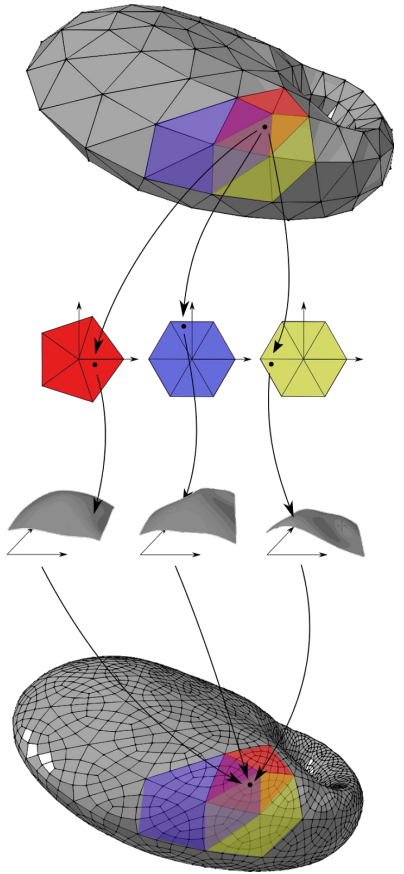
CIS 565 Milestone 2 - PPM Optimization

Daniel Krupka and Gabriel Naghi

A Review

- PPMs are a hybrid of Bezier surfaces + triangular meshes
- PPMs approximate given geometry via sampling
- PPMs are parallelizable by surface patch
- PPMs have preprocessed topology + updated geometry



Inner Loop Optimization

- Most data (vertex positions, normals, UVs) processed here
- Need to regenerate samples affected by base mesh changes
  - E.g. deformations, currently placeheld

Base Case

- Heavy computation to get sample coordinates
- One large matrix multiplication to get Bezier coefficients
- Heavy memory to access+update new vertex data

Sample Coordinates

- Sampling pattern is constant (but not compile-time)
- Precompute + store in textures
- Small but definite improvement

