Modeling and Planning of Snow Sweeping on Main Roads

TEAM MEMBERS
Qin Yipeng, Gao Zhenyuan and Su Cong
TEACHER
Mr. Cai Bin
SCHOOL
Suzhou High School of Jiangsu Province

Abstract

Heavy Snow will turn to a natural disaster, and will bring big economic losses. The authors hope to establish a mathematic model and a plan to illustrate how to sweep off the snow on main roads in a city so as to ensure the smooth flow of traffic. The research will become a basis on which the government can workout some plans against the snow disasters.

Firstly we made some assumptions, then studied the working process of snow-sweepers and derived a snow sweeping model. Based on this model, the relationship between working speed of snow-sweeper and snow thickness, a working model of snow-sweeper, and minimum sets of snow-sweeper needed were established. A deep-searching model and a model of snow-sweeping tree, and a computer program in PASCAL to determine the minimum number of snow-sweeper and the snow sweeping plans on main roads were obtained.

We tried several solutions to deal with this problem, such as using piecewise curve fitting to transform a formula $h\sim v_c$ relating to a cubic equation operation to formula $v_c\sim h$,using trial and error method to determine the minimum number of snow sweepers, using deep searching and tree structure in one-stroke processing in a complicated two-way road system, and also a matrix to deal with the deep searching.

1. Introduction

In the early 2008, an unusual extra-heavy snow struck southern China. More than ten provinces suffered seriously in this snow disaster, all the airlines, railroads, highways and power supplies were damaged even destroyed, millions and millions of money were lost.

The disaster has pasted now but we should study seriously some problems related in this disaster and do our best to enhance our ability against the disasters. This paper will establish a mathematic snow sweeping model and based on this model snow sweeping plans can be worked out.

2. Problem and Assumptions

(1) Problem

According to the weather forecasting a heavy snow will strike a certain city in China. The forecasting says that the highest snowing speed v=0.3*mm/s* and snowing time T=3*h*. In order to have a smooth traffic flowing, the biggest height of accumulated snow on main roads can not be higher than 0.3*m* when sweeping the snow. It is required to determine the minimum number of snow sweepers and prepare a snow sweeping plan.

(2) Assumptions

a. no snow is melting when snowing.

b. snowing process is divided into three phases on snowing velocity basis: early $\frac{T}{3}$ is in uniform accelerating, middle phase uniform speed and the later $\frac{T}{3}$ a uniformly retarded motion.

c. snow sweeper will swerve and turn around on the road and do not time it.

d. all the snow sweepers are with same type and model and no malfunction or failure occurred in snow sweeping.

e. snow sweepers will worked only on two-way roads.

f. no cross interferes between snow sweepers and other automobiles.

The units appeared in this paper are：*m*、*s*、*m/s*、*W*、*N*、*kg*、*N/kg*、*kg/m*^{3}.

(3) Study method

Firstly we study the relationship between the working speed of snow sweeper and snow thickness, then determine the snow sweeper working mode; establish a mathematic model to determine the minimum number of snow sweepers; and at last give a workable plan of snow sweeping on main roads.

3. Snowing velocity and thickness

From the problem and assumptions mentioned above the snowing velocity *u(t)* and snow thickness *h(t)* can be derived as follows[1]:

$$u(t)=\begin{cases}\frac{3v}{T}t, 0\le t\le\frac{T}{3}\\ v\ ,\frac{T}{3}\le t\le\frac{2T}{3}\\ 3v\ -\frac{3v\ t}{T},\frac{2T}{3}\le t\le T\end{cases}\qquad (1)$$

$$h(t)=\begin{cases}\frac{3v}{2T}t^2, 0\le t\le\frac{T}{3}\\ v\ t-\frac{vT}{6},\frac{T}{3}< t\le\frac{2T}{3}\\ 3v\ t-\frac{3v}{2T}t^2-\frac{5v\ T}{6}t,\frac{2T}{3}< t\le T\end{cases}\qquad (2)$$

4. Working speed of snow sweeper and snow thickness

We studied the various kinds of snow sweepers such as model B3500, C2750 and C3100 produced in Herbin, Heilongjiang Province and select a snow sweeper with the following specification :

Effective width of snow sweeper $d\ (m)$ =3;

Mass of snow sweeper $M\,(kg)$ =3000;

Power of snow sweeper $P(W)$ =200000.

