Synchronization for Optical PPM with Inter-Symbol Guard Times

Ryan Rogalin* and Meera Srinivasan†

Deep space optical communications promises orders of magnitude growth in communication capacity, supporting high data rate applications such as video streaming and high-bandwidth science instruments. Pulse position modulation is the modulation format of choice for deep space applications, and by inserting inter-symbol guard times between the symbols, the signal carries the timing information needed by the demodulator. Accurately extracting this timing information is crucial to demodulating and decoding this signal. In this article we propose a number of timing and frequency estimation schemes for this modulation format, and in particular highlight a low complexity maximum likelihood timing estimator that significantly outperforms the prior art in this domain. This method does not require an explicit synchronization sequence, freeing up channel resources for data transmission.

I. Introduction

Deep space optical communications promises orders of magnitude growth in communication capacity, supporting high data rate applications such as video streaming [1]. One method of representing digital information in this regime is pulse position modulation (PPM), where information is encoded in one of M time slots to denote a binary sequence of length $\log_2 M$. Clearly, being able to resolve the correct time slot in which a pulse was transmitted is crucial to the operation of a PPM system, and in order to do so, an estimate of the timing offset at the receiver must be made.

The methods of timing estimation described in this article exploit the use of inter-symbol

*Communications Architectures and Research Section

†Flight Communications Systems Section

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

guard times (ISGTs) to infer this relative timing offset. Inter-symbol guard times are advantageous in optical PPM communication systems due to their elimination of back-to-back laser pulses [2]. Because there are no signal pulses transmitted during these ISGTs, the identification of the timing offset may be made by determining the time slots during which no signal pulses are present.

The primary contribution of this article is a low-complexity maximum-likelihood timing estimator for PPM+ISGT optical links developed in Section III [3]. Existing timing estimators of PPM+ISGT photon counting detectors exploit first moment properties of the arrival process, but do not make claims of optimality [2]. The next closest known result to this estimator is a similar approach based on using dead times before and after each signal pulse rather than between PPM symbols [4]. Maximum likelihood results for multipulse PPM (MPPM) and overlapping PPM (OPPM) are known, but they report high complexity and the OPPM method relies on the existence of a primary synchronization that occurs before slot synchronization [5, 6]. General maximum likelihood estimators of optical pulse arrival times are also known [7], but they are not directly useful for communication synchronization due to their complexity. A novel coding-theoretic based method has also been presented [8], but it relies on an explicit sequence for synchronization. The maximum likelihood scheme presented in this article is of $\mathcal{O}(M)$ complexity, yet as shown in Section IV achieves the Cramér-Rao bound even in relatively low flux scenarios. Further, the approach outlined in this article can be utilized with an explicit synchronization sequence as well as using just the data itself (data-driven synchronization), which is analyzed in Section V. This is particularly important in low flux scenarios, in which transmitting a sufficiently long synchronization sequence may severely impact the data throughput.

We additionally develop an even lower complexity hybrid correlation-ML scheme in Section III which matches the performance of the ML scheme in high flux scenarios. In Section VI we also develop an impractically high-complexity minimum mean square error (MMSE) estimator based on Markov Chain Monte Carlo (MCMC) methods to serve as a benchmark. Finally, we demonstrate how any of the developed estimators may be fed into standard phase locked loops for frequency and phase tracking in Section VII.

II. Model

The synchronization schemes outlined in this article assume the use of pulse position modulation with an inter-symbol guard time. In conventional PPM, information is encoded in the timing position of an optical pulse. For a symbol of length T_{sym}, the symbol is divided into M slots of length $T_{slot} = T_{sym}/M$, and messages of length $\log_2 M$ are encoded by sending an optical pulse in one of the M slots. With the addition of an ISGT, the symbol is divided into $M + P$ slots of length $T_{slot} = T_{sym}/(M + P)$, and messages of length $\log_2 M$ are encoded by sending an optical pulse in one of the first M slots, while the remaining P slots do not contain signal pulses. An example of 8-PPM is shown in Figure 1 in which a pulse is transmitted

