The Digital Ceraunoscope: Synthetic Thunder and Lightning, Part 2

Andrew Glassner

Microsoft Research

In the last issue (March/April 2000), I started discussing my design for a digital *ceraunoscope*—an ancient machine designed to simulate thunder and lightning. I talked about how cloud-to-ground lightning formed and how to write a program to generate synthetic lightning strokes that matches real data gathered for the Apollo space program. This time I'll focus on making thunder an integral part of the simulation.

Let's back up for a moment and think about how we might go about getting data on the shape and structure of lightning. The most straightforward way is probably to take a photograph of some lightning, then measure the photograph. That's pretty reasonable and it has led to several mutually consistent studies on lightning shape, including my results in the last column.

But what if you can't photograph the lightning? An important and common form of lightning is called *intercloud discharge*. This name describes lightning that leaps from one cloud to another. Even though it passes through unobstructed air, clouds below the discharge might block it from a photographer on the ground. Even trickier is *intracloud discharge*, when everything happens inside a single cloud. This can happen when two different regions of the same cloud have strong but opposite charges. Then, like cloud-to-ground discharges, the electric potential breaks down the air in between the two regions and creates lightning.

How might we measure these difficult-to-see lightning strokes? One idea is to set several microphones on the ground and record the thunder they create. If you have a good model of how thunder is formed by a particular lightning stroke, then you might have a chance of running the model backward. In other words, you could analyze the recorded thunder sound pattern, or *signature*, and from it compute the shape of the lightning stroke that caused it. You'd probably want to use a few calibrated microphones and triangulate the results to locate the position of the lightning in space.

This idea helped to drive the development of exactly such a model in the early 1980s. Similar approaches have been refined now to the point where desktop computers can be used to identify and locate lightning.[1]

Using a combined lightning and thunder model, we can create coordinated, unified lightning and thunder, starting from either phenomenon to produce the other. It will look and sound as we wish, start and end as we wish, and basically become another digital tool for us to direct freely—a digital ceraunoscope.

Let's start by looking at how lightning creates thunder.

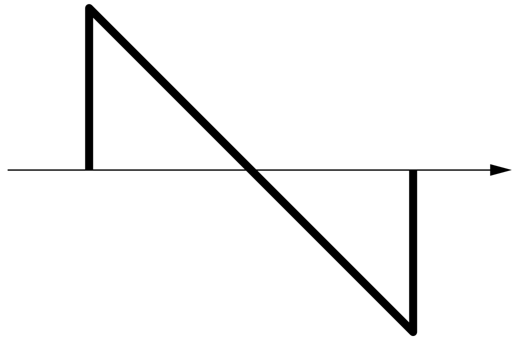
The shock of thunder

Thunder results from a sudden and intense heating of the air in the lightning channel, caused by the huge flow of current. Let's think of a small region of space somewhere along the lightning channel—I'll call this the source volume. As a result of all the electrical current flowing through the air, the gases in the source volume rapidly heat to temperatures that often exceed 30,000 degrees Kelvin—that's almost six times hotter than the surface of the sun. This intense and sudden heating causes the pressure in the source volume to increase to levels as high as 10 to 100 atmospheres.[2] Such a sudden and extreme burst of pressure causes a shock wave. Technically, a shock wave is a disturbance that simultaneously and significantly compresses and heats the medium it passes through—in this case, air.

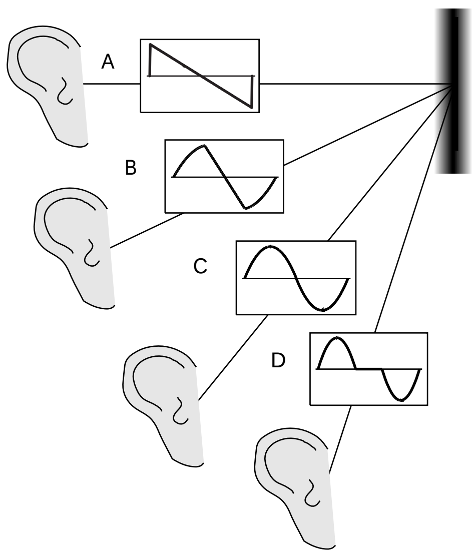
This shock wave expands outward from the source volume. Within a few meters, the wave dissipates almost 99 percent of its energy into heating the air. The one percent that remains travels through the air at the speed of sound, but now as a more normal, though intense, acoustic wave. Even though the original shock wave has lost all but a small fraction of its power, its original energy is so enormous that the resulting acoustic waves are still overwhelming, even to people on the ground kilometers away. Those powerful, acoustic waves are the phenomenon we call thunder.