The parameters appeared in this study are as follows:

$h_0\ (m)$ --snow thickness when snowing begins;

$v\,(m/s)$ --snowing velocity;

$\rho\ (kg/m^3)$ --snow density， ρ =250;

$v_c\,(m/s)$ --working speed of snow sweeper;

μ_1 --friction coefficient between sweeper and road， $\mu_1 = 0.4$;

μ_2 --adhesion coefficient the road exert to the sweeper， $\mu_2 = 0.1$;

μ_3 --resistance coefficient between the lowest iced layer and the road, $\mu_3 = 0.5$;

$g\,(N/kg)$ --gravity acceleration, g=9.8;

$F\,(N)$ --hauling capacity of sweeper;

$f\,(N)$ --resistance road exert to sweeper, $f = (\mu_1 + \mu_2)Mg$.

The resistance snow exerts to sweeper can be obtained by the momentum law.

Since $Ft = m\Delta v$, so

$$P = Fv_c = (hd\rho v_c^{\ 2} + h\mu_3 d\rho g v_c + f)v_c \qquad (3)$$

Substitute the corresponding numbers:

$$3hv_c^{\ 3} + 15hv_c^{\ 2} + 60v_c - 800 = 0 \qquad (4)$$

$$h = \frac{800 - 60v_c}{3v_c^{\ 3} + 15v_c^{\ 2}} \qquad (5)$$

Formula (5) relates $h \sim v_c$, it has to be in the model needed form of $v_c \sim h$. A segmental function is used to approach accurately the formula $v_c \sim h$.

The results of pircewise fitting are as follows, which is obtained by Excel with v_c step size 0.1:

$$v_c = 7.8426e^{-0.7235h} \qquad R^2 = 0.9982 \qquad h > 0.35m \qquad (6)$$

$$v_c = 8.6339e^{-1.0044h} \qquad R^2 = 0.9985 \qquad 0.24m < h \le 0.35m \qquad (7)$$

$$v_c = 9.3879e^{-1.3621h} \qquad R^2 = 0.9987 \qquad 0.16m < h \le 0.24m \qquad (8)$$

$$v_c = 10.1000e^{-1.8130h} \qquad R^2 = 0.9988 \qquad 0.11m < h \le 0.16m \qquad (9)$$

$$v_c = 10.8110e^{-2.4196h} \qquad R^2 = 0.9986 \qquad h \le 0.11m \qquad (10)$$

5. Working modes of snow sweeper

There are two types of snow sweeper[1]. One is single-sided which has to turn around to sweep the snow on another road side, while the double-sided sweeper will sweep all the snow on the road, but two double-sided sweepers can not be intersected on a road.

A double-sided sweeper will run backwards and forwards on a certain portion of road, as shown in Fig.1

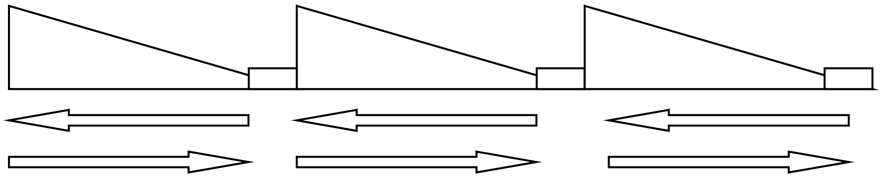


Fig.1

Since double-sided sweepers are usually more expensive,low efficiency, and will block other automobiles when it works, we will not utilize it in this research.

Two working modes can be adopted for a single-sided sweeper, namely big cycle (Fig.2) or little cycle (Fig.3). The sweeper working in little cycle mode will turn around more times than the big cycle, but it runs in a same portion and easy to match the road environment.

