NON-LINEAR NETWORK-FLOW MODEL OF ŁUKASIEWICZ'S MULTIVALUE LOGIC

Vassil Sgurev, Stefan Kojnov

Abstract: The paper presents a new network-flow interpretation of Łukasiewicz's logic based on models with an increased effectiveness. The obtained results show that the presented network-flow models principally may work for multivalue logics with more than three states of the variables i.e. with a finite set of states in the interval from 0 to 1. The described models give the opportunity to formulate various logical functions. If the results from a given model that are contained in the obtained values of the arc flow functions are used as input data for other models then it is possible in Łukasiewicz's logic to interpret successfully other sophisticated logical structures. The obtained models allow a research of Łukasiewicz's logic with specific effective methods of the network-flow programming. It is possible successfully to use the specific peculiarities and the results pertaining to the function 'traffic capacity of the network arcs'. Based on the introduced network-flow approach it is possible to interpret other multivalue logics – of E.Post, of L.Brauer, of Kolmogorov, etc.

Keywords: Łukasiewicz's multivalue logic, operational research, network flow interpretation.

ACM Classification Keywords: F.4.0 Mathematical Logic and Formal Languages – General, F.4.1 Mathematical Logic – Logic and constraint programming.

Conference: The paper is selected from XIV^th International Conference "Knowledge-Dialogue-Solution" KDS 2008, Varna, Bulgaria, June-July 2008

Introduction

Most often the popular publication of J.Łukasiewicz [1] is accepted as a beginning of the multivalue logic.

Together with the classic logical systems since the middle of the past century different models of multivalue logic were an object of significant development [2].

During the second half of the same century a research activity started to use precise quantitative methods from the operational research to describe various logical operations. To achieve this goal most widely were used the methods of mixed integer programming (MIP) [4, 5].

A network-flow interpretation of operations and formulas from the propositional logic was introduced in [7] for decision-making systems. Due to its specific character in series of cases the network-flow methods lead to a greater effectiveness compared to MIP.

An attempt was made for a network-flow interpretation of Łukasiewicz's multivalue logic in [8]. The applied models were nonlinear with a significant degree of sophistication.

The paper presents a new network-flow interpretation of Łukasiewicz's logic based on models with an increased effectiveness.

Non-Linear Network-Flow Model of Łukasiewicz's Multivalue Logic

A new nonlinear network flow will be used that most generally can be defined in the following way [9]. For every $x_k \in X$

$$\sum_{i\in\Gamma_k^+} f_i - \sum_{j\in\Gamma_k^-} f_j = \begin{cases} v_k \ iff \ x_k \in S; \\ 0 \ iff \ x_k \notin \{S \cup T\}; \\ -v_k \ iff \ x_k \in T; \end{cases} \quad (1)$$

$$F_r(f_i, f_j) = f_r, \text{ where } i, j, r \in T; \quad (2)$$

$$0 \le f_i \le 1 \text{ for each arc } u_i \in U; \qquad (3)$$

where $G(X,U)$ is a graph with a set of nodes *X* and a set of arcs *U*; f_i is a network function over the arc $u_i \in U$; *S* and *T* are respective sets of sources and consumers; c_i is the traffic capacity of the arc u_i; Γ_k^+ and Γ_k^- are the sets of indexes for all arcs that are the respective input and output for the node $x_k \in X$; v_k is the flow for the node $x_k \in S \cup T$; *T* is the set of indexes for the equalities (2).

It is assumed that the traffic capacities $\{c_i\}$ are integers and as a rule always equal to 1 while the arc flow functions $\{f_i\}$ may have various nonnegative values in the interval from 0 up to 1.

In the propositional logic of Łukasiewicz the propositions may be in one of three possible states: true, false and neutral, respectively 1, 0 and $\frac{1}{2}$.

The truth tables for disjunction and conjunction in Łukasiewicz's logic numerally have the following appearance:

Table 1 (f_3)

$f_1 \backslash f_2$	**1**	**0**	$\frac{1}{2}$
1	1	1	1
0	1	0	$\frac{1}{2}$
$\frac{1}{2}$	1	$\frac{1}{2}$	$\frac{1}{2}$

Table 2 (f_4)

$f_1 \backslash f_2$	**1**	**0**	$\frac{1}{2}$
1	1	1	$\frac{1}{2}$
0	0	0	0
$\frac{1}{2}$	1	$\frac{1}{2}$	$\frac{1}{2}$

If *A* and *B* are propositions and their respective numerical functions are f_1 and f_2 i.e.