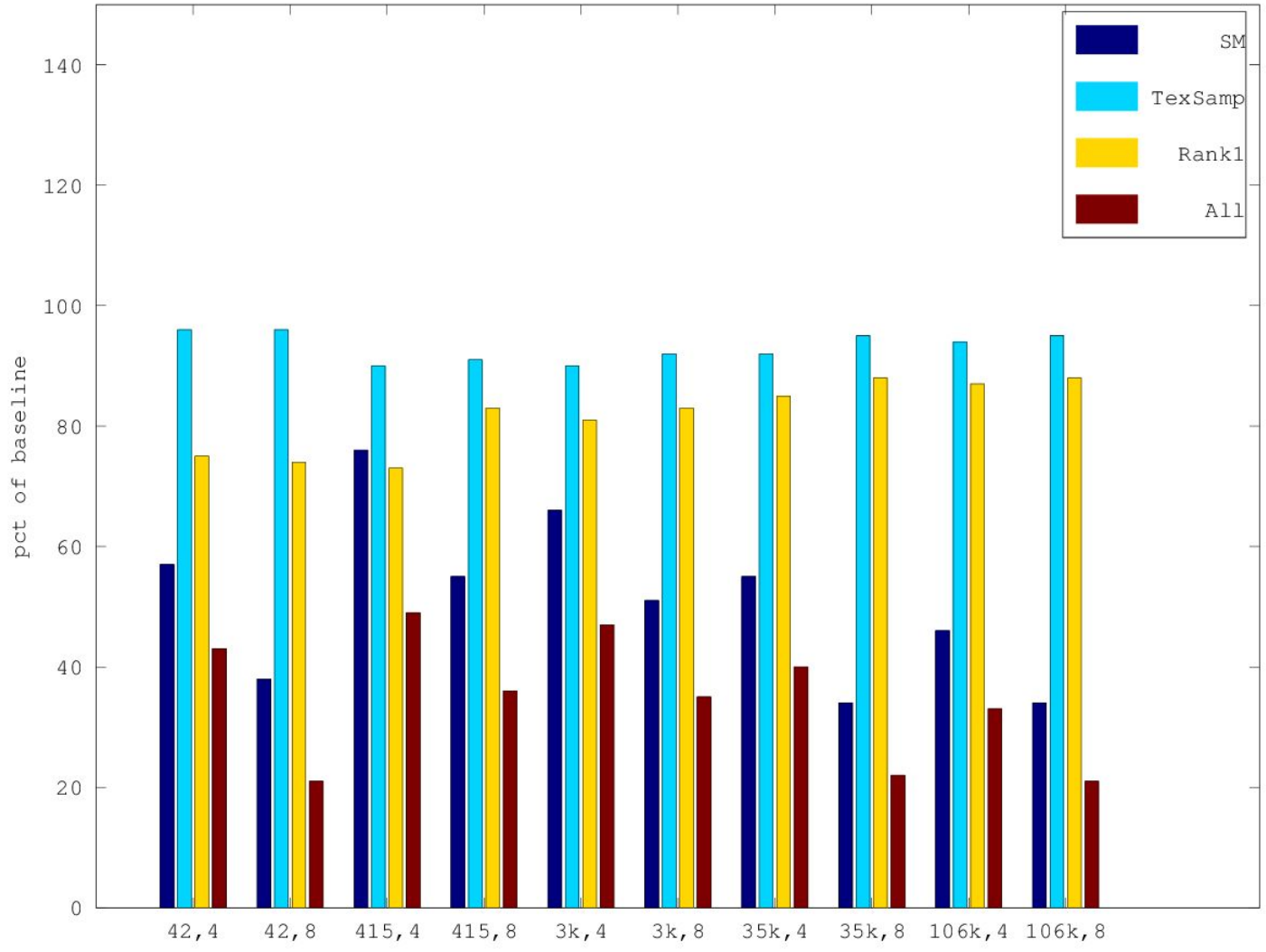
Matrix Multiplication

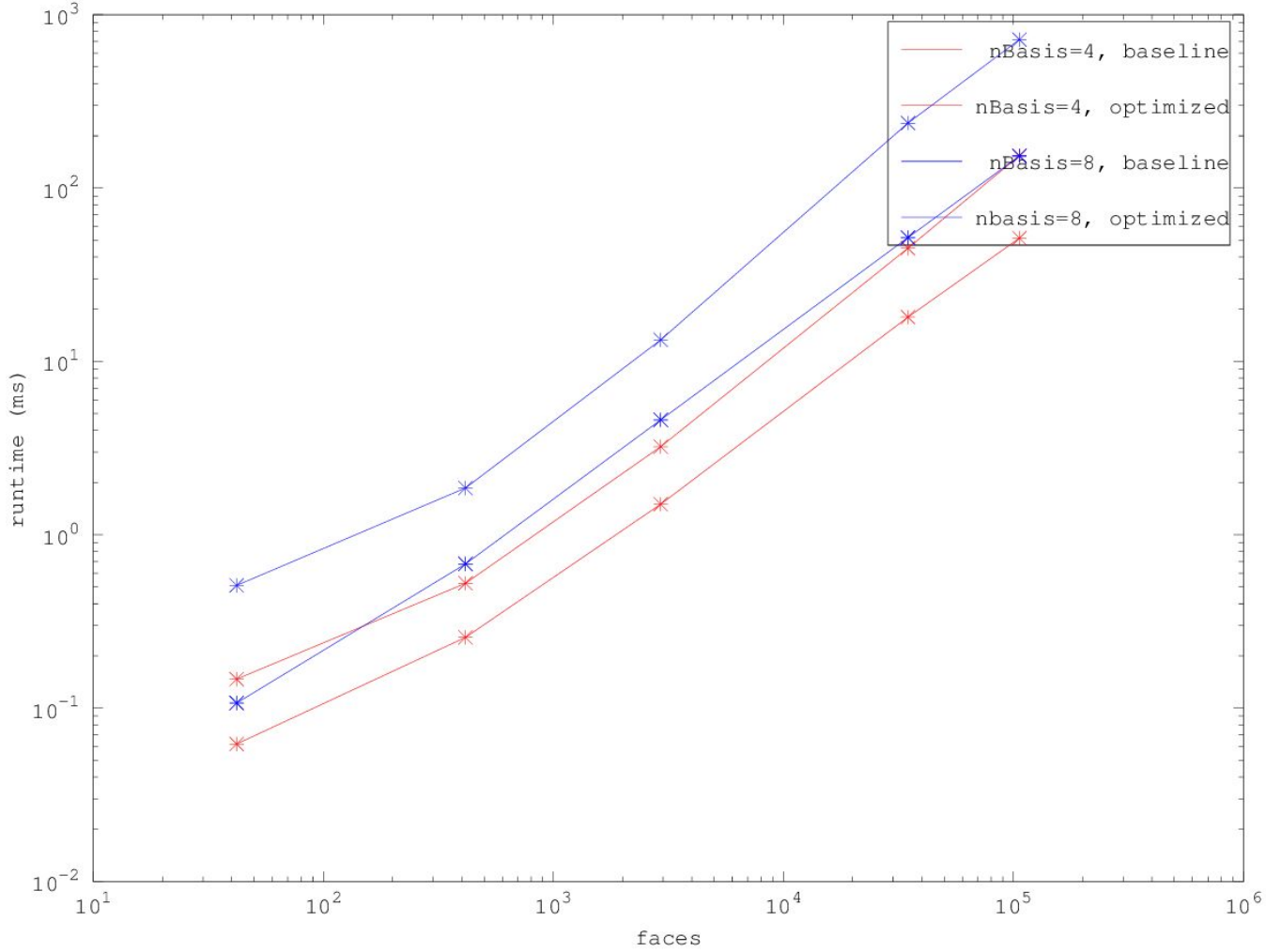
- $O(N^3)$ naively, but well known parallelization problem - CuBLAS gemm
- Use SVD structure of matrix to reduce to rank-1 update
- $O(N^2)$ and embarrassingly parallel - CuBLAS gerv
- Slightly bigger improvement