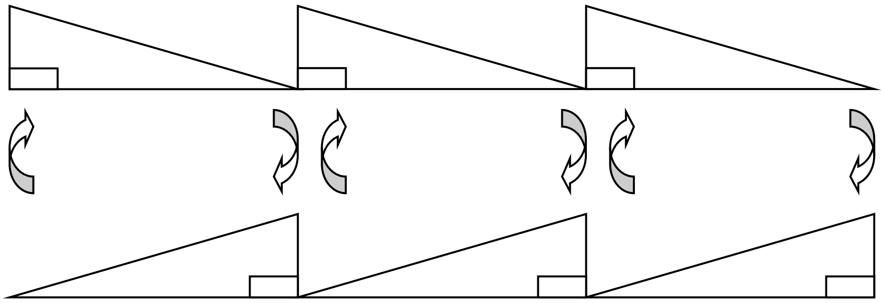


Fig.2

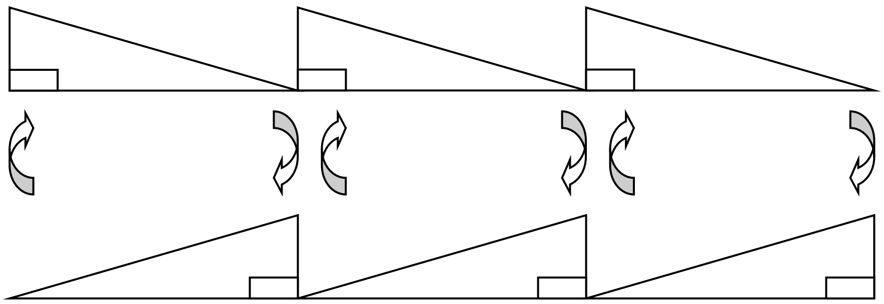


Fig.3

6. Working model of snow sweeper

The sweeper working model is established on the basis of big cycle mode with a single-sided sweeper and working at a period with uniform snowing velocity, because this period has the highest snowing velocity.

Suppose the total length of a road need to be snow sweeping is L, sweeper sets employed is n and sweeping length for each sweeper L_n=2L/n. Since the snow thickness are different, the sweeper working speed in each cycle are also different. This means that for a same length L_n the sweeping time needed will vary with the sweeping operations.

Let t_k stands for the sweeping time in the k^{th} sweeping operation, the remaining snow can be identified approximately as a triangle. If t stands for the time needed from point 0 to point x in

($k+1$) sweeping operation, the height of remaining snow at point x after this operation is[1]

$$h = vt_k \frac{L_n - x}{L_n} + vt = vt_k - \frac{v}{L_n} x t_k + vt \qquad (11)$$

and $dx = v_c(h)dt$ （12）

$x(0) = 0$ （13）

$x(t_{k+1}) = L_n$ （14）

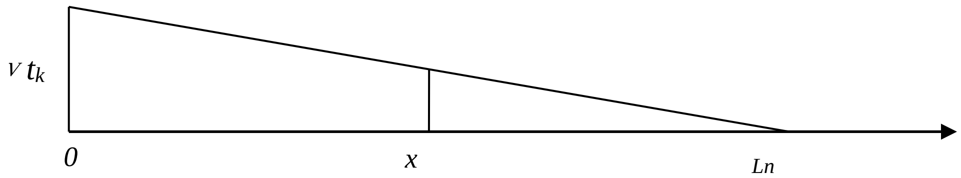


Fig.4

From formulas（6）~（10）the general formula of v_c~ h is:

$$v_c = ae^{bh} \qquad （15）$$

Substitute formula (11) and (15) and boundary conditions of (13) and (14) into (12)[2] :

$$dx = ae^{bh}dt = ae^{b(vt_k - \frac{v}{L_n} t_k x + vt)} dt$$

$$\frac{dx}{dt} = ae^{bvt_k} e^{-\frac{v}{L_n} bt_k x} e^{bvt}$$

$$\int_0^{L_n} e^{\frac{v}{L_n} bt_k x} dx = \int_0^{t_{k+1}} ae^{bvt_k} e^{bvt} dt$$

$$\frac{e^{vbt_k}}{\frac{v}{L_n} bt_k} - \frac{1}{\frac{v}{L_n} bt_k} = ae^{bvt_k} \frac{e^{bvt_{k+1}}}{bv} - \frac{ae^{bvt_k}}{bv}$$

$$\frac{ae^{bvt_k} + \frac{L_n e^{vbt_k} - L_n}{t_k}}{ae^{bvt_k}} = e^{bvt_{k+1}}$$

$$\frac{1}{bv} \ln(1 + \frac{L_n}{at_k} - \frac{L_n}{t_k ae^{bvt_k}}) = t_{k+1} \qquad （16）$$

Formula (16) is the very mathematic model showing how a snow sweeper works.

Below proves the monotone of the above model using mathematical induction.

