

工程實務

污泥沈降槽的電腦模擬設計

張有義*林揚森**

一、前　　言

在廢水處理系統中，污泥沈降槽的設計佔有非常重要的地位。例如由活性污泥槽中溢流至二級沈降槽的活性污泥，如果二級沈降槽設計不當，則此污泥無法在二級沈降槽中有效的沈降，進而會隨著溢流水從沈降槽頂端溢流排放，造成處理後廢水中 TSS 和 BOD_5 濃度值過高，整個廢水處理系統的處理效果也因而降低。而在整個污泥沈降槽的設計考慮因素中，沈降槽的溢流率(overflow rate)大小為一重要的考慮因素，而此溢流率則與沈降槽的表面積有著密切的關係。本文的主要目的，即是在設計一套電腦軟體，將實驗室中所得的污泥沈降數據輸入此軟體，經由個人電腦的運算得到沈降槽應有的設計表面積和其它的設計值，如固體負荷量和溢流率等。

二、沈降槽表面積的設計方法

依據Metcalf and Eddy, Inc. 所著的Wastewater Eng.Treatment, Disposal and Reuse一書中第6.5節的說法，設計沈降槽表面積的方法有二：1.單一的批式沈降量筒實驗法(single batch settling column test)，2.污泥固體通量法(solids flux method)。本文的主要內容即在針對第一種方法設計出一套電腦軟體。所謂的量筒沈降法即是取一有刻度的量筒，觀察在不同時間時污泥所沈降的高度，而做出如圖1所示的界面高度與沈降時間的關係圖。在此必須強調的是由於每一廢水處理系統中所產生活性污泥的

* 私立東海大學化工系副教授

**私立東海大學化工系大學部 研究生

物化性質均不相同，其沈降關係圖也因而不同，因此讀者在使用此套軟體時必須先在實驗室中完成此沈降關係圖。依照Metcalf and Eddy書中的說明，沈降槽表面積的設計公式為：

$$A = \frac{Q \cdot tu}{H_0} \quad (1)$$

其中， Q 為污泥進入沈降槽中的設計流量。

H_0 為污泥在量筒中的最初高度。

tu 為污泥濃度達到沈降槽底部污泥濃度為 C_u 時所須的時間、而 C_u 則為設計者所給予的假定設計值。

如何由圖 1 的沈降關係圖中找出公式 (1) 中的 tu 值，則為整個的設計重點所在。圖 1 中 tu 值所對應的 H_u 值可由下列公式先計算出：

$$H_u = \frac{C_0 \cdot H_0}{C_u} \quad (2)$$

式中， C_0 為污泥在量筒中的最取初濃度值，可由Standard Method測TSS的方法先測得。

H_0 為污泥在量筒中的最初高度。

如何由 H_u 值找出所對應的 tu 值，Metcalf and Eddy-書中所建議的設定步驟為：

1. 分別在圖 1 中沈降曲線的兩端做切線，取其相交點 $P_3(X_3, Y_3)$ 。
2. 將 m_1 和 m_2 交角平分為一半，經過 P_3 點可得一直線 m_3 。
3. 將 m_3 與沈降曲線的交點定為 $P_4(X_4, Y_4)$ ，在 P_4 處取一切線 m_4 。
4. 由公式 (2) 中所算出的 H_u 值處劃一水平直線，其與 m_4 的交點定為 $P_5(X_5, Y_5)$ 。
5. 由 P_5 處向下劃一垂直線，其與X軸的交點即為 tu 值，再經由公式 (1) 即可計算出沈降槽的截面積。

由上述步驟中算出的截面積是由污泥沈降過程中的濃縮區所算出，其值一般均較澄清區算出的面積值為大，故可當做沈降槽的設計值。關於此點，在Metcalf and Eddy-書中的例題6-5有詳細討論，在本文的電腦軟體中亦有考慮，在此不再冗述。

