APPLICATION OF GRAPH THEORY IN AUTOMATION

Ivana Zubac, Snjezana Rezic & Marijana Bandic Glavas

This Publication has to be referred as: Zubac, I[vana]; Rezic, S[njezana] & Bandic Glavas, M[arijana] (2016). Application of Graph Theory in Automation, Proceedings of the 27th DAAAM International Symposium, pp.0811-0815, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-08-2, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/27th.daaam.proceedings.117

Abstract

In many ways, engineering has always been dependent on mathematical background. This was usually mathematical analysis. In recent years, one of the areas of discrete mathematics - graph theory has been increasingly used in many fields, such as automation, programming or algorithms. In automatization process directed graphs are used for description and analysis of control system, where nodes represent states and arrows direct edges or transitions between states automaton that manages processes. In fact, as well as in other disciplines, automation for analysis and synthesis of control units are used in achievements of discrete mathematics. In this paper we show how to use graph to simplify the process of automation. We analyze the possibilities of using directed graphs for the purposes of analysis and synthesis of dynamic control system (elevator) with the aim of its simplification or minimization.

Keywords: graph theory; digraph; automation; matrix; Euler digraph

1. Introduction

Many problems in natural, technical and social science can be successfully formulated in terms of graph theory. Today the graph theory is well developed, strongly stimulated by technical and chemical applications. For a general background on graph theory and terminology we refer the reader to the classical book by West D.B.[1][2]. For theory of digraphs not defined here we also recommend [3,4]. In paper [5] it was researched how a bond graph model of a robotic manipulator may be used to create a model-based observer to improve the control of manipulator by using estimated link angular velocities in a conventional proportional and derivative feedback controller.

Graph theory is study of points and lines. In particular, it involves the ways in which sets of points, called vertices, can be connected by lines or arcs, called edges. Graphs in this context differ from more familiar coordinate plots that portray mathematical relations and functions.

1.1. Basic definition of graph and digraph

Graph G consists of a finite nonempty set V of p points (vertex) together with a prescribed set X of q unordered pairs of distinct points of V . Each pair $x = \{u, v\}$ of points in X is a line (edge) of G , and x is said to join u and v . A graph with p points and q lines is called a (p, q) graph.

Digraph D consists of a finite set V of points (vertex) and a collection of ordered pairs of distinct points. Any such pair (u, v) is called an arc or directed line and it is usually denoted by uv . The arc uv goes from u to v and is incident to u and v . We also say that u is adjacent to v and v is adjacent from u . The *outdegree* $od(v)$ of a point v is the number of points adjacent from it, and the *indegree* $id(v)$ is the number adjacent to it.

In this paper all digraphs are finite and may have loops and multiple edges (edges with the same initial and final vertices). Digraph D is simple if D has no loops (edges e with $\delta(e) = (v, v)$) and there is at most one edge from x to y for any $x, y \in V(D)$. Note that oppositely oriented edges between two vertices are allowed in simple digraphs, and such a pair of edges is called a digon. The completedigraph of order t , denoted $\vec{K}_t$, is a digraph with t vertices and a digon between each pair of vertices. A *subgraph* is a subset of a digraph's edges (and associated vertices) that constitutes a digraph.

1.2. Representation graph by matrix

It is customary to represent a graph by means of a diagram and by matrix representation. A graph is completely determined by either its adjacencies or its incidences. This information can be conveniently stated in matrix form. The *adjacency matrix* $A = [a_{ij}]$ of a labelled graph G with p points is the $p \times p$ matrix in which $a_{ij} = 1$ if v_i is adjacent with v_j and $a_{ij} = 0$ otherwise. A second matrix, associated with graph G in which the points and lines are labelled, is incidence to matrix $B = [b_{ij}]$. This $p \times q$ matrix has $b_{ij} = 1$ if v_i and x_j are incident and $b_{ij} = 0$ otherwise.

The *adjacency matrix* $A(D)$ of a digraph D is the $p \times p$ matrix $[a_{ij}]$ with $a_{ij} = 1$ if $v_i v_j$ is an arc of D , and 0 otherwise (Fig.1.). It is often possible to make use of these matrices in order to identify certain properties of a graph. For a digraph, the degree of a vertex is the sum of entries in the column and the row corresponding to its adjacency matrix. The entries in the row corresponding to vertex v represent those edges with v as initial vertex and the entries in the column corresponding to v represent the edges with v as final vertex. The sum of all entries in the adjacency matrix is, of course, the total number of edges in the digraph.

