Editor: Jim X. Chen, jchen@cs.gmu.edu

DESIGNING, CONTROLLING, AND VISUALIZING COMPOSITE MATERIAL MANUFACTURING

By Dale R. Shires, William Green, and Shawn Walsh

FOR MANY REASONS, BOTH MILITARY AND CIVILIAN TECHNOLOGIES INCREASINGLY USE COMPOSITE MATERIALS. THEY ARE STRONG AND LIGHTWEIGHT. THICK COMPOSITES, IN WHICH ADDITIONAL MATERIALS SUCH AS CERAMIC PLATES CAN be inserted, are even more robust. In-situ sensors, placed into a composite part during manufacture, can monitor manufacturing processes while having little or no effect on final mechanical and structural properties. Readings from these sensors can indicate or delineate different phases of a manufacturing process and determine when a process is complete. Sensors that remain in a composite part after manufacture can also be used, for example, for long-term health monitoring.

Liquid composite molding processes such as resin transfer molding and structural reaction injection molding are high-potential manufacturing methods for fabricating strong, mass-produced composite parts. In general, we place a net-shaped fiber preform (a piece of woven material, usually made from graphite, fiberglass, or kevlar) inside a matched-die mold and inject a reactive liquid resin into the mold cavity. Process engineers also put several vents into the mold cavity to prevent excessive pressure and help guide the resin flow.

However, building composite parts remains problematic. The design and construction of tooling is a tedious process. If we do it incorrectly, we produce flawed parts; then there are long delays to redesign and correct the tooling or manufacturing conditions. Also, composite parts are still generally cured in autoclaves, ovens, or presses using empirically based “recipe” cures. These cures often don’t account for the variability in batch-processed raw materials, degradation during shelf life and uncontrolled environmental conditions, and environmental factors such as temperature and humidity. As a result, recipe cures usually require a high degree of post-fabrication inspection and often produce less-than-optimal parts.

Certainly a more practical and elegant approach to overcoming these problems would be to create an intelligent composite manufacturing paradigm. We have developed a visualization tool for monitoring and displaying resin flow during the fabrication of composite laminates. The tool integrates network-based data acquisition, fast finite-element-based reconstruction (smoothing) and contouring algorithms, and scientific visualization. It helps users better understand the design process and will be at the center of our intelligent design and control manufacturing system.

Intelligent composite manufacturing

The US Army is currently emphasizing future combat systems that will be interconnected, lightweight, and quickly deployable. We are designing and using composite materials for these systems, but such nonmonolithic structures come with a degree of risk (cost and manufacturing complexity). The time and cost to produce a valid part can be prohibitive.

Researchers at the US Army Research Laboratory are developing an intelligent composite manufacturing system incorporating the elements shown in Figure 1. The system will work as follows. After experimenting to determine base material characteristics, users run the Compose modeling system (*Co*mposite *M*anufacturing *P*r*o*cess *S*imulation *E*nvironment, developed by ARL and the universities of New Orleans and Minnesota) to simulate actual processing conditions. This gives them an optimized mold or tool design that they can use to construct a composite part, with embedded sensors for tracking the part’s formation. These sensors provide real-time data to a control system that can activate and deactivate various injectors and vents, thereby speeding up the manufacturing process. Simulation results can fill in gaps left from a discrete sensor array, and experimental data acquired in real time can improve the associated material characteristics database and the fidelity of the model. The system is interactive,

1521-9615/02/$17.00 © 2002 IEEE

with components learning from each other. At the heart of the system is a visualization tool capable of displaying modeling and simulation data and real-time part manufacturing data. Such a display system is vital in this paradigm, whether the controlling decision is fully, partly, or nonautomated. A parallel-processing version of Compose is available for large-scale and complex geometries.[1]

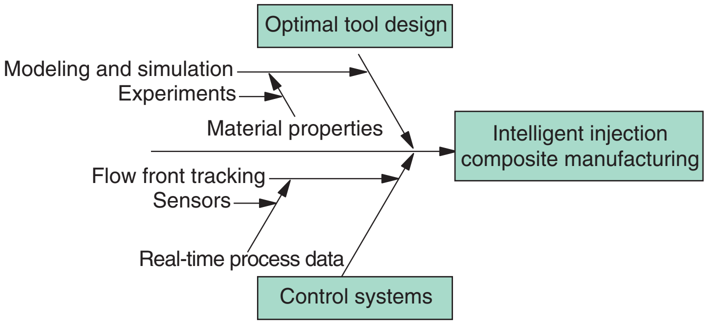

Figure 1. Intelligent composite manufacturing.

