Graphic Matrix Formalization of Logical Decision Trees in the Optimization of Machine Systems

A. Deptuła and M.A. Partyka

Abstract Multi-valued logical decision trees indicate the importance rank of construction and/or exploitation parameters. There is an isomorphic interpretation of logical transformations, thus the Quine−Mc Cluskey algorithm of the minimization of individual multi-valued logical functions can be considered by taking into account the graphic matrix formalization in the optimization of machine systems.

Keywords Matrix formalization · Multi-valued decision trees · Quine–McCluskey algorithm · Importance rank of decision parameters · Optimization

1 Introduction

The method of minimization of complex, partial, multi-valued logical functions indicates the importance rank of construction and/or exploitation parameters playing the role of logical decision variables [1–3].

Fluid-flow machines form a vast group of sets used in industry [3, 4]. Decision tables and logical functions [1, 4, 5] can be applied in the issues of modelling machine systems with differential equations (ordinary and partial ones). It results from the fact that non-linear elements can be divided into a finite number of linear elements (parts) which leads to getting several linear systems. Discrete optimization of fluid-flow machines is based on indicating the degree of importance of construction and exploitation parameters. Guidelines concerning the sequence of making decisions result from multi-valued decision trees by taking into consideration the realisation of the assumed purpose function (e.g. the system stabilisation).

All transformations concern the so-called Quine–McCluskey algorithm of the minimization of individual partial multi-valued logical functions. There is an iso-

A. Deptuła (✉) · M.A. Partyka
Opole University of Technology, Opole, Poland
e-mail: a.deptula@po.opole.pl

M.A. Partyka
e-mail: m.partyka@po.opole.pl

S. Zawiślak and J. Rysiński (eds.), *Graph-Based Modelling in Engineering*,
Mechanisms and Machine Science 42, DOI 10.1007/978-3-319-39020-8_15

morphic interpretation of logical transformations, thus the Quine–McCluskey algorithm of the minimization of individual partial multi-valued logical functions can be considered by taking into account the graphic matrix formalization. An application of the matrix formalization in case of a bigger number of decision variables (as construction and/or exploitation parameters) makes it possible to solve practical geometric problems in order to extract the most and the least important data [6].

2 Graphic Matrix Formalization of Logical Decision Trees

A logical tree is a structural presentation of a logical function, written in the form of a sum of products, where every element is the realisation of one solution and each component in the product is a logic variable [2]. Complex partial multi-valued logical functions state the degree of importance of logic variables, by means of changing the logical tree levels, from the most important ones (near the root) to the least important (in the upper part) because there is a generalisation of a bivalent indicator of quality into a multi-valued one; $(C_k - k_i m_i) + (k_i + K_i)$, where C_k—the number of branches k-th level, k_i—times simplification on the k-th level m_i—valued variable, K_i—number of branches of $(k - 1)$-th level, from which branches of k-th level which cannot be simplified were created. All transformations are described by the so-called Quine–McCluskey algorithm of minimization of individual partial multi-valued logical functions.

2.1 Example

For a multiple-valued logical function $f(x_1, x_2, x_3)$, where $x_1, x_2, x_3 = 0, 1, 2$, written by means of numbers KAPN (Canonical Alternative Normal Form): 100, 010, 002, 020, 101, 110, 021, 102, 210, 111, 201, 120, 022, 112, 211, 121, 212, 221, 122, there is one MZAPN (Minimal Complex Alternative Normal Form) after the application of the Quine–McCluskey algorithm based on the minimization of individual partial multi-valued logical functions having 13 literals [2]:

$$\begin{aligned} f(x_1,x_2,x_3) = {} & j_o(x_1)(j_o(x_2)j_2(x_3) + j_1(x_2)j_o(x_3) \ + j_2(x_2)) \\ & + j_1(x_1) + j_2(x_1)(j_o(x_2)j_1(x_3) + j_1(x_2) \ + j_2(x_2)j_1(x_3)). \end{aligned} \tag{1}$$

Figure 1 shows MAPN of a given multi-valued logical function.