Proof:

When the uniform snowing velocity begins the snow height:

$$h=\frac{1}{2}at^2=\frac{1}{2}vt=0.54>0.3,$$

Where v=0.0003, the uniform snowing velocity;

t=3600, uniform acceleration time

0.3—the limit of snow height

Above result means that at the beginning of uniform snowing the snow height is bigger than 0.3, the one we supposed sweeping height. So the snow at height 0.23-0.24m is a rectangle, which is sweeped off in the length L_n by the sweeper in the first sweeping operation in uniformly retarded motion.

According to formula (7), when h=0.24m then a=8.6339，b=-1.0014，v_c=6.8m/s, so

$$t_1=\frac{L_n}{6.8} \qquad (17)$$

And also suppose sweeper sets n=6, total length of road L=25000m, then $L_n=\frac{50000}{6}$

From formula (17), t_1=1225 (s)

While from (16), t_2=1439 (s)

Therefore the result is $t_2> t_1$

Next we prove $t_{k+1}>t_k$:

For t_n :

i . when n=2, we have known $t_2 > t_1$;

ii . when $n=k$, $k\in N^*$, k>2, then $t_k>t_{k-1}$;

iii. so when $n=k+1$

$$\frac{1}{bv}\ln(1+\frac{L_n}{at_k}-\frac{L_n}{t_k ae^{bvt_k}})=t_{k+1}$$

$$\frac{1}{bv}\ln(1+\frac{L_n}{at_{k-1}}-\frac{L_n}{t_{k-1}ae^{bvt_{k-1}}})=t_k$$

$$\frac{1}{bv}\ln(\frac{1+\frac{L_n}{at_k}-\frac{L_n}{t_k ae^{bvt_k}}}{1+\frac{L_n}{at_{k-1}}-\frac{L_n}{t_{k-1}ae^{bvt_{k-1}}}})=t_{k+1}-t_k$$

Since b<0，it is needed to prove

$$\ln(\frac{1+\frac{L_n}{at_k}-\frac{L_n}{t_k ae^{bvt_k}}}{1+\frac{L_n}{at_{k-1}}-\frac{L_n}{t_{k-1}ae^{bvt_{k-1}}}})<0$$

i.e.:

$$\frac{1+\frac{L_n}{at_k}-\frac{L_n}{t_k ae^{bvt_k}}}{1+\frac{L_n}{at_{k-1}}-\frac{L_n}{t_{k-1}ae^{bvt_{k-1}}}}<1$$

since there is a “1” in both upper and lower formulas it is only needed to prove:

$$\frac{\frac{L_n}{at_k}-\frac{L_n}{t_k ae^{bvt_k}}}{\frac{L_n}{at_{k-1}}-\frac{L_n}{t_{k-1}ae^{bvt_{k-1}}}}<1$$

$$\frac{\frac{1}{t_k}-\frac{1}{t_k e^{bvt_k}}}{\frac{1}{t_{k-1}}-\frac{1}{t_{k-1}e^{bvt_{k-1}}}}<1$$

Since $t_k > t_{k-1}$, it is easy to have the result:

$$\frac{\frac{1}{t_k}}{\frac{1}{t_{k-1}}}<1$$

Since b<0, then $e^{bvt_k} < e^{bvt_{k-1}}$, so:

$$\frac{1-\frac{1}{e^{bvt_k}}}{1-\frac{1}{e^{bvt_{k-1}}}}<1$$

therefore $t_{k+1} > t_k$. It means that the time series of snow sweeping $\{t_n\}$ is monotone increasing.

7. The minimum number of snow sweeper

We have known that the working model of snow sweeper is a monotone increasing function, i.e. $t_{k+1} > t_k$. Suppose t_u stands for the time needed to sweep the snow with thickness 0.3m, if t_m which stands for time in the m^{th} sweeping operation is bigger than t_u, then t_{m-1} is the time in the $(m-1)^{th}$ operation, and this can be used to determine the number of sweeper.

Since L_n is an unknown number, and either t_1 and t_m. Consequently this makes the number of sweeper n and length L_n unknown. Obviously those parameters are cross correlated.

Therefore a trial and error method is adopted. Firstly we suppose a number of sweeper n_0 which should be smaller, calculate L_n and further t_1 , t_m and t_{m-1} . Accumulate all the time in period of t_1~ t_{m-1}, if it is just bigger than the time in uniform snowing ,i.e.

$$\sum_{k=1}^{m-1} t_k \ge \frac{T}{3} \qquad (18)$$

Then n_0 is the number we need. Otherwise increase $n=n_0+1$ and continue this process until reaching a success.

