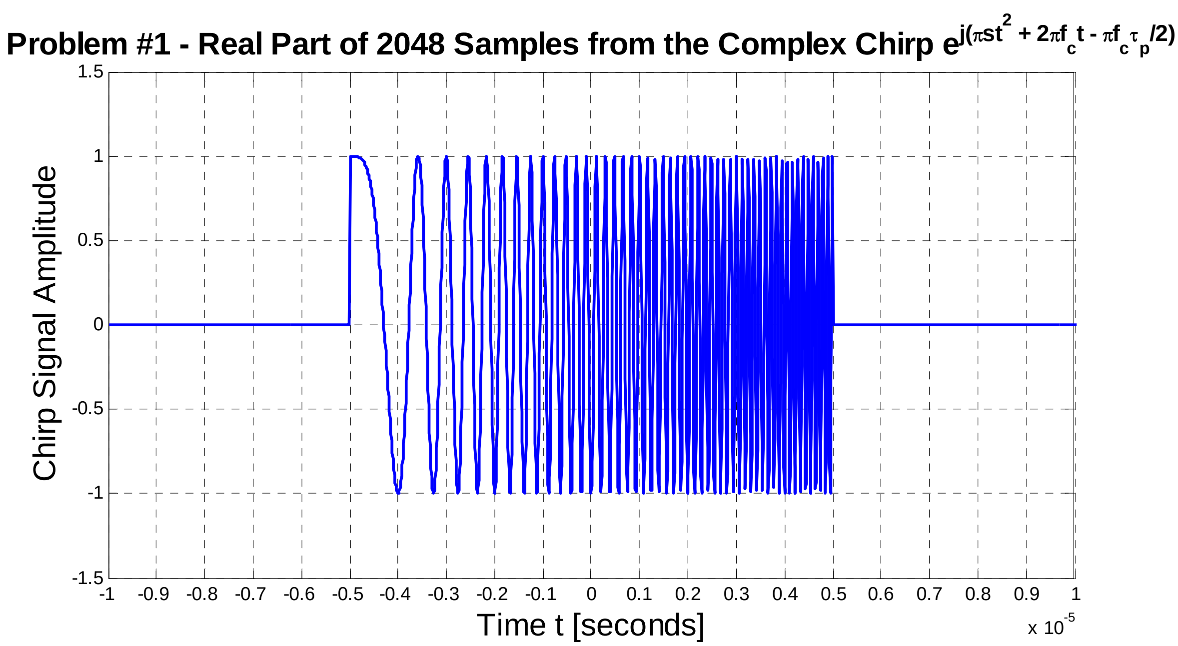
Christopher Tsai
January 24, 2007
EE 355 – Imaging Radar and Its Applications

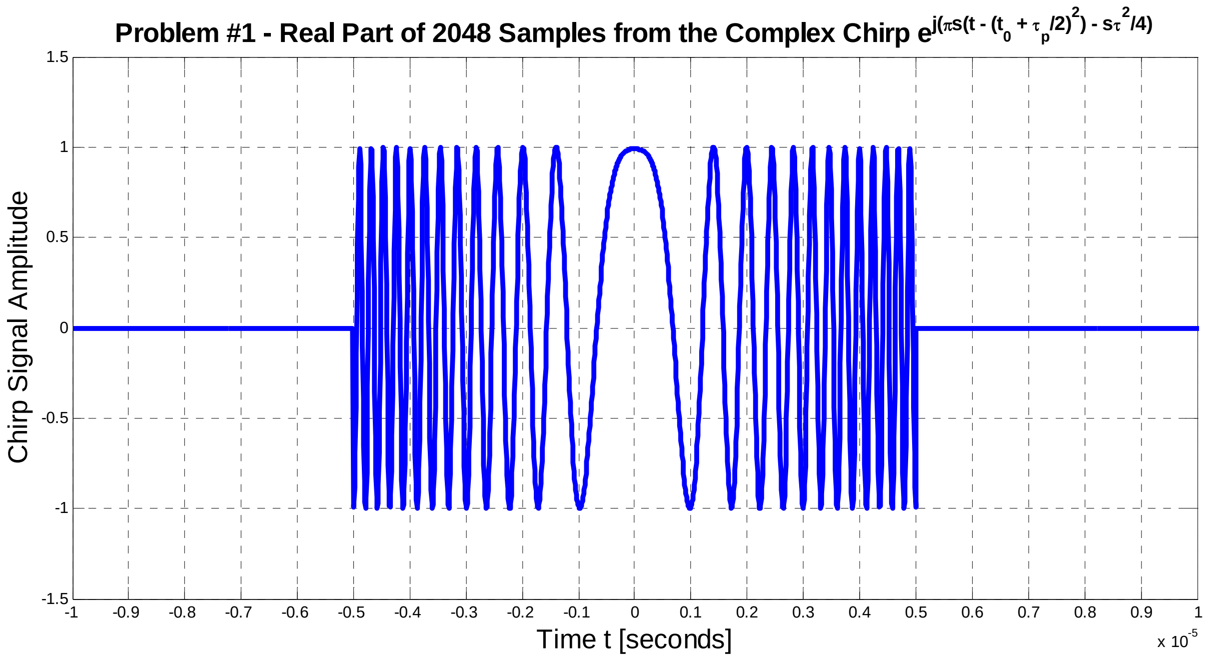
Problem Set II
The Chirp Pulse and Range Compression