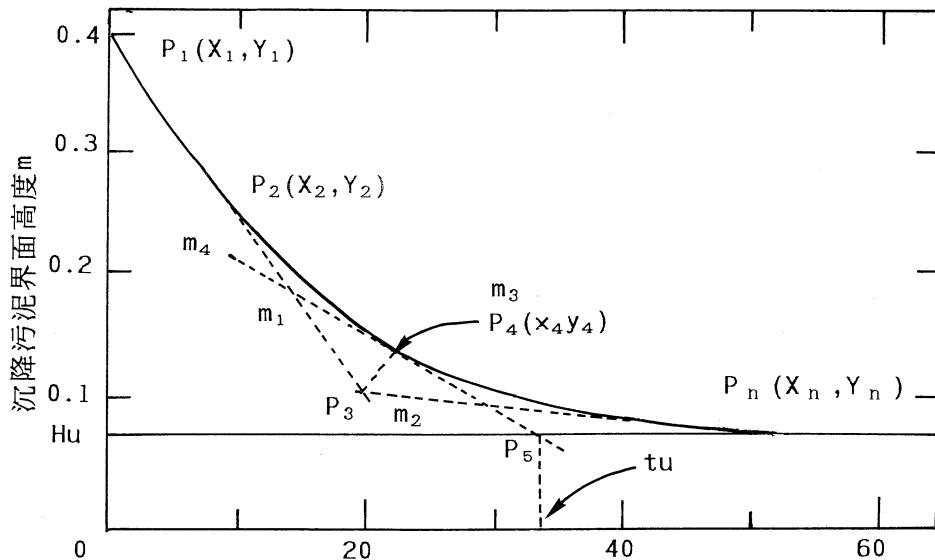


圖 1 應用於本電腦模擬之活性污泥沉降曲線

三、電腦整體設計的說明

由上述方法中可知tu值的求取可完全經由作圖法決定，但此法常會造成人為上的兩項疏失：1.沈降曲線的求取：由於量筒實驗祇能得到一組實驗數據，如何將這些數據轉變成平滑的曲線，若經由人工作圖法常造成很大的誤差，若經由數值分析方法藉著電腦之運算則所得曲線之精確性將大為增加。

一般之數值分析多採用最小平方的迴歸方法，但此法僅適用於一般的線性方程式，如圖1非線性變化則以cubic spline方法為佳，本套軟體將採用此法處理數據。2. m_1 和 m_2 切線的求取：觀察圖1.中沈降曲線其兩端的斜率變化相當不明顯，若以人為做圖法求取此切線常會造成誤差，例如無法求出 P_3 的正確位置。若藉由電腦運算，將能迅速算出曲線兩端各點斜率變化，及正確的算出 m_1 和 m_2 兩條直線。關於上述兩項設計觀念在附錄電腦程式中均有詳細之說明。

茲將作圖法中五個步驟如何轉換為數學運算公式之方法說明如下：

- 1.先以Cubic Spline法決定沈降曲線的變化。
- 2.若 m_1 線和 m_2 線的斜率經電腦算出為 M_1 和 M_2 ，則其直線方程式可寫為：

$$m_1 : Y = M_1 X + (-M_1 X_1 + Y_1)$$

$$m_2 : Y = M_2 X + (-M_2 X_n + Y_n)$$

而 P_3 的交點位置則為：

$$X_3 = (-M_2 X_n + Y_n + M_2 X_1 - Y_1) / (M_1 - M_2)$$

$$Y_3 = (M_2 (M_1 X - Y_1) - M_1 (M_2 X_n - Y_n)) / (M_1 - M_2)$$

3. 圖 1 中 m_3 線的斜率 M_3 可由下列公式算出：

$$M_3 = \tan \left[\frac{\tan^{-1}(M_1) + \pi + \tan^{-1}(M_2)}{2} \right]$$

而 m_3 線可表示為： $Y = M_3 X + (-M_3 X_3 + Y_3)$

4. 由Newton法求取 $P_4(X_4, Y_4)$ 和 m_4 線的斜率 M_4 ，則 m_4 線可表示為：

$$Y = M_4 X + (-M_4 X_4 + Y_4)$$

5. 若假設公式(2) 中的 $H_u = Y_5$ ，則：

$$t_u = X_5 = (Y_5 + M_4 X_4 - Y_4) / M_4$$

其中第4步驟求取 P_4 點的位置，由於 X_4 和 Y_4 並不正好是 Cubic Spline 法所得的內插點位置，是故本套軟體以牛頓法將沈降曲線的內插點代入 M^3 的運算方程式中，將可很正確的算出 P_4 點的位置。有關於此套軟體的設計流程圖和電腦程式則詳載於本文的附錄中。

