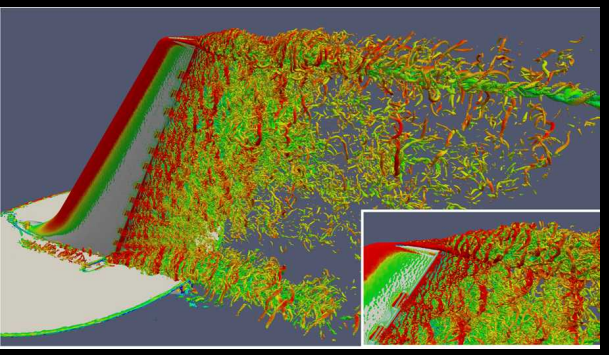
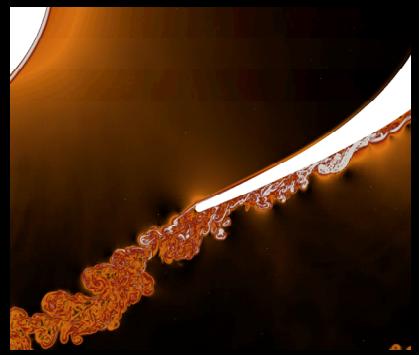
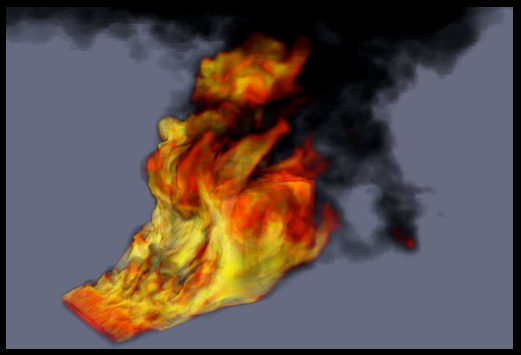
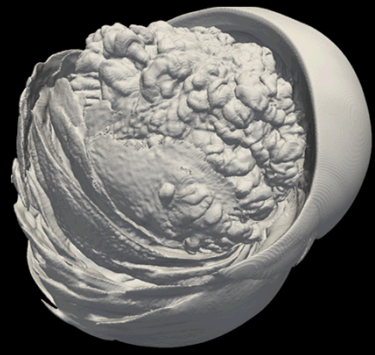
SAND2017-8530PE

High Performance Visualization in the Many-Core Era

Kenneth Moreland Sandia National Laboratories

Computing@PNNL Seminar

August 11, 2017

CCR Center for Computing Research

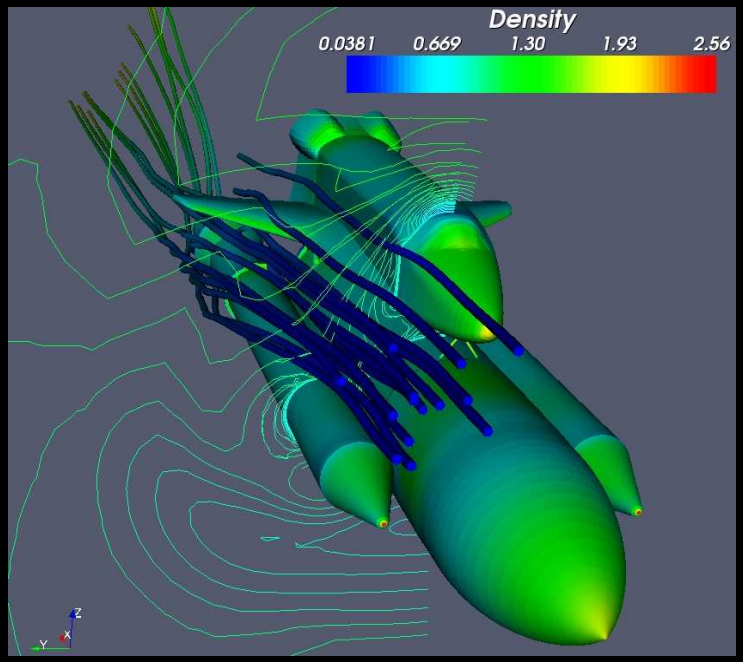
Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2015-4835 PE

Acknowledgements

- This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, under Award Numbers 10-014707, 12-015215, and 14-017566.
- This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering, and early testbed platforms, in support of the nation's exascale computing imperative.
- Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.
- **Thanks to many, many partners in labs, universities, and industry.**

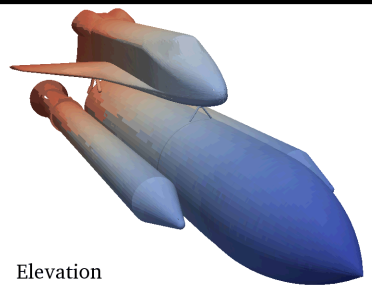
Basics of Visualization

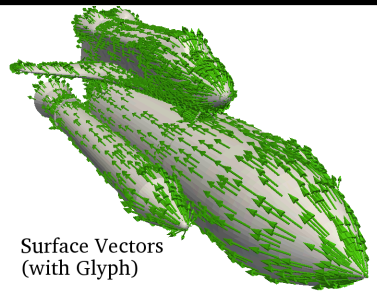
```
0265640 132304 133732 032051 037334 024721 015013 052226 001662
0265660 025537 064663 054606 043244 074076 124153 135216 126614
0265700 144210 056426 044700 042650 165230 137037 003655 006254
0265720 134453 124327 176005 027034 107614 170774 073702 067274
0265740 072451 007735 147620 061064 157435 113057 155356 114603
0265760 107204 102316 171451 046040 120223 001774 030477 046673
0266000 171317 116055 155117 134444 167210 041405 147127 050505
0266020 004137 046472 124015 134360 173550 053517 044635 021135
0266040 070176 047705 113754 175477 105532 076515 177366 056333
0266060 041023 074017 127113 003214 037026 037640 066171 123424
0266100 067701 037406 140000 165341 072410 100032 125455 056646
0266120 006716 071402 055672 132571 105645 170073 050376 072117
0266140 024451 007424 114200 077733 024434 012546 172404 102345
0266160 040223 050170 055164 164634 047154 126525 112514 032315
0266200 016041 176055 042766 025015 176314 017234 110060 014515
0266220 117156 030746 154234 125001 151144 163706 136237 164376
0266240 137055 062276 161755 115466 005322 132567 073216 002655
0266260 171466 126161 117155 065763 016177 014460 112765 055527
0266300 003767 175367 104754 036436 172172 150750 043643 145410
0266320 072074 000007 040627 070652 173011 002151 125132 140214
0266340 060115 014356 015164 067027 120206 070242 033065 131334
0266360 170601 170106 040437 127277 124446 136631 041462 116321
0266400 020243 005602 004146 121574 124651 006634 071331 102070
0266420 157504 160307 166330 074251 024520 114433 167273 030635
0266440 133614 106171 144160 010652 007365 026416 160716 100413
0266460 026630 007210 000630 121224 076033 140764 000737 003276
0266500 114060 042647 104475 110537 066716 104754 075447 112254
0266520 030374 144251 077734 015157 002513 173526 035531 150003
0266540 146207 015135 024446 130101 072457 040764 165513 156412
0266560 166410 067251 156160 106406 136770 030516 064740 022032
0266600 142166 123707 175121 071170 076357 037233 031136 015232
0266620 075074 016744 044055 102230 110063 033350 052765 172463
```

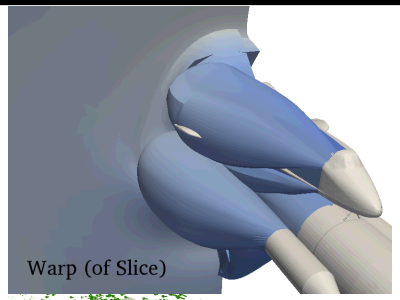


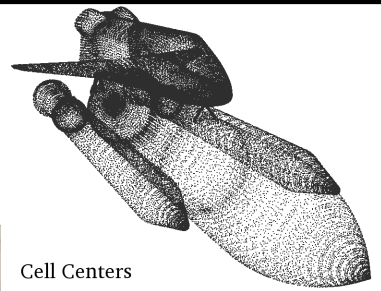
Visualization is Not Just Rendering

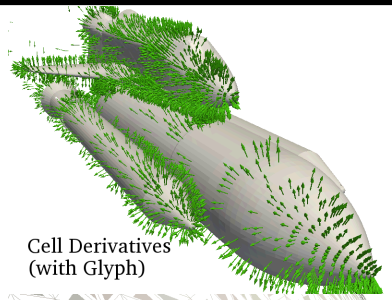
Sandia National Laboratories

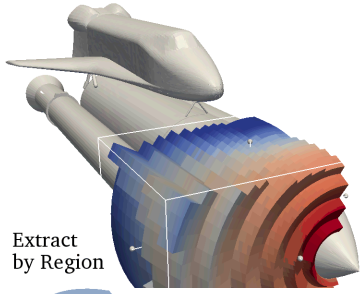

Elevation

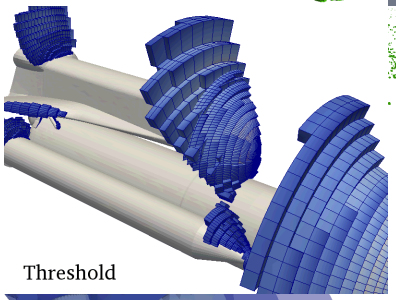

Surface Vectors (with Glyph)

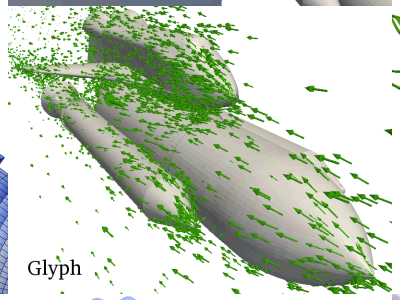

Warp (of Slice)

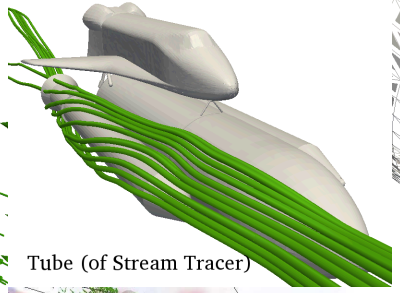

Cell Centers


Cell Derivatives (with Glyph)

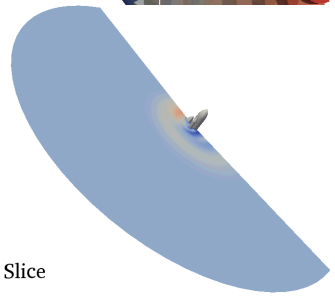

Extract by Region

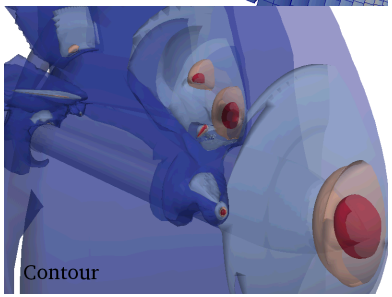

Threshold

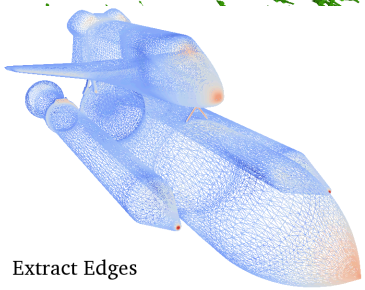

Glyph

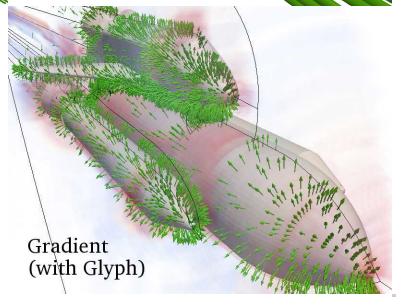

Tube (of Stream Tracer)

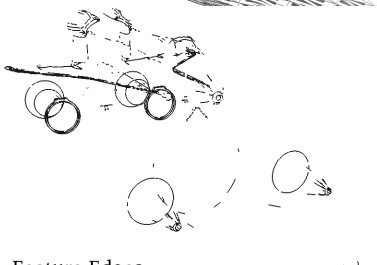

Shrink

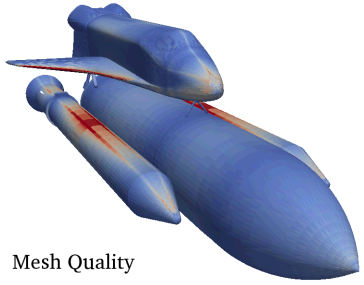

Slice

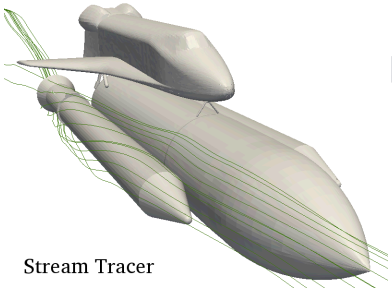

Contour

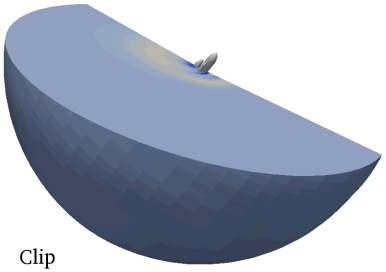

Extract Edges

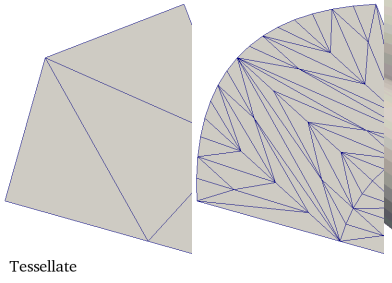

Gradient (with Glyph)

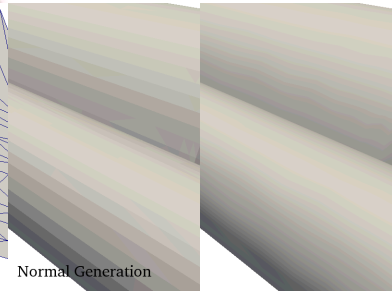

Feature Edges


Mesh Quality


Stream Tracer


Clip


Tessellate

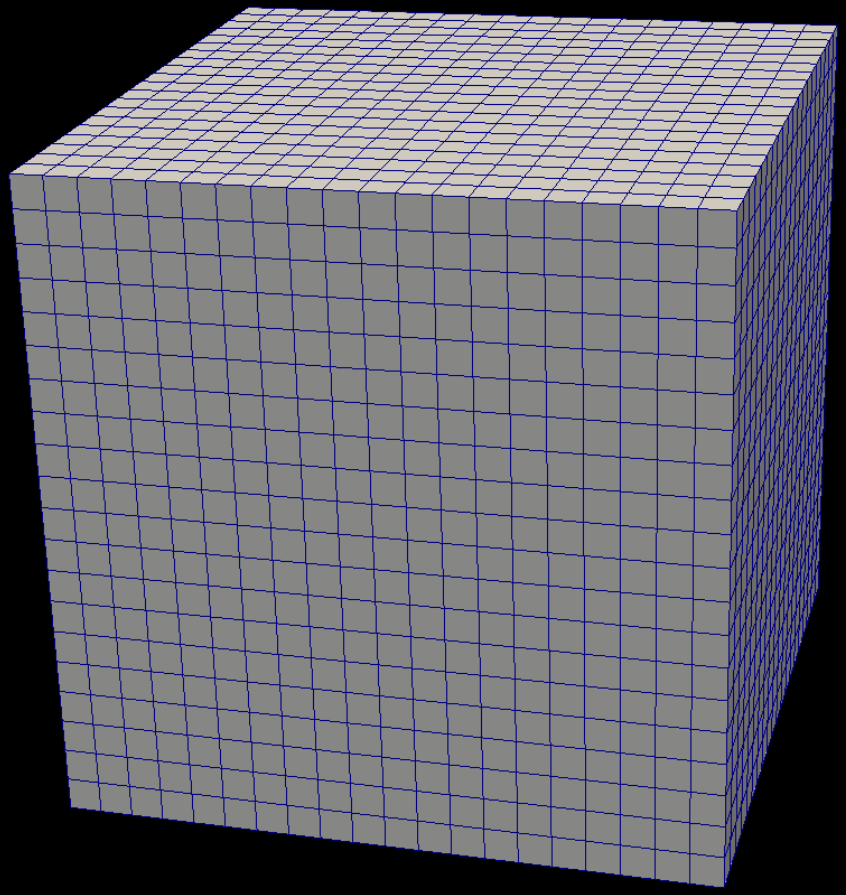

Normal Generation

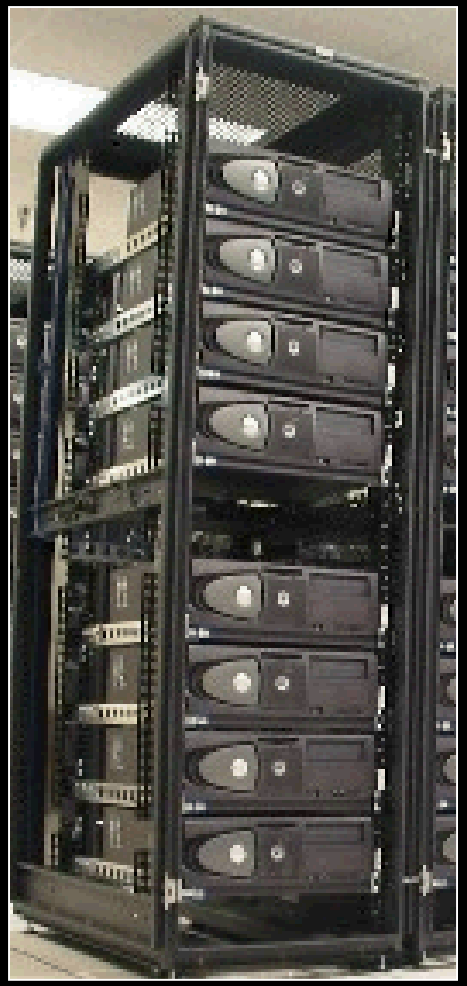

A Brief History of ~~Time~~
Interactive Visualization

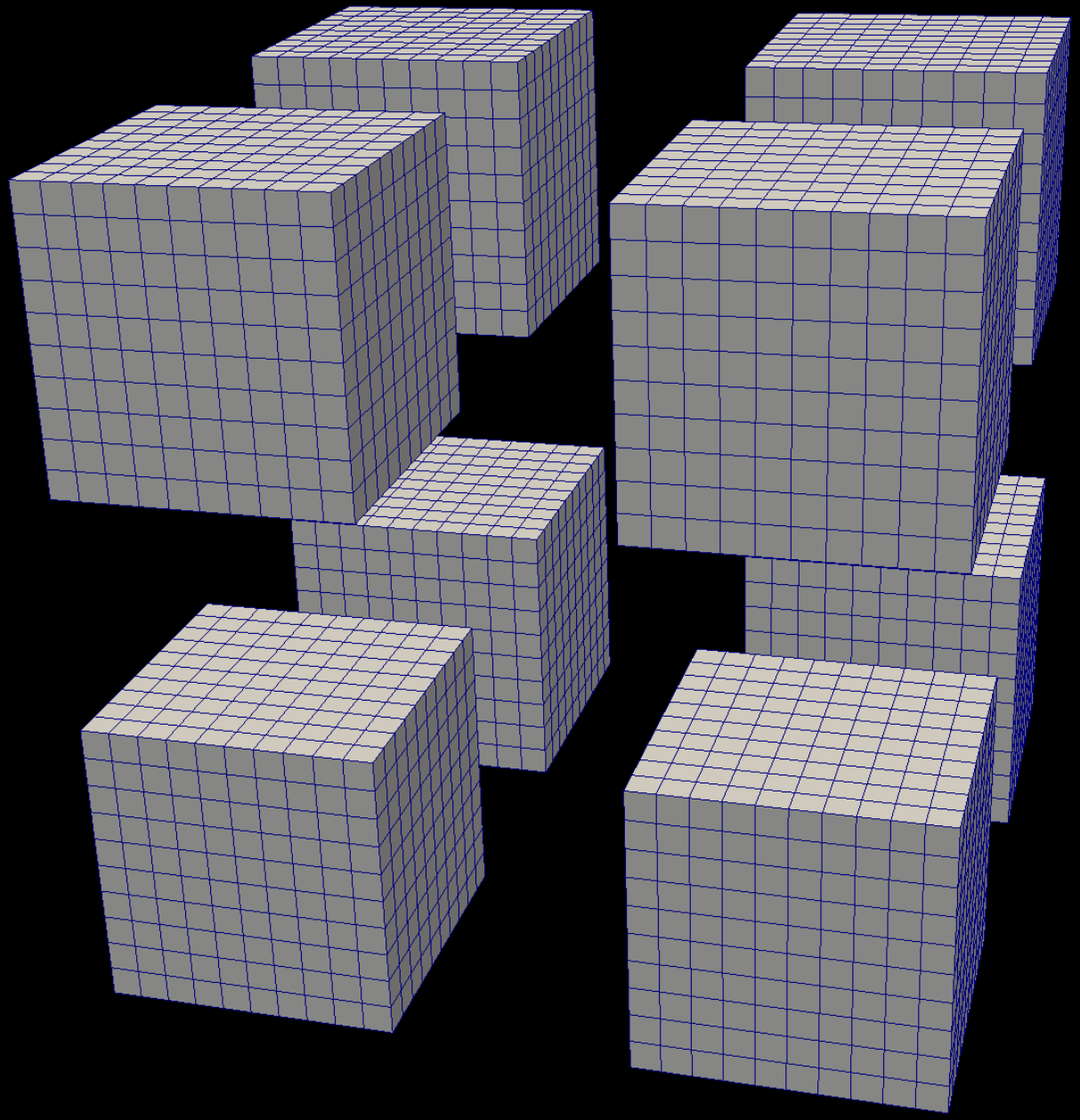
The Big Iron Era (circa 1995)

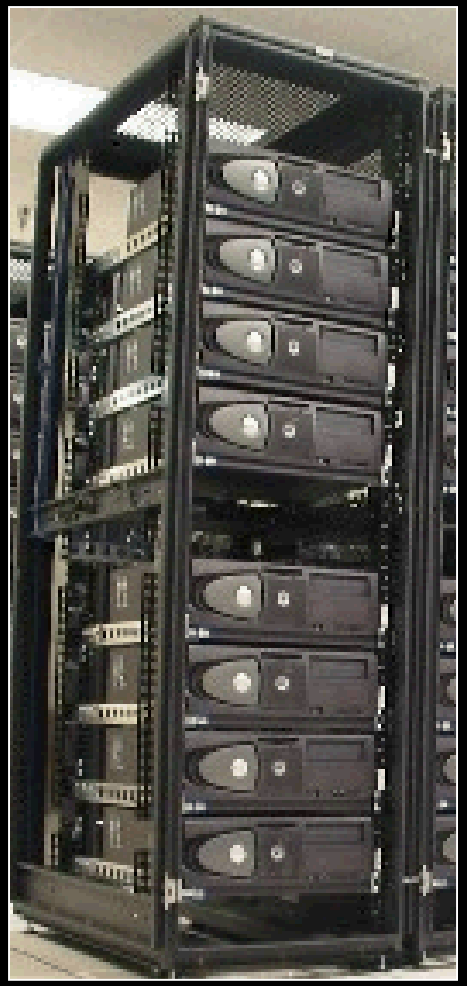
GPU Cluster Era (circa 2002)

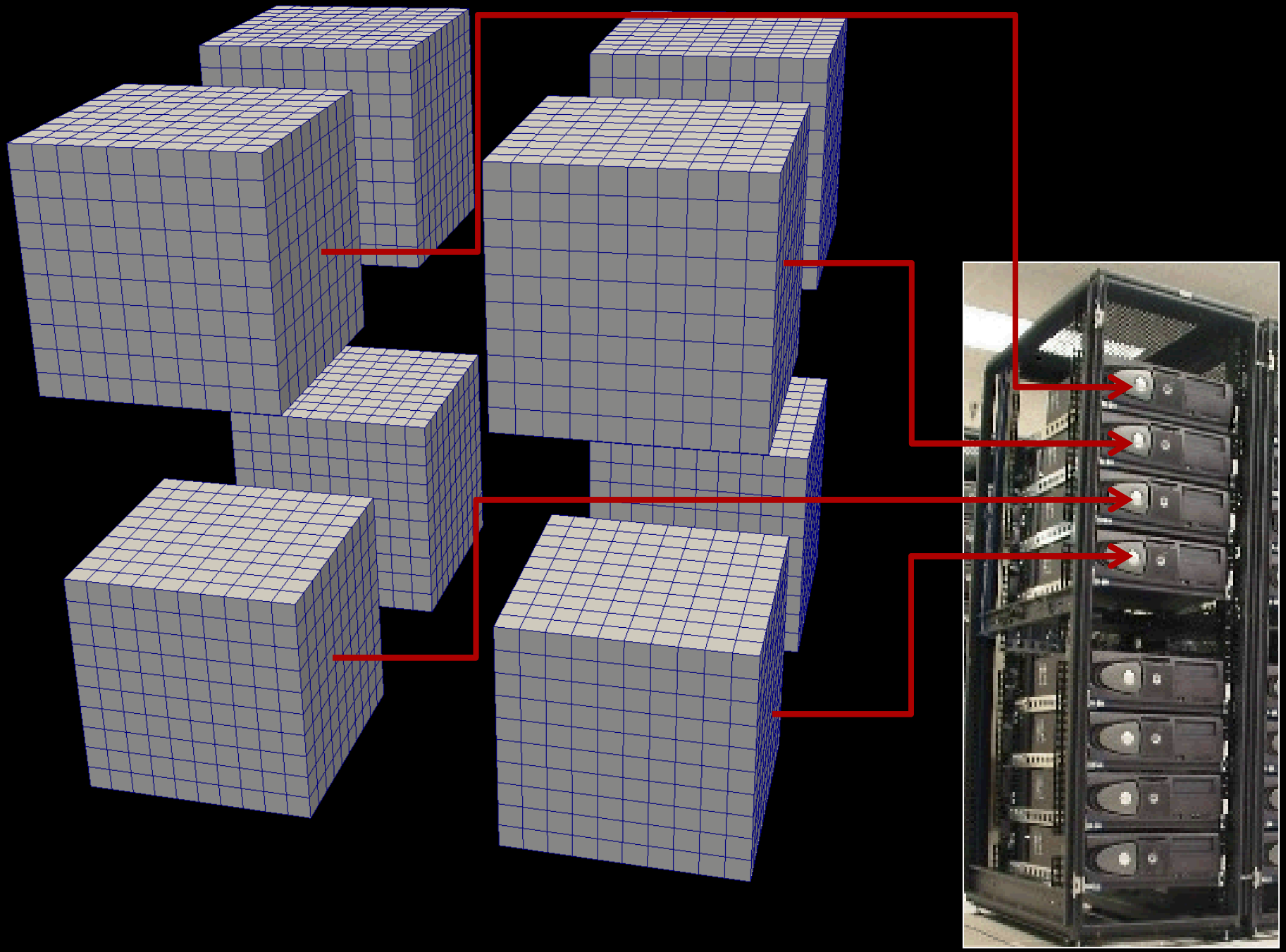
Sandia National Laboratories

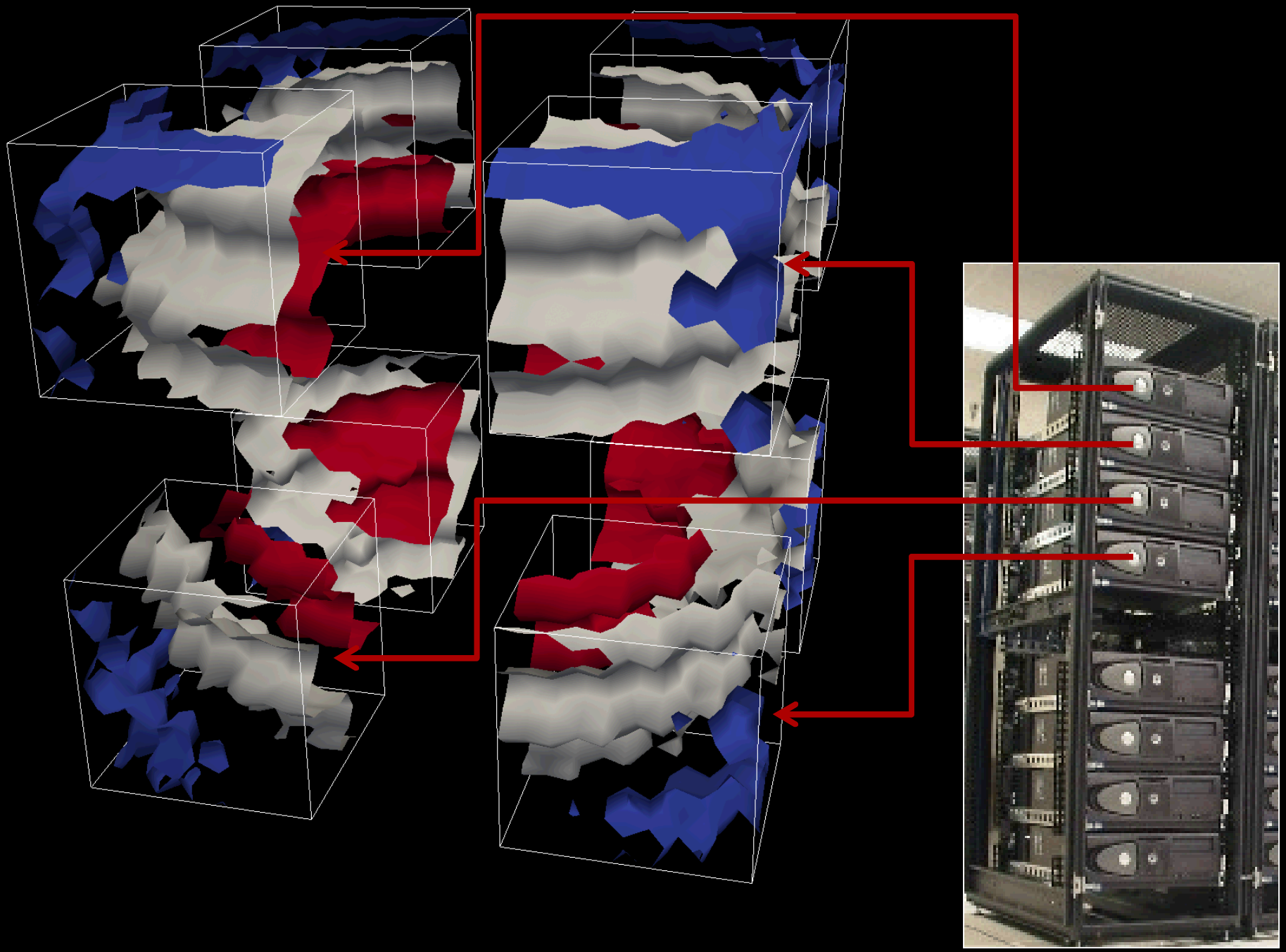




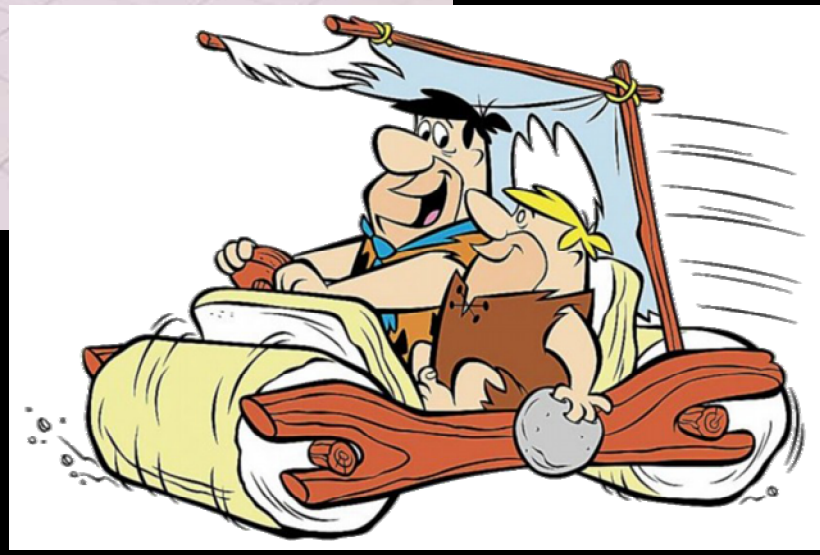






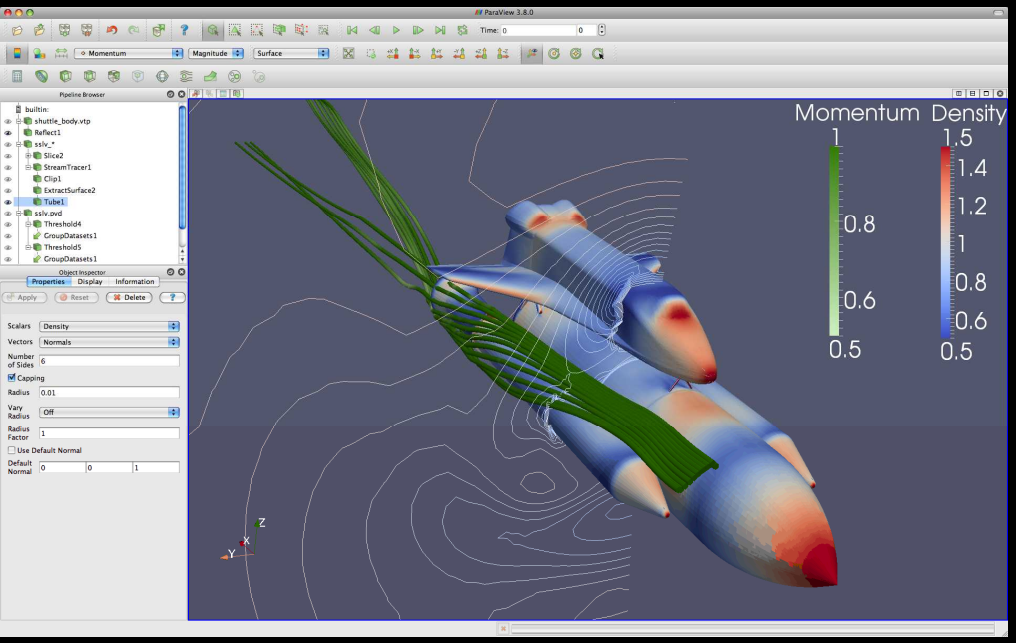


Distributed Memory Supercomputing (circa last week)



