

## DOMBORZAT MODELLEZÉS - INTERPOLÁCIÓ

---

Terepfelmérés során többnyire szórt pontokban kapunk magassági értékeket, melyekből szeretnénk digitális domborzatmodellt készíteni, szintvonalas ábrát vagy metszeteket rajzolni az éppen aktuális feladatnak megfelelően.

A digitális domborzatmodell készítése során a szórt pontokban történő felmérés eredményéből szeretnénk egy olyan modellt előállítani, amiből a felmért terület bármely pontjában kiszámolható a magasság értéke. Ez történhet például Delaunay háromszögeléssel és utána lineáris interpolációval, vagy akár spline interpolációval is. A domborzatmodellt többnyire rácshálóra interpolált értékekkel adjuk meg.

A már elkészült domborzatmodellből levezethetünk szintvonalas térképeket vagy terepmetszeteket is.

Töltse le a `meres_coo.txt` és a `clabel2.m` fájlokat és másolja be a munkakönyvtárba.

A terepfelmérés során mért koordinátáink a `meres_coo.txt` fájlban találhatóak a következő formátumban (pontszám, EOY koordináták (Y,X), Balti tengerszint feletti magasság):

```
2001 577057.011 188795.517 142.042
2002 577051.903 188783.028 140.821
2003 577051.044 188772.868 140.317
2004 577053.460 188758.714 139.622
2005 577053.097 188757.082 139.478
2006 577052.829 188752.585 139.186
2007 577043.018 188773.355 139.426
```

...

Miután azonos típusú, sorhosszúságú adataink vannak a beolvasás egyszerűen megtörténhet a `load`-dal.

```
> clear all; close all; clc;
> page_screen_output(0);% laponkénti megjelenítés leállítása Octave-ban
>
> data=load('meres_coo.txt');
> x = data(:,2);
> y = data(:,3);
> z = data(:,4);
> % Nézzük meg az adatok minimális, maximális értékeit
> format long
> xmin = min(x), xmax = max(x)
> ymin = min(y), ymax = max(y)
> zmin = min(z), zmax = max(z)
>
> % Kiterjedés
> xmax-xmin
> ymax-ymin
```

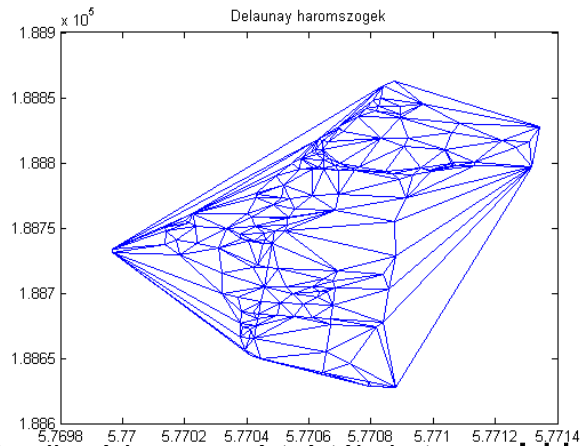
## LINEÁRIS INTERPOLÁCIÓ - RÁCSRA

---

A lineáris interpoláció szórt pontokra többnyire Delaunay háromszögelés alapján történik, ezekre a háromszögekre síkot illetve ki lehet számolni minden pont magasságát.

Jelenítsük meg a háromszögeket!

```
> figure(1)
> % a csúcspontok indexei:
> tri = delaunay(x, y)
> triplot(tri, x, y);
> title('Delaunay háromszögek');
```



Interpoláljuk méterszer méteres rácshálóra lineárisan a mért értékeket a **griddata** parancs használatával! Ehhez először egy megfelelő rácshálót kell generálnunk a **meshgrid** használatával.

```
> % Interpoláció rácshálóra meshgrid és griddata használatával
> x_vekt=round(xmin):1:round(xmax); % egész méterenkénti x koordináták
> y_vekt=round(ymin):1:round(ymax); % egész méterenkénti y koordináták
> % kerekítési parancsok:
> % round-legközelebbi egészre, floor-lefelé, ceil-fölfelé kerekít
> [XI YI] = meshgrid(x_vekt, y_vekt); % rácsháló
> ZI=griddata(x, y, z, XI, YI, 'linear'); % lineáris interpoláció rácsra
```

A **griddata** használatával rácsra interpolálhatunk. Lehetőségünk van legközelebbi szomszéd ('nearest') vagy lineáris ('linear') interpolációra. Ez utóbbi az alapértelmezett, ha nem adunk meg semmilyen interpolációs módszert. A Matlab-ban ezen kívül lehetőség van spline ('cubic') interpolációra is, ez azonban sajnos még nincs implementálva Octave-ba. Spline interpolációhoz ezért a spline csomagot fogjuk most használni.

