Communication Channel Optimization

David A. Edwards[1] †, Pak-Wing Fok[1] , Richard Moore[2] , Brian Sutton[3] and Maxim Zyskin[4]

[1] *University of Delaware; Newark, DE, USA*
[2] *Society for Industrial and Applied Mathematics; Philadelphia, PA, USA*
[3] *Randolph-Macon College; Ashland, VA, USA*
[4] *Oxford University; Oxford, UK*

(*Communicated to* MIIR *on October 24, 2024*)

Study Group: Mathematical Problems in Industry Workshop, University of Vermont, June 25–29, 2024
Communicated by: Richard Moore
Industrial Partner: Raytheon RTX
Presenter: Ben Dolgin
Team Members: Taras Lakoba, University of Vermont; Noah Leedom, University of Vermont; Christopher Raymond, University of Delaware; Lawan Wijayasooriya, Georgia State University
Industrial Sector: Communications/Networks

Key Words: Differential phase shift keying; transmission encoding/decoding; numerical optimization; variational methods.

MSC2020 Codes: 78M30, 68P30, 65K10, 94A14

Summary

Transmission by laser is a cost-effective way to send signals to space vehicles. Signals are encoded so that zeroes and ones are recognized by different intensity levels at the receiver. The process should be optimized so that the gap in intensity between zeroes and ones is large enough that correct decoding will occur even in the presence of noise. Global and local optimization schemes are presented using numerical and analytical techniques. The results obtained produce wider gaps than those currently available.

1 Introduction

It is important to establish accurate communication with long-distance space vehicles (such as Mars orbiters). Unfortunately, standard radio-frequency transmission equipment takes up far too much volume and weight to be economical. A more cost-effective method is transmission of signals by laser, where fiber optics advances make the necessary equipment more lightweight.

† Corresponding Author: `dedwards@udel.edu`

Unfortunately, due to dispersion in the atmosphere, such signals require postprocessing to be interpreted correctly. In this manuscript, we discuss ways to optimize the encoding of the signal so that the decoding process will have a high probability of faithfully reproducing the transmission. In particular, signal bits of 0 and 1 are encoded to have different energy levels. Our goal is to design an algorithm such that even with noise added, one can determine with high probability whether a received energy level corresponds to a 0 or 1.

In §2 we mathematically describe the transmission device we are modeling and its three main components: the transmitter, the *etalon* demodulator, and the decoder. In §3 we demonstrate an algorithm that globally maximizes the gap in energy levels between a typical 0 bit and a typical 1 bit. In §4 we describe a local optimizer for the same problem both from a theoretical perspective and a more realistic perspective including device limitations. These results are largely numerical; in §§5 and 6 we present a variational approach which admits analytical solutions for simple cases.

2 Device Description

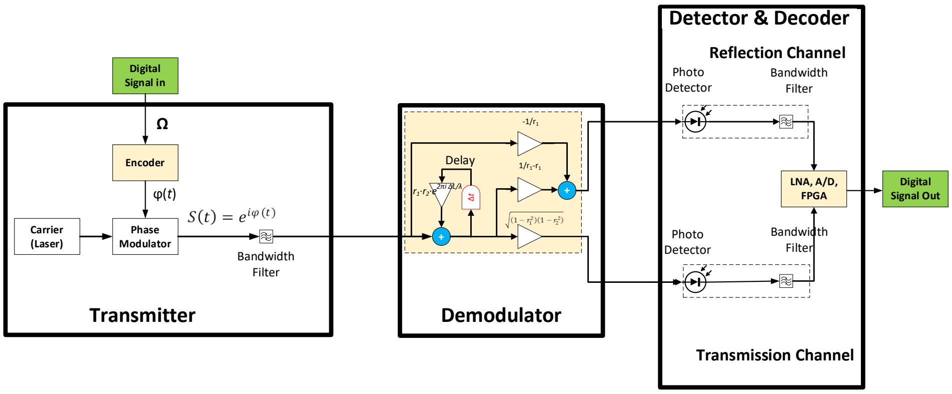


Figure 1. Schematic of transmission system.

Figure 1 shows a simplified schematic of the full processing system. We discuss each section in turn.

2.1 Encoder

In the first part of the device, a carrier wave is encoded to form a signal wave

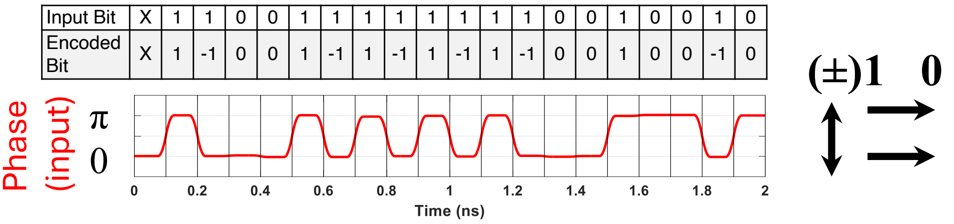
$$S(t) = e^{i\phi(t)}. \tag{2.1}$$

The jth bit is encoded in the time interval

$$[t_{j-1}, t_j], \qquad t_j = j\tau, \tag{2.2}$$

where τ is the *bit width.*

This encoding process is called *differential phase shift keying* (DPSK) [Binh, 2008], and has two nonintuitive properties. First of all, the binary bits comprising the signal are encoded indirectly through the signal phase ϕ, which in an ideal system alternates between 0 and π (see red curve in Fig. 2).



Input Bit	X	1	1	0	0	1	1	1	1	1	1	1	1	0	0	1	0	0	1	0
Encoded Bit	X	1	-1	0	0	1	-1	1	-1	1	-1	1	-1	0	0	1	0	0	-1	0

Figure 2. Typical phase encoding, $\tau = 0.1$ ns.

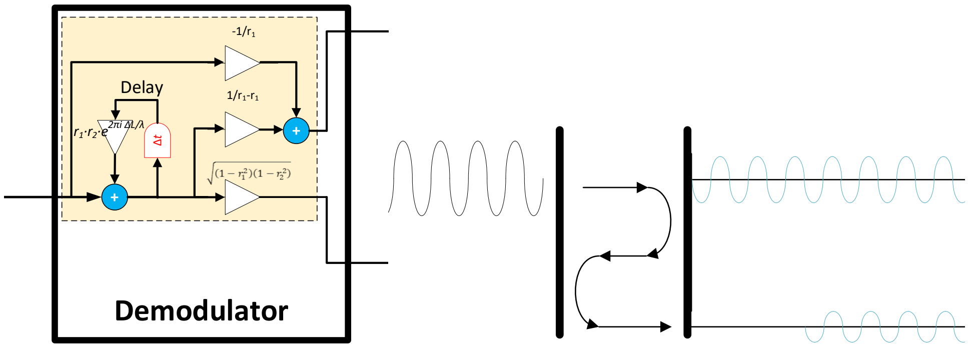
However, the two bit values 0 and 1 are not encoded by the phase itself. Rather, the jth bit is a “1” if the signal changes phase around t_{j-1}, and is “0” if it does not. See the input bit string and key in Fig. 2. However, the decoder interprets the signal differently if the phase jumps up rather than jumps down. So in some sections we will treat the encoded bit as 0 or ±1, depending on the direction of the change in phase.

We also highlight two important consequences of the encoding process:

(1)An encoded bit of 1 cannot be followed by another 1, since the phase can’t increase twice. Similarly, a -1 must not be followed by another -1. These rules constrain words in our ternary alphabet.

(2)Since $\phi = \pm\pi$ are the same, the first nonzero encoded bit can be either ±1, as long as the rule above is followed.

2.2 The Demodulator



-1/r₁
Delay
1/r₁-r₁
$r_1 \cdot r_2 \cdot e^{2\pi i \Delta L/\lambda}$
Δt
$\sqrt{(1-r_1^2)(1-r_2^2)}$
+
+
Demodulator

Figure 3. Demodulator. Left: schematic. Right: wave path.

The second part of the device is the etalon demodulator, a schematic of which is shown at left in Fig. 3. We ignore the top (reflected) channel and focus only on the lower (transmitted) channel. The wave enters the left side of the etalon (right of figure), which multiplies its amplitude by a factor of $\sqrt{1-r_1^2}$, where r_1 is the reflection coefficient of the entry surface.

The wave then bounces back and forth inside the etalon, making a round trip in time Δt. (An exit with one round trip is shown at right of Fig. 3.) Each trip reduces the amplitude by a factor of $r_1 r_2$ (representing reflection against each side—here r_2 is the reflection coefficient of the exit surface) and introduces a delay factor of $e^{i\delta}$. At each exit, the amplitude of the signal is multiplied by the factor $\sqrt{1-r_2^2}$, representing transmission through the exit surface.

Using these physical facts, we see that the *complex* amplitude $A(t)$ of the transmitted signal is given by

$$\begin{aligned} A(t) = \sqrt{1-r_1^2}\Bigg[&\sqrt{1-r_2^2}S(t) + \sqrt{1-r_2^2}r_1 r_2 e^{i\delta} S(t-\Delta t) \\ &+\sqrt{1-r_2^2}(r_1 r_2 e^{i\delta})^2 S(t-2\Delta t) + \cdots \Bigg], \end{aligned}$$

where we define t as the time when the original signal exits the etalon. Combining terms and simplifying the algebra, we obtain

$$A(t) = \sqrt{(1-r_1^2)(1-r_2^2)} \sum_{k=0}^{\infty} f_2^k S(t-k\Delta t), \quad f_2 = r_1 r_2 e^{i\delta}, \tag{2.3a}$$

$$= f_1 \sum_{k=0}^{\infty} f_2^k S(t-k\Delta t), \quad f_1 = \sqrt{(1-r_1^2)(1-r_2^2)}, \tag{2.3b}$$

$$A(t) = f_1 \sum_{k=0}^{\infty} f_2^k \mathcal{B}^k S(t), \qquad \mathcal{B}S(t) \equiv S(t-\Delta t), \tag{2.4}$$

where $\mathcal{B}$ is the backward shifting operator. We note from (2.1) that $|S(t)| = 1$ for all t; hence $|\mathcal{B}S| = |S|$. From Appendix A we see that r_1 and r_2 are less than 1; thus so is f_2. Therefore, the sum converges and we have

$$\begin{aligned} (\mathcal{I} - f_2\mathcal{B})A(t) &= f_1 S(t) \\ A(t) &= f_1 S(t) + f_2 A(t-\Delta t), \end{aligned} \tag{2.5}$$

where we have used the definition of $\mathcal{B}$.

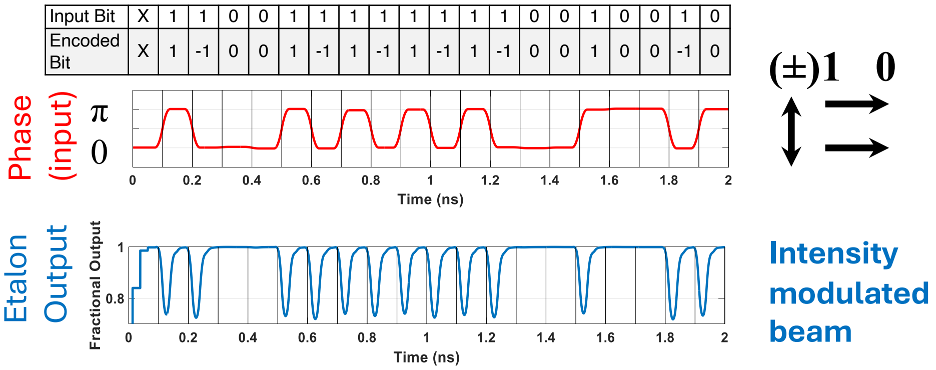
Equation (2.5) hence yields a value of A given the signal received and the (known) previous value of A. This can be converted into intensity given the relationship that

$$\tilde{I}(t) = |A(t)|^2, \tag{2.6}$$

where the tilde indicates that (for now) the intensity is dimensional.

This relationship is shown visually in Fig. 4. If the phase angle is 0, then the waves exiting the demodulator exhibit constructive interference, which causes a high intensity. Hence 0 bits are characterized by high levels of the intensity. Similarly, if the phase angle is π, then the waves exiting the demodulator exhibit destructive interference, which causes a low intensity. Hence 1 bits are characterized by low levels of the intensity.