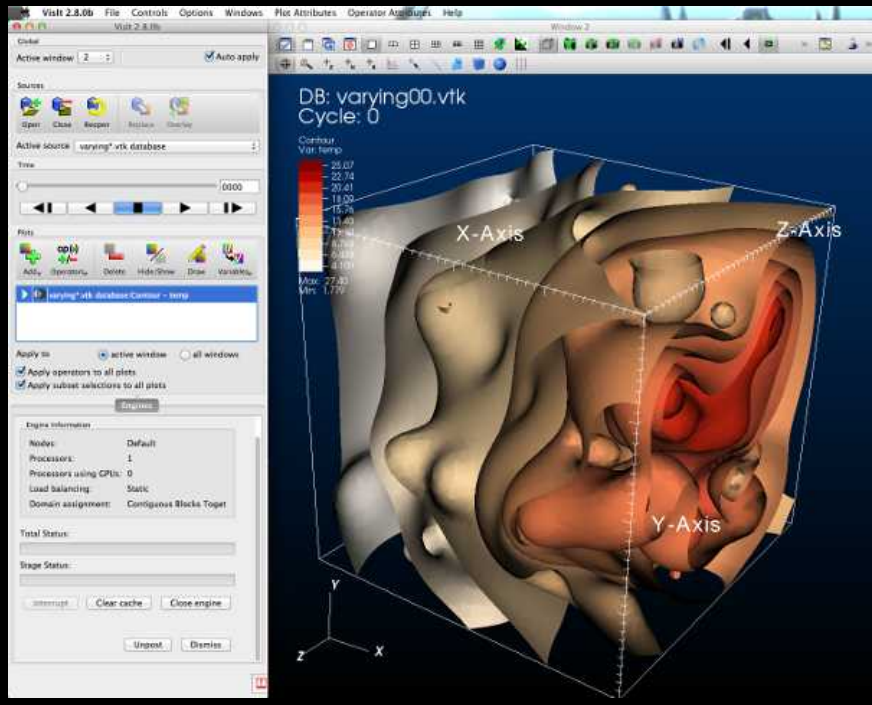
Encapsulating Large-Scale Visualization in Production Tools

Sandia National Laboratories



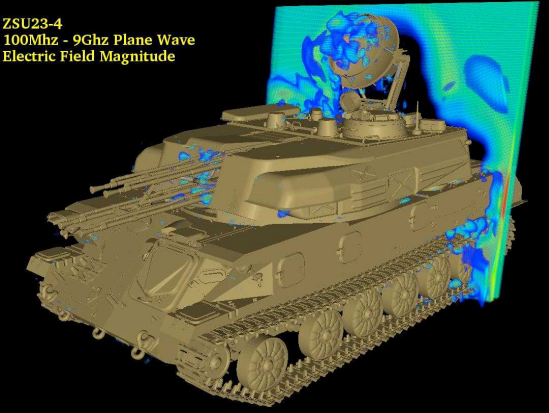
ParaView

http://paraview.org

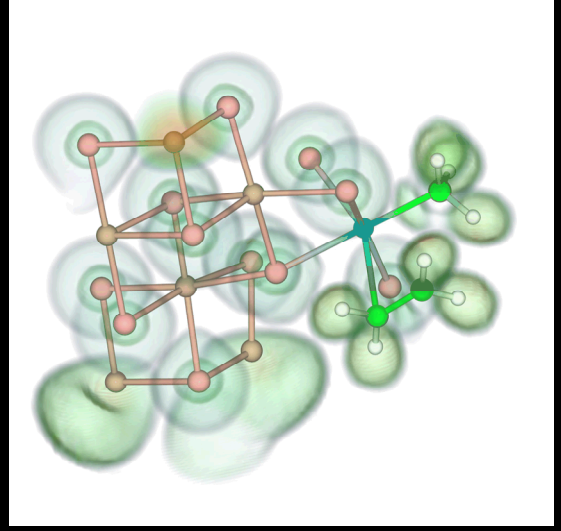


VisIt

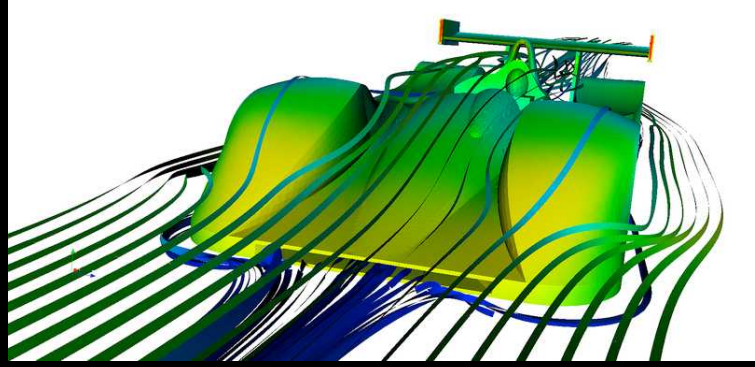
http://visit.llnl.gov/



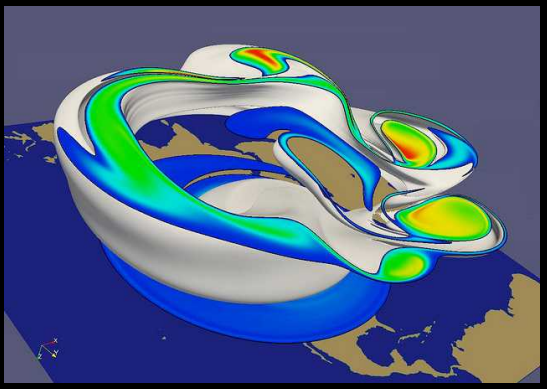
Jerry Clarke, US Army Research Laboratory

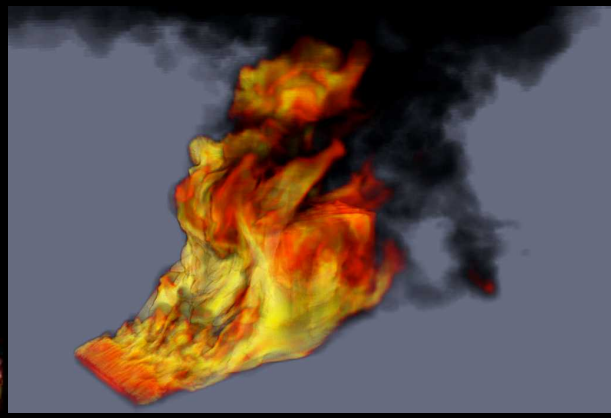


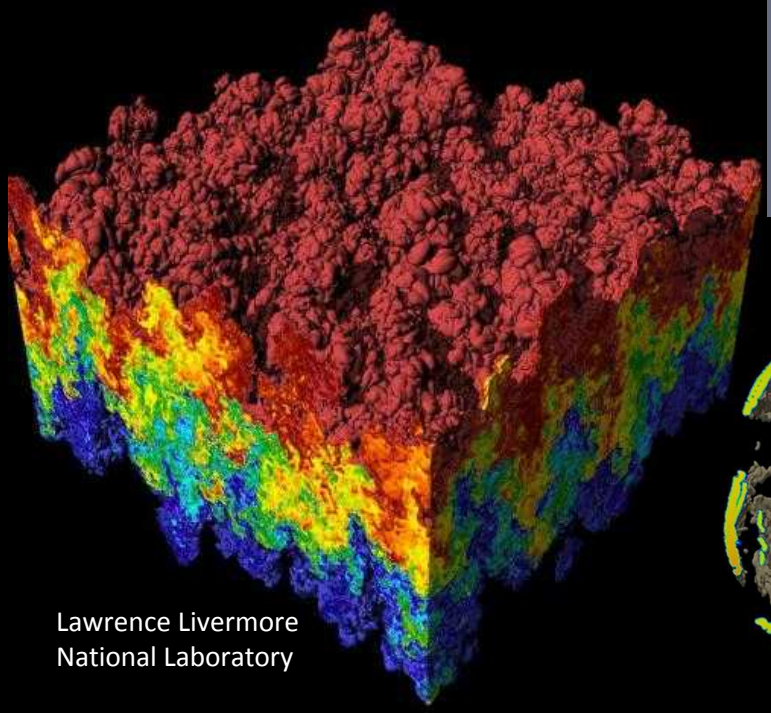
Swiss National Supercomputing Centre



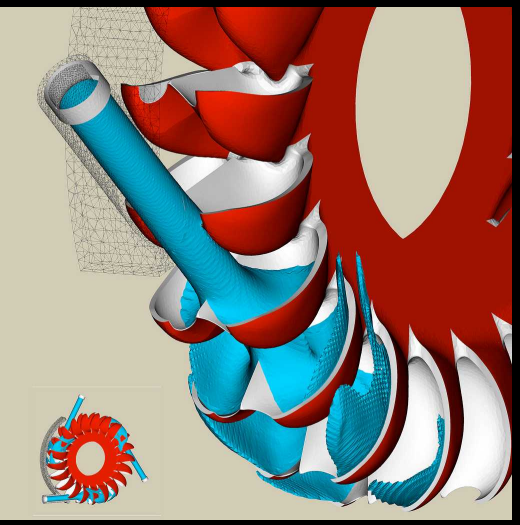
Renato N. Elias, NACAD/COPPE/UFRJ, Rio de Janerio, Brazil



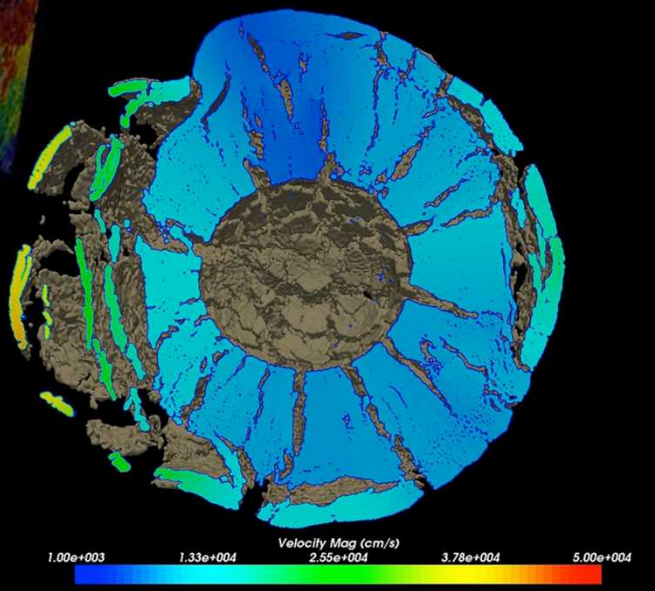


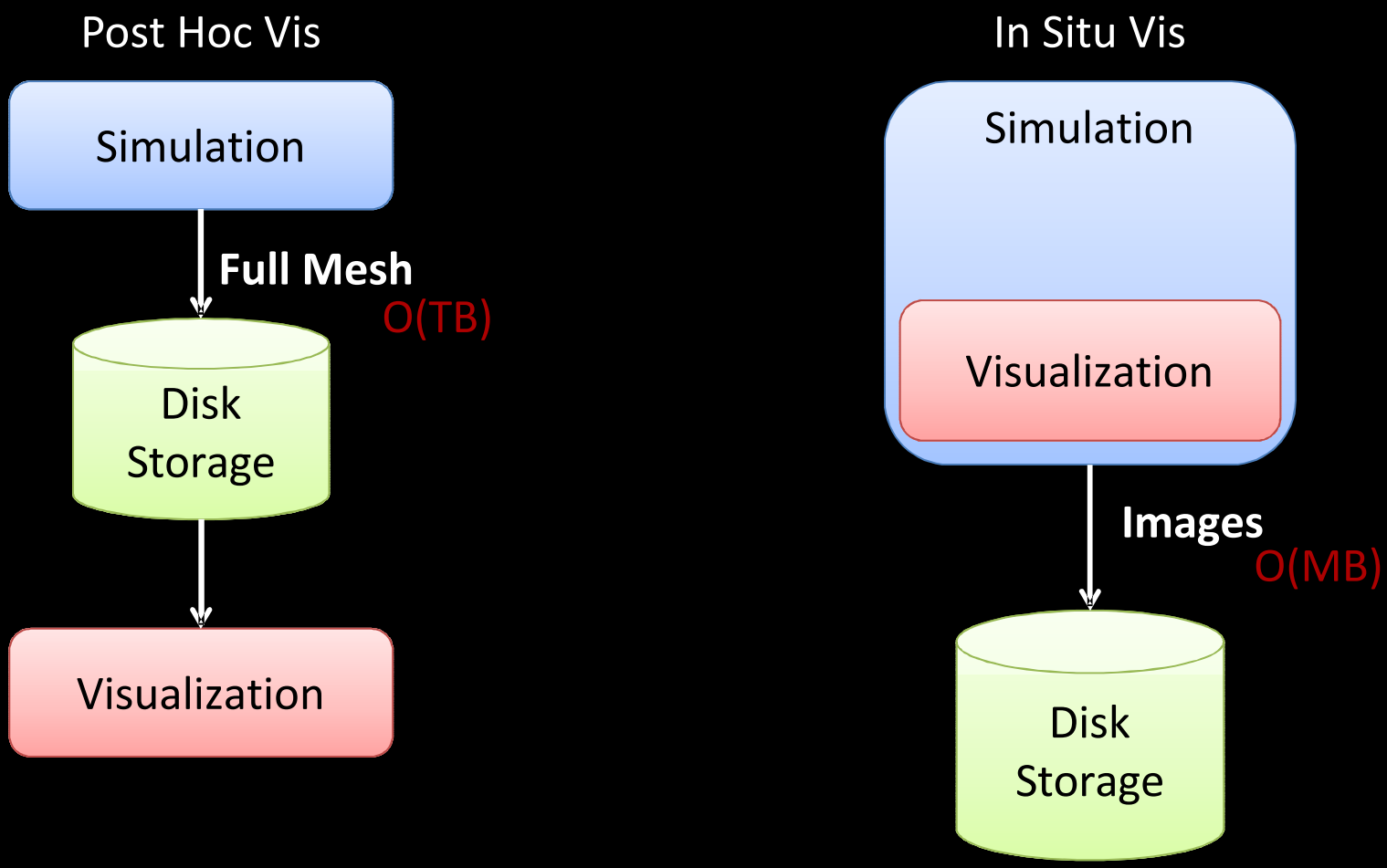


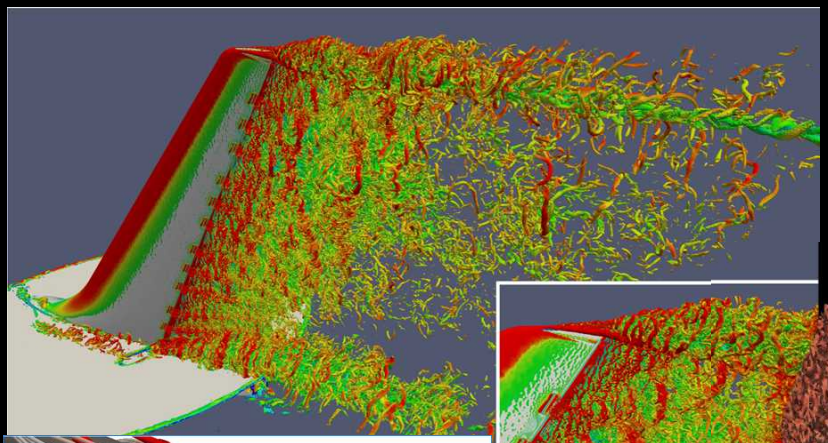
Lawrence Livermore
National Laboratory

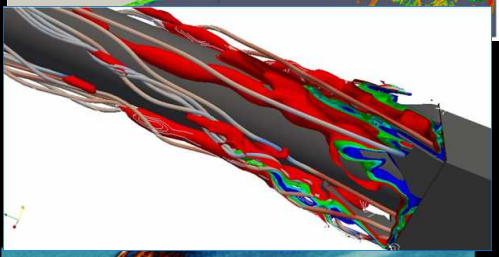


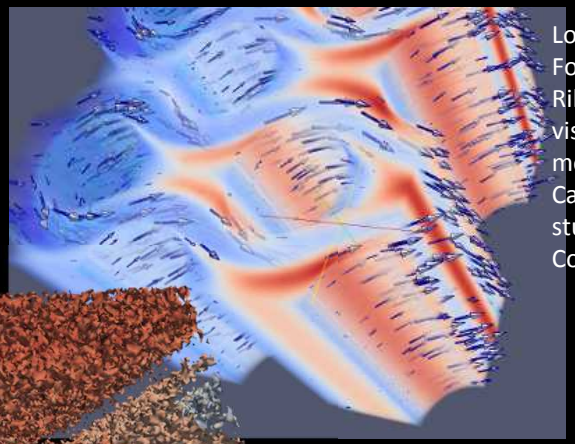
Swiss National Supercomputing Centre



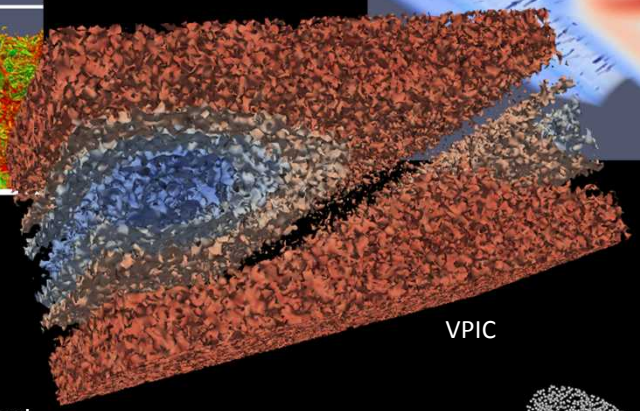

Post Hoc Vis
Simulation
Full Mesh
O(TB)
Disk
Storage
Visualization
In Situ Vis
Simulation
Visualization
Images
O(MB)
Disk
Storage



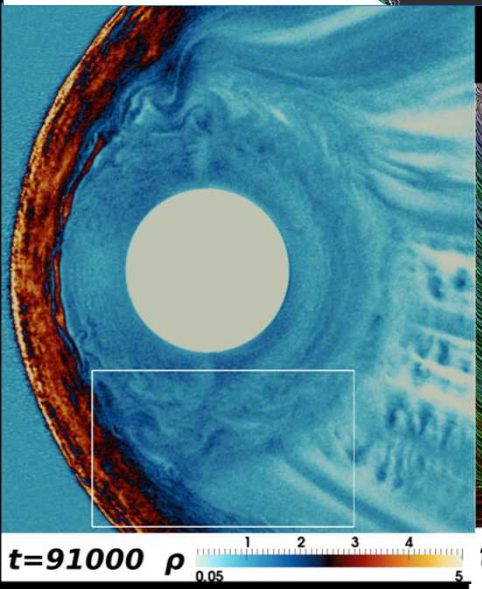




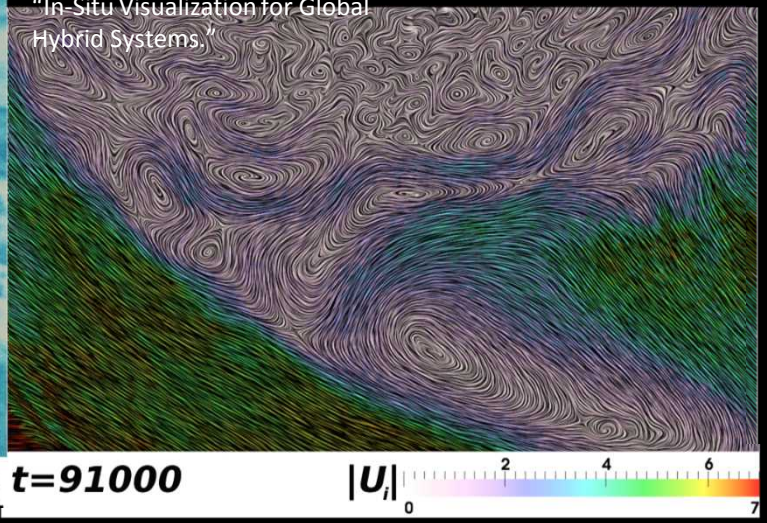
Lorendeau, Fournier, and Ribes. “In-Situ visualization in fluid mechanics using Catalyst: a case study for Code_Saturne.”

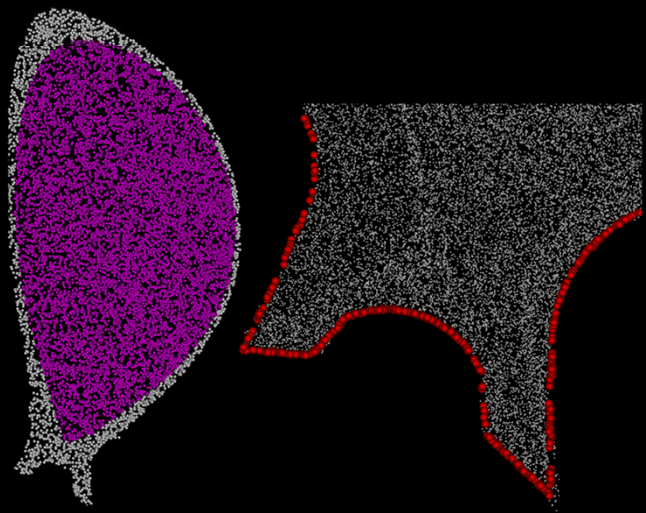


VPIC



Krimabadi, O’Leary, Tatineni, Loring, Majumdar, and Geveci. “In-Situ Visualization for Global Hybrid Systems.”





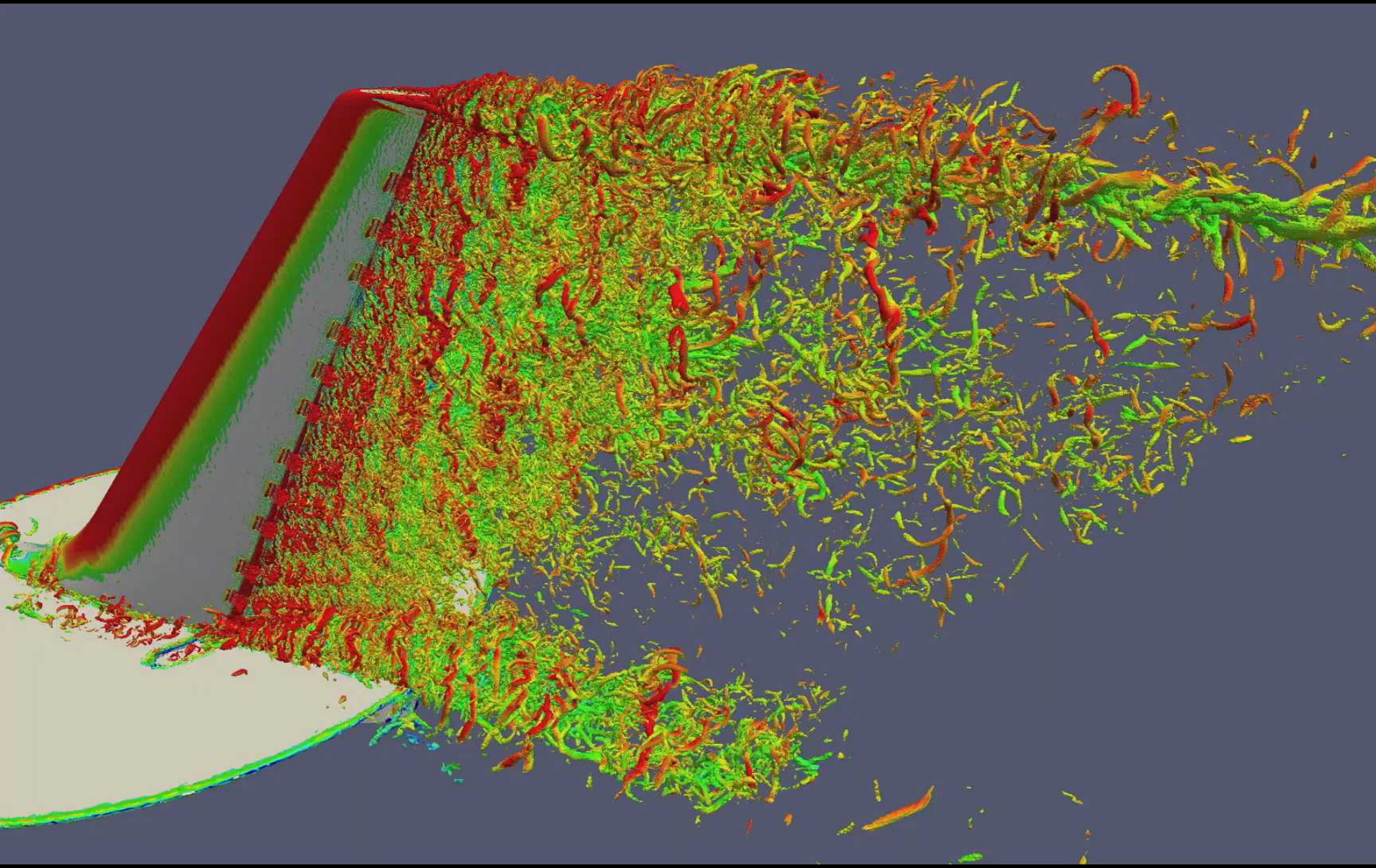


Image Credit: Jonathan Kos-Read

Extreme Scale is Threads, Threads Threads!

- A clear trend in supercomputing is ever increasing parallelism
- Clock increases are long gone
  - "The Free Lunch Is Over" (Herb Sutter)

	Jaguar – XT5	Titan – XK7	Exascale*
Cores	224,256	299,008 and 18,688 gpu	1 billion
Concurrency	224,256 way	70 – 500 million way	10 – 100 billion way
Memory	300 Terabytes	700 Terabytes	128 Petabytes

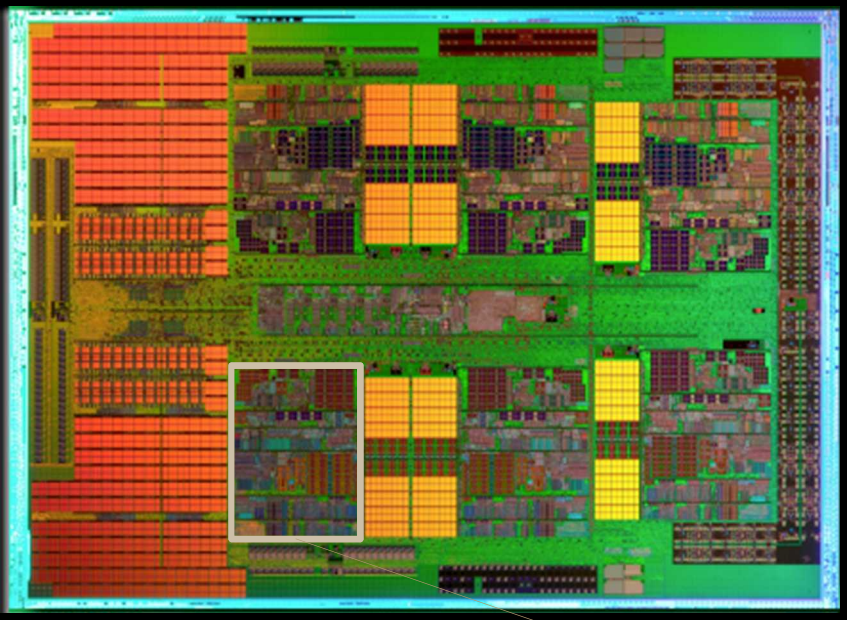
*Source: Scientific Discovery at the Exascale, Ahern, Shoshani, Ma, et al.

My new computer's got the clocks, it rocks
But it was obsolete before I opened the box

– “Weird” Al Yankovic, *It’s All About the Pentiums*, circa 1999

Moore’s Law is dead.

