

Aplikace matematiky

Václav Fabian

Blokové logické schéma automatického optimalizátoru

Aplikace matematiky, Vol. 7 (1962), No. 6, 426–440

Persistent URL: <http://dml.cz/dmlcz/102826>

Terms of use:

© Institute of Mathematics AS CR, 1962

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library* <http://dml.cz>

BLOKOVÉ LOGICKÉ SCHÉMA AUTOMATICKÉHO OPTIMALIZÁTORU

VÁCLAV FABIAN

(Došlo dne 1. září 1961.)

Je popsána konstrukce automatického optimalizátoru z jednoduchých logických elementů. Strategii optimalizátoru určuje stochastická aproximační metoda.

1. Shrnutí. Je sestaven algoritmus, realizující aproximační metodu popsanou v [1]. Ze základních elementů popsaných v odst. 8 je pak sestaven centrální blok, který spolu s ostatními bloky, jejichž funkce je velmi jednoduchá a je popsána v odst. 2–5, realizuje odvozený algoritmus. Struktura optimalizátoru nezávisí na počtu optimalizovaných parametrů a jejich počet lze zvýšit přidáním standardních bloků pro nové parametry.

2. Úvod. Předpokládáme, že čtenář zná odstavce 2, 3 a 10 práce [1]: používáme tam zavedených označení, zejména pak těchto:

- k – počet optimalizovaných parametrů,
- $x_n = [x_n^{(1)}, \dots, x_n^{(k)}]$ – aproximace optima v n -tém cyklu,
- $x_{n,j} = [x_{n,j}^{(1)}, \dots, x_{n,j}^{(k)}]$ – podmínka j -tého kroku n -tého cyklu,
- $y_{n,j}$ – odhad kvality příslušející podmínce $x_{n,j}$,
- $t_{n,j}$ – začátek j -tého kroku n -tého cyklu,
- ε – doba potřebná k provedení jistých operací v optimalizátoru, zejména k nastavení vhodného obsahu paměti R ,
- $t_{n,j} + \varepsilon_1$ – začátek sledování kvality,
- $t_{n,j} + \varepsilon_2$ – konec sledování kvality,
- $t_{n,j+1} = t_{n,j} + \varepsilon_2 + \varepsilon$ – konec j -tého kroku; předpokládáme, že $0 < 2\varepsilon < \varepsilon_1 < \varepsilon_2$.

Odhad kvality $y_{n,j}$ je dán vztahem

$$(2.1) \quad y_{n,j} = \int_{t_{n,j} + \varepsilon_1}^{t_{n,j} + \varepsilon_2} y(t) dt,$$

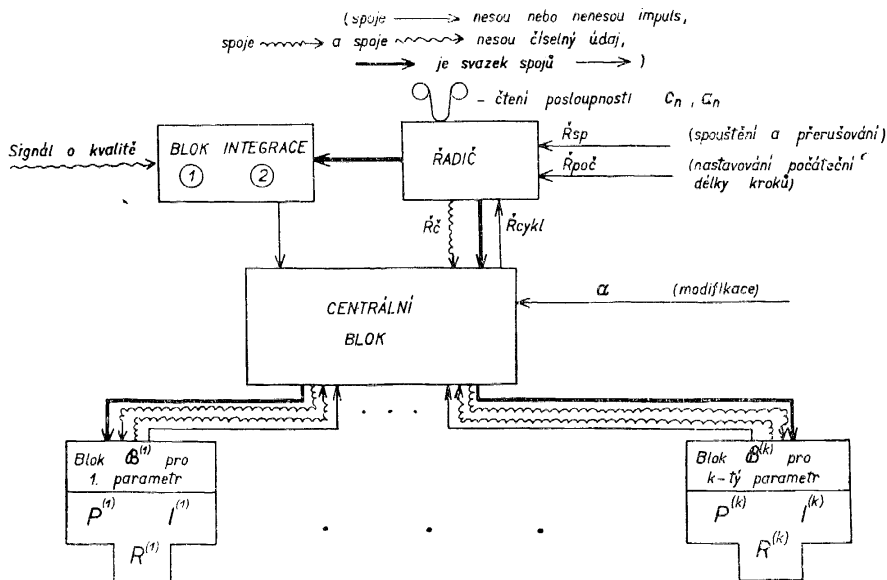
kde $y(t)$ značí odhad kvality v okamžiku t ; signál o kvalitě $y(t)$ je jedinou informací, kterou dostává optimalizátor z optimalizované soustavy.

Výstup optimalizátoru tvoří paměť $R = [R^{(1)}, R^{(2)}, \dots, R^{(k)}]$ ($0 \leq R^{(i)} \leq 1$); její obsah určuje, jak má být v různých časových intervalech určena podmínka optimalizované soustavy; je tedy¹⁾

$$(2.2) \quad R = x_{n,j} \quad \text{pro} \quad t \in \langle t_{n,j} + 2\varepsilon, t_{n,j} + \varepsilon_2 \rangle^2)$$

(viz [1], odst. 3 a 4).

Další vstupy slouží k ovládání optimalizátoru zvenku: ke spouštění, přerušení, k nastavení obsahu paměti P^3), k volbě posloupnosti čísel c_n, a_n , k volbě jedné z modifikací $\alpha = 0, 1$ a případně k volbě, aspoň v jistých mezích, časových konstant $\varepsilon_1, \varepsilon_2$. Hrubé blokové schéma je zakresleno v obrázku 1.



