**Paper#1 TITLE: AnisoMPM: Animating Anisotropic Damage Mechanics (2020)**

**What was the main contribution of the paper?**

The paper introduces AnisoMPM, an augmentation of the Material Point Method (MPM) that can handle the dynamic fracture of anisotropic materials. Its key novelty lies in using a non‐local continuum damage mechanics (CDM) approach combined with structural tensors to encode material anisotropy. In addition, the authors propose integrating damage evolution with anisotropic elastic responses—implemented using a QR-decomposition based constitutive model—and introduce a Galerkin weak form discretization for embedded directional inextensibility to address problems with extremely stiff fibered materials. This combination affords both robustness and computational efficiency as well as a high degree of artistic control for visual effects applications.

**List two things you liked or found interesting about the paper:**

1. The adoption of structural tensors to naturally encode anisotropy in the damage evolution is both elegant and powerful. It generalizes the approach so that it covers isotropy, transverse isotropy, and orthotropy in a unified framework.
2. **Coupling of Damage and Elasticity**

The paper’s approach to decomposing the hyperelastic response into tensile and compressive contributions—and then selectively degrading the tensile component based on damage—provides a clever mechanism to generate realistic fracture patterns. The additional treatment for stiff fibers (through the weak form discretization) further enhances its practical applicability.

**What did you not like about the paper?**

While the method is robust and flexible, its efficiency when scaling up to very large or highly detailed simulations is not entirely clear. There might be challenges when the simulation involves extremely complex geometries or when many fracture events occur concurrently.

**What questions do you have for the author about their research?**

1. Have you explored or do you plan to extend AnisoMPM to handle multi-physics scenarios, such as thermo-mechanical coupling, where additional physical effects could interplay with fracture dynamics?
2. Could you provide more details on how the method scales with increasing model complexity? In practical VFX applications, are there strategies to mitigate potential computational bottlenecks?

**Paper#2 TITLE:** **Shortest Path to Boundary for Self-Intersecting Meshes**

**What was the main contribution of the paper?**

This paper presents a robust algorithm for computing the exact shortest path from an interior point of a self-intersecting mesh to its boundary. The authors develop a formal definition of “shortest path” in the presence of self-intersections and propose an algorithm that leverages an efficient tetrahedral (or triangular in 2D) ray traversal to verify candidate paths. The method does not require costly pre-computations or volumetric storage and is well-suited to applications in collision and self-collision handling for deformable simulations.

**List two things you liked or found interesting about the paper**

1. The paper provides a clear set of definitions and formalizations regarding what constitutes the shortest boundary path even when a point is simultaneously on the boundary due to self-intersection. This theoretical clarity is important in resolving ambiguities inherent in self-intersecting models.
2. The authors’ approach to robust tetrahedral (or more generally, simplex-based) ray traversal is both innovative and practical. The method efficiently avoids issues like infinite loops caused by numerical inaccuracies near edges or vertices, which is critical for reliable simulation results.

**What did you not like about the paper?**

Although the method is robust, its performance and accuracy might be sensitive to the quality of the input mesh. Highly irregular or noisy meshes could potentially degrade performance or lead to ambiguities that the algorithm might struggle to resolve.

**What questions do you have for the author about their research?**

1. Have you evaluated how your algorithm performs on highly irregular or noisy meshes? Are there pre-processing steps you recommend to improve mesh quality before applying your method?
2. In dynamic simulation scenarios, can your method be integrated into real-time collision resolution frameworks, and if so, what are the observed performance benchmarks compared to existing approaches?

**Paper#3 Computational Design of High-level Interlocking Puzzles**

**What was the main contribution of the paper?**

This work introduces a computational framework for designing high-level interlocking puzzles—puzzles that require multiple moves to disassemble the initial configuration. The approach is built around a graph-based disassembly planner that encodes all possible configurations and transitions, allowing the calculation of an intrinsic “level of difficulty” defined by the minimum number of moves needed to remove the first subassembly. Additionally, the framework guides the iterative construction of puzzle piece geometry and incorporates a shape optimization step to ensure the pieces are both visually pleasing and structurally sound.

**List two things you liked or found interesting about the paper:**

1. The novel idea of representing puzzle configurations and disassembly transitions in a rooted graph structure enables an exact calculation of puzzle difficulty. This formalization provides designers with direct control over the complexity of the puzzle, which is both innovative and practically valuable.
2. The framework’s ability to iteratively construct the geometry of each puzzle piece—coupled with post-processing shape optimization for a smooth finish—highlights a thoughtful balance between computational design and aesthetic considerations. This ensures that puzzles are not only challenging but also appealing from a design perspective.

**What did you not like about the paper?**

The approach is based on voxelized shapes and assumes that disassembly motions are limited to translations along major axes. While this makes the problem tractable, it might limit the framework’s ability to design puzzles with more diverse or complex movement patterns.

**What questions do you have for the author about their research?**

1. Do you foresee extensions of your framework that can accommodate rotational movements or more complex disassembly motions beyond simple translations?
2. How does the algorithm scale as the puzzle’s complexity increases in terms of both the number of pieces and the level of difficulty? Are there potential optimizations that could improve performance for very large puzzles?