A Unified Algorithm for Adaptive Spacetime Meshing with Nonlocal Cone Constraints

Shripad Thite
University of Illinois at Urbana-Champaign

EuroCG — March 9, 2005
(Making)

Waves
Physics of Wave Propagation

- Waves propagation modeled by spacetime hyperbolic Partial Differential Equation (PDE).

**Example:** Wave equation describing displacement $u(x, t)$ about the mean.

$$u_{tt} - \omega^2 u_{xx} = 0$$

(i) wavespeed $\omega$ can be a constant or a function of the independent variables (linear PDE), or
(ii) $\omega$ can be a function of $u$ and its derivatives (nonlinear PDE).

- **Goal:** Solve PDE to compute $u(x, T)$ at target time $T$.

- **Applications:** Conservation Laws in Elastodynamics and Fluid Dynamics. Model Ocean waves, Storms, Acoustics, Gas dynamics, Traffic flow.
Adaptively construct tetrahedral spacetime meshing in $2D \times \text{Time}$.

Mesh facets satisfy **cone constraints**; *not* a Delaunay mesh.

Supports fast, accurate numerical solution of hyperbolic PDEs.

Exploit flexibility of novel discontinuous Galerkin (DG) finite element methods.

Solve a large class of interesting and useful physics problems with high fidelity: **nonlinear**, **anisotropic**.

Algorithm is easy to parallelize; *but* need some nontrivial parallel data structures.
Advancin’ front spacetime meshin’

- **Objective:** “Good quality”, size optimal, simplicial spacetime mesh that adapts to numerical error and can be solved efficiently (in parallel).
- Advancing front procedure; incremental construction in patches
- Spacetime mesh is unstructured, nonconforming.
- In $2D \times$ Time, spacetime tetrahedra with provable bounds on worst-case *temporal aspect ratio*.
- Spacetime mesh is a “*weak simplicial complex*”: adjacent tetrahedra share the facet of the smaller tetrahedron.
Local progress by tent pitching

Figures from Erickson et al., [IMR’02]
Advancin’ front spacetime meshin’

- **Front**: piecewise linear terrain whose spatial projection is a triangulation
- Advance in time local neighborhood of front by pitching a tent.
- Greedily maximize progress in time at each step.
- Decompose tent into few *causal* tetrahedra; guarantees convergence.
**Example**

**Input:** Space mesh $M$

**Output:** Tetrahedralization of $M \times [0, \infty)$
Cone Constraints
Influence and Dependence

- Spacetime domain $\bigcup_{t \geq 0} M_t = M \times [0, \infty)$

- $P$ influences $Q$ if changing input at $P$ changes output at $Q$. $Q$ is in the domain of influence of $P$.

- $P$ influences $Q$ ($P \prec Q$) if and only if $Q$ depends on $P$ ($Q \succ P$). $P$ is in the domain of dependence of $Q$.

- Approximate domains of dependence and influence by a circular symmetric double cone.
Causal Fronts

TentPitcher maximizes progress locally while maintaining causal front, allowing patches to be solved immediately.

New front is causal $\iff$ Every point in tent depends only on points on the old front.
Progress constraint

Limit progress at each step to guarantee progress in the next step.

Causality alone not enough when space mesh contains obtuse angles.
Next step is legal
Causality violated in next step
Forbidden zones

- Gradient of qr
- Gradient of pr

Diagram showing points Q, P, R, and vectors qr and pr.
Quality and size optimality

Temporal Aspect Ratio of tetrahedron := \( \frac{\text{Temporal Height}}{\text{Temporal Duration}} \geq \frac{\varepsilon}{2} \)

Number of tetrahedra in our mesh is \( \tilde{O}(\frac{1}{\varepsilon^2}) \) times that of an optimal causal mesh.
Mesh Adaptivity

Assume a fixed scalar wavespeed (for now)
Newest Vertex Refinement

[Sewell ’72, Mitchell ’88]

Bisecting triangles on the current front decreases size of future spacetime elements

Coarsening $\equiv$ De-refinement

Adapt progress constraint to changing front geometry
New Progress Constraint

Each inequality limits gradient along one of four directions. Gradient vector is excluded from eight forbidden zones.
Nonlocal Cone Constraints
Assume a fixed triangulation (for now)
Nonlocal cone constraints

Different wavespeeds due to:
1. different material properties;
2. nonlinear PDEs, e.g., $u_t + uu_x = 0$.

