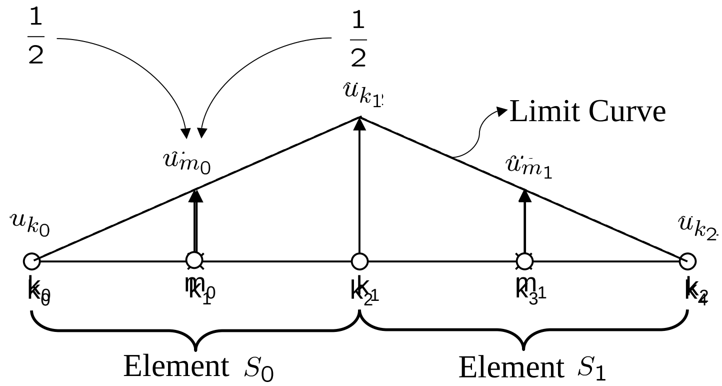
Simple Linear Interpolation



$$\left(u_{k_0} + u_{k_1} z^{-2} + u_{k_2} z^{-4}\right) \times \left(\frac{1}{2} z + 1 + \frac{1}{2} z^{-1}\right) =$$

$$\frac{1}{2} u_{k_0} z + u_{k_0} + \frac{1}{2}\left(u_{k_0} + u_{k_1}\right) z^{-1} + u_{k_1} z^{-2} + \frac{1}{2}\left(u_{k_1} + u_{k_2}\right) z^{-3} + u_{k_2} z^{-4} + \frac{1}{2} u_{k_2} z^{-5}$$

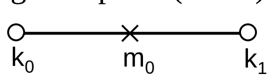
Unchanged

Interpolating Subdivision Schemes

- Given a set of data $\{u_{j,k_0}, u_{j,k_1}, \ldots, u_{j,k_N}\}$, find filters $h_j[k,m]$ such that:

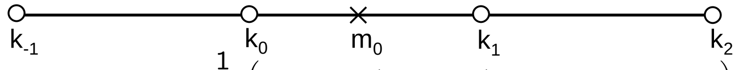
$$\left.\begin{aligned} u_{j+1,k} &= u_{j,k} \\ u_{j+1,m} &= \sum_{k \in N(j,m)} h_j[k,m] u_{j,k} \end{aligned}\right\} \underline{u}_{j+1} = \mathrm{S}\underline{u}_j$$

- e.g. two point (linear) scheme



$$u_{j+1,m_i} = \frac{1}{2}\left(u_{j,k_i} + u_{j,k_{i+1}}\right)$$

four point (cubic) scheme

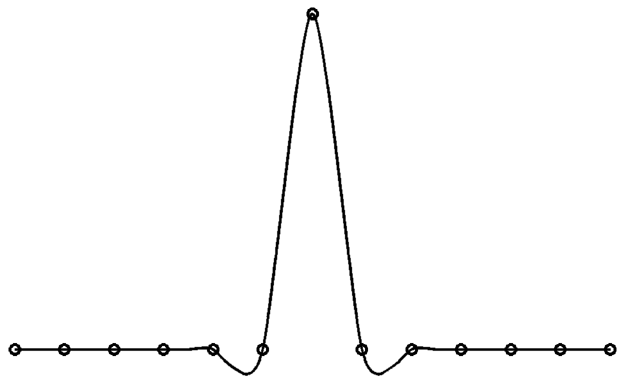


$$u_{j+1,m_i} = \frac{1}{16}\left(-u_{j,k_{i-1}} + 9u_{j,k_i} + 9u_{j,k_{i+1}} - u_{j,k_{i+2}}\right)$$

- Generalizes easily to multiple dimensions, non-uniformly spaced points, boundaries, etc.

Interpolating Subdivision Schemes

- Limit curve is an interpolating function

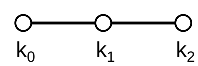


Wavelets From Subdivision

- Limit curves can be used to interpolate data.

