General-Purpose Visualization of Large-Scale Finite Element Analysis Simulations

Voicu Popescu
Purdue University
West-Lafayette, IN
popescu@cs.purdue.edu

Christoph Hoffmann
Purdue University
West-Lafayette, IN
cmh@cs.purdue.edu

ABSTRACT

Finite Element Analysis (FEA) is a versatile numerical simulation method that replicates with great fidelity complex physical interactions. FEA software systems are enhanced with postprocessors that allow simulation experts to examine simulation results visually. However, such postprocessors are typically quite limited when it comes to imparting the results of the simulation beyond the narrow community of numerical simulation experts that devised the simulation. The lack of such general-purpose visualization capability is particularly limiting as the power of compute platforms and the sophistication of simulation codes increases.

We propose to achieve high-quality, general-purpose visualization of FEA simulation data through automatic, scalable, and robust translation into forms suitable for rendering with state-of-the-art animation systems. The translator converts the finite elements into polygonal surface representations, physical material models into material models describing surface/light interactions, and deformation and displacement data into animation data. The method was used to produce visualizations of our Pentagon and World Trade Center simulations, where it proved its effectiveness for conveying the results of the simulations to the general public. The visualizations have been downloaded millions of times, they have been relayed by hundreds of mass media outlets, and they were requested by the September 11 National Memorial and Museum for permanent exhibition.

ABOUT THE AUTHORS

Voicu Popescu is an associate professor of computer science at Purdue University. His research interests span the fields of computer graphics, visualization, and computer vision. Current projects focus on large scale automated 3-D modeling, on high-quality visualization of finite element analysis simulation data, and on effective distance learning. He is a founding member of Purdue's Envision Center, a campus wide facility for data visualization. A flagship project enabled by the center was a large-scale high-fidelity simulation of the September 11 Attacks on the Pentagon and on the World Trade Center, a multidisciplinary effort for which Dr. Popescu led the visualization team. Dr. Popescu received a computer science Ph.D. degree from the University of North Carolina at Chapel Hill in 2001. He is the author of over 50 peer reviewed publications.

Christoph Hoffmann is a professor of computer science at Purdue University, the Co-Director of the Product Lifecycle Management Center, and the Director of the Rosen Center for Advanced Computing. Hoffmann's research is in geometric modeling and geometric constraint solving, as well as their many applications in manufacturing, simulation, and visualization. Hoffmann has been involved with simulations since 2001 when he collaborated in simulating the 9/11 Pentagon Attack. The author of numerous scholarly articles and two monographs, Hoffmann's research is internationally recognized and has been supported continuously by grants since 1978. He serves on the editorial boards of ACM Transactions on Graphics, Computer Aided Design, and Computer Aided Geometric Design.

General-Purpose Visualization of Large-Scale Finite Element Analysis Simulations

Voicu Popescu
Purdue University
West-Lafayette, IN
popescu@cs.purdue.edu

Christoph Hoffmann
Purdue University
West-Lafayette, IN
cmh@cs.purdue.edu

BACKGROUND

Visualization has long been recognized as an essential tool in science and engineering. Sustained research efforts have produced specialized visualization tools that let the domain expert inspect vast quantities of data in detail. However, there is inadequate support for presenting science and engineering data in a visual form accessible to users whose expertise is outside of the domain in which the data was generated. Such users include decision makers, trainees, students, or citizen scientists and the public. The lack of adequate support for such *general purpose visualization* is a problem of great and growing importance:

- The power of simulation codes and of computing hardware, and the resolution, range, diversity, and number of sensors increase at a rapid pace, generating data and affording insights of great interest beyond a single laboratory, research group, or domain.

- Teams in science and engineering research are becoming large and diverse, and general purpose visualization can serve as a common language that supports effective collaboration in such multidisciplinary teams.

- Fueled by the enormously popular applications in entertainment, computer graphics algorithms and hardware have progressed at a rapid pace. Graphics processing units (GPUs) programmable through complex shading languages, and animation systems that implement the latest graphics algorithms, can produce renderings of stunning quality. Although computer graphics hardware and software tools could be used to satisfy the science and engineering needs for general purpose visualization, these tools are too low-level and too complex to be readily accessible to researchers in science and engineering, practitioners, and educators. The reality is that the gap between *what is* and *what could be* produced in visualizations of science and engineering continues to widen.

- Last but not least, general purpose visualization is a fundamental tool in science and engineering education. By providing accessible illustrations of abstract concepts and by shedding light on the connection between applications and the underlying fundamental principles, general purpose visualization can help reverse the alarming trend of students losing interest and proficiency in science and engineering.

