Driving Object Deformations from Internal Physical Processes

Zeki Melek et al.
2007 ACM symposium on Solid and Physical modeling

Presented by Tae-hyeong Kim
2010 / 12 / 08
Korea University
Computer Graphics Lab.
1 min. Intro

Proxy object
Abstract

• We present a method for
  ▪ **deforming objects for graphics applications**, based on the results of **internal physical simulations**.
  ▪ describe in detail methods for simulating the **bending of burning matches**, and the **crumpling of burning paper**.
    • In these cases, the small-scale changes in a chemical process result in large-scale deformations of the given object.
  ▪ We propose the use of a **free form deformation** to model such large scale deformations.

• Changing object properties are mapped onto the edges of a **proxy object**, which is then modified by treating the edges as springs.
  ▪ This proxy object then serves as a control structure for defining the deformation of the underlying object.

• The results are fast, controllable, and visually plausible.
1 Introduction

• Complex physical simulations are very challenging.
  ▪ Many complex physical simulations require modeling of a number of different phenomena.
  ▪ An “ideal” simulation might accurately simulate all details of all physical phenomena, this is usually impractical.
• Instead, we use a simpler simulation, and eliminate certain secondary effects in order to achieve a plausible result in reasonable time.
• The goal of our work is
  ▪ to develop a method for efficiently modeling certain secondary effects in physical simulations,
  ▪ thereby increasing visual plausibility of the overall simulation for only a reasonable cost in efficiency.
• We do this by
  ▪ attempting to model the large-scale effects of certain physical processes.
• We propose a way of approximating larger-scale deformations of objects guided
  ▪ by the results of a simulated physical process.
• An example of such complex physical processes is burning objects.
• To simulate a burning object,
  • the combustion reaction, heat distribution, fuel consumption, and even object shape must be modeled and changed over time.
  • The pyrolysis process, where an object releases combustible gases, causes decomposition and additional structural changes in burning objects.
• Although these structural effects are usually minor, some create a dominant deformation on burning objects.
• We can simulate
  ▪ the combustion reaction, the object catching fire and burning, and even the decomposition of the object as it burns,
  ▪ but **unless we model the fiber contraction**, the simulated matches **will not behave like actual burning matches.**
• We present a free form deformation (FFD) based method for approximating large-scale deformations due to smaller-scale physical simulations.
1 Introduction

Major Contributions

• We present a framework for creating deformations guided by physical simulations.
  ▪ Defining a proxy object. The deformations are simulated on the proxy instead of the actual object.
  ▪ Mapping simulation parameters to the proxy object.
  ▪ Modifying the proxy object based on the simulation results.
  ▪ Creating a deformation using the modified proxy object.
1 Introduction

Major Contributions

• We present the first models that give a physics-driven simulation of two physical phenomena.
  ▪ Bending burning matches.
  ▪ Crumpling burning paper.
2 Background - FFD
3 Simulation Guided FFD

- We propose a **simplified model** to mimic similar behavior for secondary deformations
  - Rather than modeling the actual chemical process fully
- We assume
  - there is some physically based simulation determining the “major” processes acting on the object,
  - and the effects of any “secondary” processes are either **too complex** to model, or **too time-consuming** to simulate.
- Therefore, we use a simplified deformation,
  - guided by the primary simulation,
  - to model the deformations created by these secondary effects, rather than modeling the effects themselves.
The overview of our proposed method is as follows (fig. 3):

- Run the simulation for the primary physical processes affecting the initial object.
- Place a proxy object around the deforming object of interest.
- Map the simulation results from the initial object onto the proxy object.
- Determine an approximated deformation for the proxy object using the mapped properties.
- Use the deformed proxy object to control a deformation applied to the original object.

Figure 3: The proposed simulation driven deformation process. (a) initial object, (b) placing proxy, (c) subdivide along the deformation axis, (d) mapping parameters onto the proxy, (e) deformation defined on the proxy, (f,g) apply the deformation onto the initial object.
3.1 Placing a Proxy Object

- The basic idea is to define a simplified object
  - that can later be used to deform the object of interest.
  - We refer to this simplified shape as a “proxy object”.
- Given a particular object or region of interest, there are many ways of defining a proxy object.
  - Ex) A bounding box, the convex hull of the object, a simplified skeleton of the object, or a user-specified simple shape.
• We will consider the proxy object to be a polygonal boundary representation that **encapsulates the object** of interest.
  - However, it is certainly possible to use other proxy objects.
  - For example, a tricubic spline function, a medial or skeleton-based representation...
• Obviously, in most cases the proxy object should be significantly simpler than the original object.
• We will make an assumption that we have a single proxy object of **genus zero**.
  - In reality, even if we start with a genus zero object, solids might undergo topological changes during complex physical simulations.
  - We assume that a genus zero proxy will be sufficient to represent any deformations.
  - If the proxy would need to change in genus, another method might be more appropriate.