On coarse grid $\quad \mathcal{K}(j) = \{k_0, k_1, \ldots\}$

$$f_j(x) \;=\; \sum_{k \in \mathcal{K}(j)} u_{j,k}\varphi_{j,k}(x)$$



On fine grid $\quad \mathcal{K}(j+1) = \{k_0, m_0, k_1, \ldots\}$

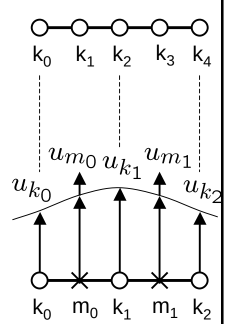
$$f_{j+1}(x) \;=\; \sum_{l \in \mathcal{K}(j+1)} u_{j+1,l}\varphi_{j+1,l}(x)$$

Suppose that $u_{j+1,l}$ is coarsened by subsampling

$$u_{j,k} = u_{j+1,k}$$

and remaining data is predicted using subdivision

$$u_{j,m} = u_{j+1,m} - \sum_{k \in N(j,m)} h_j[k,m]u_{j,k}$$



Wavelets From Subdivision

- Does this fit the wavelet framework?

$$\begin{aligned} f_{j+1}(x) &= \sum_{l\in\mathcal{K}(j+1)} u_{j+1,l}\varphi_{j+1,l}(x) \qquad \text{fine approximation} \\ &= \underbrace{\sum_{k\in\mathcal{K}(j)} u_{j,k}\varphi_{j,k}(x)}_{\text{coarse approximation}} + \underbrace{\sum_{m\in\mathcal{M}(j)} u_{j,m}w_{j,m}(x)}_{\text{details}} \end{aligned}$$

If we set $u_{j,k} = 0$, $u_{j,m} = \delta_{m,m'}$, our coarsening/prediction strategy gives

$$\begin{aligned} u_{j+1,k} &= u_{j,k} &&= 0 \\ u_{j+1,m} &= u_{j,m} + \sum_{k\in N(j,m)} h_j[k,m]u_{j,k} &&= \delta_{m,m'} \end{aligned}$$

So the "wavelets" are

$$w_{j,m}(x) = \varphi_{j+1,m}(x)$$

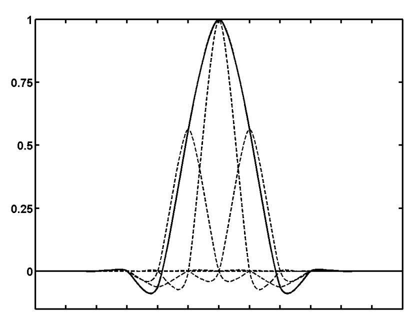
Wavelets From Subdivision

- Similarly, setting $u_{j,k} = \delta_{k,k'}$, $u_{j,m} = 0$

$$\begin{aligned} u_{j+1,k} &= u_{j,k} &&= \delta_{k,k'} \\ u_{j+1,m} &= \sum_{k\in N(j,m)} h_j[k,m]u_{j,k} &&= h_j[k',m] \end{aligned}$$

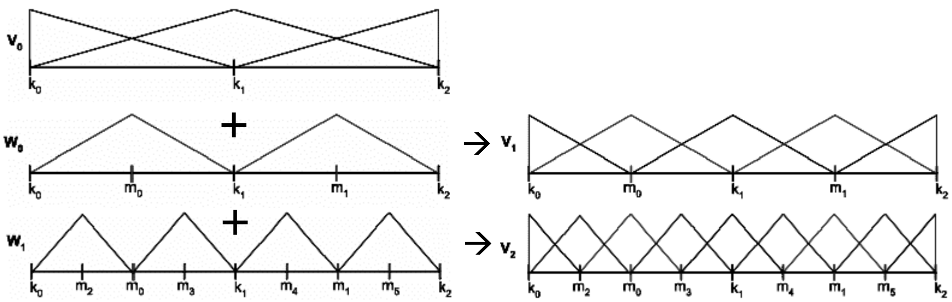
produces the refinement equation:

$$\varphi_{j,k}(x) = \varphi_{j+1,k}(x) + \sum_{m\in n(j,k)} h_j[k,m]\varphi_{j+1,m}(x)$$