Trial and error method can be also carried out by accumulated time to determine t_m :

$$\sum_{k=1}^{m} t_k \ge \frac{T}{3} \qquad (19)$$

Then calculate the snow thickness vt_m , if vt_m smaller than 0.3m, the corresponding n is the number of sweeper. Otherwise increase sweeper number and continue this process. (Appendix 1, the computer program is based on this method).

The way to estimate n_0 is that suppose snow sweeping will begin as soon as the snowing begins which is in uniform velocity and reaches to maximum velocity v (in this way n_0 is affirmatively smaller than the actual number needed) , then

$$t_0 = \frac{0.30}{v} \quad , \qquad L_n = \int_0^{t_0} v_c dt = \int_0^{t_0} ae^{bh} dt = \int_0^{t_0} ae^{bvt} dt \quad , \qquad n_0 = \frac{2L}{L_n}$$

Where a、b can be obtained by formula (8), i.e. a=9.3879，b=-1.3621.

Since n (actual number of sweeper) is a limited number, the complex degree of time series in the above trial and error method is O((n_1-n_0)+i) (n_1 is the upper limit to determine the value of n). This degree is very small, which means the trial and error is a feasible method.

In the condition of L=25000，T =3600，v=0.0003，n_1=n_0+50，a=8.6339，b=-1.0044，h_0=0.3，using a computer program the minimum number of sweeper needed is 8. (Appendix 1 lists the source code).

8. Snow sweep planning on main roads

We apply the mathematic model in Xuzhou, Jiangsu Province to workout a plan of snow sweeping on main roads.

Fig.5 shows the total length 122.335 *km* main roads within Xuzhou city. The problems are what the minimum number of sweeper is and how to dispose the sweepers.

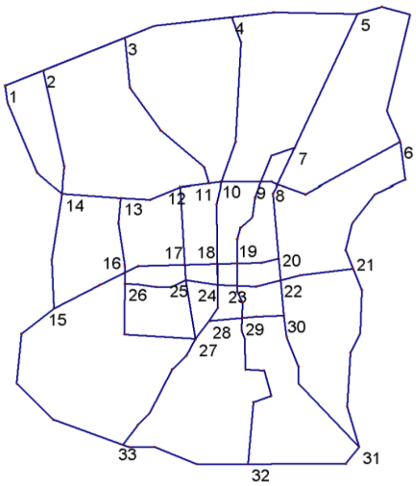


Fig.5

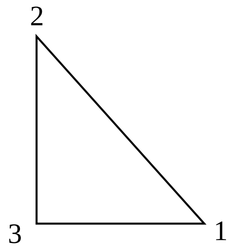
Obviously the best way to sweep the roads is connect all the roads into one series. This is actually a one-stroke (Euler theorem) problem in mathematics.

Now we provide two methods to construct the model for the one-stroke snow sweeping.

The first method:

deep searching model

The conditions of one-stroke is that only two or zero odd points can be included in a point set. The information of points can be quantified by a matrix, as shown below for a three points connection:



$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array}\begin{array}{c} \begin{array}{ccc} \mathbf{1} & \mathbf{2} & \mathbf{3} \end{array} \\ \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$$

In the matrix, "1"stands for connection while "0" not connected. a_{12} =1 means that point 1 and point 2 is connected. In one-stroke operation, one principles are drawing a line and deleting a line at the same time, the value of deleted matrix element should be changed from 1 to 0

Using the following 3 regulations we can now look for the elements in matrix whose value is 1:

(1) Symmetrically deleting

take a diagonal line (point a、f、k、p) as symmetry axis, the value of symmetrical elements

$$\begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix}$$

located along two sides of the axis must be changed synchronously.

(2) Transversely and vertically processing

Search transversely for an element whose value will be changed from 1 to 0, then search for such kind of element in the matrix vertically.

(3) Changing and judging

Once a one-stroke processed the matrix changed and a new matrix produced. The new matrix must be judged before it changes.

① if the current element is the starting point then all the elements in the new matrix with value 1 are even points.

② if the current element is not starting point, then both the current point and starting point are odd ones, and all the others are even points.

Arithmetic of computer program

z[a,b] stands for matrix elements. If point *a* and *b* can be connected then *z[a,b]*≠0. an one-dimensional matrix *c* will store the processed points.