## INTERPOLÁCIÓ RÁCSRÓL TETSZŐLEGES PONTRA

---

Az **interp2** paranccsal kérdezhetünk le magasság értékeket a rácshálóban előállított domborzat modellünkből. Ez pont a fordítottja a korábbi szórt pontokról rácsra történő interpolációnak, itt a rácsháló pontjait használjuk fel, hogy tetszőleges pontokban magasságot számoljunk. Kérdezzük le például az  $x_1 = 577080.3$  és  $y_1 = 188750.3$  pontban a magasságot!

```
> x1 = 577080.3; y1 = 188750.3;
> z1 = interp2(XI, YI, ZI, x1, y1)
```

Az eredmény  $z_1 = 141.424$  m lesz.

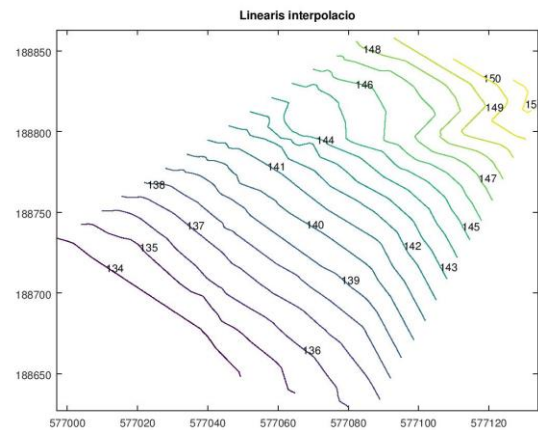
Az **interp2**-nél is beállítható, hogy milyen módszerrel történjen a rácsról az adott pontra az interpoláció, lehet 'nearest', 'linear' vagy 'cubic', ha nem adunk meg semmit, akkor lineáris az alapértelmezett.

## DOMBORZAT MEGJELENÍTÉSI LEHETŐSÉGEK

### SZINTVONALAS MEGJELENÍTÉS

```
> % szintvonalak rajzolása a griddata eredményének felhasználásával
> % Figyelem: meglehetősen lassú a szintvonalak generálása!
```

```
> figure(2)
> a = ceil(zmin)
> b = floor(zmax)
> [C, h] = contour(XI, YI, ZI,
    a:b);
> % clabel(C, h, a:b);
> % töltsük le a clabel2.m fájlt!
> clabel2(C, a:b);
> title('Linearis interpolacio');
> % mentjük el képként!
> print('interp_lin.jpg', '-djpeg')
```



Megjegyzés: a szintvonalak feliratozására a `clabel` parancsot is használhatjuk. Octave-ban azonban célszerű ennek egy módosított változatát használni (`clabel2`), mivel az eredeti bizonyos esetekben túl sűrűn feliratozza a szintvonalakat.

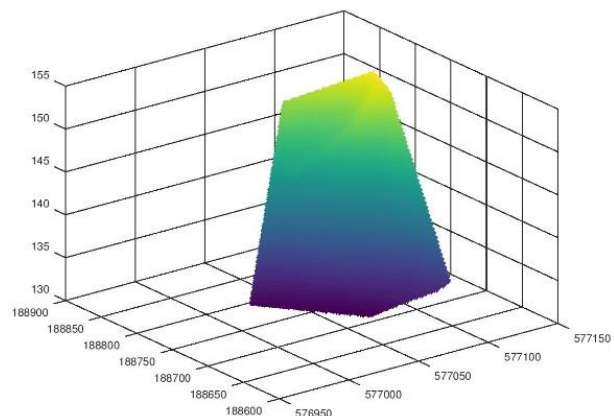
Vegyük észre, hogy extrapoláció nem történt az egész rácsra, csak a mért pontjainkat befoglaló konvex sokszögön belül vannak értékeink. (Ellenőrizzük le a ZI interpolált értékeket! Ahol nem volt adat oda NaN (not a number) értékek kerültek.)

### FELÜLET ÁBRÁZOLÁS

Másik ábrázolási lehetőség a 3D-ben színezett felülettel történő megjelenítés, a `surf` paranccsal.

```
> figure(3);
> surf(XI, YI, ZI, 'EdgeAlpha', 0)
```

Itt az `EdgeAlpha` paraméter 0-ra állítása azért szükséges, hogy a zavaró felülethatárok ne rajzolódjanak ki.



## MEGJELENÍTÉS SZÍNÁTMENETTEL

Szintvonalas megjelenítésen kívül magasság szerinti színátmenetes megjelenítéssel is ábrázolhatjuk a domborzatmodellünket. Ehhez használhatjuk az `imagesc` parancsot. Ez tulajdonképpen egy mátrixot jelenít meg képként.

