Rozhledy matematicko-fyzikální

Stanislav Trávníček

Triangulace

Rozhledy matematicko-fyzikální, Vol. 88 (2013), No. 3, 15–22

Persistent URL: http://dml.cz/dmlcz/146533

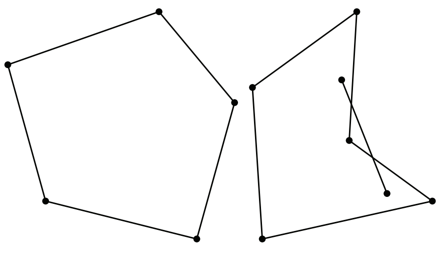
Triangulace

Stanislav Trávníček, PřF UP, Olomouc

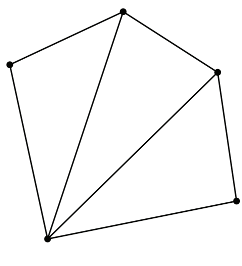
Abstract. The method of triangulation used for calculation of the area of a convex polygon is well known. This article describes a program which at first detects whether the given polygon is convex and then calculates its area in two ways, using the method of triangulation.

Naším dnešním cílem je pojednat o výpočtu obsahu konvexního mnohoúhelníku, jsou-li jeho vrcholy zadány souřadnicemi. Zopakujme si pojem „konvexní“. Daný útvar U nazveme *konvexním*, právě když má tuto vlastnost: při libovolné volbě dvou bodů X, Y útvaru U obsahuje tento útvar celou úsečku XY. Na obr. 1 je zobrazen jeden útvar konvexní a jeden nekonvexní, další výklad k obr. 1 jistě není zapotřebí.

Pro výpočet obsahu konvexních mnohoúhelníků je nejvhodnější metodou *triangulace* (obr. 2). Zvolíme jeden bod mnohoúhelníku, zpravidla jeden z vrcholů, spojíme ho se zbývajícími vrcholy a dostaneme v tomto speciálním případě $n-2$ trojúhelníků. Vypočteme obsah každého z nich a obsahy sečteme. Pohlédněme znovu na nekonvexní pětiúhelník na obr. 1. Kdybychom zvolili za výchozí vrchol ten dolní, stejně jako na obr. 2, pak by se nám i tento nekonvexní pětiúhelník rozdělil na vhodné trojúhelníky. Jestliže bychom však na obr. 1 za výchozí zvolili horní vrchol nekonvexního pětiúhelníku, tak vhodné rozdělení nedostaneme a v případě mechanického výpočtu podle výše uvedeného pravidla bychom dostali nesmyslné číslo.



Obr. 1



Obr. 2

Problém je v tom, že na obr. 1 a 2 ty zadané mnohoúhelníky *vidíme*, jestliže jsou však vrcholy zadány souřadnicemi a navíc, je-li jich více, pak nevidíme nic a museli bychom se spolehnout, že zadavatel vrcholů nějak ví, že jde o vrcholy konvexního mnohoúhelníku. Pokud by však třeba omylem přehodil pořadí u nějakých dvou vrcholů, tak by výpočet obsahu dal opět nesmyslné číslo. Proto *před výpočtem obsahu bychom se měli přesvědčit, že daný n-úhelník je skutečně konvexní*, a protože nám jde o sestavení vhodného počítačového programu, musí tuto konvexnost rozpoznat počítač.

V zájmu větší názornosti proveďme výklad pro pětiúhelník. Jsou dány body A_1, A_2, A_3, A_4, A_5, přímku A_1A_2 nazveme p_1. Přímka p_1 dělí rovinu na dvě poloroviny a je zřejmé, že pro konvexnost pětiúhelníku musí být všechny ostatní vrcholy v téže polorovině. Na obr. 1 však vidíme, že kdyby přímka p_1 byla dána dolními dvěma body, pak by všechny ostatní vrcholy ležely v téže polorovině i pro nekonvexní pětiúhelník, takže p_1 jako kritérium nestačí. Nic nám zatím nebrání, použít jako kritérium všech pět přímek daných stranami, tedy

$$p_1 = A_1A_2, \quad p_2 = A_2A_3, \quad p_3 = A_3A_4, \quad p_4 = A_4A_5, \quad p_5 = A_5A_1.$$

Jestliže některá z těchto přímek má tu vlastnost, že některé ze zbývajících vrcholů leží v opačných polorovinách, pak je daný mnohoúhelník nekonvexní.