In the isomorphic interpretation of the Quine–McCluskey algorithm three steps are taken for the graphic matrix formalization:

(a) putting decision $(m_1, \ldots, m_n)$-valued variables $x_1, \ldots, x_n$ in a certain order; creating the n! primary matrices, relative to all combinations of variables,

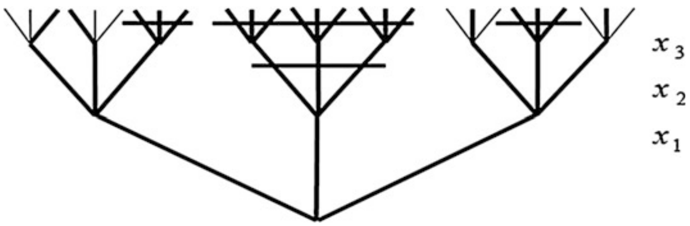

Fig. 1 A multi-valued decision tree for the parameters x_1, x_2, x_3 with an appropriate layout of levels

(b) prioritising the numbers relative to $(m_1, \ldots, m_n)$ the valence in the increasing order from the left side of the matrix,
(c) combining numbers and removing them (minimization).

Figure 2 shows the matrix formalization for the tree from Fig. 1.

3 Application of the Graphic Matrix Formalization in the Optimization of Machine Systems

Logical functions are taken into consideration in the tasks concerning modelling of machine sets described among others by means of differential equations (ordinary and partial ones). If a machine system is described by means of a set of functions f_1, f_2, f_3,…, f_n, dependent on time t, the change of construction parameters x_1,…,x_n impli-es a (good or bad) change in the plot of such functions. It is admissible to decrease, increase or leave the figures of construction parameters unchanged in the optimization process.

3.1 Analysis of a Degree of Importance of Construction Parameters of the Overflow Valve

The overflow valve is applied in systems in order to let excess fluid flow to the container where the pump efficacy is bigger than the need. An example of a drive system of an actuator with an overflow valve is presented in Fig. 3.

The equation of forces acting on the closing component of a valve is presented in the following way [6]:

$$\frac{Q_p^2}{A_1}\rho + \rho \cdot A_2 + \rho \cdot l\frac{dQ_p}{dt} = G_{ap} + S + k \cdot x + f\frac{dx}{dt} + m\frac{d^2x}{dt^2} + \Phi\sqrt{2 \cdot \rho} \cdot \cos(v) \cdot Q_p\sqrt{p} \tag{2}$$

(a)

(x_3)	j_2	j_0	~~j_0~~	~~j_1~~	~~j_2~~	~~j_0~~	~~j_1~~	~~j_2~~	~~j_0~~	~~j_1~~	~~j_2~~	~~j_0~~	~~j_1~~	~~j_2~~	j_1	~~j_0~~	~~j_1~~	~~j_2~~	j_1
(x_2)	j_0	j_1	j_2			~~j_0~~			~~j_1~~			~~j_2~~			j_0	j_1			j_2
(x_1)	j_0					j_1									j_2				

(b)

(x_3)	j_2	j_0													j_1				j_1
(x_2)	j_0	j_1	j_2												j_0	j_1			j_2
(x_1)	j_0					j_1									j_2				

Fig. 2 Matrix formalization for the decision tree from Fig. 1: **a** prioritising and combining numbers, **b** removing numbers from the table

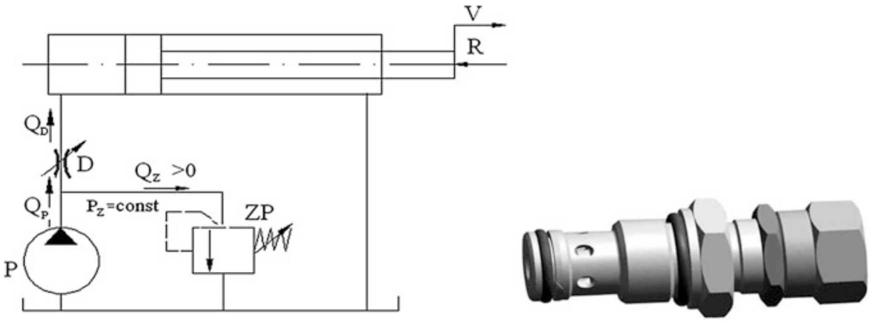

Fig. 3 A scheme of an actuator and a view of the valve

while equations of flows have the following form:

$$Q = \mu \cdot K \cdot x\sqrt{p} + A_1 \frac{dx}{dt} + \frac{V}{B}\frac{dp}{dt} \tag{3}$$

$$Q_p = \mu \cdot K \cdot x\sqrt{p} + A_1 \frac{dx}{dt} \tag{4}$$

where:

$$K = \pi \cdot d_m \sqrt{\frac{2}{\rho}} \tag{5}$$

Equations of the valve work in a dimensionless form used to make simulation are presented in the following way:

$$\rho \frac{Q_o^2}{A_1 S_o} Q_{pw}^2 + \frac{A_2 p_o}{S_o} p_w + \frac{T_{Qp}}{T_o} \frac{dQ_{pw}}{dt_w} = 1 + \frac{k x_o}{S_o} x_w + \frac{T_f}{T_o} \frac{dx}{dt_w} + \left(\frac{T_{ms}}{T_o}\right)^2 \frac{d^2 x}{dt_w^2} + \Phi \frac{\sqrt{2\rho}}{S_o} \cos(v) Q_o Q_{pw} \sqrt{p_o} \sqrt{p_w} \tag{6}$$

$$Q_w = \mu x \sqrt{p_w} + \frac{T_A}{T_o} \frac{dx}{dt_w} + \frac{dp_w}{dt_w} \tag{7}$$

$$Q_{pw} = \mu x \sqrt{p_w} + \frac{T_A}{T_o} \frac{dx}{dt_w}. \tag{8}$$

In order to make a discrete optimization, changes in parameters have been coded in the following way: 0- large decrease, 1- small decrease, 2- without changes, 3- increase, 4- large increase (for *m* and *k*) and : 0- small decrease, 1- without changes, 2- increase (for *d*). Depending on the adopted combinations of code changes in parameters *m, k* and *d* in canonical products, the behaviour of functions that depend on time is different.

When we look at the behaviour of the functions *x*, *Q* and *p* at the time of stabilisation $t_w < 300\ t_0$, we see 20 charts that have been chosen for which code changes of construction parameters m, k and d are presented (Table 1).

Figure 4 shows exemplary charts of the functions *x*, *Q* and *p* for the code changes of parameters (*m, k* and *d*) 332.

Table 1 KAPN (Canonical Alternative Normal Form) for given changes of parameters *m, k* and *d* ($t_w < 300\ t_0$)

m k d	*m k d*	*m k d*	*m k d*
2 2 2	1 1 2	0 2 2	0 0 2
2 1 2	1 3 2	0 3 2	3 2 2
1 2 1	2 0 2	0 1 1	3 1 2
1 2 2	1 0 2	0 1 2	0 3 1
1 1 1	0 2 1	0 0 1	0 1 0

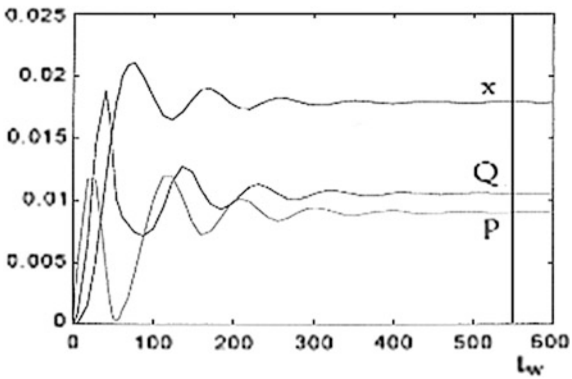

Fig. 4 Characteristics of the valve work for changes in the code of parameters *m, k* and *d*: 332

