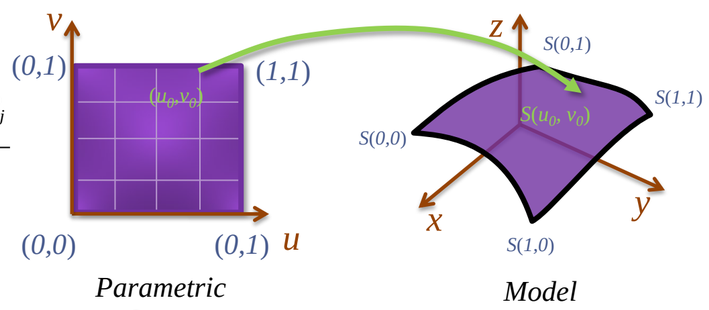
NURBS Evaluation and Rendering

Adarsh Krishnamurthy, Sushrut Pavanaskar

Non Uniform Rational B-Splines

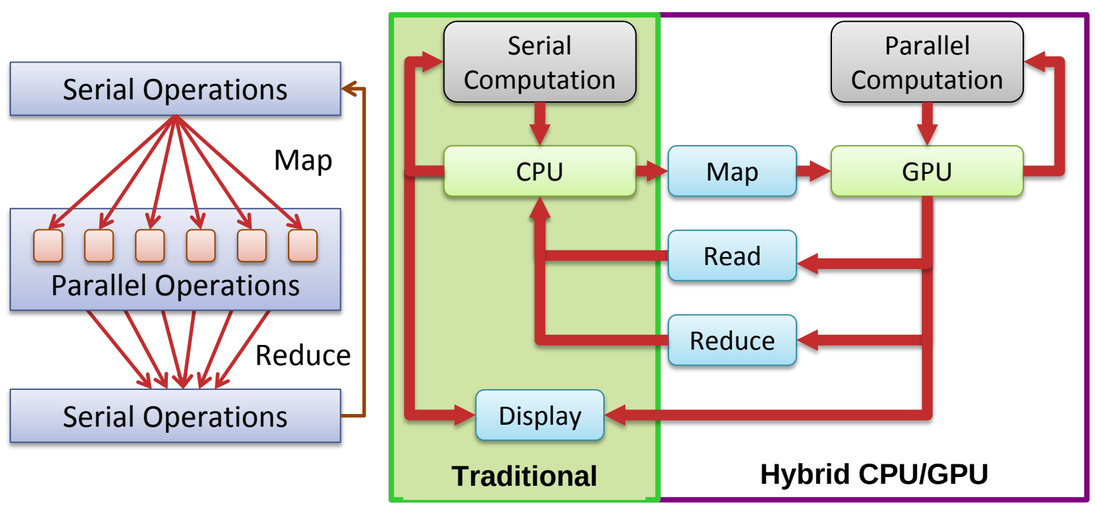
- Piecewise-polynomial tensor product surfaces
- Most general spline surface
  - De facto surface representation for CAD
  - Can be used to represent Bezier and b-spline surfaces
- Compact definition
  - Control points
  - Knot vectors for u and v parametric directions

$$S(u,v) = \frac{\sum_{j=0}^{m}\sum_{i=0}^{n} N_i^p(u) N_j^q(v) w_{ij} P_{ij}}{\sum_{j=0}^{m}\sum_{i=0}^{n} N_i^p(u) N_j^q(v) w_{ij}}$$



Hybrid CPU/GPU Computations

- Extensive use of Map-Reduce
- Interoperability between CUDA, OpenGL, and Cg
- Direct rendering using Vertex Buffer Objects (VBOs)

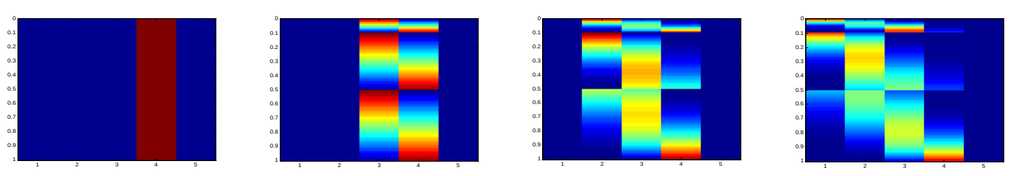


Basis Function Evaluation

- First-order basis function is the step function
- Calculate higher order basis functions from lower order basis functions in parallel
- Compute k^{th}-order basis function in k-1 parallel passes

$$N_i^p(u) = \frac{u-u_i}{u_{i+p}-u_i} N_i^{p-1}(u) + \frac{u_{i+p+1}-u}{u_{i+p+1}-u_{i+1}} N_{i+1}^{p-1}(u), \quad \frac{0}{0} = 0$$

