International Journal of Mining Science and Technology 28 (2018) 513–517

Contents lists available at ScienceDirect

International Journal of Mining Science and Technology

journal homepage: www.elsevier.com/locate/ijmst

Automation in detection of recirculation in a booster fan ventilation network

Mahesh Shriwas [*], Felipe Calizaya

Department of Mining Engineering, University of Utah, Salt Lake City 84112, USA

ARTICLE INFO

Article history:
Received 22 December 2016
Received in revised form 11 April 2018
Accepted 12 April 2018
Available online 24 April 2018

Keywords:
Recirculation
Ventilation
Booster fan

ABSTRACT

Recirculation is prohibited in many coal mining countries because of the fear that the re-use of return air would allow the build-up of air contaminants at the workings. The incorrect design and location of a booster fan in any ventilation network can create unsafe condition due to recirculation. The current approach to investigating recirculation using simulation software requires manual effort which becomes tedious in a complex and a large network. An algorithm-based C++ program was designed to detect the recirculation in a booster fan ventilation networks. This program needed an input file prepared from output file generated by any ventilation simulator. This program created an output file for recirculation. This program demonstrated the strong capability to detect the recirculation in a sample network and a coal mine ventilation network. The outcomes of this program were documented in this paper.

1. Introduction

The main function of mine ventilation is to dilute and exhaust hurtful gasses, dusts, heat and humidity [1]. Mine ventilation system must not only provide for the health and safety of underground personal but also operate in compliance with regulatory bodies [2]. The practice of ventilation is continually evolving with new technological advances developed in the mining industry [3]. The mine can be ventilated by the use of either surface main fans alone or in combination with booster fans. A booster fan is an underground fan installed in series with a main surface fan and is used to boost the air pressure of the ventilation passing through it [4]. The use of booster fans underground is a potentially practical alternative to the addition of shafts and surface fans [5]. The proper design and location of booster fans eliminate the safety hazards associated with high main fan pressure. Conversely, the incorrect design of booster fans in any ventilation network can create unsafe conditions due to uncontrolled recirculation. The proper design and utilization of a booster fan derives benefits to reduce the main fan pressure, leakage, and consequently ventilation cost especially with high resistance network due to deep and extensive working area in a mine. An installation of booster fan may create uncontrolled recirculation, if it is not sized and located properly with respect to main fan. Recirculation is prohibited in many coal mining countries because of the fear that the reuse of return air would allow the build-up of air contaminants at the workings. Therefore, it is imperative to check the recirculation while designing and operating booster fan for a ventilation network. This paper deals with an algorithm based C++ program to detect the recirculation in a booster fan ventilation networks. This program was implemented to detect the recirculation in a sample network and a coal mine network.

1.1. Statement of problem

The ventilation simulation programs do not give much information about recirculation for the generated solution (fan pressures). The current approach to investigating recirculation using simulation software requires manual effort, i.e., traversing from node to node in the output network by the simulator while keeping tracks of all the nodes that form recirculation paths. This is an easy task in a simple ventilation network, but it becomes tedious and sometimes very confusing in a complex and large network. Moreover, the presence of multiple recirculation paths makes the manual effort more complex and difficult, especially in separating multiple loops that are connected together. Thus, there is a need for a more efficient program that will detect and quantify the recirculation efficiently, for the generated fan pressures. A program is developed to detect the recirculation(s) in a network.

1.2. Graph terminology

The detection of recirculation in any ventilation network is based on graph theory. Therefore, it is very important to walk

* Corresponding author.
E-mail addresses: maheshshriwas1971@gmail.com (M. Shriwas), felipe.calizaya@utah.edu (F. Calizaya).

https://doi.org/10.1016/j.ijmst.2018.04.009
2095-2686/© 2018 Published by Elsevier B.V. on behalf of China University of Mining & Technology.