The options presently available to the science and engineering research community for satisfying the need for general purpose visualization have severe shortcomings. One option is to use the general purpose visualization capabilities built into the visualization modules of domain specific software. However, such capabilities are very limited and are typically one or several steps behind the state of the art. For example, post-processors attached to finite element analysis (FEA) simulation codes are weak when it comes to conveying the simulation results beyond the narrow circle of researchers that devised the simulation. It will never be—nor should it be—the case that software used in science and engineering will closely replicate the state-of-the-art in computer graphics.

A second option is for science and engineering research teams to partner with visualization and graphics researchers, developers and/or digital artists who implement the general purpose visualization task using graphics programming toolkits. Projects fortunate enough to have this option available stand out through effective dissemination of their results and through widespread adoption of the tools they produce, all resulting from quality general purpose visualization.

However, the option is inefficient: the collaboration with visualization and graphics researchers is typically wasted on *re-implementing* state-of-the-art general purpose visualization functionality instead of on *advancing* the state-of-the-art. Based on textual and on coarse visual descriptions, digital artists painstakingly create the digital content necessary for quality general purpose visualization. The process is time consuming, introduces unwanted subjectivity, and needs to be repeated for every experiment or scenario investigated.

APPROACH

We became acutely aware of the importance and the lack of support for high-quality general purpose visualization during our project that developed high-fidelity large-scale FEA simulations of the September 11 terrorist attacks on the Pentagon and on the World Trade Center (WTC). Although the initial motivation for the simulations was to understand the buildings' behavior under the impact, it quickly became apparent that the interest in the simulations went beyond civil engineering and that a high-fidelity general purpose visualization of the simulation results would be needed.

Since conventional post-processing tools could not achieve adequate visual sophistication, we pioneered the approach of *translating* the FEA simulation output data into a 3D scene that could be loaded into a state-of-the-art animation system. The animation system enabled producing a high-quality visualization of the simulation results integrated within the surrounding scene. The visualization was shown by media outlets world wide. Our Pentagon attack visualization is Purdue University's all-time most downloaded file. Our WTC attack visualization could not be served by Purdue's websites and therefore was moved to YouTube (2008) where several copies of it were downloaded over 4.1 million times altogether. Also, the WTC visualization was requested by the September 11 National Memorial and Museum in New York for permanent exhibition.

Our approach of data translation does not provide general purpose visualization services itself, but rather provides convenient access to such services already implemented by readily-available, commercial graphics infrastructure. The approach effectively unlocks state-of-the-art general purpose visualization for FEA simulation, without the disadvantages of prior methods discussed above:

- The translation is automatic resulting in low cost and great efficiency; making possible extensive visualization of many scenarios, and achieving quality general purpose visualization that is affordable for virtually any project.

- Moreover the visualization is constructed objectively from the actual output data bypassing the subjectivity of manual replication. What the user sees are the actual results of the scientific or engineering experiment.
- Rendering algorithms and graphics hardware state-of-the-art is likely to continue to advance at a rapid pace. Our approach takes advantage of such advances often "for free" or occasionally at the cost of small enhancements to the translator.

- Collaborations with visualization researchers are able to focus from day one on devising novel visualization techniques, bypassing wasteful re-implementations of prior art.

The remainder of this paper is organized as follows. Prior work is reviewed next. Section 4 lists the technical challenges and solutions for connecting the simulation and visualization software. Note that these systems have different ways of conceptualizing geometry and interaction, with different priorities and with different error metrics. Section 5 presents and discusses results. Section 6 concludes the paper and gives directions for future work.

PRIOR WORK

As mentioned in the previous section, our group pioneered the approach of general-purpose-visualization of FEA simulation data with state-of-the-art commercial animation systems, so there is no directly related prior work upon which our efforts build in this regard.

Baker (1998) led a project whose goal was to simulate and visualize the impact of a bomb detonation on a nearby multi-story concrete building. Although the report does not explicitly mention the 1996 attack on the Khobar towers, the scenario investigated matches that incident. The simulation had two distinct parts: the blast was simulated first, and the computed pressure loadings were then used to compute an FEA simulation of the structural response of the building. The resulting data was rendered using the Visualization Tool Kit (VTK, 2008), a software system frequently used in scientific visualization. The effective communication of the simulation results was identified by the team of researchers as an area to be targeted by future efforts.