<table>
<tr><td>k</td><td>1</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td>1</td><td>2</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td>0</td><td>1</td><td>2</td><td>1</td><td>2</td></tr>
<tr><td>m</td><td>0</td><td colspan="4">0</td><td colspan="2">1</td><td colspan="4">0</td><td colspan="4">1</td><td colspan="3">2</td><td colspan="2"></td></tr>
<tr><td>d</td><td>0</td><td colspan="6">1</td><td colspan="13">2</td></tr>
<tr><td>m</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>2</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td>0</td><td>1</td></tr>
<tr><td>k</td><td>1</td><td>0</td><td colspan="2">1</td><td colspan="2">2</td><td>3</td><td colspan="3">0</td><td colspan="4">1</td><td colspan="4">2</td><td colspan="2">3</td></tr>
<tr><td>d</td><td>0</td><td colspan="6">1</td><td colspan="13">2</td></tr>
<tr><td>k</td><td>1</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td>1</td><td>2</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td>1</td><td>2</td><td>3</td><td>1</td><td>2</td></tr>
<tr><td>d</td><td>0</td><td colspan="4">1</td><td colspan="4">2</td><td colspan="2">1</td><td colspan="4">2</td><td colspan="3">2</td><td colspan="2">2</td></tr>
<tr><td>m</td><td colspan="9">0</td><td colspan="6">1</td><td colspan="3">2</td><td colspan="2">3</td></tr>
<tr><td>m</td><td>0</td><td>0</td><td>1</td><td>2</td><td>0</td><td>0</td><td>1</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td>0</td><td>1</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td><s>3</s></td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>d</td><td>1</td><td colspan="3">2</td><td>0</td><td colspan="2">1</td><td colspan="4">2</td><td colspan="2">1</td><td colspan="4">2</td><td>1</td><td colspan="2">2</td></tr>
<tr><td>k</td><td colspan="4">0</td><td colspan="7">1</td><td colspan="6">2</td><td colspan="3">3</td></tr>
<tr><td>d</td><td>1</td><td>2</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td>1</td><td>2</td><td>1</td><td>2</td><td>2</td><td>1</td><td>2</td><td>1</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td></tr>
<tr><td>k</td><td colspan="2">0</td><td colspan="3">1</td><td colspan="2">2</td><td colspan="2">3</td><td>0</td><td colspan="2">1</td><td colspan="2">2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>1</td><td>2</td></tr>
<tr><td>m</td><td colspan="9">0</td><td colspan="6">1</td><td colspan="3">2</td><td colspan="2">2</td></tr>
<tr><td>d</td><td>1</td><td>2</td><td>2</td><td>2</td><td><s>0</s></td><td><s>1</s></td><td><s>2</s></td><td>1</td><td>2</td><td>2</td><td>2</td><td>1</td><td>2</td><td>1</td><td>2</td><td>2</td><td>2</td><td>1</td><td>2</td><td>2</td></tr>
<tr><td>m</td><td colspan="2">0</td><td>1</td><td>2</td><td colspan="3">0</td><td colspan="2">1</td><td>2</td><td>3</td><td colspan="2">0</td><td colspan="2">1</td><td>2</td><td>3</td><td colspan="2">0</td><td>1</td></tr>
<tr><td>k</td><td colspan="4">0</td><td colspan="7">1</td><td colspan="6">2</td><td colspan="3">3</td></tr>
</table>

Fig. 5 Matrix formalization with prioritising and combining numbers from Table 1

Each of the *canonical alternative normal form* products for appropriate changes in the value of parameters *m, k, d* (from Table 1), has 6 combinations of the matrix formalization assigned [7, 8]. Having prioritised and combined numbers, we obtain a real importance rank of parameters *m, k* and *d*.

Figure 5 presents a matrix formalization with prioritising and combining numbers from Table 1.

Having considered combining numbers from Table 1, we obtain the minimum form of the graphic matrix formalization presented in Fig. 6.

The minimum graphic form of the matrix formalization from Fig. 6 indicates the importance rank of construction and/or exploitation parameters from the most important ones at the bottom to the least important ones at the top.

<table>
<tr><td>k</td><td>1</td><td></td><td></td><td></td><td></td><td>1</td><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>0</td><td>1</td><td>2</td><td>1</td><td>2</td></tr>
<tr><td>m</td><td>0</td><td colspan="4">0</td><td colspan="2">1</td><td colspan="4">0</td><td colspan="4">1</td><td colspan="3">2</td><td colspan="2">3</td></tr>
<tr><td>d</td><td>0</td><td colspan="6">1</td><td colspan="13">2</td></tr>
</table>

Fig. 6 The minimum form of the graphic matrix formalization from Fig. 5

Table 2 KAPN (Canonical Alternative Normal Form) for given changes of parameters *m, k* and *d* ($t_w < 1000\ t_0$)

m k d	*m k d*	*m k d*	*m k d*	*m k d*
2 2 2	3 3 2	0 1 1	0 3 1	2 1 1
2 1 2	1 2 1	0 1 2	0 1 0	2 0 1
1 2 2	2 0 2	0 0 1	2 3 2	1 0 1
1 1 1	1 0 2	0 0 2	3 0 2	4 2 2
1 1 2	0 2 1	3 2 2	0 2 0	4 1 2
1 3 2	0 2 2	3 1 2	2 2 1	0 0 0
1 0 0	0 3 2	1 1 0	4 0 2	0 3 0
2 4 2	4 3 2			