– Gordon Moore, circa 2005

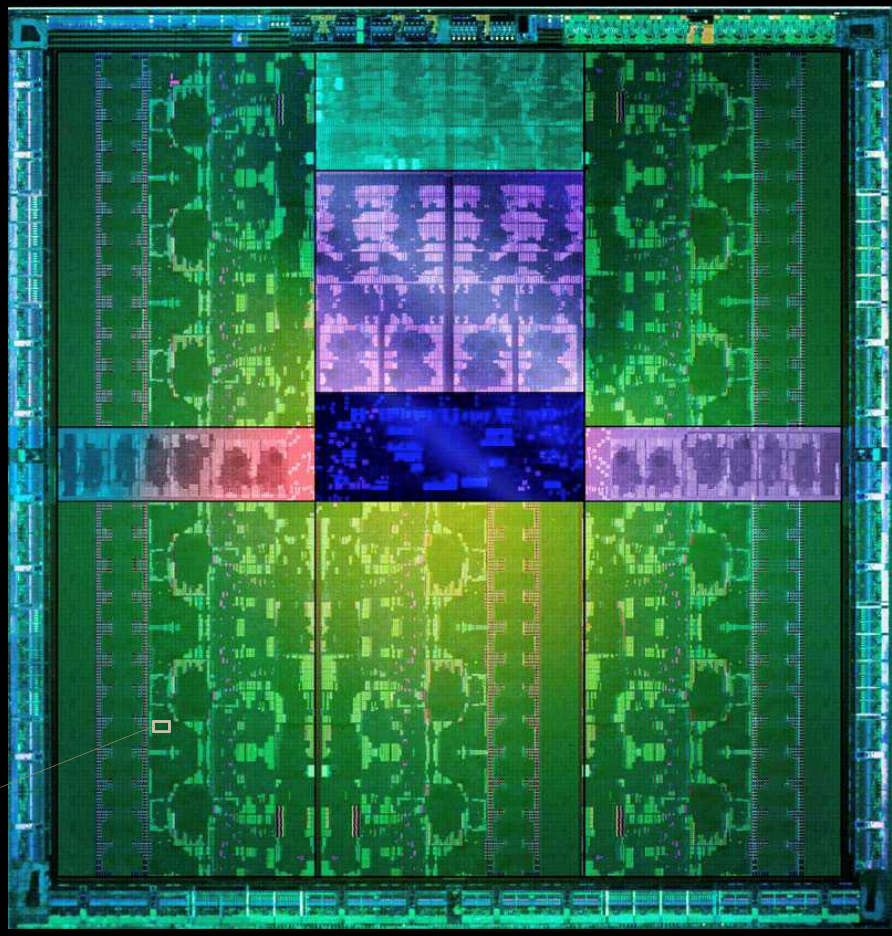


AMD x86

Full x86 Core
+ Associated Cache
6 cores per die
MPI-Only feasible

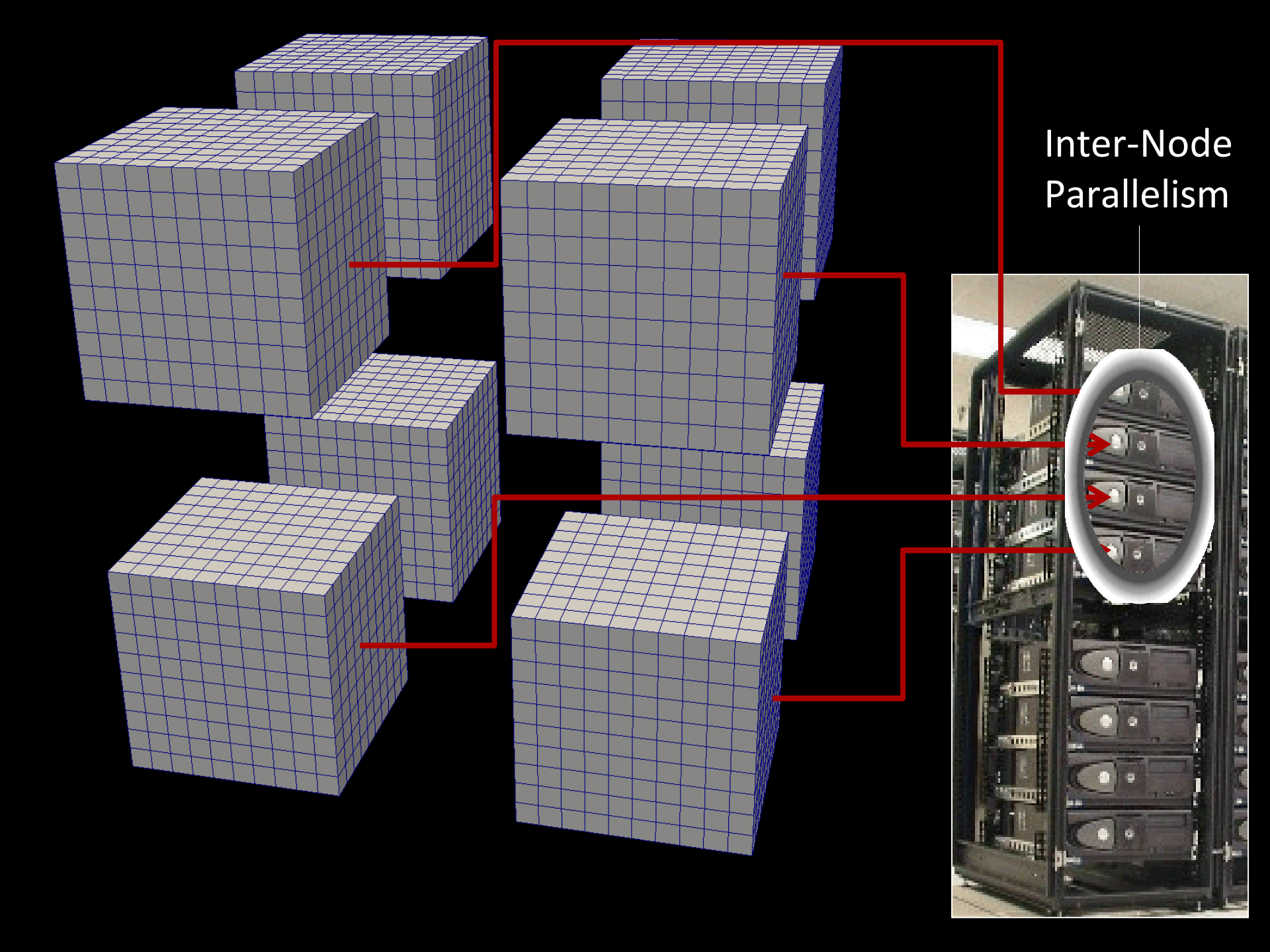
1 x86 core

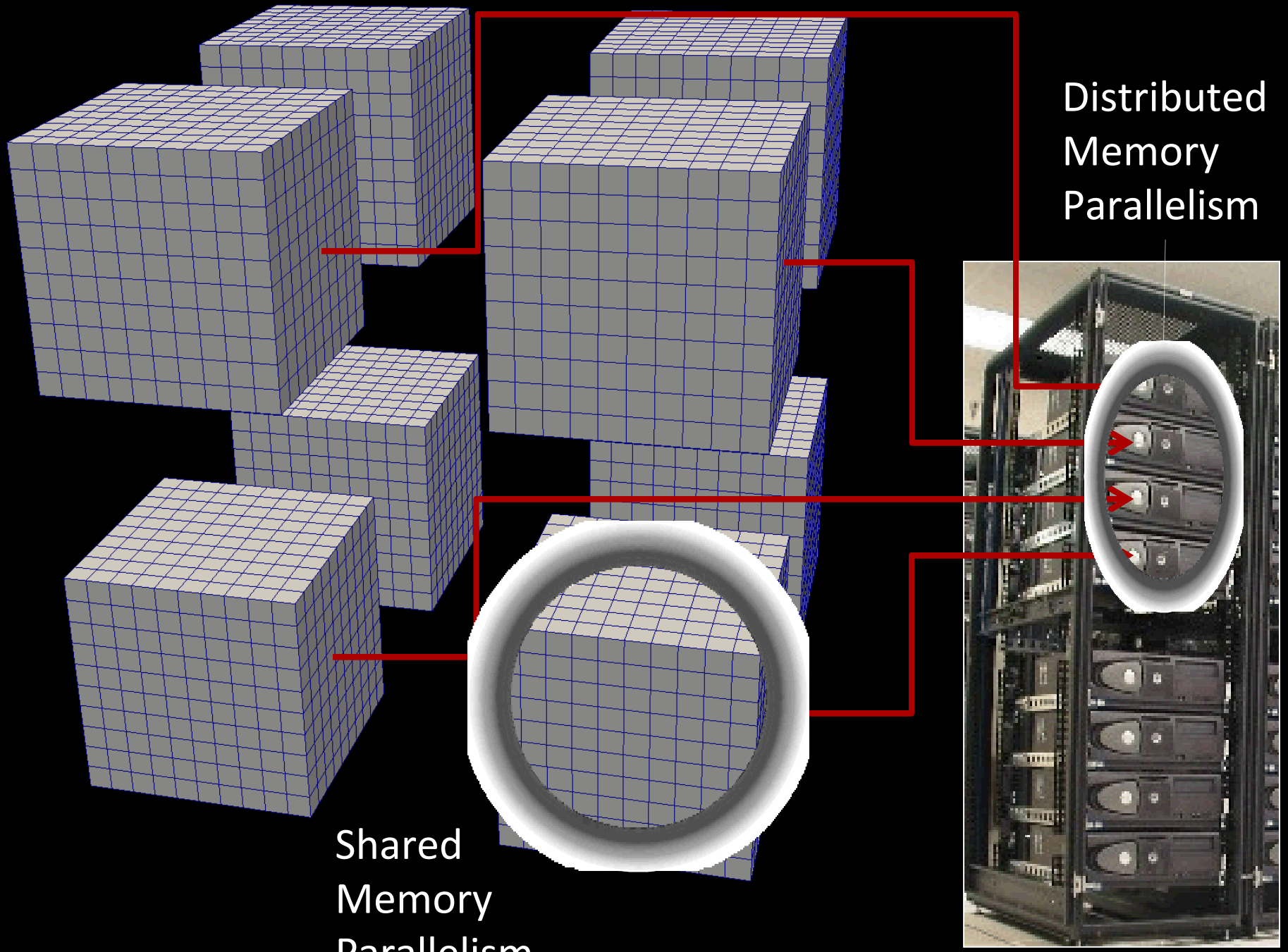
1 Kepler "core"



NVIDIA GPU

2,880 cores collected in 15 SMX
Shared PC, Cache, Mem Fetches
Reduced control logic
MPI-Only not feasible


Inter-Node
Parallelism


Distributed
Memory
Parallelism
Shared
Memory
Parallelism

VTK-m

Sandia National Laboratories

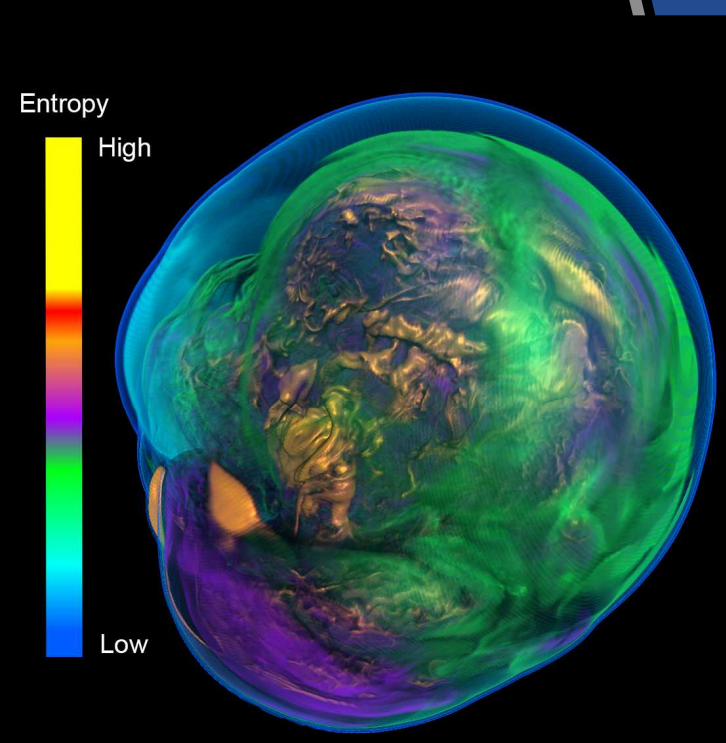
Los Alamos National Laboratory

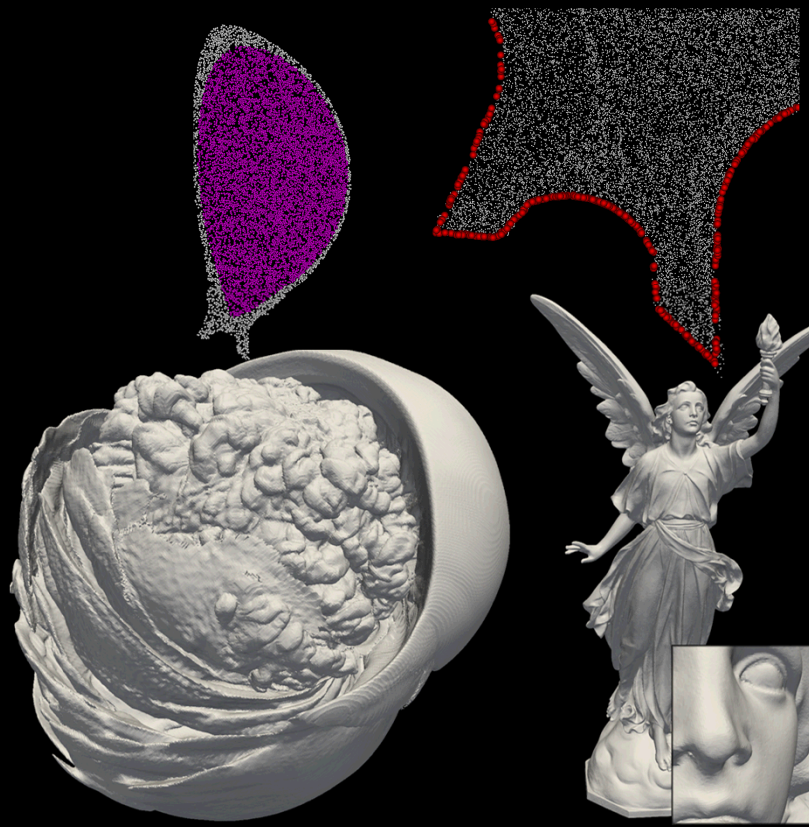
Oak Ridge National Laboratory

Kitware

University of Oregon

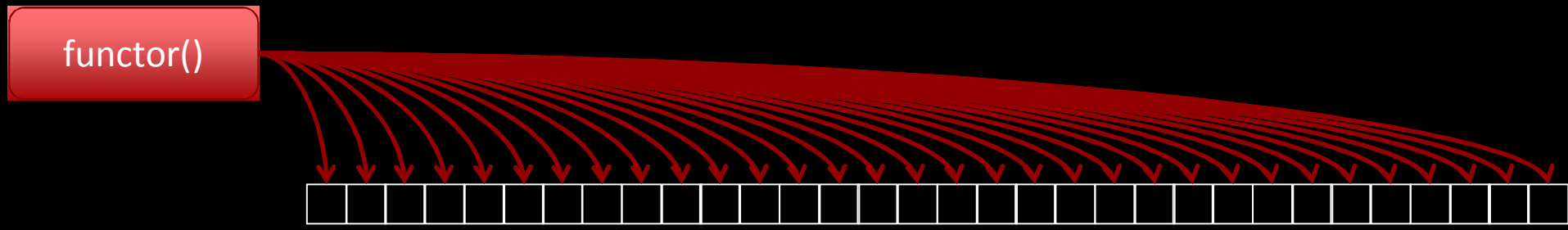
http://m.vtk.org



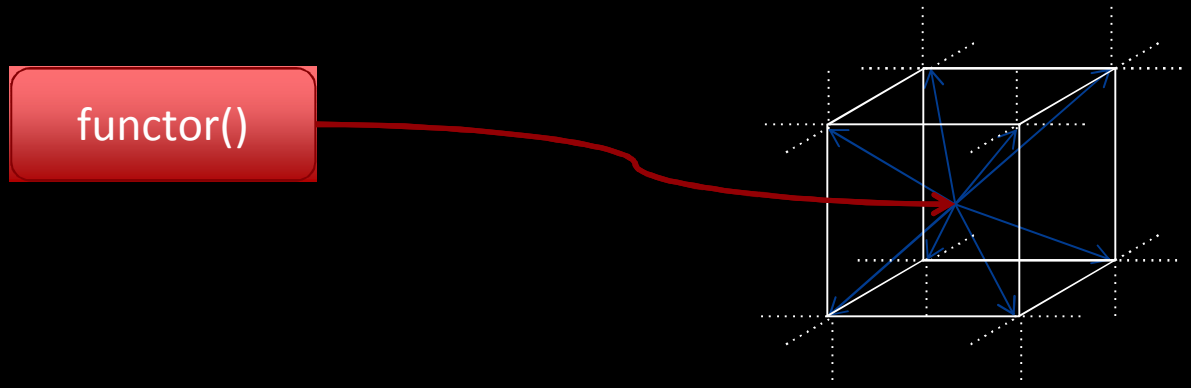


VTK-m: Basic Approach

- Functor mapping [Baker, et al. 2010]

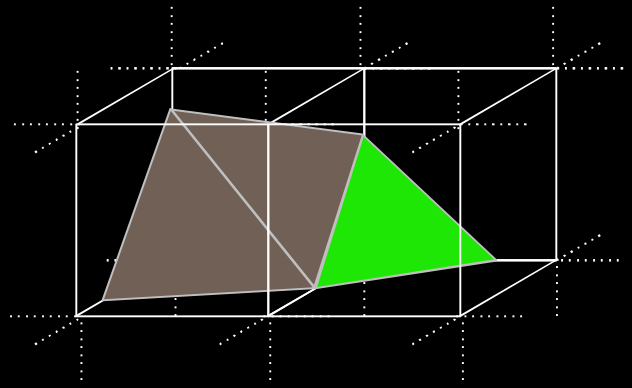


Applied to Topologies

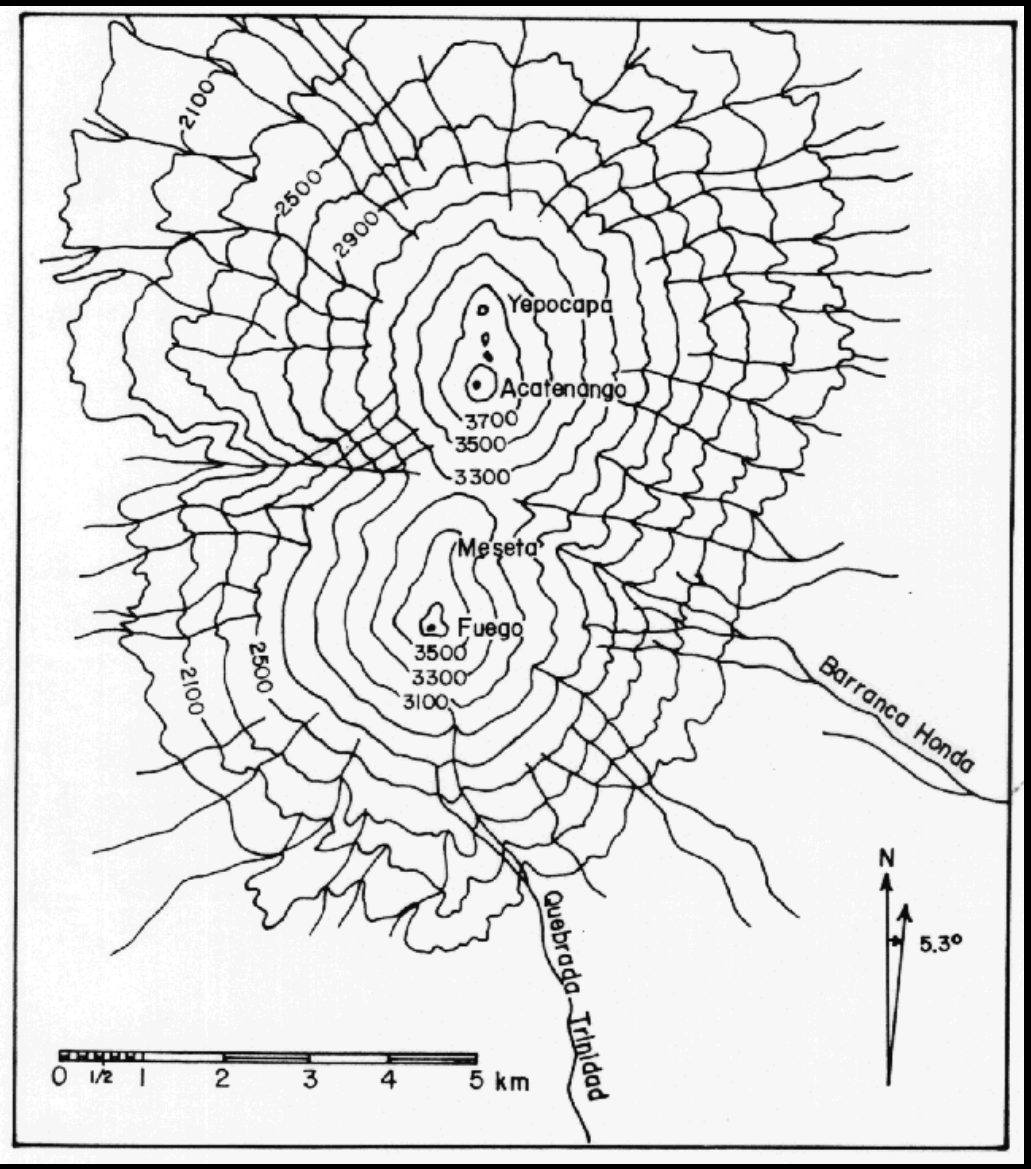


Applied to Topologies

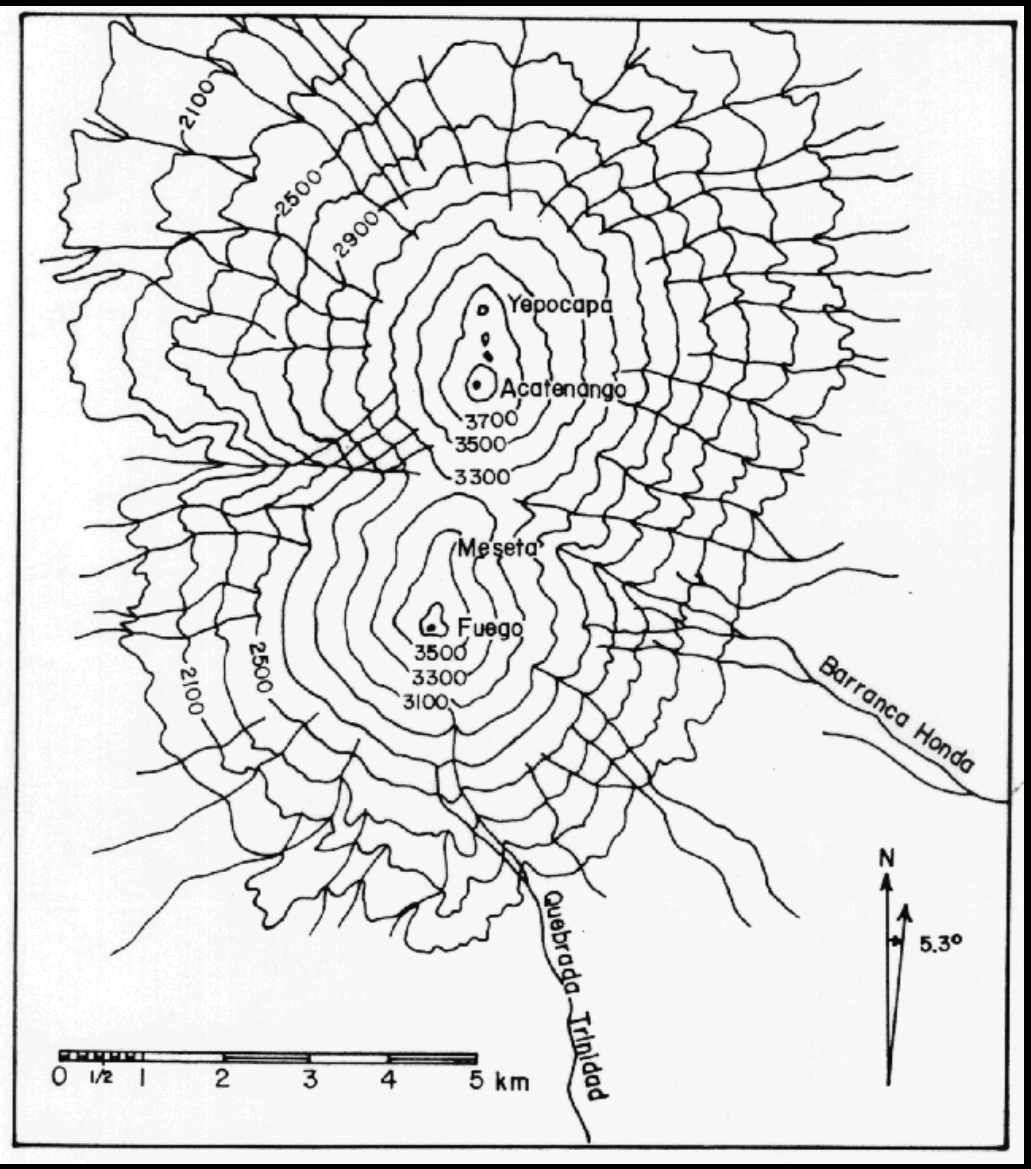
functor()