There is significant interest in determining the response of buildings and concrete structures to aircraft impact. Such work, applied to reactor buildings, is used to develop building codes to prevent that nuclear material would escape. Sugano (1993) conducted a full-scale experiment with an actual fighter aircraft. The

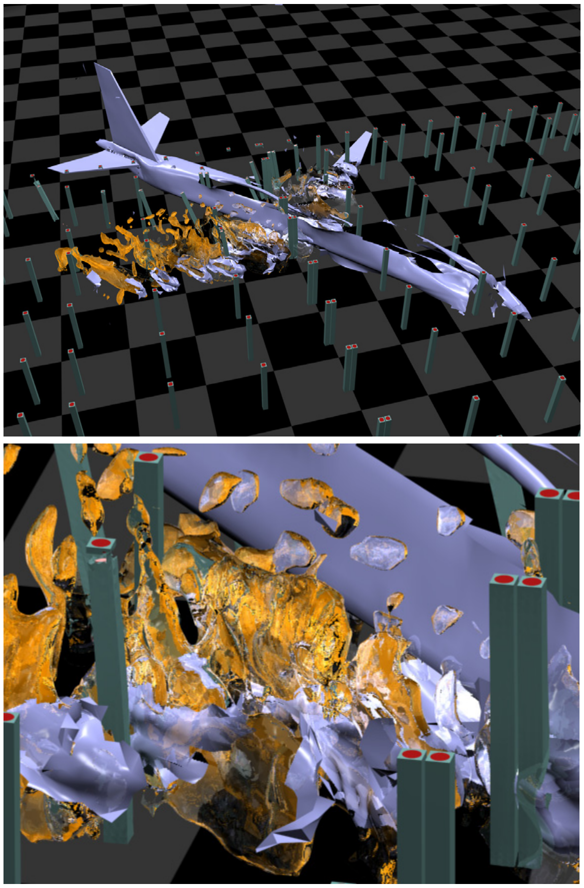

Figure 1. Snapshots from Pentagon Visualization.

experiment provided impact force/deflection measurements used to validate subsequent simulations.

FEA TO VISUALIZATION TRANSLATION

We developed the program, converting FEA simulation data into a 3-D scene that can be rendered with a commercial animation system, in two phases. First, we developed the translator for the kind of FEA data that arose in our Pentagon simulation (Popescu et al., 2003, Hoffmann and Popescu, 2004, Popescu and Hoffmann, 2005). We then extended the translator to the additional data computed in the WTC simulation (Irfanoglu and Hoffmann, 2007, Rosen et al., 2008). Both simulations used LS-DYNA (Hallquist and Benson, 1987), a state of the art commercial FEA simulation system. Both visualizations were computed with 3ds Max, a state-of-the-art commercial animation system (3ds Max, 2008). Although the translator was implemented as an import plugin that loads LS-DYNA data into 3ds Max, the translator is general and can be retargeted to other simulation/animation system pairs.

Developing the FEA to general-purpose-visualization translator implied overcoming several technical challenges:

- efficient and effective solid material visualization,
- liquid visualization,
- scalable animation of deforming entities,
- automatic dust and debris visualization,
- visualization in the context of surrounding scene.

Next we briefly describe how each of these challenges was overcome. The resulting translator is a powerful link between LS-DYNA and 3ds Max which can be reused for many other simulations. The translator is publicly available at our website (Rosen and Popescu, 2008).

Efficient and Effective Solid Material Visualization

FEA models solids with beam elements (segments defined by 2 nodes and a normal), shell elements (quadrilateral patches defined by 4 nodes), thick shell elements (hexahedra defined by 4 nodes and a thickness), and hexahedral elements (hexahedra defined by 8 nodes).

Shell elements have the most natural conversion to a surface boundary representation suitable for rendering—each element is converted into two triangles readily accepted by graphics algorithms and their hardware implementation.

Beam elements pose the challenge of creating a triangle mesh that visualizes the actual profile of the element with high-fidelity. Consider for example a beam element with an “I” profile: Simply connecting the two nodes defining the endpoints of the element leads to a crude visualization. Instead one has to create the “I” profile using 12 vertices at each end of the beam. A second challenge is to ensure continuity between neighboring connected elements. This is achieved by snapping together the elements by sharing vertices on a plane oriented along the angle bisector of the two beam elements.

Hexahedral elements are expensive for visualization because 12 triangles have to be rendered. Although the surface of each element (e.g. a cube) is modeled with

12 triangles, the internal faces of adjacent hexahedral elements of an opaque 3-D object are irrelevant from a visualization standpoint and, in the visualization system, require unnecessary computation. We reduce the geometric load by detecting and removing such internal faces. The simplification algorithm runs in time linear with the number of elements, using a hash table.

Once elements are converted to triangle meshes, materials with appropriate visual properties are assigned, leveraging extensive libraries of graphics materials and sophisticated graphics material editors. In Figure 1 the fuselage and columns are modeled with shell and hexahedral elements, respectively. In Figure 2 the façade and structural steel columns are modeled with beam elements with various profiles. Visualization time is 27 times slower than real time.