Note that the intensity is normalized in Fig. 4, and its maximum value occurs for a 0 encoded bit. This motivates the introduction of a series of zeroes before the information signal begins. This series allows any initial transients (as shown at left of the figure) to die down. (This string is called "padding" or "calibration bits" in later sections.) To



Input Bit	X	1	1	0	0	1	1	1	1	1	1	1	1	0	0	1	0	0	1	0
Encoded Bit	X	1	-1	0	0	1	-1	1	-1	1	-1	1	-1	0	0	1	0	0	-1	0

Figure 4. Typical phase encoding (top) and intensity output (bottom), $\tau = 0.1$ ns.

introduce this string, we simply redefine the time at which $t = 0$ to correspond to when the information signal actually begins.

Moreover, we can use this series to calculate a candidate normalization factor I_0 for the intensity. If $\phi(t) \equiv 0$, then $S(t) \equiv 1$ and $A(t)$ should similarly be a constant A_0, which one can derive from (2.5):

$$\begin{aligned}
A_0 &= f_1 + f_2 A_0 \\
A_0 &= \frac{f_1}{1 - f_2} = \frac{f_1}{1 - r_1 r_2 e^{i\delta}}, \qquad (2.7a) \\
|1 - r_1 r_2 e^{i\delta}|^2 |A_0|^2 &= f_1^2 \\
(1 + r_1^2 r_2^2 - 2 r_1 r_2 \cos\delta) I_0 &= f_1^2 \\
I_0 &= \frac{f_1^2}{1 + r_1^2 r_2^2 - 2 r_1 r_2 \cos\delta}. \qquad (2.7b)
\end{aligned}$$

As shown at the bottom of Fig. 4, the normalization of intensity is such that ideally a "0" bit has full intensity throughout the bit. In contrast, a "1" bit experiences a dip in intensity (though not too much; note the small scale in I). Moreover, the dip in the intensity is not uniform for each "1" bit—it can depend on the history of bits before it, as can be inferred from (2.4). (This will become important later when we design our optimization algorithms.)

2.3 The Decoder

Once demodulation is complete, the signal is then decoded and the energy in each bit interval measured. This energy $\tilde{a}_j$ is the average intensity over the bit:

$$\tilde{a}_j = \frac{1}{\tau} \int_{t_{j-1}}^{t_j} \tilde{I}(t)\, dt. \qquad (2.8)$$

A typical situation is sketched in Fig. 5. Since the intensity of the same sort of bit can vary with history, $\tilde{a}_j$ will vary from bit to bit, and an effective system will be one that can accurately distinguish zeroes and ones. Hence we want the area measurements for the two types of bits to be as far apart as possible so they don't get misidentified.

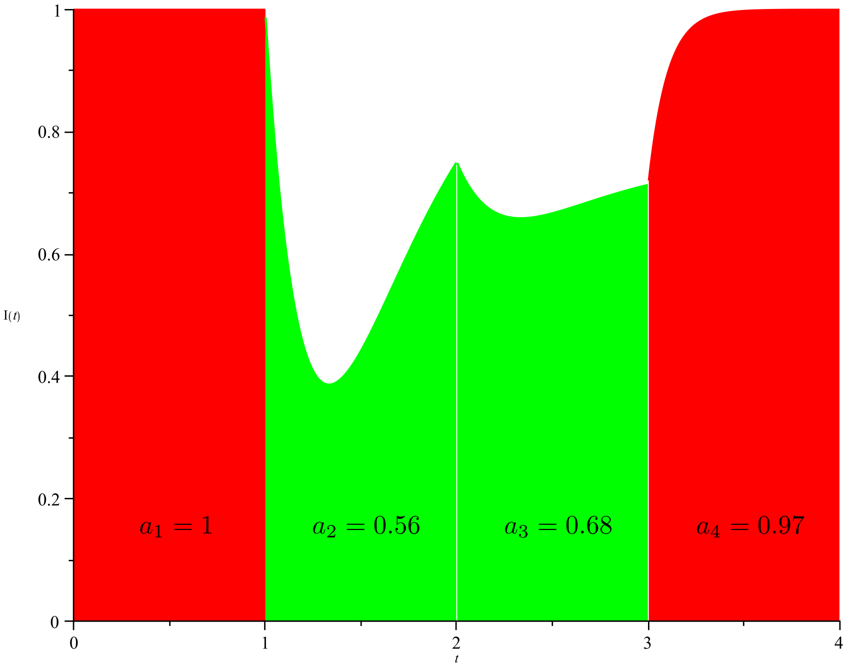


Figure 5. Typical $I(t)$ and areas. Red areas correspond to 0 bits; green areas correspond to 1 bits.

Let J_0 be the set of all bit indices j corresponding to zeroes, and similarly for J_1. $\tilde{a}_j$ for $j \in J_0$ will be large (ideally equal to I_0), so we wish to identify the smallest value of $\tilde{a}_j$ in that set. Similarly, since $\tilde{a}_j$ for $j \in J_1$ will be smaller, we wish to identify the largest value of $\tilde{a}_j$ in that set. So we wish to maximize the following *gap* $\tilde{g}$:

$$\tilde{g} = \min_{j \in J_0} \tilde{a}_j - \max_{j \in J_1} \tilde{a}_j. \tag{2.9}$$

For instance, for the data in Fig. 5, $J_0 = \{1, 4\}$, corresponding to areas of $\{1, 0.97\}$. $J_1 = \{2, 3\}$, corresponding to areas of $\{0.56, 0.68\}$. Hence $g = 0.97 - 0.68 = 0.29$. Also see the schematic in Fig. 6.

This discussion ignores any consideration of noise in the signal, which would complicate the analysis significantly. The overall goal of maximizing the gap in this noiseless case is to make sure that even when signals are noisy, a one bit has a low enough value of $\tilde{a}$ that it doesn't get mixed up with a zero.

3 Optimizing the Encoder

Currently Raytheon has optimized the parameters in the etalon (r_1, r_2, and δ in (2.3)) by assuming that the signal consists of a linear ramp between phase states, as shown in Fig. 7. To improve the decoding, one reduces the growth interval to make the ramp as steep as possible. However, reducing the interval beyond 15 ps showed no appreciable improvement, which may have to do with the 12 GHz bandwidth of the receiver.

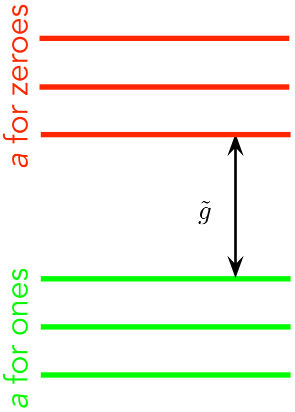


Figure 6. Illustration of gap.

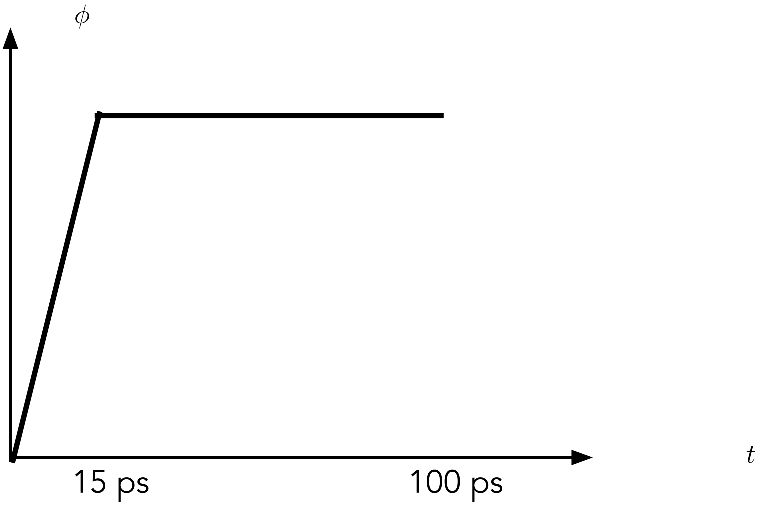


Figure 7. Ramp phase function.

Therefore, in this section we treat the f_k as fixed parameters given by the (black box) etalon. Instead we attempt to optimize $\tilde{g}$ by adapting the ϕ input function for a given bit sequence. Our discussion in §3 indicates that the intensity of each bit will depend on the history of bits before; due to the small size of r_k (see Appendix A), we typically consider only the previous two bits. As a very simple test case, we optimize the phase for the following bit sequence:

$$\text{Test Case 1: } 0, 0, 1. \tag{3.1}$$

To optimize over a wider range of three-digit sequences, we note that given the rules in §2.1, there are only 15 possible encoded 3-bit sequences:

$$\{(0,0,0), \pm(0,0,1), \pm(0,1,0), \pm(0,1,-1), \pm(1,0,0), \pm(1,0,-1), \pm(1,-1,0), \pm(1,-1,1)\}. \tag{3.2}$$

Parameter	Value	Meaning
`delta_L`	0.05	Deviation of etalon length from resonance
`L`	20 μm	Length of etalon
`r1`	0.4	Reflection coefficient
`r2`	0.4	Reflection coefficient
`lambda`	1.5525 nm	Wavelength
`dist_filt`	1	Distortion filter
`c_light`	299792458 m/s	Speed of light
`FSR`	150	Free Spectral Range
`F_P_detune_set`	0	Parameter related to etalon length
`n_mat`	3.47	Index of refraction for silicon
`Length_units`	1	Variable controlling how etalon length is measured
`GRID_Side_Band`	0	Frequency offset in GHz
`F0`	1.93×10^{14}	Center frequency in Hz
`Reciever_on`	4	Type of filter used in receiver simulation
`tau`	0.1 ns	Symbol length

Table 1. Parameters fixed in the optimization procedure.

Rather than test all fifteen sequences of length 3, we embed them into a larger sequence of length 17. We can then use our algorithm to optimize the gap for this sequence, and then break out subsequences as needed. Hence we have that

$$\text{Test Case 2: } 0,0,0,1,0,-1,1,-1,0,0,-1,0,1,-1,1,0,0. \tag{3.3}$$

This sequence contains all fifteen sequences in (3.2) as subsequences. It is important to note that the "black box" etalon demodulator automatically introduces a series of calibration bits, so the test cases refer just to the information sequences.

We sample $\phi(t)$ at a series of times spaced τ/s apart (so there are s samples per each of the M bits) and put the results into a vector $\boldsymbol{\phi}$:

$$\boldsymbol{\phi} \in \mathcal{R}^{sM}, \qquad \phi_n = \phi(n\tau/s). \tag{3.4}$$

Here $M = 3$ if we are in test case 1, and $M = 17$ if we are in test case 2.

Though straightforward to understand, this approach suffers from the fact that the number of samples will increase as M increases. There is also the question of an optimal value for s. If we look at the ramp function arguments, it might seem that it is sufficient to take $s = 6$, since oversampling would force the sample distance to be less than 15 ps. However, there was some discomfort with that idea and it was decided to take s larger to capture higher-frequency features. (We take $s = 100$ to start.)

This approach was coded in Matlab. We used the ramp function as in Fig. 7 as an initial guess, since the engineers expect this to be nearly optimal. However, they also have evidence that overshooting $\phi = \pi$ may also improve performance. This generally seems to be the case from our optimization study, which we now describe.

The output of the etalon `synched_T_R.m` consists of a transmitted intensity $\tilde{I}_{\mathrm{u}}(t)$, and a reflected intensity $\tilde{R}_{\mathrm{u}}(t)$. Given a phase function ϕ as in (2.1), the signal is defined as $S = e^{i\phi}$ and we were provided with a Matlab function `synched_T_R.m` that computes

Parameter	Meaning
`steps_per_cavity`	"Length" of etalon, measured as a multiple of `del_t`
`del_t`	time stepsize = `tau/steps_per_cavity`
s	identical to `steps_per_cavity`
M	Total number of bits in signal (including padding)
`m`	Number of bits in signal (excluding padding)

Table 2. Parameters that are varied in the optimization procedure.

etalon intensities through

$$\tilde{I}_{\mathrm{u}}(t) = \texttt{synched\_T\_R}(e^{i\phi}). \tag{3.5}$$

The transmitted intensity $\tilde{I}_{\mathrm{u}}$ is filtered to give a final intensity $\tilde{I}$ which is then used to compute the gap using equations (2.8) and (2.9). We adapted the provided codes to write a Matlab function `BlueBox2.m` that takes in the phase function and outputs the gap $\tilde{g}$:

$$\tilde{g} = \texttt{BlueBox2(phi)}. \tag{3.6}$$

`BlueBox2.m` uses many other parameters not indicated by (3.6). Most of them are fixed with values given in Table 1. Others are related to the time grid: see Table 2. The function `define_time_grid2.m` creates a uniformly spaced time grid (`t_vec`) given the symbol length (`tau`), the number of bits in the string (`m`), the number of padded zeros before (`symbols_before_first`), the number of padded zeros after (`symbols_after_last`), and the number of steps per symbol (`steps_per_cavity`):

$$\texttt{t\_vec = [0:del\_t:T]}, \tag{3.7}$$

where $T = \texttt{tau} \times (\texttt{m} + \texttt{symbols\_before\_first} + \texttt{symbols\_after\_last})$ and `del_t` is defined in Table 2.

