

Lecture 15- Pre-lecture reading

Digital Radio and its Effect on Cellular

In the 1980s, the telephone network went digital, and in the 1990s, for an entirely different reason, cellular went digital as well. In this lecture, we will discuss how digital communications works, and why these networks went from analog to digital.

We will begin with a quick look at what bits are and how they are sent on wires and waves. Then we will turn to the digital revolution in telephony. Why did the telephone network go digital, beginning in the 1980s? We will see that it was mostly due to the arrival of fiber-optic communications— pulses of light traveling through glass fibers the thickness of a hair. With seemingly infinite capacity, this technology revolutionized telephony and created the Internet, but as we have seen, radio channels are not the same as fiber—their information capacity per Hz of spectrum has changed very little. “Second generation” (2G) digital radio channels didn’t carry dramatically more bits/second as fiber did; but because of the new technology of voice compression calls needed fewer bits/second for good quality. This sudden increase in voice capacity was expected to triple the users per cell and thus reduce the investment per user, but it didn’t turn out that way. Finally, we will see the impressive profitability of digital text messaging for the cellular operators.

While analog transmission sends a modulated wave that actually looks like the voice, digital communication converts the signal to a bunch of numbers. Usually, those numbers are binary ones and zeros, because ones and zeros are easy to create with electrical circuits— they can be as simple as the on and off positions of a switch, or by two different voltages or two different frequencies. [slide 4]

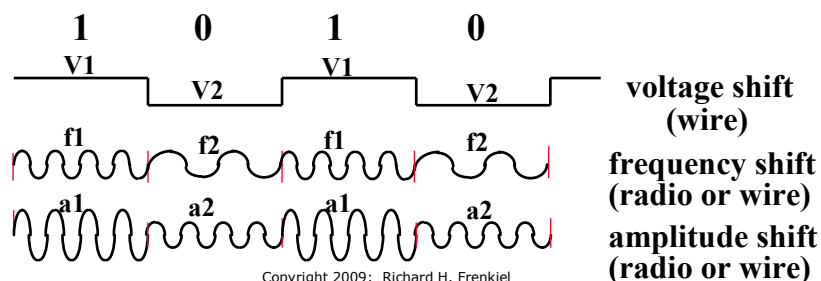
So what’s “digital” anyway?

□ Information represented by “bits”

- 1 bit is a “1” or a “0”
- A decimal number like 127 can be represented as a binary number like 1111111

□ The bits can be sent on wires and waves

- 2 levels of a voltage or frequency; 2 frequencies

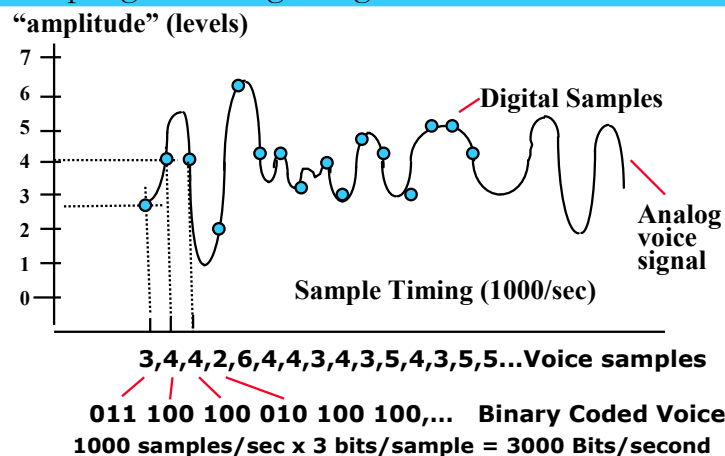


With a group of 8 bits (commonly called a “Byte”) we can represent any written character (for example let A=00000000, B=00000001, etc.), so we can sent text, but how do we turn a complex analog voice or music signal into bits?

