The Simulation of Seawater Fire-Fighting System Based On Graph Theory

Yi Xiang-lie, Qiu Jin-shui
College of Naval Architecture and Power
Naval Univ. of Engineering,
wuhan 430033, China

Abstract—**This paper presents an approach to establish model of seawater fire-fighting system based on Graph theory. Follow the steps given in this article, you can easily realize most of the functions of a given seawater fire-fighting system. Firstly, translate a given physical seawater fire-fighting system graph into the target graph in the form of knots and line based on Graph theory; then, by computing relationship of the hydrants and the water pumps, build the related matrix; and the last step, by matrix computing, analyze the system's function, and display all the results in a visualized computer program.**

Keywords-Seawater Fire-Fighting System, Graph theory, VC++.

Ⅰ INTRODUCTION

Seawater fire-fighting system is a fundamental fire-fighting in a ship, especially a warship where the risk of fire and explosion is even bigger. So the study of the structure which includes the arrangement of water pipes and water pumps and hydrants is of great importance, but there is no such simulation software for this kind of simulation, this has brought great difficulty to this kind of study. To solve this problem, establish a quick and easy model for the seawater fire-fighting system is vital.

This article introduced a very quick and easy way to set up a visualized program for the seawater fire-fighting system, Using Graph Theory based method to build a model for the targeted seawater fire-fighting system. Firstly, analyzing the system graph, then translating it into graph in the forms of knots and lines, according to the Graph Theory, and then establishing the related matrix which can reflect all the characteristic of the graph based on Graph Theory, all the practical operating to the seawater fire-fighting system like starting a water pump or shut down a valve even when part of the system damaged by some destructive reasons can turn into the computation of the related matrix, and the last thing is turning all the computation into C++ code in VC++ programming environment which can display all computation results in a visualized mode.

Ⅱ ASSUMPTIONS

1.Seawater is a kind of ideal liquid, which means it has all the ideal liquid features: density and velocity does not change as the time pass by, flown line and track superpose each other, and fluid has potential energy, so Bernoulli equation is suitable for the kinetics calculations of seawater.

2.The difference of the water pressure between two positions which has different height can be neglected, because compare to the water pressure the difference is a tiny thing, and also the simulation is only a very simple one, so it does not necessary to be that precise.

3.The sections of the main pipe of the seawater fire-fighting system have the same size and the sections of the side pipe of the seawater fire-fighting system also have the same size.

4.The friction between the water and the pipe can be neglected.

5.The pressure losing over a siphon also can be neglected.

The seawater fire-fighting system is composed by water pump and pipe and fire hydrant and valves, based on the Bernoulli equation[1]:

$$P+\rho V^2/2+\rho\cdot g\cdot h=C$$

according to the Bernoulli equation，we can get two conclusions:

1.Any chosen point, as long as is connected to the pump in the system and the valves between is open, the pipe is not damaged, the pressure are the same;

2.The flux through any chosen section, as long as which is connected to the pump in the system and the valves between is open and the pipe is not damaged, is direct proportion to the area of the section.

Ⅲ BUILDING PROCESS:

To build the simulation model, we should analyze the seawater fire-fighting system graph following several steps given blow:

1.Tag each element of the seawater fire-fighting system

2. Translate the system graph into graph in Graph Theory. Take the system in Fig.1 as an example, firstly we can find all the knots as shown in table.1,then we draw lines between each connected knots as shown in fig.2.