through the basic concepts of a graph representation of any network. A graph is a collection of vertices and edges: vertices are simple objects that can have names and properties; an edge is a connection between two vertices [6]. A path from vertex *u* to *v* in a graph is a list of vertices in which successive vertices are connected by edges in the graph. A graph is connected if there is a path from every node to every other node in the graph. A simple path is a path in which no vertex is repeated. A cycle is a path that is simple except that the first and last vertices are the same. A graph with no cycles is a tree. A directed graph is a graph with directed edges. It consists of a nonempty finite set of elements called vertices and a finite set of ordered pairs of distinct vertices called edges. A ventilation network can be recognized as a directed graph. The vertices are termed nodes, and the edges are termed as branches. A directed graph is strongly connected component, if any two vertices are mutually accessible following the directed edge. Strongly connected component consists of either single vertex known as trivial or multiple vertices known as nontrivial. Each such non-trivial strongly components show the path of recirculation [7]. An adjacency matrix is a mathematical representation of the network that shows whether any two nodes are connected or not. For detection of recirculation, an adjacency matrix is prepared. In such matrix, an element shows a "1" when corresponding nodes are connected and a "0" when corresponding nodes are not connected. The depth-first search and linear graph algorithm is widely used in computer science to find the strongly connected components in a directed graph [8]. Another algorithm is also used to find the cycles in a directed graph [9].

2. Recirculation algorithm

In an attempt to implement algorithms to detect the recirculation in a ventilation networks, it is very important to understand the categorization of nodes. Nodes are classified as source, sink, and saddle nodes. A source node is defined that belongs to only outgoing branches. A sink node is defined that belongs to only incoming branches. A saddle node is defined that belongs to both outgoing and incoming branches. This algorithm was coded in C++ to detect the recirculation. Recirculation detection algorithm iteratively identifies source and sink nodes and deletes them. The remaining nodes known as saddle nodes show where recirculation may occur. This program was used to detect the recirculation, and minimize the need for manual effort. The resulting process was found efficient and fast.

The recirculation detection algorithm consists of seven steps.

Step 1: Prepare the adjacency matrix of [N] [N] for the sample network, where N represents the number of nodes in the network.
Step 2: Identify the source and sink nodes and delete the columns and rows associated with these nodes.
Step 3: Update and resize the matrix while preserving the original node numbers by applying a tracking node system.
Step 4: Repeat steps 2 and 3, until all the source and sink nodes are deleted.
Step 5: Check for the residual adjacency matrix. If it is a nonempty matrix, the network contains at least one recirculation cycle. A null matrix shows no recirculation.
Step 6: Check for nontrivial, strongly connected nodes, identifying the multiple connected cycles that may allow recirculation.
Step 7: Display the output list of recirculation cycles.

Based on the above algorithms, a computer program was developed. This program was used to detect and flow recirculation paths in a sample and a coal mine ventilation networks. To this program, an input file is prepared based on branch results obtained by using a standard ventilation network solver (simulator). This input file consists of three columns for each branch: "from", "to", and "flow rate". Flow rates may be positive or negative. For any branch with a negative flow, the nodes are swapped. This procedure is repeated for all the negative flows. The program prepares an adjacency matrix from an input file and generates an output file.

3. Sample network

To demonstrate the capabilities of the recirculation detection programs, a sample network was used. Fig. 1 shows the sample ventilation network. The network includes 65 branches, 45 nodes, one surface main fan, one booster fan, six working areas, one return airway, and two intake airways. The airway resistances and other network parameters are shown in Table 1. Table 2 shows the airflow requirements. The network was solved for the combination of main and booster fan pressure of 3.0 and 2.8 kPa respectively to ensure that recirculation would occur. In this case, the solution satisfied the airflow requirements, but with uncontrolled recirculation.

An input file was prepared for the sample network as discussed above based on the solution for this network. Table 3 shows the details of an input file. Recirculation detection program is executed for this network. Fig. 2 shows a screenshot of the output of the

