Automatische Orientierung im Labyrinth*)

Automatic path-finding in the maze

von R. EIER und H. ZEMANEK

Technische Hochschule, Wien

Elektron. Rechenanl. 2 (1960), H. 1, S. 23—31

Manuskripteingang: 19. 1. 1960

Die Orientierung im Labyrinth wird als Automatisierung des Ariadnefadens aufgefaßt. Die erforderlichen Informationen, für das Zielfinden und für die Wegwiederholung oder den Rückweg, müssen dazu gespeichert und durch ein entsprechendes Programm ausgewertet werden. Methode und ausgeführtes Modell weisen gegenüber bisherigen verschiedene Verbesserungen auf.

Path-Finding in the maze is considered as automatization of Ariadne's thread. The necessary informations, for goal-seaking as well as for repeating the way or finding back to the start, must be stored and then may be applied by a suited program. Method and hard-ware model display several improvements compared to known ones.

1. Einleitung

1.1 Die Theseus-Sage [1]

Theseus, der Sohn des Königs Aigeus, fährt als Mitglied des alljährlichen Tributs der Stadt Athen an den Minotauros von sieben Jünglingen und sieben Jungfrauen nach Kretä, um Athen von dieser Last zu befreien. Der Minotauros, ein gewaltiger, von den Göttern in Irrsinn gestürzter Stier, wohnt im Labyrinth, einem Palast von verwirrender Weiträumigkeit und unübersichtlicher Anlage, in dem sich die Opfer bisher hoffnungslos verirrten.

Theseus aber erhält von Ariadne, der Tochter des Königs Minos von Kreta, einen Wollknäuel, den er auf dem Weg in das Labyrinth abwickelt und mit dessen Hilfe er den Minotauros findet. Nach siegreichem Zweikampf zeigt der Wollfaden dem Theseus und seinen Begleitern auch den Weg aus dem Labyrinth hinaus.

Die Idee des Ariadnefadens stammt jedoch nicht von Ariadne, sondern von Daidalos, dem Erbauer des Labyrinths. König Minos ist sehr ergrimmt, als er erfährt, daß Daidalos an Ariadne diese wirksame Methode verraten hat, und sperrt ihn samt seinem Sohne Ikaros in das Labyrinth ein; die beiden verfertigen sich daraufhin unter Verwendung von Wachs Flügel und fliehen. Ikaros nähert sich im Fluge trotz der Warnung seines Vaters zu sehr der Sonne, das Wachs schmilzt, und er stürzt in das nun nach ihm benannte Ikarische Meer. Daidalos erreicht Sizilien und wird von König Kokalos aufgenommen, der seine Auslieferung an König Minos verweigert.

Theseus nimmt Ariadne auf der Flucht mit. Auf Naxos läßt er sie dann allein zurück — aber das ist eine andere Geschichte.

1.2 Das Problem der Orientierung im Labyrinth

In der Theseus-Sage haben ebenso wie in der Herakles-Sage viele Heldentaten den Charakter einer Problemlösung. Die Orientierung im Labyrinth ist als ganz schwere Aufgabe herausgehoben, und neben ihr verschwindet der Zweikampf mit dem Minotauros geradezu. In dieser Aufgabe spiegelt sich wohl jenes Unbehagen, das jeder Mensch erlebt, wenn er nahe daran ist, in einer nur teilweise bekannten Gegend die Orientierung zu verlieren. Die Einzelheiten — häufig sind sie durchaus unbewußte Daten —, nach denen man sich zurechtfindet, bilden nur eine Ausschmückung zur Grundaufgabe, sich in einem Labyrinth mit relativ wenigen und nicht einprägsamen Landmarken zurechtzufinden. Es gibt einen Orientierungssinn, der sich nicht beschreiben läßt und der, vielleicht mit ein bißchen Übung verbunden, manche Leute auf wunderbare Weise leitet.

Der Ariadnefaden aber ist ein Hinweis, daß die mechanische Hilfe doch sicherer ist, und im Zeitalter der Rechenmaschinen und Automaten muß der Versuch, das Orientierungsproblem mit elektronischen Mitteln zu lösen, große Anziehungskraft haben.

Die genauere Untersuchung zeigt zwei Fragestellungen oder mindestens zwei Hauptabschnitte: zuerst geht es darum, überhaupt einen Weg vom Start zum Ziel zu finden; und dann um die Wiederholung dieses Weges auf kürzeste Weise oder um den Weg zurück vom Ziel zum Start, ebenfalls auf kürzeste Weise. Demgemäß braucht man auch zwei Arten der Information: solche, die bei der Zielsuche unnötige oder gar endlose Wiederholungen verhindert, und solche, die einfach den kürzesten Weg angibt.

Führt man das Labyrinth auf seine topologische Struktur zurück, so erkennt man sofort die Verwandtschaft mit dem Schaltnetzwerk: es gibt Knoten und offene oder geschlossene Pfade. Ein Verfahren für die Orientierung im Labyrinth läßt sich daher ohne weiteres auf die Orientierung in Schaltnetzwerken anwenden. In dem Maße, in dem unsere Informationsübertragungsnetze so kompliziert werden, daß das Labyrinth des Daidalos vergleichsweise primitive Struktur hatte, in diesem Maß wird die automatische Orientierung im Schaltnetzwerk eine gewichtige Fragestellung. Das Modell erscheint nicht mehr als Spielzeug, sondern als Schritt auf dem Weg sogar zu wirtschaftlicher Bedeutung.

1.3 Bisherige Modelle

Obwohl es darüber nur eine einzige ausführlichere Veröffentlichung an wenig verbreiteter Stelle zu geben scheint [2], ist die „Maus im Labyrinth" von *C. E. Shannon* verhältnismäßig gut bekannt. Auf einem Quadrat von fünf mal fünf Einheitslängen kann man nach Belieben Trennwände setzen. Der Suchfinger, nur für spektakuläre Zwecke durch einen Magneten mit Mausattrappe vom Abstrakten ins Konkrete projiziert, liefert dem Relaisspeicher die erforderlichen Informationen. Nach dem ersten Lauf schaltet sich das Gerät selbst vom Zielsuchprogramm auf das Wegwiederholungsprogramm, und wenn während der Wiederholung durch Veränderung des Labyrinths der Finger ins Kreisen kommt, schaltet es wieder auf das Zielsuchprogramm zurück. Die Einzelheiten von Speicher und Programm sind nicht bekannt geworden; wir werden am Schluß des Beitrags noch einmal auf einige Unterschiede zwischen *Shannons* Modell und unserem zurückkommen.

*) Mitteilung aus dem Institut für Niederfrequenztechnik der Technischen Hochschule Wien

Eine mehr von der experimentellen Psychologie herkommende Anordnung stammt von *J. A. Deutsch* [3], dem es vorwiegend auf die Analogie zu den Rattenversuchen im Labyrinth ankam. Auch in dieser Veröffentlichung werden keine Einzelheiten über Speicher und Programme gegeben. Es handelt sich ebenfalls um eine Relaisschaltung unter Verwendung von Drehwählern.

2. Die Aufgabenstellung

Das Absuchen eines Labyrinths ohne System wird im allgemeinen erfolglos sein. Ein Teil der Gänge wird immer wieder betreten werden, während ein anderer völlig unbeachtet bleiben mag.

Auch die Vorschrift, an jedem Verzweigungspunkt den nächsten rechten (oder linken) Ausgang zu versuchen, bringt keine Lösung, wenn in sich geschlossene Wege bestehen. Wird nämlich einer davon einmal durchlaufen, so ermöglicht es diese Vorschrift nicht, aus der Kreisbahn auszubrechen. Sie allein gestattet nicht einmal die Feststellung, daß dieser Fall eingetreten ist. Aber selbst die Erkenntnis, daß es so ist, muß nicht zum Ziel führen, denn ein systemloses Ausbrechen schützt nicht vor beliebig vielen Wiederholungen, von Kreisbahn zu Kreisbahn ausbrechend.