Liquid Visualization

Liquid materials have an important role in many simulations, including in our case where the jet fuel

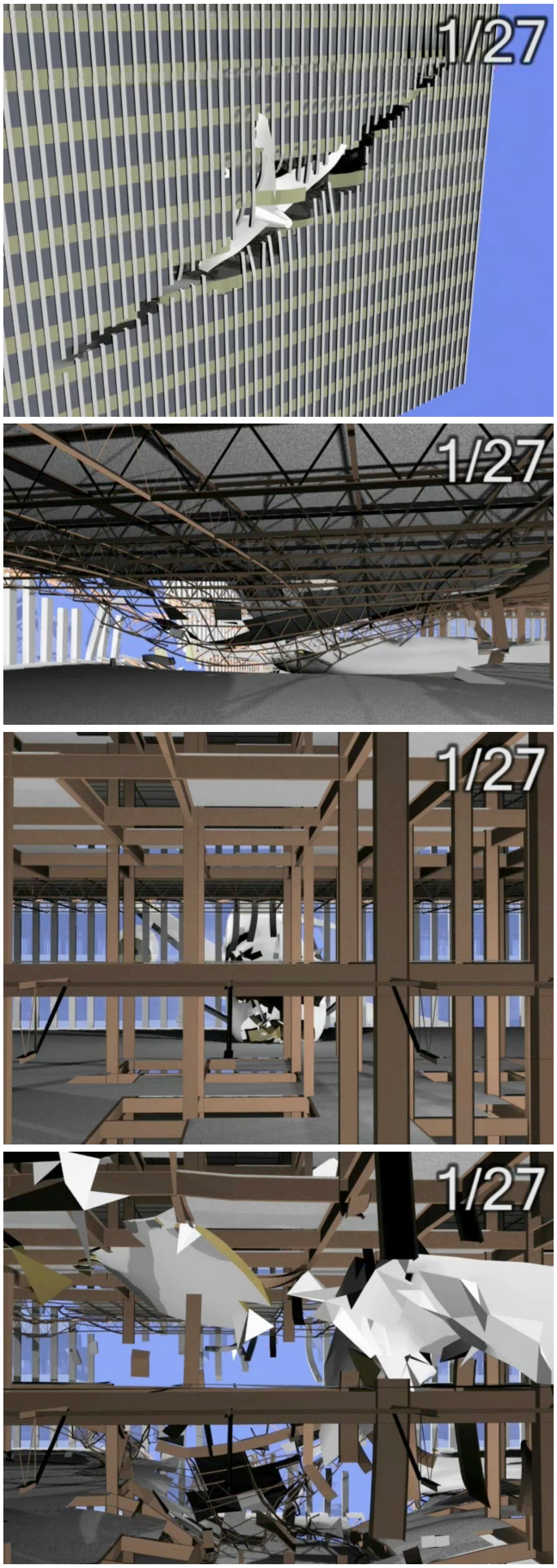

Figure 2. Snapshots from the WTC Visualization.

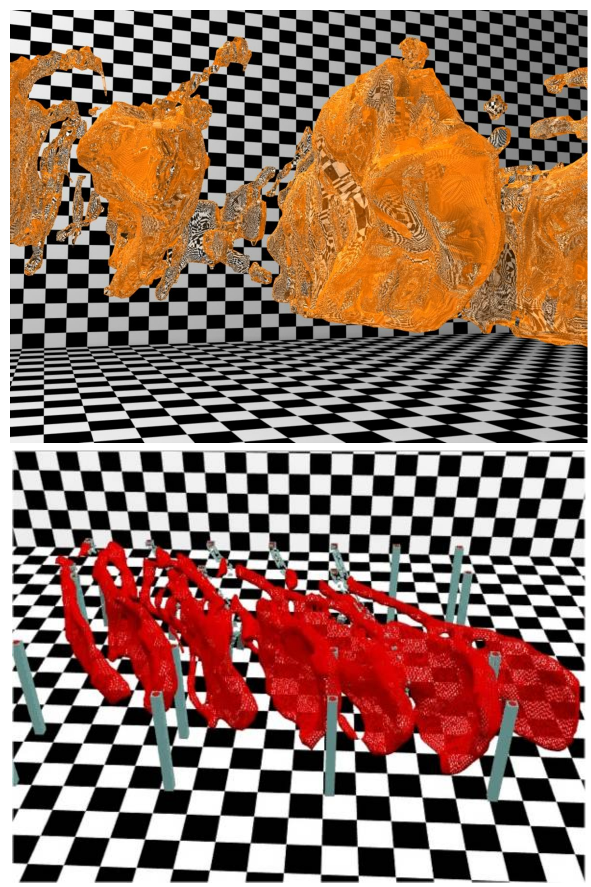

Figure 3. Visualization of ALE Simulated Jet Fuel.