A voice signal is information, and so is music or a movie. We can represent the now-familiar analog voice signal we’ve been using as a series of bits, send those bits somewhere, and then reconstruct the original voice when we want to. We do that by measuring the voltage of the voice wave, as shown on slide 6. We decide on a “sampling” rate (e.g., 1000 samples a second). Then we measure the amplitude (height) of the wave at each sampling point and assign it a “level,” according to the scale shown on the left. The voice wave is now defined by that series of samples shown below, which is actually a series of numbers (3,4,4,2, ...). We don’t send 3s and 4s, of course, so we convert those numbers to binary, where 3 = 011, and 4 = 100, and so on. Now we can send the digital representation of the wave (without those commas separating the samples) as a long string of bits (011100100010100100).

Since we are taking 1000 samples a second, and coding each sample as three bits, in the example shown on slide 6 we need to send 3000 bits/sec.

Turning the “analog” voice signal into numbers
“Sampling” and “digitizing” that familiar voice wave



Copyright 2008: Richard H. Frenkiel

6

Once we decide to use digital coding, a number of practical problems have to be solved. We are going to send a long, continuous string of bits, and the receiver has to be able to find each one in the presence of noise. This requires a clock that tells the receiver when to look for each bit. We use the bits to create the clock and the clock to find the bits. That seems illogical, but it isn’t. The clock depends on a long-term look at the data. A synchronization process at the beginning, before the data is sent, helps create the clock. There needs to be enough clarity in the data to keep the clock going, but it can coast through short periods when the data is noisy.

In addition to those extra “overhead” bits that establish the clock, we need other overhead bits that tell the receiver when the “payload”— the data we really want-- starts and ends. Even with a good clock and initial synchronization circuits that can count bits very well, it’s possible to get out of sync over time, and if we slip our count by one bit for

those 3-bit speech samples, every number will be wrong. To prevent this we define a total length for a “word” or “packet” of data, after which we resynchronize and start a new word.

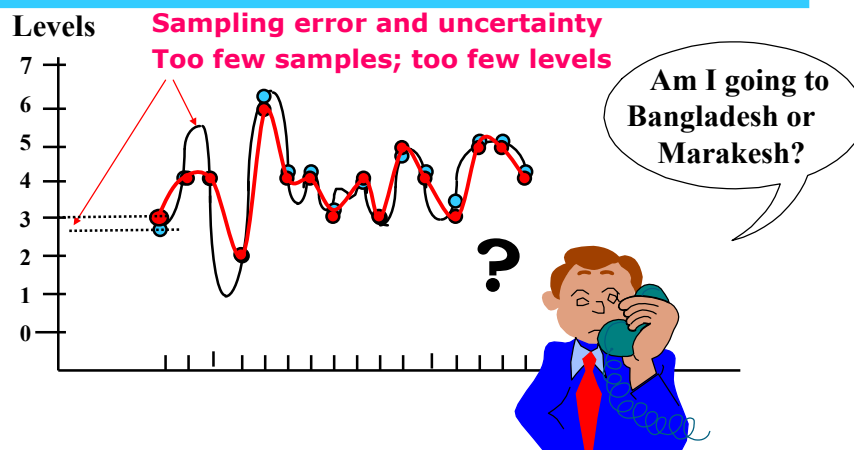
Within each word, we have to define how the “payload” bits are coded (are they 8 bit symbols for letters and numbers, or 3-bit speech samples as in the figure above? Whatever they are, we need to know how they were coded, so we can decode them.

Do we need to know if we’ve made an error? Must we correct it? We can identify and correct errors, but that takes extra overhead bits and a more complex receiver, so we will have trade-offs to evaluate. Overhead bits can represent just a few percent of the total information, or more than 50% if heavy coding is required for error detection and correction. Since more information requires wider channels, channels with heavy coding for error detection and correction will be wider, but they will tolerate more interference. If we were to double the channel bandwidth with error detection and correction, but were then able to use a three- or four-cell reuse pattern instead of a 7-cell reuse pattern, we would end up with about the same number of users per cell.

One of the most important questions in creating a digital standard is deciding how many samples per second to take, and how many levels to use to represent each sample. We never have enough information to reproduce the wave exactly, but we want it to be “close enough.” Returning to the first example (1000 samples per second and 8 levels, as shown on slide 6), we see that the first sample had a level of about 2.7, but we only had 8 possible levels to choose from, so we rounded it off to a “3”. As a result, the receiver can’t know it was originally a “2.7”, so that information has been lost forever. This is called quantizing error. We also have no information between the first and second sample, or between the second and third sample, and so on. This produces “sampling error.” As illustrated on slide 8, if we try to reconstruct the original signal with too much quantizing error and sampling error, our reconstruction is very uncertain, and therefore provides poor quality. We might not recognize the voice of the speaker, or understand what the speaker is saying.