Um mit Sicherheit alle möglichen Wege eines Labyrinths zu durchlaufen und dabei das gegebene Ziel zu finden, müssen alle abgesuchten Verzweigungspunkte markiert werden. Der Ariadnefaden ist für die Markierung bestens geeignet.

Er zeigt an, wo man noch nicht war, welchen Weg man zweimal durchlaufen hat und wie man zum Ausgangspunkt zurückfindet. Eine geringe Verschärfung der Aussagen und Definitionen erlaubt eine Systematik, die sich für die Automatisierung eignet.

Zunächst sei das Prinzip des Ariadnefadens noch in der üblichen, zu wenig präzisen Sprache beschrieben. Beim Absuchen des Labyrinths wird der Faden ausgelegt; ein Ende ist beim Eingang und das andere am jeweiligen Standort, wo es den Rückweg anzeigt. Im allgemeinen wird ein Verbindungsgang nur einmal durchlaufen, so daß ein einfacher Faden darin liegen bleibt. Nur in Sackgassen ist man gezwungen, bis zur Möglichkeit einer anderen Abzweigung zurückzukehren, so daß in der Sackgasse ein doppelter Faden liegen bleibt und von einem dritten Betreten abhält. Bei der Abzweigung wird man nur unmarkierte Verbindungsgänge betreten; erst wenn alle möglichen Fortschreitungsrichtungen ebenfalls als Sackgassen gekennzeichnet sind, wird man sich entschließen müssen, in der Richtung des einfachen Fadens zurückzulaufen bis zu einer anderen Verzweigung, wodurch der ganze abgesuchte Teilkomplex als Sackgasse gekennzeichnet erscheint.

Es kann aber auch der Fall eintreten, daß ein unmarkierter Verbindungsgang zu einer bereits einmal markierten Verzweigung zurückführt und damit also einen Kreisweg schließt. In einer solchen Verzweigung liegen dann drei einzelne Fäden; zwei davon führen zurück zum Eingang (wobei einer der beiden an dieser Verzweigung noch einmal vorbeiführt), der dritte weist in den Kreisweg hinein. Wenn der Faden keine besonderen Merkmale trägt, weiß man mit Sicherheit nur, daß der Gang, den man gerade gekommen ist, der eine Rückweg sein muß, während eine Aussage über die beiden anderen Richtungen, in denen noch einfache Fäden liegen, nicht möglich ist. Darum wird in dieser Situation der Gang, den man gerade gekommen ist, zurückverfolgt, wodurch er mit doppeltem Faden als Sackgasse gekennzeichnet ist. Diese Suchvorschrift erlaubt es, Gänge, die mit doppelten Fäden belegt sind, vom weiteren Durchsuchen auszuschließen und Kreiswege in Sackgassen umzuwandeln. Es werden alle noch nicht besuchten Gänge erkannt und nach dem Ziel abgesucht. Am Ende weist dann ein einfacher Faden den Weg vom Eingang zum Ziel und umgekehrt.

3. Labyrinth und Schaltnetzwerk

In Bild 1a und 1b sind Labyrinth und Schaltnetzwerk einander gegenübergestellt. Die Verwandtschaft ist sofort zu erkennen. Die Wege und Verzweigungen im Labyrinth bieten die Entscheidung zwischen zwei Möglichkeiten: offen oder gesperrt. Im Schaltnetzwerk bieten sich die gleichen Möglichkeiten; dabei kann man entweder an ein reines Schaltnetzwerk denken oder aber, und dies hat die größere praktische Bedeutung, an die Kombination eines Schaltnetzwerks mit einem Übertragungsnetzwerk für

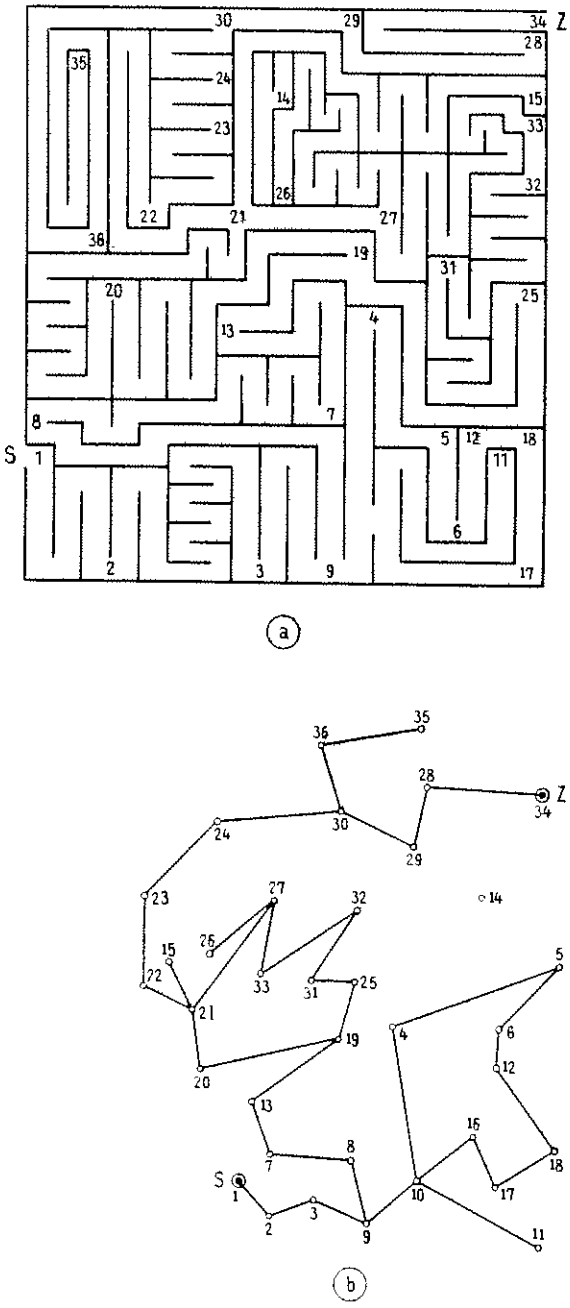


Bild 1. Labyrinth und Schaltnetzwerk.

a) Die Verzweigungen und Kreuzungspunkte sind mit 1 bis 36 numeriert. Die Verbindungsgänge könnten auf Grund ihrer Endpunkte mit Doppelnummern bezeichnet werden. Im Labyrinth sind zwei Kreiswege und vier Sackgassen.

b) Die Knoten sind mit 1 bis 36 numeriert. Die Zweige könnten auf Grund der Knoten, die sie verbinden, mit Doppelnummern bezeichnet werden. Im Netzwerk sind zwei Maschen. Die Struktur ist die gleiche wie in Bild 1a.

Material, Energie oder Information. So stellen, um ein konkretes Beispiel zu geben, Bündel von Telegraphen- oder Telephonverbindungen ein Schaltnetzwerk in einem etwas allgemeineren Sinn dar, wo zwischen den Knoten die Alternative „es ist noch eine Verbindung frei" oder „es sind alle Verbindungen besetzt" besteht. Die Beschreibung durch die symbolische Logik ergibt sich von selbst.

Eine Erweiterung entsteht dadurch, daß im Fall der nicht gesperrten Verbindung noch eine zweite Entscheidung von Interesse sein kann, nämlich die Entscheidung über die Durchlaufrichtung. Diese Entscheidung kann durch eine zusätzliche Binärinformation getroffen werden, und es erscheint sinnvoll, *alle* derartigen Erweiterungen durch zweiwertige logische Variable zu beschreiben, auch wenn dadurch eine gewisse Redundanz in Kauf genommen wird, weil dadurch wertvolle Übersichtlichkeit gewonnen werden kann.

Jedes Labyrinth läßt sich also als Netzwerk darstellen, wobei die Entsprechungen

Verzweigungen oder Kreuzungspunkte	— Knoten
Verbindungsgänge	— Zweige
Kreiswege	— Maschen

gelten. Übrigens besteht keineswegs eine Beschränkung auf planare Netzwerke, sondern es sind gekrenzte Zweige möglich, weil ja auch im Labyrinth die Gänge beliebig verwunden sein können.

