ÜBER DIE »KÜRZESTE« FORM VON BOOLESCHEN FUNKTIONEN

von

RÓZSA PÉTER

1. In der Theorie der Synthese von Stromkreisen auf Grund von gegebenen Arbeitsbedingungen spielen Boolesche Funtionen eine wichtige Rolle, d. h. Funktionen, die selber, und auch ihre Variablen, nur zwei Werte (wie »wahr« und »falsch«) annehmen können. Bekanntlich können diese Funktionen mit Hilfe von Konjunktionen, Disjunktionen und Negationen von den Variablen aufgebaut werden (sogar mit wenigeren Operationen, aber dann weniger übersichtlich); man denke z. B. an ihre eindeutige Darstellbarkeit in einer ausgezeichneten disjunktiven Normalform. L. KALMÁR gab mir die Anregung zur Untersuchung der — aus technisch-ökonomischen Gründen wichtigen — kürzestmöglichen Darstellungen der Booleschen Funktionen mit Hilfe der genannten drei Operationen; wobei unter der Länge einer Formel die Anzahl der darin auftretenden Variablen verstanden wird, jede sovielmal gerechnet, wie oft sie in dem Ausdruck vorkommt. Freilich benötigt man einen möglichst einfachen Algorithmus zur Gewinnung der kürzestmöglichen Darstellung von beliebig gegebenen Booleschen Funktionen; als erster Schritt in dieser Richtung ist es daher interessant zunächst theoretisch zu untersuchen, wasfür (inwiefern rekursive) Algorithmen hier zu erwarten sind.

In vorliegender Arbeit beweise ich, dass *sowohl die minimale Länge der Formeln, die eine gegebene Boolesche Funktion F von n Variablen darstellen, als auch eine bestimmte solche Formel von minimaler Länge, in einer passenden Wortemenge als primitiv-rekursive Funktionen von F und n definiert werden können.*

2. Betreffs der Kentnisse über zahlentheoretische rekursive Funktionen berufe ich mich auf mein Buch[1], und betreffs der Kenntnisse über Wortemengen und in Wortemengen rekursive Funktionen auf meine in dieser Zeitschrift früher erscheinende Arbeit[2], worin auch die ursprüngliche Quelle[3] dieser Untersuchungen zitiert wird, und ohne Beweis all das aufgezählt wird, was davon auch in dieser Arbeit benutzt wird, mit einigen Ausnahmen, die ich hier angebe:

Auch die folgenden Funktionen sind primitiv-rekursiv in einer Wortemenge:

1) $$b_i(x)\,,$$

das falls i eine natürliche Zahl ist, den i-ten Buchstaben des Wortes x bedeutet;

---

[1] R. PÉTER, Rekursive Funktionen, Budapest, Akademischer Verlag, 2te Auflage (1957)

[2] R. PÉTER, Über die Rekursivität einiger Übersetzungs-Transformationen (I. Mitteilung), Nr. 3.

[3] R. PÉTER, Über die Verallgemeinerung der Theorie der rekursiven Funktionen für abstrakte Mengen geeigneter Struktur als Definitionsbereiche, *Acta Math. Acad. Sci. Hungaricae* (eingegangen am 30ten September 1959).

2) die Anzahl

$$h^{(b)}(x)$$

des Auftretens des Buchstaben b im Wort x;

3) $\operatorname{subst}(x, y, z)$ und $\operatorname{sub}(x, y, z)$,

welche beide für $y \neq A$ das aus x dadurch entstehende Wort bezeichnen, dass in x jedes Vorkommen von y durch z ersetzt wird (von rechts nach links bzw. umgekehrt, was aber in dieser Arbeit ohne belang ist); bei subst (x, y, z) vorausgesetzt, dass die Ersetzung eines solchen y durch z kein neues Vorkommen von y in x zustandebringt; und bei sub (x, y, z) vorausgesetzt, dass $o(z) < o(y)$ ist.

Endlich werde ich noch benutzen, dass die »eingeschachtelte Rekursion«, worin für die Parameter Einsetzungen erfolgen, nicht von der Klasse der primitiv-rekursiven Funktionen hinausführt.

3. Eine Boolesche Funktion von n Variablen wird durch ihre Wertetabelle angegeben; daraus lässt sich aber ihre ausgezeichnete disjunktive Normalform unmittelbar entnehmen, und auch umgekehrt liest man aus einer solchen Normalform unmittelbar die Wertetabelle heraus; z. B. sicht man unmittelbar, dass die Wertetabelle

A_1	A_2	F
wahr	wahr	wahr
wahr	falsch	falsch
falsch	wahr	falsch
falsch	falsch	wahr

und die ausgezeichnete disjunktive Normalform

$$F = A_1 A_2 \vee \bar{A}_1 \bar{A}_2$$

dieselbe Funktion F darstellen (wobei das Nacheinandersetzen von Formeln ihre Konjunktion bedeutet). In den Folgenden werde ich die Booleschen Funktionen durch ihre ausgezeichnete disjunktive Normalform angeben.

Ferner werde ich mich in den Folgenden auf Formeln beschränken, in welchen keine Negation einer mehrgliedrigen Formel als Teilformel vorkommt. Die Auflösung einer mehrgliedrigen Negation mit Benutzung einer de-Morgan-Identität ändert ja nichts an der Länge der betreffenden Formel, und in diesen Untersuchungen kommt es nur darauf an.

So bietet sich als passende Wortemenge die Wortemenge $\mathfrak{M}$ mit dem Alphabet

$$\mathfrak{A} = \{\mathbf{(}, \mathbf{)}, \mathbf{V}, A_1, \bar{A}_1, A_2, \bar{A}_2, \ldots\}.$$

Zur Unterscheidung von den üblich gebrauchten Klammern- und Disjunktionszeichen werden diese als Buchstaben des Alphabets (das letztere als $\mathbf{V}$) fett gedruckt. Die natürlichen Zahlen

$$0, \quad 1, \quad 2, \quad 3, \ldots$$

können etwa mit

$$\Lambda, \quad (, \quad ((, \quad (((, \ldots$$

identifiziert werden.