Jako programátorům nám může být trochu nepříjemný poslední z vypsaných případů. U všech předchozích jdou indexy vrcholů pěkně po sobě, ale zde je skok $5 - 1$. To programátor jistě nějak zvládne, ale podívejme se znovu na obr. 1 a všimněme si těch dvou „vpáčených" stran. U nekonvexních mnohoúhelníků totiž *vždy existují alespoň dvě strany* určující přímky takové, že některé vrcholy leží v opačných polorovinách. Z toho pro nás plyne zjednodušení, že prostě případ $p_5 = A_5A_1$ můžeme vynechat.

Objasnili jsme princip, teď ještě hledáme analytickou podobu uvedeného rozhodování, do které poloroviny patří který vrchol. Uvažme nejprve případ $p_1 = A_1A_2$. Užitím vzorce známého ze školy najdeme její rovnici (pro $x_1 \neq x_2$)

$$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1),$$

odkud

$$(y - y_1)(x_2 - x_1) - (y_2 - y_1)(x - x_1) = 0.$$

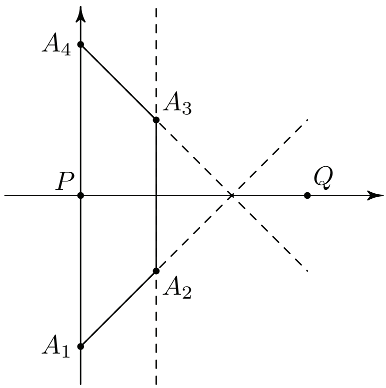
Označme výraz na levé straně symbolem $F(x,y)$. Pak platí, že pro všechny body $M[x_0,y_0]$ z téže poloroviny má výraz $F(M)$ totéž znaménko; pro body opačné poloroviny je znaménko opačné. Pro programování to znamená, že po načtení všech n zadaných vrcholů vytvoříme v cyklu od $k=1$ do $n-1$ výraz $F(x,y)$ pro vrcholy A_k, A_{k+1} a v dalším cyklu zjišťujeme znaménka výrazu $F(x,y)$ pro všech zbývajících $n-2$ vrcholů. Jestliže pro některou přímku A_kA_{k+1} nejsou znaménka $F(x,y)$ pro všechny vrcholy stejná, není zadaný n-úhelník konvexní a jeho obsah nepočítáme.

Zbývá otázka, co když např. $x_1 = x_2$. V tomto případě se výraz F redukuje na

$$F'(x,y) = -(y_2-y_1)(x-x_1).$$

Dokážeme, že tento výraz funguje stejně jako v případě $x_1 \neq x_2$. Abychom se vyhnuli trochu nepříjemnému důkazu s mnoha písmeny, provedeme toto zjištění pro nějakou vhodnou volbu čtyř vrcholů $A_1[0,-2]$, $A_2[1,-1]$, $A_3[1,1]$, $A_4[0,2]$ konvexního mnohoúhelníku (obr. 3). Vypočteme:

p_1: $F_1[x,y] = (y+2)(1-0) - (-1+2)(x-0) = -x+y+2$
p_2: $F_2[x,y] = (y+1)(1-1) - (1+1)(x-1) = -2x+2$
p_3: $F_3[x,y] = (y-1)(0-1) - (2-1)(x-1) = -x-y+2$



Obr. 3

Bod $P[0,0]$ patří zvolenému mnohoúhelníku a vidíme, že

$$F_1(P) = 2 > 0, \quad F_2(P) = 2 > 0, \quad F_3(P) = 2 > 0,$$

všechny tři hodnoty jsou kladné (jsou si rovny jen náhodou). Bod $Q[3,0]$ je zvolen tak, že je ve všech třech případech v opačných polorovinách. Platí

$$F_1(Q) = -1 < 0, \quad F_2(Q) = -4 < 0, \quad F_3(Q) = -1 < 0,$$

znaménka jsou opačná. Vidíme tedy, že výraz F správně funguje i pro $x_1 = x_2$.

Nyní se věnujeme výpočtu obsahu mnohoúhelníku, princip už jsme uvedli v souvislosti s obr. 2, takže víme, že základem je výpočet obsahu trojúhelníku. Uvedeme si dva způsoby.