Diese Entsprechungen werden im folgenden benutzt, und es wird je nach Nützlichkeit das eine oder andere Vokabular verwendet.

4. Das Konzept des automatisierten Ariadnefadens

4.1 Einteilung der Aufgabe

Die Aufgabenstellung wird unterteilt in die *Zielsuche*, die die Probleme der Orientierung zu lösen hat, und in die *Wegwiederholung*, bei der die Informationen aus der Zielsuche ausgewertet werden. Dazu ist es erforderlich, die Informationen in einem Gedächtnis unterzubringen. Im ersten Fall geht es darum, das zunächst unbekannte Labyrinth in einem bestimmten Ausmaß ins Gedächtnis abzubilden, während im zweiten Fall der Weg aus dem Gedächtnis in das Labyrinth projiziert wird.

4.2 Begriffsbestimmungen für das Labyrinth (Bild 2)

Das Labyrinth besteht aus Verzweigungspunkten, *Knoten* genannt, die untereinander durch *Verbindungsgänge* verknüpft sind. Es wird vorausgesetzt, daß zwei Knoten höchstens durch einen Gang verbunden sind.

$k = 1, 2, \ldots K$	sind die Nummern der Knoten;
$a_k = 1, 2, \ldots A_k$	sind die Nummern der Ausgänge, die vom k-ten Knoten wegführen;
v_{ij}	sind die Nummern der Verbindungsgänge im Labyrinth ($1 \leq i, j \leq K$);
V_K	ist die tatsächliche Zahl der Verbindungsgänge in einem Labyrinth mit K Knoten.

K kann als eine natürliche Zahl beliebig vorgegeben werden, A_k und V_K unterliegen den Beschränkungen

$$A_k \leq A_{\mathrm{MAX}} = K - 1$$

$$V_K = 0{,}5 \sum_{k=1}^{K} A_k \leq V_{\mathrm{MAX}} = 0{,}5\, K \cdot (K - 1).$$

Das Absuchen des Labyrinths beginnt im *Startknoten* und endet im *Zielknoten*.

Als *Lösungsweg* des Labyrinths gilt eine Aneinanderreihung von Verbindungsgängen zu einem direkten Weg vom Startknoten zum Zielknoten, der jeden Knoten höchstens einmal berührt.

00) *Freie Knoten* und *freie Verbindungsgänge* sind jene, die im Laufe des Suchvorgangs noch nicht betreten worden sind.

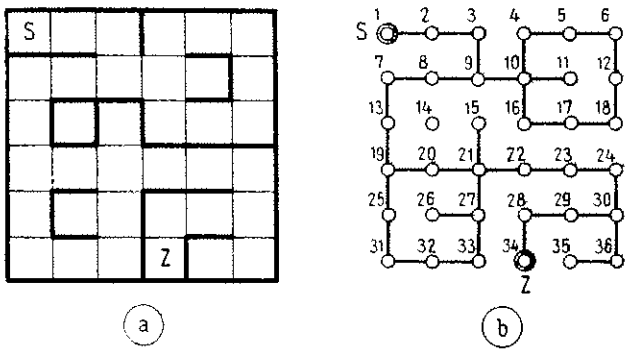


Bild 2. Begriffsbestimmungen für das Labyrinth.
a) zeigt ein Labyrinth, wie es im Modell gesteckt werden kann; b) ist die Netzwerkdarstellung davon. (Die Bilder 1a und 1b sind durch Umzeichnung aus diesen Bildern entstanden.)
Angegeben sind die Knotennummern $k = 1, 2, \ldots 36$ ($K = 36$) $k = 14$ ist ein isolierter Knoten ($A_{14} = 0$), der keine Bedeutnng für das Labyrinth hat. $k = 10$ hat die hier maximale Zahl von Ausgängen ($A_{10} = A_6 = 4$). Der Verbindungsgang $v_{25,\,27}$ ist eine Sackgasse. Die Verbindungsgänge zwischen den Knoten 4, 5, 6, 12, 18, 17, 16 und 10 bilden einen Kreisweg. $k = 1$ ist der Startknoten S und $k = 34$ ist der Zielknoten Z. $V_K = 36$.

01) *Lösungsstücke* sind Verbindungsgänge, die vorläufig oder endgültig dem Lösungsweg zugeordnet werden.

10) *Sackgassen* im engeren Sinn sind Gänge, die einseitig in einem Knoten mit nur einem Ausgang münden. Sackgassen im weiteren Sinn sind Komplexe aus Verbindungsgängen, die nicht zum Lösungsweg gehören.

11) *Kreiswege* sind Komplexe aus Verbindungsgängen, die einen in sich geschlossenen Weg bilden. Sie werden durch das Suchprogramm in Sackgassen und Lösungsstücke aufgespalten.

Der *Aufenthaltsknoten* ist der Knoten, in dem sich der Sucher eben befindet. Die *Nachbarknoten* sind jene Knoten, die mit dem Aufentbaltsknoten durch einen einzigen Verbindungsgang verknüpft sind.

Die *Orientierung* im Aufenthaltsknoten erfolgt durch die Auswahl eines Ausgangs als neue Fortschreitungsrichtung. Die Auswahl kann dabei auf einen freien oder auf einen einmal bereits durchlaufenen Verbindungsgang fallen.

4.3 Die Markierung durch den Ariadnefaden

Der Zweck des Ariadnefadens im Labyrinth ist die Markierung der Verbindungsgänge und Knoten. Die Markierung kann entweder für die Verbindungsgänge oder für die Knoten angegeben werden.

4.31 Verbindungsgänge

00) freier Verbindungsgang: kein Faden.

01) Lösungsstück: ein Faden.

10) Sackgasse: zwei Fäden.

11) Kreiswegkriterium: drei einfache Fäden würden sich treffen, weil ein freier Verbindungsgang zu einem markierten Knoten führt.

4.32 Knoten

00) freier Knoten: kein Ausgang ist durch einen Faden markiert.

01) der Knoten gehört zum Lösungsweg: in ihm sind zwei Richtungen durch einen einfachen Faden markiert.

10) der Knoten gehört einer Sackgasse an: alle Ausgänge sind mit zwei Fäden markiert.

11) ist ein dynamischer Fall und tritt nur im Aufenthalts-

knoten auf: drei Richtungen sind mit einfachen Fäden belegt, und es ist ein Kreisweg geschlossen.
Die beiden Markierungen 4.31 und 4.32 sind äquivalent. Für die Ermittlung der zur Orientierung erforderlichen Information soll die Darstellung 4.31 benützt werden; sie erlaubt nämlich, den Aufenthaltsknoten herauszuheben und ihn unabhängig von den Nachbarknoten zu beschreiben. Für die Bestimmung der Knoteninformation ist die Darstellung 4.32 vorgezogen worden, weil sie den Weg zu einer Verminderung der erforderlichen Speicherkapazität erleichtert.

4.4 Abschätzung der erforderlichen Information

Eingangs sei darauf hingewiesen, daß das Wort *Information* hier nicht im statistischen Sinn gebraucht wird, sondern in einem statischen oder logischen Sinn. Die Beschränkung des Ausdrucks Information auf den statistischen Fall mag für den Übertragungstheoretiker noch praktisch sein, für den Automatenbauer wäre sie eine unnütze Behinderung. Die verwendeten Maße beruhen auf einer Vorform der Informationstheorie [4], die Relationen zwischen Entscheidungen behandelt; der statistische Fall ist eine Weiterentwicklung der Vorform.
Da der Ariadnefaden für die Orientierung im Labyrinth ausreicht, kann die Abschätzung der erforderlichen Information von der Markierung durch den Ariadnefaden ausgehen. Es läßt sich aber zeigen, daß verschiedene Listen möglicher Fälle angesetzt werden können und daß auch die Verteilung der Information eine Rolle spielt.