① searching begins from the starting point. Both the value of *z[a,b]* and *z[b,a]* should subtract 1, which means being processed. The serial number in matrix *c* is stored in point *b*.

② according to the principles judging takes place immediately after changing. If the new matrix can not meet the requirements then repeat the searching. Otherwise the searching continues.

③ once find out the result then output it and end the program.

The problems in Arithmetic and processing

① two circumstances need to retreat if searching can not continue:

a. one-stroke can not be carried out in a new matrix;

b. one-stroke can be processed but the current point is a isolated one and no new path can be processed.

When retreating the pointer in matrix *c* moves backward and 1 is added to the element values in matrix *z*.

② optimize the Arithmetic, decrease the repeating and increase the running speed.

Add a column in matrix $n \times n$ and store element degree in the corresponding row so as to facilitate the judgement.

③ for the two-sided road how to make the sweeper running two times in one road: put 2 into the elements whose original values are 1.

Appendix 2 shows the program source code. The result of one-stroke for the Fig.5 as follows:

1-2-1-14-2-3-2-14-13-12-11-3-4-3-11-10-4-5-4-10-9-7-5-6-5-7-8-6-8-7-9-8-9-10-11-12-13-14-15-16-13-16-15-33-27-25-17-12-17-16-17-18-10-18-17-25-24-18-19-9-19-18-24-23-19-20-8-20-19-23-22-20-22-21-6-21-22-23-24-25-26-16-26-25-27-26-27-28-24-28-27-33-32-29-23-29-28-29-30-22-30-29-32-31-21-31-30-31-32-33-15-14-1

The second method:

The one-stroke processing will be much easier if we turn the main roads into a tree structure as shown in Fig.6.

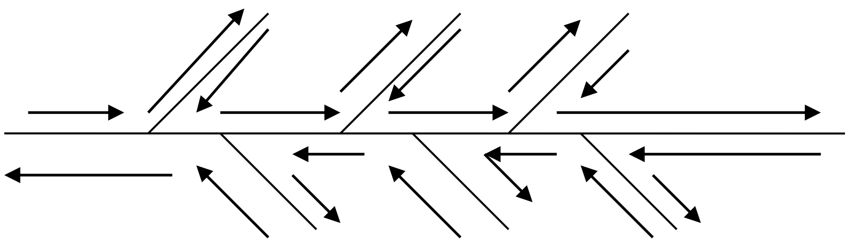


Fig.6

Let us turn the Fig. 5 into a snow sweeping tree as shown in Fig.7: make the north-south road “4-33” a main trunk, east-west roads the first-degree branches and all the other north-south roads the second-degree branches. And same way for the rest roads.

The result of one-stroke processing for the constructed tree:

33-27-26-27-25-27-28-29-23-29-30-22-30-31-30-29-32-29-28-24-25-17-25-26-16-26-25-24-23-19-23-22-20-22-21-22-23-24-18-17-12-17-16-13-16-15-16-17-18-19-9-19-20-8-20-19-18-10-11-3-11-12-13-14-2-14-13-12-11-10-9-7-9-8-5-8-6-8-9-10-4-10-18-24-28-27-33-(clockwise)33-(anticlockwise)33

One-stroke processing actually turn the main roads in a city to a big loop, and all the sweepers are deployed on this loop with same distance and worked in a same direction.

The snow sweeping plan shows that the minimum number of sweeper in Xuzhou city is 39 sets.