*genus zero* is equal to the number of "holes" it has.
3.2 Mapping Simulation Result

• Given a proxy object and a simulation on the underlying object, the second step involves mapping the “interesting” simulation properties from the original object onto the proxy.

• To simulate, we are mapping properties that can be used to later define the deformation.

• In burning objects, we map internal heat distribution and rate of pyrolysis onto the proxy faces.
  ▪ For a different simulation, we might map another parameter, e.g. air pressure, onto a surface.
• Note that this mapping onto the proxy is not necessarily one-to-one;
  ▪ several points of data from the underlying simulation can map to the same point on the proxy object,
  ▪ or none of the underlying points on the object might map to a particular point on the proxy.
  ▪ Also parameters could map to different parts of the proxy object.
• For example, for an encapsulating polygonal proxy, we could conceivably map parameters to the faces, or to the edges, or to the vertices.
3.3 Defining Deformation of the Proxy

- The critical part of our proposed method is defining the approximated deformation on the simplified proxy object.
- The main point is that
  - we take the parameters that have been mapped onto the proxy object,
  - and use these to define a deformation of the proxy object itself.
A few examples of the ways that the mapped properties (the “values”) might define a deformation on the proxy object include:

- The values can be used as constants for a spring system along edges of the proxy.
  - The proxy object can then be simulated to equilibrium.
- The values can be used to define a local transformation (e.g. scaling, translation) of the proxy mesh.
  - The deformed proxy is formed from the superposition of these local transformations.
Generally, a “good” deformation will have these properties:

- **Similarity**: The proxy object should “behave” in the same way (though at a coarser level) as you would want the underlying object to behave in response to those simulation parameters.

- **Simplicity**: It will be significantly cheaper to compute than the simulation on the original object would have been.

- **Control**: The user should be able to control the way the deformation behaves.
3.4 Applying the Deformation

- Finally, we **apply the deformation** defined on the proxy to the encapsulated object.
  - The proxy will define the deformation that will be used to warp the local coordinate system of the actual object.
• For basic encapsulating polygon proxies, the simplest approach is to use the proxy object as a free form deformation (FFD) lattice around the deforming object.
  - We can decompose the cells into tetrahedra and apply piecewise linear interpolation using barycentric coordinates inside the tetrahedra.
  - This approach can be made arbitrarily more complex.
  - If more continuity is desired, a nonlinear interpolation method could be used.
  - Or, one could easily use mean value coordinates or similar interpolation schemes to define a deformation from a given polygonal proxy.
• Other types of proxies would afford different deformations.
  
  ▪ For example, tricubic spline proxies could directly define an FFD.
  ▪ Skeletal or medial proxies could be used as the basis for a distance-field based deformation of space.
3.5 Evaluation

- This procedure allows us to define a deformation behavior similar to the desired physical process.
  - Note again we are not simulating the actual physical process,
  - but rather we use a computationally cheap approximation of the same phenomenon to give visually “plausible” results.
- The key issue of our proposed method is that
  - by using a set of related object and simulation properties
  - we are able to approximate our desired behavior as a simple deformation.
4 Bending Matches

Figure 4: Before and after simulation driven bend deformation.
Matches are made out of fibrous material oriented along the length of the match.

- During burning, the fibers lose water and other chemicals.
- Due to the shape of a flame, the **upper part of a match receives more of the heat** generated from the combustion reaction that forms the flame.
- Thus, it is hotter on the upper side of the match compared to the bottom;
- This means that the fibers on the top contract more than the ones at the bottom.
- This imbalance accumulated at the microscopic level forces the match to bend upwards at the macroscopic level.
• We begin by selecting a proxy object for the match.
• Due to the shape of the match, we use a simple bounding box aligned along the match axis as a proxy.
• This bounding box is subdivided into a number of individual segments along the bending axis (the length of the match), creating a 1x1xN FFD lattice surrounding the match.
• During the simulation of burning and decomposition, the rate of pyrolysis \( \frac{d}{dt}P \) and internal heat \( T \) of the object are mapped onto the faces of the proxy object.