4.41 Eine rohe Abschätzung

Die Listen 4.31 oder 4.32 umfassen vier Fälle, die mit zwei bit bezeichnet wurden. Da in jedem Knoten A_k Ausgänge zu betrachten sind, wäre die Orientierungsinformation je Knoten

$$EG_k = 2 \cdot A_k \quad \text{bit} \tag{1}$$

und die erforderliche Speicherkapazität wäre

$$EG_{\text{GES}} = \sum_{k=1}^{K} 2 \cdot A_k = 2 \sum_{k=1}^{K} A_k. \tag{2}$$

Diese Abschätzung berücksichtigt aber das Orientierungsprogramm nicht und liegt daher zu hoch. Es werden im folgenden zwei Orientierungsprogramme angegeben und unter ihrer Berücksichtigung andere Werte für EG_k und EG_{GES} bestimmt.

4.42 Ein Orientierungsprogramm

1. Der Sucher betritt den Knoten zum ersten Mal:

Ein einziger Ausweg ist als Rückweg zum Startknoten (einfacher Faden) zunächst blockiert, die übrigen Ausgänge sind frei und es kann einer von ihnen untersucht werden. – In Knoten mit nur einem Ausgang ergibt sich in trivialer Weise, daß sie auf dem Rückweg zum Startknoten verlassen werden müssen (Sackgasse).

2. Der Sucher betritt den Knoten noch einmal:

a) Der Sucher kommt aus einer Sackgasse zurück, wenn in diesem Knoten nur eine einzige Richtung durch einen einfachen Faden markiert ist (eindeutiger Rückweg zum Startknoten). Gibt es in diesem Knoten noch freie Ausgänge, dann werden diese zuerst untersucht, ansonsten wird der Rückweg zum vorhergehenden Knoten eingeschlagen.

b) Der Sucher hat einen Kreisweg geschlossen, wenn in dem Knoten drei Richtungen durch einfache Fäden markiert sind. Zwei Richtungen davon weisen einen Weg zurück zum Startknoten, die dritte führt in den Kreisweg. Da man nur von jener Richtung, aus der der Sucher zuletzt gekommen ist, mit Sicherheit sagen kann, daß sie zum Startknoten zurückführt, wird diese sofort eingeschlagen; und dies auch dann, wenn es in diesem Knoten noch freie Ausgänge gibt (der Sucher kommt ohnehin noch einmal dahin zurück).

4.43 Die Liste der möglichen Fälle

Das Programm schränkt die Zahl der möglichen Fälle im Knoten ein; wenn man alle möglichen Fälle erfaßt, kann man daraus die erforderliche Information bestimmen.

Es sind zwei Teillisten zu betrachten.

1. Im gegebenen Knoten ist nur eine einzige Richtung durch einen einfachen Faden markiert. Die Zahl der möglichen Verteilungen von freien Ausgängen, Rückweg und Sackgassen ist in diesem Knoten mit A_k Ausgängen (davon sind a_j freie Ausgänge, 1 Rückweg und $A_k - a_j - 1$ Sackgassen)

$$EV_{aj} = \frac{A_k!}{a_j! \cdot 1! \cdot (A_k - a_j - 1)!} = A_k \cdot \binom{A_k - 1}{a_j}. \tag{3}$$

Der Umfang der Teilliste ist gleich der Summe über die Verteilungsmöglichkeiten EV_{aj}

$$EV'_{\text{OR}} = \sum_{a_j=0}^{A_k-1} EV_{aj} = A_k \cdot \sum_{a_j=0}^{A_k-1} \binom{A_k - 1}{a_j} = A_k \cdot 2^{A_k-1}. \tag{4}$$

2. Im gegebenen Knoten sind drei Richtungen durch einen einfachen Faden markiert. In diesem Fall liegt ein Kreisweg vor. Das Orientierungsprogramm sieht hier keine Entscheidung über freie Ausgänge oder Sackgassen vor. Der Umfang dieser Liste ist daher

$$EV''_{\text{OR}} = 1. \tag{5}$$

Die Liste der möglichen Fälle umfaßt also einen Elementarvorrat von

$$EV_{\text{OR}} = EV'_{\text{OR}} + EV''_{\text{OR}} = 1 + A_k \cdot 2^{A_k-1}. \tag{6}$$

4.44 Die erforderliche Orientierungsinformation

Der Entscheidungsgehalt der Liste der möglichen Fälle ergibt sich zu

$$EG_{\text{OR}} = \text{ld}\,(1 + A_k \cdot 2^{A_k-1}), \tag{7}$$

und bezogen auf den Ausgang entsteht

$$EG_{\text{OR1}} = \frac{1}{A_k} \cdot \text{ld}\,(1 + A_k \cdot 2^{A_k-1}). \tag{8}$$

Da diese Formel etwas unhandlich ist, sei aus der Ungleichung

$$A \cdot 2^A \geqq 1 + A \cdot 2^{A-1} \geqq A \cdot 2^{A-1} \quad (\text{für } A \geqq 1)$$

durch Logarithmieren und Division durch A die Abschätzung

$$1 + \frac{\text{ld}\,A}{A} \geqq \frac{\text{ld}\,(1 + A \cdot 2^{A-1})}{A} \geqq 1 + \frac{\text{ld}\,(A - 1)}{A} \tag{9}$$

abgeleitet. Die Majorante $1 + \text{ld}\,A/A$ hat ein Maximum bei $A = e$ in der Größe von

$$1 + \frac{\text{ld}\,e}{e} < \text{ld}\,3$$

und somit gilt

$$EG_{\text{OR1}} < \text{ld}\,3.$$

Grundsätzlich könnte man also mit drei Markierungen je Ausgang auskommen; man nimmt dabei eine Entscheidungsredundanz von

$$ER = A_k \cdot \text{ld}\,3 - \text{ld}\,(1 + A_k \cdot 2^{A_k-1}) = \text{ld}\,\frac{3^{A_k}}{1 + A_k \cdot 2^{A_k}} - 1 \tag{10}$$

in Kauf. Für große A_k nimmt diese etwa proportional zu A_k zu und man müßte sich eine ausgeklügeltere Markierung zurechtlegen, gäbe es nicht ein wirkungsvolleres Verfahren, das im Abschnitt 4.5 gezeigt werden wird.
Aus Gl. (7) folgt die Kapazität des Gesamtspeichers

$$EG_{\text{GES}} = \sum_{k=1}^{K} EG_{\text{OR}} = \sum_{k=1}^{K} \text{ld}\,(1 + A_k \cdot 2^{A_k-1}) \tag{11}$$

und mit Hilfe der Majorante aus Gl. (9) kann man behaupten

$$EG_{\mathrm{GES}} = \sum_{k=1}^{K} (A_k + \mathrm{ld}\, A_k) = \sum_{k=1}^{K} A_k + \mathrm{ld} \prod_{k=1}^{K} A_k, \qquad (12)$$

was im Vergleich zu Gl. (2) diese mit einer Redundanz von

$$ER = \sum_{k=1}^{K} A_k - \mathrm{ld} \prod_{k=1}^{K} A_k \qquad (13)$$

behaftet erscheinen läßt. Um diese Größe ist das neue Ergebnis besser.

4.5 Betrachtung der Nachbarknoten

Bisher wurde die Orientierung so aufgefaßt, daß die Orientierungsinformation im Aufenthaltsknoten vollständig greifbar ist. Für jeden Knoten ist dabei eine *Knoteninformation* gleich der Orientierungsinformation erforderlich. Daher ergibt sich die erforderliche Kapazität des Gesamtspeichers mit Gl. (11).
In diesen Angaben verbirgt sich aber insofern Redundanz, als verbundene Knoten in der Verteilung ihrer Markierungen nicht unabhängig sind. Denn ein Verbindungsgang muß in beiden Knoten, die er verbindet, die gleiche Markierung haben.
Betrachtet man für die Orientierung nicht nur den Aufenthaltsknoten, sondern auch seine Nachbarknoten, so läßt sich nicht nur ein einfacheres Orientierungsprogramm angehen, sondern man kommt auch zu einem Gesamtspeicher noch kleinerer Kapazität.