$$A = f_1 \text{ and } B = f_2; \qquad (4)$$

then from the two tables above it follows that for the disjunction $A \vee B$ and the conjunction $A \wedge B$ we may denote respectively

$$f_3 = max(f_1, f_2) \text{ and } f_4 = min(f_1, f_2). \qquad (5)$$

The two logical operations will be interpreted by the following below subgraph and the corresponding to it flow equalities and inequalities:

$$f_1 + f_2 - f_3 - f_4 = 0; \qquad (6)$$

$$k(f_1 - f_2) + f_2 = f_3; \qquad (7)$$

$$f_1 \le f_3; \quad f_2 \le f_3; \qquad (8)$$

$$0 \le f_i \le 1; \; i = 1; 2; \ldots; 4 \text{ and } k = 0 \text{ or } 1 \qquad (9)$$

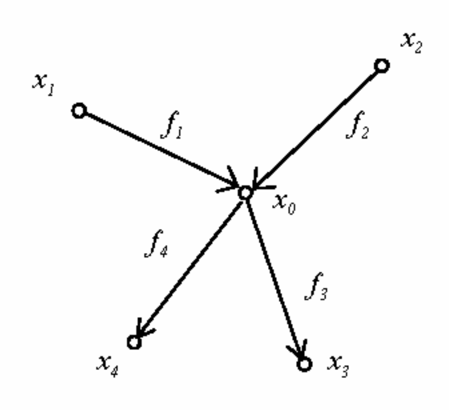

Fig. 1

where (6) is an equality for preservation in node x_0; (7) is a nonlinear dependency corresponding to (2) via which the disjunction from (5) is realized; (8) are additional linear constraints assisting the choice of the greater variable from f_1 or f_2.

The equations (1) for the nodes from x_1 to x_4 do not need to be explained because they are trivial: $v_i = f_i$ for $i \in \{1,2\}$ and $v_j = f_j$ for $j \in \{3,4\}$.

The variable f_3 takes the bigger value from f_1 and f_2: if $k = 1$ then the value is from f_1 and if $k = 0$ then the value is from f_2.

The dependencies from (5) to (9) serve the determination of the variables f_1 and f_2 with the bigger value i.e. these dependencies guarantee the requirements from Table 1.

The conjunction, the second equality from (5) may be realized via the same dependencies from (6) to (9) but the inequalities (8) must be replaced by the new dependencies

$$f_1 \geq f_4 \text{ and } f_2 \geq f_4 \tag{10}$$

assisting the choice of the smaller variable from f_1 or f_2.

On the other hand if the disjunction f_3 from (5) is already determined then it is possible to determine f_4 directly from the equation of preservation (6):

$$f_4 = f_1 + f_2 - f_3; \tag{11}$$

$$f_3 \leq f_1 + f_2 \text{ and } f_4 \leq f_1 + f_2 \tag{12}$$

Therefore the subgraph from Fig. 1 and the dependencies from (6) to (9) are ample factors to determine synonymously the disjunction $A \vee B = f_3$ and the conjunction $A \wedge B = f_4$ in Łukasiewicz's logic.

Negation in the same logic is determined by the functions:

$$\neg A = 1 - f_1 \text{ and } \neg B = 1 - f_2 \tag{13}$$

Implication in the logic of Łukasiewicz satisfies the requirement:

$$f_3 = \begin{cases} 1 \text{ iff } f_1 \leq f_2; \\ 1 - f_1 + f_2 \text{ iff } f_1 > f_2; \end{cases} \tag{14}$$

or otherwise

$$f_3 = min\left[1, \left(1 - f_1 + f_2\right)\right] \tag{15}$$

where

$$A = f_1; \; B = f_2 \text{ and } \left(A \to B\right) = f_3. \tag{16}$$

The truth table for implication by Łukasiewicz is of the following type:

Table 3 (f_3)

$f_1 \backslash f_2$	**1**	**0**	$1/2$
1	1	1	$1/2$
0	1	1	1
$1/2$	1	$1/2$	1

The data from this table may be juxtaposed to the subgraph from Fig. 1 and to the following dependencies for a network flow:

$$k\left(f_1 - f_2\right) + f_2 = f_3; \tag{17}$$

$$f_3 \leq 1 - f_1 + f_2 \text{ and } f_3 \leq 1; \tag{18}$$

with valid constraints (6) and (9).

In this flow the dependencies (17) and (18) guarantee the exact adherence to the equation (14). Here if $f_1 \leq f_2$ then the coefficient *k* has a value of zero and $f_3 = 1$ but in the opposite case *k* is unity and $f_3 = 1 - f_1 + f_2$.