Reconstructing the voice

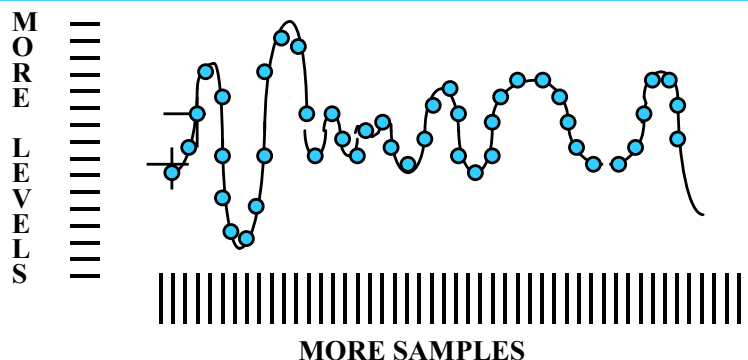
(But not too well)



There will always be some difference between the original signal and the one we reconstruct, but the more samples we take, and the more levels we use, the closer we come to the original signal. On the other hand, the more often we sample and the more levels we use, the more bits we have to send and the wider our channel gets.

Sending more accurate information

More samples; more levels per sample



“Telephone Quality” voice uses 8000 samples/sec and 256 levels
 $8000 \text{ samples/sec} \times 8 \text{ bits/sample} = \underline{64,000 \text{ Bits per second!}}$

On slide 9, we’ve increased the number of samples and the number of levels, and now it’s clear that our samples allow us to reconstruct the signal very well. Using listening tests that compared the original voice to the reproduced voice, Bell System engineers determined that for good quality in the telephone network, they needed 8000 samples per second and 256 levels (8 bits) per sample, so digital voice needed 8000 times 8, or 64,000 bits/second. This became the standard for digital telephony.

We’ve now spent some time on digital information and digital voice, but there remains an important question—*Why do we care?* Why would we want to transmit a voice as bits in the first place?

There are a number of reasons. The first is that the quality of digital signals is easier to preserve. There is noise in every transmission medium, and as the signal travels and gets weaker, the noise becomes more noticeable. Once the voice signal has become noisy, it can’t be restored very well, because it’s impossible to tell what it used to look like. Circuits have a hard time separating the voice signal from the noise, so the noise becomes part of the signal. To avoid this, the signal has to be amplified after some distance, before the quality degrades noticeably. Preserving voice quality over long distances became a real art in the telephone network, and Bell Labs engineers were justly proud of the sophisticated technology they developed for this purpose.

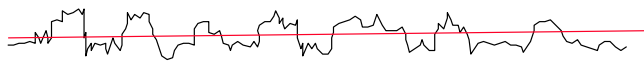
Maintaining quality is easier in digital systems, as long as the noise isn't too great. You only need to be able to tell that the original bit was a 1 or a zero. Instead of amplifying it, with any accumulated noise, you can just produce a new one or zero with no noise at all [slide 10]. A good analogy is the deterioration in old fashioned records when they got scratched, versus a digital version like a music CD or MP3. Once a record is scratched, the problem is permanent. CDs can be destroyed as well, but it takes a lot more deterioration. They tend to be perfect until a significant amount of wear is present, and can be reproduced any number of times without losing quality in the copies.

Why bother with digital telephony?

Reason 1: reconstructing the noisy signal



Original information



After transmission (added noise)



After reconstruction (good as new)

Noise on an analog signal is hard to remove

There were other advantages to digital as well. With those extra “overhead bits,” it’s possible to identify and even to correct the errors that do occur. This is important in sending important data over radio channels. No matter what you do in radio, there can always be those huge drops in signal level over a few inches (Rayleigh fading caused by the many transmission paths) that will cause errors. An error might cause a click that we can ignore in a voice conversation, but as we start to send important data over radio channels it is necessary for some of it (like a financial transaction) be perfect. Additionally, when we do start using the same channel for voice and other types of information, it’s easier if it’s all digital. And finally, progress in digital chips in terms of speed, capacity and cost has been faster than in analog chips, partly because they are more generic and easier to organize and partly because computers have driven that progress.