1. Rozdíly souřadnic použijeme jako délky stran pomocných trojúhelníků a obdélníků.

Zvolme trojúhelník ABC (obr. 4) a jeho vrcholy veďme rovnoběžky s osou x i rovnoběžky s osou y. Označme obsah S trojúhelníku ABC symbolem $|ABC|$ a podobně i další obsahy. Pak platí

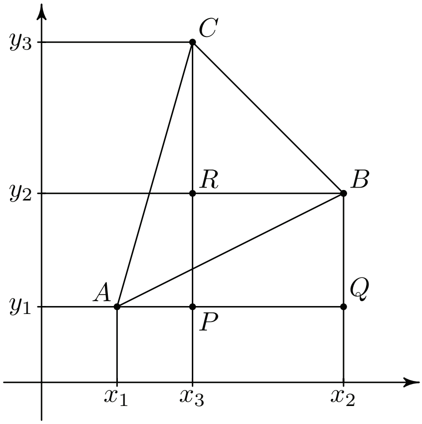
$$|ABC| = |APC| + |RBC| + |PQBR| - |AQB|,$$

což vyjádřeno v souřadnicích dává

$$\begin{aligned} S' = \frac{1}{2}(x_3 - x_1)(y_3 - y_1) + \frac{1}{2}(x_2 - x_3)(y_3 - y_2) + \\ + (x_2 - x_3)(y_2 - y_1) - \frac{1}{2}(x_2 - x_1)(y_2 - y_1) = \\ = \frac{1}{2}[x_3(y_3 - y_1) - x_1(y_3 - y_1) + x_2(y_3 - y_2) - x_3(y_3 - y_2) + \\ + 2x_2(y_2 - y_1) - 2x_3(y_2 - y_1) - x_2(y_2 - y_1) + x_1(y_2 - y_1)] = \\ = \frac{1}{2}[x_3(y_3 - y_1 - y_3 + y_2 - 2y_2 + 2y_1) - x_1(y_3 - y_1 + y_1 - y_2) + \\ + x_2(y_3 - y_2 + 2y_2 - 2y_1 - y_2 + y_1)] = \\ = \frac{1}{2}[x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)]. \end{aligned}$$

Tato hodnota je však závislá na orientaci vrcholů trojúhelníku, může vyjít i záporně (proto je použit zápis s S'), takže musíme vzít její absolutní hodnotou $S = |S'|$, takže výsledný vzorec je

$$S = \left| \frac{1}{2}[x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)] \right|.$$



Obr. 4

2. Pro výpočet obsahu trojúhelníku využijeme vzorec $S = \frac{1}{2}ab\sin\gamma$.

Označme $\boldsymbol{u} = \overrightarrow{AB}$, $\boldsymbol{v} = \overrightarrow{AC}$. Pak platí $\boldsymbol{u}\cdot\boldsymbol{v} = |\boldsymbol{u}|\cdot|\boldsymbol{v}|\cdot\cos\alpha$, kde α je velikost úhlu při vrcholu A. Odsud

$$\cos\alpha = \frac{\boldsymbol{u}\cdot\boldsymbol{v}}{|\boldsymbol{u}|\cdot|\boldsymbol{v}|}.$$

Úhel α může být i tupý, pak je kosinus záporný, ale sinus je i v tomto případě kladný, takže můžeme oprávněně použít k výpočtu sinu vzorec $\sin\alpha = \sqrt{1-\cos^2\alpha}$. Tedy

$$\sin\alpha = \sqrt{1-\left(\frac{\boldsymbol{u}\cdot\boldsymbol{v}}{|\boldsymbol{u}|\cdot|\boldsymbol{v}|}\right)^2} = \frac{\sqrt{|\boldsymbol{u}|^2\cdot|\boldsymbol{v}|^2-(\boldsymbol{u}\cdot\boldsymbol{v})^2}}{|\boldsymbol{u}|\cdot|\boldsymbol{v}|}.$$

Hledaný obsah trojúhelníku ABC je pak

$$S = \frac{1}{2}\,|\boldsymbol{u}|\cdot|\boldsymbol{v}|\cdot\frac{\sqrt{|\boldsymbol{u}|^2\cdot|\boldsymbol{v}|^2-(\boldsymbol{u}\cdot\boldsymbol{v})^2}}{|\boldsymbol{u}|\cdot|\boldsymbol{v}|} = \frac{1}{2}\sqrt{|\boldsymbol{u}|^2\cdot|\boldsymbol{v}|^2-(\boldsymbol{u}\cdot\boldsymbol{v})^2}.$$

Abychom mohli porovnat tyto dva způsoby výpočtu obsahu mnohoúhelníku, vytvoříme cvičný počítačový program, který provede výpočet oběma způsoby. Protože chceme jen ověřit zjišťování konvexnosti a výpočet obsahu, zvolíme program co nejjednodušší, nebudeme např. provádět žádné kontroly, samozřejmě s výjimkou kontroly konvexnosti, která současně ověřuje, že vrcholy jsou zadávány ve správném pořadí.