We were tasked with finding the optimal ϕ that maximizes the gap $\tilde{g}$. To accomplish this, we used the Matlab function `fmincon.m` which performs a multi-dimensional constrained minimization. Mathematically, we wish to find a vector $\boldsymbol{\phi}^*$ such that

$$\boldsymbol{\phi}^* = \arg\min_{\phi} \left[-\texttt{BlueBox2}(\phi) + \alpha \int_0^T \left(\frac{d^2\phi}{dt^2} \right)^2 dt \right], \tag{3.8}$$

subject to the constraints that the first ($s \times \texttt{symbols\_before\_first}$) components of $\boldsymbol{\phi}$ are set to zero and the remaining components lie in the interval $[-5, 5]$. The interval $[-5, 5]$ was arbitrarily chosen: the most important property for the interval is that it should be a superset of $[-\pi, \pi]$, the range of the phase.

Note that minimizing $-$`BlueBox2` is equivalent to maximizing `BlueBox2` and hence the gap $\tilde{g}$. The second term in (3.8) represents a regularization. Occasionally, when we tried to minimize $-$`BlueBox2` on its own ($\alpha = 0$), our optimal solutions were highly oscillatory, for example fluctuating from $-\pi/2$ to $\pi/2$ within a single time step `del_t`, representing just a few picoseconds. We thought these solutions were unphysical and the effect of the regularization term with $0 < \alpha \ll 1$ is to penalize highly oscillatory solutions so that they are not selected by `fmincon.m`. A smoothly varying optimal ϕ would maximize `BlueBox2` *and* have a second derivative $\frac{d^2\phi}{dt^2}$ that was not too large.

We also wrote a Matlab function `phasefunc.m` which generates an initial guess for the optimization. The function generates a phase ϕ that starts at zero and increases by π when the bit value is 1, decreases by π when the bit value is -1, and stays the same when the bit value is 0. When the phase changes, it ramps up/down over a period of 15 picoseconds. An example is shown in Figure 8 which produces a (suboptimal) gap of 0.005773.

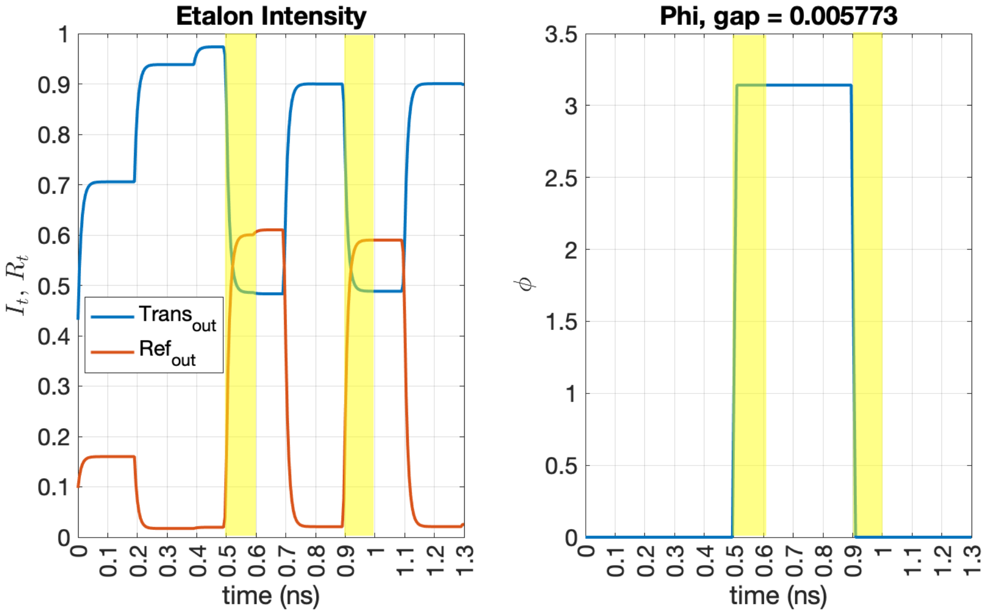


Figure 8. Normalized etalon intensities (left) and an initial guess for the phase function (right). The phase function corresponds to the bit string [0 0 1 0 0 0 -1] with 3 padded zeros on either side, for a total of $M = 13$ bits. The gap for this phase function (computed from the transmitted intensity) is 0.005773. Highlighted cells denote nonzero bits.

Note that throughout this section, three padding bits were used before the signal was sent. We learned after the workshop that the etalon reaches steady state at 0.5 ns, and so experiments are usually run with a 5-bit-padding before and a 3-bit-padding after.

In Figure 9 we show an optimized phase function with no regularization ($\alpha = 0$). The optimal ϕ resembles a "square wave", slowly ramping up at 0.3 ns and then quickly spiking up at 0.5 ns. The gap is about 0.049, almost 10× larger than the simple phase function from Figure 8.

Figures 10 and 11 show optimized phase functions for the bit string 0,0,1,0,0,0,-1 with $\alpha = 0$ and $\alpha = 10^{-10}$ respectively. The etalon intensities are qualitatively similar and the optimal gaps are also similar (0.0447 and 0.0443). The unregularized ϕ in Figure 10 has a downward spike at 0.3 ns which seems unphysical, especially since the bit string is 0 during the first 0.6 ns: there should not be any changes in the phase during this time.

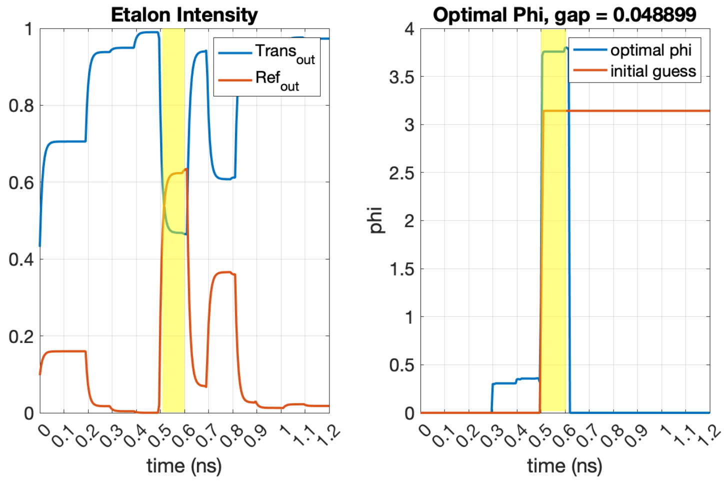


Figure 9. Optimized phase function ϕ (right) along with corresponding normalized etalon intensities (left) for bit string 0,0,1. The string was padded with 3 and 6 zeros before and after, $\texttt{steps\_per\_cavity} = 20$, and $r_1 = r_2 = 0.4$. The regularization parameter $\alpha = 0$. Highlighted cells denote nonzero bits.

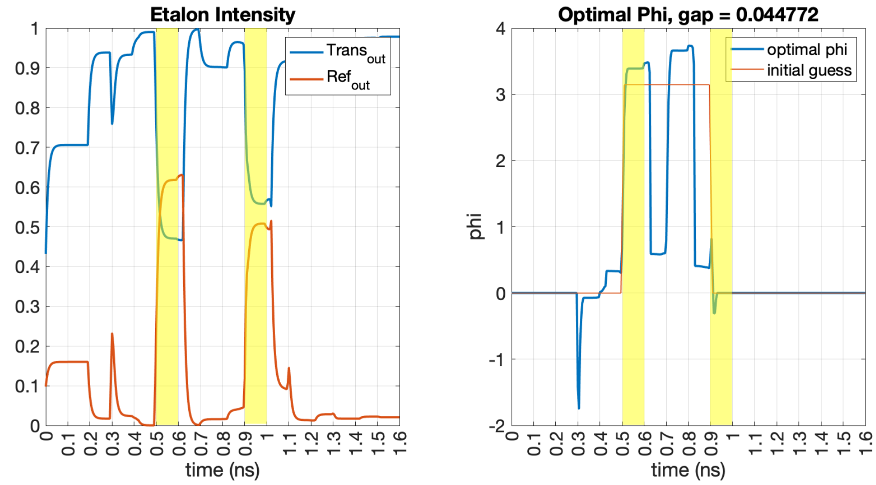


Figure 10. Optimized phase function ϕ (right) along with corresponding normalized etalon intensities (left) for bit string 0,0,1,0,0,0,−1. The string was padded with 3 and 6 zeros before and after, $\texttt{steps\_per\_cavity} = 20$, and $r_1 = r_2 = 0.4$. The regularization parameter $\alpha = 0$. Highlighted cells denote nonzero bits.

This spike has been mostly damped out when $\alpha = 10^{-10}$ in Fig. 11. At the same time, the reglularized phase retains most of the important qualitative features from Fig. 10: a large phase for $0.5 < t < 0.6$ and $0.7 < t < 0.8$ with a dip in-between.

Finally, we perform a convergence study to see if our results are robust when the time resolution is increased. In Figure 12, we double and quadruple `steps_per_cavity` which

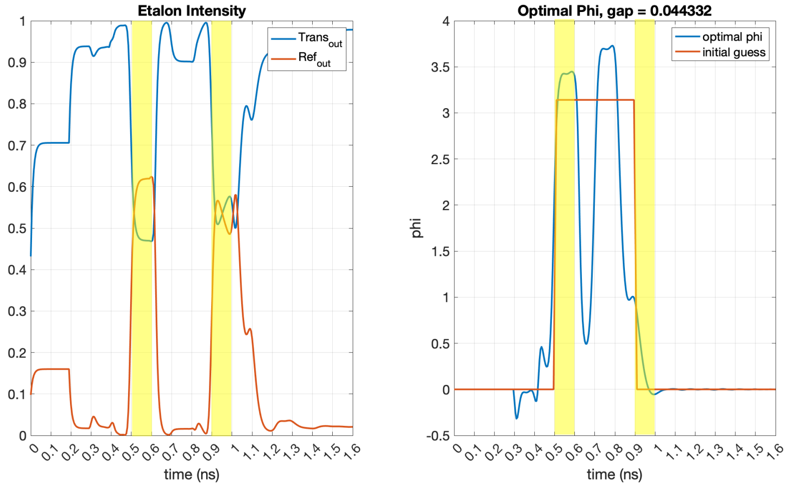


Figure 11. Optimized phase function ϕ (right) along with corresponding normalized etalon intensities (left) for bit string 0,0,1,0,0,0,−1. The string was padded with 3 and 6 zeros before and after, `steps_per_cavity` $= 20$, and $r_1 = r_2 = 0.4$. The regularization parameter $\alpha = 10^{-10}$. Highlighted cells denote nonzero bits.

is passed as the first argument to the function `synched_T_R.m`. The etalon intensities, gaps, and optimal ϕ do not change significantly as the number of time points is increased from 261, to 521, to 1041.

4 Amplitude Optimization

Though the primary measurement variable is the accumulated area $\tilde{a}_j$, we may also optimize $A(t)$ locally. Hopefully optimizing locally will produce results nearly as good as trying to maximize the gap directly.

To proceed, we rewrite the second term on the right-hand side of (2.5) in polar coordinates:

$$f_2 A(t - \Delta t) = Re^{i\theta(t)}, \tag{4.1}$$

and rewrite (2.5) using phase notation:

$$A(t) = f_1 e^{i\phi(t)} + Re^{i\theta(t)} \tag{4.2a}$$

$$\tilde{I}(t) = |A(t)|^2 = f_1^2 + R^2 + 2Rf_1 \cos(\phi(t) - \theta(t)). \tag{4.2b}$$

Equation (4.2b) now provides a function we can optimize. In this section we constrain $\theta \in [0, \pi]$ and do not consider the type of overshoot discussed in §3.