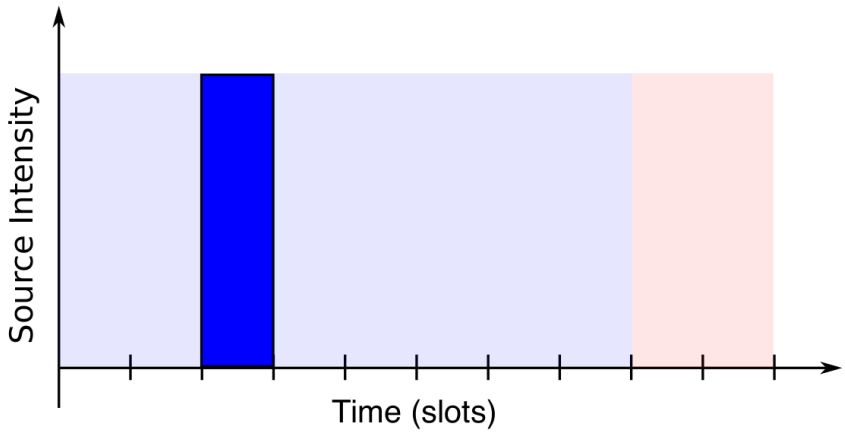

Figure 1. Format of a pulse-position modulation symbol with inter-symbol guard times for modulation order $M = 8$ and ISGT length $P = 2$. A pulse is contained in one of the first eight slots, whereas the last two slots never contain a pulse. This PPM symbol contains a pulse in the second slot (indexing from zero).

in the second slot (indexing from zero). Note that the selection of the parameter P depends upon a combination of synchronization requirements, hardware constraints and data rates, and is outside the scope of this article.

We assume the use of an optical detector assembly that provides the number of photo-electon counts per given time interval. The output process of such a detector is well-modeled as a Poisson point process whose mean number of counts is proportional to the incident light on the detector [9]. This mean number of counts may be non-linearly proportional to the incident light due to phenomena such as blocking [10]. We observe the detector process $x[n]$ over N symbols, and assume a single sample per slot is measured. For Poisson distributed counts, the sum of the counts per slot provides a sufficient statistic for the timing offset, justifying the use of a single sample per slot. In practice, samples are summed over the slot duration to generate the single slot statistic. Thus the indices of $x[n]$ range in $n \in \{0, \ldots, (M+P)N-1\}$. The time over which we collect counts, or the integration time, is given by $T_{int} = N(M+P)T_{slot}$. The variable N may be increased to collect signal flux for a sufficient amount of time in order to be able to distinguish signal slots from ISGT slots. The observations are binned into a vector $\mathbf{y}$ such that $y_m = \sum_{i=0}^{N-1} x[m+i(M+P)]$ with $m \in \{0, \ldots, M+P-1\}$. In the following analysis we assume that there is no frequency offset (or equivalently that the frequency offset is known). However, in concordance with other works [2], this algorithm can still operate in the presence of unknown frequency offsets by adjusting the integration time N.

With no timing offset and conditioned on there not being a signal pulse in slot n, the distribution of the observation $x[n]$ is Poisson with parameter K_b, the mean number of background photon counts per slot. With no timing offset and conditioning on there being a signal pulse in slot n, the distribution of the observation $x[n]$ is Poisson with parameter $K_s + K_b$, where K_s is the mean number of signal photon counts per symbol. The condition of knowing whether

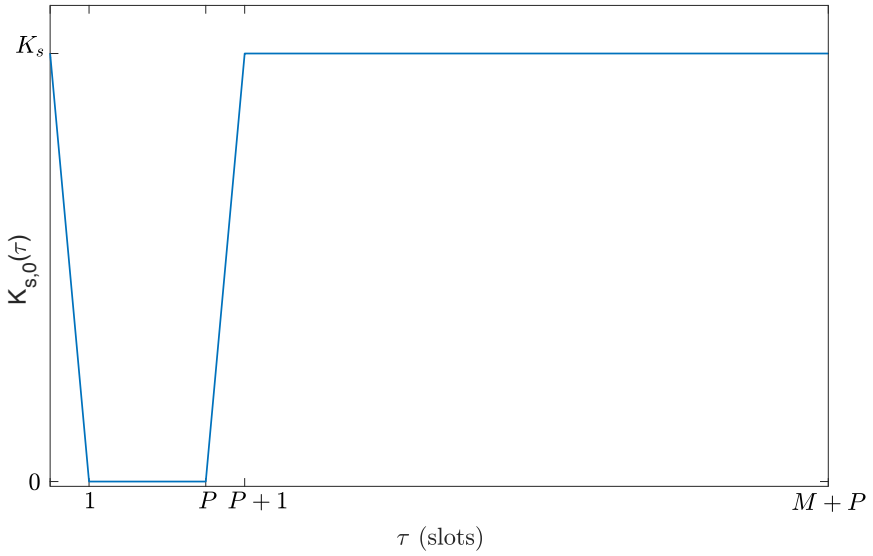

Figure 2. Arrival rate function for the first aggregate slot, y_0 as a function of the timing offset τ.

the slot contains or does not contain a signal pulse is akin to knowing the transmitted pulse sequence (e.g. observing a known training sequence). In this case, because the elements of $\mathbf{y}$ are deterministic sums of Poisson random variables, they are also Poisson distributed. In the case where the symbols are unknown, the elements of $\mathbf{y}$ are no longer Poisson distributed since the number of summed signal slots is now random, as will be discussed in Section V. Assuming that the signal slots are deterministically uniform (i.e. that the same number of pulses appear in each signal slot over the duration of the integration time), then there are $\frac{N}{M}$ aggregated pulses in $y[m]$ for $m \in \{0, \ldots, M-1\}$. Note that this is also true if the particular pulse sequence is unknown, but the number of pulses in each position is deterministically uniform. The aggregate observation vector is then Poisson distributed with parameter $\frac{N}{M}K_s + NK_b$ for $m \in \{0, \ldots, M-1\}$ and Poisson distributed with parameter NK_b for $m \in \{M, \ldots, M+P-1\}$.