四、設計例題

以 Metcalf and Eddy 書中的例題 6-5 為例，如果量筒的高度為 0.4m，污泥的最初濃度為 4000ppm，流量為每天 400M³，若在實驗室中的沈降數據為：

t(min)	0	5	10	15	20	25	30	35	40	45	50	55	60
H(m)	0.4	0.33	0.25	0.2	0.16	0.13	0.11	0.1	0.09	0.085	0.08	0.075	0.07

則欲使沈降槽底部的污泥濃度達到 24,000ppm 時，其表面積的設計值，沈降槽的溢流率和固體負荷量可經由此套軟體很快的算出，其時間較作圖法節省許多，所得的答案與該書的結果相當接近，證明其程式的準確性相當高。

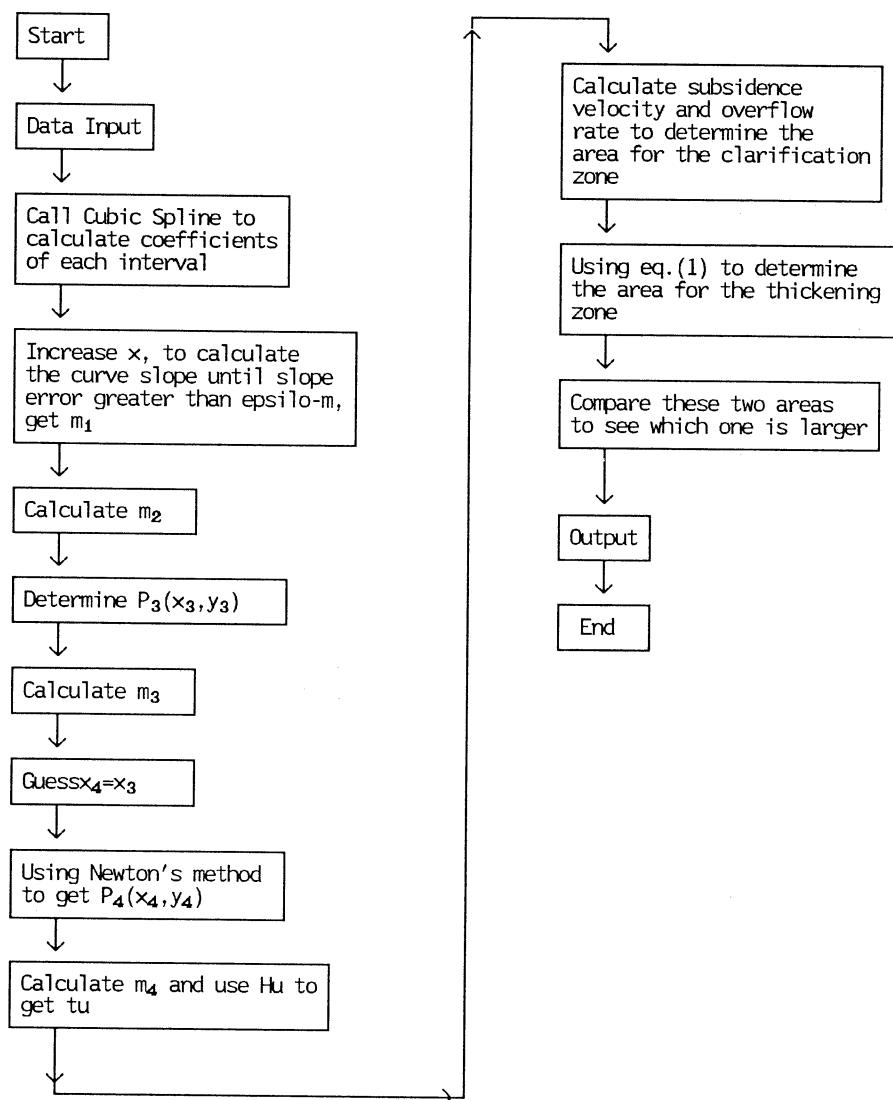
五、結論

本套軟體係以 C 語言寫出，在一般的個人電腦上均可執行，希望能對國內的廢水處理工業有所幫助。

附錄一

程式流程圖

Flow Chart:



本程式作者為

1. 將沈降數據等資料輸入。
2. 以Cubic Spline法求沈降曲線斜率變化，以決定出 $P_3(X_3, Y_3)$ 。
3. 設 $X_4=X_3$ ，以牛頓法求出 $P_4(X_4, Y_4)$ 。
4. 藉Hu線得tu線
5. 計算沈降速率及溢流率，以決定截面積。
6. 利用eq(1) $A=\frac{Q \ tu}{H_0}$ 求出截面積。
7. 比較5,6結果輸出較大者為答案。

附錄二

1. 程式列表

```

type waste.c
#include <stdio.h>
#include <math.h>
#define PI      3.14159
#define epsilon 1e-4
#define epsilon_m 2e-3
#define delta   1e-6
#define delta_x 0.1

typedef float arr1[100];
typedef float arr2[100][4];
float X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5,Xn_1,Yn_1,Xn,Yn,
float m1,m1,m3,m4,C0,Cu,Q,H0,Hu,tu,A,theta,X4old;
float m1old;
arr1 x,y;
arr2 s;
int n,choice;
float v,A2,overflow_rate,solid ,solid loading;

float f1(float X);
float f2(float X);
float df2(float X); /*differential f2*/
float f3(float X);
float df3(float X);           /*differential f3*/
void data_file_input (void);
void cubic_spline(int choice);
float cubic_solve(float intx);
float dcubic_solve(float intx); /*differential cubicsolve*/

main()
{

```

```
printf("Input:\n");
data_file_input();
X1=x[0]; Y1=y[0]; Xn=x[n]; Yn=y[n];
C0=4000.0;Cu=24000.0;Q=400.0; H0=0.4; choice=1;

cubicspline(choice);
X2=delta_x;
m1old=(f2(X2)-f2(X1))/(X2-X1);
do {
    X2=delta_x;
    m1=(f2(X2)-f2(X1))/(X2-X1);
} while (fabs(m1-m1old)<epsilo_m); /*determine p2*/
m2=df2(Xn);
X3=(-m2*Xn-Yn+m1*X1-Y1)/(m1-m2);
Y3=(m2*(m1*X1-Y1)-m1*(m2*Xn-Yn))/(m1-m2);
thita= (atan(m1)+PI+atan(m2)) /2.0;
m3=tan(thita);
X4=X3;           /*initial guess of X4*/
do {
    X4old=X4;
    X4=X4-f3(X4)/df3(X4);
} while (fabs(X4old-X4)>epsilo); /*Newton's method*/
Y4=f2(X4);
m4=df2(X4);
Hu=Y5=C0*Y1/Cu;
tu=X5=(Y5+m4*X4-Y4)/m4;
printf("Result Output :\n");
printf(" tu :=%5.1f\n",tu);
A=Q*tu/H0/(24.0*60.0);
printf(" A1 =%5.1f\n",A);
v=(H0-f2(X2))/X2/(1/60.0); /*subsidence velocity*/
overflow_rate=Q*(H0-Hu)/H0; /*overflow rate*/
A2=overflow_rate/24.0/v; /*area required for clarification*/
solid_loading=(solid=Q*C0/1.0e3)/A;
```

```

printf("compared with A2=%5.1f\n",A2);
if (A2>A) A=A2; /*compare 2 areas*/
printf(" *We get Area :=%5.1fm^2/n",A);
printf(" *overflow rate :=%5.1f m^3/d\n",overflow_rate);
printf(" *solid loading :=%5.1f kg/m^2/d\n",solid_loading);
}/*end of main()*/



float f1(float X)
{return(m3*X+(-m3*X3+Y3));}
float f2(float X)
{return(cubicsolve(X));}
float df3(float X)
{return(dcubicsolve(X));}
floatd f3(float X)
{return(f1(X)-f2(X));}
float df2(float X)
{return(f3(X+delta)-f3(s-delta))/delta/2);}
void data_fileinput(void)
{FILE *f1;
char str[15]
int i;

printf(" Data file name?");gets(str);
f1=fopen(str,"r+t");
if (f1==NULL) printf ("cannot open the file ! \n");exit(1);
fscanf(f1,"%d",&n);
if (n<=100)
n-=1;
for (i=0;i<=n;i++)
fscanf(f1,"%f %f",&x[i], &y[i];
fclose (f1);}
else}
printf ("Number of datas is over 100 ! ! ");
fclose (f1);