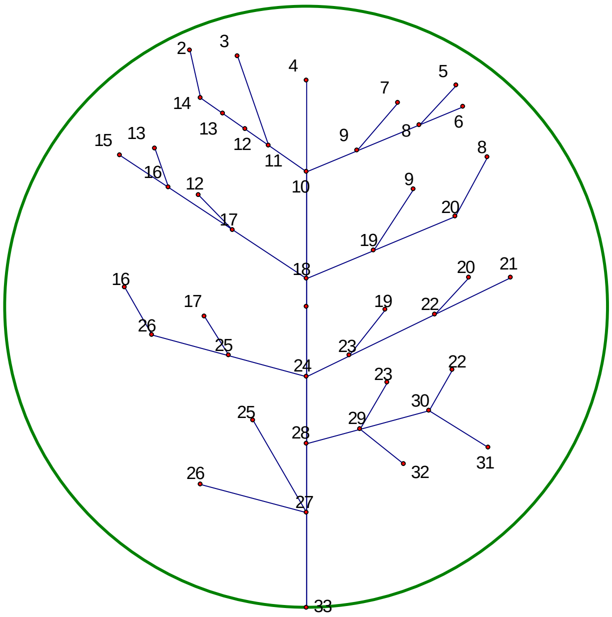


Fig.7

Acknowledgements

Many thanks to Mr. Sun and Mr. Li, director and engineer of Xuzhou observatory, for their friendly help when we visited the observatory. Many thanks to the leaders and teachers of Suzhou Middle School for their instruction, encouragement and support. Many challenges in the study but the feeling of satisfaction and achievement as well as happiness always accompany us. We appreciate the organizers very much for the great opportunity this <u>competition</u> provided to us.

References:

[1] Jian Guoming, Mathematic Model of Snow SweeperNnumber，”Pratise and Understanding of Mathematics”，2002,32(6)：885-890

[2] Mathematics, Edited by Mmathematic Dept., Tongji Univ. (third version)，Press of High Education，1988.4

Appendix 1:Source code to determine the minimum sets of snow sweeper (PASCAL)

```
var a,b,t,s,h,t1,x,l,l0,t0,v,h0,vc:real;
    i,k,n0,n1:integer;
begin
  assign(input,'input.txt');assign(output,'output.txt');
  reset(input);rewrite(output);
  readln(l0,t0,v,h0,vc);
  readln(a,b);
  l0:=l0*2;
  t:=h0/v;
  l:=a*exp(b*v*t)/(b*v)-a/(b*v);
```

```
  n0:=trunc(l0/l);
  for k:=n0 to n0+50 do
  begin
    l:=l0/k;
    t1:=0;i:=0;
    t:=l/vc;
    h:=v*t;
    i:=i+1;
    t1:=t1+t;
    while t1<t0 do
        begin
          x:=b*v*t;
          t:=(ln(1+l/(a*t)-l/(t*a*exp(x))))/(b*v);
          i:=i+1;
          t1:=t1+t;
          h:=v*t;
        end;
    if h<=h0 then begin writeln(k,h*100); break; end;
    end;
close(input);close(output);
end.
```

Appendix 2: Source code of deep seraching for one-stroke processing (PASCAL)

```
const n=33;
            m=56;
type aa=array[1..n,1..n+1] of integer;
var z:aa;
        c:array[1..2*m+1]of integer;
        st,w,k,te,i,a,b:integer;
        label abc;
function search(be,en:integer):boolean;
var i,j:integer;
begin
  search:=true;
  if en=te then
      begin
        for i:=1 to n do if z[i,n+1] mod 2<>0 then begin search:=false; exit; end;
      end
      else if not((z[en,n+1] mod 2<>0)and(z[te,n+1] mod 2<>0)) then search:=false;
end;
procedure main(s,e:integer);
var i:integer;
begin
```

```
  c[w]:=e;
  if (e=te)and(w=2*m+1) then goto abc;
  if (not search(s,e))or((w<2*m+1)and(z[e,n+1]=0)) then exit
    else for i:=1 to n do
                if z[e,i]<>0 then
                  begin
                    dec(z[i,e]);dec(z[e,i]);dec(z[i,n+1]);dec(z[e,n+1]);inc(w);
                    main(e,i);
                    dec(w);inc(z[i,e]);inc(z[e,i]);inc(z[i,n+1]);inc(z[e,n+1]);
                  end;
end;
begin
  assign(input,'in3.txt');{assign(output,'out.txt');}
  reset(input);{rewrite(output);}
  fillchar(z,sizeof(z),0);
  fillchar(c,sizeof(c),0);
  readln(st,te);
  c[1]:=st;w:=1;
  for i:=1 to m do
    begin
      readln(a,b);
      z[a,b]:=2;z[a,n+1]:=z[a,n+1]+2;
      z[b,a]:=2;z[b,n+1]:=z[b,n+1]+2;
    end;
  for k:=1 to n do
    if z[st,k]<>0 then
      begin
        dec(z[st,k]);dec(z[k,st]);dec(z[st,n+1]);dec(z[k,n+1]);inc(w);
        main(st,k);
        dec(w);inc(z[st,k]);inc(z[k,st]);inc(z[st,n+1]);inc(z[k,n+1]);
      end;
  abc:
  for i:=1 to 2*m+1 do write(c[i],' ');
  close(input);{close(output);}
end.
```