To send a 0 bit, we want to make $\tilde{I}(t)$ as large as possible, so make the cosine as large as possible. Clearly, if $\theta(t) \in [0, \pi]$, we simply take $\phi(t) = \theta(t)$ so the argument of the cosine is 0 and the cosine is maximized. But if we assume that $\phi(t)$ is bounded in $[0, \pi]$, then we must use the endpoints when θ is outside that range. (Note that the endpoints correspond to the first term on the right-hand side of (4.2a) being a positive or negative real number.)

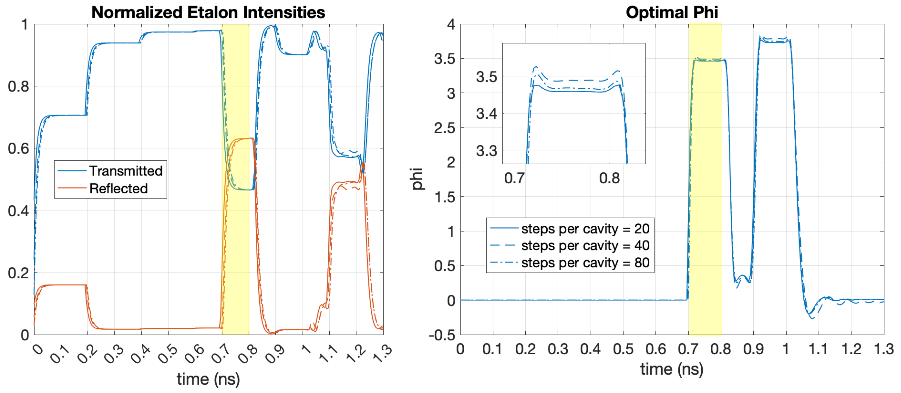


Figure 12. Convergence study with increasing time resolution. The optimized phase function ϕ (right) is shown along with corresponding normalized etalon intensities (left) for the bit string 0,0,0,0,0,0,0,1,0,0,0,0,0: a '1' with 7 zeros preceding it and 5 zeros following. The gap sizes were 0.044795, 0.043416 and 0.042608, corresponding to 261 (`steps_per_cavity` $= 20$), 521 (`steps_per_cavity` $= 40$) and 1041 (`steps_per_cavity` $= 80$) time points respectively. Other etalon parameters were $r_1 = r_2 = 0.4$, $L = 20$. The regularization parameter $\alpha = 10^{-11}$. Highlighted cells denote nonzero bits.

If the real part of the right-hand side of (4.1) is positive, then we want to add a positive number in (4.2a), so we take $\phi(t) = 0$. Similarly, if the real part of the right-hand side of (4.1) is negative, then we want to add a negative number, so we take $\phi(t) = \pi$. The end result is as follows:

$$\phi(t) = \begin{cases} \pi, & \theta \in [-\pi, -\pi/2), \\ 0, & \theta \in (-\pi/2, 0], \\ \theta, & \theta \in [0, \pi]. \end{cases} \tag{4.3a}$$

To send a 1 bit, the process is essentially reversed. In this case, we want to make $\tilde{I}(t)$ as small as possible, so make the cosine as small as possible. Clearly, if $\theta(t) \in [-\pi, 0]$, we simply take $\phi(t) = \pi + \theta(t)$ so the argument of the cosine is π and the cosine is minimized. Otherwise, we must use the endpoints.

If the real part of the right-hand side of (4.1) is positive, then we want to add a negative number, so we take $\phi(t) = \pi$. Similarly, if the real part of the right-hand side of (4.1) is negative, then we want to add a positive number, so we take $\phi(t) = 0$. The end result is as follows:

$$\phi(t) = \begin{cases} \pi + \theta, & \theta \in [-\pi, 0], \\ \pi, & \theta \in [0, \pi/2), \\ 0, & \theta \in (\pi/2, \pi]. \end{cases} \tag{4.3b}$$

With this formulation, we don't need to worry about a $+1$ or -1 encoded bit; we just consider the signal bit and let the algorithm decide whether it is a jump up or down in ϕ. Therefore, to test all possible combinations we can create a sequence motivated by the *pseudorandom binary sequence* (PRBS) [Davies, 1970] for 3-bit words, PRBS$_3$. PRBS$_3$

is a 7-bit sequence that contains every nonzero 3-bit word, allowing cycles through the end.

Unfortunately, those characteristics are too restrictive for us, so we have to adapt PRBS_3 for two reasons:

- We will need to cover the sequence 000.
- Given the fact that the behavior of the etalon depends on the signal history, we need to include each subsequence directly in the list.

Once we obey those two rules, we have the following test sequence:

$$\text{Test Case 3: } 1, 0, 1, 1, 1, 0, 0, 0, 1, 0. \tag{4.4}$$

4.1 Unfiltered Implementation

To implement our scheme, we first initialize the problem by letting

$$A(t) \equiv A_0, \qquad S(t) = 1, \quad t \in [-\Delta t, 0]. \tag{4.5}$$

We also introduce three *calibration bits* of 0 at the front of our sequence to eliminate any transients. (By the discussion in the previous section, we should really use five calibration bits.)

We implement a discretization scheme similar to that in (3.4). We take $s = 100$ as in §3. At each time step, we look back and calculate $\theta(t)$ using (4.1). Then given the signal we wish to send (0 or 1), we set ϕ using (4.3). This will then give us an optimal $A(t)$.

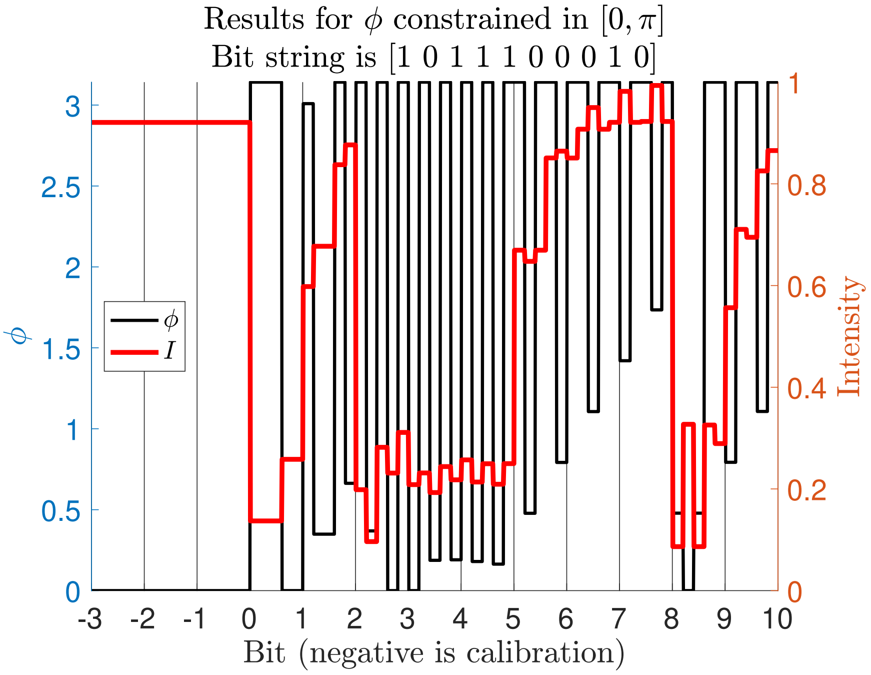


Figure 13. Signal results from constrained algorithm. Left scale (black): ϕ. Right scale (red): I. Initialization string is omitted; calibration bits have negative t-values.

The results are shown in Fig. 13. As expected, the intensity (red curve) is higher for the 0 bits and lower for the 1 bits. We also see significant oscillations in the ϕ values, as opposed to the ramp function in Fig. 7. This was unexpected, but the reason can be seen from (4.3). If we are sending a 0 bit and $\theta \approx -\pi/2$, then ϕ will vary quickly between 0 and π depending on the sign of $\theta + \pi/2$. Similarly, if we are sending a 1 bit, ϕ will vary quickly between 0 and π depending on the sign of $\theta - \pi/2$.

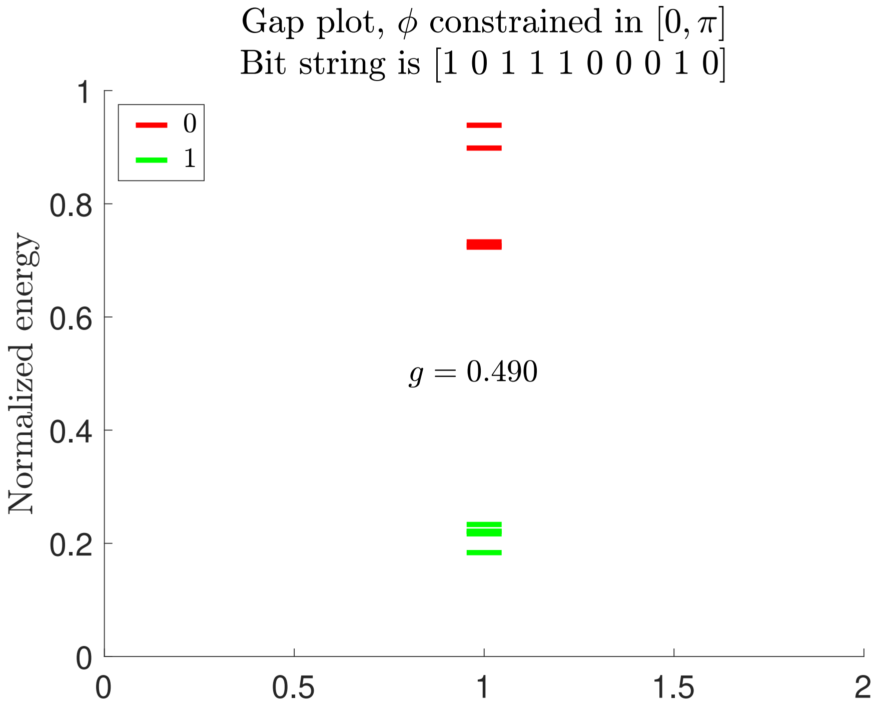


Figure 14. Energy results from constrained algorithm. Red: zero bit values. Green: one bit values.

Once we do the process for the entire string length, we then compute $\tilde{a}_j$ for each signal bit. The results shown in Fig. 14 show a substantially large gap. (Note that these values are normalized, and hence this gap cannot be compared directly to those in §3.) These results give us confidence that even in the case where the data is noisy, the $\tilde{a}_j$ values for zeroes and ones will remain segregated enough that they will be easy to classify.

Though our algorithm is mathematically elegant and provides much wider gaps than shown in Fig. 4, an obvious question is whether we can actually vary ϕ that quickly in hardware. The answer is no, and in the next subsection we will present results that take this into account.

4.2 Matching Raytheon's Approach

We change notation in this subsection to match some prior equations derived by Ben Dolgin at Raytheon. Let $S(t)$ be the signal entering the front of the etalon from outside (see right of Fig. 3). Let $A(t)$ be the signal hitting the back of the etalon. Let

$B(t)$ be the signal hitting the front of the etalon in the reverse direction, having been reflected off the back of the etalon. Let $T_{\mathrm{t}}(t)$ be the signal leaving the etalon out the back (the received "transmission" as opposed to the received "reflection.") Then we have

$$\begin{aligned} S(t) &= e^{i\phi(t)}, \\ A(t) &= r_1 e^{i\delta_1} B(t-\Delta t/2) + \sqrt{1-r_1^2}\, S(t-\Delta t/2), \\ B(t) &= r_2 e^{i\delta_2} A(t-\Delta t/2), && (4.6a) \\ T_{\mathrm{t}}(t) &= \sqrt{1-r_2^2}\, A(t+\Delta t/2). && (4.6b) \end{aligned}$$

In actuality, $T_{\mathrm{t}}(t)$ would be in phase with $A(t)$, but the artificial shift of $\Delta t/2$ in the last equation makes some things prettier. (In this section, T_{t} plays the role of A in the other sections.)