Developing a visualization tool

The most important aspect of an injection-based composite processing visualization tool is the monitoring and display of the front of a moving resin flow. Such a system will let users “see” resin progression inside a closed mold, with updates occurring at or near real-time speeds. By incorporating novel sensors, a powerful flow front reconstruction algorithm, and enhanced visualization, process engineers can now better understand injection- and vacuum-based liquid composite molding processes.

Data acquisition

ARL has developed a system known as Smartweave (from *s*ensors *m*ounted *a*s *r*oving *t*hreads).[2] This is a novel approach to creating sensors by weaving flexible, conductive fiber strands through composite preforms. The user places a set of parallel strands between two preform layers, then places a second set of parallel strands at about 90° to the first set and between different layers. The strands are placed between different layers to prevent them from coming into contact with each other. The system then multiplexes the strands by applying low-voltage electrical current to one set and measuring voltage on the other. Basically, this creates a sensor grid of overlapping strands. For example, a 7 × 7 grid would yield 49 sensor locations (see Figure 2).

We can easily apply this type of setup to simple preform shapes such as relatively thin flat panels, but we can use any number of strands, in any orientation relative to each other. So, we can place sensor grids throughout a preform of practically any thickness, including curved sections and around corners from one plane to another. This makes the Smartweave system highly flexible and widely applicable. Smartweave can collect data from any position in a composite preform and thus determine the flow front’s current location.

Early developments in the Smartweave system produced a simple yet effective monitoring system. Customized National Instruments Labview software running on a PC interpreted the multiplexed data that was collected at sensor locations (the grid overlap points). Based on predetermined thresholds, voltages at the sensors indicated whether resin had reached the sensor. The user was presented a graphical, 2D representation of the preform that was colored based on the voltage readings, including whether voltage thresholds had been exceeded. However, this visualization tool was constrained in two ways: only 2D cases could be fully visualized, and the 2D representation was constructed using rectangles delimited by the grid strands.

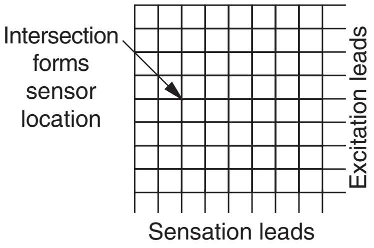

Figure 2. A Smartweave 7 × 7 grid.

This always produced a flow front with 90° steps. Of course, we could add more strands to decrease this effect, but Smartweave’s great advantage is that we can determine flow where no sensors exist by analyzing sparse sensor data. Flow in these areas could not be accurately shown with the “blocky” output from the earlier PC-based visualization tool. Also, adding more strands becomes impractical at a point.

The Smartweave system is designed to be inexpensive and can be taken on the road to perform analysis and gather data quickly. In most cases, we can use a simple laptop computer as a sensor data collection device. To visualize 3D experiments, we developed a method to transfer the data from the monitoring computer to an SGI workstation in near-real time, thanks to network access. We developed software for the PC and for the SGI to enable commu-

