ein Plus

oder ein Minus setzen darf, hat auch Auswirkungen auf andere Felder; ich "propagiere" also diese "Constraint" weiter. "Constraint Propagation" ist die wichtigste Methode, mit der ein Computerprogramm Logikaufgaben lösen kann.

| | Zitrone | Apfel | Birne | Bonbon | Zucker | Lolli |
|---------|---------|-------|-------|--------|--------|-------|
| Anna | + | - | - | | | |
| Berta | - | | | | | |
| Claudia | - | | | | | |
| Bonbon | | | | | | |
| Zucker | | | | | | |
| Lolli | | | | | | |

Wenn ein weiterer Hinweis nun lautet, dass Anna keine Bonbons mag, dann bedeutet das zwar in erster Linie, dass ich im Feld "Anna - Bonbon" ein Minus setzen muss; es hat aber auch die Auswirkung, dass ich im Feld "Bonbon - Zitrone" ein Minus setzen muss, weil Anna ja mit der Zitrone positiv verknüpft ist. Das heißt, alles, was für Anna gilt, gilt auch für die Zitrone, und umgekehrt. Auch das ist "Constraint Propagation". Der Computer muss lediglich so programmiert werden, dass er jedesmal, wenn ein Minus gesetzt wird, überprüft, ob in der gleichen Zeile oder in der gleichen Spalte ein Plus steht; wenn das der Fall ist, dann muss er diese Constraint "propagieren".

Die Aufgaben im PM Logiktrainer können ohne Ausnahme durch "Constraint Propagation" gelöst werden. Im Prinzip geht der Computer genauso vor, wie es auch ein geübter menschlicher Rätsellöser tun würde.

Davon abgesehen, besteht für den Computer noch eine weitere Möglichkeit, Logikaufgaben zu lösen: nämlich indem er systematisch alle Möglichkeiten durchprobiert und überprüft, ob sie die "Constraints" erfüllen. Diese Methode nennt man "Backtracking". Sie ist langsam, führt aber in jedem Fall zum Ziel, selbst wenn "Constraint Propagation" nicht reicht (was zum Beispiel bei Sudoku-Rätseln vorkommen kann).

Ich hoffe, dass der Artikel interessant war und man daraus lernen konnte.

Claus D. Volko, cdvolko@gmail.com