3.Reading the graph. Translate the graph into related matrix. when knot i connect directly with knot j, then $e_{ij}=1$, when knot i does not connect directly with knot j, then $e_{ij}=0$, E is the so-called related matrix .To Fig.2, the related matrix is A[2]:

Published by Atlantis Press, Paris, France.

$$A = \left\{\begin{matrix}
1&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
&1&0&0&0&0&0&0&0&0&0&0&0&1&0\\
&&1&0&0&0&0&0&0&0&0&0&0&1&0\\
&&&1&0&0&0&0&0&0&0&0&0&1&0\\
&&&&1&0&0&0&0&0&0&0&0&1&0\\
&&&&&1&0&0&0&0&0&0&0&1&0\\
&&&&&&1&0&0&0&0&0&0&1&0\\
&&&&&&&1&0&0&0&0&0&1&0\\
&&&&&&&&1&0&0&0&0&1&0\\
&&&&&&&&&1&0&0&0&1&1\\
&&&&&&&&&&1&0&0&0&1\\
&&&&&&&&&&&1&0&0&1\\
&&&&&&&&&&&&1&1&0\\
&&&&&&&&&&&&&1&0\\
&&&&&&&&&&&&&&1
\end{matrix}\right\}$$

4.Simulation of operation seawater fire-fighting system in the related matrix. As to water pump, there are only two statuses: on/off, damaged pump also can treat as off status. If the pump is in the off status, and the pump knot is knot i, make all elements in line i and row i of matrix A become 0, if the pump is in the on status, keep all elements in line i and row i of matrix A stay the same. As to the valve, there are only two statuses: on/off, if the valve is in off status, and the valve knot is knot j, make all elements in line j and row j of matrix A become 0, if the valve in the on status, keep all elements in line j and row j of matrix A stay the same. As to hydrant, there are three statuses: on/ off/ damaged. If the hydrant is in on status, we should calculate it's flux, if the hydrant is in off status, we should calculate it's pressure, and if the hydrant is in damaged status, there are two states: leaking or just can't turn it on, if the hydrant is leaking, then we should shut down all the valves that connected directly with the hydrant, and change the elements of matrix A as stated before as a result of the valves turn-offs, but if the hydrant is just can't turned on, we can ignore it. As to the pipe, there are two statuses: intact/damaged. If the pipe in the intact status, the pipe knot is knot k keep all elements in line k and row k of matrix A stay the same, if the pipe is in the damaged status, then we should turn off all valves connected directly with the pipe and change the elements of matrix A as stated before as a result of the valves turn-offs. Take Fig.1 as an example: turn on pump1, all hydrants are in the off status, all pipes are in the intact status, shut down valve2/3/4, turn on valve1, then the related matrix A should turn into B.

$$B = \left\{\begin{matrix}
1&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
&1&0&0&0&0&0&0&0&0&0&0&0&1&0\\
&&1&0&0&0&0&0&0&0&0&0&0&1&0\\
&&&1&0&0&0&0&0&0&0&0&0&0&0\\
&&&&1&0&0&0&0&0&0&0&0&1&0\\
&&&&&1&0&0&0&0&0&0&0&0&0\\
&&&&&&1&0&0&0&0&0&0&1&0\\
&&&&&&&1&0&0&0&0&0&1&0\\
&&&&&&&&1&0&0&0&0&1&0\\
&&&&&&&&&1&0&0&0&0&0\\
&&&&&&&&&&1&0&0&0&1\\
&&&&&&&&&&&1&0&0&1\\
&&&&&&&&&&&&1&1&0\\
&&&&&&&&&&&&&1&0\\
&&&&&&&&&&&&&&1
\end{matrix}\right\}$$

5.Algorithm of the related matrix. According to the Graph Theory, if knot i connected with knot j, and knot j connected with knot k, then knot i connected with knot k.

$$C_{i,k} = 1, \mathit{when} \begin{cases} B_{i,j} = 1 \\ B_{j,k} = 1 \end{cases}$$