Obr. 1. Schéma automatického optimalizátoru.

Popíšeme nyní zhruba funkci jednotlivých bloků. Řadič čte vloženou posloupnost čísel c_1, a_1, c_2, \dots a dělí čas na cykly, cykly na kroky a v jednotlivých krocích dává signály (do centrálního bloku a do bloku integrace) k započítání jednotlivých operací. Do centrálního bloku také předává čtený člen posloupnosti c_1, a_1, c_2, \dots . V bloku integrace získáváme odhad kvality příslušné jednotlivým krokům. Dvojice integrátorů umožňuje porovnávání kvalit dvou kroků. Informace o kvalitě, která vchází do bloku integrace, je jedinou informací, kterou optimalizátor dostává od optimalizované

¹⁾ Bez nebezpečí nedorozumění značíme v tomto článku stejnými symboly paměti i obsahy těchto pamětí.

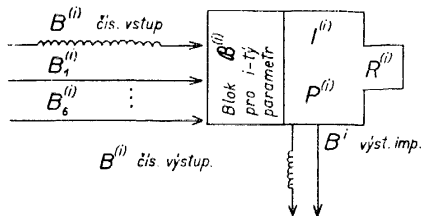
²⁾ V [1] používáme symbolu ε ve smyslu našich 2ε .

³⁾ P je další paměť v optimalizátoru, v níž se uchovávají x_n ; je třeba mít možnost nastavit prvou aproximaci x_1 .

soustavy. Bloky $\mathcal{B}^{(i)}$ jednotlivých proměnných jsou vzájemně zaměnitelné. Každý blok $\mathcal{B}^{(i)}$ má tři paměti $R^{(i)}$, $P^{(i)}$, $I^{(i)}$. Z toho prvé dvě paměti čísel, třetí je paměť impulsů. Obsah paměti $R = [R^{(1)}, R^{(2)}, \dots, R^{(k)}]$ je určen vztahem (2.2) a je roven té podmínce, která je předepsána pro daný krok. V paměti $P = [P^{(1)}, P^{(2)}, \dots, P^{(k)}]$ se uchovává ve zkušebních krocích poslední aproximace x_n optimální podmínky. V pracovních krocích se obsah P postupně mění. Účelem paměti P je uchování hodnoty, z níž lze jednoduše určit obsah R v dalším kroku. Hlavním úkolem paměti $I^{(i)}$ je uchování informace, získané v i -tém zkušebním kroku o tom, zda je výhodné zvětšit či zmenšit i -tou souřadnici podmínky x_n . Mimoto je paměti I využito v 0-tém kroku k nastavení hodnoty $x_{n,0}$. Bloky $\mathcal{B}^{(i)}$ nejsou však jen prostým souhrnem svých pamětí, ale provádějí přesuny obsahů těchto pamětí. Poslední část schématu, centrální blok, přijímá signály ze všech ostatních částí, zpracovává je a ovlivňuje bloky $\mathcal{B}^{(i)}$ tím, že dává příkazy ke změně pamětí těchto bloků. Dává též příkaz řadiči k započetí nového cyklu.

V dalších odstavcích popíšeme podrobněji činnost jednotlivých bloků optimalizátoru. Přitom značíme \longrightarrow prosté spoje, tj. spoje s dvěma možnými stavy 0 a 1. Číselný spoj má více možných stavů: v daném okamžiku buď nese číslo (to je jeden z možných stavů) nebo nese číslo z intervalu $(-1, 1)$. Takovýto spoj je značen \rightsquigarrow . Spoje, které nenesou nikdy záporná čísla, označujeme \rightsquigarrow . Je-li stav prostého spoje 0, říkáme, že tento spoj (v daném okamžiku) nese impuls, a v opačném případě, že nese impuls. Podobné terminologie používáme pro paměti se dvěma možnými stavy, 0 a 1, a nazýváme takové paměti paměti impulsů. Je-li A paměť impulsů nebo prostý spoj, značíme $\hat{A} = 1$ a $\hat{A} = -1$, je-li po řadě $A = 0$ a $A = 1$. Je tedy $\hat{A} = -1$ právě tehdy, je-li na A impuls.

3. Blok $\mathcal{B}^{(i)}$ pro i -tou proměnnou má dvě číselné paměti $P^{(i)}$ a $R^{(i)}$, paměť impulsů $I^{(i)}$ a vstupní a výstupní spoje zakreslené ve schématu s funkcí níže popsanou:



Impuls na spoji

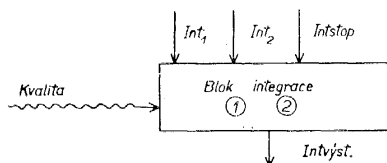
$B_1^{(i)}$
 $B_2^{(i)}$
 $B_3^{(i)}$
 $B_4^{(i)}$
 $B_5^{(i)}$
 $B_6^{(i)}$

způsobí přenesení:

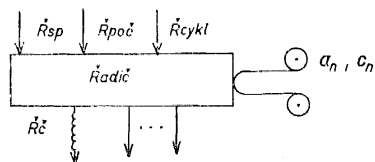
$1 \rightarrow I^{(i)}$,
 $0 \rightarrow I^{(i)}$,
 $R^{(i)} \rightarrow P^{(i)}$,
 $P^{(i)} \rightarrow B^{(i)}$ **čís. výstup**,
 $B^{(i)}$ **čís. vstup** $\rightarrow R^{(i)}$,
 $P^{(i)} \rightarrow R^{(i)}$.