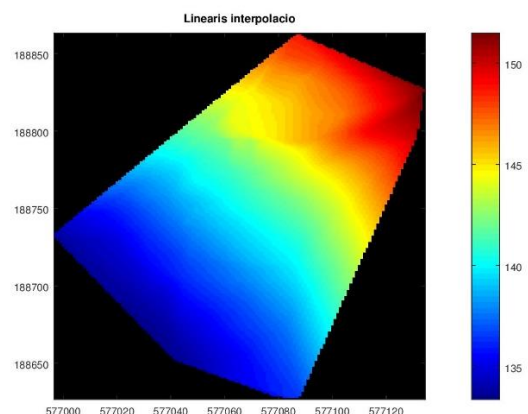
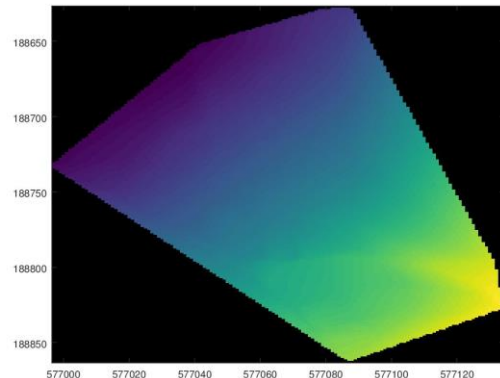
```
> figure(4);
> imagesc(XI, YI, ZI);
> % imagesc(XI(1,:), YI(:,1), ZI);
```

Octave-ban a bemenő x,y,z koordináták is lehetnek a `meshgrid` által előállított mátrixos formában. Matlabban ki kell vegyünk a mátrixokból egy sort illetve egy oszlopot, mivel itt a bemenő x és y koordinátákat vektorban kell megadni, míg a z koordinátákat egy rácsháló pontjaiban.

Ha összehasonlítjuk az fenti képet a korábbiakkal, akkor feltűnik, hogy olyan a domborzatmodell, mintha függőleges irányban tükrözve lenne. Ez tényleg így van, ugyanis az `imagesc` koordináta rendszere a képek koordináta rendszerének felel meg (ij), azaz a bal felső sarok a kezdőpont, szemben a nálunk használt észak-keleti (xy) tájolású koordináta rendszerekkel. A kettő között az `axis xy` vagy `axis ij` paranccsal lehet váltani. A színskálát kitenni a `colorbar` paranccsal tudjuk, a színskálát változtatni pedig a `colormap` paranccsal.

Próbáljunk ki különböző színskálákat! Alapértelmezett Octave alatt a `viridis`, Matlab alatt a `jet` skála. Egyéb pl.: `hot`, `hsv`, `summer`, `autumn`, `spring`, `winter`, `cool`, `gray`, `lines`.

```
> axis xy;
> colorbar;
> title('Linearis interpolacio');
> colormap(jet)
```



## EREDMÉNYEK LEKÉRDEZÉSE TETSZŐLEGES PONTRA GRAFIKUSAN

Használjuk most is az `interp2` parancsot, csak most grafikusan jelöljük ki azt a pontot, ahol szeretnénk lekérdezni a magasságot! A `ginput(n)` paranccsal a képre kattintva n pont koordinátáit kérdezhajtuk le. Az `msgbox` paranccsal megjeleníthetünk egy üzenetet a felhasználónak egy grafikus ablakban. A `text` paranccsal szöveget írhatunk az ábra tetszőleges pontjára.

```
> msgbox('Kattintson a terkepen!')
> [xA yA] = ginput(1)
> zA = interp2(XI, YI, ZI, xA, yA)
> text(xA,yA,num2str(zA))
> msgbox(sprintf('A pontban a magassag: %.2f', zA))
```

## SPLINE INTERPOLÁCIÓ (RÁCSRA)

---

A köbös spline interpoláció Matlabban megoldható a `griddata` egy másik beépített módszerével, a `'cubic'` használatával, ezt azonban Octave alá még nem implementálták. Helyette használjuk a spline csomag `tpaps` parancsát (thin plate spline). Először nézzük meg, hogy telepítve van-e a csomag! Ha nincs telepítsük az internetről!

```
> pkg list
> % spline csomag telepítése internetről ha szükséges:
> % pkg install -forge splines
>
> pkg describe splines -verbose %kilistázza a splines csomag parancsait
> pkg load splines; % splines csomag betöltése
> ZI2 = tpaps([x y], z, 1, [XI(:) YI(:)]);
>
> % Ugyanez a Matlabban két paranccsal oldható meg:
> %   st = tpaps([x y]', z', 1);
> %   ZI2 = fnval(st, [XI(:) YI(:)]');
>
> ZI2 = reshape(ZI2, size(XI)); % visszaméretezés vektorból mátrixba
> figure(5);
> surf(XI, YI, ZI2) % extrapoláció is!
> ZI2(isnan(ZI)) = nan; % az extrapoláció kiküszöbölése
> surf(XI, YI, ZI2) % extrapoláció nélkül
```