When the shock wave first forms at the source volume, its pressure profile is very complex and changes quickly and nonlinearly as it expands. As I mentioned earlier, when the wave has lost most of its energy, it turns into a more tame acoustic wave. The distance at which this happens from the original source volume is called the *relaxation radius*. Beyond the relaxation radius, the structure of the pressure wave becomes very simple. Figure 1 shows an example of the idealized pressure profile, which for obvious reasons is called an N-wave. Real N-waves, of course, have finite rise and fall times and rounded corners. The relaxation radius is typically on the order of a few meters—more on that later.

This basic model could be used to compute thunder, but it would be expensive. Think of the entire lightning stroke—branches and all—as a bunch of strings of pearls. We could take each pearl one by one, treat it as a source volume, and compute the N-wave that it creates. Then we could follow that wave through the air and determine how long it takes to reach a listener at a particular position on the ground (and how loud it is at that point). After we do this for each pearl, we have a bunch of pressure waves of different strengths, starting points, and durations. All we have to do then is add them up, play it back, and voila, we have thunder!



1 **An idealized N-wave. A listener would experience a sudden positive pressure, which linearly drops to a large negative pressure, which then abruptly disappears. In practice, the sharp corners are rounded, the vertical rise and fall take some time, and the general progression is not so perfect.**



2 **The shape of a WM-wave depends on the angle between the listener's ear and the radiating segment. When perpendicular to the segment, the wave is like an idealized N-wave. As the angle to the segment increases, the WM-wave rounds out and eventually separates into two parabolas.**

The next segment

The string-of-pearls model is a conceptually fine way to create a *thunder signature*, or the pressure profile, that's created for a given listener from a given lightning stroke. But this sort of point-sampling thunder technique has two drawbacks. First, it's computationally expensive, because we would require a great many points along the lightning channel to get a good approximation to it. The other problem is that it would be prone to all the aliasing and undersampling problems that plague point-sampling rendering algorithms.

An alternative is available through the work of Wright and Meddendorp.[3] They started by thinking of a lightning channel as a tree of many small, straight segments—just as we modeled it in the last column. They reasoned that if they could precompute the pressure wave emitted by one of these segments, then they could dispense with all that point-sampling along the channel.

Following this line of thought, they integrated the N-waves along a straight linear segment and found an analytic expression for the composite acoustic wave that results—now called a WM-wave in their honor. Thus, you need only walk along the stroke segment by segment and accumulate the WM-waves arriving at the listener's location.

The WM-wave that arrives at a listener's ear depends on the listener's orientation with respect to the radiating segment, as Figure 2 shows. If the listener is perpendicular to the segment, the result is simply an N-wave. As the listener moves, the ends of the wave turn into parabolic arcs. At the critical angle when $\sin\theta = \varphi$, the WM-wave has two distinct parabolas with no linear segment between them. At larger angles the two parabolas separate.

The equations for a WM-wave follow. Two cases exist, depending on whether $\sin\theta$ is less than or greater than φ. I present the two cases individually—the differences are in the limits on the parameter τ, defined below:

$$p_{\sin\theta<\varphi} = \begin{cases} 0 & \tau < -\varphi - \sin\theta \\ (-B/\sin\theta)[(\tau+\sin\theta)^2 - \varphi^2] & -\varphi - \sin\theta < \tau < -\varphi + \sin\theta \\ -4B\tau & -\varphi + \sin\theta < \tau < \varphi - \sin\theta \\ (B/\sin\theta)[(\tau-\sin\theta)^2 - \varphi^2] & \varphi - \sin\theta < \tau < \varphi + \sin\theta \\ 0 & \varphi + \sin\theta < \tau \end{cases}$$