Na výstupu $B^{(i)}$ **výst. imp.** je impuls tehdy a jen tehdy, je-li impuls v $I^{(i)}$; není-li impuls na $B_4^{(i)}$, nenese $B^{(i)}$ **čís. výstup** číslo.

4. Blok integrace obsahuje dvě integrující (kvalitu) zařízení **1** a **2**; impuls na **Int**; maže a spouští integrátor **i**, impuls na **Intstop** zastavuje oba integrátory; na výstup **Intvýst** je impuls tehdy a jen tehdy, je-li impuls na **Intstop** a je-li obsah integrátoru **1** větší než **2**.



5. Řadič a jeho zapojení s blokem integrace. Řadič má 3 vstupní prosté spoje **Řsp**, **Řpoč**, **Řcykl** jeden výstupní číslicový spoj **Řč** a několik dalších prostých výstupních spojů.



Podstatou funkce řadiče je čtení do řadiče vložené (např. na děrné pásce) posloupnosti čísel $c_1, a_1, c_2, a_2, \dots$ a dělení cyklů na kroky.

Popíšeme tuto funkci nejprve za předpokladu, že je trvale impuls na spoji **Řsp** (tj. řadič je spuštěn). V době ε po impulsu na **Řpoč** nebo na **Řcykl** začíná cyklus (v prvním případě prvý, ve druhém případě další); přitom se přenese číslo c_1 resp. další dosud nečtené číslo c_n na spoj **Řč** a zůstává zde až do konce k -tého kroku. Během cyklu impulsy na výstupních prostých spojích provádějí dělení běžícího cyklu na kroky časové délky $\varepsilon_2 + \varepsilon$. Signalisuje tedy řadič okamžiky $t_{n,j}$ pro $j = 0, 1, 2, \dots, n$ -tého cyklu; v časovém intervalu $(t_{n,k} + \varepsilon_2, t_{n,k+1})$ dochází ke změně stavu **Řč** z c_n na a_n .

Při odnětí impulsu z **Řsp** ustane činnost řadiče a při jeho opětném přivedení začíná ten cyklus, během kterého došlo k přerušení.

Výstupních prostých spojů je využito k signalizaci začátku různých operací v optimalizátoru; tyto všechny signály lze odvodit známými způsoby ze stavů jediného spoje udávajícího impulsem začátek kroku; můžeme tedy předpokládat, že máme potřebné spoje již na výstupu z řadiče. Označme symbolem (t_j^s) pro $j = 0, 1, \dots, s = 0, 1, 2$ spoj, který nese impuls právě v okamžiku

$$\begin{aligned} t_{n,j} & \text{ pro } s = 0, \\ t_{n,j} + \varepsilon_1 & \text{ pro } s = 1, \\ t_{n,j} + \varepsilon_2 & \text{ pro } s = 2, \end{aligned}$$

n -tého cyklu a to pro každé $n = 1, 2, \dots$

Předpokládáme, že tedy máme k dispozici tyto spoje:

$$\begin{aligned} (t_j^0) & \text{ pro } j = 0, 1, \dots, k + 1, \\ (t_j^2) & \text{ pro } j = 0, 1, \dots, k + 1, \end{aligned}$$

dále spoje, které názorně značíme

$$\left(\bigvee_{j \cong 1} t_{k+2j}^2 \right) \text{ resp. } \left(\bigvee_{j \cong 1} t_{k+2j+1}^2 \right) \text{ resp. } \left(\bigvee_{j \cong 1} t_{k+j}^0 \right)$$

a které mají impulsy v okamžicích

$$t_{n,k+2} + \varepsilon_2, t_{n,k+4} + \varepsilon_2, \dots,$$

resp.

$$t_{n,k+3} + \varepsilon_2, t_{n,k+5} + \varepsilon_2, \dots,$$

resp.

$$t_{n,k+1}, t_{n,k+2}, \dots$$

n -tého cyklu pro každé n .

Další spoj **tpřer** má impuls v okamžiku, kdy se stav **Řsp** mění z 1 na 0.