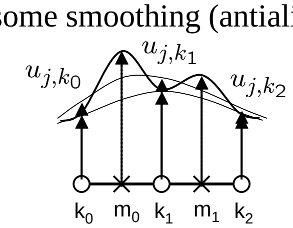
Wavelets From Subdivision

- So subdivision schemes naturally lead to hierarchical bases



Wavelets From Subdivision

- The coarsening strategy $u_{j,k} = u_{j+1,k}$ is generally less than ideal – some smoothing (antialiasing) desirable



Accomplished by forcing the wavelet to have one or more vanishing moments

$$\int w_{j,m}(x) x^k dx = 0 \ , \ k = 0, 1, \cdots, p-1$$

Larger p means smaller coefficients $u_{j,m}$ in wavelet series

$$f(x) \ = \sum_{k \in \mathcal{K}(j)} u_{j,k} \varphi_{j,k}(x) + \sum_{j=0}^{\infty} \sum_{m \in \mathcal{M}(j)} u_{j,m} w_{j,m}(x)$$

$$u_{j,m} \ \sim \ h_j^p f^{(p)}(x_m)$$

Wavelets From Subdivision

- How to improve wavelets using lifting

$$w^{new}_{j,m}(x) \;=\; w_{j,m}(x) - \sum_{k\in\mathcal{K}(j)} s_j[k,m]\varphi_{j,k}(x)$$

$\varphi_{j,k}(x)$ as before

tunable parameters

Choose $s_j[k,m]$ to make the moments zero.

- Regardless of the choice for $s_j[k,m]$, $\varphi_{j,k}(x)$ and $w^{new}_{j,m}(x)$ are orthogonal to the dual functions

$$\tilde{w}^{new}_{j,m}(x) \;=\; \tilde{\varphi}^{new}_{j+1,m}(x) - \sum_{k\in N(j,m)} h_j[k,m]\tilde{\varphi}^{new}_{j+1,k}(x)$$

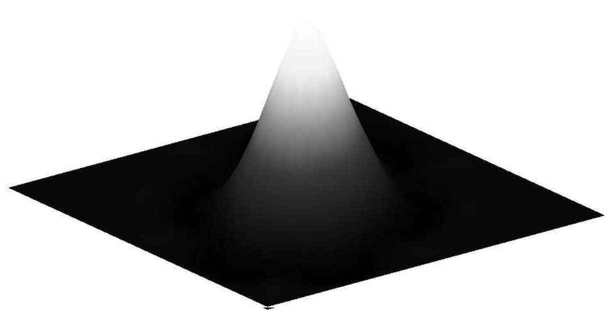
$$\tilde{\varphi}^{new}_{j,k}(x) \;=\; \tilde{\varphi}^{new}_{j+1,k}(x) + \sum_{m\in\mathcal{M}(j)} s_j[k,m]\tilde{w}^{new}_{j,m}(x)$$

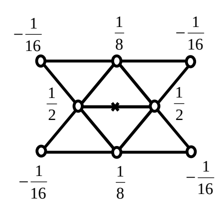
from which we obtain an improved coarsening strategy:

$$u_{j,m} = u_{j+1,m} - \sum_{k\in N(j,m)} h_j[k,m]u_{j+1,k}$$ Predict as before

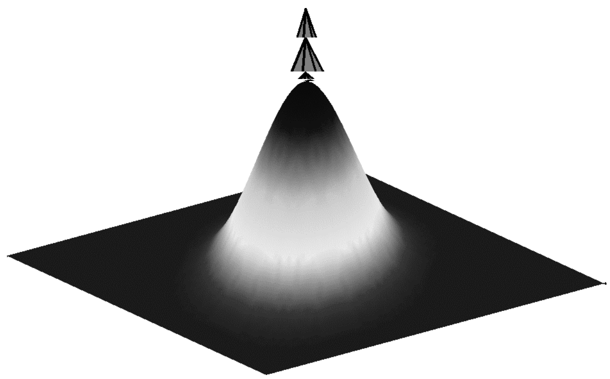
$$u_{j,k} = u_{j+1,k} + \sum_{m\in\mathcal{M}(j)} s_j[k,m]u_{j,m}$$ Then update