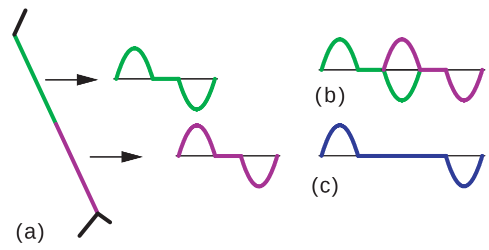
$$p_{\sin\theta\geq\varphi} = \begin{cases} 0 & \tau < -\varphi - \sin\theta \\ (-B/\sin\theta)[(\tau+\sin\theta)^2 - \varphi^2] & -\varphi - \sin\theta < \tau < \varphi - \sin\theta \\ -4B\tau & \varphi - \sin\theta < \tau < -\varphi + \sin\theta \\ (B/\sin\theta)[(\tau-\sin\theta)^2 - \varphi^2] & -\varphi + \sin\theta < \tau < \varphi + \sin\theta \\ 0 & \varphi + \sin\theta < \tau \end{cases}$$

where

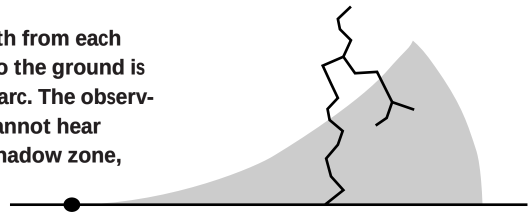
$$\tau = \frac{ct - r}{l} \qquad \varphi = \frac{cT}{l} \qquad B = \frac{Al^2}{2rcT}$$

θ angle from listener to segment normal
A arbitrary scale factor (usually 1.0)
T duration of wave (approximately 5 ms)
l length of segment (usually 3 meters)
c speed of sound (343 meters per second)
r distance to listener in meters

Imagine two adjacent segments colinear and equal in



3 Two sequential, parallel segments of lightning produce WM-waves that cancel each other out. (a) Two sequential, parallel segments and their waves. (b) The two waves superimposed in time. (c) The resulting composite pressure wave.



4 The refraction path from each lightning segment to the ground is treated as a circular arc. The observer (at the big dot) cannot hear anything from the shadow zone, shaded gray.

length. Then the positive lobe of the second segment will exactly cancel the negative lobe of the first, as Figure 3 shows. In fact, any straight chain of segments will cancel out, leaving only the parabolas at the start and end. The remarkable result is that, for any two adjacent segments, it's the kink, or angle, between them that determines the acoustic wave they generate.

A unified model

Now we can turn to putting it all together. To help us, let's look at some techniques for coupling lightning and thunder measurements that several researchers have developed.

Few and colleagues[2] developed a method that starts with several simultaneous thunder recordings taken in different locations, then uses triangulation to find points on the lightning channel. Ribner and colleagues[4] worked the other way around. They began by synthesizing a 3D lightning stroke geometry and developed the thunder corresponding to that stroke. When the original lightning geometry is statistically accurate, this process generates convincing thunder statistically similar to real thunder.[5,6] We'll follow their approach.

The idea is to generate statistically correct 3D stroke geometry, then accumulate the WM-waves from each segment along the bolt at an observer's location.

The simulator

In my last column, I chose 3 meters as the length of the small, fine-level details in a synthetic lightning bolt. I defended that choice based on rendering considerations, but also promised that in this column I would present two good reasons based on thunder measurements. Let's look at those reasons.

The first justification for a 3-meter, fine-structure length comes from the geometry of the situation and the measured spectra of thunder.[6] Consider a typical lightning bolt that spans about 5 km from ground to cloud, and an observer about 3 km away—the distance from the observer to the top of the stroke is about 5.8 km. At the speed of sound (343m per second), it will take about 8.25 seconds for the acoustic wave at the top of the bolt to reach the listener, so this is about how long the thunder will last. Measured thunder spectra fall off at around 200 Hz. That is, most of the energy is at 200 Hz or lower. A 200-Hz signal lasting for 8.25 seconds needs to have about 1,650 cycles, or zero-crossings. Each WM-wave contributes one such zero-crossing, so we would need about 1,650 of these waves. Since there's one wave per segment, we need 1,650 segments, and 5 km per 1,650 segments is about 3 m per segment. Whew.