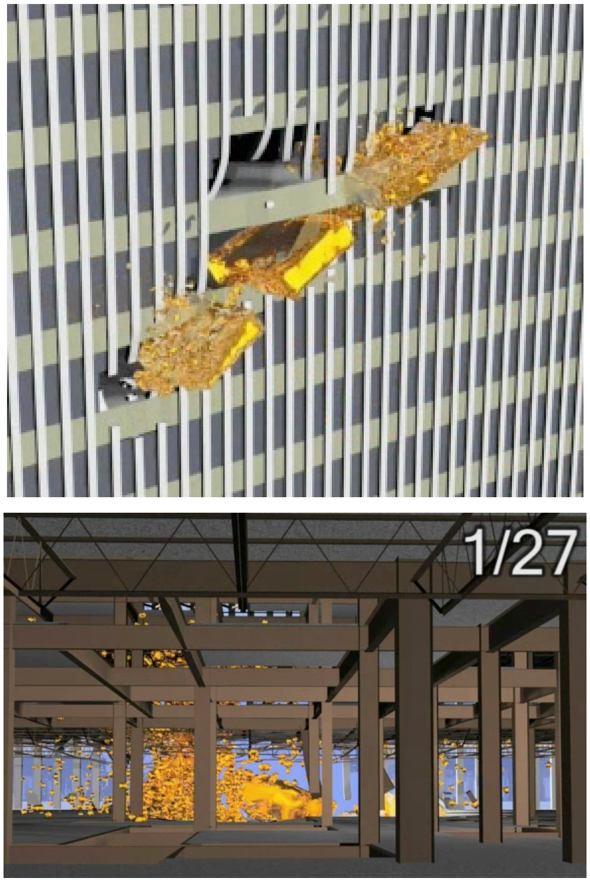

Figure 4. Visualization of SPH Simulated Jet Fuel.

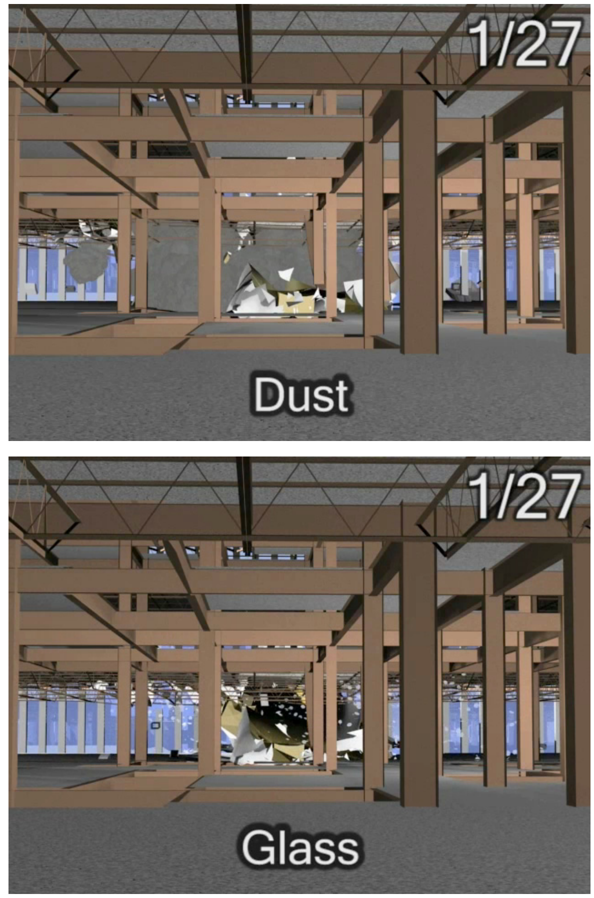

Figure 5. Dust and Glass Debris Visualization.

concentrated a large fraction of the kinetic energy of the aircrafts. The Pentagon simulation modeled the liquid with an arbitrary Lagrange-Eulerian (ALE) mesh, whereas the WTC simulation used a Smoothed Particle Hydrodynamics (SPH) approach.

Each cell of the ALE mesh records the fractional liquid occupancy with a scalar. The first step towards visualization is computing the isosurface for the user-selected fractional occupancy threshold. The isosurface is then assigned a liquid material with the appropriate transparency, color, refraction and reflection indices (Figure 1 and Figure 3, *top*). In the bottom image in Figure 3 the isosurface was rendered as red wireframe.

In the SPH approach the liquid is approximated by a large number of (spherical) particles which are visualized by first merging regions with a large concentration of particles into extended volumes of liquid and then by applying appropriate graphics materials (Figure 4). The particle merging was achieved leveraging geometry processing tools of the animation system.

Scalable Animation of Deforming Entities

A goal of an FEA simulation is to compute displacements and deformations of interacting entities. The simulation output data records, with dense intermediate time steps, the position of each node of the finite elements. This amounts to a large number of positions proportional to the product of the number of states with the number of nodes. This number is further multiplied by the fact that a node corresponds to several triangle mesh vertices used to render the visualization.