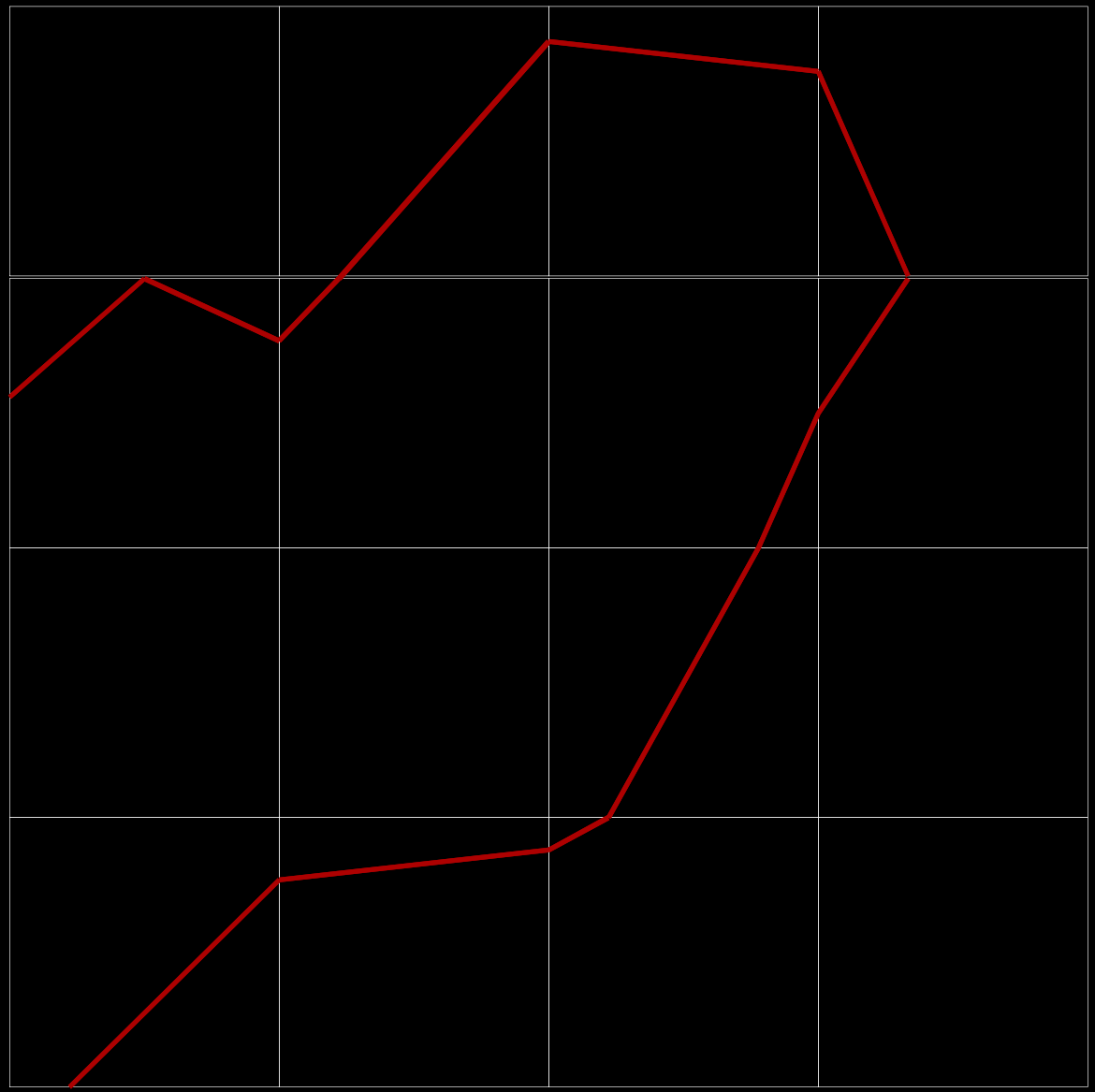


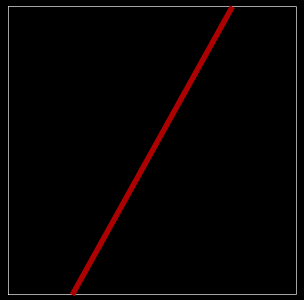
Example Algorithm: Contours

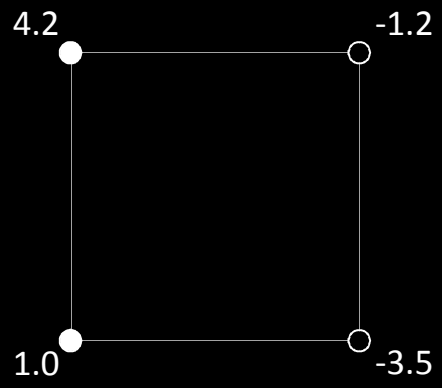


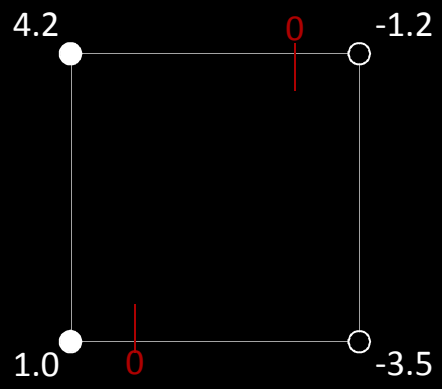
Example Algorithm: Contours

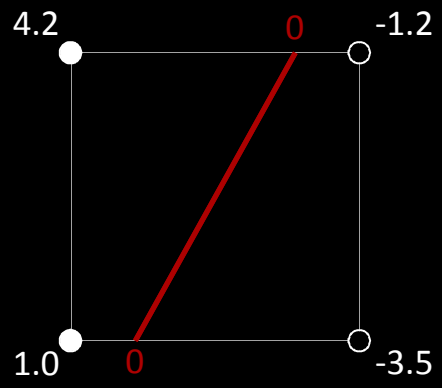


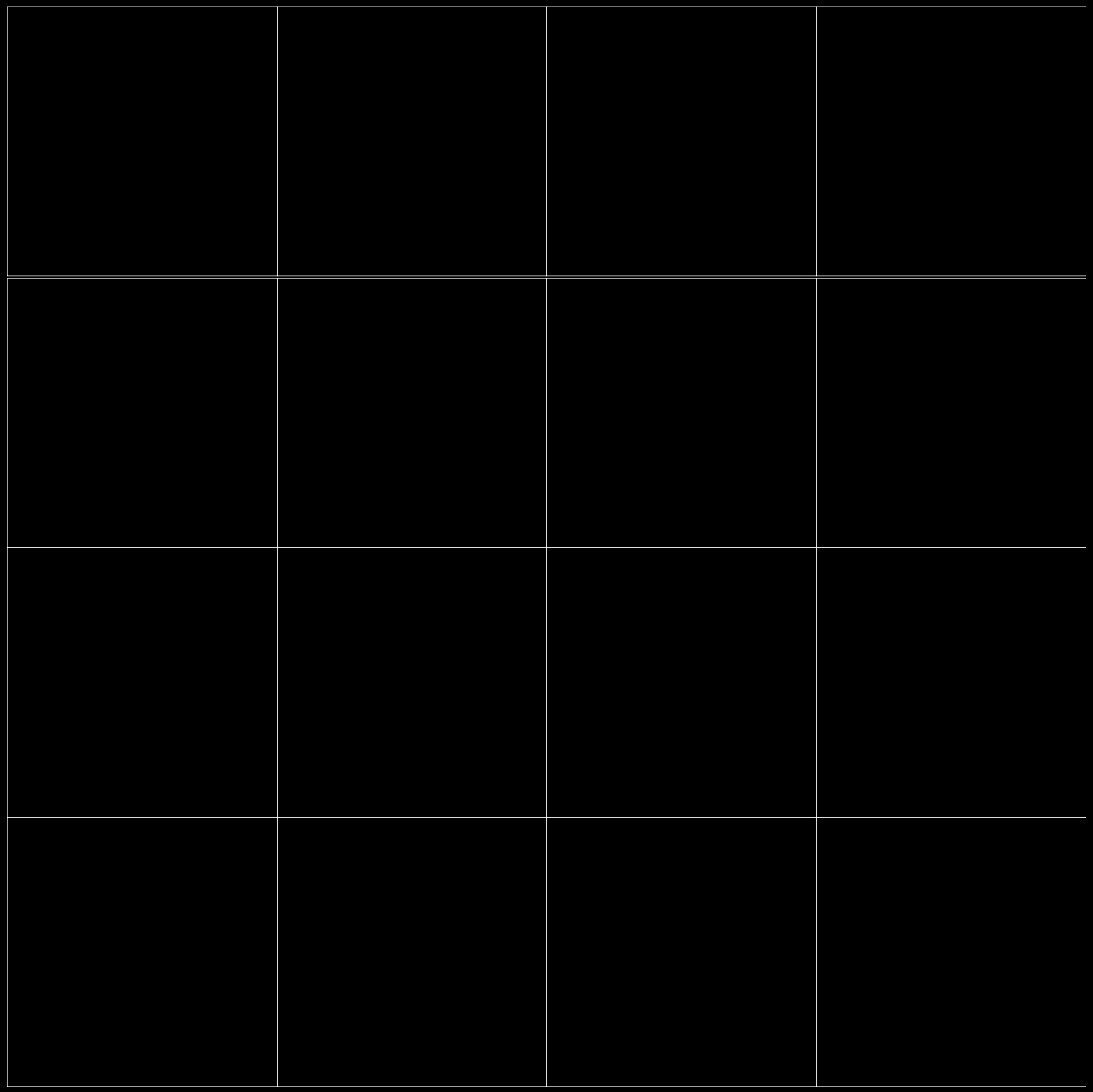


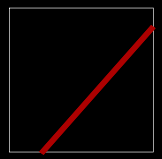


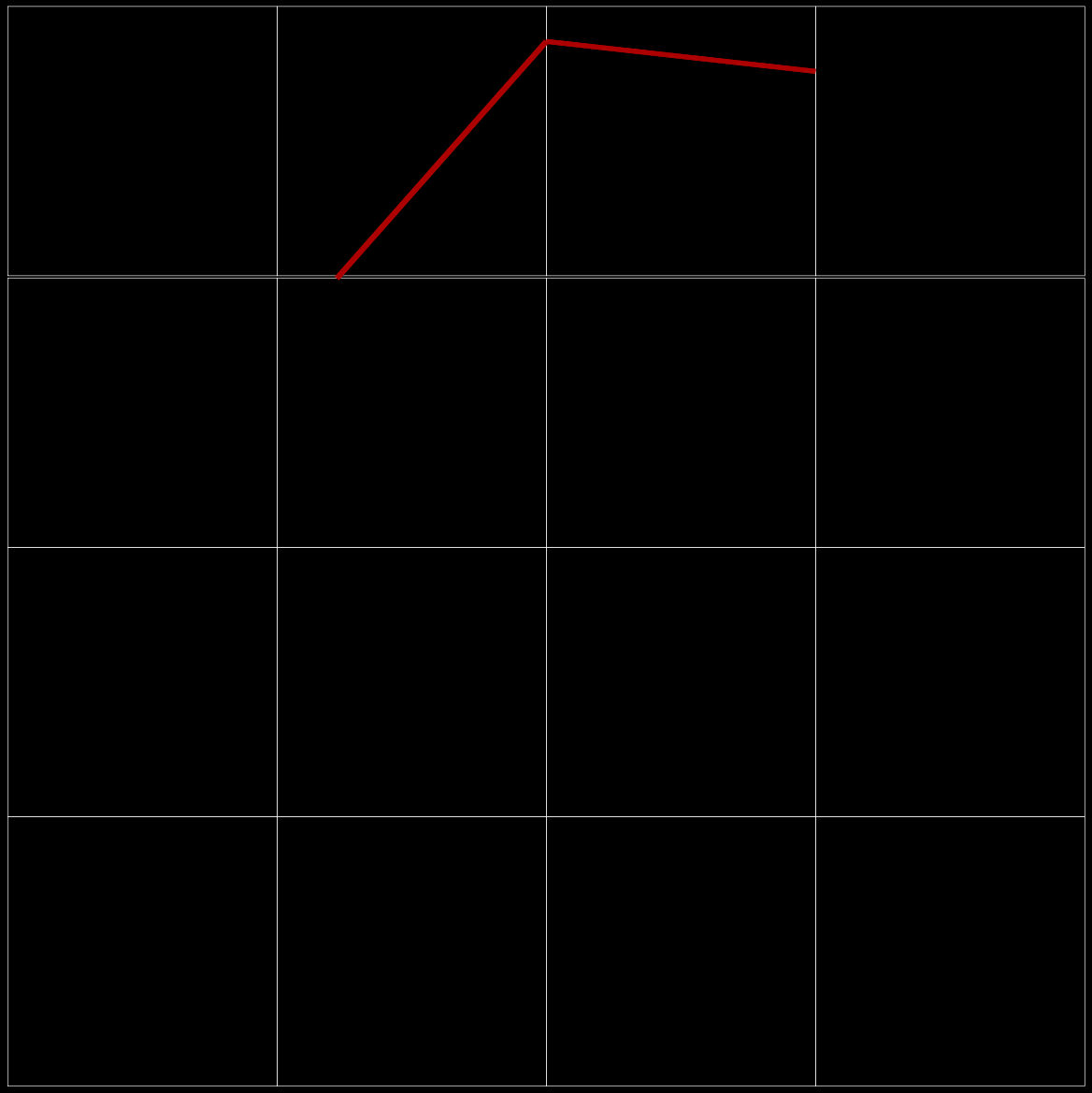

4.2
-1.2
1.0
-3.5

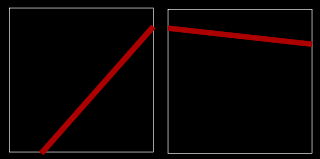

4.2
0
-1.2
1.0
0
-3.5

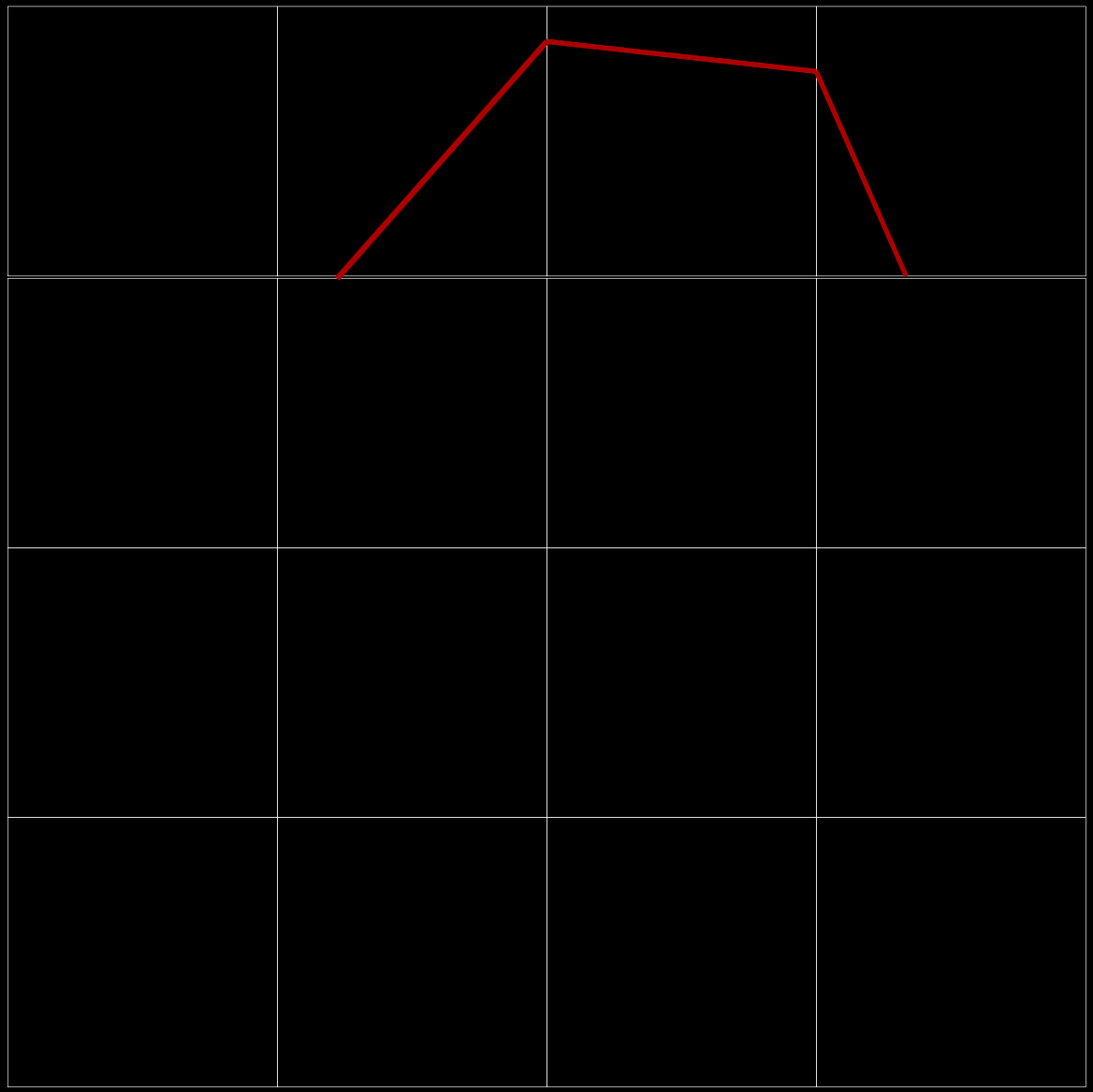

4.2
0
-1.2
1.0
0
-3.5

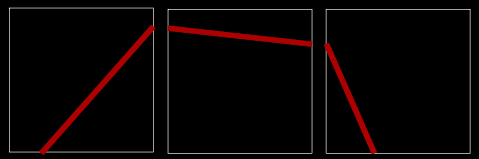


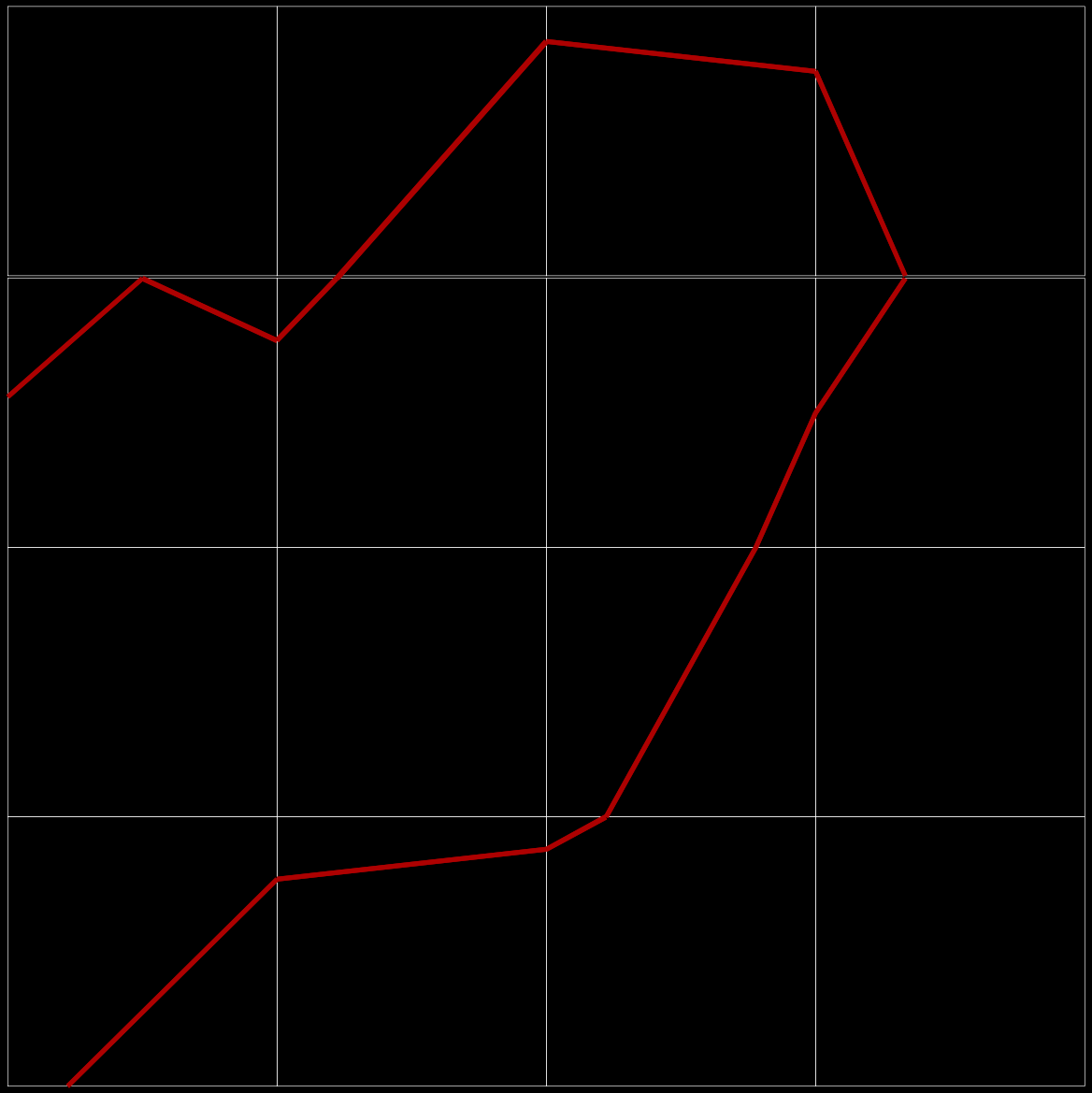


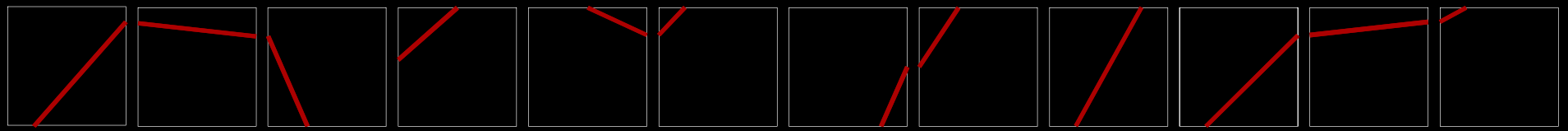


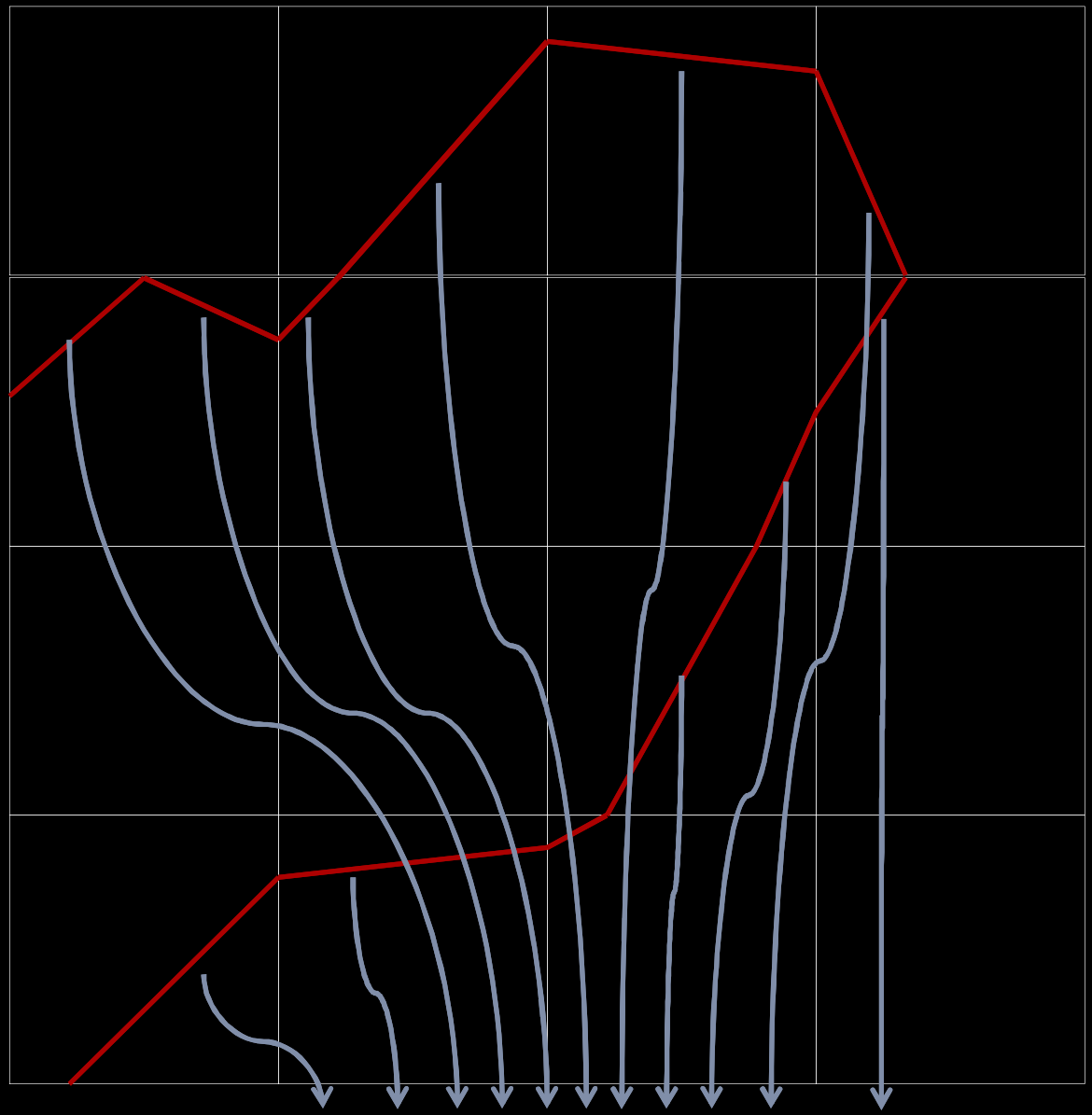


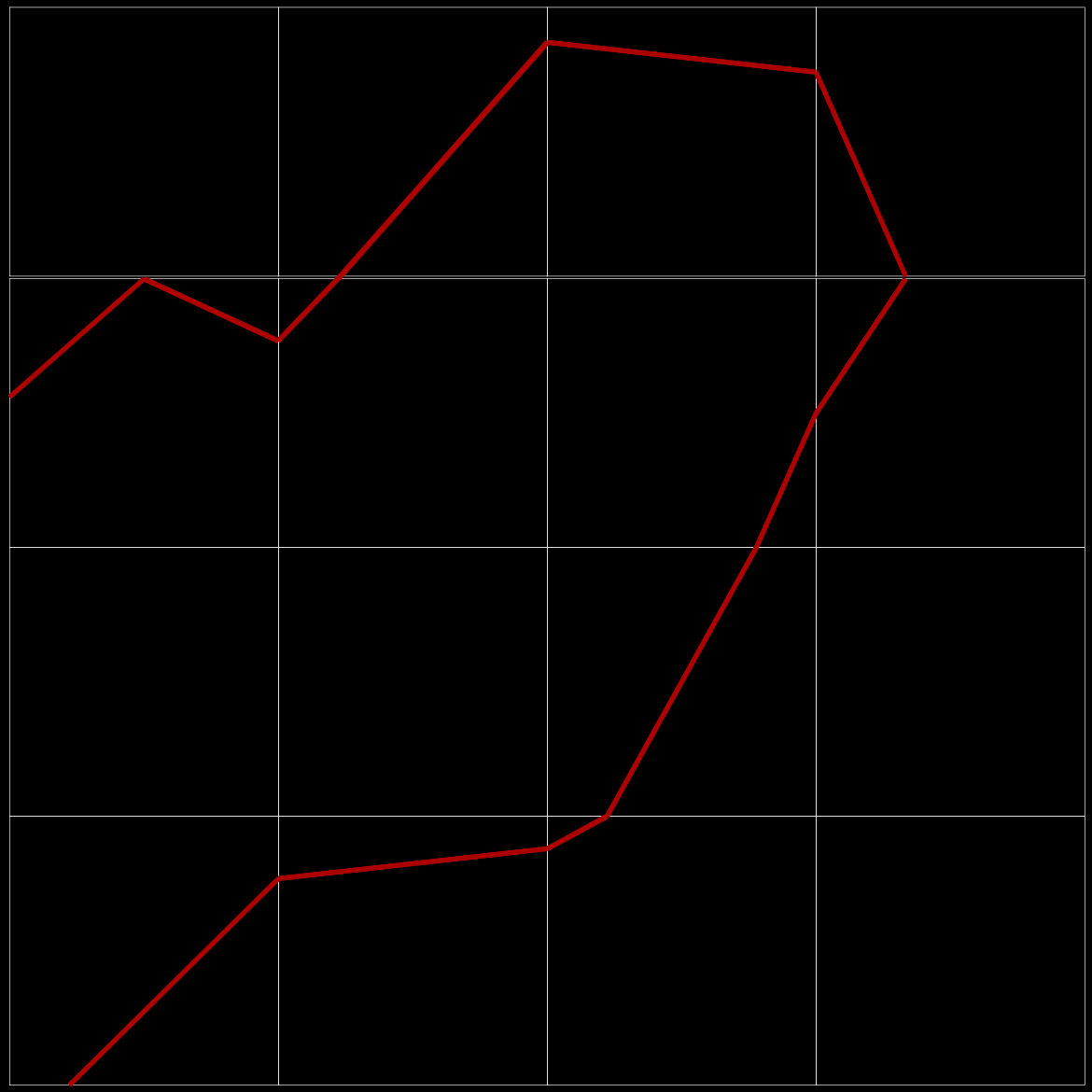




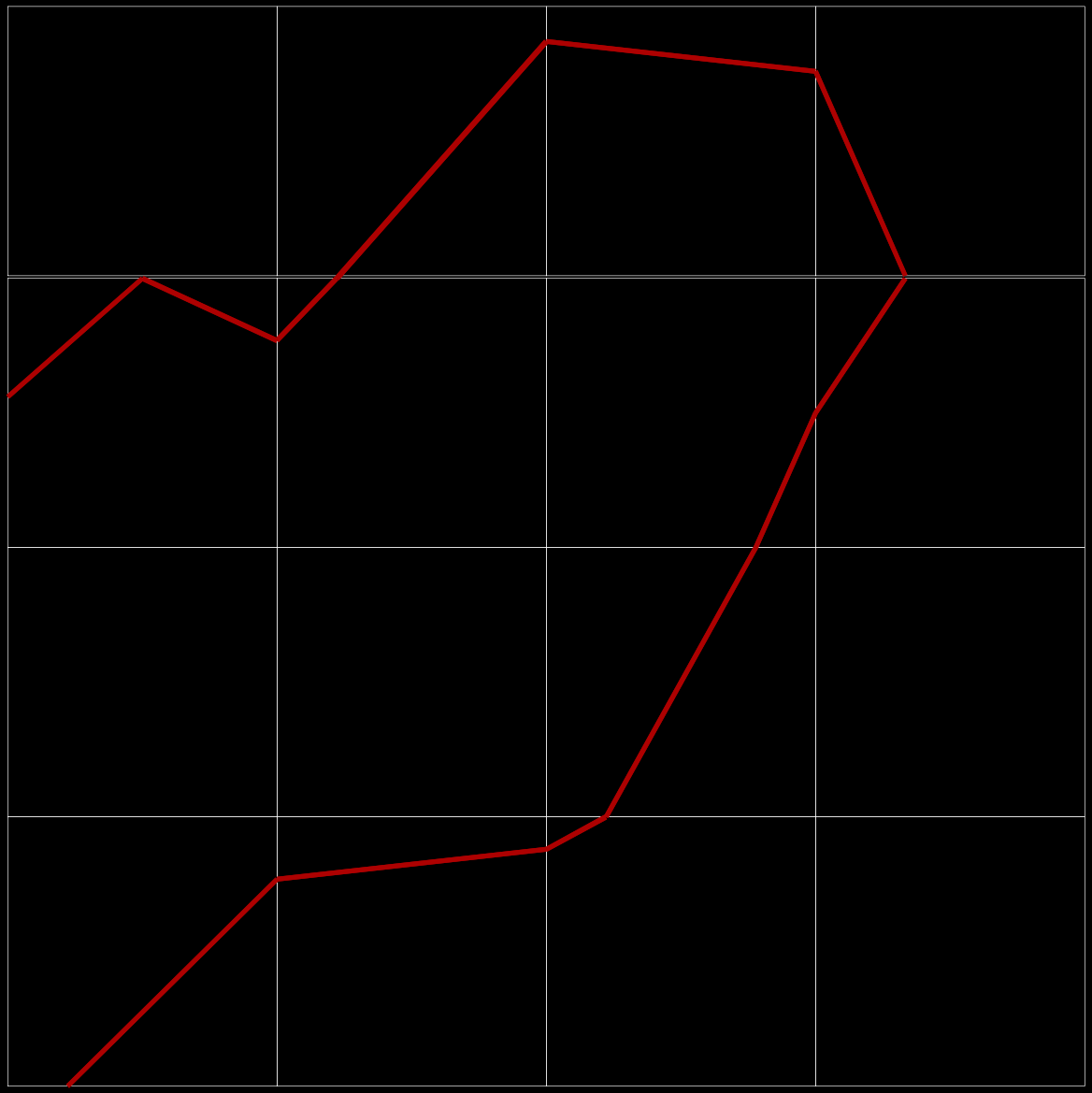






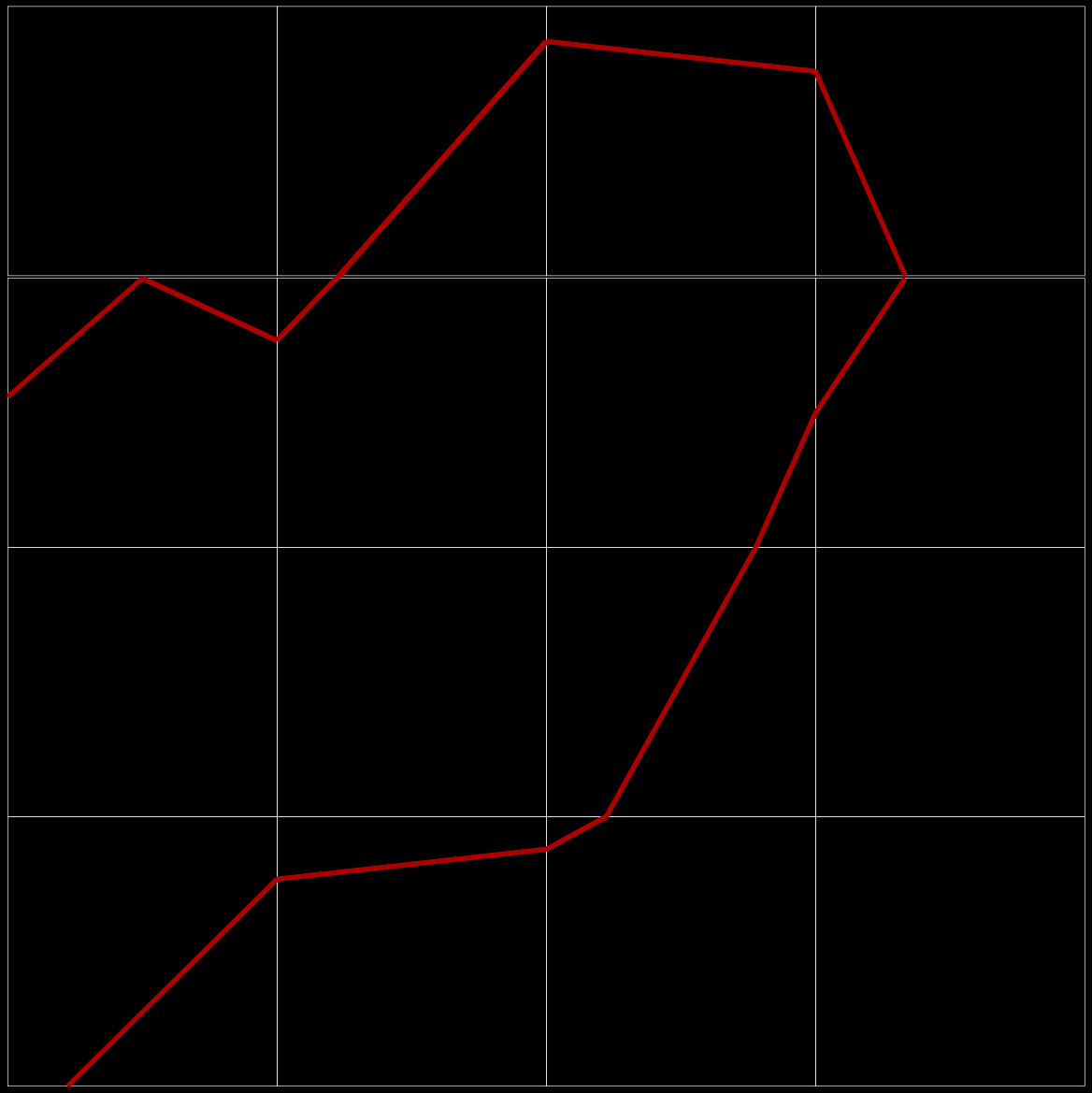


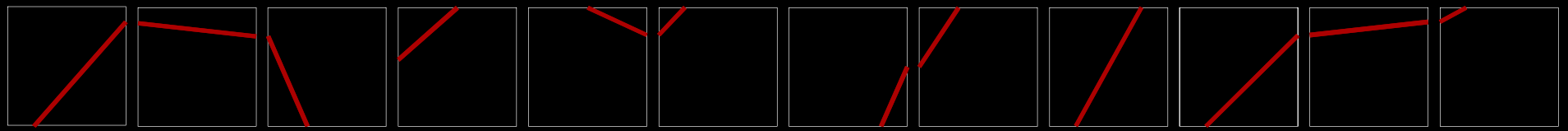
0	1	1	1	2	1	1	1	0	0	1	0	1	1	1	0

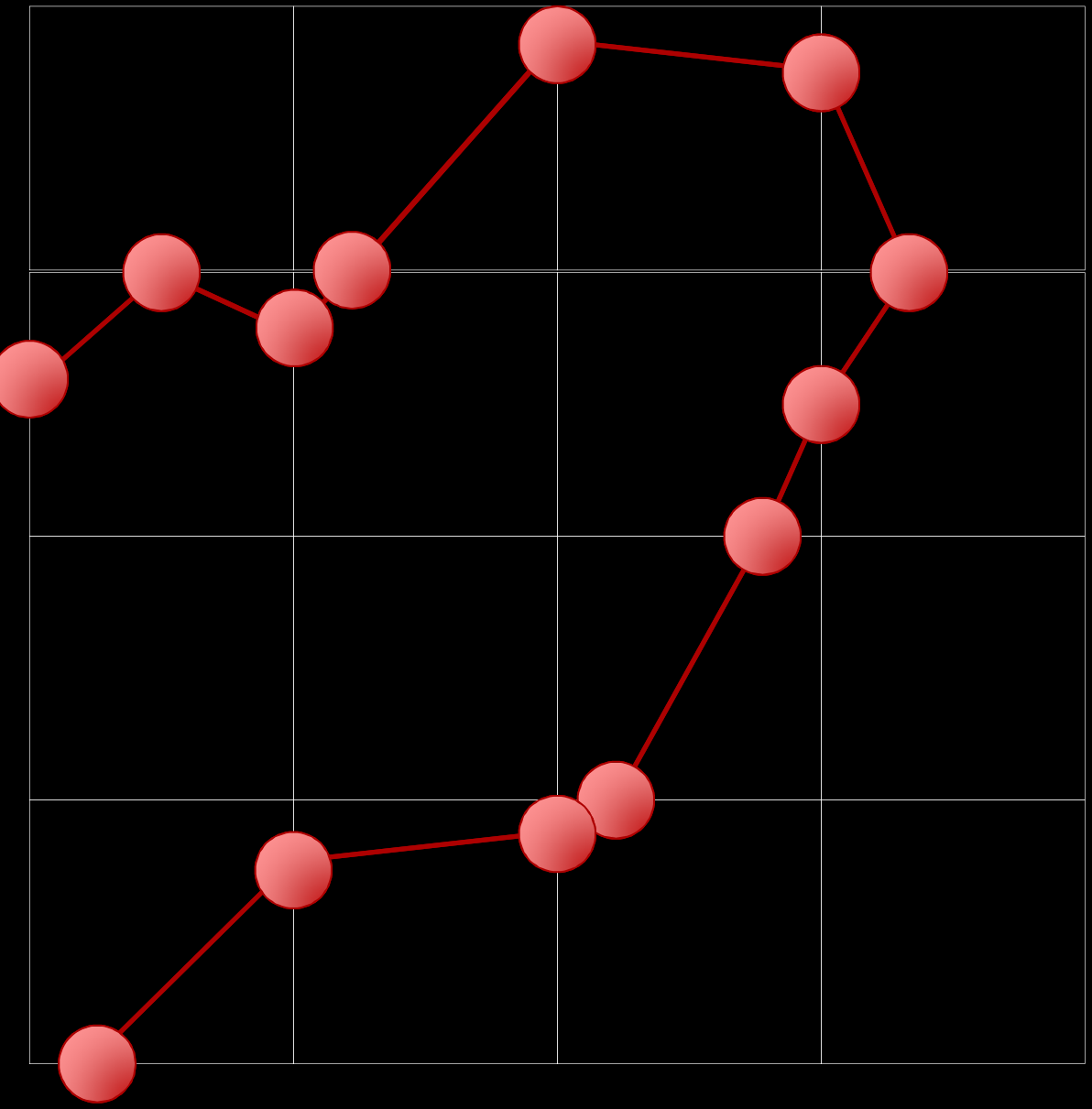


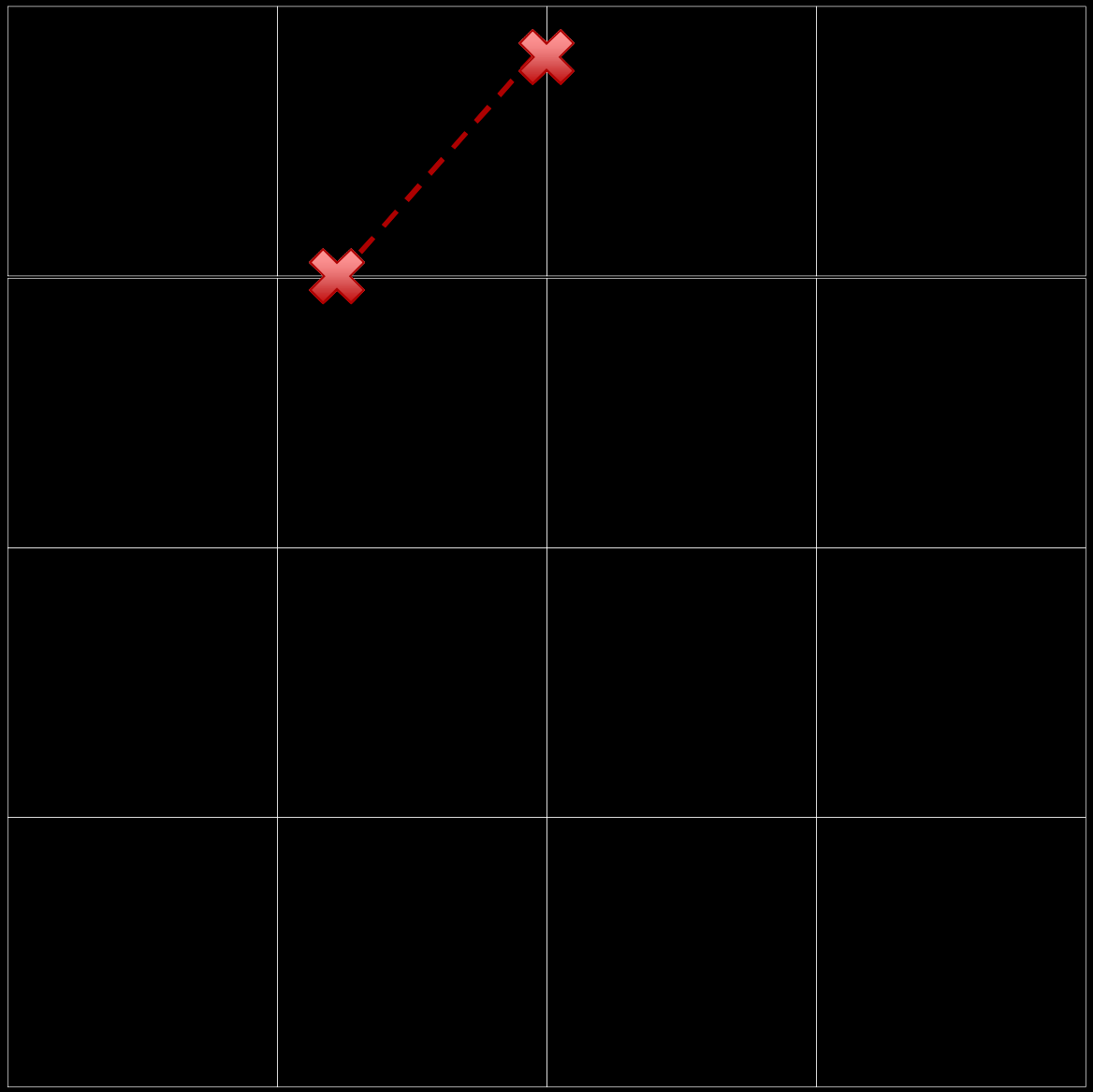
+0	+1	+1	+1	+2	+1	+1	+1	+0	+0	+1	+0	+1	+1	+1	+0
0	0	1	2	3	5	6	7	8	8	8	9	9	10	11	12