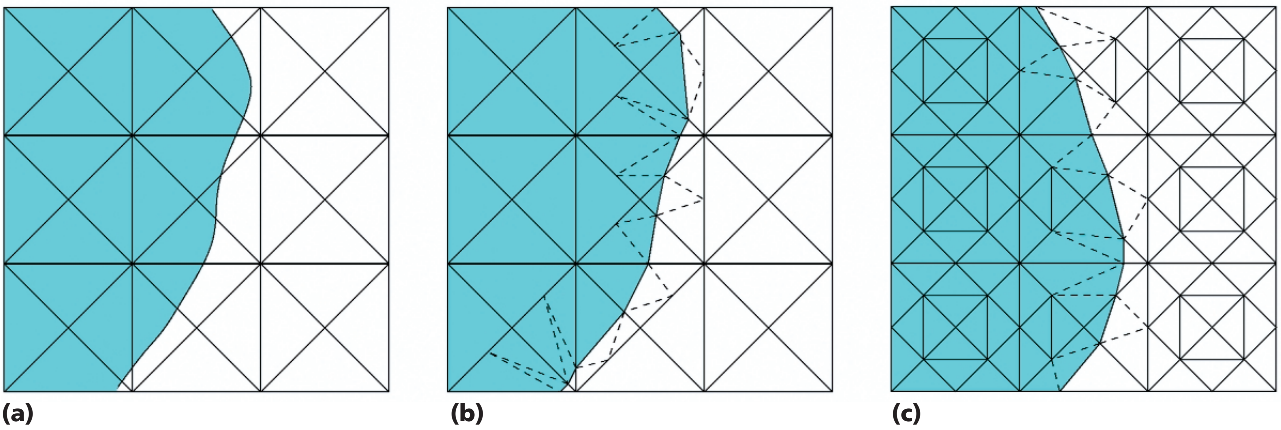

Figure 3. Flow front reconstruction: (a) the front moves left to right; (b) four-element refinement; (c) element repositioning by node relocation.

nication between the two existing TCP/IP network ports. The Labview software on the PC, which is connected to the data acquisition system, collects sensor data and transmits the information over a network to the SGI.

Reconstructing the flow front

Smartweave can handle any number of sensors, but it is desirable to use as few as possible to correctly determine the location of the flow front. Currently, the sensors activate when the resin flows through the preform and hits the point where the sensor leads overlap. But besides knowing the discrete grid of sensors and their associated states (on or off), we also want to know where the flow front is between the sensor points. The system can quickly reconstruct the continuous flow front based on sensor locations and sensor activation (on) times. In the 2D case, the reconstruction (smoothing) algorithm uses triangular, three-noded elements based on a finite-element model.

A finite-element mesh is a series of points in 3D space connected by lines to form elements (this article's examples are based on triangular elements). A finite-element mesh is useful in two ways. First, it lets us abstract from a full-featured computer-aided design entity to the simple concept of a part being built from nodes and elements in 3D space. This abstraction is vital to our approach to scientific visualization: we can change the shape of elements and create resin-saturated regions by simply modifying the mesh. So, we can create a flow front contour where elements in front of the boundary are unfilled and elements behind are filled. The visualization software then colors these regions appropriately.

Second, we also use the finite-element mesh for simulation analysis of resin progression in the mold. The Compose software can use the same mesh for modeling and simulating various process scenarios. This preprocessing step helps us better determine the optimal placement of injectors and vents in the mold and establish a working production methodology on the first attempt, rather than using a costly trial-and-error approach in which the mold must be retooled.

The algorithm determines the flow front time solution for every node in the finite-element model given the time that sensors are activated. Thus, it analyzes sparse data from a small number of sensors to estimate for the entire finite-element model, thus giving the resin flow front at discrete time intervals. Some elements will have all their nodes in the filled (on) state, some will have all their nodes in the unfilled (off) state, and some will have only one or two nodes in the filled state. Any element that has only one or two nodes behind the flow front is considered partially filled with resin. We used a linear first-approximation contouring methodology to determine what areas of partially filled elements should be considered filled. Contouring is generally not necessary for a relatively fine finite-element mesh. However, because the system applies contouring only to partially filled elements, this happens only along the flow front. Contouring a relatively small number of partial elements can take less time than smoothing a very fine mesh to avoid contouring, because solving for such a mesh can be computationally intensive.

Consider the mesh sections shown in Figure 3. Figure 3a is a 2D finite-element mesh, or grid, with the flow front moving from left to right. Figure 3b is the same grid with four-element dynamic mesh refinement (DMR) applied to the partially filled elements along the flow front. All the elements to the left of the flow contour line are colored as filled, those to the right as unfilled. Four-element refinement is the process of creating four new elements that collectively describe the same spatial area as the partial element they temporarily replace; the flow front is delineated, or contoured, in the area of the partial element because each temporary new element is either completely in the flow region or completely out of the flow region. As the flow front moves through the grid, the system adds and removes nodes and elements from it as necessary to contour partial elements.

Figure 3c shows a similar case using a more refined finite-element mesh. Once again, flow is moving left to right, with elements changing states from empty to partly filled to filled. Whereas nine elements are dynamically replaced with 36 elements in Figure 3b, Figure 3c uses simple element repositioning by node relocation. The finer mesh requires no expansion to the total number of elements. However, increased processing time for this more refined mesh makes it less desirable. To use as coarse a grid as possible and to minimize processing time, we prefer DMR, as in Figure 3b. Because our flow simulation codes are sometimes more accurate with a finer mesh, we support both techniques to potentially limit the number of meshes we must create for a part.

