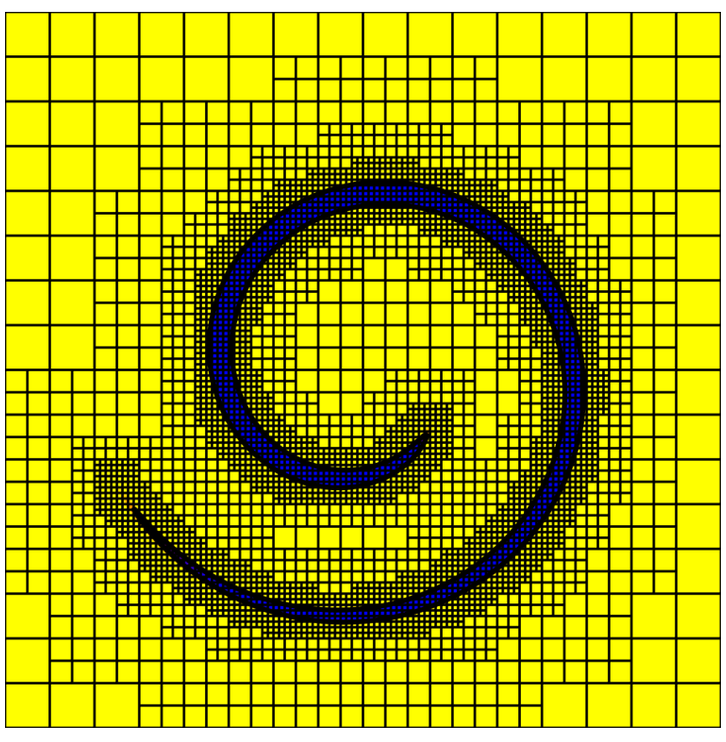
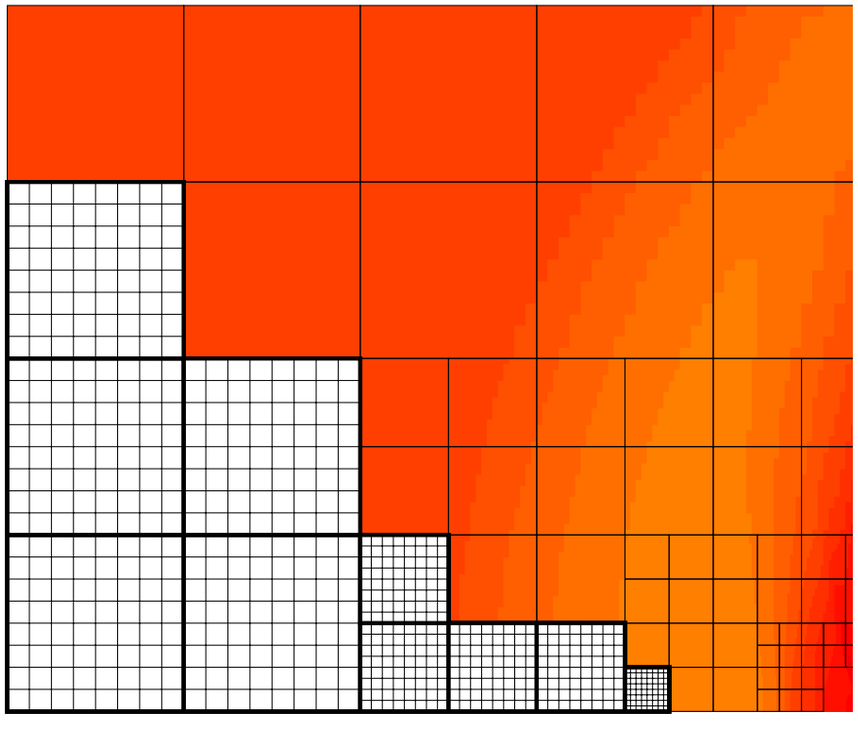
ForestClaw : Parallel, adaptive mesh refinement for Cartesian block structured meshes

Goal : Improve performance of codes used in natural hazards modeling (tsunamis, storm surges, flooding, debris flows, avalanches, earthquakes and so on).

Solution method : Explicit finite volume methods on structured Cartesian blocks. **Patches are dynamically created and destroyed** to follow the solution features of interest.





GPU approach : Improve performance of kernels used to update the solution on a Cartesian patch.

Approaches

First approach :

1. **Allocate** memory on the GPU; copy solution to the GPU; call kernel and update the solution; copy solution back from GPU; **deallocate** memory
   -- Really **slow** (too many **cudaMalloc**'s)

Second approach :

1. **Allocate memory only once** when patch is created
   -- Significant improvement in speed

Test	WALLTIME	ADVANCE	AMR	GHOSTFILL	MEMCOPY	ALLOCATE	STEPS
T1 (gpu)	964.406	940.189	3.981	17.141	135.628	**655.382**	2.38554e+06
T2 (gpu)	286.727	260.483	4.958	17.676	127.956	**1.357**	2.38554e+06

Current approach : (20x speed up in ADVANCE**)**

Test	WALLTIME	ADVANCE	GFILL	B4STEP2	STEP2	TOTAL STEPS
T1 (gpu)	140.612	137.670	4.691	128.616	8.936	709636
T2 (cpu)	170.595	**167.298**	4.616	128.629	38.601	709636
T3 (gpu)	15.263	**9.941**	4.798	0	9.866	707802