The second justification comes from considering that the emissions from very small segments get blurred as the initial shock wave expands—detailed pressure information within the relaxation radius is effectively lost.[7] Since the relaxation radius is about 3 to 5 meters, any effects due to structure smaller than that would get lost or sonically blurred away.

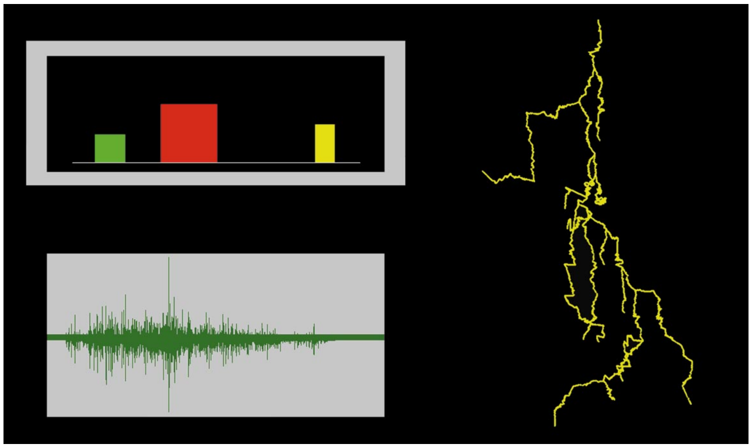
Once we've built the fine-structure lightning bolt, we can walk along the stroke, accumulating one WM-wave per segment. We simply compute the angle made by the listener to the line of the segment, create the appropriate WM-wave, determine its time of flight, and add it into the accumulating sound signal at the listener's location. The flight time for each WM-wave needs to account for atmospheric refraction, which changes the length of the waves and alters their flight path from linear to curved. As the flight time increases, the stretch of N-wave number i is modeled approximately by the following relations,[6] which give us new values for φ and B when computing the WM-wave using the equations above:

$$k = \frac{\ln(r_0 / a)}{\ln(r_i / a)} \qquad \varphi_i = \frac{cT_0\sqrt{k}}{l} \qquad B_i = \frac{Akl^2}{2r_i cT_0}$$

where $a = 10$ and $r_0 = 1{,}000$m.

But the sound does not travel in a straight line. Changes in temperature and atmospheric density cause it to bend, like a visual mirage due to temperature gradients. A convenient approximation to the refraction path is to treat it as a circular arc. This naturally creates a "shadow zone" near the ground, from which the observer hears nothing, as Figure 4 shows. Normally, the radius of curvature due to refraction in air is about 80 km without wind, though this decreases to about 55 km during thunderstorms.[5] This means that for a ground observer at a distance of about 3 km from the stroke, the sounds waves generated for the lowest 82m are inaudible. In reality, some of these waves do reach the listener because of other atmospheric effects. To account for this, in my simulator I modulate the waves from the shadow zone with a smooth ease-out curve over the length of the shadow zone.

Refraction, reflection, wind effects, and other atmospheric phenomena contribute to the propagation of acoustic waves through the atmosphere.[8] A detailed simulation of all of these effects would be overkill for computer graphics purposes. I mean, we want to model nature accurately, but we need to draw the line somewhere. Rather than try to model all of these effects, I apply a small amount of random variation to the time of flight of the WM-waves. To account for the fact that the atmospheric disturbances are spatially localized, I change the random time displacement changes slowly over the length of the channel. This is easily accomplished using a low-frequency volumetric noise function for the atmosphere. I also apply a small amount of low-pass filtering on the final signal to smooth out sharp pops that would dissipate through the normal motion of the air.



5 Thunder design. In the upper left, the designer has placed three blocks, one each for roar (green), clap (red), and rumble (yellow). On the right, a synthetic lightning stroke with natural statistics that generates the specified thunder. In the lower left, a plot of the resulting thunder signature.

Designer thunder

In my last column I discussed how to create designer lightning from a simple 2D skeleton. We've seen above how to create the thunder due to that, or any other, lightning bolt.

We can run the process the other way around. Suppose we have a scene where a little boy is curled up in his bed, scared of the thunder outside. He hears a thunderclap and ducks under the covers, then a long, slow rumble makes him shiver in fright. Then a final, louder burst of sound rolls into the room and sends the boy scrambling to his parents. Once the performance is captured on film, we'll want to add the appropriate thunder in postproduction.