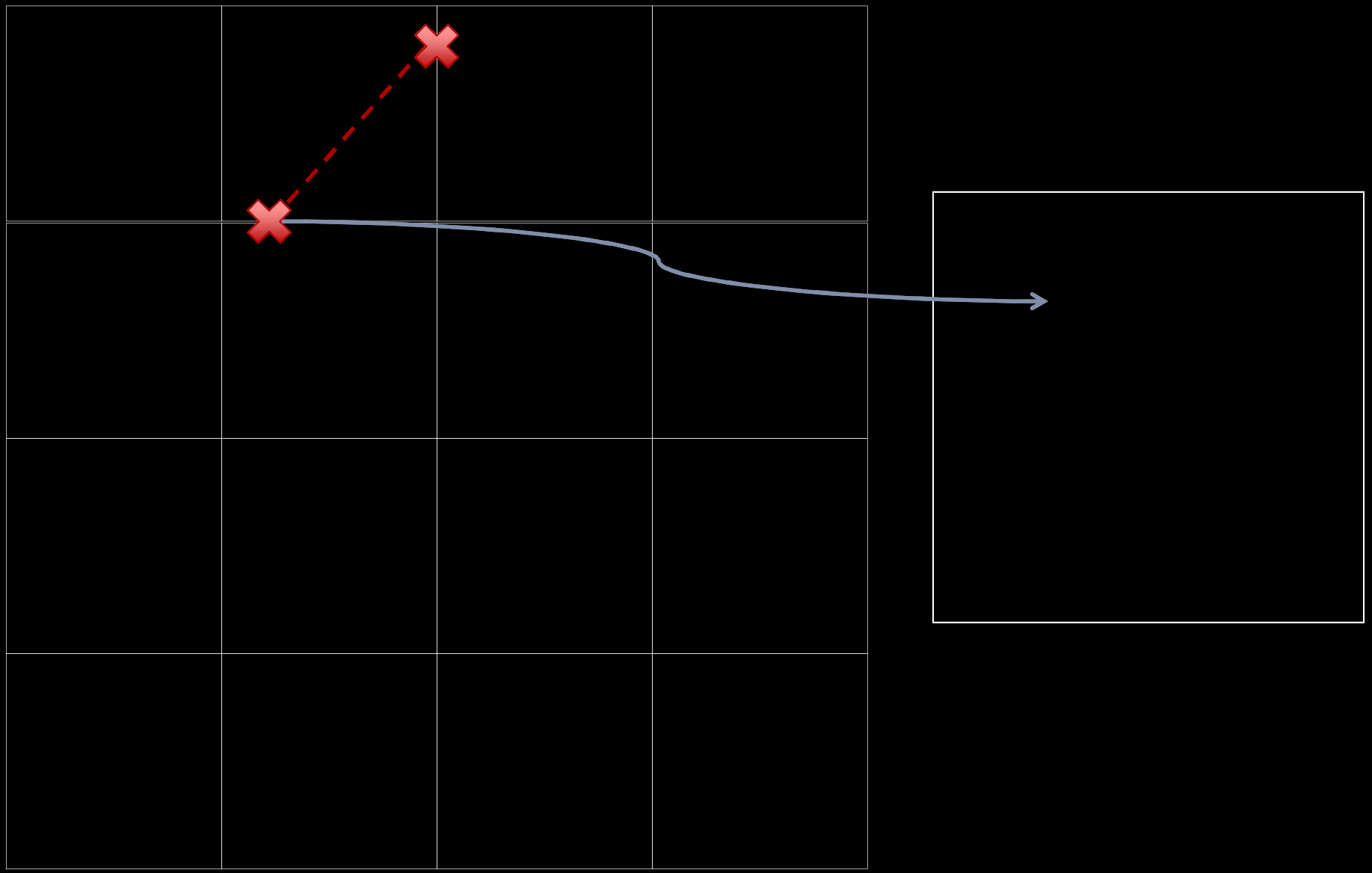
Total: 12

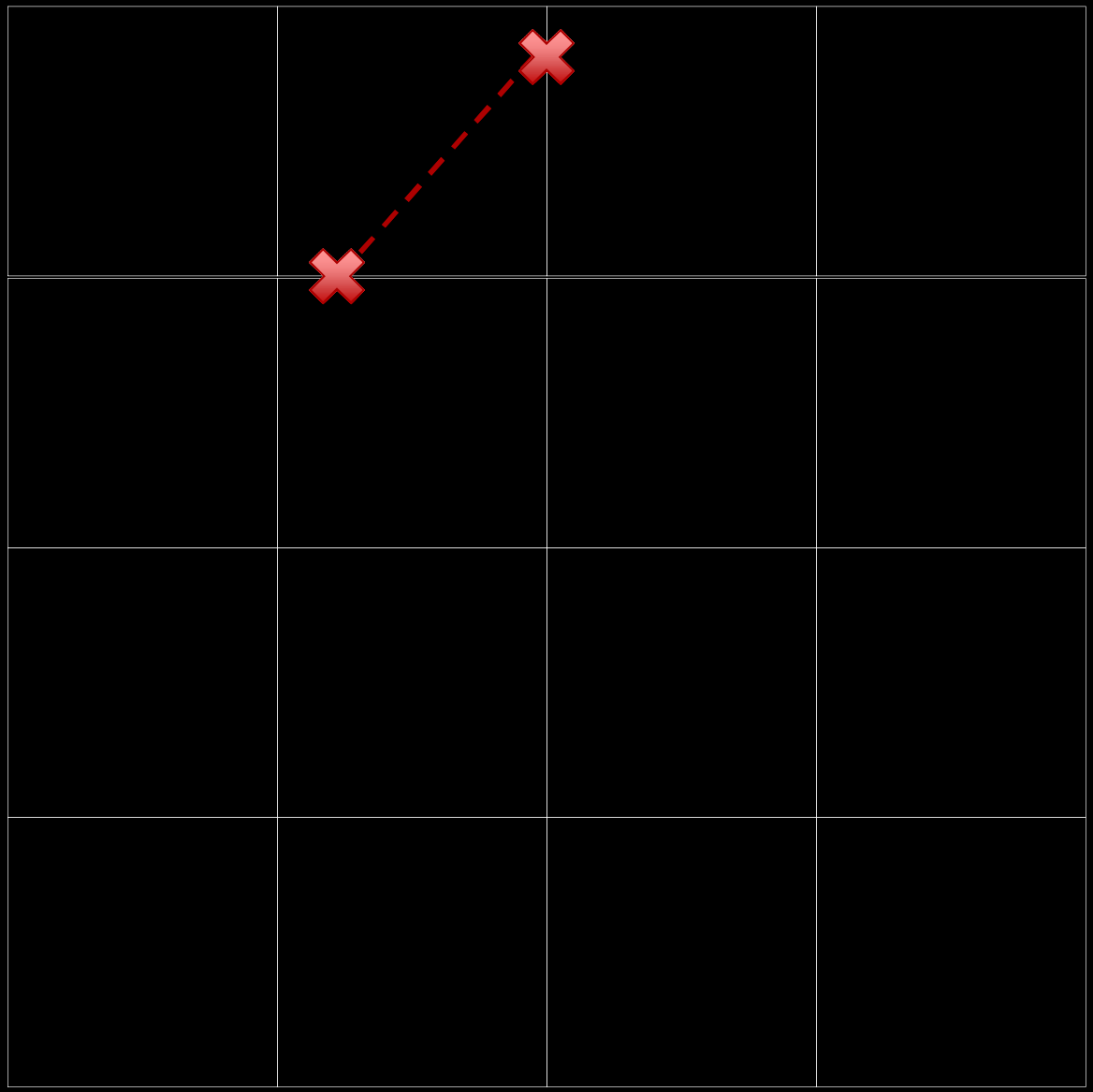


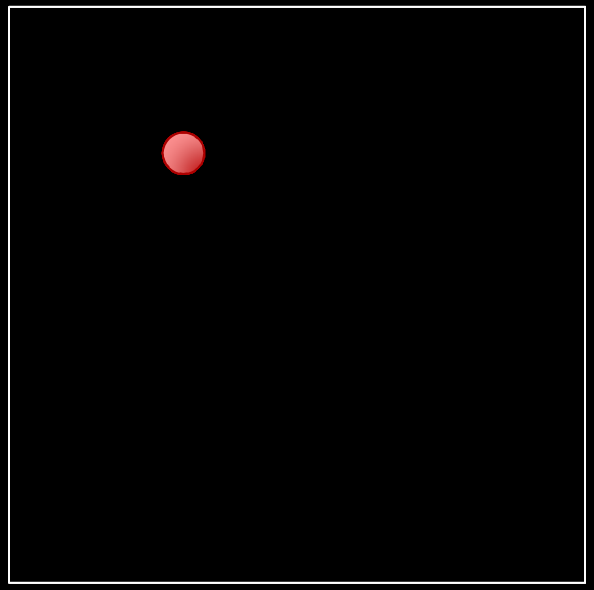


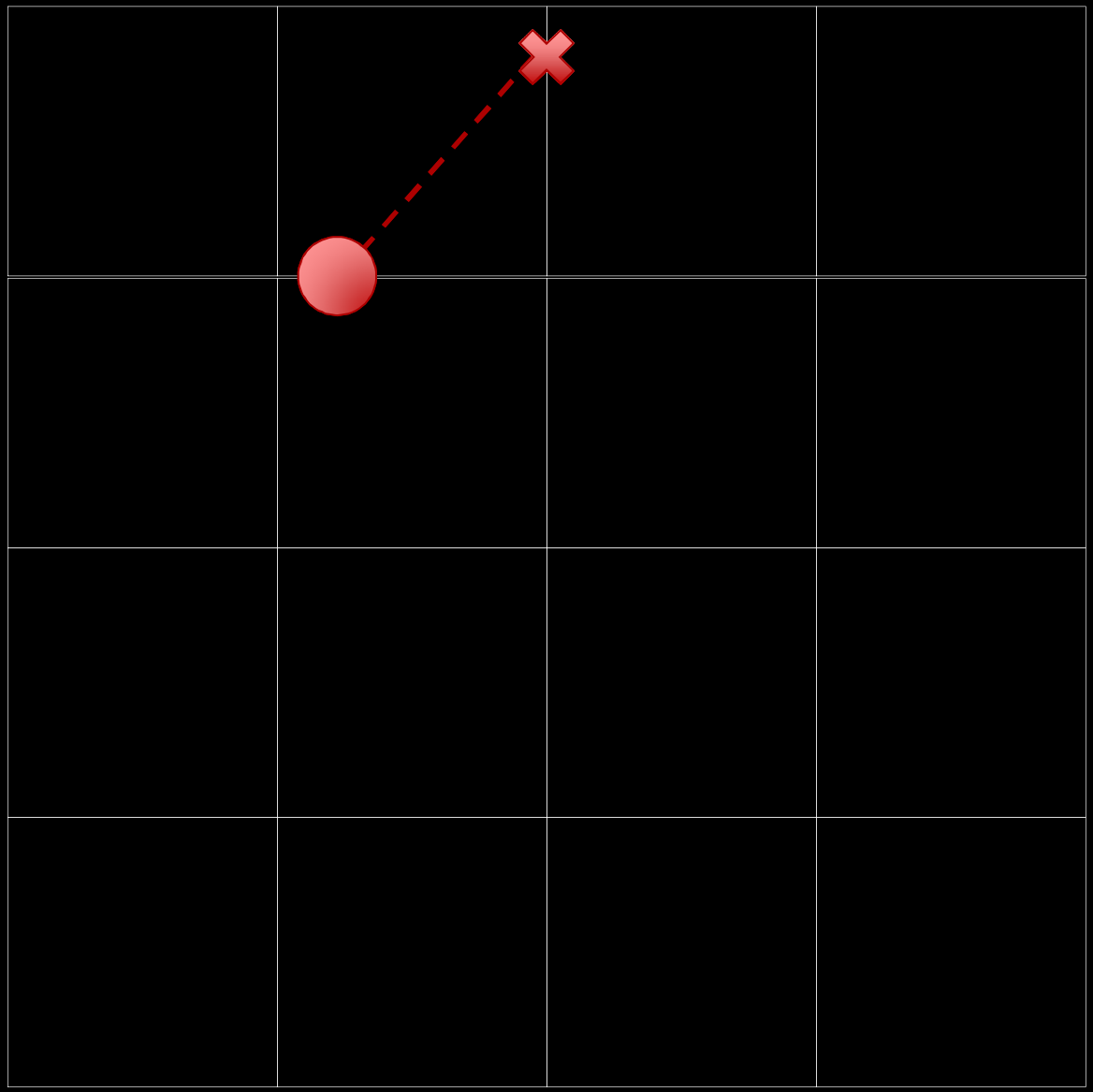


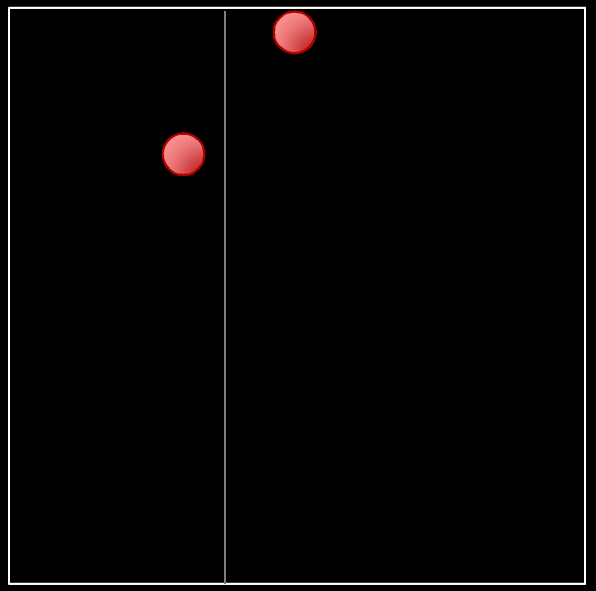


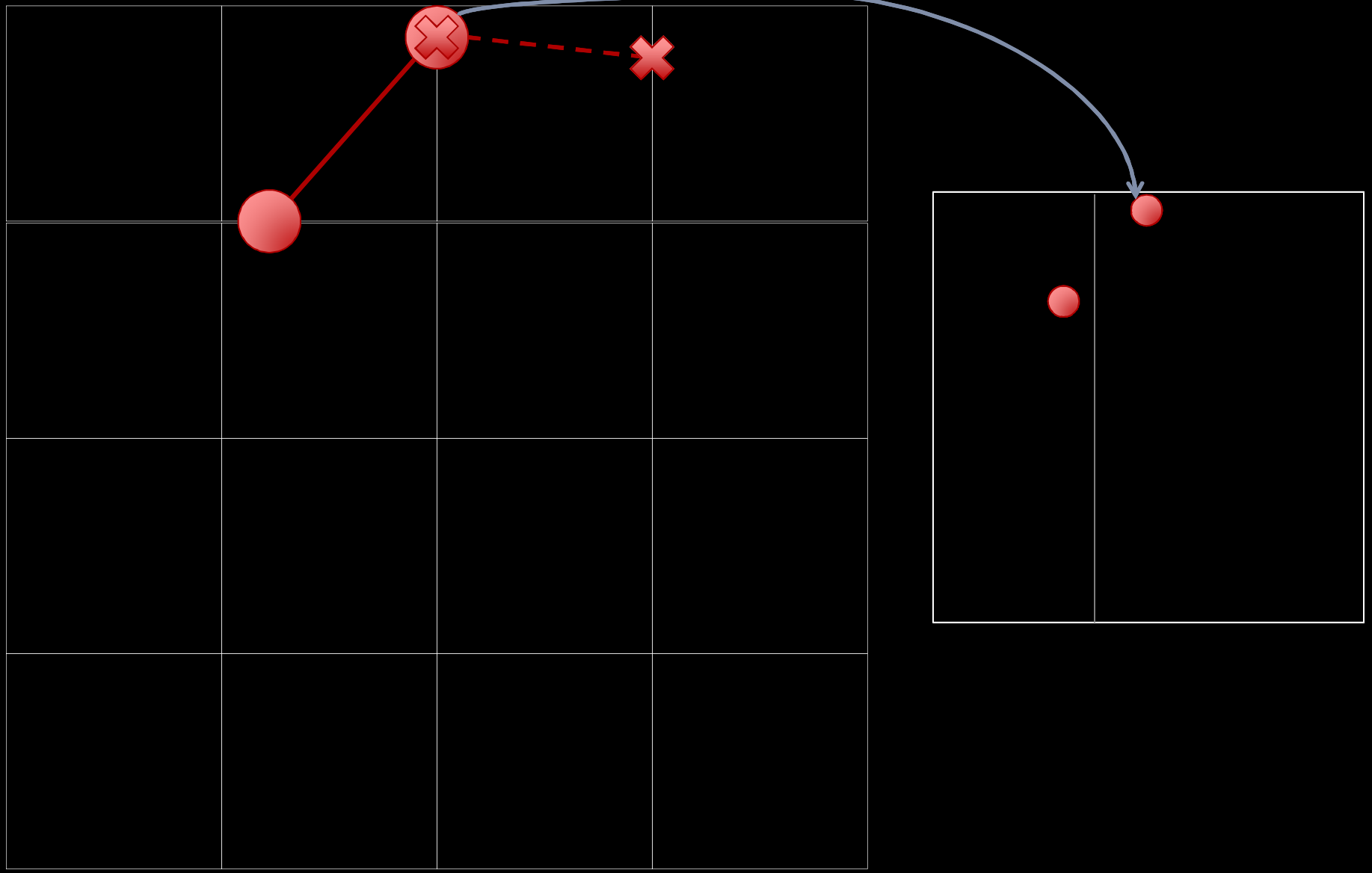


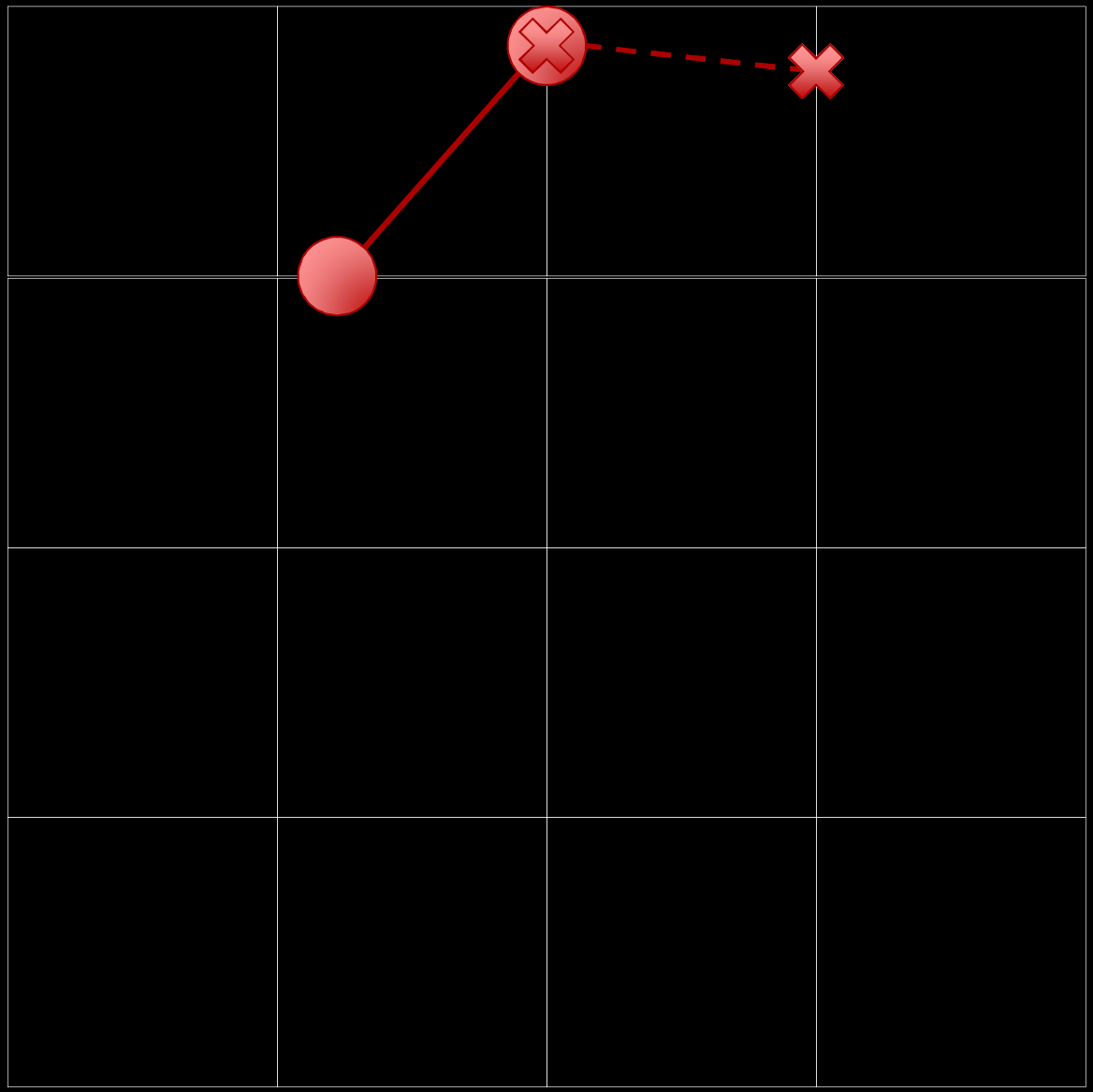


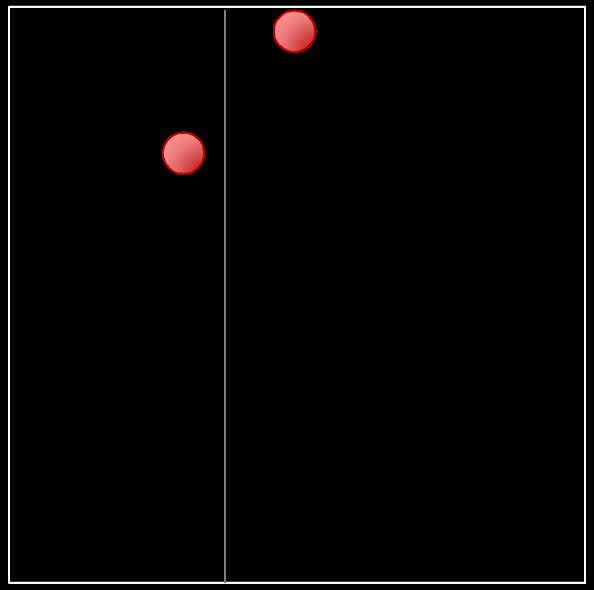


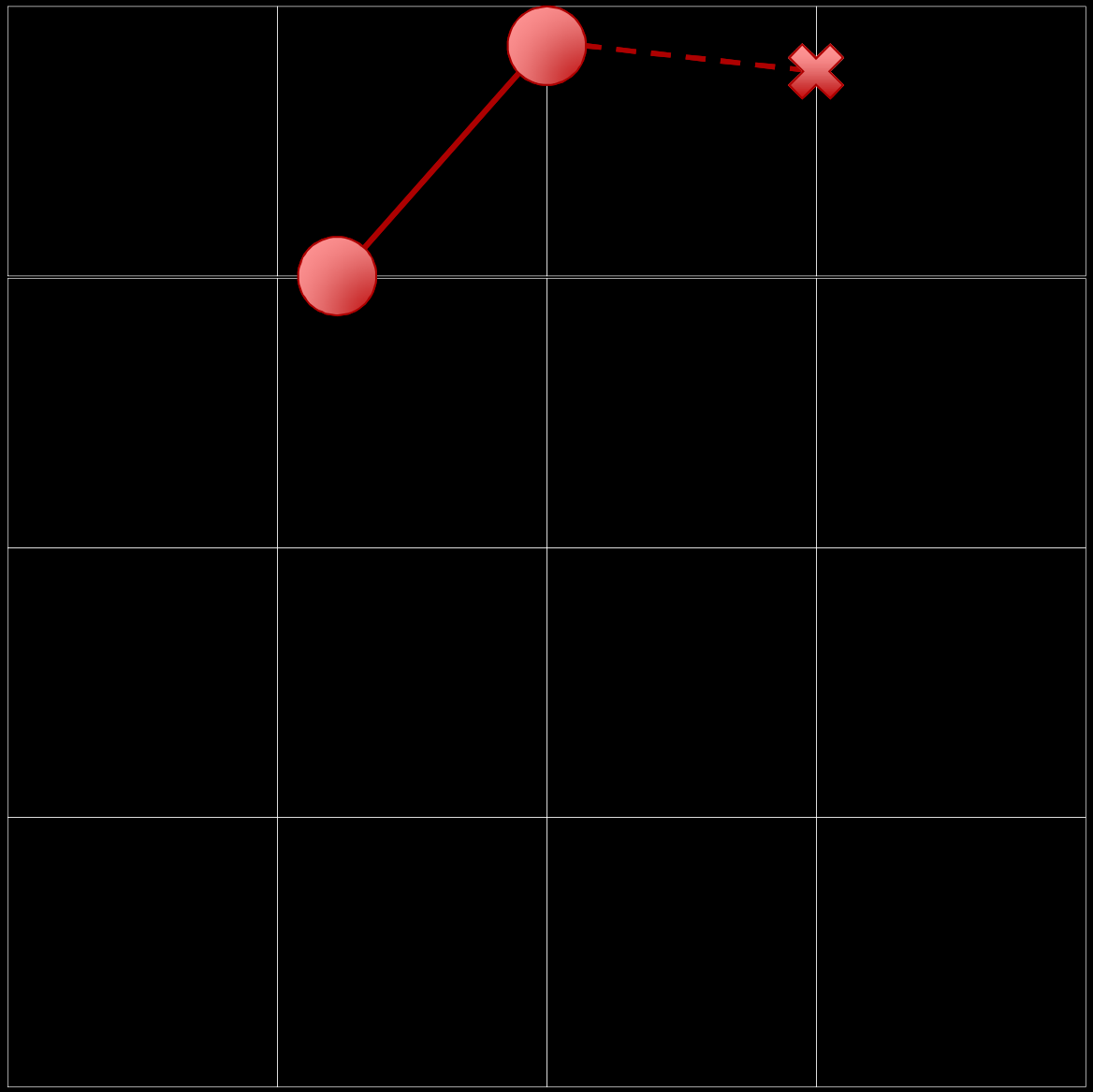


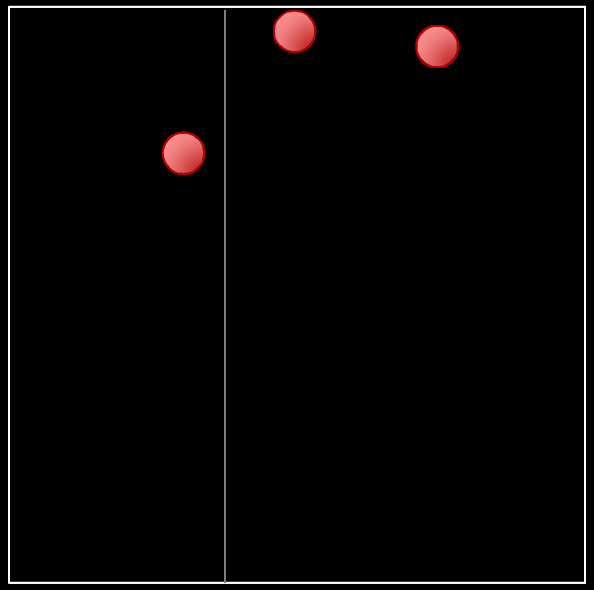


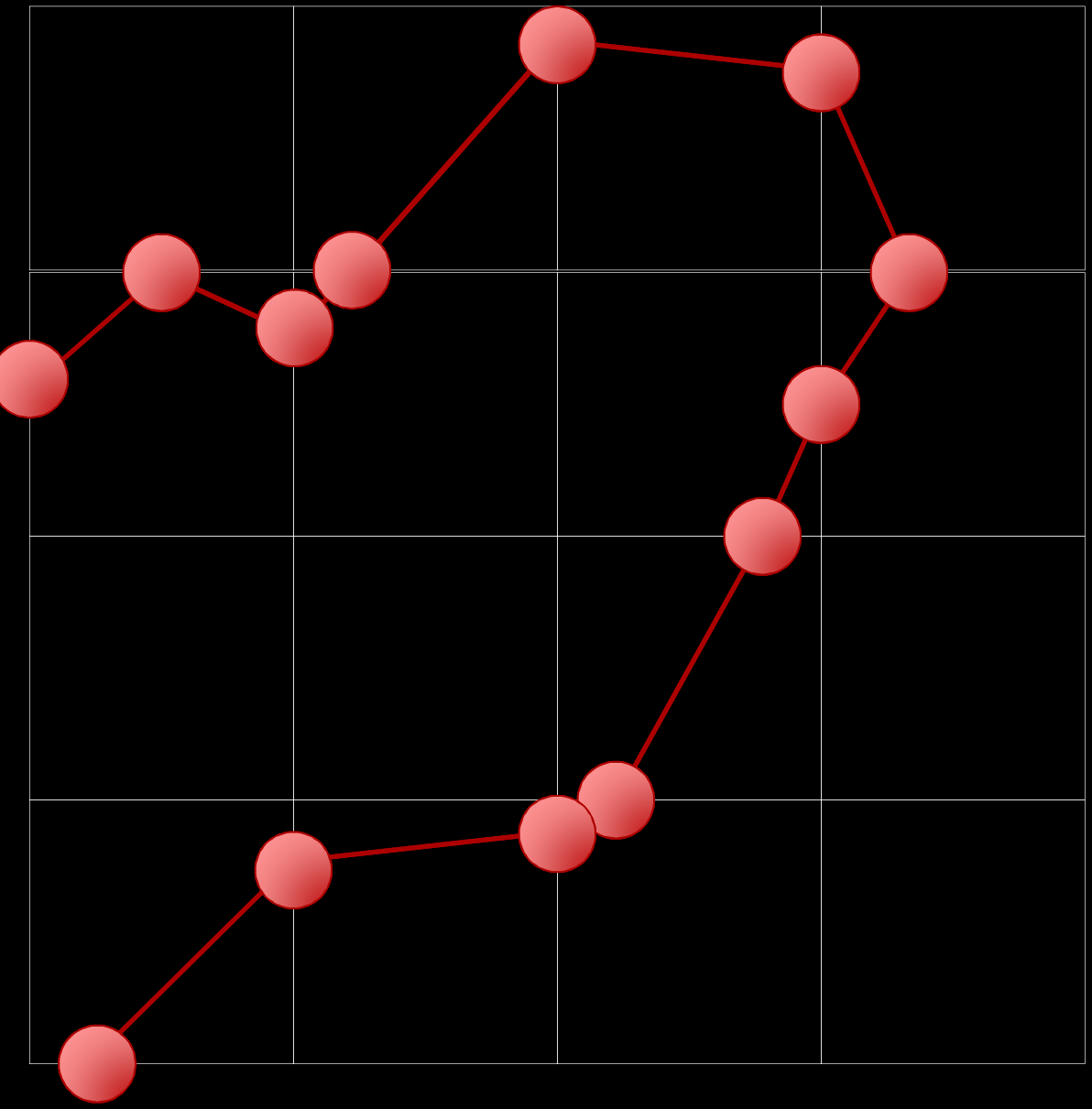


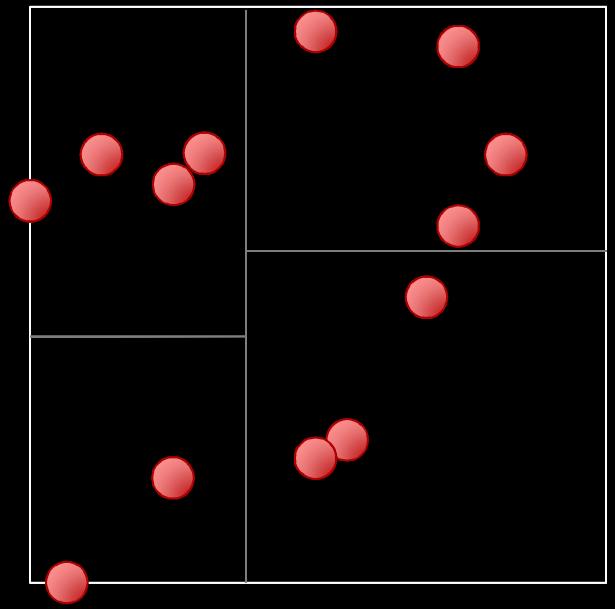


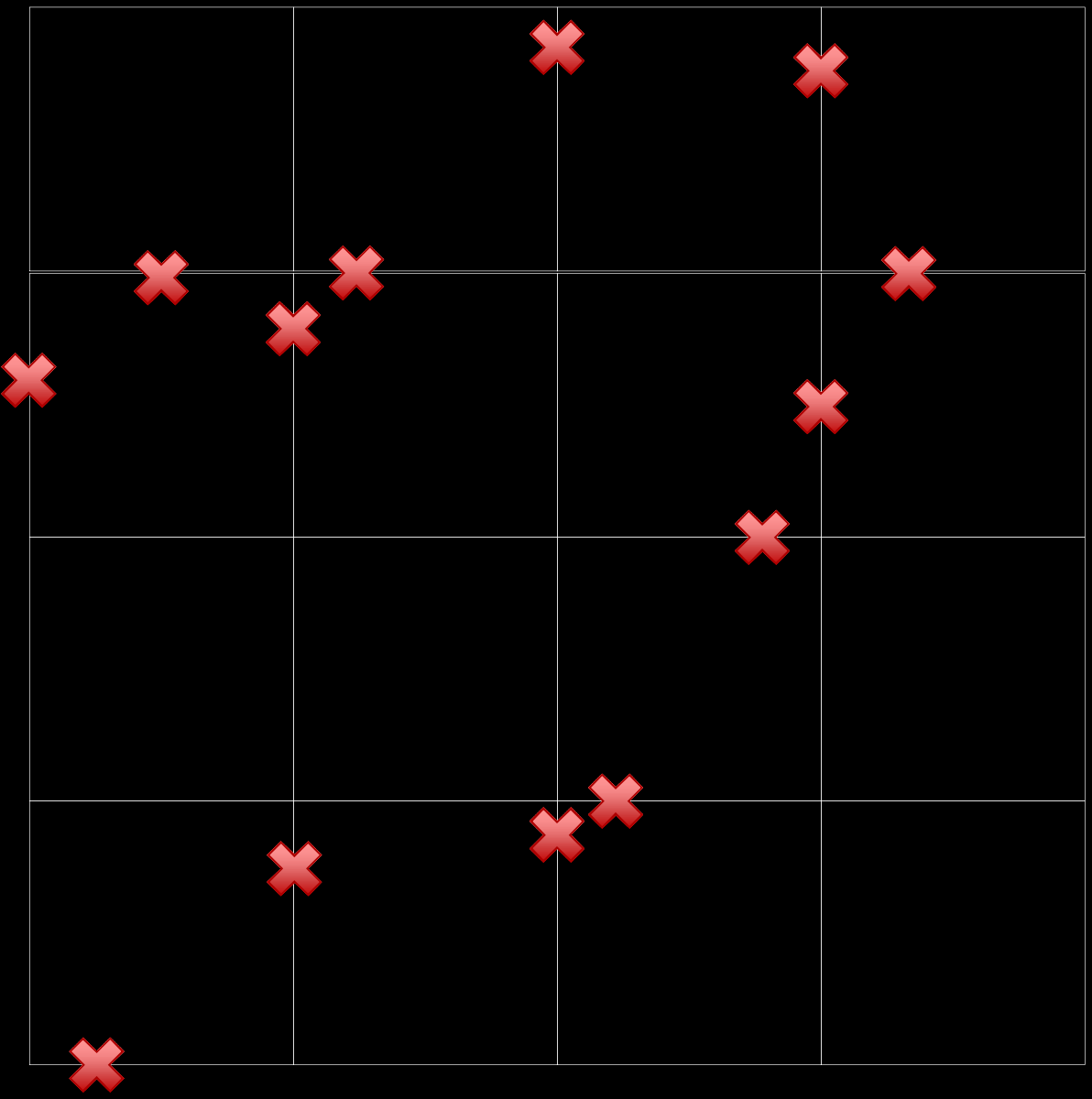


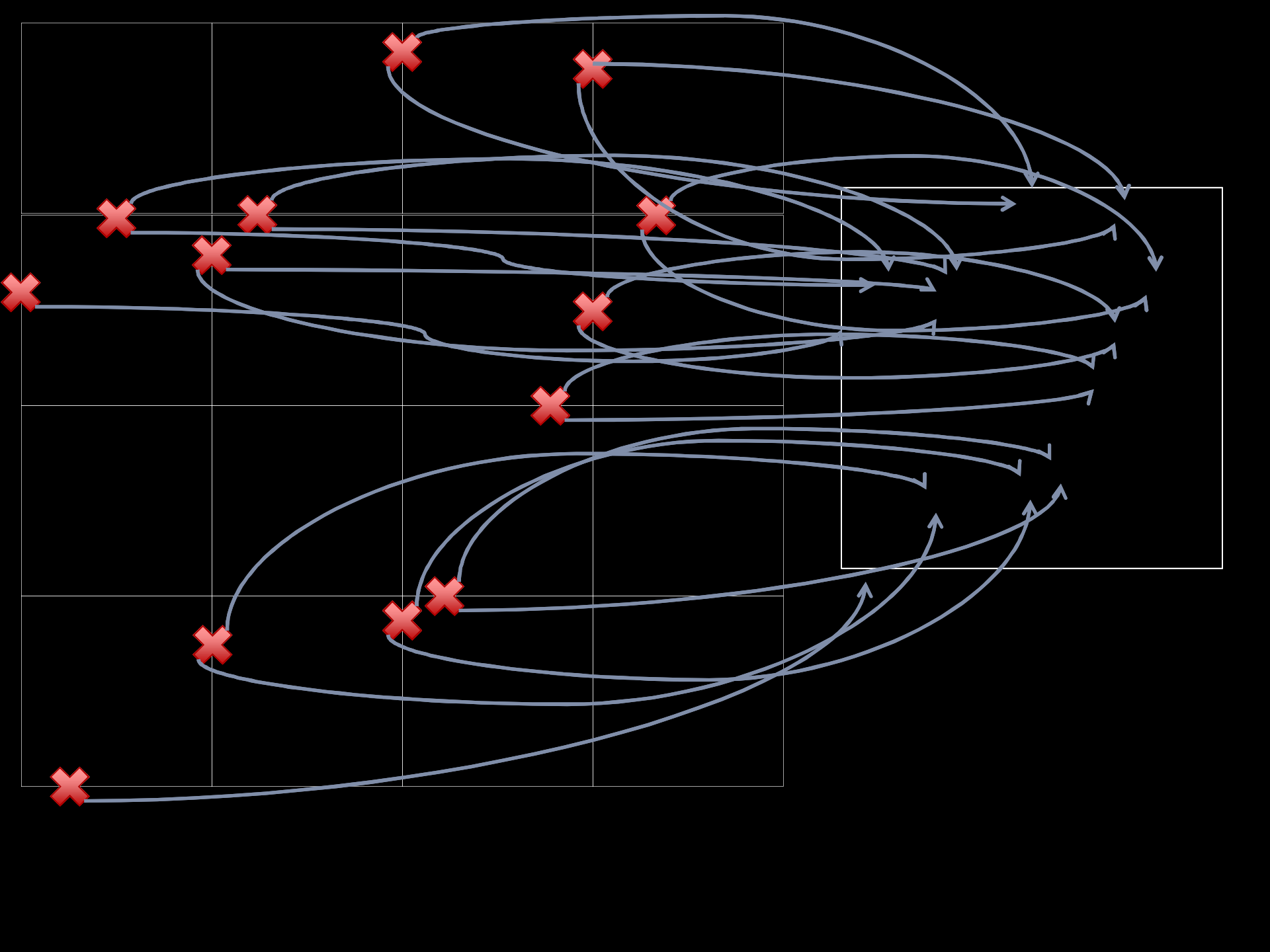