Mit der Konvention, dass die Konjunktion »stärker bindet« als die Disjunktion, können Klammern erspart werden : zur Konjunktion »geschlossener« Formeln brauchen die Konjunktionsglieder nicht in Klammern gesetzt werden; dabei bedeutet die »Geschlossenheit« einer Formel, dass darin entweder kein **V**-Zeichen vorkommt, oder es gibt links von jedem ihrer **V**-Zeichen mehr **(**-Zeichen als **)**-Zeichen (so muss zu mindestens einer der links vom betreffenden **V** stehenden Anfangsklammern eine rechts von diesem **V** stehende Endklammer gehören; in einer Formel muss ja die Anzahl der Endklammern mit der Anzahl der Anfangsklammern übereinstimmen). Die Konjunktion geschlossener Formeln ist wieder eine geschlossene Formel.

So ist jede Formel die Disjunktion von eindeutig bestimmten geschlossenen Formeln, »Disjunktionsglieder« genannt (falls sie geschlossen ist, ist sie eine eingliedrige Disjunktion). Eine geschlossene Formel x enthält entweder keine **(**-Zeichen (und dann auch keine **V**-Zeichen: sie ist dann eine »reine Konjunktion« von unnegierten und negierten Variablen), oder kann sie in der Form

$$x = z_1 (y_1 \mathbf{V} y_2) z_2$$

geschrieben werden, wo das ausgeschriebene **(**-Zeichen das von links erste **(**-Zeichen in x (und so z_1 eine reine Konjunktion) ist, und das ausgeschriebene **)**-Zeichen sein Klammernpaar (d.h. vom ausgeschriebenen **(** nach rechts gehend sind vor ihm immer mehr **(**-Zeichen als **)**-Zeichen, und wenn man das ausgeschriebene **)** erreicht, genau so viele; wonach z_2 eine geschlossene Formel ist), ferner y_1 eine geschlossene Formel ist.

§ 1.

4. Nun erweisen sich leicht die charakteristischen Funktionen (welche die Werte Λ, **(** annehmen) der Eigenschaften:

»**(** zu sein«, «**)** zu sein», »**V** zu sein«, »ein A_i zu sein«, »ein $\bar{A}_i$ zu sein«,

die der Reihe nach mit

$$f_((x), \; f_)(x), \; f_\mathbf{V}(x), \; f_A(x), \; f_{\bar{A}}(x)$$

bezeichnet werden, als primitiv-rekursiv in $\mathfrak{M}$. Z.B. lautet die Definition von $f_((x)$:

$$f_((\Lambda) = ($$

und für $a \in \mathfrak{A}$

$$f_((xa) = \begin{cases} \Lambda, & \text{falls} \quad x = \Lambda \;\&\; a = (\\ (& \text{sonst}; \end{cases}$$

und von $f_A(x)$:

$$f_A(\Lambda) = ($$

$$f_A(xA_i) = \begin{cases} \Lambda, & \text{falls} \quad x = \Lambda \\ (& \text{sonst} \end{cases}$$

$$f_A(x\,() = f_A(x\,)) = f_A(x\,\mathbf{V}) = f_A(x\,\bar{A}_i) = (\,.$$

5. Zur Definition der charakteristischen Funktion $f_F(x)$ der Eigenschaft »Formel zu sein« benötigt man die Hilfsfunktion x^*, welche für eine geschlossene Formel x diese Formel selbst, und für andere Formeln $\mathbf{(}x\mathbf{)}$ bedeutet. (Falls x keine Formel ist, so ist der Wert von x^* belanglos für uns.) Nach der Definition

$$x^* = \begin{cases} x, \text{ falls } \mathbf{V} \nleqq x \vee (y)\left[y \leqq x \rightarrow \left(l_1(e_{\sigma(y)}(x)) = \mathbf{V} \rightarrow \right.\right. \\ \qquad \left.\left. \rightarrow h^{()}(e_{\sigma(y)}(x)) \prec h^{(()}(e_{\sigma(y)}(x))\right)\right] \\ \mathbf{(}x\mathbf{)} \text{ sonst} \end{cases}$$

ist auch x^* primitiv-rekursiv in $\mathfrak{M}$.

Für eine Formel x ist

$$x = x^*$$

gleichbedeutend damit, dass x geschlossen ist.

Die Formeln werden nun folgendermassen aufgebaut:

1) Es wird auch eine »leere« Formel zugelassen.

2) Jedes A_i und jedes $\bar{A}_i$ ist eine Formel.

3) Sind $\mathfrak{f}_1$ und $\mathfrak{f}_2$ bereits nicht leere Formeln, dann sind

$$\mathfrak{f}_1 \mathbf{V} \mathfrak{f}_2 \text{ und } \mathfrak{f}_1^* \mathfrak{f}_2^*$$

auch Formeln.

Daher kann endlich $f_F(x)$ wie folgt definiert werden:

$$f_F(\Lambda) = \Lambda$$

und für $a \in \mathfrak{A}$

$$f_F(xa) = \begin{cases} \Lambda, \text{ falls } f_A(xa) = \Lambda \vee f_{\bar{A}}(xa) = \Lambda \vee \\ \qquad \vee (Ey_1)(Ey_2)\,[y_1 \leqq x \;\&\; y_2 \leqq {}^{-1}(xa) \;\& \\ \qquad \& \; y_1 \neq \Lambda \;\&\; y_2 \neq \Lambda \;\&\; f_F(y_1) = f_F(y_2) = \Lambda \;\& \\ \qquad \& \; (xa = y_1 \mathbf{V} y_2 \vee xa = y_1^* y_2^*)] \\ (\qquad \text{sonst.} \end{cases}$$

Da $f_F(xa)$ mit Hilfe von Funktionswerten $f_F(y_1)$ und $f_F(y_2)$ definiert wird, wo $y_1 \leqq x$ und $y_2 \leqq {}^{-1}(xa)$, zeigt man leicht, dass dies eine Wertverlaufsrekursion, und daher $f_F(x)$ in $\mathfrak{M}$ primitiv-rekursiv ist. Ich berufe mich dabei (und auch in den folgenden ähnlichen Fällen) auf meine frühere Arbeit,[2] worin ich solche Gedankengänge bis auf die Einzelheiten durchgeführt habe. Und dasselbe geht genau so für jede Definition dieser Arbeit, wo der Wert einer Funktion an einer Stelle mit Hilfe von ihren an Vorgängern dieser Stelle angenommenen Werte definiert wird.