Problem #1 – Chirp Compression

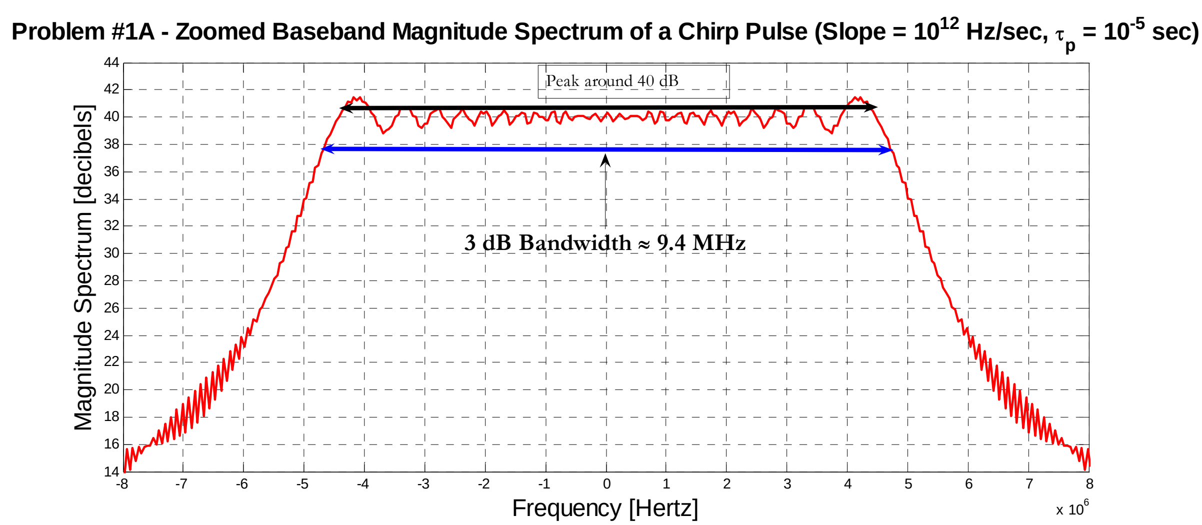
By default, we center our chirp pulse around the time origin (t = 0); each side of the time origin contains an equal number of chirp samples as well as an equal number of zero samples, which we must add as padding for our total chirp signal to reach 2048 samples. However, we can center the frequency of our complex chirp signal at zero or at a positive value. Thus, the `GenerateChirp` function coded in Matlab offers a user-input parameter specifying the orientation of the center frequency. To visualize the real chirp signal in time, we linearly sweep a set of positive frequencies passing through a nonzero center frequency:



The real chirp pulse begins at zero frequency and sweeps linearly to the maximum frequency, which we compute as $s\tau = 10^7$ Hz = 10 MHz. Unfortunately, the sweep beginning at zero frequency necessarily contains a one-sided spectrum, which we often need to shift to baseband for azimuth processing. Thus, we might find it more convenient to generate a complex-valued chirp signal with zero center frequency, whose real part might appear odd but whose spectrum is now centered around baseband (DC). In order to visualize this complex chirp in time, we must extract its real component, which looks strange because it seems like a frequency downsweep followed by a frequency upsweep. This symmetry actually stems from the spectral symmetry; by centering the spectrum around a center frequency of zero, we introduce negative frequencies that appear in reality as high frequencies approaching low frequencies:



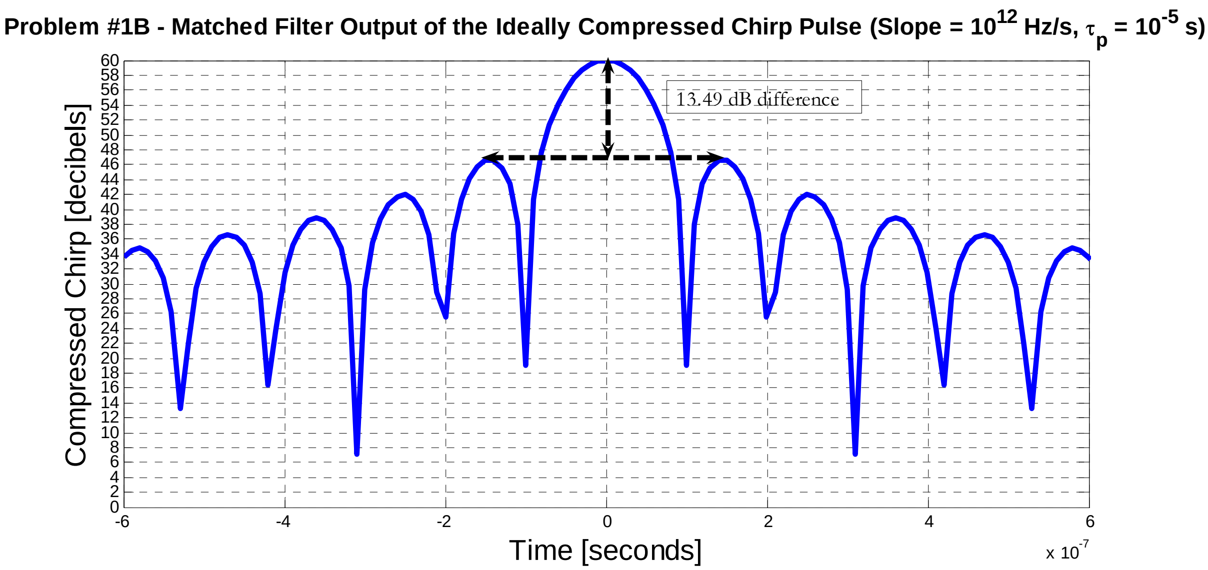
Despite its strange appearance, we do not care about the signal in time once we render the chirp complex; instead, we plot its spectrum on a logarithmic (decibel or dB) scale, from which we immediately see the benefits of our zero center frequency: a symmetric centered magnitude spectrum:



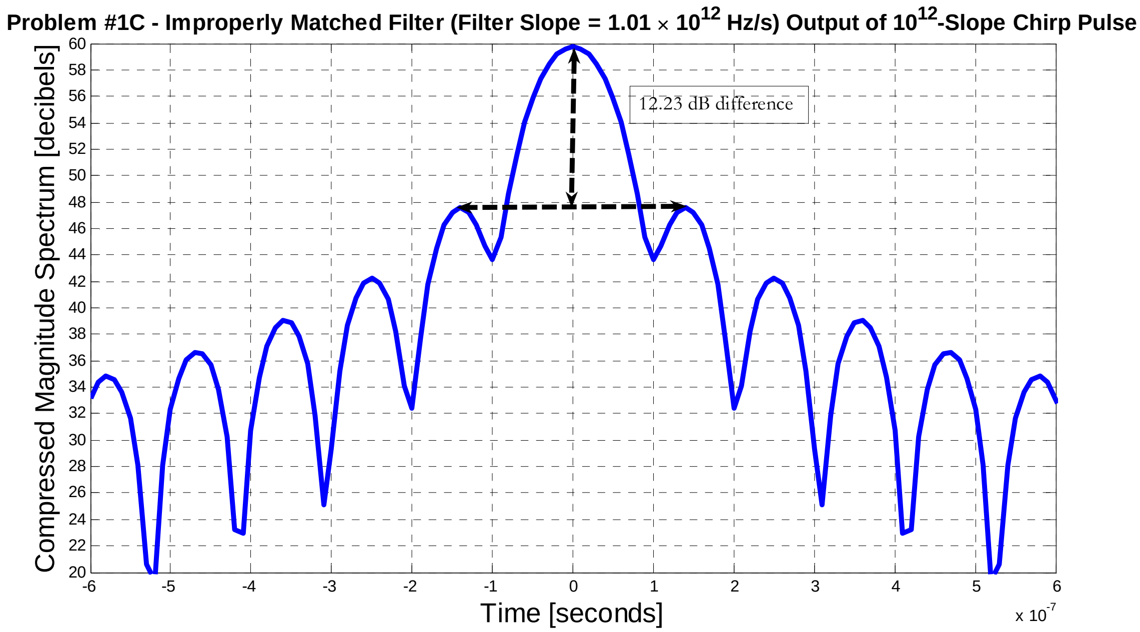
The centered spectrum reveals significant (> 37 dB) frequency content up to approximately 4.7 MHz on either side of the zero center frequency, so the 3-dB bandwidth is approximately 9.4 MHz from inspection. This bandwidth value is essentially 10 MHz, indicating that our generated chirp, indeed, sweeps across $s\tau$ =

10^7 Hz = 10 MHz frequencies, with approximately equal (flat) spectral content in frequency in the sweep range. Furthermore, the bandwidth remains approximately 10 MHz whether or not we center frequency at zero, meaning that the precise form of chirp we generate and the spectral shift of a chirp with nonzero center frequency have no effect on the bandwidth, which remains the same regardless of where we move our chirp spectrum. For spectral analysis and processing, we prefer zero carrier frequency, so the centered spectrum above will suffice.

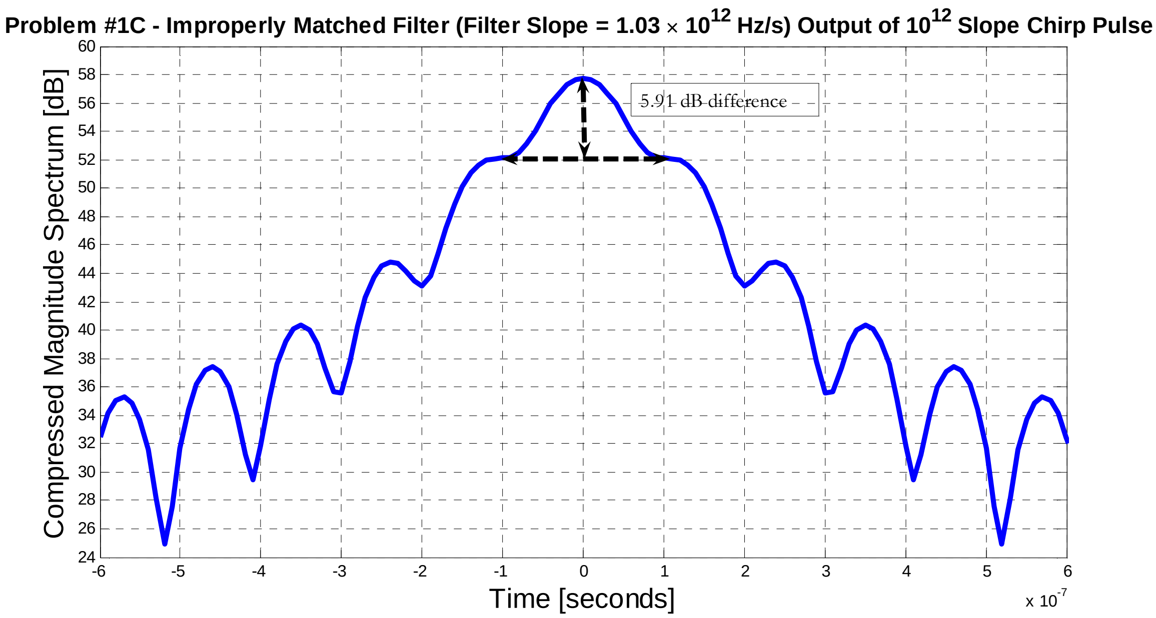
Compressing the chirp with a perfect reference signal (our matched filter) through correlation in time or conjugate multiplication in frequency, we obtain the following compressed pulse as the matched filter output:



The compressed pulse – the main lobe of our matched filter output – has much shorter duration than the originally generated (10 μs) chirp pulse, but we also see sidelobes surrounding the main lobe. The first sidelobes – the greatest threats to detection – peak at 13.49 dB below the main lobe maximum; in other words, the main lobe pulse maximum is approximately 4.72 times larger than the undesirable sidelobe, meaning that the chirp pulse still stands out considerably from the matched filter artifact replicas. However, this simulated output assumes that our reference signal perfectly matches the transmitted chirp. If, instead, we estimate the reference chirp slope incorrectly, creating instead a reference signal with slightly higher slope (1.01×10^{12} Hz/s), then our compressed pulse also degrades in quality to reflect the imperfect matching:



The main lobe still stands considerably apart from the sidelobes, but the difference in peak amplitude has decreased to 12.23 dB difference; the main lobe is only 4.029 times greater than the first sidelobe. If we misrepresent the chirp slope with even greater error by correlating the original transmitted chirp with a matched filter reference signal with higher unmatched slope (1.03 x 10^{12} Hz/s), then the fact that our imperfect reference chirp fails to match the transmitted chirp results in an even broader main lobe with sidelobes much closer in peak amplitude:

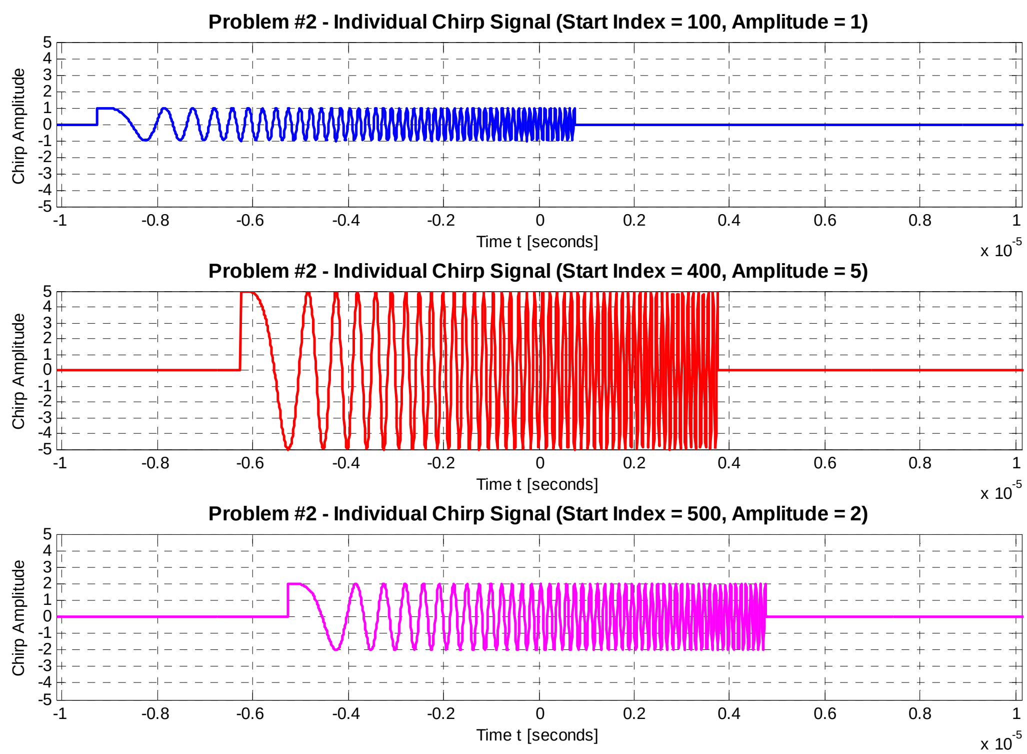


Notice that the first sidelobes have merged into the main lobe so that the chirp pulse, while still compressed to a fraction of its original pulse length, no longer has a protruding peak as desired. Consequently, with sidelobes of similar peak amplitude merged into its main lobe, the pulse would now smear some of the imaged features when convolved with surface brightness or point targets. The resulting compressed pulse still has a main lobe to sidelobe difference of 5.98 dB, meaning that the main lobe is approximately twice as large as the first sidelobes, but the relative significance of those sidelobes will prevent us from refining the pulse resolution, since convolution with any surface feature will spread the feature across the vicinity defined by the sidelobes, with possible replicas of nearby scatterers interfering with the main target we are trying to image with the main lobe.

Thus, as our reference signal deviates further and further from the actual transmitted pulse – as our matched filter grows more and more unmatched due to improper chirp generation, slope drift, or slope estimation errors – our ultimate compressed pulse's sidelobes protrude with more and more prominence, degrading resolution and making precise imaging more and more difficult. It is therefore important to match our reference chirp to the transmitted chirp as closely as possible, since we want as prominent and protruding a main lobe as possible in the compressed pulse. Any sidelobes that rise comparably high as the main lobes could produce unwanted replication or cancellation in the final image of a scattering target.

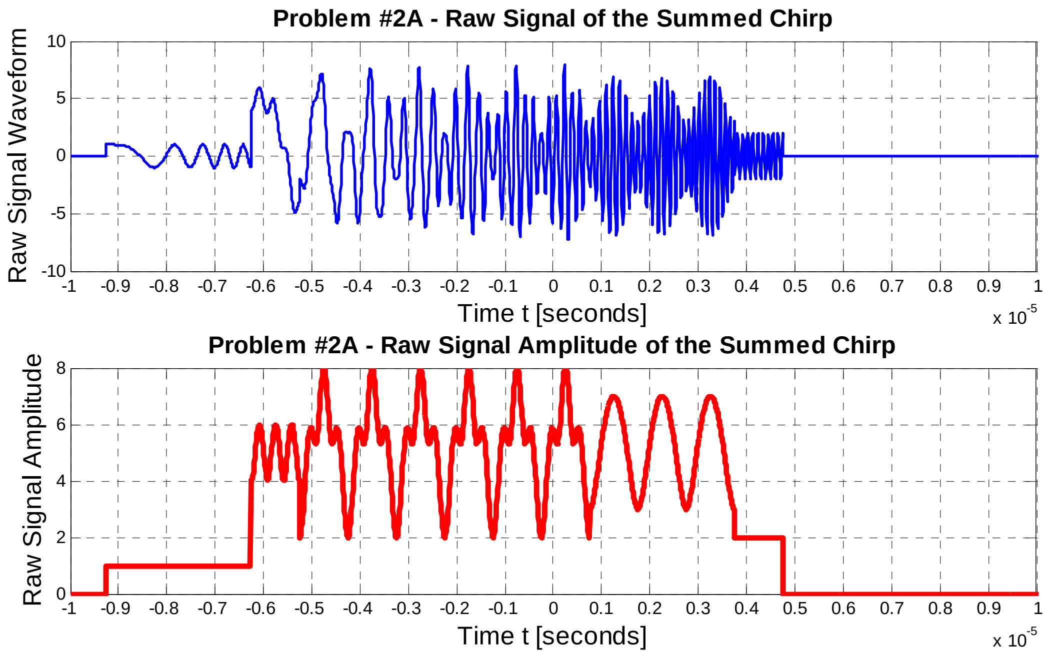
Problem #2 – Separating Multiple Chirps

In generating the three individual chirp pulses that contribute to the return signal, we pay no attention to the form of the chirp; since we are not plotting the spectrum, we never enter the frequency domain and therefore care not about the center frequencies of the chirps. Thus, we linearly sweep each of our three chirps from zero frequency to the maximum frequency $s\tau = 10^7$ Hz = 10 MHz, and stagger them accordingly in the composite signal comprising the sum of three individual chirps of identical slope but differing amplitudes:

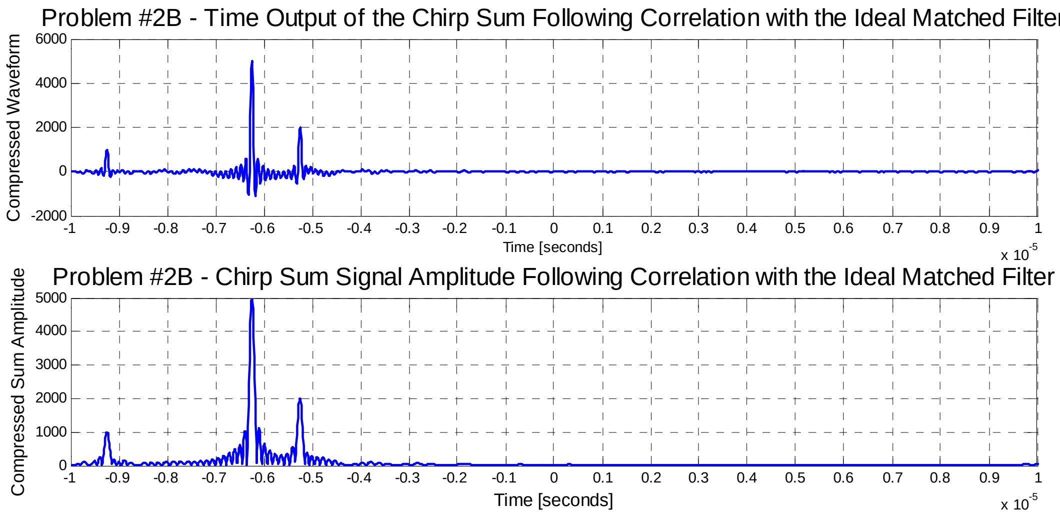


Such a signal could result from the interaction of a single transmitted chirp with three point scatterers of varying distance and reflectivity. The strongest signal (of amplitude 5) could result from reflection off the brightest target, while the two delays likely arise from more distant targets.

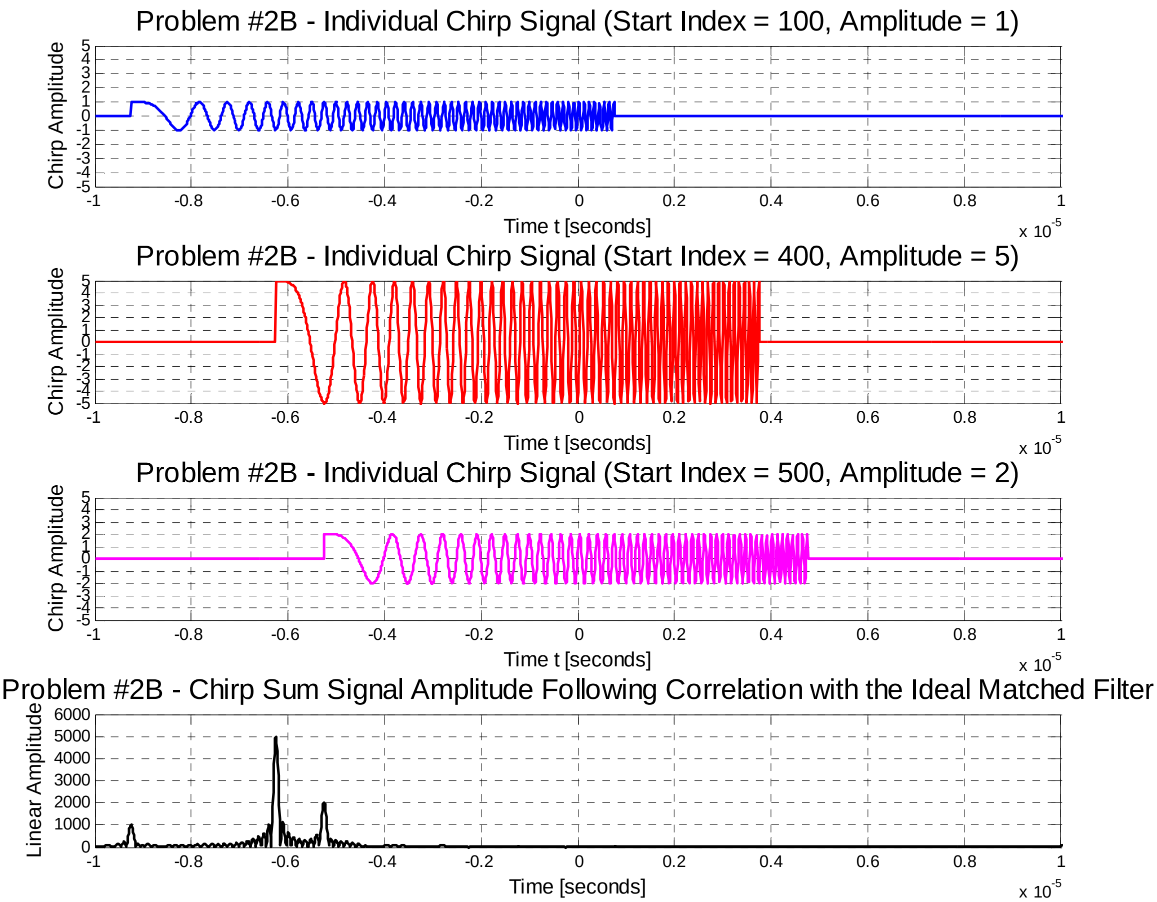
In developing a reference signal to compress the chirps and facilitate target detection, we need preserve only the key form or shape of the chirp; we therefore generate a chirp with the same slope (10^{12} Hz/sec) and pulse length (10^{-5} sec) but unit amplitude, so that we can infer the relative amplitudes of the individual scatters following matched filter correlation. Before applying this correlation, we observe the composite received signal:



The waveform itself looks crude and unintelligible, but the raw amplitude displays clear contributions from the three individual chirps. We can easily distinguish the first pulse from the initial flatline amplitude, while the last pulse manifests itself with the heightened plateau at the end. However, in between the two lone constant amplitudes, a mixture of signals results from the coincidence of multiple chirps. We can marginally deduce the presence of chirps from the peak amplitudes – 6 (= 1 + 5), 8 (= 1 + 5 + 2), 7 (=5 + 2) – but, if we did not know the individual chirp amplitudes, it would be impossible to determine whether we have three pulses of precise amplitudes 1, 5, and 2, or four, five, or six pulses of different amplitudes summing in the same way. In other words, while *we* can separate the chirps given the relative amplitudes of our three component signals, the raw signal composition is, in general, difficult or impossible to unravel because of the multitude of possibilities leading to the same composite sum. The pulses are too wide to know each one's individual location for certain, so we must compress them and improve resolution. Following compression, we see that the sum has shrunk noticeably into three near-impulses.



The three spikes that appear so prominently represent the locations where our reference chirp most closely resembles a part of the composite return signal. Because our summed signal comprises three chirps that behave identically to our matched filter, we see three spikes at the start locations of these three component chirps; the spikes occur at the *start* locations (as opposed to the midpoints or end times) of the individual chirps because we design our reference chirp pulse to begin at t = 0; hence, time offset during correlation exactly matches the position of the matched filter, so the index of maximal overlap will occur when the reference chirp has slid exactly the number of indices in the matched chirp's delay:

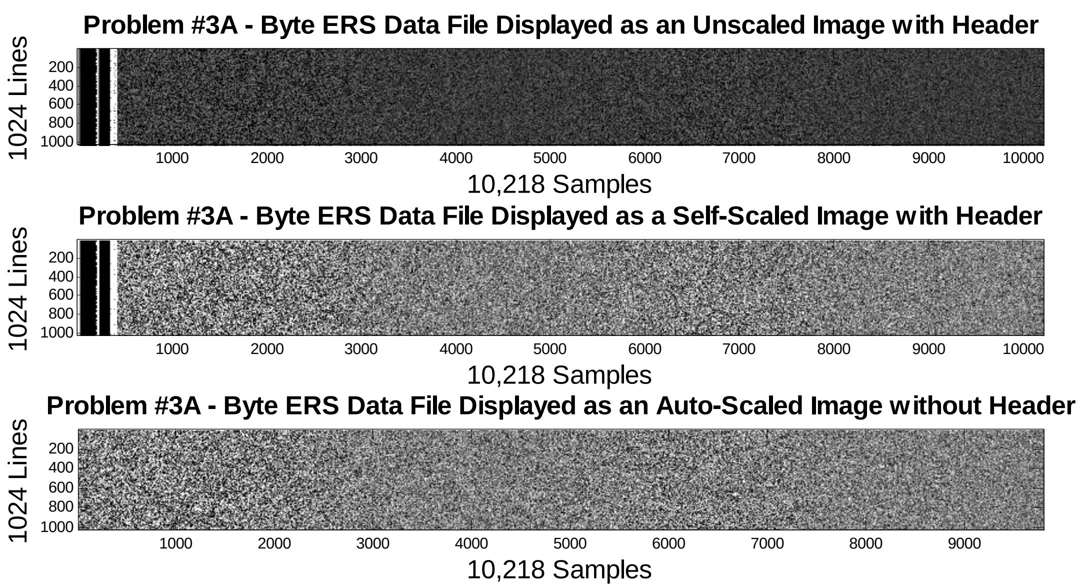


The resulting time-compressed signal clearly separates the three individual chirps in the summed signal, since their effective pulse widths have *compressed* down to skinny sinc functions by a factor of the time-bandwidth product. This phenomenon reveals pulse compression in its fully glory – seemingly inseparable chirp pulses have become so sharp and well-defined following compression that we no longer have any doubt about their exact start time and relative amplitude. Individual scatterer resolution skyrockets, and we can distinguish our targets effortlessly. We have decoupled resolution and pulse length, allowing us to transmit pulses of any length while still retaining individual target recoverability and fine resolution.

Furthermore, remark that the matched filter compressed pulses also have amplitudes that scale proportionally with the amplitudes of their respective individual chirp pulses. Thus, the correlation of our unit-amplitude reference signal reveals not only the starting times of the individual component chirps but also the relative amplitudes (scaled by 1000) of the individual pulses, allowing us to infer not only time information but also the brightness or reflectivity of our targets.