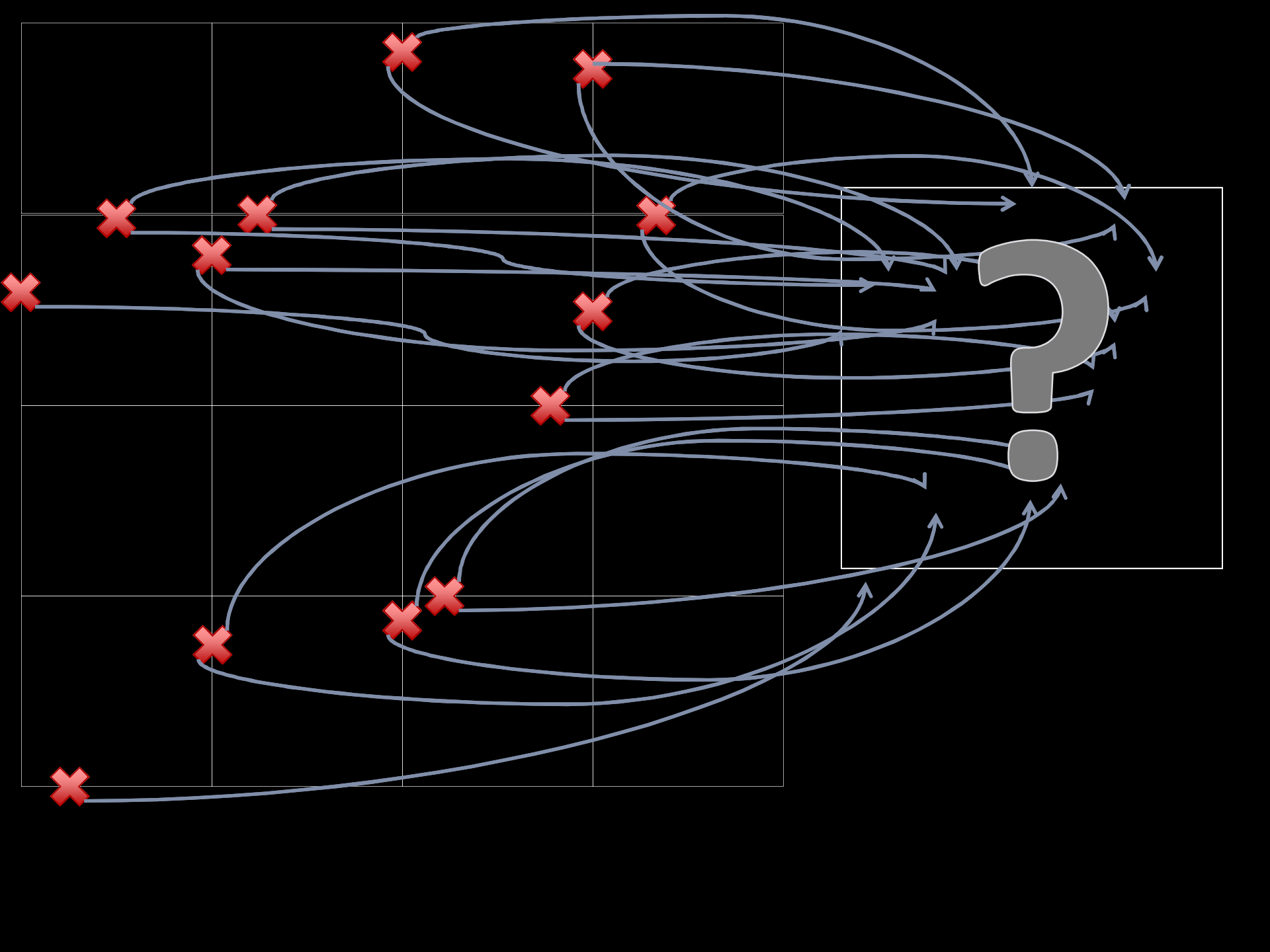


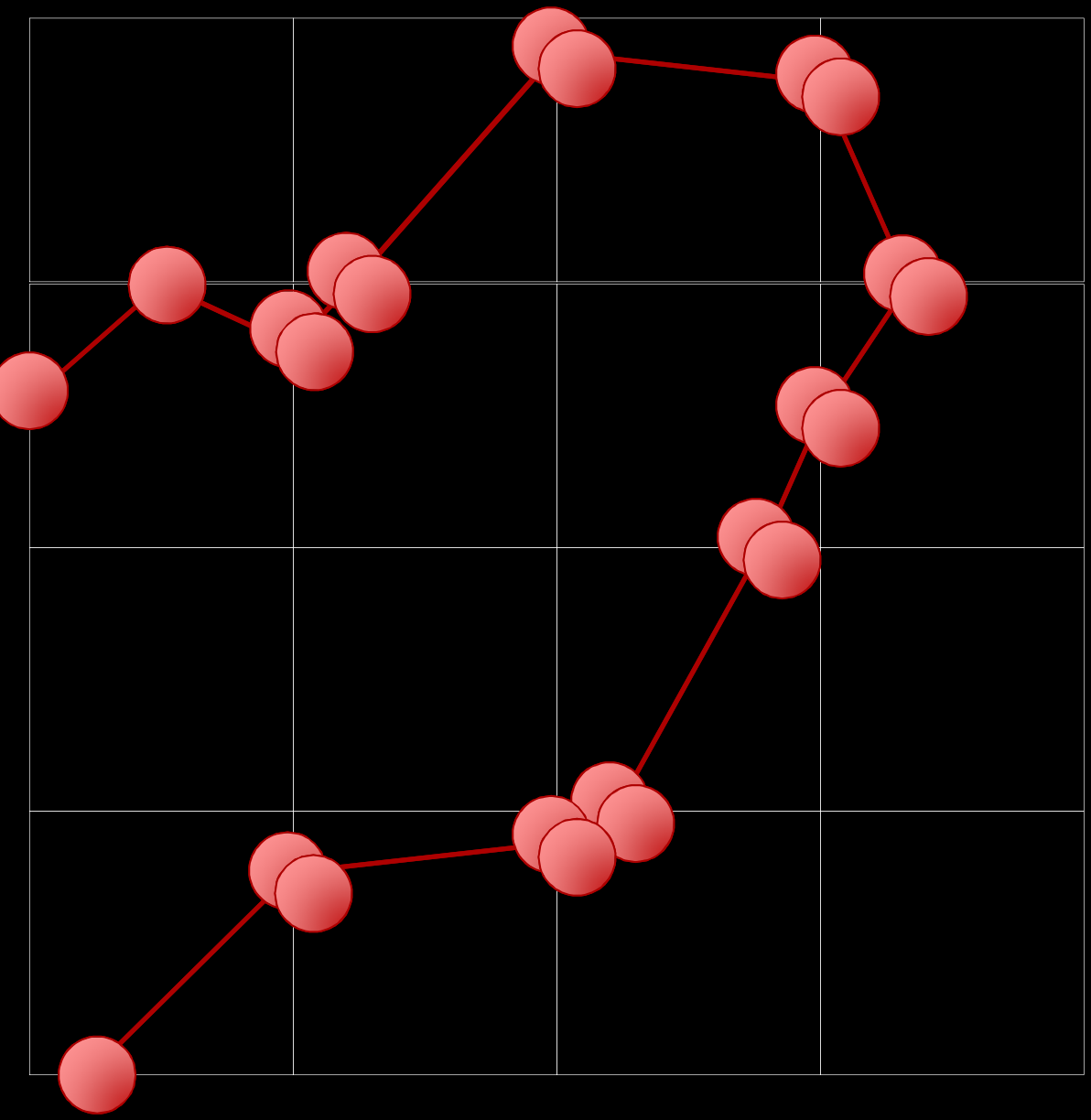






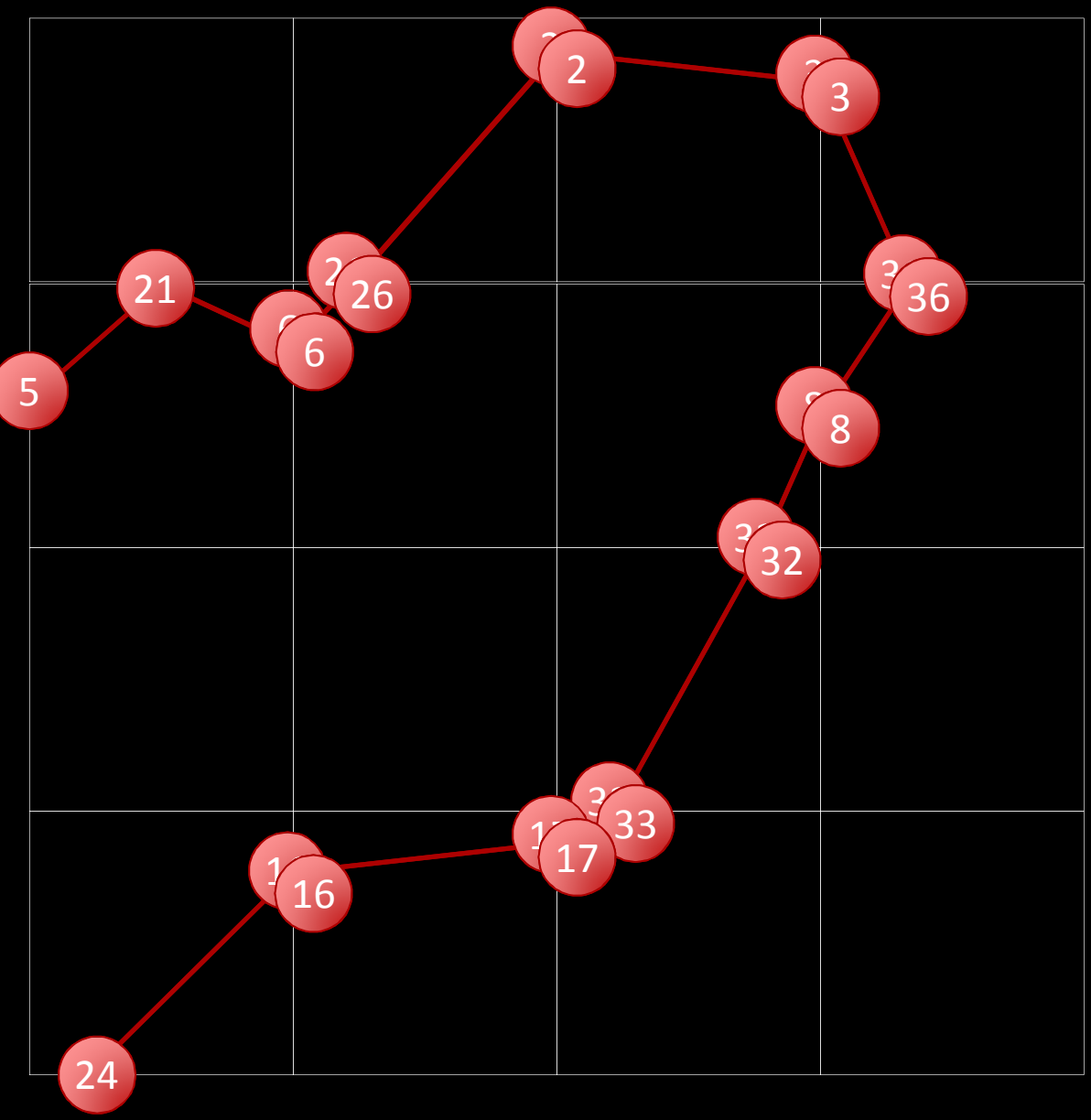


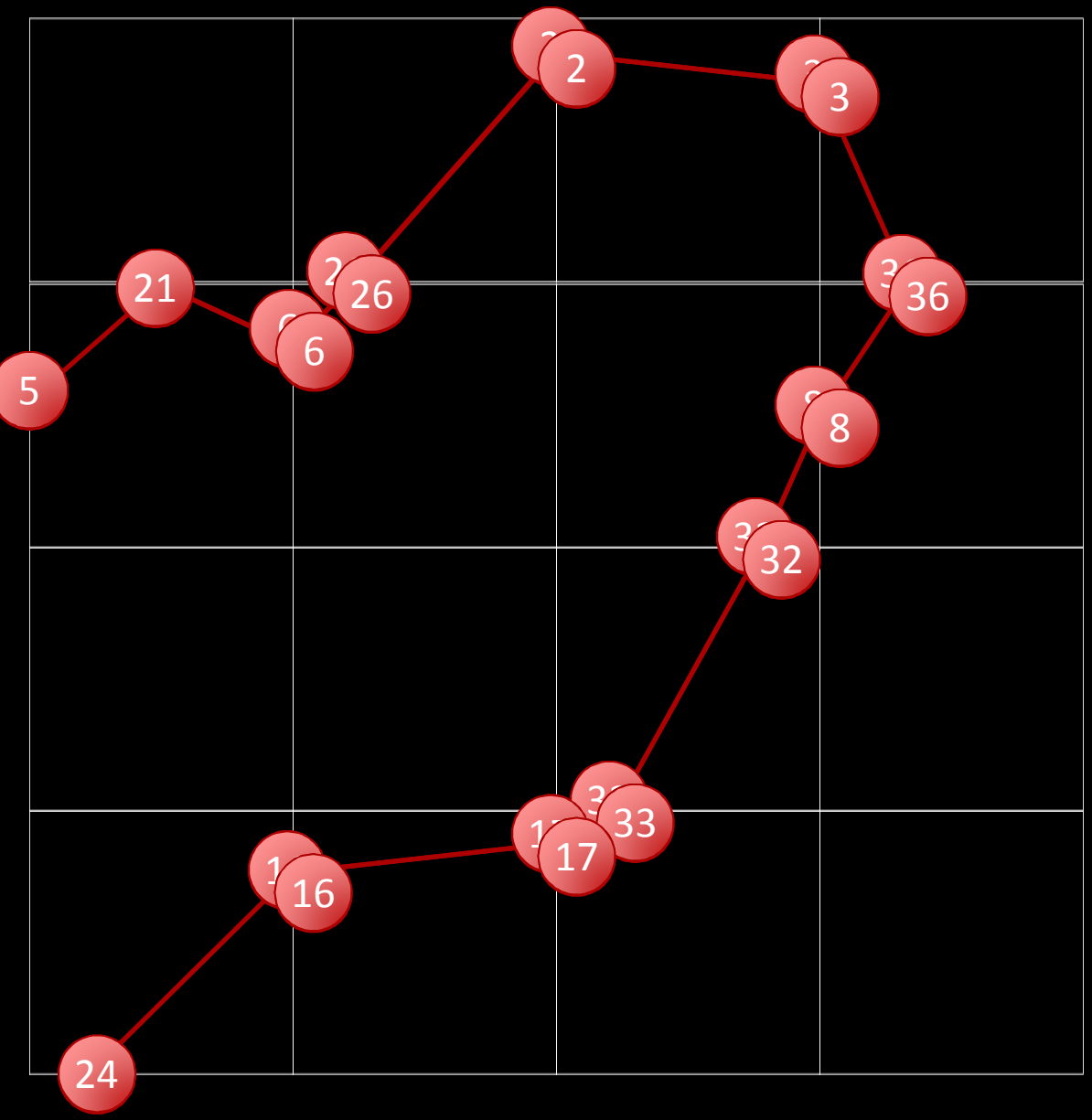


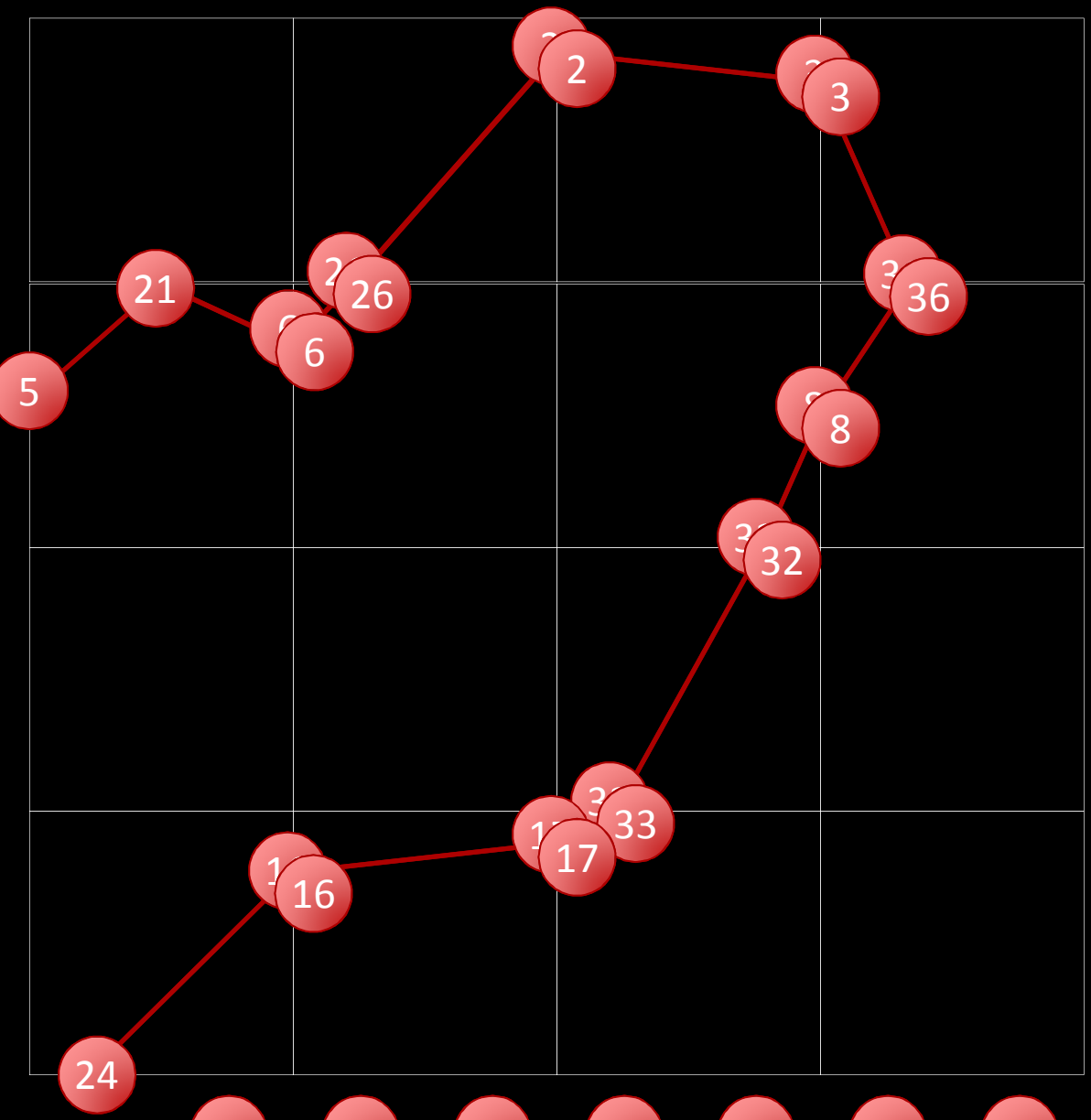


Faceted Normals

Smoothed Normals

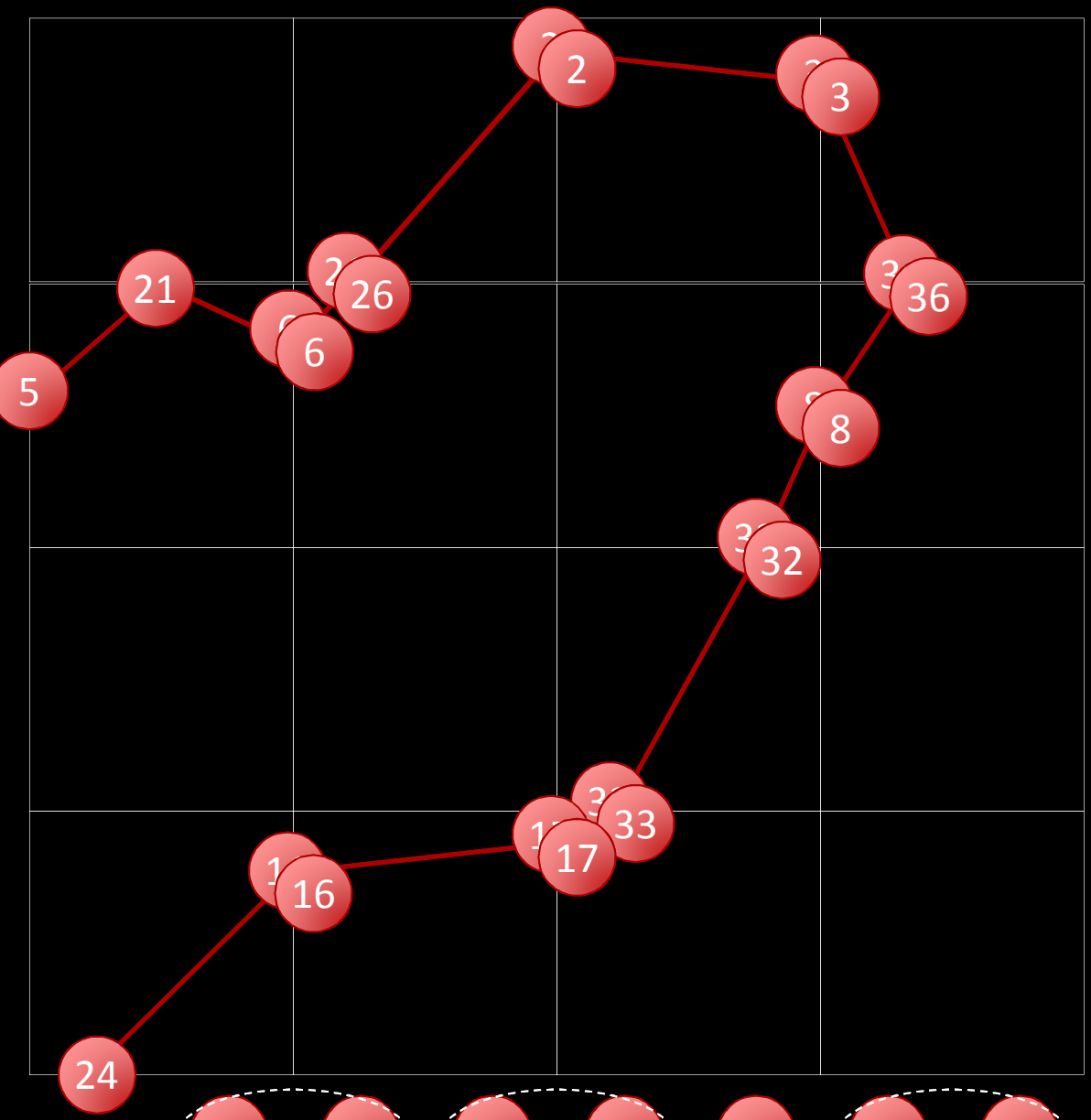

2
3
21
26
36
6
5
8
32
33
17
16
24


2
3
21
26
36
6
5
8
32
33
17
16
24



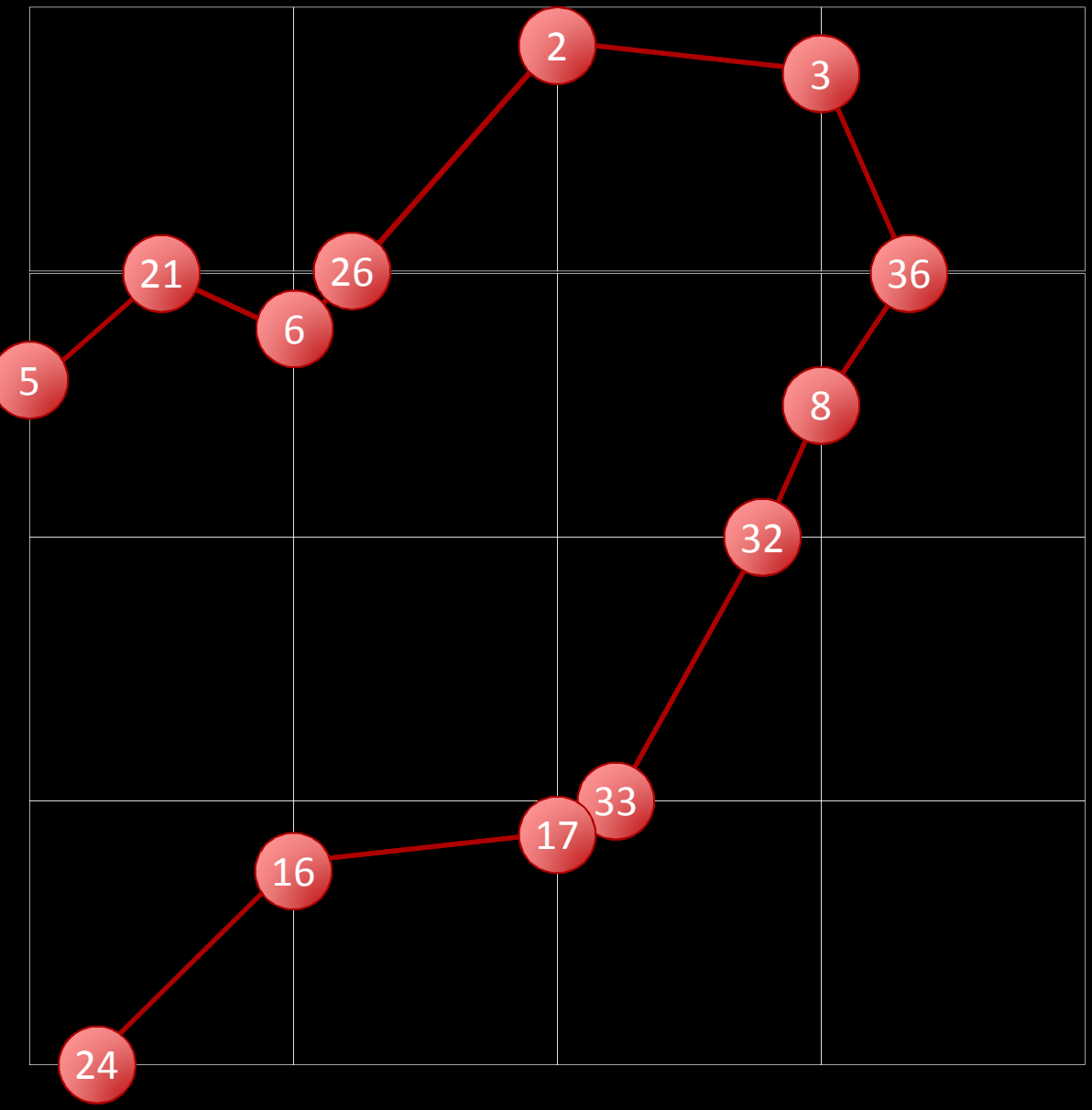
2 26 2 3 36 3 6 21 5 26 6

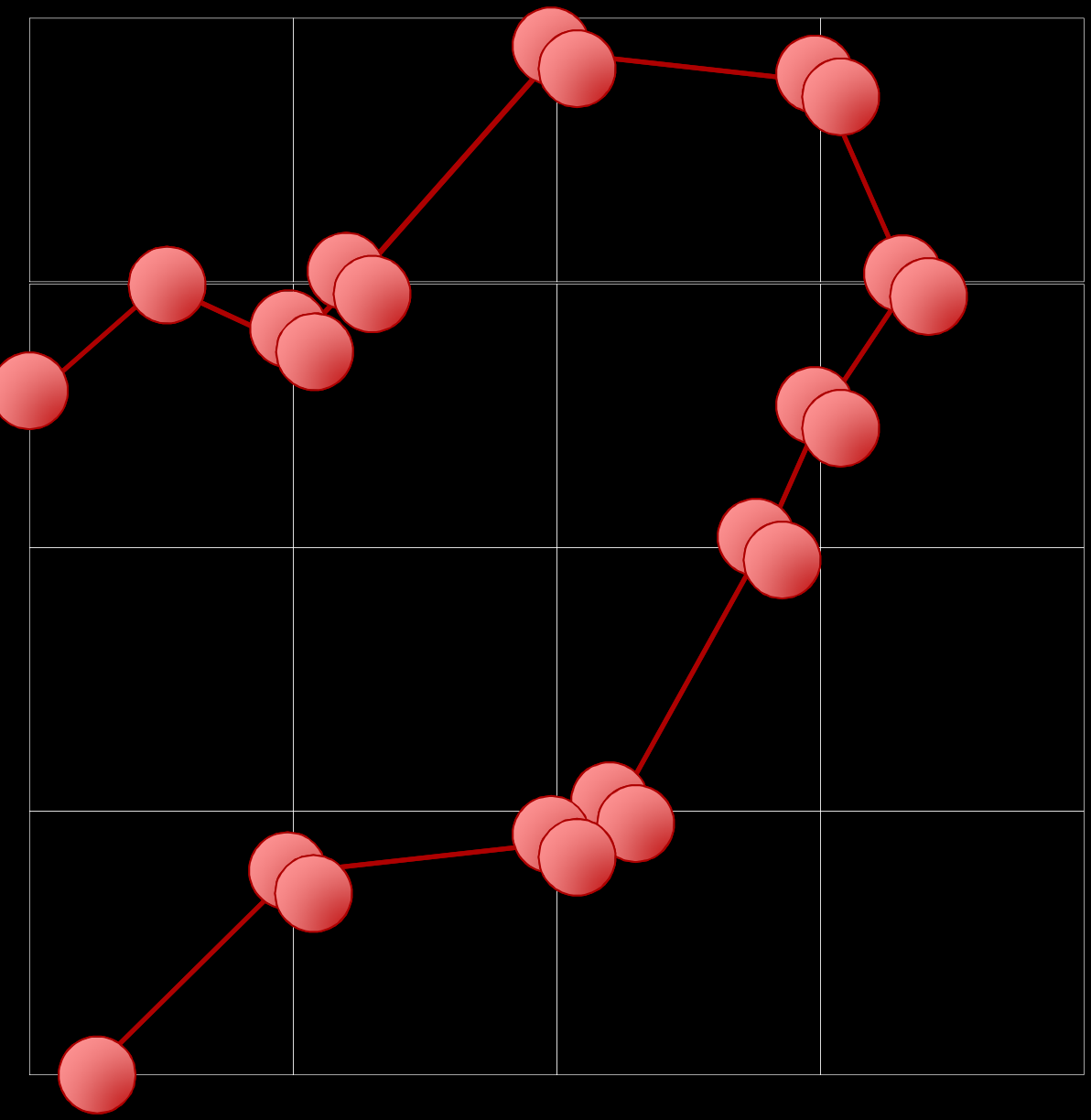
8 32 36 8 32 33 16 24 17 16 33 17

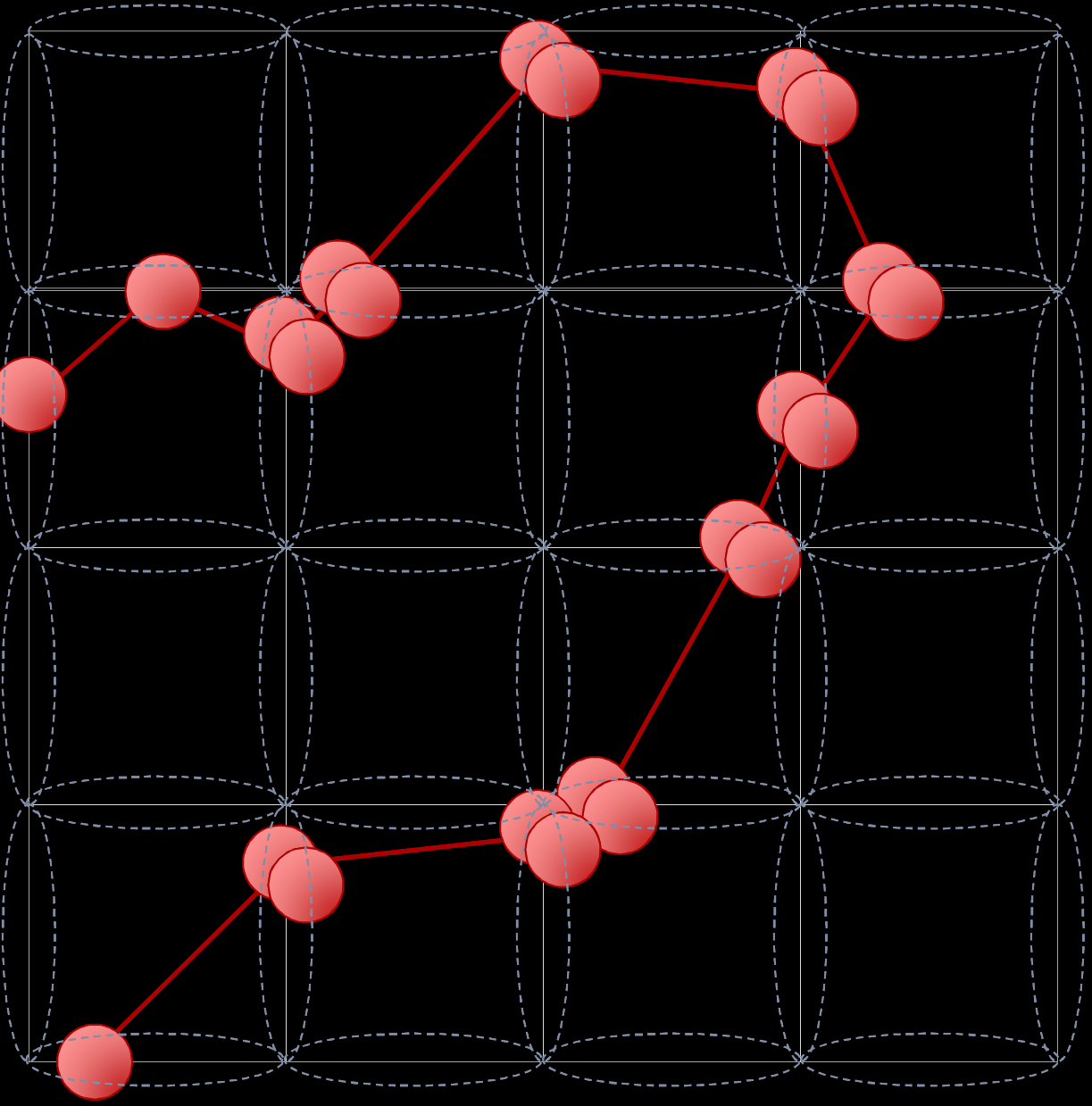


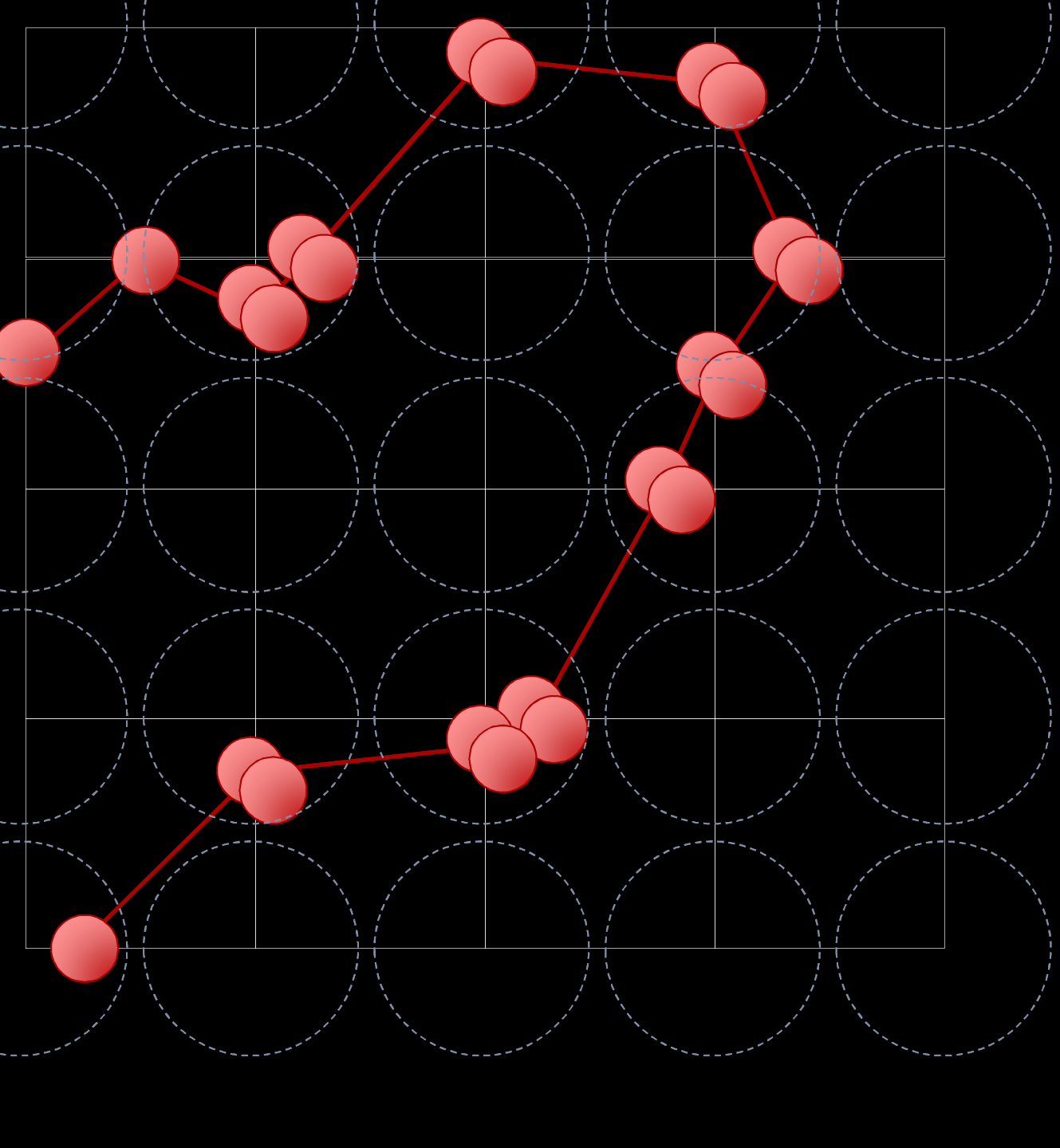
2 2 3 3 5 6 6 8 8 16 16

17 17 21 24 26 26 32 32 33 33 36 36