In binary logic there exists an important equipollence:

$$A \to B \equiv \neg A \vee B \tag{19}$$

which as it is evident from Table 3 is not valid for all meanings of *A* and *B*. For example for $A = B = 1/2$ i.e. $f_1 = f_2 = 1/2$ then

$$A \to B = \left(1/2 \to 1/2 \right) = 1 \qquad \text{(from Table 3);}$$

$$\neg A \vee B = \left(1 - 1/2 \right) \vee 1/2 = 1/2 \quad \text{(from Table 2).}$$

An analogous conclusion can be made based on the network-flow realization of the two formulas from the equipollence (19). The first realization is shown via the dependencies (6), (9), (17) and (18) and the second formula from (19) may be interpreted by the disjunction from (6) to (9) where f_1 is replaced by its negation $1 - f_2$. Then

$$1 - f_1 + f_2 - f_3 - f_4 = 0;$$

$$k\left(1 - f_1 - f_2\right) + f_2 = f_3;$$

$$1 - f_1 \le f_3; \quad f_2 \le f_3.$$

The comparison of the last equalities and inequalities with the ones from (6), (17) and (18) shows that we have two different types of implications in the examined three-value logic, i.e. that the equipollence (19) is not valid (or it is not true) for all possible estimates in Łukasiewicz's logic.

The obtained results from (6) to (18) also show that the presented network-flow models principally may work for multivalue logics with more than three states of the variables i.e. with a finite set of states in the interval from 0 to 1.

The described models give the opportunity to formulate various logical functions. If the results from a given model that are contained in the obtained values of the arc flow functions are used as input data for other models then it is possible in Łukasiewicz's logic to interpret successfully other sophisticated logical structures. The obtained models allow a research of Łukasiewicz's logic with specific effective methods of the network-flow programming. It is possible successfully to use the specific peculiarities and the results pertaining to the function 'traffic capacity of the network arcs'.

If we denote the value of some complex formula with f_3 i.e. $F\left(f_1, f_2, \dots, f_k\right) = f_3$ while observing the equalities and the inequalities of the network flow respectively from (6) to (18) and formulating the goal functions $f_3 \to max$ and $f_3 \to min$ then it is possible to determine whether this formula is tautology and also what maximal and minimal values it may accept.

From the computational point of view the nonlinearity of the used models doubtlessly is a source of difficulties. The search of effective linear – precise and approximate network-flow methods and algorithms remains an important problem.

Based on the introduced network-flow approach it is possible to interpret other multivalue logics [2] – of E.Post, of L.Brauer, of Kolmogorov, etc.

Conclusion

The paper presents a new network-flow interpretation of Łukasiewicz's logic based on models with an increased effectiveness.

The obtained results show that the presented network-flow models principally may work for multivalue logics with more than three states of the variables i.e. with a finite set of states in the interval from 0 to 1.

The described models give the opportunity to formulate various logical functions.

If the results from a given model that are contained in the obtained values of the arc flow functions are used as input data for other models then it is possible in Łukasiewicz's logic to interpret successfully other sophisticated logical structures.

The obtained models allow a research of Łukasiewicz's logic with specific effective methods of the network-flow programming. It is possible successfully to use the specific peculiarities and the results pertaining to the function 'traffic capacity of the network arcs'.

Based on the introduced network-flow approach it is possible to interpret other multivalue logics – of E.Post, of L.Brauer, of Kolmogorov, etc.

Bibliography

[1] Cleene, S.C., Mathematical Logic, John Wiley and sons Inc., N.Y., 1967.

[2] Rescher, N., Many-valued Logic, McGrow Hill, N.Y., 1969.

[3] Łucasiewicz, J., Logica tzoiwartosciowa Ruch Filozofiezny, T.V, N.9, Lwow, 1920.

[4] Blair, C.E., R.G.Jeroslow, Some Results and Experiments in Programming Techniques for Propositional Logic, Comp. and Oper.Res., V.13, N.5, 1986, pp. 633-649.

[5] Hooker, J.N., A Quantitative Approach to Logic Inference, Decision Support Systems, N.4, 1988, pp. 45-69.

[6] Ford, L.R., D.R.Falkerson, Flow in Networks, Princeton University Press, 1962.

[7] Sgurev, V.S., Network Flow Approach Logic for Problem Solving, Int.J. on Information Theory and Appl., V.2, N.7, 1995, pp. 3-8.

[8] Nikolova, M., Network Flow Interpretation of Lukasiewicz Logic System, J. of Aut. And Informatique, UAI Publ., N.1, Sofia, 1998.

[9] Sgurev, V.S., Network Flow with General Constraints, BAS Publishing, Sofia, 1991.

Authors' Information

Vassil Sgurev – Academician; Institute of Information Technologies, BAS, Acad. G.Bontchev St., bl.2, P.O.Box: 161, Sofia-1113, Bulgaria; e-mail: sgurev@bas.bg

Stefan Kojnov – Researcher, Institute of Information Technologies, BAS, Acad. G.Bontchev St., bl.29A, P.O. Box: 161, Sofia-1113, Bulgaria; e-mail: slk@iinf.bas.bg