Eliminating $B(t)$ in the above system produces

$$\begin{aligned} A(t) &= r_1 e^{i\delta_1}\left(r_2 e^{i\delta_2} A(t-\Delta t)\right) + \sqrt{1-r_1^2}\, S(t-\Delta t/2) \\ &= r_1 r_2 e^{i\delta} A(t-\Delta t) + \sqrt{1-r_1^2}\, S(t-\Delta t/2), \end{aligned}$$

where $\delta = \delta_1 + \delta_2$. At time $t + \Delta t/2$, this expression becomes

$$A(t+\Delta t/2) = r_1 r_2 e^{i\delta} A(t-\Delta t/2) + \sqrt{1-r_1^2}\, S(t).$$

Then substituting $A(u) = (1-r_2^2)^{-1/2} T_{\mathrm{t}}(u-\Delta t/2)$ gives

$$(1-r_2^2)^{-1/2} T_{\mathrm{t}}(t) = r_1 r_2 e^{i\delta} (1-r_2^2)^{-1/2} T_{\mathrm{t}}(t-\Delta t) + \sqrt{1-r_1^2}\, S(t),$$

or more simply,

$$T_{\mathrm{t}}(t) = r_1 r_2 e^{i\delta} T_{\mathrm{t}}(t-\Delta t) + \sqrt{(1-r_1^2)(1-r_2^2)}\, S(t). \tag{4.7}$$

(Note that (4.7) is exactly (2.5) written in the new notation.) $T_{\mathrm{t}}(t)$ is the present value, $T_{\mathrm{t}}(t-\Delta t)$ is a past value, and $S(t)$ is under our control: the only constraint on $S(t)$ is that its modulus is 1.

In order to drive the received intensity up or down, we choose $S(t)$ to have either the same phase as $T_{\mathrm{t}}(t-\Delta t)$ or the opposite phase:

$$S(t) = \begin{cases} \dfrac{T_{\mathrm{t}}(t-\Delta t)}{|T_{\mathrm{t}}(t-\Delta t)|} & \text{if transmitting a 0 bit at time } t, \\ -\dfrac{T_{\mathrm{t}}(t-\Delta t)}{|T_{\mathrm{t}}(t-\Delta t)|} & \text{if transmitting a 1 bit at time } t. \end{cases}$$

In the first case, $S(t)$ and $T_{\mathrm{t}}(t-\Delta t)$ have the same phase, so their moduli add without cancelation. In the second case, the signals have opposite phases, so there is as much cancelation as possible.

Example. Suppose we want to transmit the bit string ↑ ↑ ↑ ↓ ↑ ↑ ↓ ↓ ↓ ↓ ↓ ↑ ↑ ↑ ↓ ↑ ↑ ↑ ↑ ↑. Through trial and error, the best parameters we have found are

$$\frac{\Delta t}{\tau} = \frac{1}{2}, \qquad r_1 = r_2 = 0.55.$$

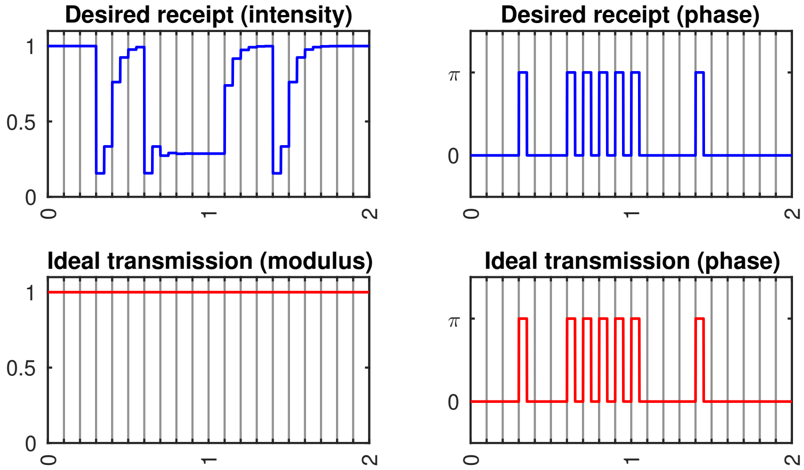


Figure 15. Unfiltered results from unconstrained algorithm. Top left: $|T_{\rm t}(t)|^2$. Bottom right: $\phi(t)$ for the computed $S(t)$.

Graphical results are shown in Fig. 15, which includes step functions because no filtering is applied. The top-left plot is the computed $|T_{\rm t}(t)|^2$. Each vertical grid line separates symbols, and we want the blue curve to be high when transmitting a ↑ and low when transmitting a ↓. The red curve in the bottom-right is the phase $\phi(t)$ for the computed $S(t)$.

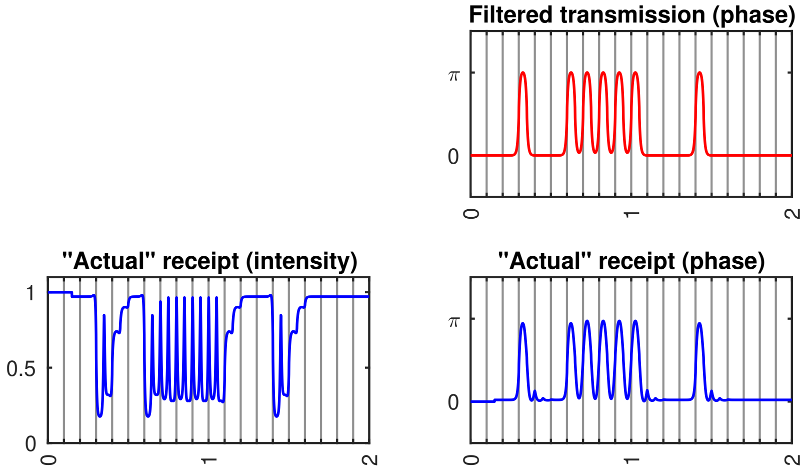


Figure 16. Filtered results from unconstrained algorithm. Bottom left: $|T_{\rm t}(t)|^2$. Top right: $\phi(t)$ for the computed $S(t)$.

The solution is checked by filtering $S(t)$ and pushing it forward through the system, as shown in Fig. 16. The output appears to degrade gracefully with filtering. Moreover, even with filtering we still produce 1-bit intensity values less than those shown in Fig. 4. Hence this algorithm improves the gap distance and hence the transmission process.

Some results that we have found based on simulations with this scheme:

•The etalon length, in relation to the symbol duration, is the same as found earlier by Raytheon: we should have $\Delta t = \tau/2$.

•The reflectivity coefficients that appear to be optimal in our simulations are close to those found by Raytheon: We found $r_1 = r_2 = 0.55$, while earlier simulations indicated $r_1 = r_2 = \sqrt{0.4} = 0.63$.

•The major lesson, and the major difference from Raytheon's previous simulations, concerns the *pulse duration*: If a bit is to be transmitted over a period of time τ, then phase discontinuities should be transmitted at the beginning of this period and again halfway through. See the red square wave above, and notice how the jump discontinuities are happening at twice the frequency of the symbol transmission.

5 Variational Principles: Fabry-Perot Model

To create a variational formulation of the problem, we use (2.3b) in (2.8), keeping in mind that the intensity is just $A(t)$ times its complex conjugate:

$$\tilde{a}_j = \frac{1}{\tau}\int_{t_{j-1}}^{t_j} f_1^2 \sum_{n=0}^{\infty}\sum_{k=0}^{\infty}\left(r_1 r_2 e^{i\delta}\right)^k S(t-k\Delta t)\left(r_1 r_2 e^{-i\delta}\right)^n S^*(t-n\Delta t)\,dt$$

$$a_j = \sum_{n=0}^{\infty}\sum_{k=0}^{\infty}(r_1 r_2)^{k+n}\int_{t_{j-1}}^{t_j}\exp\left(i\Phi_{k,n}(t)\right)\,dt, \tag{5.1a}$$

$$a_j = \frac{\tilde{a}_j\tau}{f_1^2}, \qquad \Phi_{k,n}(t) = (k-n)\delta + \phi(t-k\Delta t) - \phi(t-n\Delta t), \tag{5.1b}$$

where in (5.1a) we have used (2.1). Note also that in (5.1b) our normalization factor is slightly different from that computed in (2.7b) (f_1^2 *vs.* I_0).

We wish to optimize the a_j using variations in ϕ, so we isolate those terms that depend on ϕ. Let's examine (5.1b). Any term with $k = n$ will not depend on ϕ since those terms cancel. Moreover, $\Phi_{k,n} = -\Phi_{n,k}$. Therefore, we may simplify the ϕ-dependent terms of a_j:

$$\begin{aligned} a_j &= \sum_{k=1}^{\infty}\sum_{n=0}^{k-1}(r_1 r_2)^{k+n}\int_{t_{j-1}}^{t_j}\exp\left(i\Phi_{k,n}(t)\right) + \exp\left(-i\Phi_{k,n}(t)\right)\,dt \\ &= 2\sum_{k=1}^{\infty}\sum_{n=0}^{k-1}(r_1 r_2)^{k+n}\int_{t_{j-1}}^{t_j}\cos\left(\Phi_{k,n}(t)\right)\,dt. \end{aligned} \tag{5.2}$$

IMPORTANT: Note that we have changed our notation slightly, and a_j here and below will *not* be the measured energy level, as we are now ignoring those parts of the energy which are independent of ϕ. However, it may be the case that the terms which are independent of ϕ just sum to I_0, but time ran out before we could verify this.

Since $\Phi_{k,k}(t) = 0$, the contribution of the $k = n$ terms is given by a geometric series and is easy to compute; their total contribution to $\tilde{a}_j$ and a_j are as follows:

$$\tilde{a}_j : \frac{f_1^2}{1-(r_1 r_2)^2}, \qquad a_j : \frac{\tau}{1-(r_1 r_2)^2}.$$

Also note that if $\phi(t)$ is a constant, in particular if $\phi(t) \equiv 0$, $\tilde{a}_j$ is given, in total, by an absolute value of a geometric series, squared:

$$\tilde{a}_j = f_1^2 \left| \frac{1}{1 - r_1 r_2 e^{i\delta}} \right|^2,$$

as consistent with (2.7).

We also note that if $t_j - k\Delta t < 0$, then both ϕ terms have negative arguments and are hence 0. This upper limit maybe calculated as follows:

$$\begin{aligned} t_j - k\Delta t &< 0 \\ k_{\max} &= \left\lfloor \frac{j\tau}{\Delta t} \right\rfloor = \left\lfloor \frac{j}{0.6} \right\rfloor = \left\lfloor \frac{5j}{3} \right\rfloor, \end{aligned} \tag{5.3}$$

where we have used (A 1) and (A 2a). Continuing to simplify, we obtain

$$a_j = 2 \sum_{k=1}^{k_{\max}} \sum_{n=0}^{k-1} (r_1 r_2)^{k+n} \int_{t_{j-1}}^{t_j} \cos\left(\Phi_{k,n}(t)\right)\, dt. \tag{5.4}$$

To illustrate the procedure, consider a two-bit string:

$$a_1 = 2(r_1 r_2) \int_0^{\tau} \cos\left(\Phi_{1,0}(t)\right)\, dt, \tag{5.5a}$$

$$a_2 = 2 \sum_{k=1}^{3} \sum_{n=0}^{k-1} (r_1 r_2)^{k+n} \int_{\tau}^{2\tau} \cos\left(\Phi_{k,n}(t)\right)\, dt, \tag{5.5b}$$

where in (5.5b) we have used (5.3) to determine that $k_{\max} = 3$. For the string 01, the expression for g becomes

$$g = a_1 - a_2, \tag{5.6}$$

where we assume the same normalization for a and g. Note also that **for the two-bit string considered**, the gap subtracts out the terms that are independent of ϕ, so g here *is* the correct normalized gap measurement.