2
3
21
26
36
6
5
8
32
33
17
16
24



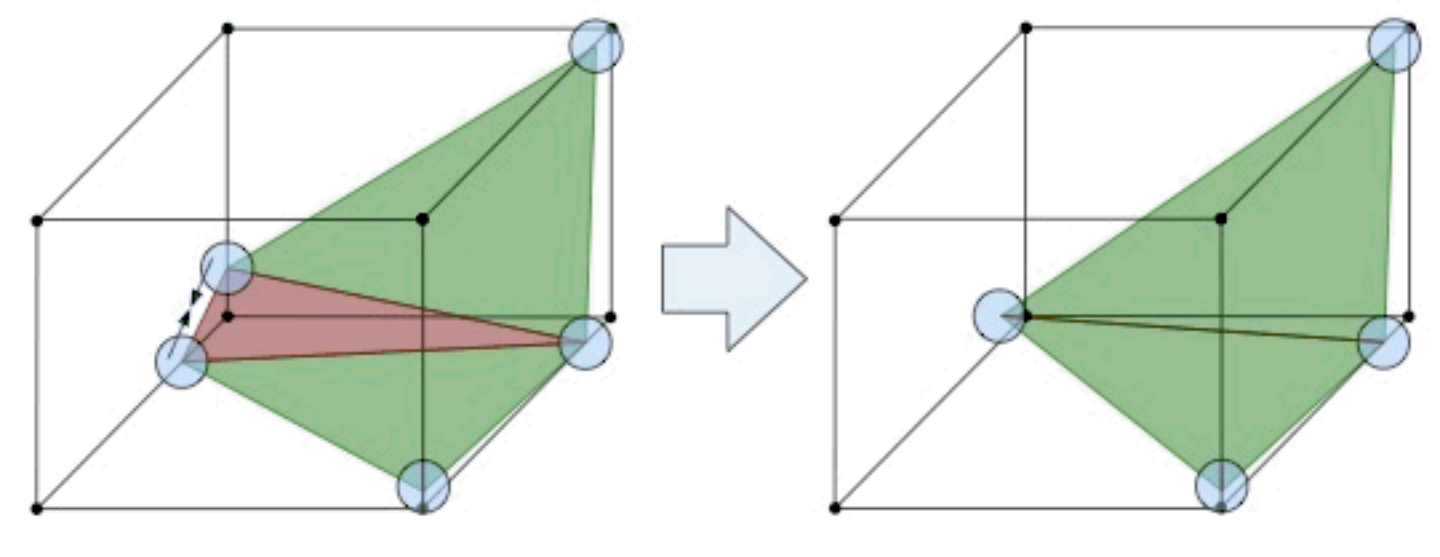


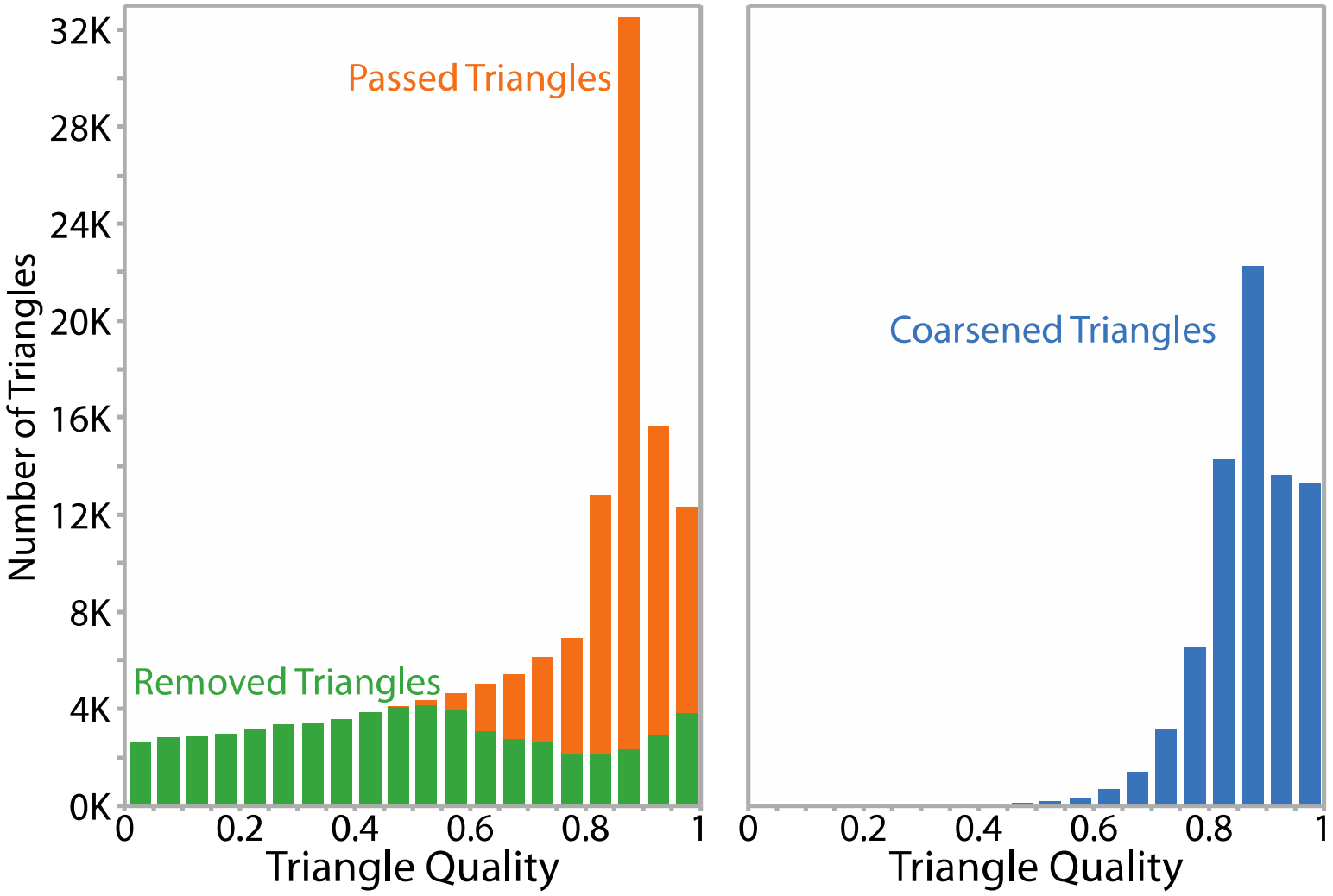


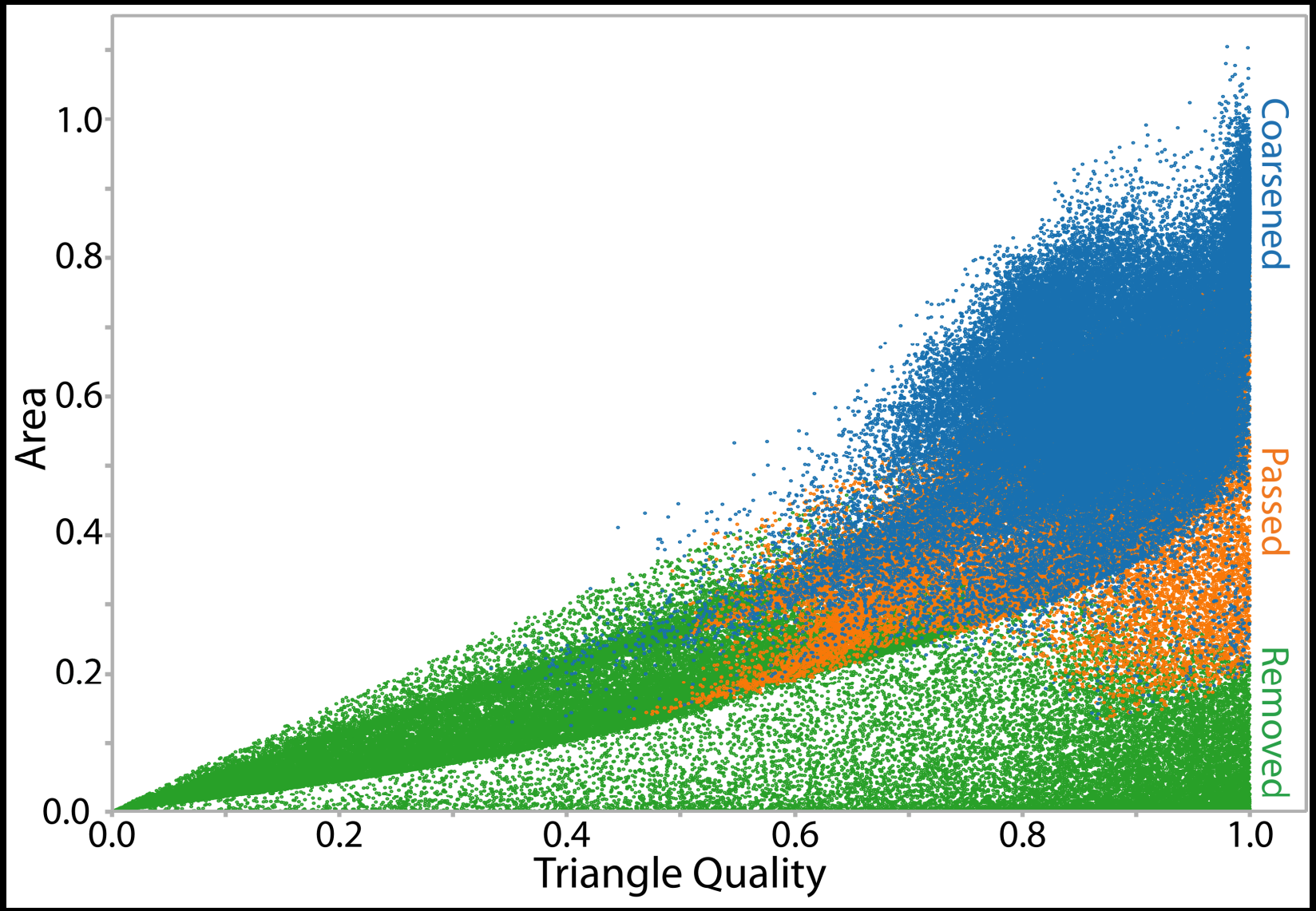
Faceted Normals

Smoothed Normals

Simplified Mesh




Number of Triangles
32K
28K
24K
20K
16K
12K
8K
4K
0K
Passed Triangles
Removed Triangles
0
0.2
0.4
0.6
0.8
1
Triangle Quality
Coarsened Triangles
0
0.2
0.4
0.6
0.8
1
Triangle Quality


Area
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Triangle Quality
Coarsened
Passed
Removed

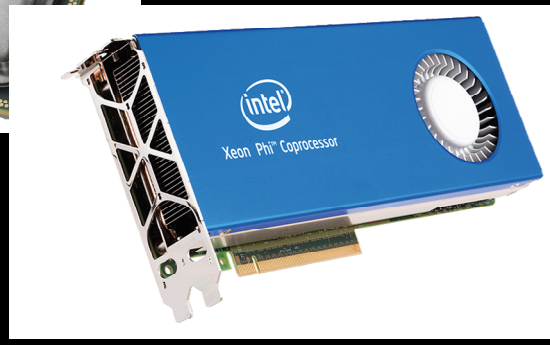
Like a washbasin knocking on your door…

we’re gonna let that sink in.