Konečně poslední tři spoje jsou určeny k připojení na vstup do bloku integrace

Int₁ má impuls v okamžiku $t_{n,i} + \varepsilon_1$ pro $i = 0, k + 1, k + 3, \dots$

Int₂ má impuls v okamžiku $t_{n,i} + \varepsilon_1$ pro $i = 1, 2, \dots, k, k + 2, k + 4, \dots$

Intstop = $\left(\bigvee_{j \cong 0} t_j^2 \right)$ má impuls v okamžiku $t_{n,j} + \varepsilon_2$ pro $j = 0, 1, 2, \dots$

Tyto spoje spojíme se vstupními spoji **Int₁**, **Int₂**, **Intstop** integračního bloku. Pak zřejmě v okamžicích $t_{n,j} + \varepsilon_2$ se v bloku integrace zaznamenává kvalita $x_{n,j}$ a to v integrátoru **1** pro $j = 0, k + 1, k + 3, \dots$ a v integrátoru **2** pro $j = 1, 2, \dots, k, k + 2, k + 4, \dots$. Odtud a z odst. 4 plyne, že na výstupním spoji **Intvýst** může být impuls pouze v okamžicích $t_{n,j} + \varepsilon_2$ cyklu n pro $n = 1, 2, \dots$. Přitom je pro $j \neq 0$ a $j \neq k + 1$ v okamžiku $t_{n,j} + \varepsilon_2$ impuls na **Intvýst** právě tehdy, když buď

$$(5.1) \quad j = 1, 2, \dots, k \quad \text{a} \quad y_{n,0} > y_{n,j},$$

nebo

$$(5.2) \quad j = k + 2, k + 4, \dots \quad \text{a} \quad y_{n,j-1} > y_{n,j},$$

nebo

$$(5.3) \quad j = k + 3, k + 5, \dots \quad \text{a} \quad y_{n,j-1} < y_{n,j}.$$

6. Funkce centrálního bloku. Úlohou centrálního bloku je (i) vyhodnocovat signál přicházející z bloku integrace a podle něho určit směr v n -tém cyklu a určit konec cyklu a (ii) provádět změny pamětí R, P, I tak, aby bylo zaručeno splnění podmínky (2.2). Příklad detailnější funkce centrálního bloku popíšeme v dalším textu:

(6.1) Stav paměti I (určení směru). Z důvodů, které se projeví později, je výhodné uložit impulsy do paměti I na začátku každého cyklu, tj. provést operaci

$$!(6.1.1) \quad - \mathbf{1} \rightarrow \hat{I} \text{ v okamžiku } t_{n,0}$$

(kde $-\mathbf{1} = \underbrace{[-1, -1, \dots, -1]}_{kx}$). Hlavní účel paměti I však spočívá v tom, že se do ní

během kroků prvního až k -tého uloží a až do konce cyklu uchová informace o směru. Při spojitém měření kvality můžeme vyloučit případ $y_{n,j} = y_{n,0}$ a jestliže tedy chceme, aby bylo

$$(6.1.2) \quad \hat{I} = \Delta_n \text{ v časovém intervalu } \langle t_{n,k}, t_{n+1,0} \rangle$$

stačí podle (5.1) a podle [1], (3.3.2) provést operace

$$!(6.1.3) \quad \text{Intvýst} \rightarrow I^{(j)} \text{ v okamžiku } t_{n,j} + \varepsilon_2 \text{ pro } j = 1, 2, \dots, k.$$

(6.2) Začátek nového cyklu.

K přerušení cyklu dochází (a po době ε k začátku nového cyklu):

(i) při modifikaci $\alpha = 0$

v okamžiku $t_{n,k+1} + \varepsilon_0$

(ii) při modifikaci $\alpha = 1$

v okamžiku $t_{n,k+i} + \varepsilon_2$, je-li $i \geq 2$ a $y_{n,k+i-1} > y_{n,k+i}$.

Začátek cyklu se zaručuje tedy v n -tém cyklu touto operací (viz (5.2), (5.3); protože tuto operaci sloučíme později s jinou, značíme ji $x(\dots)$):

$x(6.2.3)$ impuls \rightarrow **Řcykl!** v okamžiku t , jestliže je buď

$$(6.2.3.1) \quad t = t_{n,k+1} + \varepsilon_0, \alpha = 0^4)$$

nebo

$$(6.2.3.2) \quad t = t_{n,k+2j} + \varepsilon_2, \text{Intvýst} = 1, j \geq 1,$$

nebo

$$(6.2.3.3) \quad t = t_{n,k+2j+1} + \varepsilon_2, \text{Intvýst} = 0, j \geq 1.$$

(6.3) Stav paměti P, R :

Na začátku n -tého cyklu je $P = x_n$. To je zaručeno závěrem $n - 1$. cyklu, je-li $n > 1$ (viz níže); v případě $n = 1$ je třeba, aby byla prvá aproximace x_1 vnějším zásahem do P nastavena⁵); podmínku $x_{n,0}$ pak do R nastavíme v 0-tém kroku operací

$$x(6.3.1) \quad P \oplus \text{Řč} \times \hat{I} \rightarrow R \text{ v okamžiku } t_{n,0} + \varepsilon;$$

(U operací, v nichž se vyskytuje sčítání, je udaný okamžik začátkem operace, která je skončena v době ne delší než ε .)

K významu operace: protože podle !(6.1.1) v čase $t_{n,0} + \varepsilon$ je $\hat{I} = -\mathbf{1}$, $\text{Řč} = c_n$, $P = x_n$, je po provedení operace $R^{(i)} = x_n^{(i)} \ominus c_n = x_{n,0}^{(i)}$ (viz [1], (3.3.1)) pro každé $i = 1, 2, \dots, k$.

⁴) $\varepsilon_0 < \varepsilon$.

⁵) Mechanismus tohoto nastavení bude záviset na realizaci paměti, zde se jím nezabýváme.