6. Ich schicke noch die Definition einiger Hilfsfunktionen voraus.

(1) Sei x eine Formel, und $l(x)$ das letzte (geschlossene) Disjunktionsglied von x. Diese ergibt sich nach der Definition

$$l(x) = \begin{cases} x, \text{ falls } x = x^* \\ \mu_y\,[y \leqq x \;\&\; f_F(y) = \Lambda \;\&\; y = y^* \;\&\; l_{\sigma(y)+1}(x) = \mathbf{V}\, y] \text{ sonst} \end{cases}$$

als primitiv-rekursiv in $\mathfrak{M}$ (falls x keine Formel ist, ist der Wert $l(x)$ belanglos für uns; ähnliches gilt in den Folgenden mehrmals).

(2a) Sei $f(y)$ primitiv-rekursiv in $\mathfrak{M}$ mit $f(y) \neq \Lambda$ für $y \neq \Lambda$, und x eine Formel; dann ist $F_f^{(d)}(x)$, das aus x so entsteht, dass jedes Disjunktionsglied y von x durch $f(y)$ ersetzt wird, auch primitiv-rekursiv in $\mathfrak{M}$. Denn die Definition von $F_f^{(d)}(x)$ lautet, mit Hilfe der durch

$$h_{\mathbf{V}}(x) = \begin{cases} \mathbf{V}, & \text{falls } x \neq x^* \\ \Lambda & \text{sonst} \end{cases}$$

definierten, in $\mathfrak{M}$ primitiv-rekursiven Funktion:

$$F_f^{(d)}(\Lambda) = \Lambda$$

und für $a \in \mathfrak{A}$

$$F_f^{(d)}(xa) = F_f^{(d)}((xa)^{-(1+o('(xa)))})\, h_{\mathbf{v}}(xa)\, f(l(xa)) ,$$

und man zeigt leicht, dass dies eine Wertverlaufsrekursion ist.

(2b) Noch einfacher definiert man $F_f^{(k)}(x)$, das aus der reinen Konjunktion x entsteht, wenn darin jedes Konjunktionsglied y (d.h. jeder Buchstabe y) durch $f(y)$ ersetzt wird, als primitiv-rekursive Funktion in $\mathfrak{M}$ (hier kann auch Λ beliebig unter den Werten $f(y)$ vorkommen):

$$F_f^{(k)}(\Lambda) = \Lambda$$

und für $a \in \mathfrak{A}$

$$F_f^{(k)}(xa) = F_f^{(k)}(x)\, f(a) .$$

Wird diese Definition so modifiziert, dass darin $f(o(xa))$ statt $f(a)$ steht, so erhält man

$$K_f(x) = f(1)\, f(2) \ldots f(o(x))$$

als mit f primitiv-rekursive Funktion in $\mathfrak{M}$. (f und damit auch K_f kann auch von Parametern abhängen; diese schreibe ich nie an die erste Argumentenstelle.)

(3) Mit einer Funktion $f(x)$ (mit beliebigen Parametern) ist auch ihre $o(y)$-te Iteration $f^{o(y)}(x)$ primitiv-rekursiv in $\mathfrak{M}$, zufolge der Definition:

$$f^{o(\Lambda)}(x) = (x)$$

und für $a \in \mathfrak{A}$

$$f^{o(ya)}(x) = f(f(^{o(y)}(x)) .$$

($f^{o(y)}(x)$ ist eine Funktion von x und y, welche nur von x und $o(y)$ abhängt. Ähnliche Bezeichnungen kommen hier öfters vor.)

§ 2.

7. In diesem Paragraph kommt es darauf an, die ausgezeichnete disjunktive Normalform in n Variablen einer Formel x (von höchstens n Variablen) zu definieren. Das geschieht in kleinen Schritten, mit Hilfe einer Kette von Hilfsfunktionen.

Zuert soll die Folge der Variablen von x, falls sie keine Teilfolge der Folge

$$A_1, A_2, \ldots, A_n$$

bildet, zu einem solchen umbenannt werden.

Man zeigt leicht, dass $A_{o(x)}$ und $\bar{A}_{o(x)}$ (mit $A_o = \bar{A}_o = \Lambda$) primitiv-rekursiv in $\mathfrak{M}$ sind. Dann ist mit $f(x) = A_{o(x)}\bar{A}_{o(x)}$

$$A_1 \bar{A}_1 \dots A_{o(y)} \bar{A}_{o(y)} = K_f(y) ;$$

sei dies kurz mit $g_{o(y)}$ bezeichnet.

Wenn nun $i_{o(y)}(x)$ die kleinste natürliche Zahl $i \leqq o(x)$ bezeichnet, für welche das i-te Zeichen des Wortes x ein in $g_{o(y)}$ nicht vorkommendes A_j oder $\bar{A}_j$ ist, dann kann dies durch

$$i_{o(y)}(x) = \mu_i \left[i \leqq o(x) \mathbin{\&} \left(f_A(b_i(x)) = \Lambda \vee f_{\bar{A}}(b_i(x)) = \Lambda\right) \mathbin{\&} b_i(x) \mathrel{\underline{\not\leqq}} g_{o(y)}\right]$$

als in $\mathfrak{M}$ primitiv-rekursive Funktion definiert werden.

Auch

$$j_{o(y)}(x) = \mu_j \left[j \leqq o(y) \mathbin{\&} A_j \mathrel{\underline{\leqq}} x \mathbin{\&} \bar{A}_j \mathrel{\underline{\leqq}} x\right]$$

ist primitiv-rekursiv in $\mathfrak{M}$.

Man braucht noch eine Hilfsfunktion $\bar{x}$, welche einem A_i immer $\bar{A}_i$ zuordnet und umgekehrt (und an belanglosen Stellen als Λ definiert werden kann). Diese kann durch die

$$\bar{\Lambda} = \Lambda$$

$$\overline{xA_i} = \begin{cases} \bar{A}_i, & \text{falls } x = \Lambda \\ \Lambda & \text{sonst} \end{cases}$$

$$\overline{x\bar{A}_i} = \begin{cases} A_i, & \text{falls } x = \Lambda \\ \Lambda & \text{sonst} \end{cases}$$

$$\overline{x\,(} = \overline{x\,)} = \overline{x\mathbf{V}} = \Lambda$$

primitive Rekursion definiert werden.