In 1980, these factors were not yet clear and dominant. A forward-looking person might have seen that the transition to digital in the telephone network was inevitable, and many did (even at AT&T) but AT&T had a lot of analog equipment that would be expensive to replace, and it worked well. Why throw it away? Bell Labs engineers worked on digital transmission in the long-distance network, but the company continued to argue that analog was “just as good”. There are those, even today, who claim that music CDs, which use many samples and many levels, don’t sound as good as the old analog record masters, though listening tests suggest that this is an illusion. Humans

cannot tell analog music from a good digital reproduction if enough samples are taken, and if enough bits per sample are used. Music quality CDs used 44,000 samples per second and 16 bits per sample, creating an information rate of 700,000 bits per second, while telephone quality voice used only 64,000 bits per second.

For the telephone network, communication using pulses of light through fibers of glass changed everything. A single glass fiber carrying pulses of light represents a huge bandwidth— more than all our radio channels put together. Glass is cheaper than copper wire as well, and once the fiber was installed, bits became almost free, and advances in circuit technology delivered dramatic increases in bit-rate. The capacity of a single fiber went from 50 million bits/sec in 1977, to 1 trillion bits/sec in 1997, and continued to increase.

At the same time, the FCC allowed increasing competition in the long distance network, and when competitors like MCI began to install fiber, AT&T was forced to accelerate its plans to modernize its network. It abandoned the argument that digital and analog were equally good, and wrote off \$6B of old analog equipment. The telephone network was soon totally digital and fiber-based.

Those free bits also allowed the Internet to be created, and to grow in capacity at unheard of rates. Today we download hundreds of megabits— even gigabits— for a few dollars a month. In fact, the dot-com mania of the late 1990s created such enthusiasm for fiber that by the end of the century too much got installed. This contributed to the crash of the stock market at the end of the century, but the same technology allowed the Internet to seem to be “infinite and free”.

In mobile radio systems, however, the issues were very different. We have seen that mobile radio channels operating in noisy, variable environments have limited information capacity. As we saw in the last lecture, when we try to put more information through a radio channel, the channel has to get wider. In fact, 2G cellular channel could carry only about 1.5 bits/sec for each Hz of bandwidth and 3G/4G technology has only increased radio capacity by a factor of about 5.

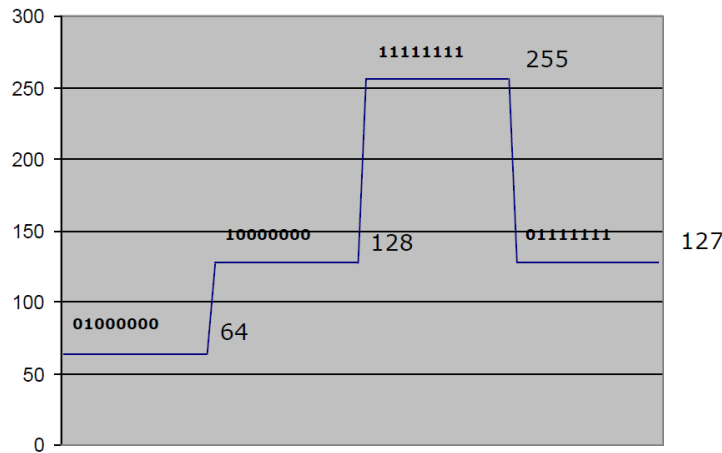
Those 30 KHz cellular channels could only carry about 45,000 bits per second, and since telephone-quality voice needed 64,000 bits/sec, the cellular channel would have had to get wider just to carry the same voice quality conversation digitally. A digital cellular system would have required channels wider than 30 KHz, and the result would have been fewer channels per cell. For that reason a digital cellular system would have cost more than an analog cellular system (for the same voice quality), so there was no reason to consider digital cellular before 1990.