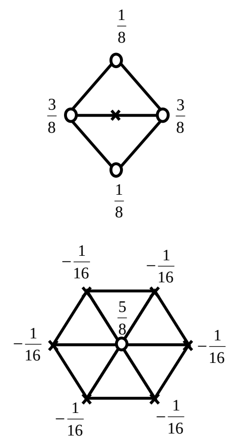
Butterfly Subdivision





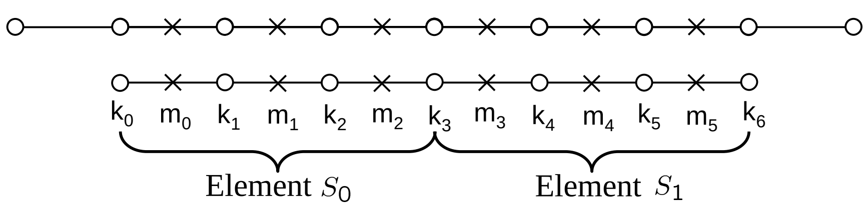
Loop Subdivision





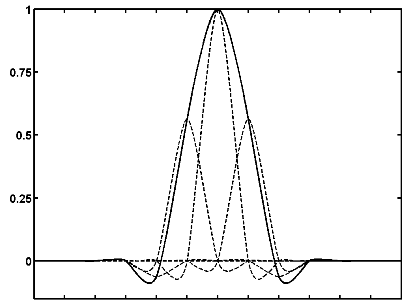
Finite Elements From Subdivision

- Key difference: subdivision mask is varied so that prediction operation is confined *within* an element



- Limit functions are finite element shape functions

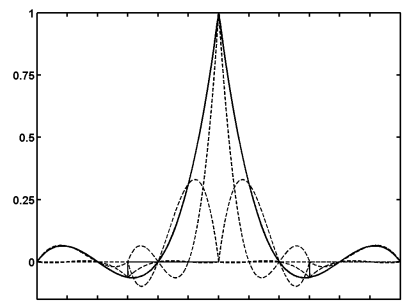
Finite Elements From Subdivision



Scalar subdivision

$$\frac{1}{16}\{-1, 0, 9, 16, 9, 0, -1\}$$

Finite Element generated from vector subdivision - piecewise polynomial, but lacks smoothness at element boundaries



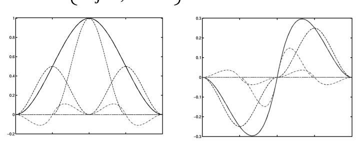
Smoother vector subdivision schemes also possible

Vector Refinement

- e.g. vector refinement relation for Hermite interpolation functions

$$\begin{Bmatrix} \varphi^u_{j,k}(x) \\ \varphi^\theta_{j,k}(x) \end{Bmatrix} = \begin{Bmatrix} \varphi^u_{j+1,k}(x) \\ \varphi^\theta_{j+1,k}(x) \end{Bmatrix} + \sum_{m\in n(j,k)} \mathbf{H}_j[k,m] \begin{Bmatrix} \varphi^u_{j+1,m}(x) \\ \varphi^\theta_{j+1,m}(x) \end{Bmatrix}$$

$$\mathbf{H}_j[k,m] = \begin{bmatrix} \varphi^u_k(x_m) & \frac{d\varphi^u_k(x_m)}{dx} \\ \varphi^\theta_k(x_m) & \frac{d\varphi^\theta_k(x_m)}{dx} \end{bmatrix}$$



Cubic subdivision for displacements and rotations

- Wavelets

$$\begin{Bmatrix} w^u_{j,m}(x) \\ w^\theta_{j,m}(x) \end{Bmatrix} = \begin{Bmatrix} \varphi^u_{j+1,m}(x) \\ \varphi^\theta_{j+1,m}(x) \end{Bmatrix} - \sum_{k\in A(j,m)} \mathbf{S}^T_j[k,m] \begin{Bmatrix} \varphi^u_{j,k}(x) \\ \varphi^\theta_{j,k}(x) \end{Bmatrix}$$

$$\sum_{k\in A(j,m)} \left[\int_S x^i \left\{ \varphi^u_{j,k}(x) \quad \varphi^\theta_{j,k}(x) \right\} dx \right] \mathbf{S}_j[k,m] = \int_S x^i \left\{ \varphi^u_{j+1,m}(x) \quad \varphi^\theta_{j+1,m}(x) \right\} dx$$