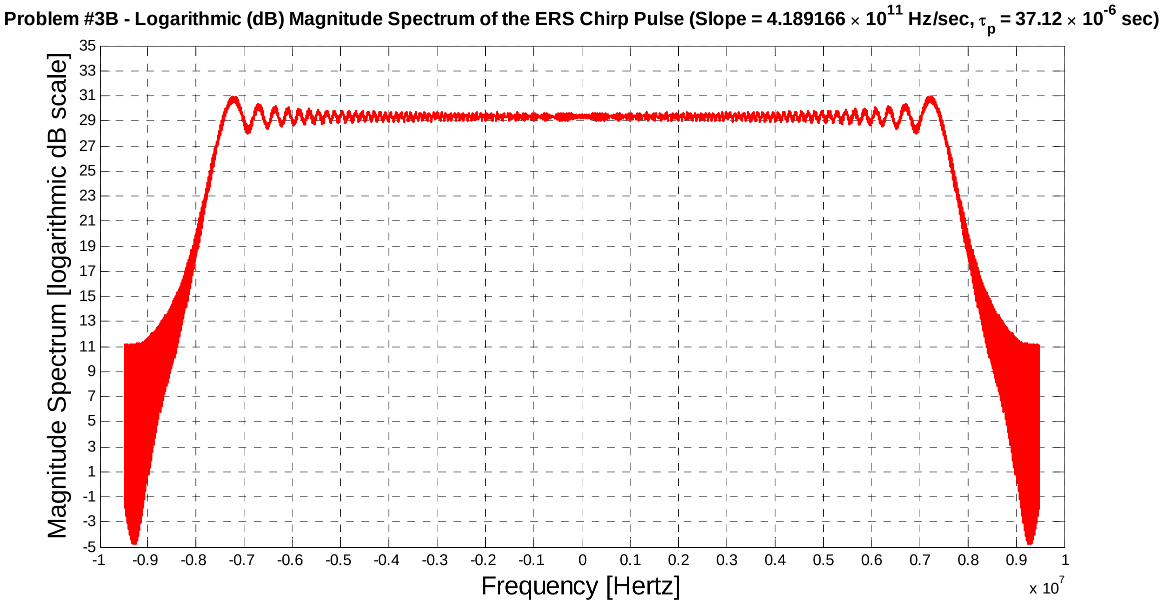
Problem #3 – Actual Data

We load the European Remote-Sensing Satellite (ERS) data into an array with 1024 lines and 10218 samples per line but must invert the matrix to align it properly for imaging. Within each line, data values alternate between real and imaginary components, the pairs of which we must join in order to form the complete complex data. Upon reading the data into Matlab, we first view the real and imaginary values on an unscaled image and then on a [0 31]-scaled image without complex combination:

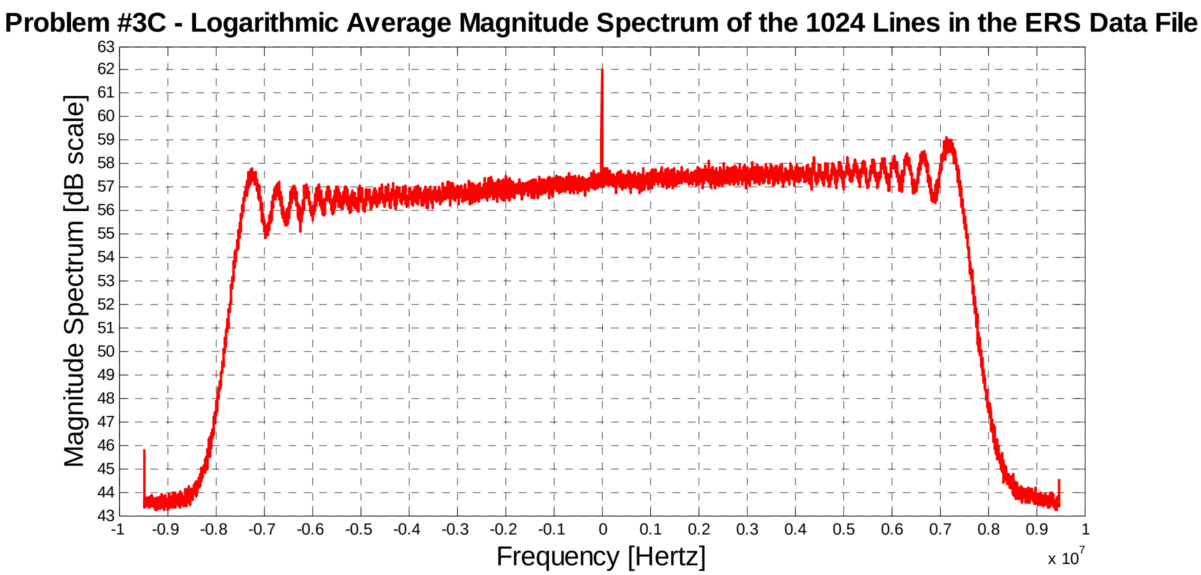


The data appears as seemingly unintelligible black and white speckle, but, because we have not combined real and imaginary entries in a meaningful way, we do not expect anything more than unidentifiable incoherent jabberwocky. We can, however, identify the 412 header bytes that precede each range line by their pitch-dark striped regularity on the left of the image. Noting that these bytes contain no actual image data, we can simply truncate the image matrix and remove them, as shown in the third subplot display.

Modeling the ERS chirp by their parameters, we generate a time-centered zero-center-frequency complex exponential chirp pulse to simulate the ideal chirp pulse. By centering the linear frequency modulation about zero carrier (zero center frequency), we can plot the centered magnitude spectrum:



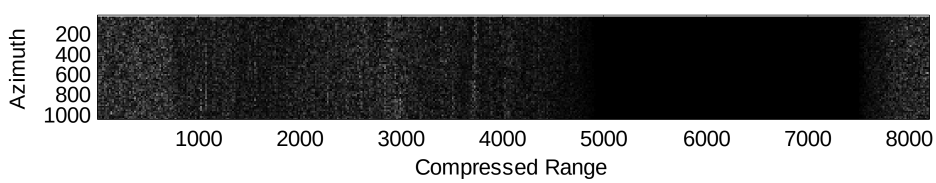
Employing Matlab's fast-Fourier transform algorithm `fft`, we compute the spectrum of each range line in the ERS data to juxtapose beside the theoretical magnitude spectrum. Fast-Fourier transforming along the rows and averaging the ERS data spectra across all range rows, we obtain the following average spectrum:



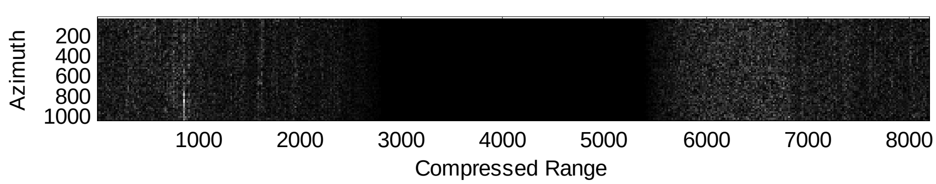
Comparing the actual data spectrum to the ideal chirp model spectrum, we see that both chirps, indeed, share zero center frequency; neither has a carrier. Furthermore, both spectra have the expected 3-dB bandwidth of approximately $s\tau \approx 15.5$ MHz. However, a few key features differ in the actual data spectrum. First of all, the peak magnitude of the ERS data spectrum hovers around 57 dB, nearly twice that of the theoretical chirp maximum amplitude, which peaks around 30 dB. Furthermore, the ERS data spectrum contains an imbalance of frequencies, with seemingly more power in the positive frequencies, as the uneven spectral weighting (see rising rectangle top) suggests. Thus, the actual ERS chirp used to obtain the data must not have swept through frequencies perfectly linearly, spending slightly more time at the end of the sweep than at the beginning; alternatively, the slope of the actual chirp could have deviated slightly during the sweep, or Doppler frequency shift could be at fault. Finally, we see an uncharacteristic spike or impulse at zero frequency, likely the artifact of a DC offset or nonzero average in the chirp signal; many communications systems apply a DC offset to ensure proper (unambiguous) positive envelope detection at the receiver. Any constant addition – whether it arises from deliberate receiver implementation or unintentionally from noise – would result in a sharp spike at or near zero frequency.

Our quest to perfect the range resolution leads us to pulse compression once again, as we seek to sharpen the peaks of our chirp pulse along the range direction and see surface features in higher definition. Since the general shape of our chirp model spectrum in Part (b.) seemed quite similar to the actual ERS data spectrum, we employ the ideal signal as a reference signal, which we correlate with each range line to obtain the compressed image. For speed consideration, we perform conjugate pointwise multiplication of the previously viewed reference spectrum and the spectrum of each range row instead of correlating in time. This range compression yields different images depending on the start time of our reference chirp:

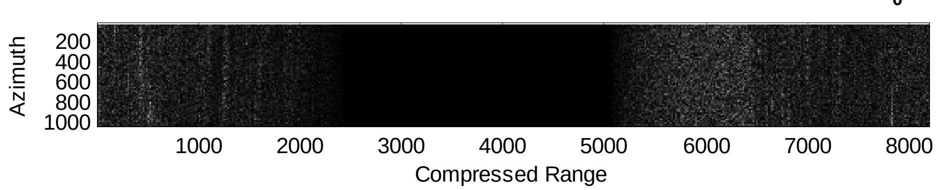
Problem #3D - ERS Image with Compressed Range and Unprocessed Azimuth (t_0 = -129.2985 μs)



Problem #3D - ERS Image with Compressed Range and Unprocessed Azimuth (t_0 = -18.56 μs)

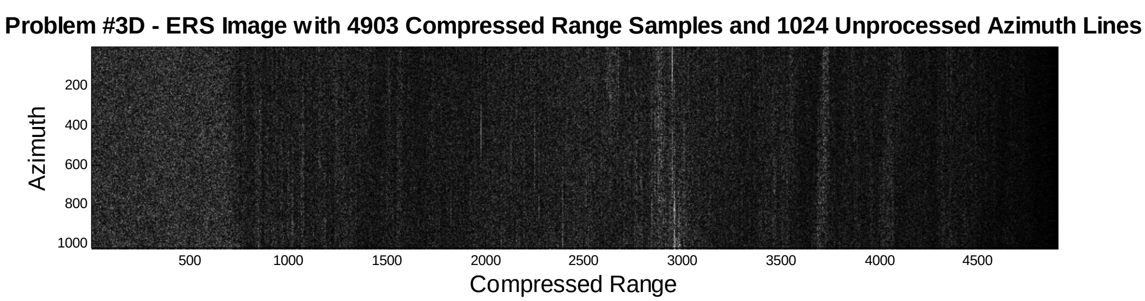


Problem #3D - ERS Image with Compressed Range and Unprocessed Azimuth (t_0 = 0 μs)



The upper image results from compression with a reference signal indexed at the beginning of the time array; because this reference signal positions the nonzero chirp samples at the left, we see a band of zeros near the right edge, where the zero-padding we performed on our chirp signal reappears as zero correlation, followed by some meaningless data resulting from residual correlation. The middle image employs a reference chirp centered about the time origin (t = 0), explaining the large number of zeros in the center of the strip; we would need to apply Matlab's `ifftshift` command to properly align the zeros with the original centered chirp pulse. The lower image begins the chirp at the time origin and sweeps toward positive time, much like the reference signal used to detect the individual chirps in Problem 2; here, however, such indexing means little since we aim not to detect the start times of signals, but rather to detect relative spatial position. For that purpose, positioning our chirp at the beginning of the array seems most logical, since we restrict the meaningful data to the first 4903 lines of the image and remove all spatial dependence on the arbitrary "time" values we assign to the chirp pulse.

All of these images comprise 8192 samples in the range direction, but only 4903 of these samples actually carry meaningful data, since the original ERS data comprised only 4903 complex data values. The remaining 3289 samples include the zeros used to pad the off-time of the chirp pulse, as well as meaningless correlation values between the chirp pulse edge and ERS data points. Thus, to avoid misinterpretation, we might truncate our range-compressed image after 4903 samples since we know the rest are merely artifacts of the difference between our chirp's fast-Fourier transform length and the number of samples in the ERS data:



Perceiving fine lines along the horizontal axis, we evidently see a lot of detail in the range direction, but remark that the azimuth (vertical axis) boasts no such fine resolution, as we have yet to perform azimuth processing. Range processing has successfully refined the range resolution, but azimuth brightness remains blurry and hazy.