Úpravu R v krocích $1, 2, \dots, k$ zaručují tyto dvě operace (viz [1], (3.3.1))

$$!(6.3.2) \quad P^{(j)} \oplus \check{R}\check{c} \rightarrow R^{(j)} \text{ v okamžiku } t_{n,j} \text{ pro } j = 1, 2, \dots, k$$

$$!(6.3.3) \quad P^{(j)} \ominus \check{R}\check{c} \rightarrow R^{(j)} \text{ v okamžiku } t_{n,j} + e_2 \text{ pro } j = 1, 2, \dots, k^6)$$

(je opět $\check{R}\check{c} = c_n$).

V dalších krocích vždy na začátku provedeme operaci

$$x(6.3.4) \quad P \oplus \check{R}\check{c} \times \check{I} \rightarrow R \text{ v okamžiku } t_{n,j} \text{ pro } j = k+1, k+2, \dots;$$

její význam souvisí s operací

$$x(6.3.5) \quad R \rightarrow P \text{ v okamžiku } t,$$

jestliže je buď

$$(6.3.5.1) \quad t = t_{n,k+1} + e_0, \alpha = 0,$$

nebo

$$(6.3.5.2) \quad t = t_{n,k+2j} + e_2, j \geq 1 \text{ a } \text{Intv}\check{y}\check{s}t = 0$$

nebo

$$(6.3.5.3) \quad t = t_{n,k+2j+1} + e_2, j \geq 1 \text{ a } \text{Intv}\check{y}\check{s}t = 1.$$

Jestliže totiž je modifikace $\alpha = 0$, pak cyklus končí podle $x(6.2.3)$ v okamžiku $t_{n,k+1} + e_0$, kdy je provedena poprvé operace $x(6.3.1)$, která nastavuje do R podmínku $x_n \oplus a_n \Delta_n$ (jednak viz (6.1.2), za druhé je v krocích $k+1$ a dalších $\check{R}\check{c} = a_n$). V okamžiku $t_{n,k+1} + e_0$ však se tato podmínka pošle do P a tím se nastaví v P $(n+1)$ -vá aproximace (viz [1], (3.4.1)) operací $x(6.3.5)$ a začne (po době ε) nový cyklus operací $x(6.2.3)$.

Jestliže je modifikace $\alpha = 1$, pak ve všech krocích $(k+j)$ s $j \geq 1$, s výjimkou posledního, v němž dojde k přerušení cyklu (tj. k operaci $x(6.2.3)$), se operací $x(6.3.4)$ na začátku kroku vkládá podmínka $x_{n,k+j}$ do R a po době e_2 (kdy se zjistí, že cyklus ještě nekončí) se vkládá $x_{n,k+j}$ operací $x(6.3.5)$ též do P . Právě proto, že v P je pak $x_{n,k+j}$ lze v $(j+1)$ -vém kroku vytvořit $x_{n,k+j+1}$ pouhým přičtením $a_n \Delta_n$, které provádí operace $x(6.3.4)$. Je-li však krok $k+j$ poslední v daném cyklu, pak je $j \geq 2$ (viz $x(6.2.3)$) a $x_{n,k+j}$ se do P již neukládá, takže na začátku dalšího cyklu je v P podmínka $x_{n,k+j-1} = x_{n+1}$.

Tak jsme ukázali, že uvedené operace zaručují splnění podmínky (2.2) i výchozího předpokladu $P = x_n$ v okamžiku $t_{n,0}$.

Všimněme si, že v P je stále uložena podmínka s významem poslední aproximace optima, kdežto v R se nastavují podmínky, které umožňují např. hledat směr.

S tím souvisí poslední operace

$$!(6.3.6) \quad P \rightarrow R, \text{ je-li impuls na } \text{tpřer},$$

kteřá po dobu přerušení optimalisace předpisuje pro podmínku procesu poslední získanou aproximaci optima. Tato aproximace zůstává v P a při novém spuštění je výchozí podmínkou dalšího cyklu.

⁶⁾ Zde bychom mohli psát $j = 1, 2, \dots, k-1$; provedení operace pro $j = k$ není nezbytné.

Tabulka 1. Přehled operací centrálního bloku.

Nové označení	Operace	Okamžik začátku operace	Odpovídá operacím
7.1	$1 \rightarrow I$	$t_{n,0}$! (6.1.1)
7.2	$\text{Intvýst} \rightarrow I^{(j)}$	$t_{n,j} + \varepsilon_2, j = 1, 2, \dots, k$! (6.1.3)
7.3	je-li $t = t_{n,k+1} + \varepsilon_0, \alpha = 0$, pak $R \rightarrow P, 1 \rightarrow \mathbf{\check{R}cykl}$; je-li $t = t_{n,k+2j} + \varepsilon_2, j \geq 1$, pak (je-li $\text{Intvýst} = 1$, pak $1 \rightarrow \mathbf{\check{R}cykl}$, jinak $R \rightarrow P$); je-li $t = t_{n,k+2j+1} + \varepsilon_2, j \geq 1$, pak (je-li $\text{Intvýst} = 1$, pak $R \rightarrow P$, jinak $1 \rightarrow \mathbf{\check{R}cykl}$);	t (splňující podmínku uvedenou vlevo)	$x(6.2.3)$ a $x(6.3.5)$
7.4	$P \oplus \mathbf{\check{R}\check{e}} \times I \rightarrow R$	$t_{n,0} + \varepsilon, t_{n,j}$ pro $j = k + 1, k + 2, \dots$	$x(6.3.1)$ a $x(6.3.4)$
7.5	$P^{(j)} \oplus \mathbf{\check{R}\check{e}} \rightarrow R^{(j)}$	$t_{n,j}, j = 1, 2, \dots, k$! (6.3.2)
7.6	$P^{(j)} \ominus \mathbf{\check{R}\check{e}} \rightarrow R^{(j)}$	$t_{n,j} + \varepsilon_2, j = 1, 2, \dots, k$! (6.3.3)
7.7	$P \rightarrow R$	je-li impuls na $\mathbf{tpřer}$! (6.3.6)

Tabulka 2. Rozdělení operací podle času.