Animation systems provide mechanisms for animating characters and objects in a 3-D scene, by using position controllers. A position controller allows the animator to conveniently specify (x, y, z) coordinates for a given point along the animation timeline. The system automatically moves the animated object between

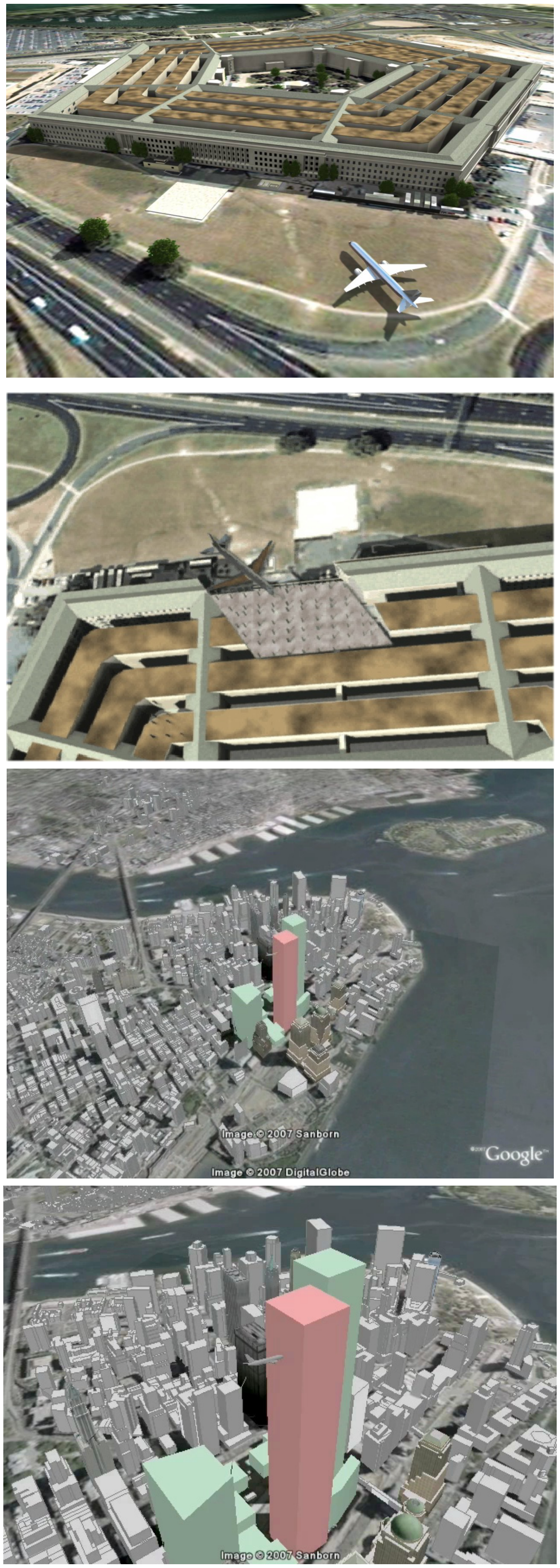

Figure 6. Simulations Visualized in Context.

position controllers offering several options for interpolation. But animation systems do not anticipate a large number of position controllers. Position controllers are typically allocated to an entire object for a few representative key frames along the time line, and not for all vertices in the scene nor for each time stamp. The simulation however computes the position of each vertex for every intermediate state. This fundamental difference between the animation needs of simulation and general-purpose visualization is the most challenging performance bottleneck for the translator.

The first step is to reduce the number of node positions computed by the simulation. The goal is to eliminate positions without affecting the simulation result. Positions for stationary nodes and positions for intermediate states for nodes moving with a constant velocity vector can be safely discarded. However, this does not eliminate enough position controllers.

In the Pentagon work, we resorted to lossy trajectory simplification to reduce the animation load: a node trajectory that was nearly on a straight line was approximated by an exact straight line trajectory defined by its two endpoints. In order to avoid excessive deviation from the deformation computed by the simulation, the threshold used to decide whether a trajectory is sufficiently straight to be simplified was chosen conservatively. This, in turn, limited the effectiveness of the simplification.

For the WTC work we implemented an out-of-core translation of the animation data which required loading only two states at a time. Then, a script controlled loading consecutive states and computing the visualization, based on the previous and current simulation states. This approach eliminates the animation translation bottleneck.

Automatic Dust and Debris Visualization

Another discrepancy between simulation and visualization priorities consists of the different importance of small particles resulting from the destruction of the entities under the impact. Whereas dust and small debris has little kinetic energy and can be safely ignored by the simulation, such small particles have great visual impact and have to be considered by any high-fidelity visualization. For example, dust greatly lowers visibility and has to be rendered when using the simulation for training emergency responders. In another example, high velocity flying debris can severely injure a building