In the presence of a timing offset τ the distribution of the observation vector changes. Let $X \sim \text{Poi}(r)$ denote that X is a Poisson random variable with mean r. If τ is integer-valued, then the distributions of the elements of $\mathbf{y}$ are cyclically shifted by τ (e.g., if $\tau = 1$, then $y_m \sim \text{Poi}(\frac{N}{M}K_s + NK_b)$ for $m \in \{1, \ldots, M\}$ and $y_m \sim \text{Poi}(NK_b)$ for $m \in \{M+1, \ldots, M+P-1, 0\}$. If τ contains a fractional component then the means of the two slots adjacent to the ISGT are scaled between the count and background parameters. Decomposing the timing offset $\tau = k + \epsilon$ into integer and fractional components k and ϵ, respectively, we observe that

$$y_m \sim \begin{cases} \text{Poi}((1-\epsilon)\frac{N}{M}K_s + NK_b) & m = k \bmod (M+P) \\ \text{Poi}(\frac{N}{M}K_s + NK_b) & m \in \{k+1, \ldots, k+M-1\} \bmod (M+P) \\ \text{Poi}(\epsilon\frac{N}{M}K_s + NK_b) & m = (k+M) \bmod (M+P) \\ \text{Poi}(NK_b) & m \in \{k+M+1, \ldots, k+M+P-1\} \bmod (M+P). \end{cases} \tag{1}$$

Alternatively, we can write the probability mass function (pmf) of each slot given a timing offset using an intensity function $\lambda_m(\tau)$ which varies with τ:

$$P_{Y_m|\tau}(y_m|\tau) = \frac{\lambda_m(\tau)^{y_m} e^{-\lambda_m(\tau)}}{y_m!}. \tag{2}$$

The intensity function can be written as

$$\lambda_m(\tau) = \frac{N}{M} K_{s,m}(\tau) + NK_b \tag{3}$$

where $K_{s,m}(\tau)$ is the mean signal count in the mth slot in the presence of a timing offset τ. The effective signal count of the zeroth aggregate slot, $K_{s,0}(\tau)$ can be written as a piecewise linear function

$$K_{s,0}(\tau) = \begin{cases} (1-\tau)K_s & \tau \bmod (M+P) \in [0,1] \\ 0 & \tau \bmod (M+P) \in [1,P] \\ (\tau - P)K_s & \tau \bmod (M+P) \in [P, P+1] \\ K_s & \tau \bmod (M+P) \in [P, M+P] \end{cases} \tag{4}$$

and is shown in Figure 2. Subsequent arrival rate functions can be written as cyclic shifts of $K_{s,0}(\tau)$, such that

$$K_{s,m}(\tau) = K_{s,0}((\tau - m) \bmod (M+P)). \tag{5}$$

III. Maximum Likelihood Timing Estimation

Given the pmf of the aggregate observation vector $\mathbf{y}$ given τ, we can form the log likelihood function:

$$\begin{aligned} \ell(\tau; \mathbf{y}) &= \log \prod_{m=0}^{M+P-1} P_{Y_m|\tau}(y_m|\tau) && (6) \\ &= \sum_{m=0}^{M+P-1} \log P_{Y_m|\tau}(y_m|\tau) && (7) \\ &= \sum_{m=0}^{M+P-1} y_m \log \lambda_0((\tau - m) \bmod (M+P)) \\ &\qquad - \lambda_0((\tau - m) \bmod (M+P)) - \log(y_m!) && (8) \\ &= \sum_{m=0}^{M+P-1} y_m \log \lambda_0((\tau - m) \bmod (M+P)) + \text{constant} && (9) \end{aligned}$$

where in the last step we utilize the fact that the sum of these cyclically shifted $\lambda_i(\tau)$ functions is a constant (to see this, note that in the interval $[0,1]$ the function $K_{s,0}(\tau) = (1-\tau)K_s$ and the function $K_{s,M}(\tau) = \tau K_s$, so $K_{s,0}(\tau) + K_{s,M}(\tau) = K_s$; this is true for each integer interval). We wish to maximize this likelihood function with respect to τ, yet we observe that it is not differentiable at integer values due to the piecewise definition of $\lambda_m(\tau)$. However, on