At this point, it is useful to mention that under some conditions of low noise and interference, a digital radio channel can carry more than 1.5 bits/sec per Hz of bandwidth. Periodically, we read about some “magic” radio technology that allows “ten times the bit rate on a channel.” Can this be possible? Theoretically it can, because on a very low-noise channel, one “symbol” can be seen to have many different levels, so each “symbol” can carry multiple bits of information. Slide 15 shows an extreme example in which each “symbol” can be one of 256 different voltage levels (from level 0 to level 255). In the case shown on the slide, the first symbol has a level of 64, the second is 128, the third is 256, and the fourth is 127. Because there are 256 different possible levels, each of those symbols can be thought of as representing 8 bits. The first symbol shown, at level 64, would represent **01000000**. The second symbol, at level 128, could represent **10000000**. Level 255 would be **11111111** and level 127 would be

01111111. By sending (say) 1000 of these symbols per second we would be sending 8000 bits per second, so our radio channel would carry more bits per second without getting wider.

What about higher data rates with radio?

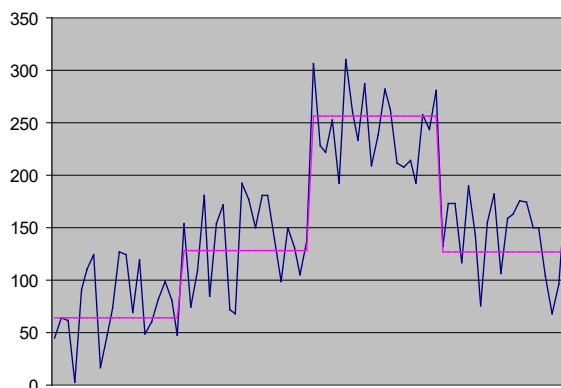
"We use symbols with many levels to get many times the data-rate"



15

The problem is that our cellular channel is very noisy, because we have interference and that multi-path Rayleigh fading that makes the level change dramatically every few inches. Instead of that nice clear picture we saw in the last slide, the demodulated signal could look more like slide 16. We can tell the third symbol is a different level from the first, and maybe we can tell that the second symbol is different from the first, but we certainly can't tell that the fourth symbol is different from the second. We can't get rid of the interference because we need as much channel reuse as possible. We can mitigate the Rayleigh fading with expensive techniques, but we can't get rid of it completely. As a result, we can use our basic binary signal with two levels, or at most a signal with 4 levels. As a practical matter, our cellular channel can only carry about 1.5 bits/second/Hz.

Small differences can not be distinguished on the noisy mobile radio channel



Thus, the key technology that would bring digital voice to cellular had to be different from the “infinite and free bits” that fiber brought to communications. In radio, the number of bits per second through a channel wasn’t going to improve by millions of times. The key in cellular was a technology called data compression that reduced the number of bits needed to represent the voice, with only a small reduction in quality. It’s a complex technology to work with, but we can understand the general logic of it through some simple examples.

Consider the lowly fax. To create a fax of a printed page, it was necessary to define whether every pixel on the page was it black or white. If you added it all up, there were millions of bits per page, even if the page was blank. With a little planning, however, the fax could send a message like “the pixels in this area are all black,” or “the pixels in this area are all white.” Using that kind of information, we could describe the page with fewer bits.

The same logic applied to photos or TV, where a whole area might be the same color. Moreover, in TV the whole screen is refreshed 60 times a second, but in many cases little has changed, so we can simply describe the changes.

Speech and music also have characteristics that allow compression. For example, the basic frequency of someone’s speech is fairly predictable, and the way the human mouth shapes sounds is fairly consistent, so smart algorithms can describe the voice with fewer bits than it takes to exactly describe an analog signal we know nothing about. Scientists knew how to do this in the 1980s, but the compression process involves complex calculations that must be done quickly (because a conversation must be sent as the person is speaking) and at low cost (because telephones are consumer products).

