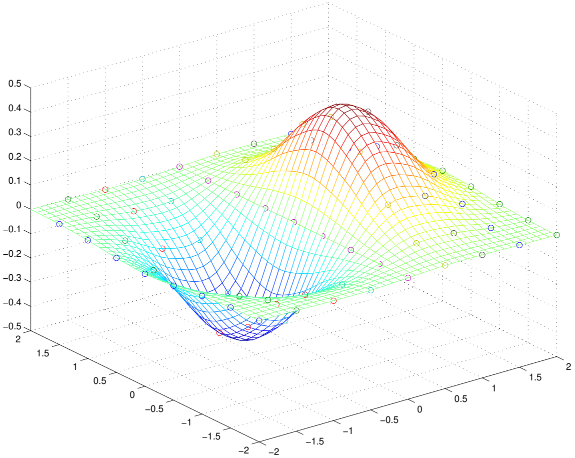
Two Dimensions



Regular grids

- Now $\mathbf{x} = \{x_1, \ldots, x_n\} \in \mathbf{R}^n$ is a multidimensional data point. Focus on **two-dimensions** (2D) since **three-dimensions** (3D) is similar.
- The easiest case is when the data points are all inside a **rectangle**

$$\Omega = [x_0, x_{m_x}] \times [y_0, y_{m_y}]$$

where the $m = (m_x + 1)(m_y + 1)$ nodes lie on a **regular grid**

$$\mathbf{x}_{i,j} = \{x_i, y_j\}, \quad f_{i,j} = f(\mathbf{x}_{i,j}).$$

- Just as in 1D, one can use a different interpolation function $\phi_{i,j} : \ \Omega_{i,j} \rightarrow \mathbb{R}$ in each rectangle of the grid (pixel)

$$\Omega_{i,j} = [x_i, x_{i+1}] \times [y_j, y_{j+1}].$$

Bilinear Interpolation

- The equivalent of piecewise linear interpolation for 1D in 2D is the **piecewise bilinear interpolation**

$$\phi_{i,j}(x,y) = (\alpha x + \beta)(\gamma y + \delta) = a_{i,j}xy + b_{i,j}x + c_{i,j}y + d_{i,j}.$$

- There are 4 unknown coefficients in $\phi_{i,j}$ that can be found from the 4 data (function) values at the corners of rectangle $\Omega_{i,j}$. This requires solving a small 4×4 linear system inside each pixel independently.
- Note that the pieces of the interpolating function $\phi_{i,j}(x,y)$ are **not linear** (but also **not quadratic** since no x^2 or y^2) since they contain quadratic product terms xy: **bilinear functions**.
  This is because there is not a plane that passes through 4 generic points in 3D.

Piecewise-Polynomial Interpolation

- The key distinction about **regular grids** is that we can use **separable basis** functions:

$$\phi_{i,j}(\mathbf{x}) = \phi_i(x)\phi_j(y).$$

- Furthermore, it is sufficient to look at a **unit reference rectangle** $\hat{\Omega} = [0,1] \times [0,1]$ since any other rectangle or even **parallelogram** can be obtained from the reference one via a linear transformation.
- Consider one of the corners $(0,0)$ of the reference rectangle and the corresponding basis $\hat{\phi}_{0,0}$ restricted to $\hat{\Omega}$:

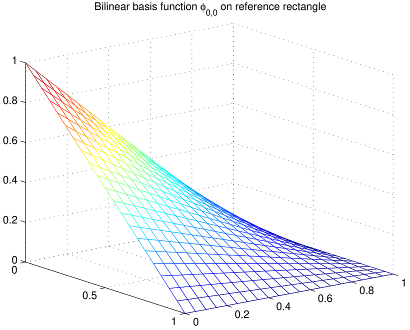
$$\hat{\phi}_{0,0}(\hat{x},\hat{y}) = (1-\hat{x})(1-\hat{y})$$