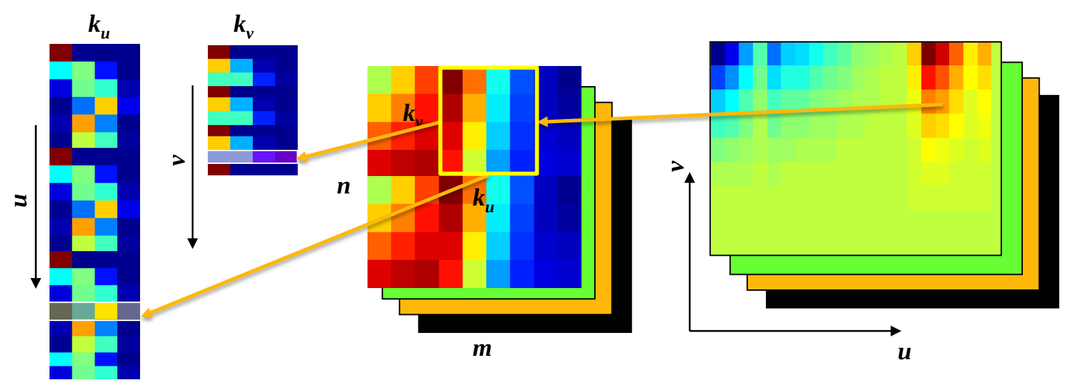
$$N_i^0(u) = \begin{cases} 1 & \text{if } u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$



Output during intermediate passes while computing a cubic basis function

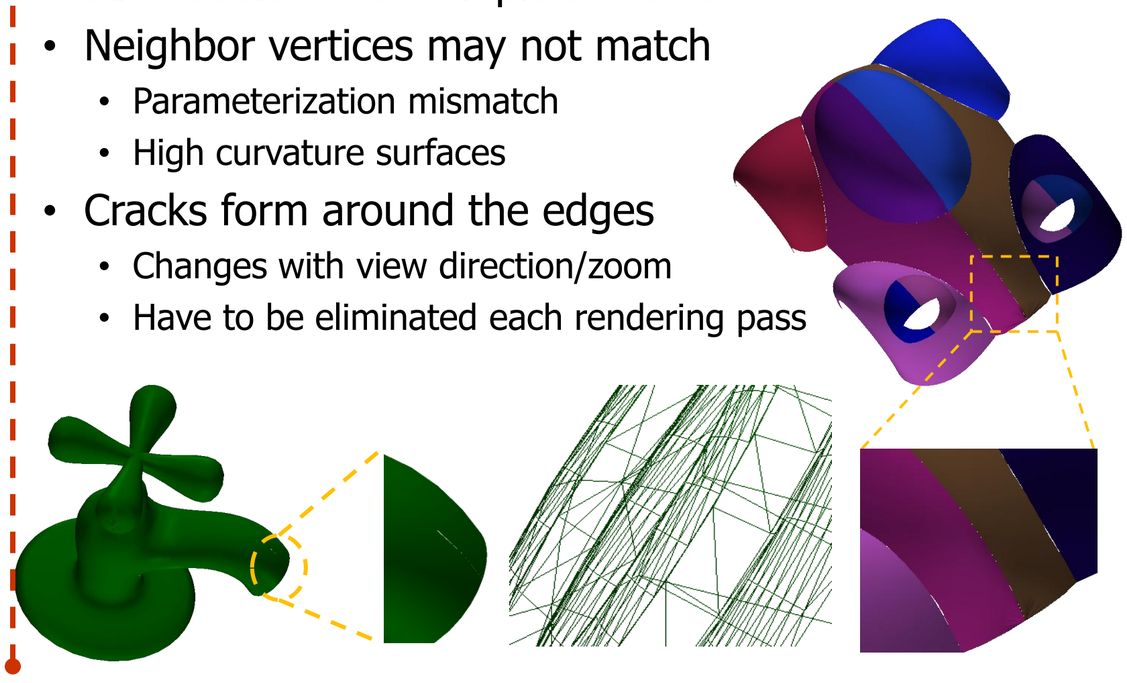
Surface Evaluation

- Locate the sub-mesh of control points that influence the evaluation point coordinates
- Compute basis functions along u and v directions
- Multiply the basis functions with their corresponding control points from the sub-mesh