Řádka	Krok j	Začátek operace $z_0 + f^0 u_j$ $z_1 + f^1 u_j$ $z_2 + f^2 u_j$ $z_3 + f^3 u_j$	Viz operací	Popis operace	Poznámka o významu operace
1	0	●	7.1	$1 \rightarrow I$	pomocná operace
2		●	7.4	$P \oplus \check{R}\check{e} \times \check{I} \rightarrow R$	$x_{n,0} \rightarrow R$
3	1, 2, ..., k	●	7.5	$P^{(j)} \oplus \check{R}\check{e} \rightarrow R^{(j)}$	$x_{n,j} \rightarrow R$
4		●	7.2	$Intv\check{y}st \rightarrow I^{(j)}$	zapamatování směru
5		●	7.6	$P^{(j)} \ominus \check{R}\check{e}^{(j)} \rightarrow R$	$x_{n,0} \rightarrow R$
6	$k + 1, k + 3, \dots$	●	7.4	$P \oplus \check{R}\check{e} \times \check{I} \rightarrow R$	$P \oplus a_n \times \check{I} \rightarrow R$
7		●	7.3	je-li $\alpha = 0$, pak $R \rightarrow P$, $1 \rightarrow \check{R}cykl$	skončení cyklu při $\alpha = 0$
8	$k + 3, k + 5, \dots$	●	7.3	je-li $Intv\check{y}st = 1$, pak $1 \rightarrow \check{R}cykl$	skončení cyklu
9		●	7.3	je-li $Intv\check{y}st = 0$, pak $R \rightarrow P$	pokračování v cyklu dalším pracovním krokem
10	$k + 2, k + 4, \dots$	●	7.4	$P \oplus \check{R}\check{e} \times \check{I} \rightarrow R$	$P \oplus a_n \times \check{I} \rightarrow R$
11		●	7.3	je-li $Intv\check{y}st = 1$, pak $R \rightarrow P$	pokračování cyklu
		●	7.3	je-li $Intv\check{y}st = 0$, pak $1 \rightarrow \check{R}cykl$	skončení cyklu

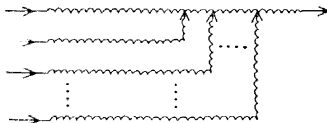
7. Přehled operací centrálního bloku. V předešlém odst. jsme jednotlivé operace uvedli v takovém sledu a rozčlenění, aby byla patrna jejich funkce. V tabulce 1 je uvádíme v seskupení vhodnějším z hlediska konstrukce, v tabulce 2 uvádíme jejich časový sled. V tabulce 2 není zahrnuta operace (7.7). Připomeňme, že současně řadič řídí jednoduchým způsobem činnost bloku integrace: V každém kroku se provádí integrace v časovém intervalu $(t_{n,j} + \varepsilon_1, t_{n,j} + \varepsilon_2)$ na jednom z obou integrátorů s porovnáním obsahů obou integrátorů na konci tohoto intervalu. Přitom se integruje v krocích $0, k + 1, k + 3, \dots$ na prvním integrátoru a v ostatních krocích na prvém.

V dalších dvou odstavcích uvedeme příklad rozkreslení centrálního bloku v bloky s jednodušší funkcí; k tomu v dalším odstavci zavedeme označení pro některé základní logické elementy.

8. Označení základních elementů.

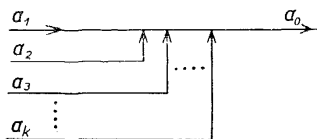
(8.1) Spoje. Jako výše značí \longrightarrow prostý a \rightsquigarrow respektive \rightsquigarrow číselný spoj, nesoucí nebo nenesoucí číslo z intervalu $\langle 0, 1 \rangle$ respektive $\langle -1, 1 \rangle$.

Schématu

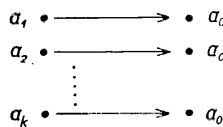


použijeme v situacích, kdy nanejvýš jeden ze vstupních spojů ponese číslo; výstupní spoj pak nese číslo, jestliže žádný ze vstupních spojů nese číslo, a v případě, že právě jeden ze vstupních spojů nese číslo, nese je i výstupní spoj.