Shared Memory

- Bezier coefficients same for entire triangle, load to SM first
- Parallelize by face (block.X) and subvertex (block.Y)
- First try: (X,0) threads load SM in loop
- Better: Y threads ‘repurposed’ to parallelize loading loop
- Substantial improvement - 2-3x alone

Verts/Tris/BezDeg	No Opt	SM	TexSamp	Rank-1	All Opt
42/80/4	.147	.084 (57%)	.141 (96%)	.106 (75%)	.062 (43%)
42/80/8	.510	.196 (38%)	.490 (96%)	.379 (74%)	.107 (21%)
415/824/4	.525	.397 (76%)	.475 (90%)	.383 (73%)	.256 (49%)
415/824/8	1.86	1.03 (55%)	1.70 (91%)	1.55 (83%)	.678 (36%)
3k/6k/4	3.22	2.13 (66%)	2.91 (90%)	2.60 (81%)	1.50 (47%)
3k/6k/8	13.3	6.72 (51%)	12.3 (92%)	11.0 (83%)	4.60 (35%)
35k/69k/4	45.0	24.8 (55%)	41.6 (92%)	38.2 (85%)	18.0 (40%)
35k/69k/8	237	80.9 (34%)	225 (95%)	208 (88%)	78.8 (33%)
106k/212k/4	154	71.8 (46%)	145 (94%)	134 (87%)	51.4 (33%)
106k/212k/8	718	242 (34%)	681 (95%)	629 (88%)	178 (25%)


SM
TexSamp
Rank1
All
pct of baseline
0
20
40
60
80
100
120
140
42,4
42,8
415,4
415,8
3k,4
3k,8
35k,4
35k,8
106k,4
106k,8


nBasis=4, baseline
nBasis=4, optimized
nBasis=8, baseline
nbasis=8, optimized
runtime (ms)
faces
10^{3}
10^{2}
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}

CPU Baseline?

- Two seconds for the smallest model
- Horror story from there on (~3 minutes for largest!)

Next Steps

- Full profiling
- Deformation