For proof of concept, we assume that the r_j are so small that we can neglect all terms smaller than $r_1 r_2$. In that case, we may write g as

$$\begin{aligned} g[\phi] &= 2(r_1 r_2) \left[\int_0^{\tau} \cos\left(\Phi_{1,0}(t)\right)\, dt - \int_{\tau}^{2\tau} \cos\left(\Phi_{1,0}(t)\right)\, dt \right] \\ \frac{g[\phi]}{2r_1 r_2} &= \int_0^{\tau} \cos\left(\delta - \phi(t) + \phi(t - \Delta t)\right)\, dt - \int_{\tau}^{2\tau} \cos\left(\delta - \phi(t) + \phi(t - \Delta t)\right)\, dt \\ &= \int_0^{2\tau} [1 - 2H(t - \tau)] \cos\left(\delta - \phi(t) + \phi(t - \Delta t)\right)\, dt, \end{aligned} \tag{5.7}$$

where we have used the definition of $\Phi_{1,0}$. Calculating the variation, we have

$$\begin{aligned} \frac{g[\phi + \epsilon v]}{2r_1 r_2} &= \int_0^{2\tau} [1 - 2H(t - \tau)] \cos\left(\delta - \phi(t) - \epsilon v(t) + \phi(t - \Delta t) + \epsilon v(t - \Delta t)\right)\, dt \\ \frac{\delta g}{2r_1 r_2} &= \int_0^{2\tau} [1 - 2H(t - \tau)][v(t) - v(t - \Delta t)] \sin\left(\delta - \phi(t) + \phi(t - \Delta t)\right)\, dt. \end{aligned} \tag{5.8}$$

At this point we note that since $\phi(t)$ is identically zero for negative argument, so is $v(t)$

(since no variation is allowed). We next undo the shift so the only variation is $v(t)$:

$$\begin{aligned}\frac{\delta g}{2r_1r_2} &= \int_0^{2\tau} v(t)[1-2H(t-\tau)]\sin(\delta-\phi+\phi_-)\,dt \\ &\quad - \int_0^{2\tau-\Delta t} v(t)[1-2H(t-\tau+\Delta t)]\sin(\delta-\phi_+ +\phi)\,dt, \qquad (5.9a) \\ \phi_\pm &= \phi(t\pm\Delta t), \qquad (5.9b)\end{aligned}$$

where we again use the fact that $v(t)=0$ for negative argument. Then combining terms and rewriting, we obtain the following:

$$\frac{\delta g}{2r_1r_2} = \int_0^{2\tau} v(t)\left\{[1-2H(t-\tau)]\sin(\delta-\phi+\phi_-) + h(t)\sin(\delta-\phi_+ +\phi)\right\}\,dt, \qquad (5.10a)$$

$$h(t) = \begin{cases} -1, & t < \tau-\Delta t, \\ 1, & \tau-\Delta t < t < 2\tau-\Delta t, \\ 0, & 2\tau-\Delta t < t < 2\tau. \end{cases} \qquad (5.10b)$$

Then setting the braced term equal to 0, we would seem to have four equations:

$$\begin{aligned} \sin(\delta-\phi+\phi_-) - \sin(\delta-\phi_+ +\phi) &= 0, \quad 0 < t < \tau-\Delta t, && (5.11a) \\ \sin(\delta-\phi+\phi_-) + \sin(\delta-\phi_+ +\phi) &= 0, \quad \tau-\Delta t < t < \tau, && (5.11b) \\ -\sin(\delta-\phi+\phi_-) + \sin(\delta-\phi_+ +\phi) &= 0, \quad \tau < t < 2\tau-\Delta t, && (5.11c) \\ -\sin(\delta-\phi+\phi_-) &= 0, \quad 2\tau-\Delta t < t < 2\tau. && (5.11d) \end{aligned}$$

However, it is even more complicated than this. In the region $0 < t < \tau$ (which overlaps the region in (5.11b) by the choice of (A 2a) for Δt), ϕ_- will vanish, which adds another region to analyze. This process repeats throughout the analysis. Moreover, we note from (5.11d) that the solution will be dependent on ϕ values *before* the region (through ϕ_-), while in the first part of the region in (5.11a) (where the ϕ_- drops out), the solution is dependent on ϕ values *after* the region (through ϕ_+). So we would expect that this overlap may force some sort of compatibility condition on the solution.

The problem with this recursive subdivision of domains can be fixed if we set τ and Δt such that their ratio is a rational number. We discuss some examples below.

5.1 $\Delta t = \tau$

In the simplest case of $\Delta t = \tau$, noting that in that case $\phi_-(t) = 0$ for $t \in [0,\tau]$ and that by shifting the integration variable the variational functional in (5.7) can be written as

$$\frac{g[\phi+\epsilon v]}{2r_1r_2} = \int_0^{\tau} \cos\left(\delta-\phi(t)\right) - \cos\left(\delta-\phi(t+\Delta t)+\phi(t)\right)\,dt. \qquad (5.12)$$

With $\Delta t = \tau$, the only variational equations that hold are (5.11b) and (5.11d):

$$\begin{aligned} \sin(\delta-\phi) + \sin(\delta-\phi_+ +\phi) &= 0, \quad 0 < t < \tau, \quad \text{(region 1)} && (5.13a) \\ -\sin(\delta-\phi+\phi_-) &= 0, \quad \tau < t < 2\tau. \quad \text{(region 2)} && (5.13b) \end{aligned}$$

But note that with our choice of Δt, by our definitions of $\phi_\pm$,

$$(\phi_+ \text{ in region } j-1) = (\phi \text{ in region } j) = (\phi_- \text{ in region } j+1). \tag{5.14}$$

Therefore, working backwards, we see that enforcing (5.13b) in region 2 forces the sine term above it in (5.13a) to vanish. Thus we must have that

$$\begin{aligned} \sin(\delta-\phi) = 0 \qquad &\Longrightarrow \qquad \phi(t) = \delta + \pi m_1, \quad 0 < t < \tau, \\ \sin(\delta - \phi + \delta + \pi m_1) = 0 \qquad &\Longrightarrow \qquad \phi(t) = 2\delta + \pi(m_1 + m_2), \quad \tau < t < 2\tau, \end{aligned}$$

where the m_j are arbitrary integers. Taking $m_1 = 0$ (as justified below) and m_2 odd would maximize the variational functional (5.12), but this will push ϕ outside of the interval $[0, \pi]$ in the region $(\tau, 2\tau)$. However, since $\delta \ll 1$ the deviation will be only slight, which is consistent with the overshoot approach discussed in §3.

5.2 $\Delta t = \tau/2$

For the purposes of this section we take $\tau = 2\Delta t$, as motivated by (A 2b). In this case, (5.11) becomes

$$\begin{aligned} \sin(\delta-\phi) - \sin(\delta - \phi_+ + \phi) &= 0, \quad 0 < t < \tau/2, \quad \text{(region 1)} && \text{(5.15a)} \\ \sin(\delta - \phi + \phi_-) + \sin(\delta - \phi_+ + \phi) &= 0, \quad \tau/2 < t < \tau, \quad \text{(region 2)} && \text{(5.15b)} \\ -\sin(\delta - \phi + \phi_-) + \sin(\delta - \phi_+ + \phi) &= 0, \quad \tau < t < 3\tau/2, \quad \text{(region 3)} && \text{(5.15c)} \\ -\sin(\delta - \phi + \phi_-) &= 0, \quad 3\tau/2 < t < 2\tau, \quad \text{(region 4)} && \text{(5.15d)} \end{aligned}$$

where in (5.15a) we have used the fact that $\phi_- = \phi(t - \tau/2) \equiv 0$ in region 1.

Working backwards, we see that the solution of (5.15d) is

$$\begin{aligned} \phi &= \phi_- + \delta + m_4\pi, \quad \text{(region 4)} && \text{(5.16a)} \\ \phi(t) &= \phi(t - \tau/2) + \delta + m_4\pi, \quad 3\tau/2 < t < 2\tau. && \text{(5.16b)} \end{aligned}$$

for some integer m_4. Exploiting (5.14), we may again work backwards. Enforcing (5.15d) in region 4 forces the sine term above it in (5.15c) to vanish, which means that (5.15c) also reduces to (5.15d).

This process repeats, so (5.15) is replaced with

$$\begin{aligned} \sin(\delta-\phi) &= 0, \quad 0 < t < \tau/2, \quad \text{(region 1)} && \text{(5.17a)} \\ \sin(\delta - \phi + \phi_-) &= 0, \quad \tau/2 < t < \tau, \quad \text{(region 2)} && \text{(5.17b)} \\ -\sin(\delta - \phi + \phi_-) &= 0, \quad \tau < t < 3\tau/2, \quad \text{(region 3)} && \text{(5.17c)} \\ -\sin(\delta - \phi + \phi_-) &= 0, \quad 3\tau/2 < t < 2\tau. \quad \text{(region 4)} && \text{(5.17d)} \end{aligned}$$

(Even though the equations in regions 3 and 4 are identical, we list them separately since the choice of optimizer will be different.) Hence in region 1 we have

$$\sin(\delta - \phi) = 0 \qquad \Longrightarrow \qquad \phi(t) = \delta + m_1\pi$$

for some integer m_1. But we expect that there will be a calibration signal for negative t with $\phi \equiv 0$, so we take $m_1 = 0$ in the above for continuity, which yields

$$\phi(t) = \delta, \quad \Phi_{1,0}(t) = \delta - \delta = 0, \quad \cos(\Phi_{1,0}(t)) = 1, \quad 0 < t < \tau/2. \tag{5.18a}$$

Now moving to region 2, we have that

$$\sin(\delta - \phi + \delta) = 0 \qquad \Longrightarrow \qquad \phi(t) = 2\delta + m_2\pi,$$

for any integer m_2, and in general

$$\phi(t) = j\delta + m_j\pi, \quad \Phi_{1,0}(t) = -(m_j - m_{j-1})\pi, \quad \cos\left(\Phi_{1,0}(t)\right) = (-1)^{m_j - m_{j-1}} \text{ in region } j. \tag{5.18b}$$

The fact that (5.6) has multiple optimizers means we have to compute g directly to see which works best. Substituting (5.18) into (5.7), we obtain the following:

$$\begin{aligned} \frac{g[\phi]}{2r_1r_2} &= \int_0^{\tau/2} \cos\left(\Phi_{1,0}(t)\right)\, dt + \int_{\tau/2}^{\tau} \cos\left(\Phi_{1,0}(t)\right)\, dt - \int_{\tau}^{3\tau/2} \cos\left(\Phi_{1,0}(t)\right)\, dt \\ &\qquad - \int_{3\tau/2}^{2\tau} \cos\left(\Phi_{1,0}(t)\right)\, dt \\ &= \int_0^{\tau/2} dt + \int_{\tau/2}^{\tau} (-1)^{m_2}\, dt - \int_{\tau}^{3\tau/2} \cos\left(\Phi_{1,0}(t)\right)\, dt - \int_{3\tau/2}^{2\tau} \cos\left(\Phi_{1,0}(t)\right)\, dt \\ &= \frac{\tau}{2}\left[1 + (-1)^{m_2}\right] - \int_{\tau}^{3\tau/2} \cos\left(\Phi_{1,0}(t)\right)\, dt - \int_{3\tau/2}^{2\tau} \cos\left(\Phi_{1,0}(t)\right)\, dt \qquad (5.19) \\ &= \frac{\tau}{2}\left[1 + (-1)^{m_2} - (-1)^{m_3 - m_2} - (-1)^{m_4 - m_3}\right]. \end{aligned}$$

Working from left to right, to maximize the gap, we take $m_2 = 0$ to make the second term positive, and then take $m_3 = 1$ to make the third term positive. But then we have to take $m_4 = 0$ to make the last term positive.

In summary, our phases are $(\delta, 2\delta, \pi + 3\delta, 4\delta)$ as we cycle through the regions. So we have oscillatory behavior in ϕ when we are sending a 1 bit. This is consistent with Fig. 13, where one sees more oscillations in the 1 bits. In particular, we see the following behavior:

$$\text{every } \Delta t\text{: add } \delta \text{ to } \phi; \qquad \text{in each 1 bit: alternate } \phi \text{ by } \pm\pi \text{ every } \Delta t. \tag{5.20}$$

Note that (5.13) also exhibits this behavior.

5.3 $\Delta t = 2\tau/3$

To illustrate variational solutions further, we briefly describe the case of $\Delta t = 2\tau/3$. Considerations similar to the above show that the simplest gap-maximizing solution is of the form

$$\phi(t) = \begin{cases} 0, & t < 0, \\ \delta, & t \in \frac{\tau}{3}(0, 2), \\ 2\delta, & t \in \frac{\tau}{3}(2, 3), \\ 2\delta + \pi, & t \in \frac{\tau}{3}(3, 4), \\ 3\delta + \pi, & t \in \frac{\tau}{3}(4, 5), \\ 3\delta, & t \in \frac{\tau}{3}(5, 6), \end{cases} \tag{5.21}$$

which also matches the behavior in (5.20). For completeness, we note that we could add any even multiple of π to our solutions above, but that would drive us far from the desired range of $\phi \in [0, \pi]$.