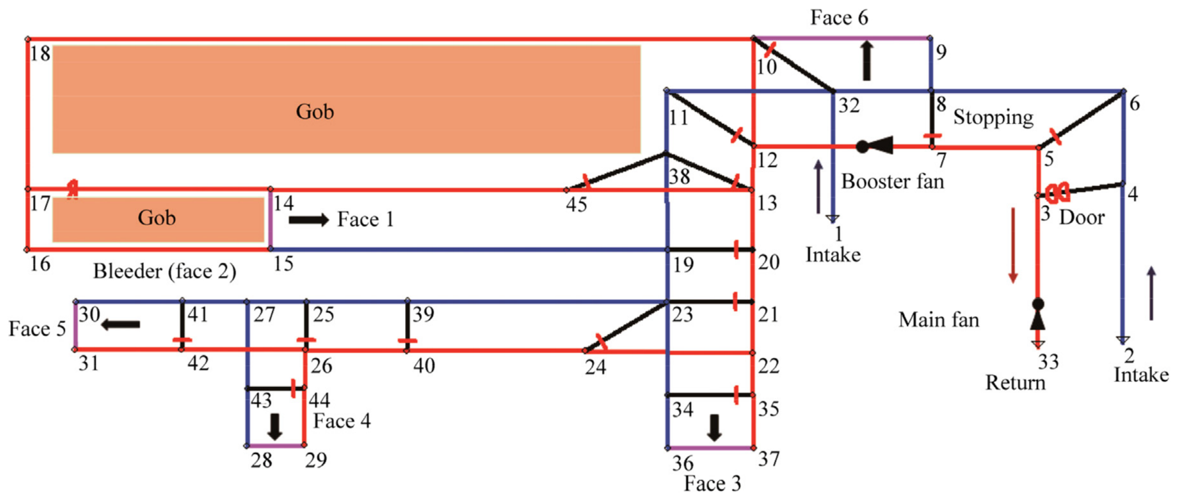


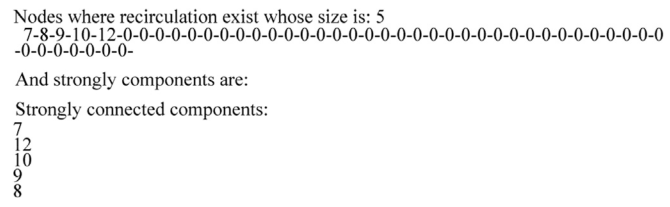
Fig. 1. Sample network.

Table 1
Airway resistance of sample network (Ns^2/m^8).

From	To	R (Ns^2/m^8)	From	To	R (Ns^2/m^8)	From	To	R (Ns^2/m^8)
7	5	0.006	28	29	0.25	34	35	6
6	8	0.006	25	26	6	34	36	0.05
10	12	0.1	19	23	0.0015	37	35	0.01
1	32	0.006	23	21	6	36	37	0.25
3	33	0.006	19	15	0.08	23	34	0.0015
13	12	0.001	8	9	0.08	38	13	6
20	13	0.001	9	10	0.2	11	38	0.001
21	20	0.001	27	41	0.00055	38	19	0.002
22	21	0.001	31	42	0.00481	14	45	0.04821
14	17	5	27	43	0.00093	39	25	0.03011
25	27	0.0015	29	44	0.00084	40	24	0.04512
24	22	0.0015	2	4	0.09	39	40	6
26	40	0.03488	32	11	0.0015	41	30	0.00095
4	3	6	12	7	0.0015	42	26	0.00548
6	5	6	11	12	6	41	42	6
8	7	6	5	3	0.0015	43	28	0.00057
19	20	6	4	6	0.0015	44	26	0.00066
8	32	0.0015	18	10	0.0015	43	44	6
30	31	0.25	17	18	0.0015	45	13	0.01949
16	17	0.0015	23	39	0.04989	38	45	6
15	16	1	23	24	6	32	10	6
15	14	0.25	35	22	0.01			

Table 2
Fixed quantity requirements.

FQ ID	Branch	Airflow requirements (m^3/s)	Working area
1	15–14	47	Face 1
2	15–16	15	Face 2
3	36–37	40	Face 3
4	28–29	33	Face 4
5	30–31	33	Face 5
6	9–10	66	Face 6



```
Nodes where recirculation exist whose size is: 5
 7-8-9-10-12-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0-0
-0-0-0-0-0-0-0-

And strongly components are:

Strongly connected components:
7
12
10
9
8
```

Fig. 2. Screenshot output for the sample network.

recirculation detection program implemented in a sample network.

A quick evaluation of the output shows that the final residual matrix consists of five nodes, namely 7–8–9–10–12 and shows recirculation among them in a sample network. The output also shows that the matrix consists of single strongly connected components. These results were also validated from VnetPC results with manual effort to detect recirculation and were found identical results.

4. Coal mine ventilation network

A coal mine ventilation network used by Calizaya et al. was modified to create another circular path [10]. The network origi-

Table 3
Input file for detection of recirculation.