So, to matrix B, after this step of computation, it turns into matrix C:

$$C = \left\{\begin{matrix}
1&1&1&0&1&0&1&1&1&0&0&0&1&1&0\\
&1&1&0&1&0&1&1&1&0&0&0&1&1&0\\
&&1&0&1&0&1&1&1&0&0&0&1&1&0\\
&&&1&0&0&0&0&0&0&0&0&0&0&0\\
&&&&1&0&1&1&1&0&0&0&1&1&0\\
&&&&&1&0&0&0&0&0&0&0&0&0\\
&&&&&&1&1&1&0&0&0&1&1&0\\
&&&&&&&1&1&0&0&0&1&1&0\\
&&&&&&&&1&0&0&0&1&1&0\\
&&&&&&&&&1&0&0&0&0&0\\
&&&&&&&&&&1&1&0&0&1\\
&&&&&&&&&&&1&0&0&1\\
&&&&&&&&&&&&1&1&0\\
&&&&&&&&&&&&&1&0\\
&&&&&&&&&&&&&&1
\end{matrix}\right\}$$

6.Pressure and flux computation on each hydrant. If the hydrant is in the off status, we calculate its pressure. If knot i stands for the pump, knot j, if $C_{i,k} = 1$, the pressure on the hydrant is equal to the pressure on the pump, else the pressure on the hydrant is zero. As to the flux, if the hydrant is in the on status, the number of pumps is n, each of them has respective flux, we assume as Q_i,and each of them has respective number of hydrants which connect with each pump, we assume as N_i, the pump can provide same flux to each hydrant, the flux of each hydrant is $Q_i^n = \frac{Q_i}{N_i}$, considering the flux of a hydrant, so the flux of the hydrant is sum of each flux of each pump which the hydrant connect with, in formula $Q = \sum_{i=1}^{n} \frac{Q_i}{N_i}$.

After all these steps have been done, the whole model of the seawater fire-fighting system has been built; the next thing is to express all the steps in programming language C++.

Ⅳ BUILDING PLATFORM IN VISUAL C++

In the programming process, the key point is matrix computation. For the sake of concision, this part will be introduced briefly compare to the above paragraphs.

Firstly we should define global variables: extern void A [ ] [ ], A stands for the related matrix[3].

Define four classes: class pump valve pipe hydrant. According the classes defined and the elements of the given system, we define the variables of each class: pump p1 p2 p3 etc. valve v1 v2 v3 etc. pipe pi1 pi2 pi3 etc. hydrant h1 h2 h3 etc.

Define two functions: function simulates damage. The function simulates is responsible for matrix computation, and the function damage is for the damage assumption, which can do any assumed damage to the seawater fire-fighting system, and display the consequence of the damaged system. The final simulation platform is shown in Fig.3[4][5]:

V SUMMARIES

Now, we can draw some conclusions about the process of the model building of the seawater fire-fighting system. This kind of simulation is obviously very simple and efficient simulation, if our engineer only wants to design or test a seawater fire-fighting system's structure and the ability of if basic functions; maybe it's a good way. But if our engineer wants to find the details of the system, like the precise flux and pressure of each elements of the system, this method of simulation can't satisfy their demands.

REFERENCES

[1] ZHANG Zhao-shun; CUI Gui-xiang. Hydro-dynamics (Second Edition). Press QingHua Univ. 2002, PP: 75-77.

[2] XIE Zheng; DAI Li. Combinatorial Graph Theory [M].Press National Defense Technology Univ. 2003, PP: 1-13.

[3] David J.Kruglinski. Technology Inside Visual C++ (Fourth Edition) [M]. Press QingHua Univ. 1999, PP:156-159.

[4] NWC-TN-4565-3-71:Vol1 and 2. MAGIC Computer Simulation: Vol1. User Manual: Vol2. Analyst Manual[S]. Armament Systems Inc. and Propulsion Development Department, 1971.

[5] Cudey Donald E; James B Flint. Target Description Computer Programs, Techniques for Developing Input Data[R]. AD379263L, 1967, PP: 3-13.