Vstupní data uložíme do textových souborů se jménem DATA-X.txt, kde X je číslice nebo jiný znak, který pak zadáme po spuštění programu. Na každém řádku je dvojice souřadnic x a y oddělená mezerou a vrcholy zadáváme postupně buď s kladnou nebo zápornou orientací.

Příklady:

```
Data-1       Data-2
-1   -1      -3    -2.5
 3   -3.4     2.5  -1
 4.6  3       3     2
 1    4.2    -1     2
-1    2       0     0
```

V souboru Data-1 jsou zadány vrcholy konvexního pětiúhelníku, v souboru Data-2 pětiúhelníku nekonvexního.

```
program MnohoUh;
  uses Crt;
var
  I, J, N: Integer; S1, S2: Real;
  X, Y: array [1..20] of Real;
  Fil: Text;  T: Boolean;
  Cis: Char;  Jmeno: string;

function SGN(A: Real): Integer;
var P: Integer;
begin
  P := 0;
  if A > 0 then P := 1;
  if A < 0 then P := -1;
  SGN := P
end;  {SGN}

procedure Test(I1: Integer);  {test konvexnosti}
var J1, K: Integer; F: Real;
begin
  K := 0; J1 := 0;
  repeat
   Inc(J1);
   if (J1 <> I1) and (J1 <> I1+1) and (T = true) then
```

```
   begin
     F := (Y[J1] - Y[I1])*(X[I1+1] - X[I1]) -
      (Y[I1+1] - Y[I1]) * (X[J1] - X[I1]);
     if (K = 0) and (F <> 0) then K := sgn(F)
     else if (F = 0) or (sgn(F) <> K) then T := false
   end
  until (J1 = N) or (T = false)
end;  {Test}

function Mod2(IM: Integer): Real;  {druhá mocnina modulu}
  var M: Real;
begin
  M := Sqr(X[IM] - X[1]) + Sqr(Y[IM] - Y[1]);
  Mod2 := M
end;  {Mod2}

function SkSou(IS1,IS2: Integer): Real; {skalární součin}
  var SS: Real;
begin
  SS := (X[IS1] - X[1]) * (X[IS2] - X[1]) +
     Y[IS1] - Y[1]) * (Y[IS2] - Y[1]);
  SkSou := SS
end;  {SkSou}

begin   program
  ClrScr; WriteLn;
  WriteLn(' Obsah mnohoúhelníku');
  WriteLn;
  Write(' Číslo vstupního souboru: ');
  ReadLn(Cis);
  Jmeno := 'Data-'+ Cis + '.txt';
  Assign(Fil,Jmeno);
  Reset(Fil);
  N := 0;
  repeat                    {načtení vrcholů}
    Inc(N);
    ReadLn(Fil, X[N], Y[N])
  until EOF(Fil);
  close(Fil);
  T := true; I := 0;
```

```
  repeat
    Inc(I);
    Test(I)                     {test konvexnosti}
  until (I = N - 1) or (not T);
  if T then
  begin                         {výpočet obsahu}
    S1 := 0; S2 := 0; J := 1;
    repeat
     Inc(J);
     S1 := S1 + (X[1]*(Y[J] - Y[J+1]) + X[J]*(Y[J+1] -Y[1])
       + X[J+1]*(Y[1] - Y[J])) / 2;
     S2 := S2 + Sqrt(Mod2(J) * Mod2(J+1)
       - SkSou(J,J+1)* SkSou(J,J+1)) /2;
    until J = N - 1;
    WriteLn;
    WriteLn(' Obsah zadaného ',N,'úhelníku: ');
    WriteLn(' 1. metodou: ',Abs(S1):5:3);
    WriteLn(' 2. metodou: ',S2:5:3)
  end else
  begin
    WriteLn;
    WriteLn(' Zadaný ',N,'úhelník není konvexní.')
  end;
  WriteLn;
  ReadLn
end.  {program}
```

Na obr. 5 a 6 je znázorněn výsledek zpracování zadaných dvou datových souborů.

```
Obsah mnohoúhelníku

Číslo vstupního souboru: 1

Obsah zadaného 5úhelníku:
1. metodou: 28.280
2. metodou: 28.280

_
```

Obr. 5

```
Obsah mnohoúhelníku

Číslo vstupního souboru: 2

Zadaný 5úhelník není konvexní
_
```

Obr. 6