Of course, we could fake it and build up a convincing thunder simulation using some recordings and a good digital audio tool. But let's go for the real deal—we'll design the kind of thunder we want in terms of sound and timing, then create a lightning bolt that would make that thunder. Then we can show that lightning bolt out the window and compute the thunder it creates, which will arrive exactly on cue and sound exactly the way we want. Hey, the audience might not notice that everything is consistent and accurate, but we'll know that our illusion is plausibly real. A small pleasure, to be sure, but small pleasures are good, too.

To describe thunder I've built a little user interface that lets the designer place three kinds of elements in time over the duration of the complete thunderclap, or thunder signature. The three elements build on the three kinds of sonic phenomena frequently used in the thunder literature: *clap*, *rumble*, and *roar*.

We start out with a graph that represents the stretched-out length of the lightning stroke's central channel. Over that channel a designer can place colored blocks that specify the type of feature, its starting time and ending positions (the left and right edges of the block), and the feature's amplitude (the height of the block). Note that placing thunder effects along the lightning channel is not the same as placing them in time, and this can be a little confusing at first. My reason for doing it this way is best explained by the clap feature.

A thunderclap results when all of the WM-waves from a section of a lightning stroke reinforce one another and arrive in-phase. This comes about when the time of flight for each WM-wave is the same. Geometrically, this means that some section of the lightning channel forms a circular arc with the listener at the center.[6] Thus when the designer assigns thunderclap to a given section of the lightning channel, the simulator shapes that section of the skeleton into an arc. Because all the wavefronts arrive at the same moment, a thunderclap has essentially no duration. For this feature, the width of the block in the design tool (and thus the amount of channel that is shaped) controls the ultimate strength of the clap, not its duration in time.

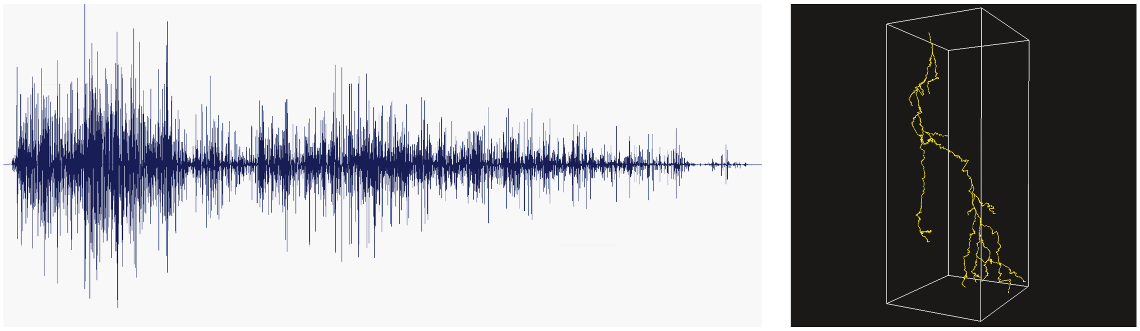
The other thunder features are rumble and roar, both of which map to specific kinds of lightning geometry. Rumble is created when the channel is just about (but not exactly) end-on to the observer. Roar is a region of high tortuosity—the random walk taken by the segments around the skeleton is much more exaggerated than where roar is not desired.

Thunderclaps, rumble, and roar are all created by shaping the skeleton, and then reshaping the tortuous lightning bolt.

Figure 5 shows an example of this design process. The designer has placed a roar, thunderclap, and rumble along the channel. The figure shows the three blocks, the lightning stroke created in response to the specification, and the pressure profile of the resulting thunder. Note the strong peak indicating the thunderclap. It would be useful to also build a time-based user interface for thunder design, but I haven't done that yet.