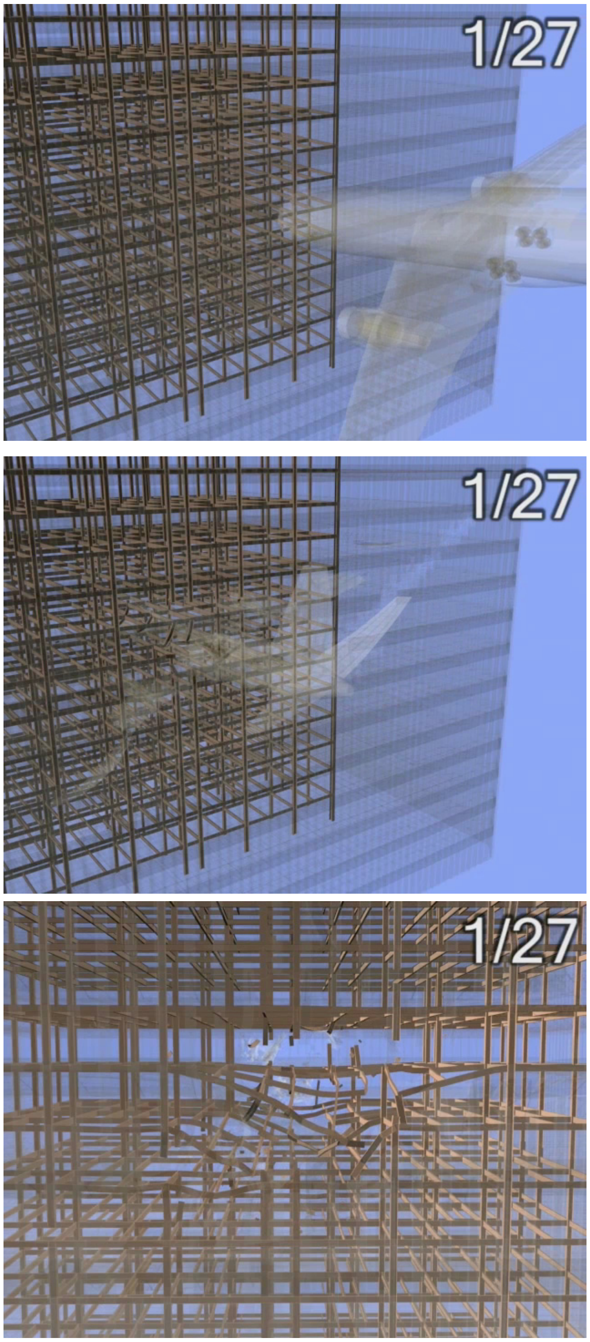

Figure 7. Visualization of Core Column Damage.

occupant or a rescuer, and therefore should not be ignored.

When a finite element undergoes stress that cannot be sustained by the material, an FEA simulation marks it as “eroded” and effectively eliminates it from subsequent computations. The translator used these eroding elements to seed debris and dust visualization effects in the animation system, so eliminating the need of laborious manual seeding. Thus, the dust and debris visualization is driven by the simulation and occurs at plausible locations and with a plausible overall visual impact (Figure 5).

Visualization in Context of Surrounding Scene

FEA simulations justifiably allocate the available computational resources to the entities most relevant to the phenomena they study. However, the simulation results are better understood if they are visualized in the context of the surrounding scene. This is particularly true when the visualization has the role of documenting an actual event, involving a highly-recognizable scene, as was the case for our September 11 simulations.

For the Pentagon simulation, we modeled the surroundings in 3ds Max using orthographic and projective texturing from satellite and aerial images. For the WTC simulation we used a 3-D map of lower Manhattan available through Google Earth (2008). The simulation is registered (i.e. aligned) to the surrounding scene based on precise correspondences. The resulting visualization clearly conveys the trajectory and velocity of the planes before the impact, the actual location in the buildings where the impact occurred, and the magnitude of the damage to the buildings.

RESULTS

The images included in this paper, as well as the actual visualizations, are available at our website (Popescu, 2008). They show the quality of the Pentagon and WTC visualizations and make the simulation results understandable to the general public.

The visualization was released to the press on June 12th, 2007. By 5PM, the video hosted by Purdue’s Computer Science web server was receiving 1,500 hits per hour. During the 8-9AM hour the next morning the video had over 7,900 hits, which forced us to remove the video file from our departmental servers and to post an aggressively compressed version on YouTube. By July 3rd 2007 the YouTube video had been seen over 1.3 million times, and the download count is now, in June 2008, at over 4.1 million. Our simulation was covered by over 200 media outlets from all over the world, including printed and online text publications, and radio and television programs. We are particularly honoured by the request from the National September 11 Memorial & Museum to permanently exhibit our visualization as part of the main narrative of the future museum. Our Pentagon visualization is Purdue University's all-time most download file.

The 9/11 events have and will continue to polarize the world's attention. We do not have the option of gauging what the public's interest in our simulation would have been in the absence of or with only a lower-quality visualization. It is clear, however, that our simulation stands out in terms of public interest among other 9/11 simulations that did not emphasize visualization.