How Many Architectures To Support?

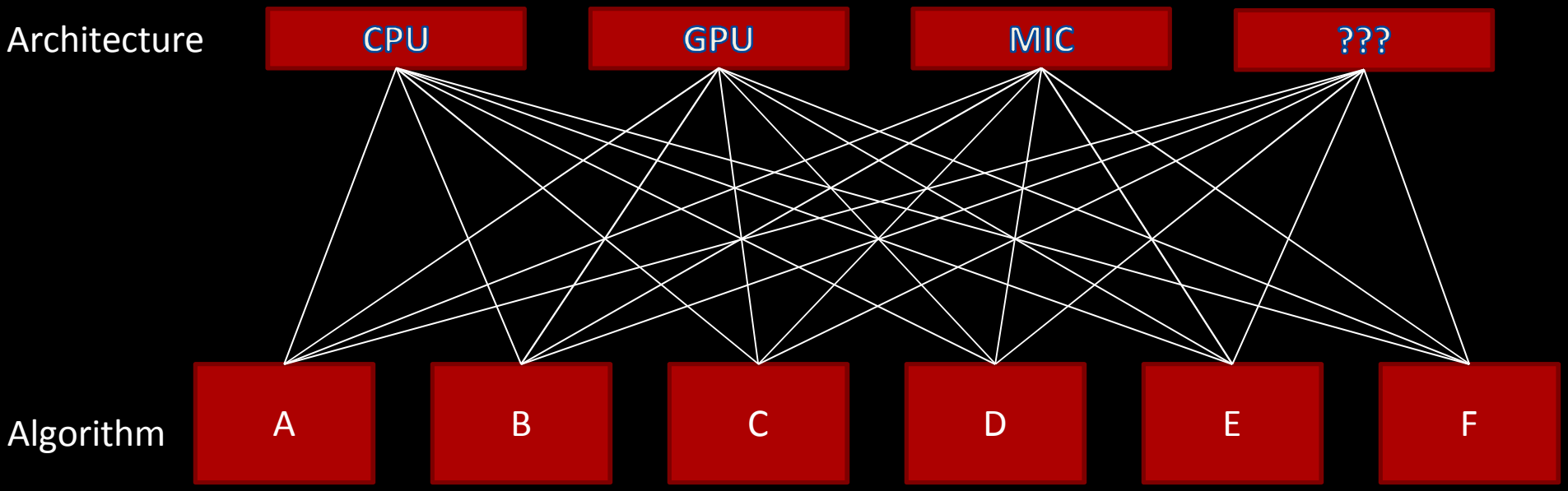
- GPU (NVIDIA)
  - Sub-architectures :
    - Fermi, Kepler, Maxwell
  - Multiple Memory Types:
    - Global, shared, constant, texture
  - Memory Amount:
    - Up to 12 GB
  - 1000s of threads
    - Grids, Blocks, and Warps

- CPU/MIC
  - Multiple ISAs:
    - Vector Unit Widths:
      - » 2,4,8 / 16
  - Single Memory Type
  - Larger Memory Size (CPU)
  - Up to 20/60 threads
    - No explicit organization



Performance Portability

- Visualization developers faced with a decision:
  - Pick a target architecture
  - Add additional implementations of the same algorithm:



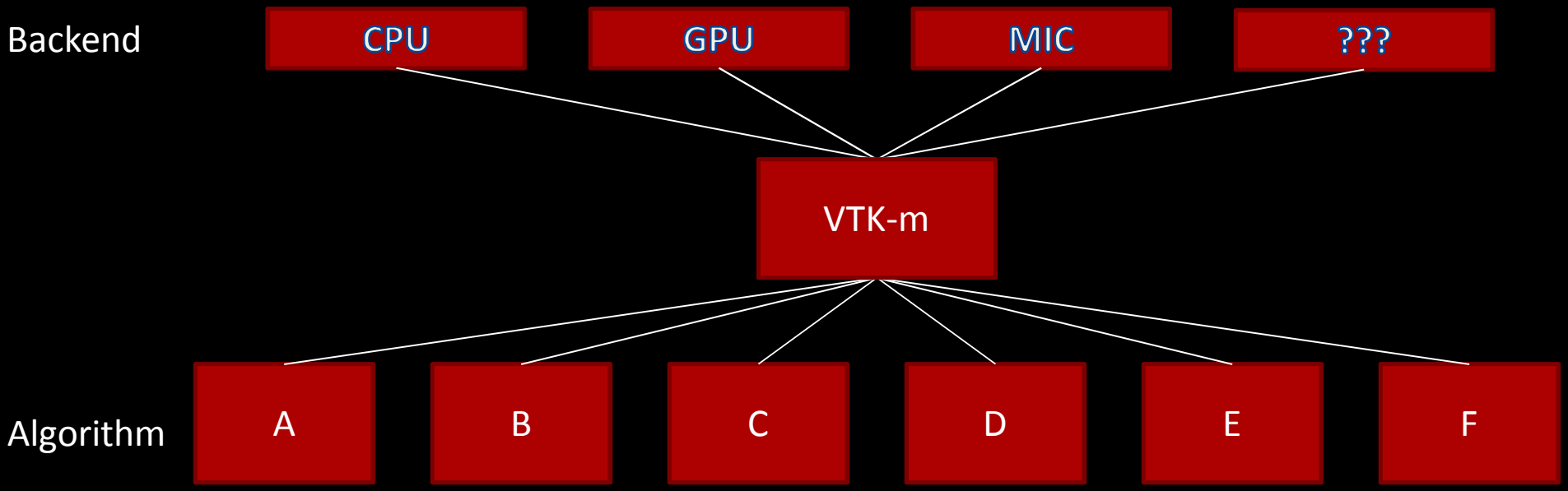
Data Parallel Primitives

Sandia National Laboratories

- What are they?
  - Provide a level of abstraction based on Blelloch's parallel primitive operators
    - Examples: Map, Reduce, Scan, Gather, Scatter, Sort
    - Provides node-level parallelism
- Benefits
  - Portable performance
  - Future-proofed implementations
  - Higher productivity

Data Parallel Primitives Benefit

- Backend: Implement fast parallel primitive operators for each new architecture
- Frontend: Re-think current algorithms in terms of the primitives



1. input

transform(classify_cell)

2. caseNums

3. numVertices

transform_inclusive_scan(is_valid_cell)

4. validCellEnum

5. CountingIterator

upper_bound

6. validCellIndices

make_permutation_iterator

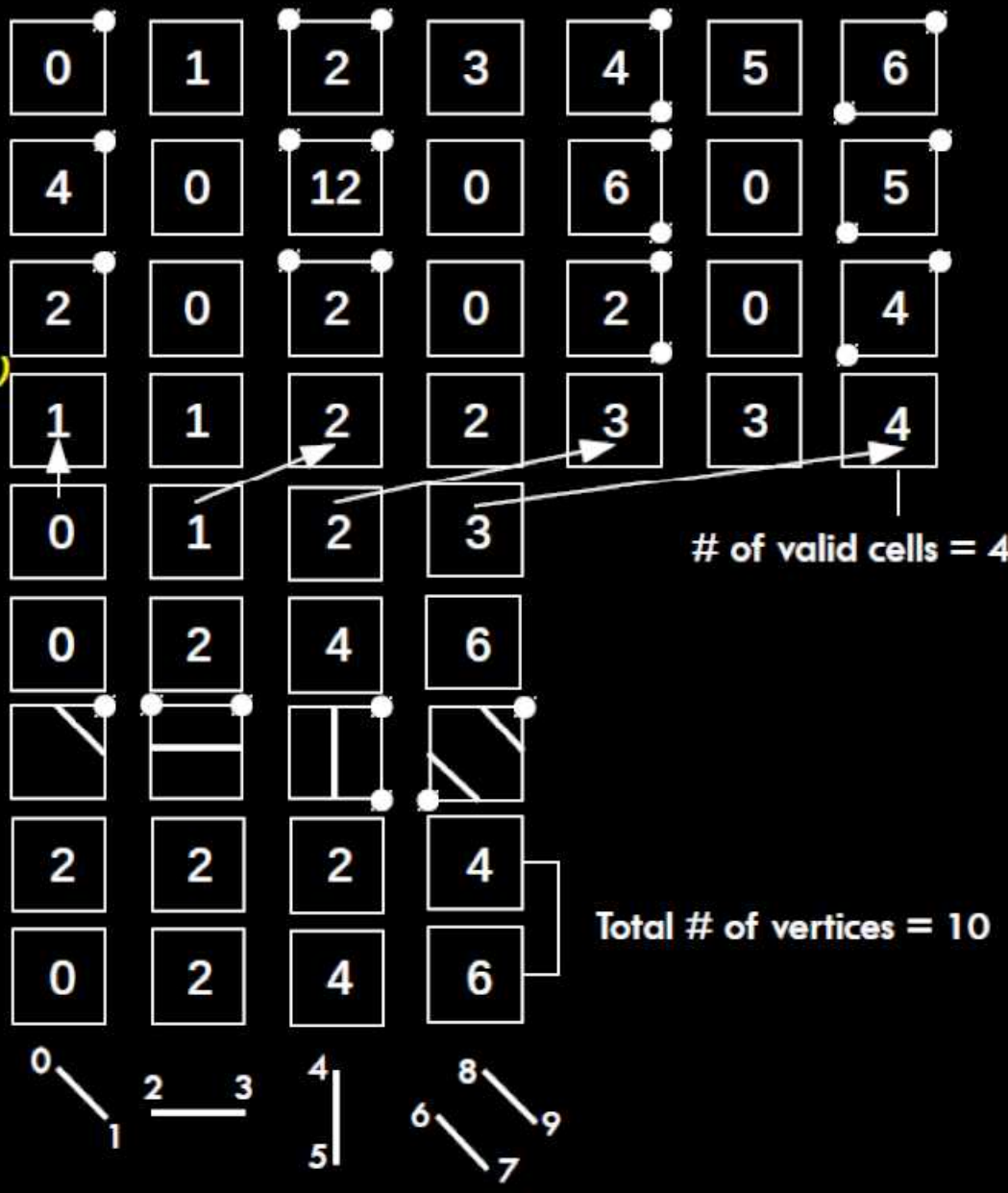
7. numVerticesCompacted

exclusive_scan

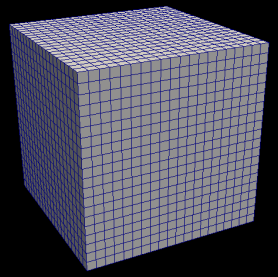
8. numVerticesEnum

for_each(isosurface_functor)

9. outputVertices



VTK-m Framework



Control Environment

Grid Topology
Array Handle
Invoke

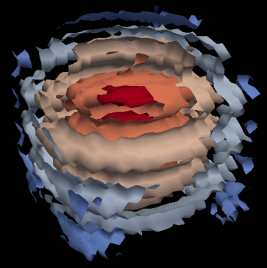
Device Adapter

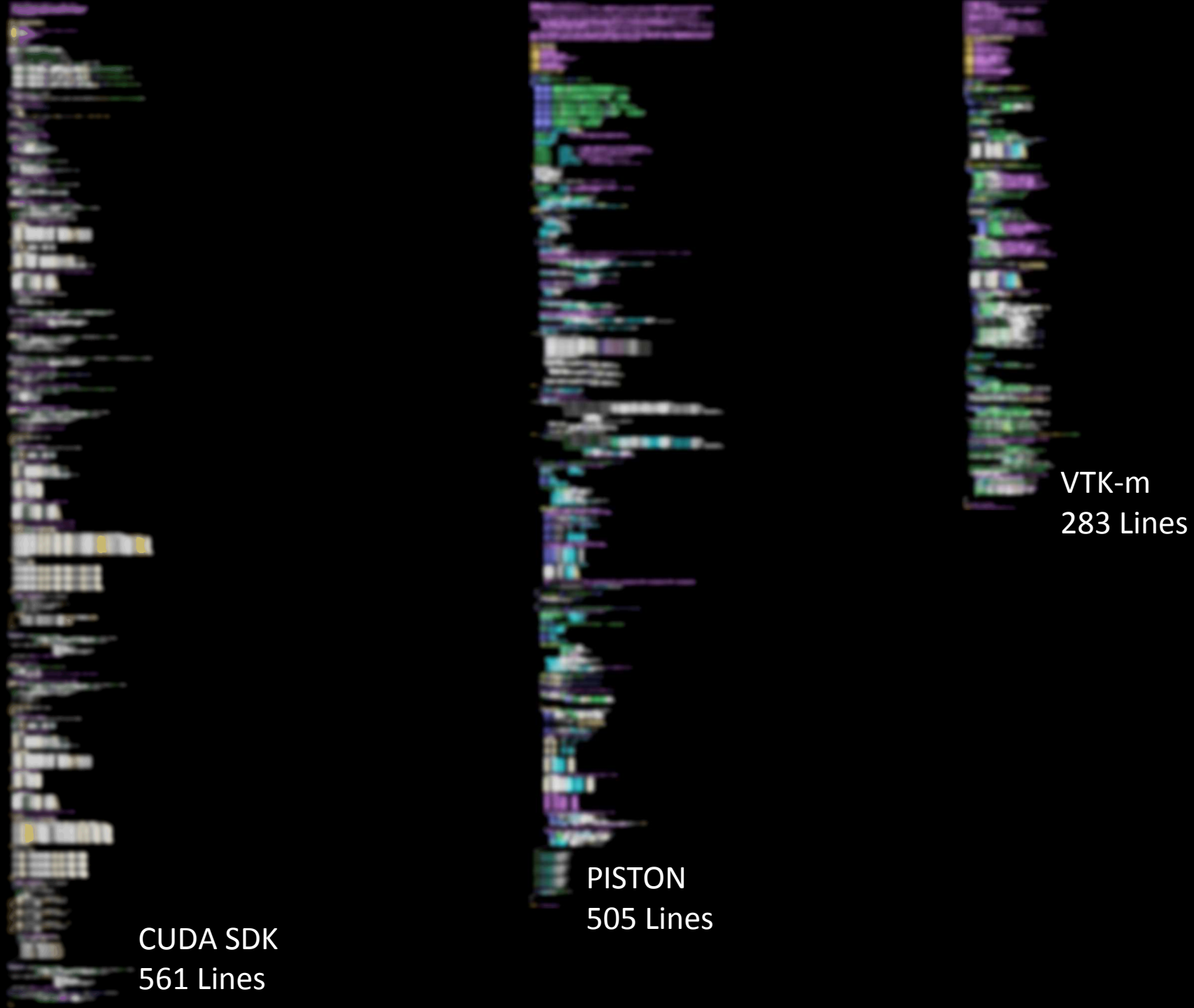
Allocate
Transfer
Schedule
Sort
…

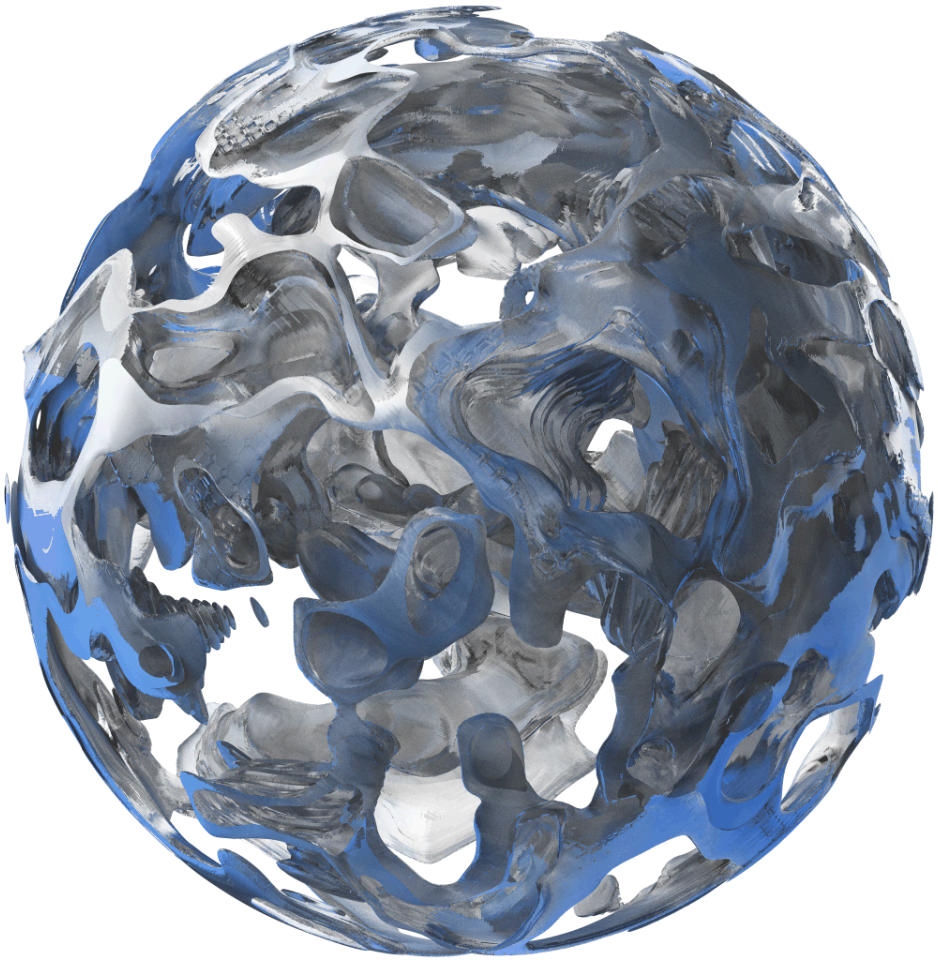
Execution Environment

Cell Operations
Field Operations
Basic Math
Make Cells

Worklet



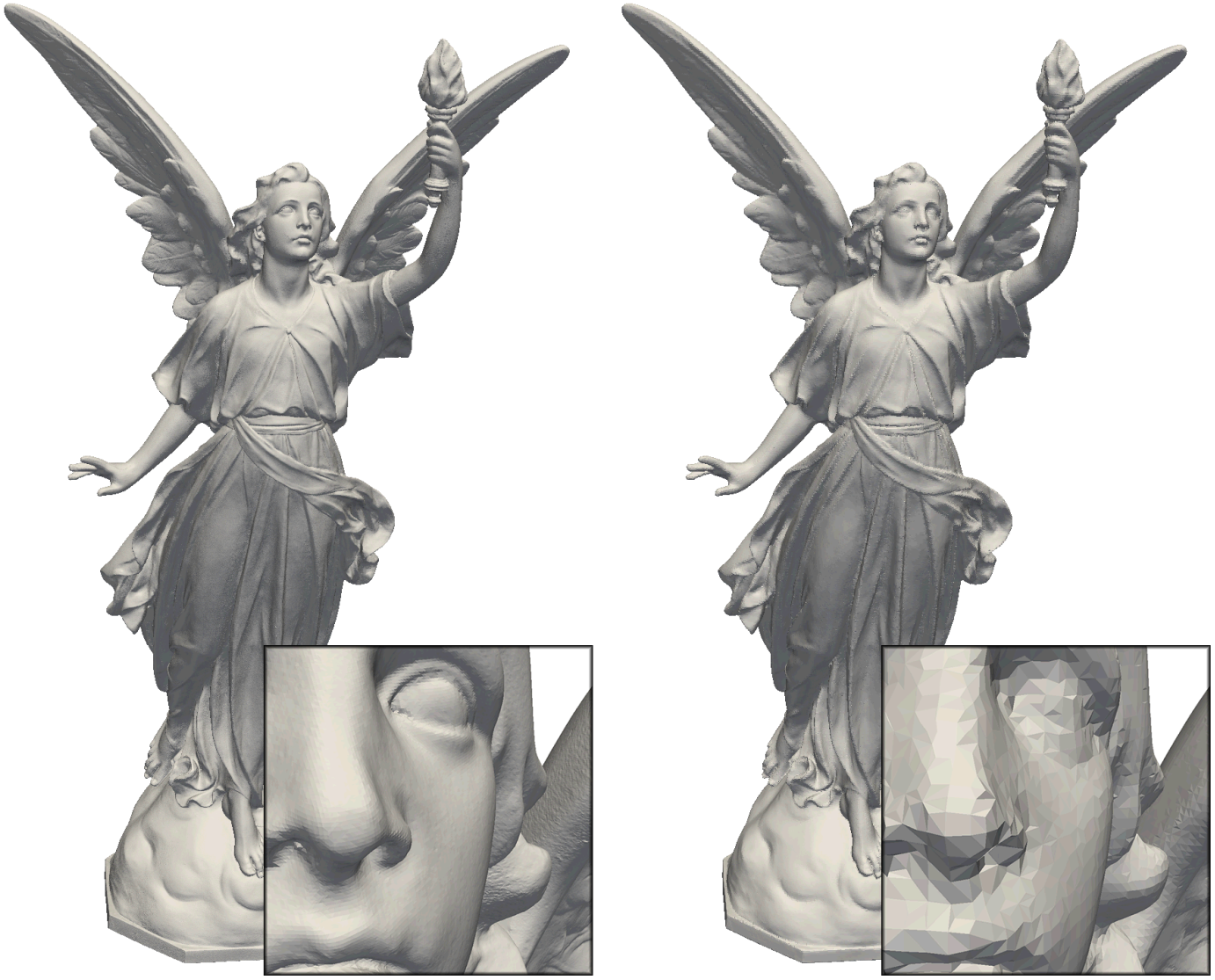

CUDA SDK
561 Lines
PISTON
505 Lines
VTK-m
283 Lines



Rendering

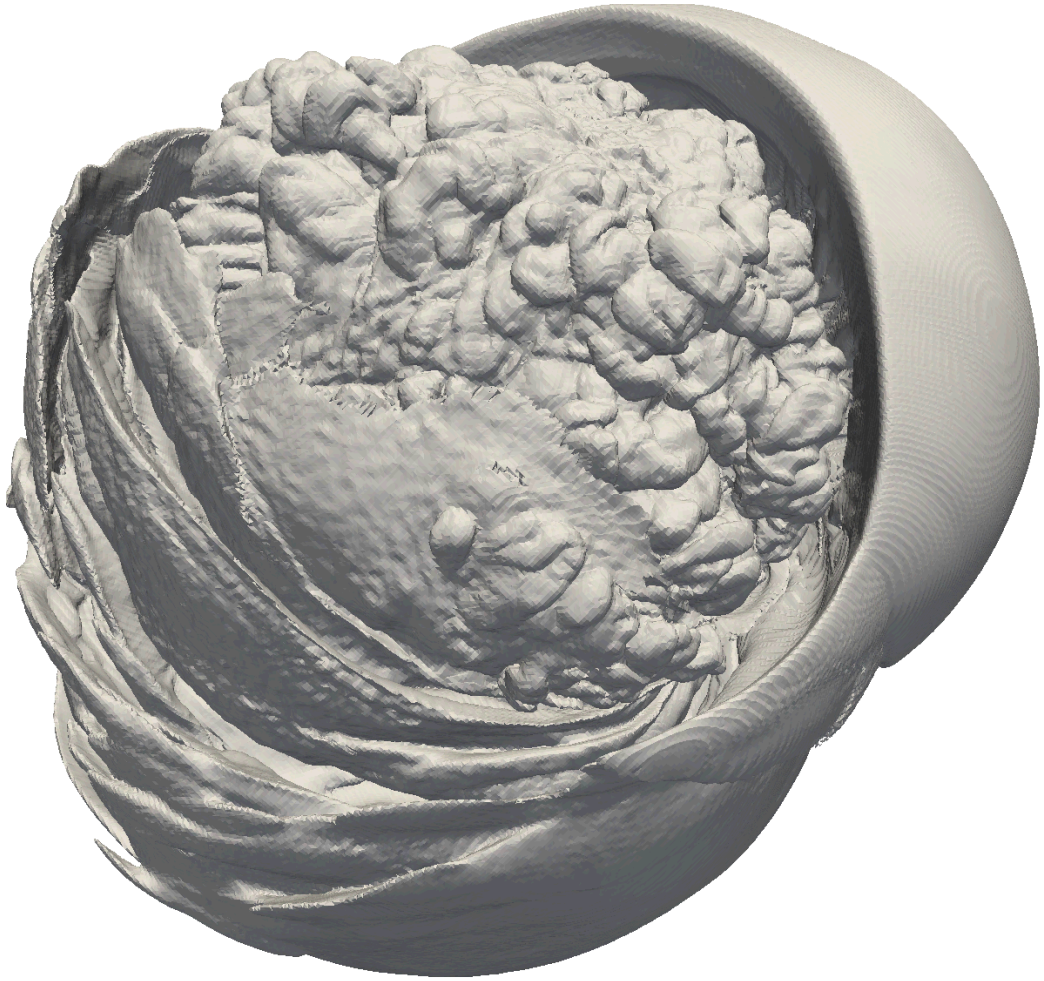
Data Set	Algorithm	Millions of Rays Per Second
LT_350K	OptiX Prime	357.6
	EAVL	150.8
	VTK-m	164.5
LT_372K	OptiX Prime	322.4
	EAVL	124.7
	VTK-m	140.8
RM_350K	OptiX Prime	436.5
	EAVL	197.5
	VTK-m	200.8
RM_650K	OptiX Prime	420.4
	EAVL	172.9
	VTK-m	166.0
RM_970K	OptiX Prime	347.1
	EAVL	152.8
	VTK-m	163.5
RM_1.7M	OptiX Prime	266.8
	EAVL	136.6
	VTK-m	148.8
RM_3.2M	OptiX Prime	264.5
	EAVL	124.8
	VTK-m	134.5
Seismic	OptiX Prime	267.8
	EAVL	106.3
	VTK-m	119.4

Data Set	Algorithm	Millions of Rays Per Second
LT_350K	Embree	51.9
	EAVL	27.7
	VTK-m	38.5
LT_372K	Embree	56.5
	EAVL	26.1
	VTK-m	36.0
RM_350K	Embree	64.8
	EAVL	33.3
	VTK-m	47.8
RM_650K	Embree	65.9
	EAVL	35.6
	VTK-m	49.1
RM_970K	Embree	59.1
	EAVL	29.3
	VTK-m	41.0
RM_1.7M	Embree	52.4
	EAVL	27.0
	VTK-m	37.8
RM_3.2M	Embree	48.4
	EAVL	28.3
	VTK-m	33.9
Seismic	Embree	43.2
	EAVL	25.2
	VTK-m	34.5



Surface Simplification

Algorithm	Device	512^3	1024^3	2048^3
VTK	Serial	3.65 s	11.40 s	
VTK-m	Serial	2.73 s	2.93 s	5.22 s
VTK-m	TBB 36 threads	0.36 s	0.45 s	0.72 s
VTK-m	TBB 72 threads	0.41 s	0.49 s	0.74 s
VTK-m	CUDA	0.18 s	0.19 s	0.20 s



Algorithm	Device	Time
vtkMarchingCubes	Serial	11.917 s
vtkMarchingCubes	32 MPI Ranks	1.352 s
vtkMarchingCubes	64 MPI Ranks	1.922 s
PISTON	Serial	19.895 s
PISTON	CUDA	0.514 s
PISTON	TBB	0.955 s
VTK-m	Serial	20.784 s
VTK-m	CUDA	0.560 s
VTK-m	TBB	1.161 s

Summary

Sandia National Laboratories

- Processing in HPC has rapidly evolved
  - Visualization (as with the rest of HPC computing) needs to adjust
- There is lots of parallelism available for visualization
  - But taking advantage of it is not always straightforward
- Updating algorithms is a lengthy process
  - There are lots of distinct algorithms (more than 400 in VTK)
- Data parallel primitives simplify the implementation and provide performance portability
- VTK-m is collecting and implementing these algorithms

http://m.vtk.org

Acknowledgements

- This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, under Award Numbers 10-014707, 12-015215, and 14-017566.
- This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering, and early testbed platforms, in support of the nation's exascale computing imperative.
- Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.
- **Thanks to many, many partners in labs, universities, and industry.**