Br.	From	To	Q (m^3/s)	Br.	From	To	Q (m^3/s)	Br.	From	To	Q (m^3/s)
3	7	5	407.48	26	19	23	7.31	47	37	35	40
4	6	8	61.16	27	23	21	190.74	48	36	37	40
5	10	12	99.81	28	19	15	17.35	49	23	34	57.02
6	1	32	346.32	29	8	9	65.31	50	38	13	18.67
7	3	33	433.82	30	9	10	66	51	11	38	310.79
8	13	12	292.48	31	27	41	66	52	38	19	273.84
9	20	13	208.52	32	31	42	40.09	53	14	45	47
10	21	20	190.74	33	27	43	33	54	39	25	87.58
11	22	21	173.39	34	29	44	40.17	55	40	24	99.26
12	14	17	3.31	35	2	4	33	56	39	40	11.69
13	25	27	80.26	36	32	11	87.5	57	41	30	33
14	24	22	116.37	37	12	7	330.26	58	42	26	40.09
15	26	40	87.58	38	11	12	411.75	59	41	42	7.09
16	4	3	14.03	39	5	3	19.46	60	43	28	33
17	6	5	12.31	40	4	6	419.79	61	44	26	40.17
18	7	8	4.27	41	18	10	73.47	62	43	44	7.17
19	19	20	17.78	42	17	18	18.31	63	45	13	65.29
20	32	8	0.57	43	23	39	18.31	64	38	45	18.29
21	30	31	33	45	23	24	99.26	65	32	10	15.49
22	16	17	15	46	35	22	17.11				
23	15	16	15	47	34	35	57.02				
24	15	14	50.31	45	23	24	17.02				
25	28	29	33	46	34	36	40				

nally consists of 27 nodes and 41 branches with single recirculation path. It has four working areas with fixed quantity of air requirements. In order to include second recirculation path, the airflow direction was arbitrarily reversed in branches (8, 11) and (7, 8). This change in direction could be the result of relocation of a booster fan in a network for the same flow requirements. The branch 27–1 was removed. Fig. 3 shows modified ventilation network. This modified network consists of 27 nodes and 40 branches with two recirculation paths 8–7–10–11 and 12–16–2 6–20–19–21–25–17–13 respectively.

4.1. Detection of recirculation

An input file was prepared for the coal mine ventilation network as discussed above based on the solution for this network. Table 4 shows the details of an input file. The computer program for recirculation detection was executed for this input file. Fig. 4 shows a screenshot of the output of the recirculation detection program. A quick evaluation of the output shows that the final residual matrix consists of five nodes, namely 8–7–10–11 and 12–16–26–20–19–21–25–17–13 respectively and shows recircula-

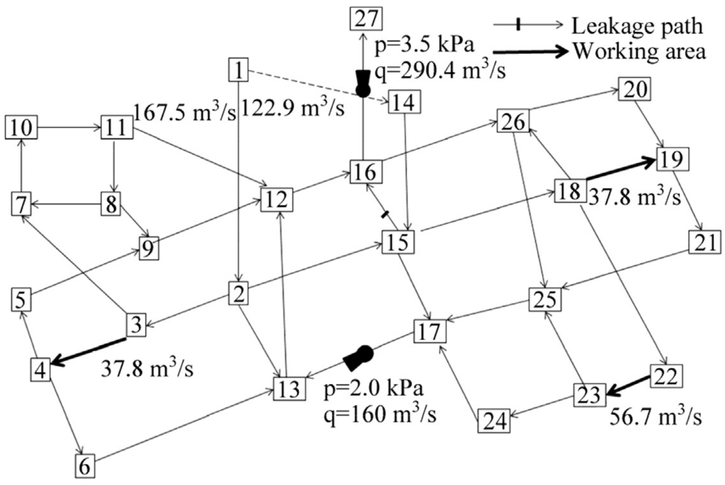
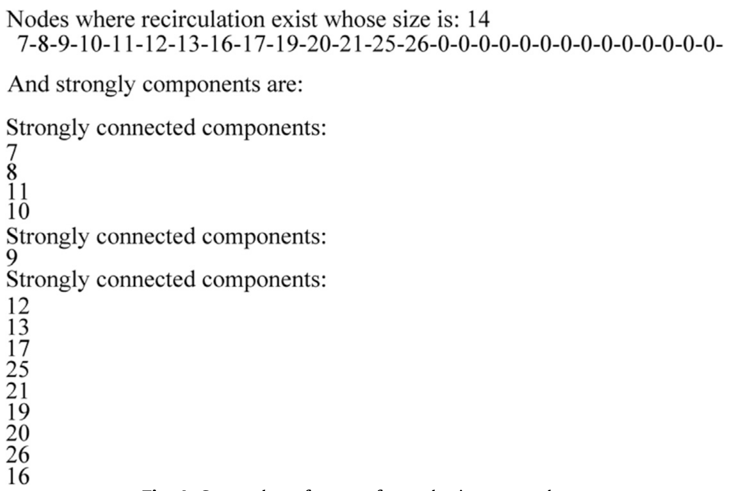