Of particular interest was the response of the structural columns of the WTC tower to the impact. A visualization focusing on the columns was easily produced in 3ds Max by setting all other entities to a transparent material (Figure 7).

CONCLUSIONS AND FUTURE WORK

Linking animation and FEA simulation systems is a powerful approach for efficient and effective general-purpose visualization, especially when the user is not a domain expert. Our initial goal of achieving visual fidelity without sacrificing physical accuracy has been reached.

The translator now supports most type of LS-DYNA output data. The translator has a modular architecture, relying on internal data structures for performing the actual translation. This implies that additional simulation and animation systems can be supported at the small cost of implementing new FEA database parsers and 3-D scene instantiation modules, reusing the actual translation module.

Another line of future work is to develop a translation that supports interactive visualization. The geometry load is easily manageable by a modern graphics card, so a single state could, in principle, be examined interactively after simplifying the material and shading models. The challenge that remains is handling the massive animation data that specifies the positions of millions of vertices independently for each frame.

The mass media interest in our visualization confirms the potential of our approach as a communication multiplier. In future work we plan to measure the benefits of our approach in the context of visualizations for law enforcement and defense applications. We expect that the accessibility and fidelity of the visual language employed, coupled with the automated nature of the visualization generation process, will prove beneficial in the quick exploration of a large number of what-if-scenarios.

ACKNOWLEDGEMENTS

The following individuals had substantial contributions to the September 11 Simulation and Visualization project: Paul Rosen, Scott Meador, Sami Kilic, Ayhan Irfanoglu, Mete Sozen, Ingo Brachmann, Oscar Ardila-Giraldo, Santiago Pujol, Tom Miller, Joe Farris, and Steve Tally. This work was supported in part by NSF, DOE, the Tellabs Foundation, and by Purdue's Rosen Center for Advanced Computing. Some of the runs were done using Purdue's Nanotechnology Computation Network's Regatta.

REFERENCES

3dsmax, (2008) URL: http://www.discreet.com/products/ 3dsmax

Baker, M. P., Bock, D., Heiland, R., and Stephens, M. (1998) Visualization of Damaged Structures, *U.S. Army Corps of Engineers Waterways Experiment Station*, Technical Report.

Federal Emergency Management Agency, (2002). World Trade Center Building Performance Study: Data Collection, Preliminary Observations, and Recommendations", *FEMA 403*.

GoogleEarth, (2008), URL: http://earth.google.com

Hallquist, J. O. and Benson, D. J., (1987) Dyna3D User's Manual (Nonlinear Dynamic Analysis of Structures in Three Dimensions), *Report #UCID-19592-revision-3, Lawrence Livermore National Laboratory, Livermore, California*, pp. 168.

Hoffmann, C., Popescu, V., Kilic, S., Sozen, M. (2004). The Pentagon on September 11th, *IEEE Computing in Science and Engineering*, pp 52-60.

Irfanoglu, A. and Hoffmann, C. (2007) An Engineering Perspective of the Collapse of WTC-I. *Journal of Performance of Constructed Facilities*, ASCE, 06/2007, in press, available at URL: http://cobweb.ecn.purdue.edu/~ayhan.

LS-DYNA, (2008), URL: http://www.ls-dyna.com

National Institute of Standards and Technology (NIST), (2005). Final Report of the National Construction Safety Team on the Collapses of the World Trade Center Towers". *NIST NCSTAR 1*, Gaithersburg, MD.

Popescu, V., Hoffmann, C., Kilic, S., Sozen, M., Meador, S., (2003). Producing High-Quality Visualizations of Large-Scale Simulations. *Proceedings of IEEE Visualization*, pp. 575-581.

Popescu, V., Hoffmann, (2005) C. Fidelity in Visualizing Large-Scale Simulations, *Journal of Computer Aided Design*, Elsevier, 37, pp 99-107.

Popescu, V. (2008). High-Fidelity Visualization of Large-Scale Simulations. Project web page, www.cs.purdue.edu/cgvlab/projects/pentagon.htm

Rosen, P., Popescu, V., Hoffmann, C., Irfanoglu, A. (2008). A High-Quality High-Fidelity Visualization of the September 11 Attack on the World Trade Center. *IEEE Transactions on Visualization and Computer Graphics, Vol. 14, Issue 4*, pp. 937-947.

Rosen, P. and Popescu (2008), V. 3ds max importer plug-in for LS-DYNA data; code, user manual, and simulation examples, http://www.cs.purdue.edu/cgvlab/projects/dynaPlugin.htm

Sugano T. et al., (1993). Full-scale aircraft impact test for evaluation of impact force. *Nuclear Engineering and Design*, Vol. 140, pp. 373-385.

The Visualization Toolkit, (2008) URL: http://public.kitware.com/VTK

Youtube, (2008), URL: http://www.youtube.com.