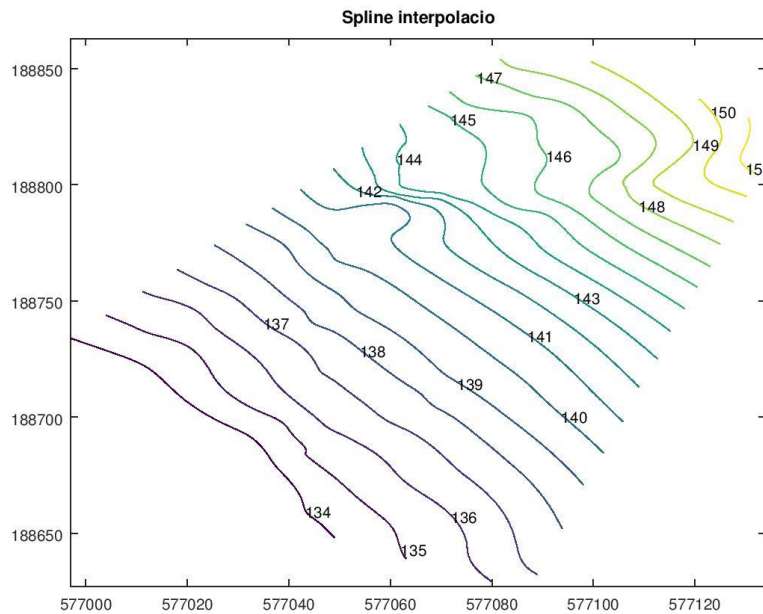
A `tpaps` parancs ellentétben a `griddata`-val nem a `meshgrid` által mátrix formában előállított rácshálóban kéri az interpolálandó pontok helyét, hanem vektorosan összetartozó `[xi yi]` értékpárokat kér. Ezáltal azonban nem csak rácshálóra végezhető interpoláció, hanem bármilyen pontokra! A bemenetnél is összetartozó `[x y]` értékpárookra van szükség egy mátrixban, és a hozzájuk tartozó `z` értékekre egy külön vektorban. Utána megadható egy 0-1 közötti szám, ami a simítási tényező, ami a 'vékony lemez' merevségét jellemzi. 0 esetén teljesen merev, ekkor egy közelítő síkot illeszt a pontokra, 1 esetén pedig interpolál, azaz minden ponton átmegy a felület. A kettő között valamilyen regresszió van, minél közelebb van a szám 0-hoz, annál inkább a síkhoz hasonló sima felületet kapunk. (Megj: gyakorlatban azonban úgy tűnik nincs különbség, hogy 0-t vagy 1-et adunk meg, ugyanaz lesz a végeredmény, az Octave 4.0.1 változatán kipróbálva.)

Octave-ban egy paranccsal hívható a módszer, Matlab esetén két parancs kell, először kiszámítjuk a spline együtthatóit, és utána az `fnval` paranccsal kiértékeljük az eredményt a rácspontokban.

A vektoros formában történő megadáshoz a `meshgrid` eredményeképpen kapott `XI`, `YI` értékeket oszlopvetorrá alakítottuk az `XI(:)` és `YI(:)` parancsokkal.

Az eredményt a szintvonalas megjelenítéshez visszaalakítjuk mátrix alakba (`reshape`). Mivel ezzel a módszerrel extrapolálni is lehet, ahol viszont nagyon rossz eredményeink lennének, ezért az extrapolált helyeket NaN-nel töltjük fel (Not a Number). Ehhez felhasználtuk a lineáris interpoláció eredményét.

```
> % A spline interpoláció szintvonalas megjelenítése  
> figure(6)  
> [C, h] = contour(XI, YI, ZI2, a:b);  
> clabel2(C, a:b);  
> title('Spline interpolacio');  
> % mentsük el képként!  
> print('interp_spline.jpg', '-djpeg')
```



Ez a módszer jóval simább szintvonalakat eredményez, mint a lineáris interpoláció.

Mentsük el a létrehozott domborzatmodelleket későbbi felhasználásra, a szórt pontokkal együtt!

```
> save domborzat.mat XI YI ZI ZI2 x y z
```