<table>
<tr><td>m</td><td>01</td><td>01</td><td>0</td><td>0</td><td>012</td><td>012</td><td>012</td><td>0</td><td>~~01234~~</td><td>~~01234~~</td><td>~~01234~~</td><td>~~01234~~</td><td>2</td></tr>
<tr><td>k</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>~~0~~</td><td>~~1~~</td><td>~~2~~</td><td>~~3~~</td><td>4</td></tr>
<tr><td>d</td><td colspan="4">0</td><td colspan="4">1</td><td colspan="5">2</td></tr>
</table>

Fig. 7 The minimum graphic form of the matrix formalization from Table 2

When we look at the behaviour of the functions *x*, *Q* and *p* at the time of stabilisation $t_w < 100\ t_0$, 20 charts have been chosen for which code changes of construction parameters *m, k* and *d* are presented (Table 2).

Figure 7 presents the minimum form of the matrix formalization for Table 2 indicting the rank of importance of parameters *m, k* and *d*.

Figure 8 presents an optimal logical tree equivalent to the minimum form of the matrix formalization from Fig. 7.

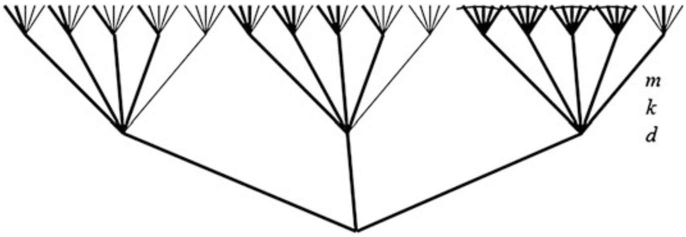

Fig. 8 An optimal multi-valued logical tree of parameters *d, k, m* as an equivalent of the minimum matrix formalization

4 Conclusions

Modelling examinations aim at rotating important parameters in order to ensure the stability of a real set. In the model verification it is important to state the rank of importance of construction and/or exploitation parameters [9, 10]. The paper presents an isomorphic interpretation of logical transformations of a Quine–McCluskey algorithm of minimizing individual partial multi-valued logical functions. Using graphic matrix formalization determined a degree of importance of construction parameters of the overflow valve. The complexity of logic tables or truth tables *grow*s exponentially in relation to the number of variables. In case of a bigger number of decision variables, there are practical geometric problems in order to extract the least and the most important data. What is more, the graphic matrix formalization can be a computer record of parametric game trees as an adjacency matrix [5].

References

1. Deptuła, A.: Application of multi-valued weighting logical functions in the analysis of a degree of importance of construction parameters on the example of hydraulic valves. Int. J. Appl. Mech. Eng. **19**(3), 539–548 (2014)
2. Partyka, M.A.: Algorytm Quine'a- McCluskeya minimalizacji indywidualnych cząstkowych wielowartościowych funkcji logicznych, Studia i Monografie Nr 109, Politechnika Opolska – Oficyna Wydawnicza, Opole (1999)
3. Berghofer, S., Reiter, M.: Formalizing the logic-automaton connection. In: Proceedings of the 22nd International Conference on Theorem Proving in Higher Order Logics, LNCS No 5674, 147–163 (2009)
4. Deptuła, A, Osiński, P., Partyka, M.A.: Discrete optimization of a gear pump after tooth root undercutting by means of multi-valued logic trees. Arch. Civil Mech. Eng. **13**(4), 422–431 (2013)
5. Deptuła, A., Partyka, M.A.: Application of game graphs in optimization of dynamic system structures. Int. J. Appl. Mech. Eng. **15**(3), 647–656 (2010)

6. Deptuła, A., Partyka, M.A.: Rozłączna analiza logiczna wytycznych projektowania z uwzględnieniem niepewności w modelowaniu układów maszynowych, J Transdisciplinary Syst. Sci. **16**(1), 161–173. ISSN 1427-275X (2012)
7. Partyka, M.A.: Optymalizacja dyskretna pompy wirowej śmigłowej w ruchu turbinowym – zastosowanie wielowartościowych drzew logicznych. Napędy i Sterowanie **1** (2004)
8. Pijls, W., de Bruin, A.: Game tree algorithms and solution trees. Theor. Comput. Sci. 252(1-2), February, 197-215 (2001)
9. Shwnoy, P.P.: Game trees for decision analysis Theory Decis **44**(2), 149–171 (1998)
10. Shiue, Y.R., Guh, R.S.: The optimization of attribute selection in decision tree-based production control systems. Int. J. Adv. Manuf. Technol. **28**, 737–746 (2006)