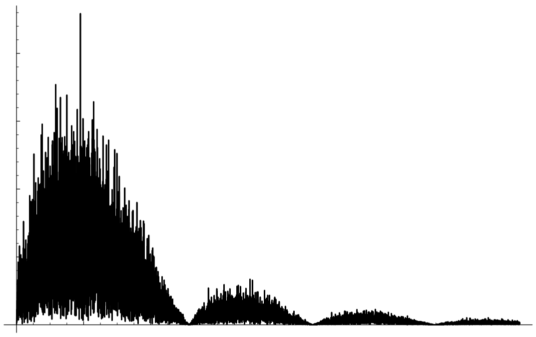
Rendering

Rendering the final sound is straightforward, once all the WM-waves have been computed and added up to create a thunder signature. I just convert the raw pressure information into a signal magnitude for a monaural sound file. Because of the distance of the thunder's origin and its composition of relatively low frequencies



(a) (b)

6 **Another synthetic pair. (a) The thunder signature. (b) The lightning stroke.**



7 **The absolute value of the low-frequency region of the Fourier transform for the thunder signature in Figure 6a.**

Digital Storm Symphony

I continue to be excited by the idea of combining physically accurate simulations with designer control. I love the idea of being able to direct large, complicated, and normally unpredictable natural phenomena with the same ease as any other synthetic element.

Digitally, you could compose, orchestrate, and perform a storm symphony, like Beethoven's *Eroica*. Rather than represent the storm musically, you would present the composed sights and sounds of the storm itself, arranged and timed with the same precision and purpose as a musical symphony. In my mind's eye, the first movement is the arrival of clouds on a calm seashore summer afternoon and the increasing agitation of the crashing waves on the beach. The second movement tells of the fury of the thunderstorm, the crash of lightning and the power of thunder, the sound of rain falling in different bursts and rhythms, the wind in the trees, and the small branches flying through the air. The third movement reveals the ending of the storm and the gradual ebbing of the violent sights and sounds. The symphony would end on a hopeful note, with a glimpse of sun through the dissipating gray clouds, and the hint of a rainbow.

(which are difficult for humans to localize), I haven't generated any stereo sound files.

Figure 6 shows another thunder signature created with this method. I used a synthetic lightning stroke that flashed five times, with a mean interflash timing of 5 ms. Figure 7 shows the Fourier transform of this signature.

The digital ceraunoscope

In this column and the last, I've discussed how to build a principled, physically accurate digital lightning and thunder synthesis algorithm. You can use a system based on this approach in several ways. You can run the lightning simulator without control, then generate the corresponding thunder. You can design a lightning stroke skeleton, then promote it to a statistically accurate channel that creates thunder. Or you can specify the thunder you want and create the lightning channel that would produce it.

I haven't been able to discover the mechanics of the original, presumably mechanical ceraunoscope—I don't know how the earliest recreators of lightning and thunder approached their task (my reference to the word ceraunoscope comes from an online copy of *Webster's 1913 Revised Unabridged Dictionary*). But with the computer, it's all just a matter of software. ■

Acknowledgments

Thank you to David Thiel and Steven Drucker for help with the text and figures.

References

1. T.P. Ludy, "Source to Strike," *Scientific Computing and Instrumentation*, Oct. 1999, pp. 12-19.
2. A.A. Few, "Thunder Signatures," *Trans. American Geophysical Union*, Vol. 55, No. 5, 1974, pp. 508-514
3. W.M. Wright, "Propagation in Air of N Waves Produced by Sparks," *J. Acoustical Society of America*, Vol. 73, No. 6, June 1983, pp. 1948-1955.
4. H.S. Ribner et al., "Computer Model of the Lightning-Thunder Process with Audible Demonstration," *Progess in Aeronautics and Astronautics,* American Inst. of Aeronautics and Astronautics (AIAA), Reston, Va., 1976.
5. D. Roy, *A Monte Carlo Model of Tortuous Lightning and the Generation of Thunder,* UTIAS tech. report no. 243, Institute for Aerospace Studies, Univ. of Toronto, Feb. 1981.
6. H.S. Ribner and D. Roy, "Acoustics of Thunder: A Quasilinear Model for Tortuous Lightning," *J. Acoustical Society of America*, Vol. 72, No. 6, Dec. 1982, pp. 1911-1925.
7. A.A. Few, "Thunder," *Scientific American*, July 1975, pp. 80-90.
8. J.E. Piercy, T.F.W. Embleton, and L.C. Sutherland, "Review of Noise Propagation in the Atmosphere," *J. Acoustical Society of America*, Vol. 61, No. 6, 1977, pp. 1403-1418.

Readers may contact Glassner by e-mail at glassner@microsoft.com.