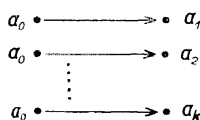
Ve schématu



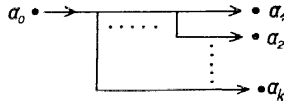
je na výstupním spoji impuls tehdy a jen tehdy, je-li impuls aspoň na jednom vstupním symbolu. Abychom se vyhnuli nepřehlednostem při např. značném množství jednotlivých vstupních spojů bloků a výstupních spojů řadiče, kreslíme někdy předešlé schéma takto:



Podobně schéma

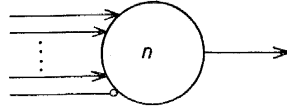


bychom mohli jinak zakreslit takto:



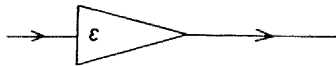
Uvedené způsoby kreslení schémat nám umožní zakreslit schéma centrálního bloku z jednoduchých samostatných dílů.

(8.2) Základní logický prvek.



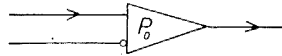
V daném okamžiku je na výstupu impuls tehdy a jen tehdy, je-li impuls na aspoň n vstupních budících spojích \longrightarrow a není-li na tlumícím spoji \longleftarrow .

(8.3) Zpoždovač.



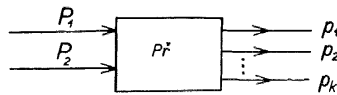
Na výstupu je v čase t impuls tehdy a jen tehdy, byl-li v čase $t - \epsilon$ na vstupu.

(8.4) Paměť impulsů. (Ve schématu je uvedeno P_0 místo PJ .)



Na výstupu je impuls tehdy a jen tehdy, jestliže v časové posloupnosti impulsů na vstupních spojích byl poslední impuls na spoji budícím a nikoliv na spoji tlumícím (tj. budící spoj vkládá a tlumící maže impuls v paměti).

(8.5) Přepínač.



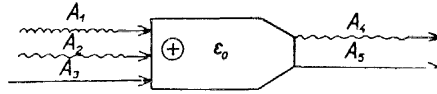
Při posloupnosti impulsů na $P_1, P_2, P_2, \dots, P_2$ v okamžicích $t_1 < t_2 < \dots < t_s$ jsou impulsy na p_i v okamžicích t_i pro $i = 1, 2, \dots, \min(s, k)$.

(8.6) Změna znaménka.



Je-li na vstupu číslo x , je na výstupu $\begin{cases} x, & \text{není-li} \\ -x, & \text{je-li} \end{cases}$ na vstupním prostém spoji impuls.

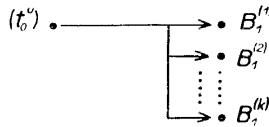
(8.3) Sčítačka.



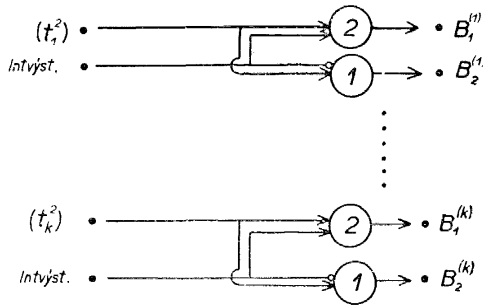
Přivedeme-li v čase t impuls na A_3 a čísla a_1 a a_2 na A_1 a A_2 , bude nejpozději v čase $t + \varepsilon_0$ (předpokládáme $\varepsilon_0 < \varepsilon/(2k)$) impuls na A_5 a spoj A_4 ponese součet $a_1 \oplus a_2$.

9. Schéma centrálního bloku. Budou popsána a zakreslena dílčí schémata; jejich zakreslením vedle sebe je dáno schéma centrálního bloku. Vstupní a výstupní spoje ostatních částí (řadiče, bloků \mathcal{B}_i) jsou značeny body s připojeným názvem, a jsou použity i vícekrát v soulase s úmlouvou z odst. (8.1).

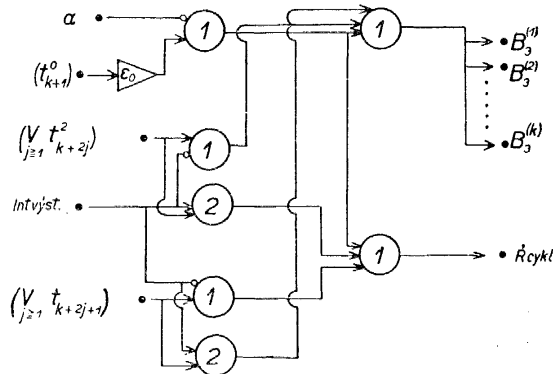
(9.1) Blok realizující operaci (7.1):