Mit diesen ist auch

$$t_{o(y)}(x) = \begin{cases} \operatorname{subst}\left(\operatorname{subst}\,(x, b_{i_{o(y)}(x)}(x), A_{j_{o(y)}(x)}), \overline{b_{i_{o(y)}(x)}(x)}, \bar{A}_{j_{o(y)}(x)}\right), \\ \qquad \text{falls } f_A\,(b_{i_{o(y)}(x)}(x)) = \Lambda \\ \operatorname{subst}\left(\operatorname{subst}\,(x, b_{i_{o(y)}(x)}(x), \bar{A}_{j_{o(y)}(x)}), \overline{b_{i_{o(y)}(x)}(x)}, A_{j_{o(y)}(x)}\right), \\ \qquad \text{falls } f_{\bar{A}}(b_{i_{o(y)}(x)}(x)) = \Lambda \\ x \text{ sonst} \end{cases}$$

primitiv-rekursiv in $\mathfrak{M}$, und auch ihre Iteration $t_{o(y)}^{o(z)}(x)$.

Dann ist für eine Formel x von höchstens n Variablen

$$t'_n(x) = t_n^n(x)$$

eine solche Transformierte der Formel x, worin als Variablen eine Teilfolge der Folge

$$A_1, A_2, \dots, A_n$$

(negiert oder unnegiert oder beiderweise) vorkommen. In den Weiteren gehe ich von derartigen Formeln x aus.

8. Der nächste Schritt ist die Auflösung der Klammern einer solchen Formel x; so wird die nächste Transformierte bereits eine disjunktive Form sein, d.h. eine Disjunktion, deren Glieder Konjunktionen von — unnegierten oder negierten — Variablen sind.

Sei erst x eine geschlossene Formel, welche auch Klammern enthält; und sei ihre in Nr. 3 besprochene Darstellung

$$x = z_1 \boldsymbol{(} y_1 \mathbf{V}\, y_2 \boldsymbol{)}\, z_2.$$

Hier ist

$$z_1 = z_1(x) = \mu_z [z \leqq x \,\&\, f_F(z) = \Lambda \,\&\, \boldsymbol{(} \mathrel{\underline{\not\leq}} z \,\&\, e_{o(z)+1}(x) = z \boldsymbol{(}\,],$$

$$y_1 = y_1(x) = \mu_y [y \leqq x \,\&\, f_F(y) = \Lambda \,\&\, y = y^* \,\&\, e_{o(z_1 \boldsymbol{(} y)}(x) = z_1 \boldsymbol{(} y],$$

$$y_2 = y_2(x) = \mu_y [y \leqq x \,\&\, f_F(y) = \Lambda \,\&\, e_{o(z_1 \boldsymbol{(} y_1 \mathbf{v} y_2 \boldsymbol{)})}(x) = z_1 \boldsymbol{(} y_1 \mathbf{V}\, y_2 \boldsymbol{)}],$$

$$z_2 = z_2(x) = \mu_z [z \leqq x \,\&\, x = z_1 \boldsymbol{(} y_1 \mathbf{V}\, y_2 \boldsymbol{)}\, z]$$

(aus der Definition von z_1, y_1 und y_2 folgt schon, dass z_2 auch eine Formel, und zwar wegen der Geschlossenheit von x eine geschlossene Formel ist). $z_1(x)$, $y_1(x)$, $y_2(x)$ und $z_2(x)$ sind primitiv-rekursiv in $\mathfrak{M}$.

Die Auflösung der ausgeschriebenen Klammern von $x = z_1 \boldsymbol{(} y_1 \mathbf{V}\, y_2 \boldsymbol{)} z_2$ besteht darin, dass man dieses Wort infolge der Distributivität durch

$$y_1 z_1 z_2 \mathbf{V}\, y_2^* z_1 z_2$$

ersetzt (die Kommutativität und Assoziativität unserer Operationen wurde auch bisher stillschweigend benutzt). Im ersten Disjunktionsglied $y_1 z_1 z_2$ kommen dann höchstens um eins wenigere Klammernpaare vor, wie im ursprünglichen Wort x, und im zweiten Disjunktionsglied $y_2^* z_1 z_2$ auch höchstens um eins wenigere, falls $y_2^* = y_2$ ist, und höchstens ebensoviele sonst, wenn nähmlich $y_2^* = \boldsymbol{(} y_2 \boldsymbol{)}$ ist. Doch enthält y_2 auch im letzteren Fall um eins wenigere Disjunktionsglieder als $y_1 \mathbf{V}\, y_2$ in x; und wenn das Verfahren auf $\boldsymbol{(} y_2 \boldsymbol{)} z_1 z_2$ wiederholt wird, worin das erste $\boldsymbol{(}$-Zeichen das vor y_2 stehende $\boldsymbol{(}$-Zeichen ist, dann vermindert sich die Anzahl der Disjunktionsglieder im entstehenden »neuen y_2«. Nach weniger als $o(x)$ Schritten kommt man so zu einer Formel, deren sämtliche Disjunktionsglieder wenigere Klammernzeichen enthalten als x.

Sei also x erst eine geschlossene Formel, und $d'(x)$ durch

$$d'(x) = \begin{cases} x, \text{ falls } \boldsymbol{(} \mathrel{\underline{\not\leq}} x \\ y_1(x)\, z_1(x)\, z_2(x) \mathbf{V}\, y_2^*(x)\, z_1(x)\, z_2(x) \text{ sonst} \end{cases}$$

definiert. $d''(x)$ entstehe durch Anwendung von $d'(x)$ auf alle Disjunktionsglieder von x, falls x eine beliebige Formel ist. Dann ist

$$d''(x) = F_{d'}^{(d)}(x)\,.$$

Die $o(x)$-te Iteration von $d''(x)$ liefert eine mit x äquivalente Formel, deren Disjunktionsglieder höchstens um eins wenigere Klammernpaare enthalten, als die Disjunktionsglieder von x. Die $o(x)$-mal $o(x)$-te Iteration von $d''(x)$ liefert also bestimmt eine in $\mathfrak{M}$ primitiv-rekursive, klammernfreie Transformierte, d.h. eine disjunktive Form

$$t''(x)$$

der Formel x.

9. Jetzt nehmen wir an, das x bereits eine disjunktive Form von gewissen der Variablen $A_1, A_2, \ldots, A_n$ ist, und wir wollen in ihren Disjunktionsgliedern die Wiederholungen streichen. Die so transformierte Formel sei mit $t_n'''(x)$ bezeichnet. Zur Definition von $t_n'''(x)$ seien einige Hilfsfunktionen vorausgeschickt.