4.51 Das vereinfachte Orientierungsprogramm

1. Der Weg zum Zielknoten führt in einen freien Nachbarknoten.

2. Ist ein solcher nicht vorhanden, so wird der Rückweg zum Startknoten (in einen markierten Nachbarknoten) eingeschlagen.
Da mehrere Nachbarknoten frei sein können, bleibt bei 1. entweder eine beliebige Auswahl offen oder es kann noch eine Regel für die Richtungsbevorzugung vorgeschrieben werden.
Der Rückweg zum Startknoten wird durch eine Information über den Ausgang, durch den der Aufenthaltsknoten betreten wurde, gesichert.

4.52 Die erforderliche Knoteninformation

Die Liste der möglichen Fälle ist auch beim vereinfachten Programm die gleiche. Daher ist auch die für die Orientierung erforderliche Information die gleiche. Sie kann aber teilweise oder vollständig von den Nachbarknoten bezogen werden; die Knoteninformation wird dadurch wesentlich geringer als die Orientierungsinformation, weil diese über den Aufenthaltsknoten und Nachbarknoten verteilt ist.
Das Orientierungsprogramm legt eine Unterteilung der Knoteninformation in zwei Arten nahe. Die 1-bit-Entscheidung „freier Knoten (0) oder markierter Knoten (1)" ist richtungsunabhängig; sie sei als *Markierungsinformation* EG_{Mk} bezeichnet. Sie wird vom Aufenthaltsknoten aus in den Nachbarknoten abgefragt. Die zweite Art gibt die Richtung an, aus der der Sucher gekommen ist und wohin notfalls der Rückweg gehen muß. Diese Richtungsinformation EG_{Rk}, eine Entscheidung zwischen A_k Möglichkeiten, braucht nur im Aufenthaltsknoten selbst abgefragt zu werden. Man hat also

$$EG_{\mathrm{Mk}} = 1 \text{ bit} \qquad EG_{\mathrm{Rk}} = \mathrm{ld}\, A_k \text{ bit}.$$

Während die Orientierungsinformation nach wie vor

$$EG_{\mathrm{OI}} = A_k - \mathrm{ld}\, A_k \qquad (14)$$

beträgt, setzt sich die Knoteninformation aus der Markierungsinformation und aus der Richtungsinformation zusammen und hat den Wert

$$EG_k = EG_{\mathrm{Mk}} + EG_{\mathrm{Rk}} = 1 + \mathrm{ld}\, A_k. \qquad (15)$$

Der Gesamtspeicher kommt demgemäß mit einer Kapazität von

$$EG_{\mathrm{GES}} = \Sigma\, EG_k = \sum_{k=1}^{K} (1 + \mathrm{ld}\, A_k) = K + \mathrm{ld} \prod_{k=1}^{K} A_k \qquad (16)$$

aus.

Die im Abschnitt 4.4 behandelte Methode hat im Vergleich dazu also eine Redundanz, die sich aus der Differenz der durch die Gl. (12) und (16) bestimmten Beiträge errechnet:

$$ER = \sum_{k=1}^{K} A_k + \mathrm{ld} \prod_{k=1}^{K} A_k - K - \mathrm{ld} \prod_{k=1}^{K} A_k = \sum_{k=1}^{K} A_k - K. \qquad (17)$$

4.6 Wegwiederholung

Markierungsinformation und Richtungsinformation zusammen erlauben und sichern die Zielsuche. Die Richtungsinformation allein gibt den Rückweg an. Hingegen bleiben für die Wegwiederholung vom Startknoten zum Zielknoten noch Entscheidungen offen (Bild 3). Aus den Knoteninformationen des Aufenthaltsknotens und seiner Nachbarknoten folgt nicht immer, welcher der Ausgänge zum Zielknoten führt.

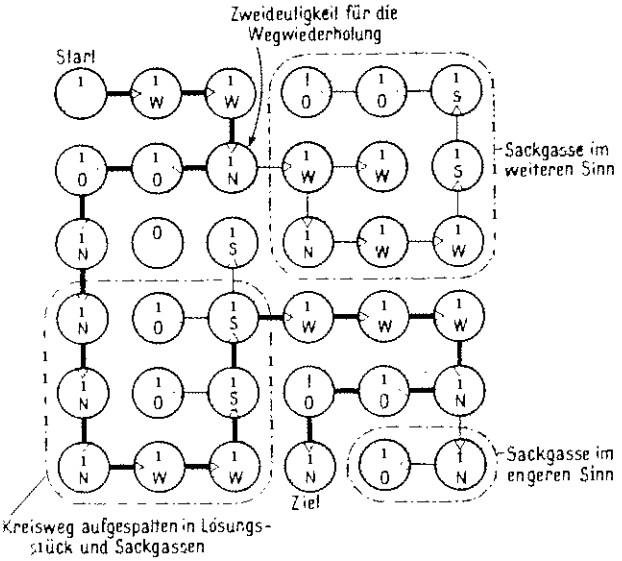


Bild 3. Die Knoteninformationen nach der Zielsuche nach dem Verfahren 4.52.

Markierungsinformation: 1 besuchter Knoten, 0 freier Knoten.
Richtungsinformation: Die Himmelsrichtung zeigt an, woher der Knoten zum ersten Mal betreten wurde.
Der Rückweg ist gesichert, die Wegwiederholung kann Zweideutigkeiten aufweisen.

Dieser Nachteil läßt sich beheben, indem Sackgassen als solche markiert werden; das vereinfachte Markierungsprogramm hat das Prinzip des Ariadnefadens insofern verlassen, als die Sackgassenmarkierung durch zwei Fäden nicht mehr von der Lösungswegmarkierung durch einen Faden unterschieden wurde. Die Vereinfachung erlaubte zwar die Lösung des Labyrinths und legte den Rückweg eindeutig fest; für die Wegwiederholung braucht man jedoch eine Markierungsinformation, die zwischen drei Fällen unterscheidet:

00) freier Knoten,
01) Knoten des Lösungswegs,
10) Knoten einer Sackgasse (im engeren und weiteren Sinn).

Mit der erhöhten Markierungsinformation von

$$EG'_{\mathrm{Mk}} = \mathrm{ld}\, 3 \text{ bit}$$

lassen sich alle Forderungen der Aufgabenstellung erfüllen

Die Ergebnisse (15) bis (17) verändern sich dann ein wenig:

$$EG_k = \mathrm{ld}\,3 + \mathrm{ld}\,A_k \tag{15'}$$

$$EG_{\mathrm{GES}} = K \cdot \mathrm{ld}\,3 + \mathrm{ld} \prod_{k=1}^{K} A_k \tag{16'}$$

$$ER = \sum_{k=1}^{K} A_k - K \cdot \mathrm{ld}\,3\,. \tag{17'}$$

4.7 Eine gleichwertige Variante

Eine gleichwertige Variante entsteht, wenn im Richtungsspeicher der Lösungsweg festgehalten wird, d. h. wenn dort nicht jene Richtung gespeichert wird, aus der der Sucher den Knoten zum ersten Mal betreten hat, sondern die Richtung, in der der Sucher den Knoten zum letzten Mal verlassen hat (Bild 4).

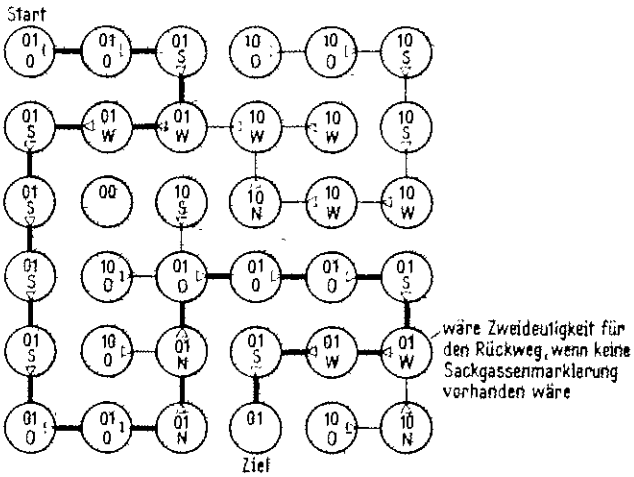


Bild 4. Die Knoteninformation nach der Zielsuche nach dem Verfahren 4.7.