Boundary Cracks in Models

- 3D models have multiple surfaces
- Neighbor vertices may not match
  - Parameterization mismatch
  - High curvature surfaces
- Cracks form around the edges
  - Changes with view direction/zoom
  - Have to be eliminated each rendering pass



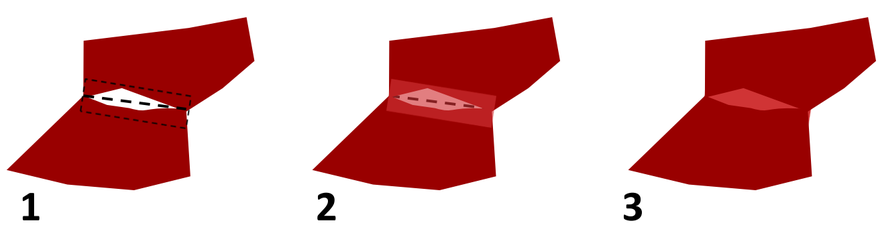
Crack Elimination by Thick Edges

- Process each edge in parallel
- Draw quads around edges, parallel to view plane
- Detect cracks by comparing surface IDs
- Render only "crack" pixels

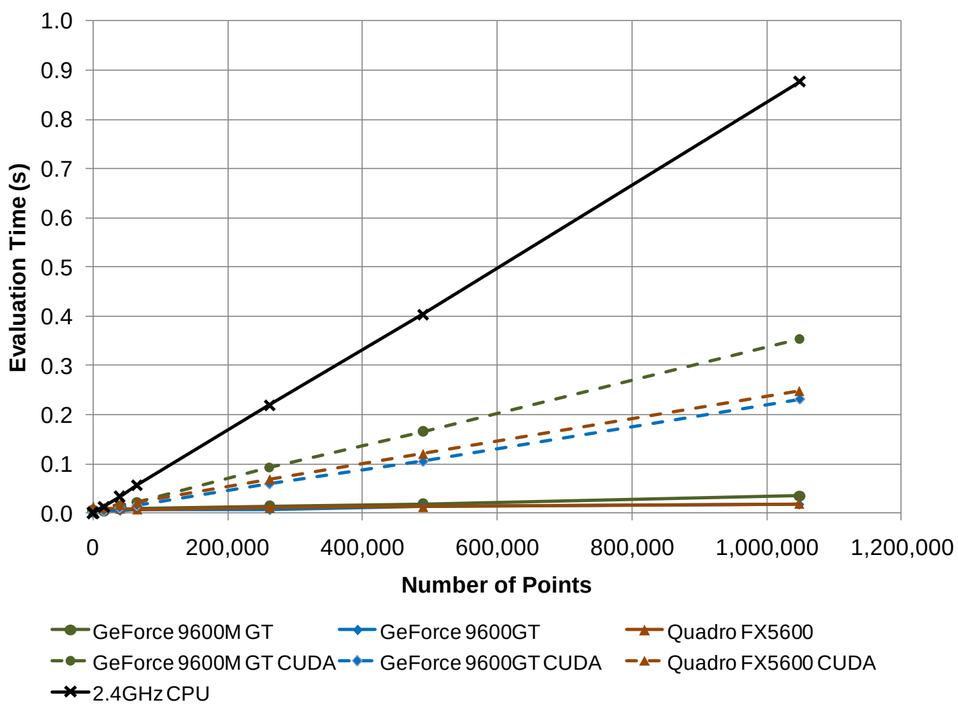
Algorithm

1. Calculate edge quad location and dimension
2. Invoke fragment program on the edge quad
3. GPU based comparison

   Pixels other than crack "discarded"; cracks filled iff present



Preliminary Timing Results



Sample Model Renderings