Erst sei für ein beliebiges Wort x das daraus durch Streichen der Wiederholungen seiner (vom Anfang des Wortes an betrachteten) Buchstaben entstehende Wort $h_1(x)$. Dies kann durch folgende primitive Rekursion definiert werden:

$$h_1(\Lambda) = \Lambda$$

und für $a \in \mathfrak{A}$

$$h_1(xa) = \begin{cases} h_1(x), & \text{falls } a \leq x \\ h_1(x)\, a & \text{sonst}. \end{cases}$$

Damit ist auch die auf eine disjunktive Form x angewandte Funktion

$$t_n'''(x) = F_{h_1}^{(d)}(x)$$

primitiv-rekursiv in $\mathfrak{M}$.

10. Nun wollen wir aus einer disjunktiven Form x jene Disjunktionsglieder streichen, in welchen sowohl ein A_i als auch $\bar{A}_i$ vorkommt.

Die durch

$$f_{A\bar{A}}^{(n)}(x) = \begin{cases} \Lambda, & \text{falls } (E_y)\,[y \leq n \;\&\; A_{o(y)} \leq x \;\&\; \bar{A}_{o(y)} \leq x] \\ x & \text{sonst} \end{cases}$$

definierte (nicht unbedingt nur für $x = \Lambda$ verschwindende) Funktion ist primitiv-rekursiv in $\mathfrak{M}$, und mit ihr auch $F_{f_{A\bar{A}}^{(n)}}^{(d)}(x)$, die aus einer disjunktiven Form zwar die ungewünschten Disjunktionsglieder streicht, dafür aber ein solches Wort zustande bringt, das benachbarte, oder am Ende bzw. am Anfang des Wortes stehende $\mathbf{V}$-Zeichen enthalten kann. Diese werden entfernt, wenn in $F_{f_{A\bar{A}}^{(n)}}^{(d)}(x)$ einfach $\mathbf{V}$ für jedes Vorkommen von $\mathbf{VV}$ gesetzt wird, und auf das Ergebnis die folgenderweise als in $\mathfrak{M}$ primitiv-rekursiv definierten Funktionen $l_{\mathbf{v}}(x)$, $e_{\mathbf{v}}(x)$ angewandt werden:

$$l_{\mathbf{v}}(x) = \begin{cases} x, & \text{falls } l_1(x) \neq \mathbf{V} \\ x^{-1} & \text{sonst}, \end{cases}$$

$$e_{\mathbf{v}}(x) = \begin{cases} x, & \text{falls } e_1(x) \neq \mathbf{V} \\ {}^{-1}x & \text{sonst}. \end{cases}$$

So ergibt sich die gewünschte Transformierte einer disjunktiven Form x als

$$t^{\mathrm{IV}}(x) = e_{\mathbf{v}}\Big(l_{\mathbf{v}}\big(\mathrm{sub}\,(F_{f_{A\bar{A}}^{(n)}}^{(d)}(x),\, \mathbf{VV},\, \mathbf{V})\big)\Big).$$

11. Um eine *ausgezeichnete* disjunktive Form *in n Variablen* der bisher transformierten Formel x zu erhalten, hat man x zu einer solchen disjunktiven Form $t_n^{\mathrm{V}}(x)$ zu transformieren, deren jedes Glied sämtliche der Variablen $A_1, \ldots, A_n$ (unnegiert oder negiert) enthält. Ein Disjunktionsglied $\mathfrak{g}$, das ein A_i mit $1 \leq i \leq n$ nicht enthält, kann durch

$$\mathfrak{g}\, A_i\, \mathbf{V}\, \mathfrak{g}\, \bar{A}_i$$

ersetzt werden. Sei $a_n(x)$ die Formel, die aus einem Disjunktionsglied x einer durch den Bisherigen transformierten Formel entsteht, wenn auf ihm die genannte Umformung mit dem kleinstmöglichen i angewandt wird. Diese Funktion ist nach der Definition

$$a_n(x) = \begin{cases} x, \text{ falls } (i)\,[i \leqq n \rightarrow (A_i \leqq x \vee \bar{A}_i \leqq x)] \\ x A_{o(y_0)} \mathbf{V}\, x \bar{A}_{o(y_0)} \text{ sonst, wobei} \\ y_0 = \mu_y\,[y \leqq n \,\&\, A_{o(y)} \not\leqq x \,\&\, \bar{A}_{o(y)} \not\leqq x] \end{cases}$$

primitiv-rekursiv in $\mathfrak{M}$; und dasselbe gilt für ihre Iteration $a_n^{o(z)}(x)$. Und die gewünschte Transformierte eines Disjunktionsgliedes x ist $a_n^n(x)$.

Ist nun x die ganze disjunktive Form (nach den bisherigen Umformungen), dann erhält man

$$t_n^{\mathrm{V}}(x) = F_{a_n^n}^{(d)}(x)$$

als primitiv-rekursive Funktion in $\mathfrak{M}$.

12. Es soll nun die natürliche Reihenfolge der Konjunktionsglieder hergestellt werden. Das leistet zuerst für ein Disjunktionsglied x einer durch die bisherigen Schritten transformierten Formel die in $\mathfrak{M}$ primitiv-rekursive Funktion $h_3^{(n)}(x)$, die sich mit Hilfe der durch

$$f(y, x) = \begin{cases} A_{o(y)}, \text{ falls } A_{o(y)} \leqq x \\ \bar{A}_{o(y)}, \text{ falls } \bar{A}_{o(y)} \leqq x \\ \text{etwa } \Lambda \text{ sonst} \end{cases}$$

definierten Funktion als

$$h_3^{(n)}(x) = K_f(n, x)$$

ergibt, und dann, falls x die ganze disjunktive Form ist, die Transformierte

$$t_n^{\mathrm{VI}}(x) = F_{h_3^{(n)}}^{(d)}(x)\,.$$

13. Die letzte Transformierte $t^{\mathrm{VII}}(x)$ der bisher transformierten Formel x entsteht durch Streichung der Wiederholungen der Disjunktionsglieder. Diese kann durch folgende Wertverlaufsrekursion definiert werden:

$$t^{\mathrm{VII}}(\Lambda) = \Lambda$$

und für $a \in \mathfrak{A}$

$$t^{\mathrm{VII}}(xa) = \begin{cases} t^{\mathrm{VII}}\left((xa)^{-(1+o(l(xa)))}\right), \text{ falls } l(xa) \leqq (xa)^{-(1+o(l(xa)))} \\ t^{\mathrm{VII}}\left((xa)^{-(1+o(l(xa)))}\right) h_{\mathbf{v}}(xa)\, l(xa) \text{ sonst}. \end{cases}$$

14. Nach Nr. 7—13 erhält man die ausgezeichnete disjunktive Normalform in n Variablen, $d_n(x)$, einer Formel x von höchstens n Variablen durch die Definition

$$d_n(x) = t^{\mathrm{VII}}\left(t_n^{\mathrm{VI}}\left(t_n^{\mathrm{V}}\left(t_n^{\mathrm{IV}}\left(t_n^{\mathrm{III}}\left(t''\left(t'_n(x)\right)\right)\right)\right)\right)\right)$$

als in $\mathfrak{M}$ primitiv-rekursive Funktion von x und n.