Markierungsinformation: 00 freier Knoten, 01 Knoten auf dem Lösungsweg, 10 Knoten einer Sackgasse.
Richtungsinformation: Die Himmelsrichtung zeigt an, wohin der Knoten zum letzten Mal verlassen wurde.
Wegwiederholung und Rückweg zum Eingang sind gesichert.

Unterscheidet die Markierungsinformation nur zwei Fälle, so ist bei dieser Variante die Zielsuche nicht mehr gewährleistet, weil dafür die Eindeutigkeit des Rückwegs erforderlich ist. Die Markierungsinformation muß daher drei Fälle unterscheiden, es gelten die Formeln des Abschnitts 4.6, und es sind dann Zielsuche, Wegwiederholung und Rückweg gesichert; da die Wegwiederholung hier leichter fällt, weil die Information des Richtungsspeichers dafür ausreicht, liegt diese Variante der Organisation des organischen Erinnerns wahrscheinlich näher. Die einzige Umständlichkeit dieser Variante liegt bei der Bestimmung des Rückwegs während der Zielsuche, wenn eben ein Kreisweg festgestellt wurde. Dann sind nämlich zwei Nachbarknoten mit 01 markiert, und der Rückweg muß auf Grund der Richtungsinformation dieser beiden Nachbarknoten bestimmt werden; nur der zuletzt besuchte Nachbarknoten weist auf den Aufenthaltsknoten hin, und dorthin führt der Rückweg.

4.8 Wegwiederholung nach geringen Veränderungen im Labyrinth

Veränderungen im Labyrinth, die den Lösungsweg nicht unterbrechen, sind für die Wegwiederholung unbedeutend. Wird durch die Veränderung hingegen der Lösungsweg unterbrochen, so muß nach dem neuen Lösungsweg gesucht werden. Die Markierungsinformation, die im Fall der Variante 4.7 für die Wegwiederholung ohnehin überflüssig ist, wird dann zur Behinderung; sie muß für das veränderte Labyrinth nicht mehr zutreffen. Man kann nun die Wirksamkeit des Orientierungsprogramms auf die Wegwiederholung bei kleinen Veränderungen ausdehnen, wenn man auf die Sicherung des Lösungsweges vom Zielknoten zum Startknoten verzichtet. Man nimmt die Variante 4.7 und löscht nach dem ersten Erreichen des Zieles die Markierungsinformation; die Wegwiederholung ist ja auf Grund der Richtungsinformation möglich. Im ungestörten Teil des Labyrinths läuft sie normal ab, es werden aber die durchlaufenen Knoten mit 01 markiert. An der Unterbrechungsstelle setzt von selbst das Orientierungsprogramm wieder ein. Nimmt man für die Richtungsbevorzugung im Orientierungsprogramm (Punkt 1. im Abschnitt 4.51) die Regel hinzu, daß die im Richtungsspeicher festgehaltene Richtung zuerst untersucht wird, wenn sie dem Programm nicht widerspricht, so werden unveränderte Teile des früheren Lösungsweges benützt, sobald der Sucher auf sie stößt.

4.9 Kein Ziel

Wenn im Labyrinth kein Ziel vorhanden ist, so führt das Programm dazu, daß der Sucher zum Startknoten zurückkehrt und diesen als Sackgasse markiert, wodurch das ganze Labyrinth zu einer Sackgasse im weiteren Sinn geworden ist.

Diese Tatsache läßt eine Abschätzung der Suchdauer zu. Das Labyrinth ohne Ziel ist nämlich eine Variante nicht weit vom ungünstigsten Fall: der Sucher könnte das Ziel nach fast vollständiger Durchsuche in einem Nachbarknoten des Startknotens finden. Mit Ausnahme jener Verbindungsgänge, die Kreiswege schließen würden (Fall 11 im Abschnitt 4.31), sind dann alle Verbindungsgänge als Sackgassen (Fall 10 im Abschnitt 4.31) markiert. Wenn also dem Verbindungsgang v_{ij} eine Durchlaufzeit t_{ij} zugeordnet ist, muß die gesamte Suchdauer T der Ungleichung

$$T \leqq 2 \sum_{i,j} t_{ij} \tag{18}$$

genügen.

5. Aufbau und Abbau der Information in den Speichern

Zu den grundsätzlichen Überlegungen des Abschnitts 4 müssen für die praktische Ausführung einer automatischen Zielsuche noch einige weitere Punkte untersucht werden. Die Gesamtaufgabe wurde als Abbildung des zunächst noch unbekannten Labyrinths in das Gedächtnis und dann des im Gedächtnis aufgezeichneten Lösungsweges zurück in das Labyrinth charakterisiert. Dazu ist ein Datenfluß erforderlich, der durch Bild 5 wiedergegeben werden kann. Das Programm liest aus dem Speicher die Knoteninformationen

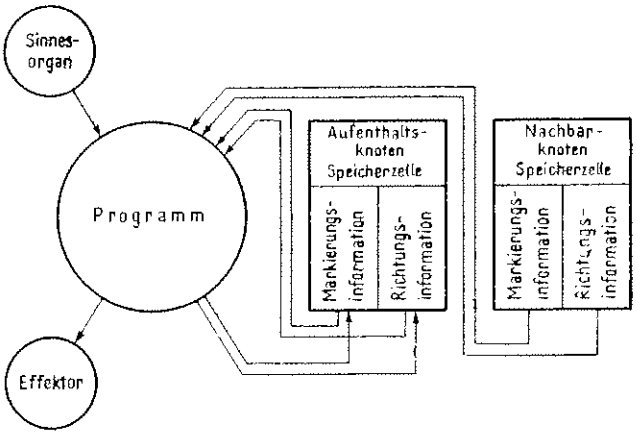


Bild 5. Der Datenfluß der Programmsteuerung.
Das Programm erhält Daten vom Sinnesorgan des Suchers und aus den Speicherzellen des Aufenthaltsknotens und der Nachbarknoten. Daraus leitet es Befehle für den Effektor des Suchers ab und Orientierungsinformation, die im Aufenthaltsspeicher aufgezeichnet wird.

ab und erhält auch vom Sucher Entscheidungen; es speichert neue Information ab und führt dem Sucher Befehle für seine Bewegungen zu.
Ganz unbekannt darf das Labyrinth freilich nicht sein, mindestens müssen gewisse Annahmen gerechtfertigt sein. Die Zahl der Knoten K zum Beispiel darf nicht größer sein als die Zahl der vorbereiteten Gedächtniszellen. Die Zahl der Ausgänge A_k im Knoten k darf nicht größer sein als ein gewisses, vorbereitetes A_0; im allgemeinen Fall wird man $A_0 = K$ ansetzen müssen: jeder Knoten kann mit jedem verbunden sein. Die Richtung, die der Sucher einschlägt, muß auf das Gedächtnis abbildbar sein; diese Abbildung braucht nicht analogen Charakter zu haben, sie kann verschlüsselt erfolgen, aber ohne Zuordnung dieser Art ist die Orientierung unmöglich.

5.1 Der Sucher

Der automatisierte Theseus legt den Ariadnefaden nicht ins Labyrinth, sondern ins Gedächtnis; er sieht auch dort nach, wenn er die „Fadenbelegung" eines Verbindungsgangs feststellen will. Er ist also über die automatische Anlage „verteilt" und keineswegs durch den Sucher wiedergegeben. Dieser faßt lediglich zusammen, was an Sinnesorganen und Effektoren äußerer Art übergeblieben ist.