Fig. 1. Example of Digraph D and his adjacency matrix $A(D)$

A binary relation on a set can be represented by a digraph. Let R be a binary relation on a set A , that is R is a subset of $A \times A$. Then a digraph representing R can be constructed as follows:

Let the elements of A be the vertices of the digraph G , and let $\langle x, y \rangle$ be an arc of G from vertex x to vertex y if and only if $\langle x, y \rangle$ is in R .

2. Process of automation

Automation of the process is replacing human labour with machines, not only in terms of strength, but also intellectual work. Technically, an automatic machine (technical system) is formed of three groups of elements:

- the senses (sensors, cameras, microphones, etc.)
- controllers (processors which process information)
- executive elements.

For theory of automation of the process not defined here we also recommend [6][7][8], (Fig.2.). *The management process* in a system, in which one or more input variables, over the legality which characterizes this system, affect other variables as output values. *The process* is quantitative and/or qualitative change depending on the time, and it takes place in nature, society and technology.

Fig. 2. Process of automation

Information is data about a particular phenomenon, concept or event. The holder of information is signal. The signal is changeable and measurable variable at the entry or exit of the system, and can take a variety of physical forms. Classification of signals primarily depends on respective parameters of amplitude and time. We distinguish: continuous, discrete and binary signals. Digital automat is a universal sequential circuit, whose behaviour only depends on the current and previous input of data-events. Working machines can be explained by theory of systems management.

Automat measures state of the process and resolves all its essential condition. On the basis of current and previous events in the process, the optimal action or series of actions can be determined. These actions should result in automat process in the optimal mode. To make this possible, the automat must have sufficient set of actions, in order to compensate any predictable impact of the environment. We say that the process must be controllable, so that the management could work. If the value of output variable depends not only on the current values of the input variable, but also of the past values (total input-output) for the digital system is said to be a sequential system or automatic.

The properties of automata:

- *Define finality (there are a finite number of states, the final memory);*
- *Define discretion (working in discrete time);*
- *Define the digital mode (available digital inputs and outputs);*
- *Define determination (unambiguously perform its function);*
- *Define the specificity (completely - all possible sequences of input events - expects an arbitrary set of inputs; incomplete - only a series of input events are possible);*
- *Define synchronicity (discrete time defined by phased signal).*

3. Solution of automaton problem for elevator

Elevator serves two floors. A button for call the elevator is installed on each floor, and in the cabin there are two buttons for selecting the desired floor. Draw a graph and make a synthesis automaton describing control of the lift while the following should be taken into account

- *The pressure on each button is a special input*
- *An output signal represents work or inaction engine*
- *The state of the automat represents inaction lifts on a floor*

First we need to define the input, output and state automata.

Inputs for automation:

- *IK - button on the first floor*
- *IKL - button in the elevator on the first floor*
- *IIK - button on the second floor*
- *IIKL - button on the elevator on the second floor*

Conditions for automaton:

- *S1 - first floor*
- *S2 - second floor*

Outputs for automaton:

- *MR - the engine is running or Y1*
- *MM - the engine is at rest or Y2*

According to these labels (inputs, outputs and states), we can draw the following graph (Fig. 3). The states represent nodes, and input and outputs arcs.

Fig. 3. Graph of transition state the elevator

Automat shown in previous graph can be displayed in a table format crossing / output (Table 1.).

Inputs/state	S1	S2
IK	S1/MM	S1/MR
IKL	S1/MM	S1/MR
IIK	S2/MR	S2/MM
IIKL	S2/MR	S2/MM

Table 1. Table of crossing/outputs

It may be noted that this is a completely defined automaton. The next step is the minimization of automata. First you need to make coding sheet of input and output (Table 2.). Based on set of coding modes, and using Grey code form transition / exit table is coded (Table 3.).

