Distributed Newest Vertex Bisection

in DUNE-ALUGRID

Martin Alkämper and Robert Klöfkorn

DUNE User Meeting 2015
Algorithm

Some Analysis

Experiments
Problem

In **Dune-ALUGrid** (among others) we provide an adaptive, conforming, distributed, unstructured, simplicial, nested grid

$$
\rightarrow \text{We need a Parallel Refinement Algorithm that closes the grid conformingly}
$$

* Distributed Newest Vertex Bisection *
Newest Vertex Bisection in 2d

What does Newest Vertex Bisection do?

- Identify triangle with its ordered vertices and generation $\text{gen}(T) = l$
  
  $$T = [z_0, z_1, z_2]_l.$$ 

- Children with $\text{gen}(T_{1,2}) = l+1$ are defined by
  
  $$T_1 = [z_0, \bar{z}, z_1]_{l+1} \quad \text{and} \quad T_2 = [z_2, \bar{z}, z_1]_{l+1}.$$ 

- Refinement edge is opposite of newest vertex:
Distributed Newest Vertex Bisection

While True:

1. Refine all marked and nonconforming elements by Newest Vertex Bisection on each partition
2. Communicate refinement status of partition boundaries to corresponding neighbour
3. Mark all elements with hanging nodes for refinement
4. Communicate globally, whether set of marked elements is empty on each partition
5. If true on all partitions → break

We will bound on the number of iterations (= global communications).
Algorithm

Some Analysis

Experiments
Compatibility Condition

To be able to prove something we need the following condition on the initial grid.

**Definition (Compatibility Condition)**

Suppose macro elements $T, T' \in \mathcal{T}_0$ are direct neighbors with common edge $T \cap T' = E$ and level $l(T) = l(T')$. Then either $E$ is the common refinement edge of both $T$ and $T'$, or $E$ is neither the refinement edge of $T$ nor of $T'$.
First Bound

**Theorem (Global Maximum Level Difference)**

Let $M$ be the set of marked elements in the current grid $\mathcal{T}$. Then the number of iterations $N$ of the while-loop satisfies

$$N \leq \max_{T \in M} \max_{T' \in \mathcal{T}} (\text{gen}(T) - \text{gen}(T') + 1) + 1$$

The $+1$ is due to the need to communicate the final status.
Global Max Level Difference is Sharp

Mesh created by refining a macro cell into a vertex and then distributing the cells onto two partitions. Now mark one element for refinement (red).
Global Max Level Difference is Sharp

⇒ We can only be better than result 1 with additional assumptions on the partitioning.
Second Bound

Assumptions:

- 2d grid,
- Partition by macro elements. (implemented in DUNE-ALUGRID)

Theorem (Bound by Mesh Constant)

Let $T_0$ be the initial grid and for a vertex $z$ let $N_z = \{ T \in T_0 : z \subset T \}$. Then the number of iterations $N$ in the while-loop satisfies

$$N \leq \max_{T \in M} \max_{z \in T_0(T)} \left( \frac{3}{4} |N_z| - \frac{1}{4} + 2 \right) \leq \max_{z \in T_0} \left( \frac{3}{4} |N_z| - \frac{1}{4} + 2 \right)$$

Important:

- Only depends on the initial grid.
Algorithm

Some Analysis

Experiments
Unit Cube 2d With One Macro Element Per Core

The final Triangulation

Refinement Loops

Max Level
2d Refinement On Sphere - Not Compatible

A distributed sphere
Summary

- Distributed Newest Vertex Bisection does terminate
- There is a mesh constant bound that holds even for highly adaptive grids (in DUNE-ALUGRID)
- The iteration count can be bound independently of the number of processors.
- Compatibility is important.
- DUNE-ALUGRID can now handle 2d parallelism.
Thank you for your interest!