§ 3.

15. Jetzt sollen die von

$$A_1, \bar{A}_1, A_2, \bar{A}_2, \ldots, A_n, \bar{A}_n$$

aufgebauten Formeln der Länge m geordnet werden. Ihre Anzahl soll mit $\alpha(m, n)$ bezeichnet werden, und die r-te Formel in unserer Anordnung mit $f(r, m, n)$. Die eben aufgezeichnete Folge gibt zugleich die Anordnung solcher Formeln der Länge 1 an; daher ist

$$\alpha(1, n) = 2n,$$

und für $1 \leqq r \leqq 2n$

$$f(r, 1, n) = \begin{cases} \bar{A}_{\left[\frac{r}{2}\right]}, & \text{falls } 2/r \\ A_{\left[\frac{r+1}{2}\right]} & \text{sonst}. \end{cases}$$

Falls nun die Reihenfolge der Formeln kleinerer Länge als m bereits angegeben ist, dann soll die Reihenfolge der Formeln der Länge m (diese entstehen aus je zwei Formeln kleinerer Länge durch Disjunktion oder Konjunktion) die folgende sein:

Der Reihe nach für $i = 1, 2, \ldots, m - 1$ sollen von den beiden Gliedern aller Paare

$$(f(1, i, n), f(1, m - i, n)), (f(1, i, n), f(2, m - i, n)), \ldots$$

$$\ldots, \big(f(1, i, n), f(\alpha(m - i, n), m - i, n)\big); \ldots; \big(f(\alpha(i, n), i, n), f(1, m-i, n)\big), \ldots$$

$$\ldots, \big(f(\alpha(i, n), i, n), f(\alpha(m - i, n), m - i, n)\big)$$

der Reihe nach erst die Disjunktionen gebildet werden, dann die Konjunktionen.

So ergibt sich für $m > 1$ für die zahlentheoretische Funktion $\alpha(m, n)$

$$\alpha(m, n) = 2 \sum_{i=1}^{m \dot{-} 1} \alpha(i, n)\, \alpha(m \dot{-} i, n),$$

und dies gibt mit

$$\alpha(1, n) = 2n$$

und etwa

$$\alpha(0, n) = 1$$

eine zahlentheoretische Wertverlaufsrekursion für $\alpha(m, n)$. Daher ist $\alpha(m, n)$ eine primitiv-rekursive zahlentheoretische Funktion; diese kann zu einer primitiv-rekursiven Funktion in $\mathfrak{M}$ erweitert werden.

16. Die Formel $f(r, m, n)$ wird nach der gegebenen Anordnung aus einer Formel der Länge i_0 und aus einer Formel der Länge $m - i_0$ aufgebaut, wobei

$$i_0 = \mu_i \left[i \leqq m \dot{-} 1 \;\&\; 2 \sum_{j=1}^{i} \alpha(j, n)\, \alpha(m \dot{-} j, n) \geqq r \right]$$

ist; und genauer ist $f(r, m, n)$ die r_0-te unter den Formeln, die aus einer Formel der Länge i_0 und aus einer Formel der Länge $m \dot{-} i_0$ aufgebaut werden, wobei

$$r_0 = r \dot{-} 2 \sum_{j=1}^{i_0 \dot{-} 1} \alpha(j, n)\, \alpha(m \dot{-} j, n)$$

ist. Noch näher entsteht $f(r, m, n)$ aus den beiden genannten Formeln durch Disjunktion oder durch Konjunktion, je nachdem

$$r_0 \leqq \alpha(i_0, n)\, \alpha(m \dot{-} i_0, n) \quad \text{oder} \quad r_0 > \alpha(i_0, n)\, \alpha(m \dot{-} i_0, n)$$

ist; im ersten Fall ist $f(r, m, n)$ die r_0-te unter den Disjunktionen, im zweiten Fall ist sie die r_0'-te unter den Konjunktionen der genannten Art, wobei

$$r_0' = r_0 \dot{-} \alpha(i_0, n)\, \alpha(m \dot{-} i_0, n)$$

ist. Und zwar ist

$$f(r, m, n) = f(r_1\, i_0, n)\, \mathbf{V}\, f(r_2, m \dot{-} i_0, n)$$

oder

$$f(r, m, n) = f^*(r_1', i_0, n)\, f^*(r_2', m \dot{-} i_0, n)\,,$$

wobei

$$r_1 = \mu_r\, [r \leqq \alpha(i_0, n) \;\&\; r\, \alpha(m \dot{-} i_0, n) \geqq r_0]\,,$$

$$r_1' = \mu_r\, [r \leqq \alpha(i_0, n) \;\&\; r\, \alpha(m \dot{-} i_0, n) \geqq r_0']\,,$$

$$r_2 = r_0 \dot{-} (r_1 \dot{-} 1)\, \alpha(m \dot{-} i_0, n)\,,$$

$$r_2' = r_2' \dot{-} (r_1' \dot{-} 1)\, \alpha(m \dot{-} i_0, n)\,.$$

Nach diesen Definitionen sind $i_0, r_0, r_0', r_1, r_1', r_2, r_2'$ zahlentheoretische primitiv-rekursive Funktionen. Mit diesen ergibt sich folgende Definition für $f(r, m, n)$:

$$f(r, 0, n) = \Lambda$$

$$f(r, 1, n) = \begin{cases} \bar{A}_{\left[\frac{r}{2}\right]}\,, & \text{falls } 2/r \\ A_{\left[\frac{r+1}{2}\right]} & \text{sonst} \end{cases}$$

und für $m > 1$

$$f(r, m, n) = \begin{cases} f(r_1, i_0, n)\, \mathbf{V}\, f(r_2, m \dot{-} i_0, n)\,, \\ \text{falls } r_0 \leqq \alpha(i_0, n)\, \alpha(m \dot{-} i_0, n) \\ f^*(r_1', i_0, n)\, f^*(r_2', m - i_0, n) \text{ sonst}\,. \end{cases}$$

Diese Definition kann (etwa mit $f(x, y, z) = \Lambda$ falls x, y oder z keine natürliche Zahl ist) auf die ganze Menge $\mathfrak{M}$ erweitert werden. Sie ist eine sogenannte eingeschachtelte Rekursion (da für den Parameter r Einsetzungen erfolgen) und zugleich eine Wertverlaufsrekursion (da sowohl i_0, als auch $m \dot{-} i_0$ Vorgänger von m sind); diese lassen sich aber auf primitive Rekursion in $\mathfrak{M}$ zurückführen.