5.11 Das Sinnesorgan

Das Sinnesorgan ist wie immer bei einem Automaten auf die einfachste Form reduziert. Benützt wird der Tastsinn und gemeldet wird eine Entscheidung zwischen vier Möglichkeiten:

1. Begehbarer Verbindungsgang,
2. Knoten erreicht,
3. Ziel erreicht,
4. Versperrter, d.h. nicht vorhandener Verbindungsgang.

Darin steckt die Voraussetzung, daß diese Information dem Sinnesorgan zugänglich gemacht ist; es wird also zum Beispiel der Knoten ein abtastbares Merkmal haben müssen.

5.12 Der Effektor

Auch der Effektor, der nur auf Bewegung eingerichtet ist, hat die einfachste Form: der Sucher bewegt sich nach vorwärts oder er wartet im Knoten auf die Entscheidung für die Bewegungsrichtung. Diese fällt zwischen A_0 Möglichkeiten.

5.2 Der Speicher

Im Speicher sind die Knoteninformationen untergebracht. Jedem Knoten ist eine Zelle zugeordnet, in welcher Markierungs- und Richtungsinformation Platz finden können. Für die Speicherkapazität und Redundanz ergeben sich unter Berücksichtigung der Tatsache, daß die Werte A_k vorerst unbekannt sind und alle durch A_0 ersetzt werden müssen, neue Formen der Gl. (16') und (17'):

$$EG_{\text{GES}} = K \cdot \text{ld}\, 3\, A_0 \qquad (16'')$$

$$ER = K \cdot \text{ld}\, 3\, A_0 - K - \text{ld} \prod_{k=1}^{K} A_k. \qquad (17'')$$

5.3 Das Programm

Dem Programm stehen als Eingangsinformationen die Entscheidung des Suchers, der Inhalt der Aufenthaltsknoten-Speicherzelle und die Inhalte der Nachbarknoten-Speicherzellen zur Verfügung. Da in diesen Angaben weder am Beginn noch später etwas über die Struktur des untersuchten Labyrinths enthalten ist, muß der Sucher immer erst herausfinden, ob eine gewählte Richtung einen Verbindungsgang darstellt oder nicht. Damit kann sich das Programm aber nicht an die Definition des Nachbarknotens halten; dazu müßte es erfahren, ob der Richtung ein Verbindungsgang entspricht oder nicht. Es sieht also alle Knoten als Nachbarknoten an ($A_0 = K$), wenn nicht weitere Einschränkungen des Labyrinths bekannt und im Speicher berücksichtigt sind. Für die Ausführung des Programms ist der entstehende Schaden nicht sehr groß; bei der Zielsuche wird der Sucher zwar manchmal gegen gesperrte (nicht bestehende) Verbindungen stoßen, sobald aber einmal die Richtungsinformation aufgebaut ist, tritt dies nicht mehr auf.

Nach den unter 4.51, 4.7 und 4.8 gegebenen Regeln bestimmt das Programm aus den Eingangsinformationen die Richtung für den Effektor des Suchers und speichert in der Aufenthaltsknoten-Speicherzelle Markierung und Richtung ab.

5.31 Zielsuche

Der Sucher wird auf den Startknoten gebracht. Alle Speicherzellen sind gelöscht oder zeigen einheitlich eine bestimmte, aber bedeutungslose Richtung an. Am Ende des Durchlaufs ist nach Variante 4.7 der Weg vom Start zum Ziel gespeichert.

5.32 Wegwiederholung

Der Sucher wird auf den Startknoten zurückgebracht. Da der Lösungsweg gespeichert ist, findet er auf kürzestem Weg wieder zum Ziel zurück, ohne irgendeine Sackgasse zu betreten oder gar eine gesperrte Richtung zu versuchen.

5.33 Wegwiederholung nach geringen Veränderungen im Labyrinth

Zum Start zurückgebracht, läuft der Sucher den ununterbrochenen Teil des Lösungsweges durch, bis er gegen eine hineingesetzte Trennwand stößt. Dann erfolgt die Neuorientierung wie unter 4.8 beschrieben, und wenn der Sucher zum Ziel gefunden hat, ist der neue Lösungsweg in den Richtungsspeichern.

5.4 Das Orientierungsprogramm als Lernprozeß

Obwohl in den bisherigen Ausführungen die Terminologie des „Lernens" nicht verwendet wurde, stellt das Orientierungsprogramm einen typischen Lernprozeß dar. Es arbeitet sich schrittweise mit einem „trial-and-error"-Verfahren zu einer Lösung durch, die verwendbar bleibt, und sogar bei Veränderungen wird von den gewonnenen Erfahrungen optimaler Gebrauch gemacht.

6. Ein ausgeführtes Modell

Für Demonstrationszwecke ist am Wiener Institut für Niederfrequenztechnik ein Modell ausgeführt worden, das in seiner Funktion auf dem oben beschriebenen Verfahren beruht und dessen äußere Erscheinung dem Modell von *Shannon* [2] nachgebildet ist.
Das Labyrinth wird auf einer Platte in einem Feld bestehend aus sechs mal sechs Quadraten durch Aufstellen von Trennwänden an Quadratseiten abgesteckt. Jedes dieser Quadrate entspricht einem Knoten, dem bis zu vier Ausgänge eingeräumt werden können, so daß innerhalb dieser 36 Quadrate eine große Anzahl verschiedener Labyrinthe absteckbar ist. Die Richtungen der möglichen Ausgänge können nach den vier Himmelsrichtungen mit N, O, S und W bezeichnet werden. Da ein Quadrat nur mit vier andern direkt verbunden sein kann, ergibt sich eine spezielle Definition des Nachbarknotens, die auch das Programm vereinfacht.
Die Struktur der Speicherzellen stimmt in den Grundzügen mit der Beschreibung in Abschnitt 4.7 überein. Den drei Möglichkeiten der Markierungsinformation ist beim Modell die Markierung des Aufenthaltsknotens hinzugefügt worden, so daß sich zwei voll ausgenützte Speicherbits ergeben.
Der Sucher im Labyrinth ist ein Magnet auf Glaskugeln, der von einem Führungsmagneten unter der Feldebene durch Anziehung nachgeschleppt wird. Die vier Möglichkeiten für die Anzeige des Sinnesorgans werden durch zwei Kontakte erfaßt; der eine stellt fest, ob sich die beiden Magneten anziehen oder nicht, und der andere, ob sich der Sucher auf einem Knoten befindet oder nicht:

		Magnete ziehen sich an	
		ja (1)	nein (0)
am Anfang		Sucher gefunden	Sucher nicht gefunden
(1) in einem Knoten (0)	ja	2. Knoten erreicht	3. Ziel erreicht
	nein	1. Begehbarer Verbindungsgang	4. Versperrter Verbindungsgang

Am Anfang kann der Sucher automatisch im Labyrinth gefunden werden, indem der Führungsmagnet der Reihe nach auf alle Knoten kommt. Werden die beiden Magneten später getrennt, so ist der Sucher entweder gegen eine Trennwand oder gegen das Ziel gestoßen; das Ziel aber befindet sich auf dem Knoten, die Trennwand hingegen in der Mitte zwischen zwei Knoten.

Der Effektor ist der Führungsmagnet; er wird von einem Nord-Süd-Motor und von einem Ost-West-Motor angetrieben und bewegt sich daher in einem rechtwinkeligen Koordinatensystem. Auf seinen Laufschienen sitzt der Kontakt, der den Knoten anzeigt.

Sobald der Führungsmagnet den Sucher gefunden hat, setzt die Zielsuche ein. Das Programm ist nach den Prinzipien des Abschnitts 4.8 gehalten, kann jedoch auch von außen beeinflußt werden. Ein Schalter bestimmt, ob das Absuchen der freien Richtungen im oder gegen den Uhrzeigersinn erfolgt. Ein weiterer Schalter stellt von der automatischen Richtungswahl auf manuelle Auswahl um; es sind vier Drucktasten für die vier Richtungen vorhanden, es werden aber nur solche Befehle ausgeführt, die den Richtlinien des Programms nach 4.51 nicht widersprechen.