5.4 Constrained Optimization, $\Delta t = \tau/2$

As discussed previously, the results in the previous sections drift slightly outside the desired interval $[0, \pi]$. But with additional bits, $n\delta + \pi$ would eventually drift to larger values. Therefore, we consider the case when ϕ is constrained to lie within $[0, \pi]$ in the case in §5.2 where $\Delta t = \tau/2$.

Since the critical point of the functional lies outside $[0, \pi]$, we simply compute g at the endpoints to see which is best. This discussion doesn't affect region 2, so (5.19) still holds with $m_2 = 0$. Then we have the following points to test in region 3:

$$\begin{aligned}
&\phi = 0, \quad \Phi_{1,0}(t) = 3\delta, \quad \cos(\Phi_{1,0}(t)) \approx 1, \\
&\phi = \pi, \quad \Phi_{1,0}(t) = 3\delta - \pi, \quad \cos(\Phi_{1,0}(t)) \approx -1.
\end{aligned}$$

Hence we take $\phi = \pi$ in region 3 to maximize g. But then in region 4, we have

$$\sin(\delta - \phi + \pi) = 0, \qquad \phi(t) = \delta + m_4\pi, \qquad \Phi_{1,0}(t) = (m_4 - 1)\pi,$$

so we take $m_4 = 0$ to maximize the fourth term.

In summary, our phases are $(\delta, 2\delta, \pi, \delta)$ as we cycle through the regions. Note that (as expected) the maximum in this case is not as large as in the unconstrained case because we don't achieve the optimal value of -1 in region 3. However, because of the oscillatory behavior of ϕ, we are able to achieve the optimal value in region 4.

6 Regularization, $\Delta t = \tau/2$

Though we have computed an optimal solution for ϕ, it is discontinuous, which cannot be implemented in hardware, as discussed in §4.2. Therefore, we introduce a regularization term to the gap in (5.6) which will penalize nonsmooth functions:

$$G = a_1 - a_2 - r_1 r_2 \alpha \int_0^{2\tau} (\dot{\phi})^2 \, dt. \tag{6.1}$$

Here $\alpha > 0$ is a regularization parameter as before, and we have chosen the coefficient of the integral for later algebraic simplicity.

Since we wish to maximize G, the penalty is negative and we must avoid large values of $\dot{\phi}$. We again take $\tau = 2\Delta t$ for simplicity. Most of the analysis proceeds as in §5.2, with the addition of the extra terms:

$$\begin{aligned}
G[\phi + \epsilon v] &= g[\phi + \epsilon v] - r_1 r_2 \alpha \int_0^{2\tau} (\dot{\phi} + \epsilon \dot{v})^2 \, dt \\
\delta G &= \delta g - r_1 r_2 \alpha \int_0^{2\tau} 2\dot{v}\dot{\phi} \, dt = \delta g - 2 r_1 r_2 \alpha \left\{ [v\dot{\phi}]_0^{2\tau} - \int_0^{2\tau} v\ddot{\phi} \, dt \right\}.
\end{aligned} \tag{6.2}$$

To close the system, we need to specify boundary conditions that will make the bracketed term disappear. Given the calibration sequence, we expect that

$$\phi(0) = 0 \qquad \Longrightarrow \qquad v(0) = 0, \tag{6.3a}$$

which makes one of the terms disappear. We don't know *a priori* what the value of ϕ should be at $t = 2\tau$, so we use the natural boundary condition that

$$\dot{\phi}(2\tau) = 0, \tag{6.3b}$$

which eliminates the remaining bracketed term.

Continuing to simplify using (5.10b), we obtain

$$\frac{\delta G}{2r_1 r_2} = \int_0^{2\tau} v(t) \left\{ [1 - 2H(t-\tau)] \sin(\delta - \phi + \phi_-) + h(t) \sin(\delta - \phi_+ + \phi) + \alpha\ddot{\phi} \right\} dt, \tag{6.4}$$

which is the analog of (5.10a). Hence at the next step we simply add a factor of $\alpha\ddot{\phi}$ to each of the equations in (5.15):

$$\alpha\ddot{\phi} + \sin(\delta - \phi) - \sin(\delta - \phi_+ + \phi) = 0, \quad 0 < t < \tau/2, \quad \text{(region 1)} \tag{6.5a}$$

$$\alpha\ddot{\phi} + \sin(\delta - \phi + \phi_-) + \sin(\delta - \phi_+ + \phi) = 0, \quad \tau/2 < t < \tau, \quad \text{(region 2)} \tag{6.5b}$$

$$\alpha\ddot{\phi} - \sin(\delta - \phi + \phi_-) + \sin(\delta - \phi_+ + \phi) = 0, \quad \tau < t < 3\tau/2, \quad \text{(region 3)} \tag{6.5c}$$

$$\alpha\ddot{\phi} - \sin(\delta - \phi + \phi_-) = 0, \quad 3\tau/2 < t < 2\tau. \quad \text{(region 4)} \tag{6.5d}$$

Unfortunately, none of these equations are integrable because of the phase-shifted terms.

6.1 A More Restrictive Case

To reduce the complexity of (6.5), recall that the unregularized form (5.15) simplifies to (5.17). We use this fact to motivate the following simplification. Instead of coupling ϕ (and hence v) through the delayed argument, we treat ϕ and ϕ_- as *independent* functions, which means that the variations v and $v_- \equiv v(t - \Delta t)$ are independent as well. Hence (5.8) becomes

$$\frac{\delta g}{2r_1 r_2} = \int_0^{2\tau} [1 - 2H(t-\tau)][v(t) - v_-(t)] \sin(\delta - \phi(t) + \phi_-(t))\, dt. \tag{6.6}$$

With this change, it is sufficient that the sine term in the integrand vanishes. Then expanding out the Heaviside term and using the fact that $\phi_- = 0$ in region 1, we again obtain (5.17).

Though we end up in the same place, for the purposes of this section it is important to delineate *how* we got there. In §5.2, we did the problem "correctly"—that is, v was considered to be the only variation, and it was just shifted. This restricted the set of possible variations, which created the permissive set of conditions in (5.15). Then because of the structure of the solution, (5.15) reduced to (5.17).

In contrast, in this section we widened the set of possible variations by letting v and v_- be independent of one another. That in turn narrowed the possible set of solutions, so each sine term *had to be* satisfied individually as in (5.17). Hence in general the optimizer satisfying (5.17) may not provide as large a gap value as the one produced by satisfying (5.15).

To exploit this approach, we add a penalty term similar to the one in (6.1), but one which exploits the independence of v and v_-. Therefore, we replace (6.1) with

$$G = a_1 - a_2 - r_1 r_2 \alpha \int_0^{2\tau} (\dot{\phi} - \dot{\phi}_-)^2 \, dt, \tag{6.7}$$

which means that solutions are penalized if they are not smooth, or they don't vary from the phase before by more than a constant. Taking the variation of (6.7), we obtain

$$\begin{aligned}
G[\phi + \epsilon v + \epsilon v_-] &= g[\phi + \epsilon v] - r_1 r_2 \alpha \int_0^{2\tau} (\dot{\phi} + \epsilon \dot{v} - \dot{\phi}_- - \epsilon \dot{v}_-)^2 \, dt \\
\delta G &= \delta g - r_1 r_2 \alpha \int_0^{2\tau} 2(\dot{v} - \dot{v}_-)(\dot{\phi} - \dot{\phi}_-) \, dt \\
&= \delta g - 2 r_1 r_2 \alpha \left\{ [(v - v_-)(\dot{\phi} - \dot{\phi}_-)]_0^{2\tau} - \int_0^{2\tau} (v - v_-)(\ddot{\phi} - \ddot{\phi}_-) \, dt \right\} \\
&= \delta g - 2 r_1 r_2 \alpha \left\{ (v - v_-)(\dot{\phi} - \dot{\phi}_-)(2\tau) - \int_0^{2\tau} (v - v_-)(\ddot{\phi} - \ddot{\phi}_-) \, dt \right\},
\end{aligned} \tag{6.8}$$

where we have used (6.3a) and the fact that the same initial condition would hold for v_-. Then zeroing out the remaining term, we have

$$\dot{\phi}(2\tau) = \dot{\phi}_-(2\tau) = \dot{\phi}(3\tau/2), \tag{6.9}$$

which is an unusual condition, but one which we simplify in the next section.

Substituting δg from (6.6) into (6.8) and using (6.9), we obtain the following:

$$\frac{\delta G}{2 r_1 r_2} = \int_0^{2\tau} [v(t) - v_-(t)] \left\{ [1 - 2H(t - \tau)] \sin(\delta - \phi + \phi_-) + \alpha(\ddot{\phi} - \ddot{\phi}_-) \right\} dt, \tag{6.10}$$

which means we simply add a factor of $\alpha(\ddot{\phi} - \ddot{\phi}_-)$ to each of the equations in (5.17):

$$\begin{aligned}
\alpha \ddot{\phi} + \sin(\delta - \phi) &= 0, \quad 0 < t < \tau/2, \quad \text{(region 1)} & \text{(6.11a)} \\
\alpha(\ddot{\phi} - \ddot{\phi}_-) + \sin(\delta - \phi + \phi_-) &= 0, \quad \tau/2 < t < \tau, \quad \text{(region 2)} & \text{(6.11b)} \\
\alpha(\ddot{\phi} - \ddot{\phi}_-) - \sin(\delta - \phi + \phi_-) &= 0, \quad \tau < t < 3\tau/2, \quad \text{(region 3)} & \text{(6.11c)} \\
\alpha(\ddot{\phi} - \ddot{\phi}_-) - \sin(\delta - \phi + \phi_-) &= 0, \quad 3\tau/2 < t < 2\tau, \quad \text{(region 4)} & \text{(6.11d)}
\end{aligned}$$

where we note that $\phi_- \equiv 0$ in region 1. (Even though the equations in regions 3 and 4 are identical, we list them separately since the choice of optimizer will be different.)

We begin by treating region 1:

$$\begin{aligned}
\alpha \dot{\phi} \ddot{\phi} + \sin(\delta - \phi) \dot{\phi} &= 0, \quad \text{(region 1)} \\
\frac{\alpha(\dot{\phi})^2}{2} + \cos(\delta - \phi) &= \text{constant},
\end{aligned} \tag{6.12a}$$

where the constant is determined by the boundary conditions (to be discussed later). We also note that by defining $\phi_\Delta = \phi - \phi_-$, (6.11b) reduces to (6.11a) in the variable ϕ_Δ, so (6.12a) becomes

$$\frac{\alpha(\dot{\phi}_\Delta)^2}{2} + \cos(\delta - \phi_\Delta) = \text{constant}, \quad \text{(region 2)} \tag{6.12b}$$

and similarly

$$\frac{\alpha(\dot{\phi}_\Delta)^2}{2} - \cos(\delta - \phi_\Delta) = \text{constant.} \quad \text{(regions 3, 4)} \tag{6.12c}$$

6.2 Initial Layer

The solutions in §5.2 were quite smooth except for the jumps at Δt. And in the limit that $\alpha \to 0$, we get the unregularized terms of the previous sections. So we are motivated to introduce boundary layers. As proof of concept, we introduce an initial layer near $t = 0$:

$$\phi(t) = \phi_0(y_0), \qquad y_0 = \frac{t}{\alpha^{1/2}}, \quad y_0 > 0. \tag{6.13}$$

The boundary condition at $y_0 = 0$ is given by (6.3a), while the solution must match to the outer solution $\phi = \delta$ as $y_0 \to \infty$:

$$\phi_0(0) = 0, \quad \phi_0(\infty) = \delta. \tag{6.14}$$

Since the solution asymptotes to a constant, $\dot{\phi}_0(\infty) = 0$, so (6.12a) becomes

$$\frac{1}{2}\left(\frac{d\phi_0}{dy_0}\right)^2 + \cos(\delta - \phi_0) = 1 \tag{6.15a}$$

$$\frac{1}{4}\left(\frac{d\phi_0}{dy_0}\right)^2 = \frac{1 - \cos(\delta - \phi_0)}{2} = \sin^2\left(\frac{\delta - \phi_0}{2}\right).$$

We expect ϕ_0 to be monotonic in the interval $[0, \delta]$, so we may take the positive square root in the above:

$$\frac{1}{2}\frac{d\phi_0}{dy_0} = \sin\left(\frac{\delta - \phi_0}{2}\right). \tag{6.15b}$$

To make the algebra somewhat simpler, we let

$$\phi(y_0) = \delta - 2\tan^{-1}\psi(y_0) \qquad \Longrightarrow \qquad \psi(0) = \tan\frac{\delta}{2}. \tag{6.16}$$

(Note that with this definition, $\psi \geq 0$ and we are restricted to the case where $0 \leq \delta \leq \pi$, which is fine since $0 < \delta \ll 1$ in practice.)