Can’t visualize as circle diagram anymore!

Increasing wavespeed prevents positive progress.

Wavespeed can increase *unpredictably* due to *shocks*, i.e., discontinuities in the solution.
Anticipating Faster Wavespeeds

Most limiting cone constraint is nonlocal
Adapting to Changing Wavespeeds

Conundrum:
1. Given causal front $\tau_i$, the next front $\tau_{i+1}$ is causal only if $\tau_{i+1}$ lies strictly below the cone of influence of every point of $\tau_i$.
2. In other words, $\tau_{i+1}$ must lie below the lower hull of cones of influence of points of $\tau_i$.
3. Additionally, $\tau_{i+1}$ is guaranteed to make progress in the next step ($\tau_{i+1} \rightarrow \tau_{i+2}$) if it satisfies a progress constraint that depends on the wavespeed of $\tau_{i+2}$.
4. But $\tau_{i+2}$ not determined until $\tau_{i+1}$ is computed!

Answer: Always look ahead (adaptively) to estimate future wavespeed and enforce future progress constraint now.
Asymmetry/Anisotropy

- Waves propagate asymmetrically through inhomogeneous media.
- Approximating domains of influence and dependence by circular cones not good enough.
  - inclined cones of influence
  - asymmetric cones with non-circular cross-sections
Maximizing Progress

- New front is causal if and only if it is *below the cone of influence* of every point on the current front.
- Use **bounding cone hierarchy** to formulate a discrete problem and to efficiently measure future wavespeed.
- Take advantage of the fact that a cone is a ruled surface.
- Maximizing tentpole height queries the **lower hull of cones** (ray shooting in 1D×Time).
- Is progress constraint satisfied? Query the **cone hierarchy** to look ahead several steps.
- Alternatively perform a binary search; raises numerical issues.
Bounding Cone Hierarchy

- If we can measure wavespeed at any future point efficiently, we can solve optimization efficiently.
- Use a tree of bounding cones to hierarchically represent successively better approximations to future wavespeeds.
- **Bounding cone hierarchy** similar to data structures used in collision detection and tree codes for $N$-body problems.
- Bounding cone hierarchy is slightly more complicated for asymmetric cones.
Adaptivity + Nonlocal Cone Constraints = Unified Algorithm
Smaller is better

- **Key intuition:** smaller triangles (more refined front) approximate lower hull of cones better than larger triangles.

- **Key useful property:** A cone is a *ruled surface*.

- Smaller triangles constrained by a subset of the cone constraints than their parents.

- **Solution:** Adapt progress constraint to degree of refinement.

- **Expected result:** Smaller elements have “better” geometric shape, hence better “quality”.
Current and future work

- Generalize newest vertex refinement beyond bisection
- Track curved spacetime features, such as shocks and other singular surfaces, by aligning mesh facets
- Track moving internal and external boundaries and material interfaces, by aligning mesh facets
- Perform operations in addition to tent pitching: spacetime equivalents of edge flips, edge contraction, and edge dilation.
- Adaptivity in higher dimensions: 3D×Time.
- As before, main challenge is proving theoretical guarantees.
Adaptively\textsuperscript{(1)} construct near size-optimal\textsuperscript{(2)} spacetime meshes in $2D \times \text{Time}$ with provable temporal aspect ratios\textsuperscript{(3)} for fast\textsuperscript{(4)}, accurate, parallel\textsuperscript{(5)} numerical solution of nonlinear\textsuperscript{(6)}, anisotropic\textsuperscript{(7)} hyperbolic PDEs.
Dank U Wel!
Non-constant wavespeed means strictest cone constraint can be arbitrarily far away.

Expensive to examine all cone constraints.
Bounding cone hierarchy

- Adapt standard technique used in computational geometry.
- Solve optimization problem: maximize height of tentpole subject to all cone constraints.
- Group constraints into a hierarchy.
- As expected, only a few constraints in the hierarchy need to be examined on average.
Hierarchical subdivision of domain
Building the hierarchy

{x, y}

{7,8}
Building the hierarchy (2)
Building the hierarchy (4)
Traversing the hierarchy
Traversing the hierarchy (2)
Traversing the hierarchy (3)
Tightest constraint is a leaf in the hierarchy.