The breakthrough for digital radio was a computing chip called the Digital Signal Processor, or DSP. The DSP could be programmed to do a limited number of different calculations, but to do them very fast and at low cost. It became possible to design a low cost cell phone that could reduce the number of bits for nice sounding speech from 64,000 bits/sec to 8,000-12,000 bits/sec. Rates as low as 4000 bits/sec were possible, but at those rates the voice started to sound robotic, and you couldn’t always recognize the speaker.

In television, compression technology allowed digital channels to replace analog channels with less bandwidth per channel (as has just occurred). The shifting of TV channels to digital cleared spectrum for that new 700 MHz auction and made HD TV practical. The fact that fewer bits were needed also made information storage cheaper. In music, this led to the MP3 player, which could reduce the bit-rate from the 700 Kbps of CDs to about 160 Kbps with about the same sound quality. In photography, it led to the digital camera and JPEG compression standards.

With voice compression, digital voice suddenly became a way to increase the capacity of a cellular system. That 30 KHz channel could still only support about 45,000 bits/sec, but now we could produce a nice sounding voice with only about 10,000 bits/sec. Even with overhead bits, it was possible to replace that 30 KHz analog cellular channel with a digital channel that carried three conversations. That meant that by “going digital” with voice compression, service providers could triple the number of call-minutes in their million dollar cells! After some bitter fights about the way the radios would work (CDMA vs. TDMA) cellular operators would soon replace all their old analog radios with digital

ones. [By contrast, in the wireline network, voice still used 64,000 bits per second, because fiber made bits so cheap there was no reason to use voice compression.]

As we discussed earlier (in Lecture 9), the way the bits were coded onto the digital radio channel for digital cellular could be achieved in different ways, and in the 1990s there were huge controversies over which digital radio technology was best for second generation (2G) cellular. The simplest approach was called TDMA, in which three users would share each 30 KHz channel. For a few milliseconds, the channel would carry bits for the first user, then for the second, and then for the third, then back to the first, and so on. The bits for any call could be selected and put back together into a single voice at the receiver. With each of the old analog 30 KHz channels now carrying three calls, it was easy to see that capacity was tripled.

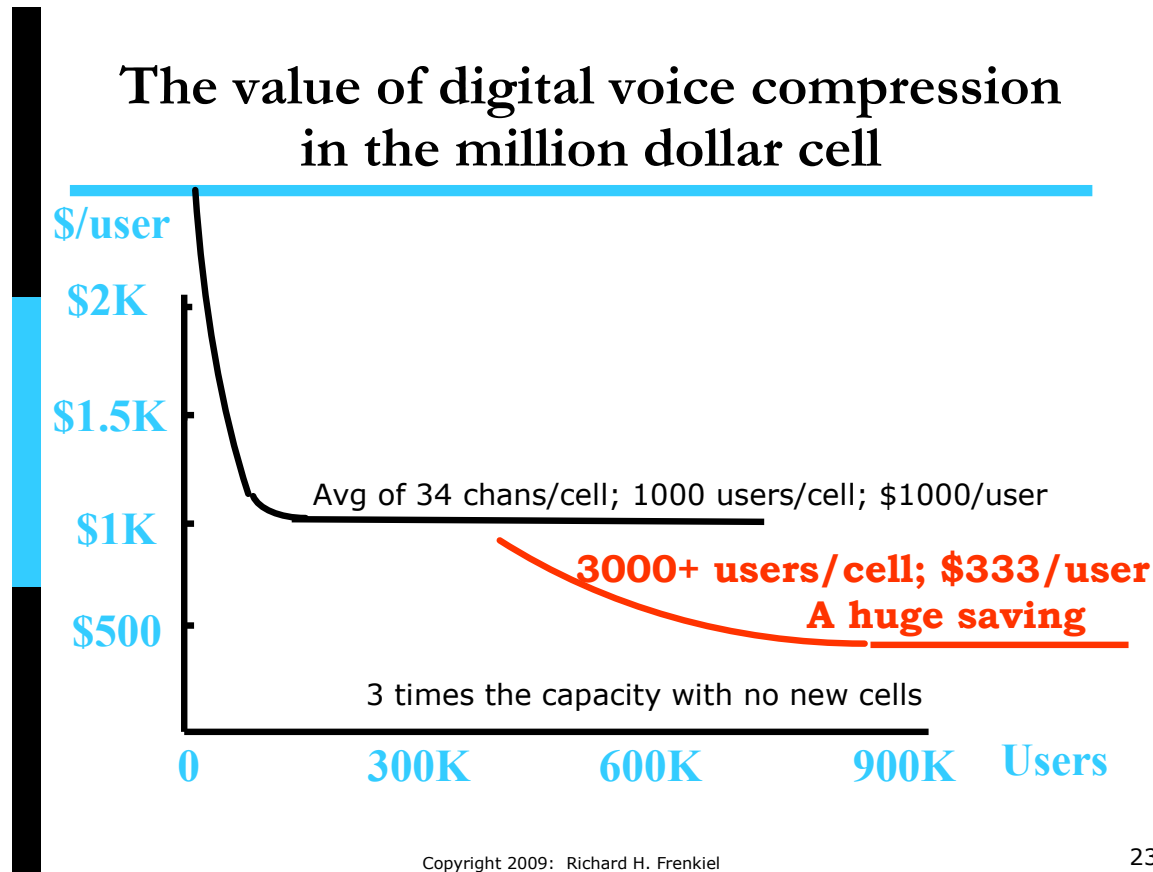
CDMA was much more complicated. It expanded every bit into a long code, so that the channel had to be much wider. In fact, the channel became more than 40 times wider (1.25 MHz). A 25 MHz cellular system would have only 10 2-way CDMA channels (rather than 400 of the old analog channels), but because the coding was very resistant to interference, some number of users in the same cell could share the same CDMA channel, and the channel could be reused in every cell.