Developing the visualization interface

To simplify the visualization task, we chose to implement our system using Open Inventor, an object-oriented 3D toolkit based on the Open GL graphics library. Open Inventor is designed to easily and quickly describe, create, and display 3D objects. The program stores all the information about an object (in this case, a computer graphic representing a composite part), such as its size, position in virtual space, and color, in a data structure known as a scene database or scene graph. The graph is rendered as it is traversed top down, left to right. Traits and characteristics added to the scene graph determine how it is rendered. We are investigating extensions of the interface to use the Visualization Toolkit and Java for cross-platform support.

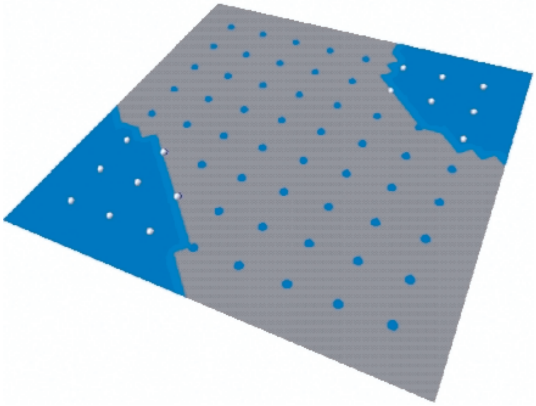

Figure 4. Coloring by finite-element node.

The parts considered here are represented as finite-element structures, so we can use an Open Inventor scene graph to describe them. The flow visualization interface quickly reads and converts a Nastran (finite-element) file into an Open Inventor scene graph, which renders the object described by the file. However, we could use any data file specifying nodes and nodal connectivity. The system uses the same finite-element mesh used for rendering the object for smoothing, contouring, and performing flow simulations. Thus, the mesh should be neither too fine nor too coarse. If it is too fine, we get good simulation results with poor rendering and contouring performance. If it is too coarse, the opposite is true.

For the flow front to display accurately when the system receives new sensor data, the time to complete smoothing, contouring, and rendering tasks must be small relative to the flow front's speed. If the computation time required to complete these tasks is too long, the displayed flow front will suffer time lag behind the actual flow front. The visualization interface lets the user turn off smoothing and contouring to view the sensor state only. The interface can also apply only smoothing to sensor data. In this case, the system compares the time solution for each node in the mesh to the current time and then fills any node with time less than or equal to the current time.

The interface uses color shading around nodes to distinguish between filled and empty nodes. However, shading by node gives the flow front a step-like, less-than-sharp edge. This method gives better results than the PC-based display but still does not clearly delineate the flow front (see Figure 4). The figure shows active sensors displayed in white and inactivated sensors in blue. Resin-filled areas are colored blue, and void areas are gray. Notice that there are no clear, defining lines for the resin front. These results provided the impetus to develop element shading with DMR for contouring, which gives the flow front a well-defined edge without step-like contours between nodes. This approach shows the entire flow, but we also developed an approach to show only the flow front, using splines to define the front's nodes.

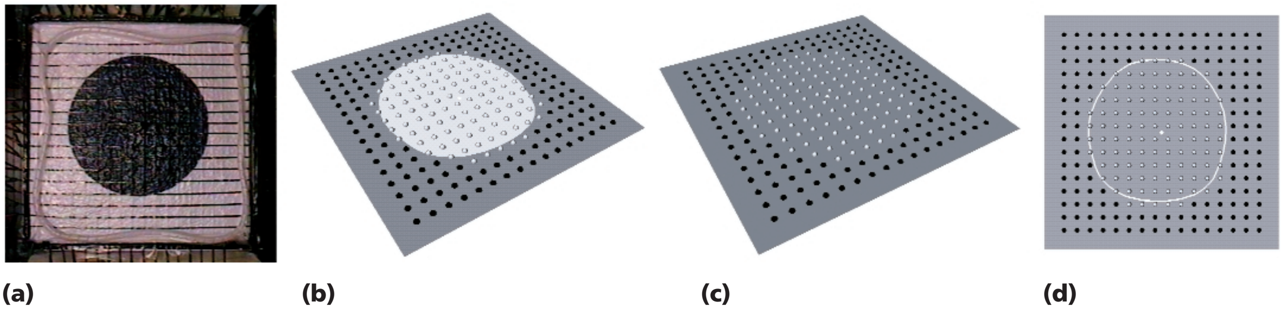

Figure 5. Flat panel experiments and visualization: (a) video image of flow; (b) sensor activation only; (c) sensor activation, reconstruction, and contouring; (d) sensor activation and flow front.