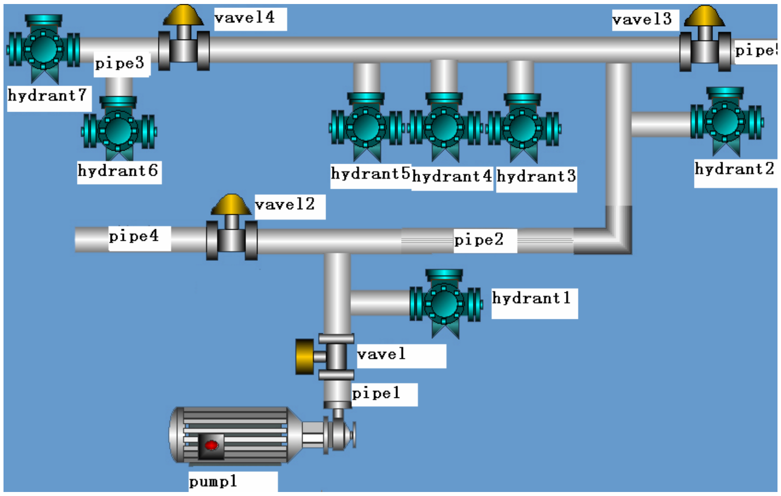


Figure 1. Seawater firefighting system graph

TABLE I. ELEMENTS—KNOTS TAB

pump1	vavel1	hydrant1	vavel2	hydrant2
Knot.1	Knot.2	Knot.3	Knot.4	Knot.5
vavel3	hydrant3	hydrant4	hydrant5	vavel4
Knot.6	Knot.7	Knot.8	Knot.9	Knot.10
hydrant6	hydrant7	pipe1	pipe2	pipe3
Knot.11	Knot.12	Knot.13	Knot.14	Knot.15

Published by Atlantis Press, Paris, France.

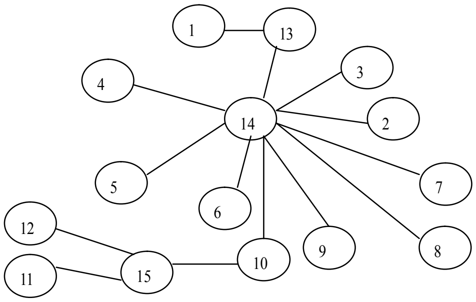
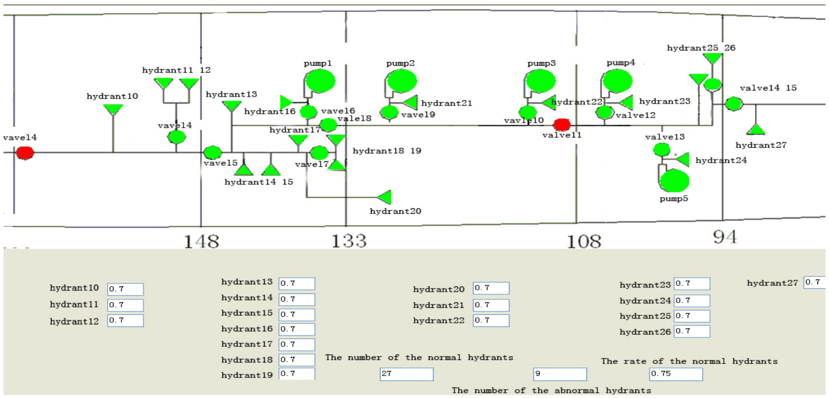


Figure 2. Seawater firefighting system knots-lines graph



hydrant10 0.7
hydrant11 0.7
hydrant12 0.7

hydrant13 0.7
hydrant14 0.7
hydrant15 0.7
hydrant16 0.7
hydrant17 0.7
hydrant18 0.7
hydrant19 0.7

hydrant20 0.7
hydrant21 0.7
hydrant22 0.7

hydrant23 0.7
hydrant24 0.7
hydrant25 0.7
hydrant26 0.7

hydrant27 0.7

The number of the normal hydrants 27

The number of the abnormal hydrants 9

The rate of the normal hydrants 0.75

Figure 3. Seawater firefighting system simulation interface

Published by Atlantis Press, Paris, France.