From an information theory point of view, both these methods can be argued to be equal in terms of the information they can carry in a cell, but each could claim some additional efficiencies based on specific attributes of a cellular voice system. Because the arguments (and the technology) were complex, these often extravagant claims were hard to prove or disprove. CDMA initially claimed to have 20-30 times the capacity of TDMA, which was false but hard to disprove. The uncertainty worried Service providers in the US and delayed the implementation of second generation “2G” cellular systems. As previously mentioned, in Europe the GSM standard focused less on capacity and more on the seamless calling features called “personal communications”. By the late 1990s, operators in the US began to install both TDMA and CDMA systems (AT&T chose TDMA and Verizon chose CDMA). It is likely that for any given cell, CDMA and TDMA would have the same capacity. For complicated reasons that we will not discuss, CDMA may ultimately have provided more capacity for voice but not for data in 3G/4G.

In summary, most of the increase in voice capacity in 2G was not that the system could carry more bits per second, but that it required fewer ($1/3$) the bits per second for a voice call. In the first two decades of cellular (1G and 2G), while the capacity of fiber grew tremendously, the information capacity of a cellular channel remained the same (1.5 bits per second per Hz). Voice compression increased capacity by requiring fewer bits/second for a voice call. The first increase of the information capacity of a channel came with 3G and 4G and was rather modest (about 5x). As we will see in future lectures, the huge difference between the dramatic (and dramatically increasing) capacity of fiber and the limited (and almost unchanging) capacity of cellular channels has posed a significant problem for cellular providers. Users have been conditioned to see the fiber-based Internet as “infinite and free,” and they have carried that expectation to Internet access through cellular systems, whose 3G/4G access to the Internet is noticeably finite and rather expensive.

Returning to 2G, we find a more pleasant picture. The existing systems did benefit from the 3x improvement in voice capacity that was created by voice compression. In the US, the transition to 2G was made within existing systems, with huge numbers of

existing analog customers. This created logistical problems. It was necessary to convert the channels to digital gradually, over a few years, putting new users on digital channels and gradually moving the old users to digital as well. Over time, however, this promised to allow three times as many users without adding any new cells. Revenues would triple with no new base stations and SG&A expenses per user would fall. The future was bright.



As we have mentioned in past readings, what actually happened was unexpected. Driven by the competitive desire to attract users from other systems, AT&T offered “packages” of many minutes of talking for a lower price. Other service providers responded with bigger, cheaper packages, and the benefits of voice compression went to the users (more minutes) rather than the providers (more profit). This seems to be common in the world of corporate competition.