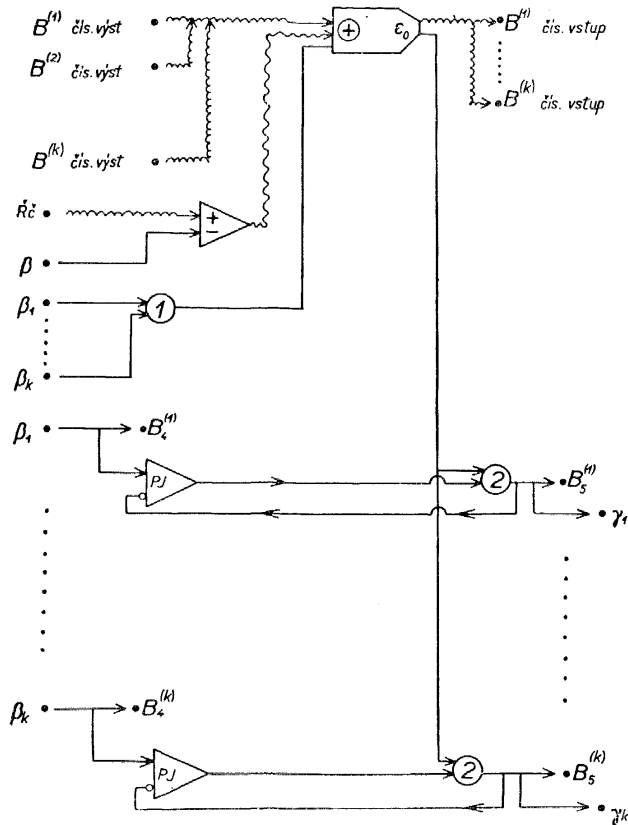
(9.2) Blok realizující operaci (7.2):



(9.3) Blok realizující operaci (7.3):



(9.4) Blok realizující operaci (7.4), (7.5) a (7.6): Prvá část tohoto bloku

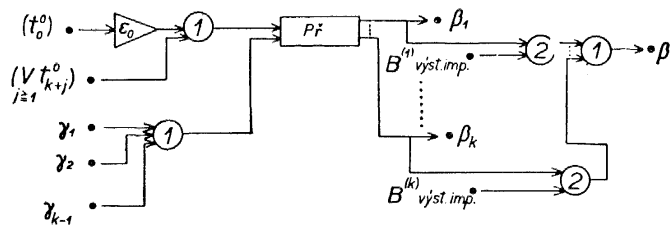


provádí operaci

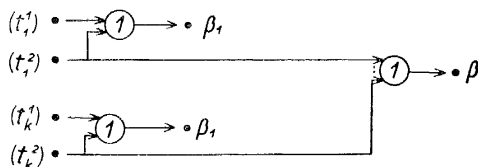
$$(9.4.1) \quad P^{(i)} + \check{R}\check{c} \times \hat{\beta} \rightarrow R^{(i)} \text{ a } 1 \rightarrow \gamma_i \text{ při impulsu na } \beta_i.$$

Impuls na β_i je přenesen na $B_4^{(i)}$ a tím přeneše $P^{(i)}$ do sčítačky, kde se sečte s $\hat{\beta} \times \check{R}\check{c}$. Při skončení sečítání impuls ze sčítačky se přeneše na $B_5^{(i)}$ a $\gamma^{(i)}$, čímž způsobí uložení součtu na $R^{(i)}$ a impulsu na γ_i . V této poslední fázi určení indexu i je umožněno impulsem uchovaným v jedné z k pamětí impulsů, jež se po provedení operace ihned maže.

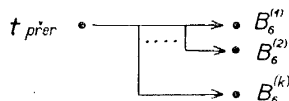
Přidáním dalšího zapojení dostaneme blok realizující operaci (7.4):



Konečně přidáním dalšího zapojení dostaneme blok, realizující operace (7.4), (7.5), (7.6) (druhý ze symbolů β_1 či β_k):



(9.5) Blok realizující operaci (7.7):



10. Poznámky: Z předešlého popisu funkce jednotlivých částí optimalizátoru je patrné, že bloky $\mathcal{B}^{(i)}$ a blok integrace nezávisí na počtu k parametrů. Dále je patrné, že optimalizátoru pro optimalizaci k parametrů můžeme použít přímo nebo po mechanickém odstranění $k - k_1$, bloků $\mathcal{B}^{(k_1+1)}, \dots, \mathcal{B}^{(k)}$ pro optimalizaci k_1 parametrů. V tomto případě bude však časový interval $(t_{n,k_1+1}, t_{n,k+1})$ nevyužit. Je však snadné konstruovat řadič s volitelným k , čímž tato nevýhoda odpadne. V centrálním bloku není třeba při přechodu k menšímu k činit žádné úpravy.

Při vhodné volbě fyzikální reprezentace, vstupních a výstupních signálů bude mít optimalizátor obecné použití ve třídě praktických situací vyžadujících aspoň řádově stejné délky kroků.

Literatura

- [1] *Václav Fabian*: Stochastická aproximační metoda pro hledání optimálních podmínek v experimentální práci a v samoadaptivních systémech, *Aplikace matematiky*, 6 (1961), 162–183.

Резюме

ЛОГИЧЕСКОЕ БЛОК-СХЕМА АВТОМАТИЧЕСКОГО ОПТИМИЗАТОРА

ВАЦЛАВ ФАБИАН (Václav Fabian)

Описано логическое блок-схема оптимизатора, стратегия которого была автором предложена в работе [1]. Схема очень проста, также в случае большого числа переменных.

Summary

LOGICAL BLOCK NETWORK OF AN AUTOMATIC OPTIMISER

VÁCLAV FABIAN

The author specified the function of an automatic optimiser in [1]; the present paper describes its logical block network. The structure of the optimiser is independent of the number of conditions optimised; this number may be increased by adding standard blocks for new parameters.

Adresa autora: Dr. *Václav Fabian* C.Sc., Výzkumný ústav matematických strojů, Loretánské nám. 3, Praha 1.