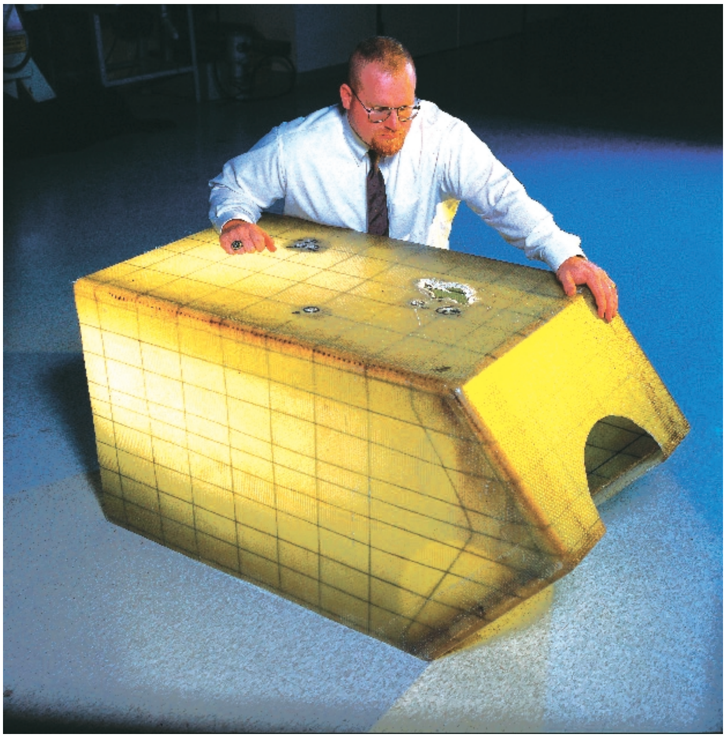

Figure 6. A more complex composite structure with sensor weave.

Results

First, we tested the software's correctness on a simple flat panel. Then, we applied it to a more complex, real-world component.

Flat panels

First, we applied the SGI visualization interface to monitor flow in flat panels that were fabricated using a vacuum-assisted resin transfer molding process. This one-sided tooling approach uses vacuum force to distribute resin over large surface areas. This approach uses a highly permeable medium placed between the fiber preform and the mold surface. The technique can also be augmented with grooves cut in the core or the mold. One-sided tooling lets us test the visualization interface quickly and easily without an autoclave. The process also allows visual tracking of the flow fronts. We impregnated the panel with resin (simulated with colored corn syrup) from the center.

Figure 5 shows several rendering abilities of the visualization system. Figure 5a is a video image snapshot of flow in the panel correlated in time with Figures 5b–d. Figure 5b shows the sensor data when the panel was about half-filled with resin. Sensors that are on are white; sensors that are off are black. Figure 5c shows the same sensor state with both reconstruction and contouring applied. The flow is shaded by element (shown in white), and the flow front is well defined by using DMR. Figure 5d is the same flow showing only the flow front, where DMR was used. To select the reconstruction visualization, the user simply chooses an appropriate menu item. These choices provide maximum flexibility in visualizing the flow front.

Ballistic shield

We also used the interface to monitor flow in a much more complex composite part, the gun mount shield, hereafter referred to as the ballistic shield (see Figure 6). Again, we used vacuum-assisted resin transfer molding to fabricate a thinner version of the full-thickness shield. We placed strands from front to back and from side to side across the top of the shield to form five planar sensor grids, including one in the top, one in each side, and two in the front. Whereas