§ 4.

17. Ist nun x eine beliebige Formel von n Variablen (x kann auch eine ausgezeichnete disjunktive Normalform, also eine Art der Angabe der Wertetabelle von x sein), dann kann man die Länge einer kürzesten mit x äquivalenten Formel, d.h. die minimale Länge $\lambda(x, n)$, wodurch x ausgedrückt werden kann, durch

$$\lambda(x, n) = \mu_m \left[m \leq o(x) \,\&\, (Er)\, [r \leq \alpha(m, n) \,\&\, d_n(x) = d_n(f(r, m, n))]\right]$$

als in $\mathfrak{M}$ primitiv-rekursive Funktion definieren.

Eine bestimmte mit x äquivalente kürzeste Formel erhält man in

$$f(r', \lambda(x, n), n)\,,$$

wobei

$$r' = \mu_r \left[r \leq \alpha(\lambda(x, n), n) \,\&\, d_n(x) = d_n\big(f(r, \lambda(x, n), n)\big)\right],$$

als primitiv-rekursive Funktion in $\mathfrak{M}$.

§ 5.

18. Der Beweis kann auch so geführt werden, dass darin die Wertetabellen der Formeln unmittelbar — ohne Berufung auf die ausgezeichnete disjunktive Normalform der Formel — zur Anwendung kommen.[4] Dadurch wird die Beweisführung etwas kürzer, aber das Alphabet $\mathfrak{A}$ muss zu

$$\mathfrak{A}' = \{(,), \mathbf{V}, \uparrow, \downarrow, A_1, \bar{A}_1, A_2, \bar{A}_2, \ldots\}$$

erweitert werden, wo $\uparrow$ und $\downarrow$ die Zeichen für »wahr« bzw. »falsch« sind. Selbstverständlich sind in der so erhaltenen Wortemenge $\mathfrak{M}'$ auch die charakteristischen Funktionen $f_\uparrow(x)$ und $f_\downarrow(x)$ der Eigenschaften »$\uparrow$ zu sein« bzw. »$\downarrow$ zu sein« primitiv-rekursiv; und alle bisher als in $\mathfrak{M}$ primitiv-rekursiv erkannten Funktionen sind auch in $\mathfrak{M}'$ primitiv-rekursiv, oder können zu in $\mathfrak{M}'$ primitiv-rekursive Funktionen sinngemäss erweitert werden (diese werden in ihrer erweiterten Form genau so bezeichnet werden wie vorher). So lautet z.B. die erweiterte Definition von $f_A(x)$:

$$f_A(A) = ($$

$$f_A(xA_i) = \begin{cases} A, & \text{falls } x = A \\ (& \text{sonst} \end{cases}$$

$$f_A(x\,() = f_A(x)) = f_A(x\mathbf{V}) = f_A(x\uparrow) = f_A(x\downarrow) = f_A(x\bar{A}_i) = (\,;$$

[4] Das war bereits bei der Axiomatisierung des Aussagenkalküls eine Vereinfachungsidee von KALMÁR: s. L. KALMÁR, Über die Axiomatisierbarkeit des Aussagenkalküls, *Acta Sci. Math. Szeged* 4 (1935) S. 222—243.

und von $\bar{x}$:

$$\bar{\Lambda} = \Delta$$

$$\overline{xA_i} = \begin{cases} \bar{A}_i, & \text{falls } x = \Lambda \\ \Lambda & \text{sonst} \end{cases}$$

$$\overline{x\bar{A}_i} = \begin{cases} A_i, & \text{falls } x = \Lambda \\ \Lambda & \text{sonst} \end{cases}$$

$$\overline{x\uparrow} = \begin{cases} \downarrow, & \text{falls } x = \Lambda \\ \Lambda & \text{sonst} \end{cases}$$

$$\overline{x\downarrow} = \begin{cases} \uparrow, & \text{falls } x = \Lambda \\ \Lambda & \text{sonst} \end{cases}$$

$$\overline{x\,(} = \overline{x\,)} = \overline{x\,\mathbf{V}} = \Lambda\,.$$

19. Für die Folge der Variablen $A_1, \ldots, A_n$, aus welchen die in Aufbau einer Formel von höchstens n Variablen teilnehmenden Variablen gewählt werden, können die aus $\uparrow$ und $\downarrow$ gebildeten n-gliedrigen Folgen eingesetzt werden (ihre Anzahl ist 2^n). Man erhält die übliche Reihenfolge der letzteren Folgen, indem man, von $\uparrow\uparrow\ldots\uparrow$ ausgehend, von einer solchen Folge x, welche mindestens ein $\uparrow$-Glied enthält, auf die nächste so übergeht, dass man von rechts nach links gehend das erste $\uparrow$-Glied von x durch $\downarrow$, und alle rechts davon stehenden Glieder durch $\uparrow$ ersetzt. Dann erhält man $h_4(x)$, wo $h_4(x)$ durch folgende primitive Rekursion definiert werden kann:

$$h_4(\Lambda) = \Lambda$$

und für $a \in \mathfrak{A}'$

$$h_4(xa) = \begin{cases} x\downarrow, & \text{falls } a = \uparrow \\ h_4(x)\uparrow, & \text{falls } a = \downarrow \\ \Lambda & \text{sonst}. \end{cases}$$

Damit kann für $o(y) \leq 2^n$ die $o(y)$-te n-gliedrige Folge $f^{(n)}_{o(y)}$ der Elemente $\uparrow$ und $\downarrow$ (kurz: die $o(y)$-te Argumentenfolge) durch die folgende primitive Rekursion definiert werden:

$$f^{(n)}_{o(\Lambda)} = \Lambda$$

und für $a \in \mathfrak{A}'$

$$f^{(n)}_{o(ya)} = \begin{cases} \text{sonst } (n, (, \uparrow), & \text{falls } y = \Lambda \\ h_4(f^{(n)}_{o(y)}) & \text{sonst}\,; \end{cases}$$

für $o(y) > 2^n$ sind die Werte dieser Funktion (die sich periodisch wiederholen) belanglos für uns (ebenso wie ihre Werte wenn n keine natürliche Zahl ist).