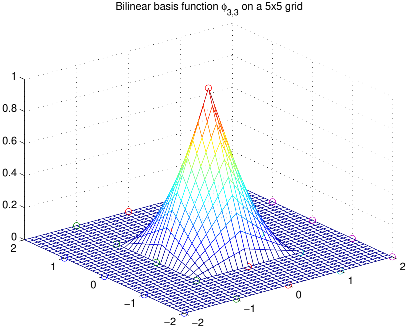
- Generalization of bilinear to 3D is **trilinear interpolation**

$$\phi_{i,j,k} = a_{i,j,k}xyz + b_{i,j,k}xy + c_{i,j,k}xz + d_{i,j,k}yz + e_{i,j,k}x + f_{i,j,k}y + g_{i,j,k}z + h_{i,j,k}$$

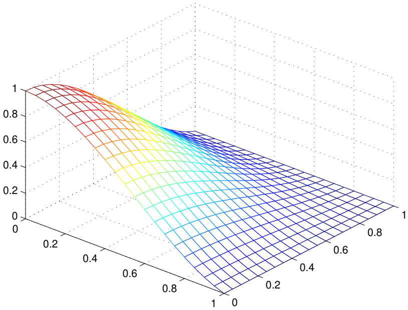
which has 8 coefficients which can be solved for given the 8 values at the vertices of the cube.

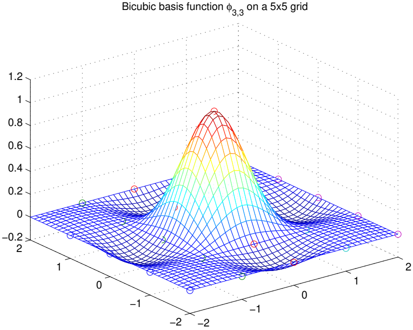
Bilinear basis functions





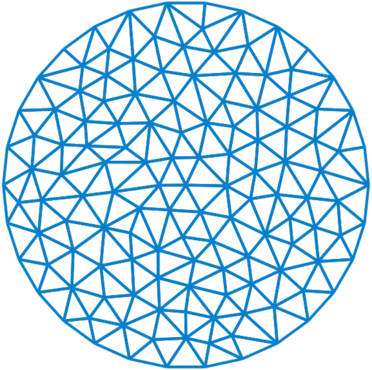
Bicubic basis functions

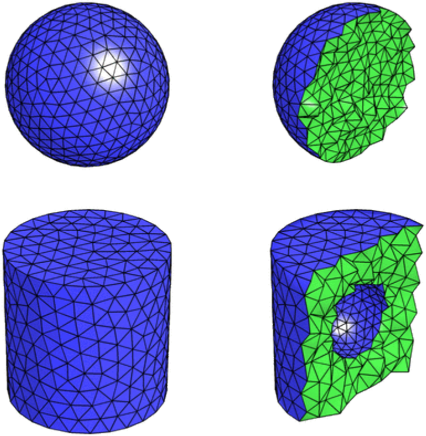




Irregular (Simplicial) Meshes

Any polygon can be triangulated into arbitrarily many **disjoint triangles**. Similarly **tetrahedral meshes** in 3D.





Basis functions on triangles

- For irregular grids the x and y directions are no longer separable.
- But the idea of using basis functions $\phi_{i,j}$, a **reference triangle**, and **piecewise polynomial interpolants** still applies.
- For a piecewise constant function we need one coefficient per triangle, for a linear function we need 3 coefficients (x, y, const), for quadratic 6 (x, y, x^2, y^2, xy, const), so we choose the **reference nodes**:

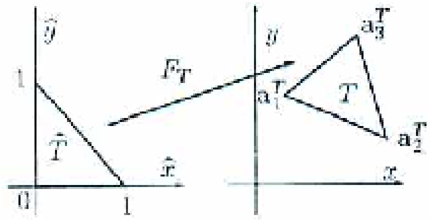


Fig. 8.8. Local interpolation nodes on $\hat{T}$ for $k = 0$ (*left*), $k = 1$ (*center*), $k = 2$ (*right*)