Inputs	X_1	X_0	States	S	Outputs	Y
P	0	0	S1	0	MM	0
PL	0	1	S2	1	MR	1
D	1	1				
DL	1	0				

Table 2. Table of code

	0	1
00	0/0	0/1
01	0/0	0/1
11	1/1	1/0
10	1/1	1/0

Table 3. The coded transition / output tables

The resulting graph of automata operation clearly shows the connection between all its internal states, and state of operand combination from which it originates, transitions, ie. The flow of information and signals. All this allows us to describe and analyze the entire control system, but also easier implementation and simulation of theoretical and actual system, ie. their synthesis.

Among the classical methods of mathematical descriptions of control systems and grapho-analytical methods there is a correlation of both models. Developed grapho-analytical methods can easily get to a complex mathematical model, which significantly reduces model setting time, synthesis of management and development of the control algorithm.

Def.1. A digraph is connected (or weakly connected) if its underlying undirected graph is connected. A digraph is *strongly connected* if, for every ordered pair of vertices (v,w), there is a directed path from v to w.

Th.1. A digraph which represents automata of elevator is strongly connected digraph.

There are notions of Eulerian digraph and Hamiltonian digraph defined in the obvious way; Eulerian digraph has a closed directed path containing every edge and Hamiltonian digraph has a directed cycle passing through every vertex.

Def.2. A connected digraph is *Eulerian* if and only if the in-degree equals the out-degree for every vertex.

Th.2. A digraph who represents automata is Eulerian digraph.

The problem of immersing a complete digraph on t vertices in a simple digraph [9] was considered in the paper. For Eulerian digraphs, it is shown that such an immersion always exists whenever minimum degree is at least $t(t-1)$, and for $t \leq 4$ minimum degree at least $t-1$ suffices.

Th.3. A digraph which represents automata is transitive.

Recal that digraph D is transitive if, for every pair xy and yz of arcs in D with $x \neq z$, the arc xz is also in D . Transitive digraphs from underlying graph-theoretical model in a number of applications. For example, transitive oriented graphs correspond very naturally to partial orders. Clearly, a strong digraph D is transitive if and only if D is complete. [10]

4. Conclusion

Unlike conventional methods of mathematical modelling and mathematical description of automatic control systems grapho-analytical methods took over the primacy not only in practice but also in research, because they lead to a simpler and understandable description of the system, and thus to its analysis and ultimately synthesis of optimal solutions.

The fact that the edge connecting node A to node B with no requirement for feedback is the reason for application of directed graph in developing automation for management processes, since in description, analysis and synthesis of automation for process management feedback is not always necessary.

5. Acknowledgments

Our paper is a result of research made within the Faculty of Mechanical Engineering and Computing, University of Mostar. Faculty is a sponsor of this paper.

6. References

- [1] West D.B. (2007). Introduction to Graph Theory, Prentice Hall, Inc., 978-0130144003, Upper Saddle River, NJ
- [2] Harary F. (1972). Graph Theory, Addison Wesley Longman Publishing Co., 978-0201410334, Massachusetts
- [3] Bang-Jensen, J.; Gutin, G. (2007). Digraphs Theory, Algorithms and Applications, Springer-Verlag, 978-1-84800-998-1, London
- [4] Garnier R.; Taylor J. (2002). Discrete Mathematics for New Technology, Institute of Physics Publishing, 0 7503 06521, Bristol and Philadelphia
- [5] Rezić, S.; Pehar, S. & Crnokić, B. (2009). Design of a Bond Graph Model for Robot Control, Annals of DAAAM for 2009 & Proceedings of the 20th International DAAAM Symposium, 25-28th November 2009, Vienna, Austria, ISSN 1726-9679, ISBN 978-3-901509-70-4, Katalinic, B. (Ed.), pp. 1887-1888, Published by DAAAM International Vienna, Vienna
- [6] Balach M. (2003). Complete Digital Design, McGraw-Hill, 978-0071737708, New York
- [7] Hopcroft, J. & Ullman, J. (1979). Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 81-7808-347-7, Boston
- [8] <http://www.intechopen.com>, (2009). Automation Control – Theory and Practice, Edited by Rodić, A.D. InTech, Accessed on: 2016-10-03
- [9] DeVos, M.; McDonald, J.; Mohar, B.; & Scheide, D. (2012). Immersing complete digraphs. Europ. J. Combin., No.33, 1294-1302
- [10] Lam C.W.H. Distance transitive digraphs, Discrete Mathematics, Vol. 29, 1980, 265-274