• Each face of the proxy will store the average of the pyrolysis rate and heat values that map to it.
• Before we define the deformation, we need to clarify one issue.

• We can define a simple cylindrical mapping at the start, but what will happen as we bend the match?
  ▪ Since the match will also decompose, the faces will change and we cannot fix the mapping.
  ▪ One thing to note here is that although the match is bending in world space, it is still undeformed in its own local space, and so is the proxy.
  ▪ Hence we can define the mapping from the object to the proxy in the unbent local object space.
\[ D_{ij}^T = \Delta T_i - \Delta T_j \quad (1) \]
\[ D_{ij}^P = \frac{d}{dt} P_i - \frac{d}{dt} P_j \quad (2) \]
\[ D_{ij} = \begin{cases} \alpha D_{ij}^T + \beta D_{ij}^P, & P_i, P_j \geq P_{\text{thresh}} \land T_i, T_j \geq T_{\text{thresh}} \\ 0, & \text{otherwise} \end{cases} \quad (3) \]

Where \( i \) and \( j \) are opposing faces. \( T_i \) and \( P_i \) are the temperature and pyrolysis values mapped to vertex \( i \). \( D_{ij} \) is the rotation amount. \( \alpha \) and \( \beta \) control how much the heat and rate of pyrolysis affect the final deformation respectively; these are the parameters that allow an element of user control to the deformation. \( T_{\text{thresh}} \) and \( P_{\text{thresh}} \) are thresholds for starting the deformation. Note that the axis around which the rotation occurs is perpendicular to both the \( \vec{i}j \) axis and the axis along the match length.
• On a rectangular proxy, we have two sets of opposing faces, hence two sets of deformation values for every cell \((D_{ij}, D_{kl})\) for two axes orthogonal to the match length.
  - We use those deformation values by rotating the rest of the cells centered around the “active” cell.
  - Here, we can either apply rotation to both sides, or fix one side (if the match is anchored or held at one end) and apply the rotation to one side only (fig. 5).

Figure 5: Curling match using simulation guided FFD.
• Finally, we use our deformed proxy to deform the match itself.
  - We subdivide the proxy lattice cells into tetrahedral elements
  - and apply piecewise linear interpolation using the barycentric coordinates inside the tetrahedra to deform the space around the burning match.
• Note in all of this the importance of the choice of proxy object.
• While a different proxy object could be used,
  ▪ it might require significant modification of the approach,
  ▪ particularly for determining how the mapped values are used to define the deformation.
5 Crumpling Paper
6 Integration

• Generally, the proposed method should be fairly easy to integrate into an existing simulation system.

Figure 11: Extending the model to include bending/crumpling.
To incorporate the deformation into this framework, we add a “wrapper” function to the solid simulation.

- The burning solid does not know that it is deformed, except during the synchronization with the flame module
- No special handling or extra code is required
- Compared to the rest of the simulation, the deformation adds only a minor overhead.

The proposed process is simple, easy to integrate to our existing simulation framework.
• As discussed in the earlier section, the deformation framework does not handle self intersections.

• However, the deformed object now exists in a different world position.
  ▪ Thus, any simulation data can be correctly placed into or interpolated from world coordinates;
  ▪ For example, if a match bends into a U shape, the ends of the match will be near each other in world space, and thus heat that is transferred via a world-space (as opposed to local object space) calculation, as in our flame simulator, will easily allow heat transfer from one end of the match to the other.
7 Conclusion

- We present a simple deformation framework
  - to approximate some physical process,
  - thus giving increased realism to a simulation using only a limited computational overhead.

- We propose the use of a free form deformation to model large-scale deformations
  - Instead of modeling a rather complex and detailed process to create such secondary behavior,
• This approach fits well in the multi-representation object modeling paradigm [Melek and Keyser 2005]

• We present results on bending burning matches as well as crumpling of burning paper that demonstrate the first such physically-based graphical simulation of these phenomena.
7.1 Limitations and Drawbacks

- Ignored topological changes.
- Material compression during the bending is ignored.
- Self collisions are not handled.
- Limited use for complex deformations.
- Not simulating the “real” physics.
7.2 Advantages

- A minor computation overhead
- Easily integrated into existing systems
- The user has some control over the desired behavior.
- The method is “based” on simulation data, and thus is driven by a physical process
- Plausible results
7.3 Future Work

- Bending of heated metal objects
  - Define torques
  - Fracturing

- Geared towards information visualization
  - We can define a deformation-based visualization
  - It could help us to visualize and investigate the data and detect features in it more easily