```

```
    exit(1);}

}

/*-----
/*Cubic Spline proceduer is to calculate the coefficents of each*/
/*interval. */

/*      switch (choice) */
/*      case 1 : Natural cubic spline. */
/*      case 2 : S"(x) is constant near the endpoints */
/*      case 3 : Extrapolate S"(x) to the endpoints . */
/*Reference ; NUMERCAL METHODS, JOHN H.MATHEWS p.238-p.249 */
/*-----*/
void cubicspline (int choice )
arrl a,b,c,d,m,h,v;
float t,w;
int i;

h[0]=x[1]-x[0];
d[0]=(y[1]-y[0])/h[0];
for (i=1;i<=n-1;i++) {
    h [i]=x[i+1]-x[i]
    d [i]=(y[i+1]-y[i])/h[i];
    a [i]=h[i]
    c [i]=h[i];
    v [i]=6*(d[i]-d[i-1]);
switch (choice)
    case 1: break;
    case 2: b[1]=b[1]+h[0];b[n-1]=b[n-1]+h[n-1]; break;
    case 3: b[1]=b[1]+h[0]+h[0]/h[1];
              c[1]=c[1]-h[0]*h[0]/h[1]
              b[n-1]=b[n-1]+h[n-1]+h[n-1]*h[n-1]/h[n-2];
              a[n-2]=a[n-2]+h[n-1]*h[n-1]/h[n-2];
              break;
default :printf("Choice is not in 1..3 ! \n");exit(1);
for (i=2;i<=n-1;i++) }
```

```

t=a[i-1]/b[i-1];
b[i]=b[i]-t*c[i-1];
v[i]=v[i]-t*v[i-1]; }
m[n-i]=v[n-i]/b[n-1];
for (i=n-2;i>=1;i--)
  m[i]=(v[i]-c[i]*m[i+1])/b[i];
switch (choice) {
  case 1: m[0]=0; m[n]=0; break;
  case 2: m[0]=m[1];m[n]=m[n-1]; break;
  case 3: m[0]=m[1]-h[0]*(m[2]-m[1])/h[1];
            break;}
for (i=0;i<=n-1++) {
  s[i][0]=y[i];
  s[i][1]=d[i]-h[i]*(2*m[i]+m[i+1])/6;
  s[i][2]=m[i]/2;
  s[i][3]=(m[i+1]-m[i])/6/h[i];}

{
float cubicsolve (float intx)
{ float w;
  int i ,k;
  if (intx<[0]||intx>>x[n]))
    printf ("intx value (%f) is out of range ! ! \n",intx);
    exit(1);
  if (intx==x[0]) k=0;
  else
    for (i=i;i<=n;i++)
      if (intx<=x[i]) {k=i-1; i=n;} }

w=intx-x[k];
return( ((s[k][3]*w+s[k][2])*w=s[k][1])*W+S[K][0]);}
{
float dcubicsolve (float intx)
float w;
int i,k;
if (intx<x[0]||intx>x[n])
```

```
- printf("intx value(%f) is out of range ! ! \n",intx);
exit(1);}

if (intx==x[0]) k=0;
else
for (i=1;i<=n;i++)
if (intx<=x[i]) k=i-1; i=n; }

w=intx-x[k];
return( (3*s[k][3]*w=2*s[k][2])*w+s[k][1]);
}
```

2.範例－沈降數據檔

C:\TC>type waste.dat

```
13
0 0.4
5 0.33
10 0.25
15 0.2
20 0.16
25 0.13
30 0.11
35 0.1
40 0.09
45 0.085
50 0.08
55 0.075
60 0.07
```

3.範例一程式執行結果

C:\TC>waste

Input:

Data file name ?waste.dat

Result Output:

tu:=32.3

A1=22.5 conpatred with A2=15.4

*We get Area:=22.5m²

*overflow rate:=333.3m³/d

*solid loading:=71.3kg/m²/d

C:\TC>