Fig. 3. Modified version of a coal mine ventilation network.

Table 4
Airway resistance of sample network (Ns^2/m^8).

From	To	From	To	From	To	From	To
1	2	8	7	15	16	20	19
2	3	7	10	15	17	21	25
2	13	8	9	15	18	22	23
2	15	11	8	16	27	23	24
3	4	9	12	16	26	23	25
3	7	11	12	17	13	24	17
4	5	10	11	18	19	25	17
4	6	12	16	18	22	26	20
5	9	13	12	18	26	26	25
6	13	14	15	19	21	1	14



```
Nodes where recirculation exist whose size is: 14
 7-8-9-10-11-12-13-16-17-19-20-21-25-26-0-0-0-0-0-0-0-0-0-0-0-0-0-

And strongly components are:

Strongly connected components:
7
8
11
10
Strongly connected components:
9
Strongly connected components:
12
13
17
25
21
19
20
26
16
```

Fig. 4. Screenshot of output for coal mine network.

tion among them in the coal mine ventilation network. These results were validated with manual effort to detect recirculation paths and were found the identical results.

5. Conclusions

An algorithm based computer program written in C++ was implemented successfully to determine the recirculation in a sample and a coal mine ventilation network. The program detected single recirculation path along 7–8–9–10–12 in a sample network. The program also detected two recirculation paths along 8–7–10–11 and 12–16–26–20–19–21–25–17–13 respectively in a coal mine network. This program can be useful while determining the optimal solution for booster fan ventilation network.

Acknowledgements

The authors would like to acknowledge the National Institute for Occupational Safety and Health of USA for providing financial support for this project.

References

[1] Chen KY, Si JH, Zhou FB, Zhang RW, Shao H, Zhao HM. Optimization of air quantity regulation in mine ventilation networks using the improved differential evolution algorithm and critical path method. Int J Min Sci Technol 2015;25(1):79–84.
[2] Euler DS. Application of ventilation management programs for improved mine safety. Int J Min Sci Technol 2017;27(4):647–50.
[3] Wallace K, Prosser B, Stinnette JD. The practice of mine ventilation engineering. Int J Min Sci Technol 2015;25(2):165–9.
[4] Calizaya F, Stephens M, Gillies S. Utilization of booster fans in underground coal mines. In: SME annual meeting and exhibit 2010. p. 443–7.
[5] Wempen J, Calizaya F, Peterson RD, Nelson MG. Evaluating the use of booster fans in two underground coal mines. Min Eng 2011;63(6):115–9.
[6] Sedgewick R. Algorithms in C++. New York: Addison Wesley; 1992.
[7] Acuña EI, Grossman PA, Rubinstein, JH. Application of graph theory algorithms to detect multiple recirculation paths. In: Proceedings of the 14th United States/North American mine ventilation symposium, Salt Lake City, Utah; 2012. p. 65–9.
[8] Tarjan R. Depth-first search and linear graph algorithms. SIAM J Comput 1972;1(2):146–60.
[9] Johnson DB. Finding all the elementary circuits of a directed graph. SIAM J Comput 1975;4(1):77–84.
[10] Calizaya F, Yang G, McPherson MJ, Mousset-Jones P. Application of graph theory to detect uncontrolled recirculation. In: Proceedings of the use of computers in the coal industry 4th conference, Morgantown, WV; 1990. p. 9–109.