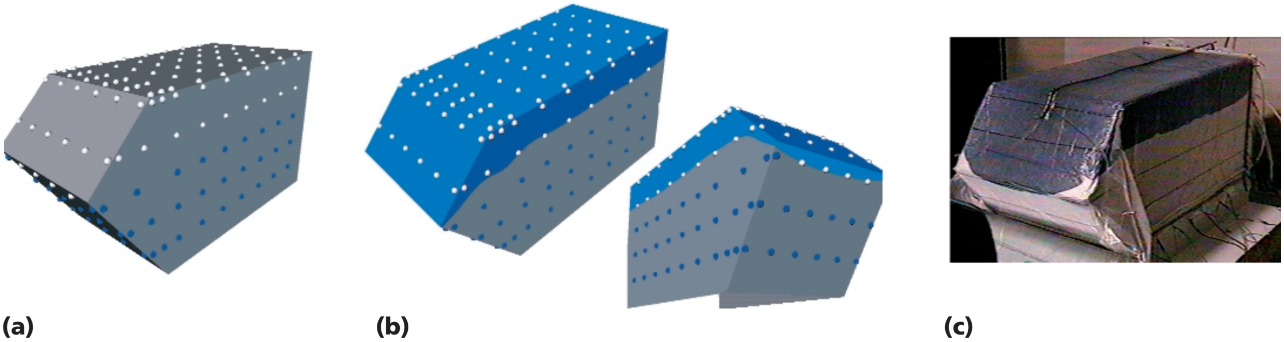

Figure 7. Ballistic shield experiments and visualization: (a) sensor activation only; (b) sensor activation, reconstruction, and contouring; and (c) photo of in-process manufacture.

the PC-based tool could display only one sensor grid at a time, the SGI visualization tool continuously displayed the entire ballistic shield, treating it as a single object with a set of properties to be rendered. Furthermore, the workstation tool let the user move or spin the shield in space to view it from any angle.

The shield was impregnated with resin from the center of the back at the top. Figure 7a shows the sensor state after flow has proceeded through the top and into the front, sides, and back of the shield. Sensors that are on are white, and sensors that are off are still blue.

Figure 7b shows two views of the reconstructed and contoured flow for the sensor state, where flow is shaded by element using DMR. Figure 7c is a video image snapshot of flow in the shield correlated in time with Figures 7a and 7b. Again, the flow is in good agreement, including flow through edges and corners from one area of the shield to another. This is critical, as effects such as race tracking can occur along edges.

We are actively engaged in basic and applied research to promote the use of composites in military and civilian systems. Scientific visualization provides critical capabilities in this endeavor. Our graphical visualization tool for monitoring and displaying resin flow in near-real time offers several extensions. In the near future, we plan to add an appropriate control decision mechanism and then investigate using the visualization interface to control the composite manufacturing process. The reconstruction algorithm can solve for the time solution of every mesh node given the current sensor grid state, and for the velocity at every mesh node. Knowledge of where the flow is (time) and where it is going (velocity) can be combined and used by some form of intelligent control. Control might be based on statistical methods, heuristic algorithms, neural networks, or other artificial intelligence algorithms. Effective visualization remains at the heart of such a system.

References

1. R.V. Mohan, D.R. Shires, and K.K. Tamma, "Advanced Manufacturing of Large-Scale Composite Structures: Process Modeling, Manufacturing Simulations, and Massively Parallel Computing Platforms," *Advances in Eng. Software*, vol. 29, Apr. 1998, pp. 249–263.
2. S.M. Walsh, *In-situ Sensor Method and Device*, US patent 5,210,499, Patent and Trademark Office, Washington, D.C., 1993.

Dale R. Shires is a computer scientist in the High Performance Computing Division at the US Army Research Laboratory. His research interests are in algorithm development and optimizations for massively parallel architectures, modeling and simulation, and scientific visualization. He received an MS in computer science from the University of Delaware and is a member of the IEEE Computer Society and the ACM. Contact him at the ARL, Computational and Information Sciences Directorate, ATTN: AMSRL-CI-HC, Aberdeen Proving Ground, MD 21005; dshires@arl.army.mil.

William Green is a materials engineer in the Weapons and Materials Research Directorate at the US Army Research Laboratory. His research interests include nondestructive test and evaluation methodologies, real-time radiography, and computed tomography. He received an MS in nuclear physics from Rensselaer Polytechnic Institute. Contact him at the ARL, Weapons and Materials Research Directorate, ATTN: AMSRL-WM-MD, Aberdeen Proving Ground, MD 21005; wgreen@arl.army.mil.

Shawn Walsh is a materials engineer in the Weapons and Materials Research Directorate at the US Army Research Laboratory. His research interests include the novel processing and evaluation of materials, sensor technologies for inspection and process control, and multifunctional materials. He is a member of the American Society of Mechanical Engineers and the Society for the Advancement of Material and Process Engineering. He received a doctorate of engineering from the University of Massachusetts. Contact him at the ARL, Weapons and Materials Research Directorate, ATTN: AMSRL-WM-MD, Aberdeen Proving Ground, MD 21005; swalsh@arl.army.mil.