Substituting (6.16) into (6.15b) and simplifying yields

$$-\frac{1}{1+\psi^2}\frac{d\psi}{dy_0} = \sin\bigl(\tan^{-1}\psi(y_0)\bigr) = \frac{\psi}{\sqrt{1+\psi^2}}$$

$$-\frac{1}{\psi\sqrt{1+\psi^2}}\,d\psi = dy_0$$

$$\int -\frac{1}{\operatorname{csch} u\sqrt{1+\operatorname{csch}^2 u}}(-\operatorname{csch} u \coth u)\,du = y_0 + c, \qquad \psi = \operatorname{csch} u$$

$$\int \frac{\coth u}{\sqrt{\coth^2 u}}\,du = y_0 + c$$

$$u = y_0 + c, \tag{6.17}$$

where we have used the fact that $\psi \geq 0$ implies $u \geq 0$, which implies that $\coth u \geq 0$.

Continuing to simplify, we obtain

$$\psi = \operatorname{csch}(y_0 + c) \tag{6.18a}$$

$$\tan\frac{\delta}{2} = \operatorname{csch} c$$

$$c = \operatorname{csch}^{-1}\left(\tan\frac{\delta}{2}\right)$$

$$\phi(y_0) = \delta - 2\tan^{-1}\left(y_0 + \operatorname{csch}^{-1}\left(\tan\frac{\delta}{2}\right)\right). \tag{6.18b}$$

A graph of (6.18b) is shown in Fig. 17.

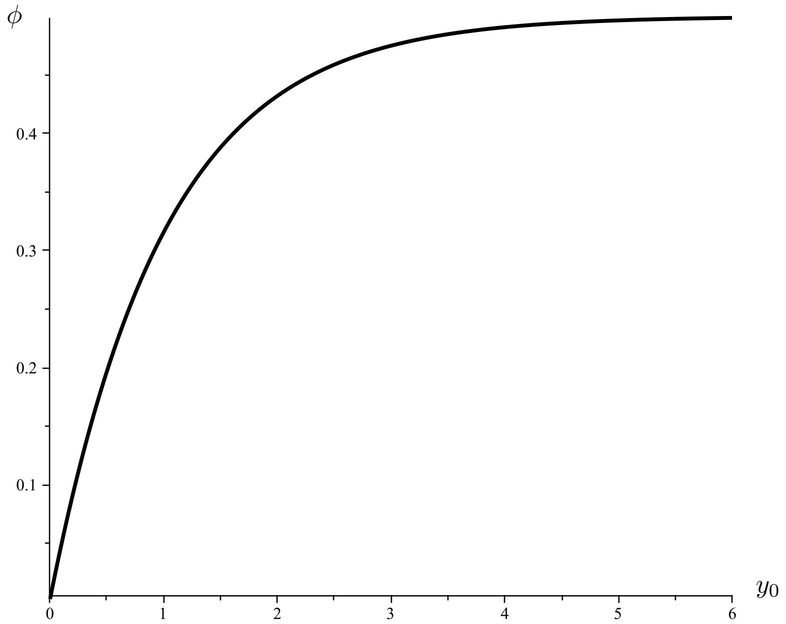


Figure 17. Initial layer solution with $\delta = 0.5$.

6.3 Other Layers

A layer will exist whenever t is a multiple of Δt, both because of the change in outer solutions as well as the change in equations. Though we ran out of time before we could derive the solutions in detail, we present the general algorithm. If we let

$$\phi(t) = \phi_1(y_1), \qquad y_1 = \frac{t - \Delta t}{\alpha^{1/2}}, \quad -\infty < y_1 < \infty;$$

$$\phi_\Delta(t) = \phi_{1\Delta}(y_1), \qquad y_1 > 0,$$

then there will be two equations, depending on the sign of y_1:

$$\frac{1}{2}\left(\frac{d\phi_1}{dy_1}\right)^2 + \cos(\delta - \phi_1) = \text{ constant}; \qquad y_1 < 0, \quad \phi_1(-\infty) = \delta, \tag{6.19a}$$

$$\frac{1}{2}\left(\frac{d\phi_{1\Delta}}{dy_1}\right)^2 + \cos(\delta - \phi_{1\Delta}) = \text{ constant}; \qquad y_1 > 0, \quad \phi_{1\Delta}(\infty) = 2\delta - \delta = \delta. \tag{6.19b}$$

These two equations are coupled by requiring that ϕ_1 and $\dot{\phi}_1$ be continuous at the patch point $y_1 = 0$.

Equation (6.19b) is exactly in the form of (6.15a) with subscript 0 replaced with subscript 1Δ. Therefore, the solution must be of the form (6.18a), with c chosen to satisfy the continuity conditions. When $y_1 < 0$, $\phi_1 > \delta$, so the operator in (6.15b) would hold for ϕ_1 with a change of sign. Again, the choice of constants would be made to satisfy the continuity conditions.

7 Conclusions and Further Research

The advent of laser transmission equipment has enabled cost-effective transmission of signals to space vehicles. The signal is encoded into energy levels: a 0 bit corresponds to a high energy level, while a 1 bit corresponds to a low energy level. By creating a large gap between "high" and "low" energies, the receiver can still distinguish between zeroes and ones even when noise is introduced to dispersion effects.

By adapting codes provided by Raytheon, we first implemented a global optimizer for the gap using the functional in (3.8). This approach optimizes the gap for a particular bit string sequence as shown in Fig. 8, and includes filtering and device limitation effects.

We then implemented a simpler optimization procedure that optimized the energy level for a given bit locally, with the hope that the results would still produce large gap values over the entire string. We first constrained $\phi \in [0, \pi]$ but allowed for instantaneous changes in the signal, which produced a large gap (see Fig. 14). However, these instantaneous changes aren't allowed by the device design, so we introduced another more realistic algorithm that included filtering while still preserving the gap size (see Fig. 16).

Our numerical solutions are robust and can handle a wide variety of cases. But to understand the underlying dynamics of the problem better, we also analyzed the problem using a variational approach. This necessitated looking at very short bit strings and limited path reflections, and restricting to the case where $\Delta t/\tau$ is rational.

In the case with no regularization, we discovered a simple rule given by (5.20) for the optimum signal. With regularization added, the problem can still be solved analytically using singular perturbation methods.

In this manuscript we have described three different techniques for maximizing the gap between the energy levels for 0 and 1 noiseless transmitted bits. These results should provide a roadmap for Raytheon to optimize the design of their etalon demodulator.

7.1 Further Research

When performing the optimization, we either assumed that the etalon could generate signals of any shape, or treated this consideration as secondary. The question of how the finite bandwidth of the etalon affects the transmitted signals (and hence the gap) needs to be further investigated. In addition, we must take into consideration the instability of the etalon and the influence of the receiver bandwidth on the apparent time shift of the symbol.

In all of our work, we assumed that δ is constant; in reality it varies randomly within a bound of $\pm 0.1\pi$. The effects of this variation on the gap need to be analyzed. However, we note that the bandwidth of these variations is below 10 Hz (it is mostly thermal) while transmission is in the GHz domain. Thus, there might be techniques that permit sending the information on the existing δ back to the transmitter to make the adjustment if the transmission range is not very long [Dolgin, 2024].

The boundaries of the bit, as received, are delayed with respect to the boundaries of the transmitted signal due to the bandwidth of the receiver. (This extra delay is close to the reciprocal of twice the filter bandwidth.) These changes were not included in the model, and experimentally it is known that such a shift typically increases the gap [Dolgin, 2024].

In the variational approach with regularization added, time ran out before we could solve the problem completely. However, we have outlined the basic solution method, and it should be straightforward to complete the solution.

Another area of future research concerns the representation of $\phi(t)$. Recall that in the numerical approaches in this manuscript, we represented $\phi(t)$ as a vector of samples as in (3.4). However, we could also express $\phi(t)$ as a series of Fourier modes:

$$\phi(t) = \sum_{n=-N/2}^{N/2} c_n \exp\left(\frac{2\pi i n t}{M\tau}\right). \tag{7.1}$$

It is possible that given the form of (2.5), this form may have advantages in speeding computation. However, if the optimal function is really of a form similar to that in Fig. 7, then the discontinuities will force slow convergence of the c_n, again forcing large numbers of unknowns.

By optimizing the gap for noiseless data, we have confidence that our results will provide a large enough gap for noisy data. However, this can only be concretely verified by adding noise to the signal and then using our algorithms. In particular, one would like to establish a signal-to-noise ratio for which our gap still holds.

Nomenclature

If a symbol appears both with and without tildes, the symbol with tildes has units, while the one without is dimensionless. Equation numbers where a variable is first defined is listed, if appropriate.

$A(t)$: amplitude of output signal.

$\tilde{a}_j$: energy measurement for jth bit (2.8).

$\mathcal{B}$: backward shifting operator (2.4).

$B(t)$: reflected signal (4.6a).

c_n: Fourier coefficient (7.1).

$f_\cdot$: factor in recursion relation (2.3a).

G: regularized optimized function (6.1).

$\tilde{g}$: gap in energy levevls (2.9).

$h(t)$: shifting function in variational method (5.10b).

$\tilde{I}(t)$: intensity (2.6).

$J.$: set of bits with a particular value (2.9).

j: bit index (2.2).

k: index of shifts (2.3a).

M: number of bits (3.4).

m: arbitrary integer (5.13).

N: number of Fourier modes (7.1).

n: index of position (3.4) or Fourier mode (7.1).

R: magnitude of scaled shifted amplitude (4.1).

$r.$: reflection coefficient in the etalon.

$S(t)$: signal function (2.1).

s: number of samples per bit (3.4).

$T_{\mathrm{t}}(t)$: received transmission from etalon (4.6b).

t: time (2.1).

u: dummy variable.

$v(t)$: variation function (5.8).

α: regularization parameter (3.8).

Δt: transit time of one reflection in the etalon.

δ: phase shift parameter in the etalon.

θ: phase of scaled shifted amplitude (4.1).

τ: bit width (2.2).

$\Phi_{k,n}(t)$: phase function in variational principles formulation (5.1b).

$\boldsymbol{\phi}$: vector of ϕ samples (3.4).

$\phi(t)$: phase function (2.1).

$\psi(t)$: simplifying function in variational approach (6.16).

Other Notation

u: as a subscript, used to represent an unfiltered quantity.

Δ: as a subscript on ϕ, refers to the difference between measurements at different times (6.12b).

0: as a subscript on a dimensional variable, used to refer to a normalization factor (2.7a).

$*$: as a superscript, used to represent an optimum (3.8) or a complex conjugate.

$-$: as a subscript, used to represent back one Δt (5.9b).

$+$: as a subscript, used to represent ahead one Δt (5.9b).

References

[Binh, 2008] Binh, L.N. (2008). *Digital optical communications.* Optics and Photonics. CRC Press.

[Davies, 1970] Davies, W.D.T. (1970). *System identification for self-adaptive control.* Wiley-Interscience.

[Dolgin, 2024] Dolgin, Ben. 2024 (October). Private communication.

Appendix A

For τ, we were given a duration of

$$\tau = 150 \text{ ps.} \tag{A 1}$$

For the values of Δt, we were first given a value of

$$\Delta t = 0.6\tau, \tag{A 2a}$$

which caused the overlap problems in §5. Later, we were then given a value of

$$\Delta t = 0.5\tau, \tag{A 2b}$$

which allows the simplification in §5.2. For the etalon factors, we were given the following values:

$$0.4 \leq r_{1,2}^2 \leq 0.5 \qquad \delta = (2\pi)(0.05) = 0.314. \tag{A 3}$$