Um die Situation des Suchers zu charakterisieren, wird ein Teil der Orientierungsinformation sichtbar gemacht. Vier Lämpchen zeigen an, welche Nachbarknoten unmarkiert sind; das ist jener Teil der Markierungsinformation der Nachbarknoten, der für die manuelle Richtungswahl behilflich ist. Vier weitere Lämpchen geben die Richtungsinformation des Aufenthaltsknotens wieder.

Ein Lämpchen signalisiert, daß sich der Sucher im Zielknoten befindet. Ein anderes leuchtet auf, wenn kein Ziel gesteckt ist oder sich der Rückweg zum Startknoten, nachdem kein Ziel gefunden wurde, als nachträglich versperrt erweist; in diesen beiden Fällen kann nach Öffnen eines Ausgangs in einen freien Nachbarknoten das Programm wieder gestartet werden.

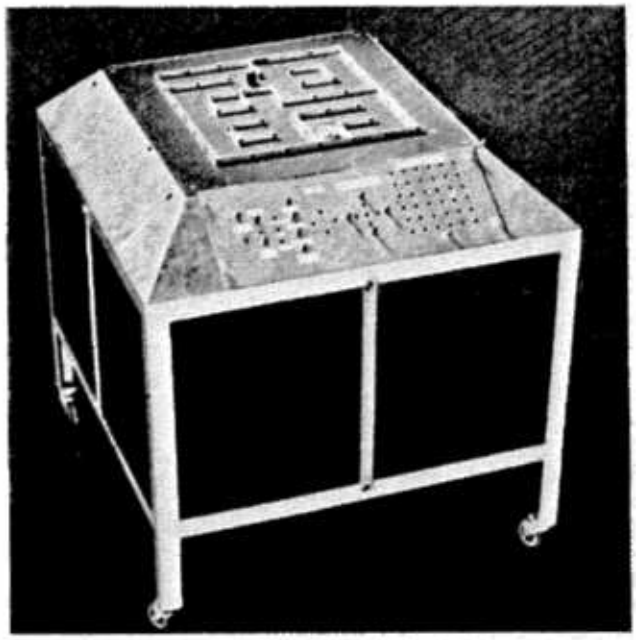
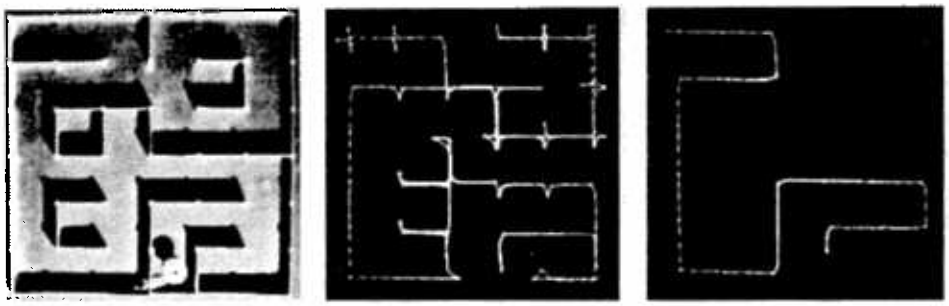


Bild 6. Das Modell.
Es ist 1 m mal 1 m mal 1 m groß und enthält 2 Drehwähler und 284 Relais. Die Trennwände sind nach Belieben steckbar. Man erkennt den etwas höheren Sucher und das Ziel. Am Bedienungspult wird der Lösungsweg angezeigt.

Durch das Ausschalten des Programms wird die Markierungsinformation im Speicher gelöscht, die Richtungsinformation hingegen bleibt gespeichert, damit sie für die Wegwiederholung zur Verfügung steht; sie wird erst durch den Hauptschalter gelöscht.

Übrigens ist durch eine Kunstschaltung die Zuordnung zwischen Knoten und Speicherzelle nicht fest, sondern kann vor dem Starten des Programms beliebig verschoben werden. Der Lösungsweg wird in einer Anordnung von sechs mal sechs Lämpchen angezeigt, aus der auch die Verschiebung entnommen werden kann.

In dem ausgeführten Modell wurden zwei Drehwähler und 284 Relais verwendet, von denen rund 80 hätten erspart werden können, wenn wir nicht auf vorhandenes Material angewiesen gewesen wären, das wir der österreichischen Generalpostdirektion verdanken. Bild 6 ist ein Photo des etwa 1 m³ großen Modells; Bild 7 veranschaulicht den Suchvorgang und die Wegwiederholung.



a b c

Bild 7. Labyrinth, Zielsuche und Wegwiederholung.

a) Das Labyrinth mit dem Ziel; der Sucher ist noch nicht aufgestellt.

b) Der Sucher, mit einem Lichtzeiger versehen, hat auf dem Photo Bewegung und Aufenthalte aufgezeichnet. Die Zielsuche dauerte etwa 3 min.

c) Bei der Wegwiederholung läuft der Sucher auf kürzestem Weg durch, ohne an eine Trennwand zu stoßen. Jetzt dauert der Vorgang nur etwa 1 min.

7. Vergleich mit dem Modell von Shannon

Für den Vergleich mit dem Modell von *Shannon* war eine Analyse seines Verfahrens erforderlich. Das Protokoll der 5. Kybernetik-Tagung der Macy-Foundation [2] ist nicht für einen solchen Zweck gedacht; unsere Ergebnisse sind daher mit einer unvermeidlichen Unsicherheit behaftet.

7.1 *Die beiden Programme von Shannon*

In den Knotenspeichern wird festgehalten, in welcher Richtung der Knoten zuletzt verlassen wurde. Ist das Gerät neu eingeschaltet, so haben alle Knoten die gleiche, aber bedeutungslose Richtung aufgezeichnet.

7.11 Wegsuche

Es wird nicht die gespeicherte, sondern eine um 90 Grad veränderte Richtung als erste versucht (zur Zeitverkürzung erwies es sich als praktisch, die Eingangsrichtung zunächst einmal zu überspringen).

7.12 Wegwiederholung

Die gespeicherte Richtung wird als erste versucht; wenn nach 25 Schritten das Ziel nicht erreicht ist, ist dieses entweder entfernt worden oder der Sucher ist in einen Kreisweg geraten. Das Modell schaltet auf die Wegsuche zurück. Dann bricht der Sucher mit Sicherheit aus dem Kreisweg aus; wenn kein Ziel vorhanden ist, so werden alle Knoten auf einem periodischen Weg immer wieder betreten.

7.2 *Vergleich*

Da *Shannon* nicht vom Konzept des Ariadnefadens aus (es hat bloß die Maus den Spitznamen Theseus), verwe er keine Markierungsinformation; er braucht also we Speicherraum, muß dafür aber zwei einander ausschlie de Programme verwenden.

Die Markierungsinformation unseres Konzepts erlaub Erkennung von Kreiswegen und ihre Aufspaltung in S gassen und Lösungsstücke. Dadurch läßt sich die maxi Suchzeit abschätzen. Auch folgt aus der Markierung Startknotens als Sackgasse eindeutig, daß kein Ziel gre ist.

Der Freiheitsgrad im Programm unseres Modells (schließlich kann zur Berücksichtigung eines übergeordn Konzepts herangezogen werden.

Literatur

[1] *H. Hunger:* Lexikon der griechischen und römischen M logie. – Verlag Hollinek, Wien 1953.

[2] *C. E. Shannon:* Presentation of a Maze Solving Machir Trans. Eighth Conference 1951 on Cybernetics, p. 173– J. Macy Foundation, N.Y. 1952.

[3] *J. A. Deutsch:* The Insightful Learning Machine. – covery 16 [1955], No. 12, p. 514–517.

[4] *H. Zemanek:* Elementare Informationstheorie. – V R. Oldenbourg, Wien und München 1959.