20. Es soll nun der Wert $w^{(n)}_i(x)$ einer aus gewissen der Variablen A_1, $A_2, \ldots, A_n$ gebildeten Formel x an der Stelle $f^{(n)}_i$ (d.h. wenn man für $A_1, \ldots, A_n$ der Reihe nach die Buchstaben von $f^{(n)}_i$ einsetzt) angegeben werden.

Dazu braucht man erst die Definition der Formel $e_i^{(n)}(x)$, welche durch diese Einsetzung aus x entsteht. Diese erhält man mit Hilfe der durch die folgende primitive Rekursion definierten Hilfsfunktion $h_{i,o(y)}^{(n)}(x)$:

$$h_{i,o(\Lambda)}^{(n)}(x) = x$$

und für $a \in \mathfrak{A}'$

$$h_{i,o(ya)}^{(n)}(x) = \text{subst}\big(\text{subst}\,(h_{i,o(y)}^{(n)}(x),\, A_{o(ya)},\, b_{o(ya)}\,(f_i^{(n)})),\, \overline{A}_{o(ya)},\, \overline{b_{o(ya)}(f_i^{(n)})}\big),$$

als

$$e_i^{(n)}(x) = h_{i,n}^{(n)}(x)\,.$$

So entsteht ein Wort, das aus Zeichen $\uparrow$ und $\downarrow$ durch (unbezeichneten) Konjunktionen und mit $\mathbf{V}$ bezeichneten Disjunktionen mit unseren Klammernkonventionen aufgebaut wird. Sei nun x ein derartiges Wort. Wird darin sukzessiv jedes Vorkommen

von $\uparrow\,\uparrow$	durch $\uparrow$
von $\uparrow\,\downarrow$	durch $\downarrow$
von $\downarrow\,\uparrow$	durch $\downarrow$
von $\downarrow\,\downarrow$	durch $\downarrow$
von $\uparrow\mathbf{V}\uparrow$	durch $\uparrow$
von $\uparrow\mathbf{V}\downarrow$	durch $\uparrow$
von $\downarrow\mathbf{V}\uparrow$	durch $\uparrow$
von $\downarrow\mathbf{V}\downarrow$	durch $\downarrow$
von $(\uparrow)$	durch $\uparrow$
von $(\downarrow)$	durch $\downarrow$

ersetzt, so ergibt die $o(x)$-te Iteration dieses — die Länge des Wortes vermindernden — Verfahrens ein einziges Zeichen $\uparrow$ oder $\downarrow$, als den Wert der Formel x.

Daher ergibt sich $w_i^{(n)}(x)$ als die $o(x)$-te Iteration der Funktion

$$\text{sub}\,(\text{sub}\,(\text{sub}\,(\text{sub}\,(\text{sub}\,(\text{sub}\,(\text{sub}\,(\text{sub}\,(\text{sub}\,(\text{sub}\,(x,\, \uparrow\,\uparrow,\, \uparrow),\, \uparrow\,\downarrow,\, \downarrow),$$
$$\downarrow\,\uparrow,\, \downarrow),\, \downarrow\,\downarrow,\, \downarrow),\, \uparrow\mathbf{V}\uparrow,\, \uparrow),\, \uparrow\mathbf{V}\downarrow,\, \uparrow),\, \downarrow\mathbf{V}\uparrow,\, \uparrow),\, \downarrow\mathbf{V}\downarrow,\, \downarrow),$$
$$(\uparrow),\, \uparrow),\, (\downarrow),\, \downarrow),$$

also als in $\mathfrak{M}'$ primitiv-rekursive Funktion.

21. Sei nun x eine Formel von n Variablen. (Falls die Wertetabelle dieser Formel gegeben ist, so liefert z.B. die daraus unmittelbar herauslesbare ausgezeichnete disjunktive Normalform die Formel, von der wir ausgehen.) Dann bedeutet $t_n'(x)$ eine Äquivalente von x, worin gerade die Variablen A_1, $A_2, \ldots, A_n$ vorkommen. So erhält man (mit der in Nr. 13 eingeführten Funktion $f(r, m, n)$)

$$\lambda(x, n) = \mu_m\big[m \leqq o(x)\;\&\;(Er)\,[r \leqq \alpha(m, n)\;\&$$
$$\&\;(i)\,[i \leqq 2^n \rightarrow w_i^{(n)}\,(t_n'(x)) = w_i^{(n)}\,(f(r, m, n))]\big]\,.$$

und eine bestimmte mit x äquivalente kürzeste Formel liefert

$$f(r', \lambda(x, n), n),$$

wobei

$$r' = \mu_r \left[r \leqq \alpha(\lambda(x, n), n) \,\&\, (i)\,[i \leqq 2^n \to w_i^{(n)}(t'_n(x)) = w_i^{(n)}(f(r, \lambda(x, n), n))] \right].$$

(Eingegangen: 11. September, 1961.)

О НАИБОЛЕЕ КРАТКОМ ВИДЕ БУЛЕВЫХ ФУНКЦИЙ

R. PÉTER

Резюме

В теории синтеза электрических сетей, на основе данных условий работы, булевые функции, т. е. такие функции, которые могут сами а также их переменные принять только две величины («правильную» и «неправильную») играют важную роль.

Известно, что эти функции могут быть построены из переменных при помощи конъюнкции, дизъюнкции и отрицания (даже и меньшими операциями, но в этом случае менее обозримым способом). Напомним, например, их меченую дизъюнктивную нормальную форму. L. KALMÁR обращает внимание на то, что с точки зрения экономичности важно производить булевые функции при помощи указанных трёх операций насколько возможно кратким способом в тех случаях, где под длиной одной формулы подразумевается число появляющихся в нем переменных, учитывая каждое переменное согласно многократности его наличия. Для возможно короткого произведения, произвольно заданных булевых функций требовалось бы, конечно, по возможности простой альгорифм; и поэтому было бы интересным первым шагом в этом направлении сперва теоретически исследовать, какие (и в каком смысле рекурсивные) альгорифмы можно здесь ожидать.

В этой статье автор доказывает, даже двумя способами, что минимальная длина таких формул, которые производят данную булевую функцию F n-переменных а также определенная формула минимальной длины, могут быть определены как примитивно-рекурсивные функции от F и n в одном подходяще выбранном «множестве слов».