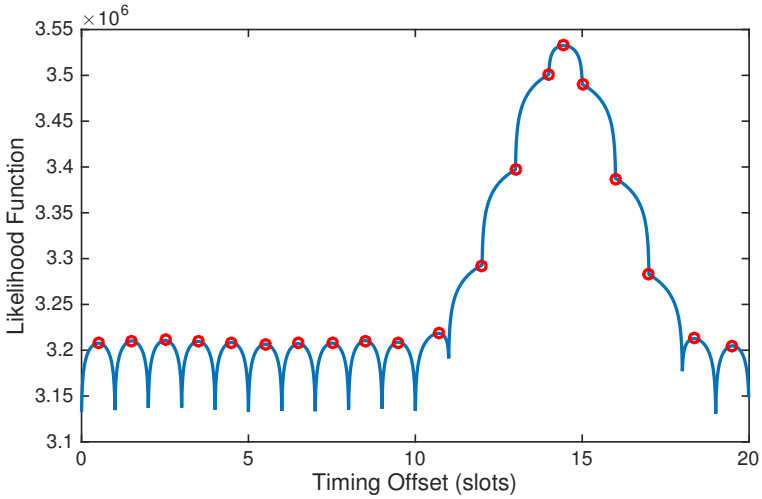

Figure 3. The log likelihood function for an example with modulation order $M = 16$ and intersymbol guard length $P = 4$ slots. The actual timing offset is $\tau = 14.4545$ slots. Each red circle identifies the maximum on that integer interval.

each open integer interval $(j, j+1)$, the likelihood function is both differentiable and concave (since it forms a sum of concave functions of affine functions [11, pp. 79]).

Taking the partial derivative with respect to τ over the interval $(j, j+1)$, we derive

$$\begin{aligned}\frac{\partial}{\partial \tau}\ell(\tau;\mathbf{y}) &= \frac{y_{j+M}\frac{N}{M}K_s}{(\tau-j)\frac{N}{M}K_s + NK_b} - \frac{y_j\frac{N}{M}K_s}{(j+1-\tau)\frac{N}{M}K_s + NK_b} \\ &= 0 \end{aligned} \tag{10}$$

where the indices of the elements of $\mathbf{y}$ are taken modulo $M+P$. Solving for τ results in the estimate

$$\widehat{\tau}_{\mathrm{ML},j} = \frac{\frac{N}{M}K_s \cdot ((j+1)y_{j+M} + jy_j) + NK_b(y_{j+M} - y_j)}{\frac{N}{M}K_s \cdot (y_{j+M} + y_j)}. \tag{11}$$

If the estimate $\widehat{\tau}_{\mathrm{ML},j}$ is less than j, then we let $\widehat{\tau}_{\mathrm{ML},j} = j$, and if $\widehat{\tau}_{\mathrm{ML},j}$ is greater than $j+1$, then we let $\widehat{\tau}_{\mathrm{ML},j} = j+1$. As such, the maximum likelihood estimate can be found by evaluating the $M+P$ quantities $\widehat{\tau}_{\mathrm{ML},j}$ for $j \in \{0, \ldots, M+P-1\}$, then choosing the overall estimate as

$$\widehat{\tau}_{\mathrm{ML}} = \arg\max_{\widehat{\tau}_{\mathrm{ML},j}} \ell(\widehat{\tau}_{\mathrm{ML},j};\mathbf{y}). \tag{12}$$

Thus the timing estimate can be formed by calculating $2(M+P)$ simple algebraic equations and evaluating a maximum over $M+P$ quantities. An example of the log likelihood function can be seen in Figure 3, in which $M = 16$ and $P = 4$, and where the timing offset is $\tau = 14.4545$. The circles denote the maxima on each integer interval, and the maximum of the overall function can be seen to lie approximately at 14.4545.

To provide a basis for performance evaluation, we compare the root mean square error (RMSE, defined as $\sqrt{\mathbb{E}[(\tau - \widehat{\tau})^2]}$) of the ML estimator to that of a correlation-based method [2]. The

correlation method utilizes the same slot count measurements as the ML method. It assumes a single slot is used for the ISGT, so the smallest slot count corresponds to the integer estimate of the timing offset:

$$\widehat{k}_{\text{corr}} = \arg\min_j y_j. \tag{13}$$

The fractional timing offset estimate is then formed from the two adjacent count statistics:

$$\widehat{\epsilon}_{\text{corr}} = \frac{y_{\widehat{k}-1} - y_{\widehat{k}+1}}{\frac{N}{M}K_s}. \tag{14}$$

Since the ML method works for an arbitrary size ISGT, a slight modification to the correlation method is necessary to perform a fair comparison between the methods. For $P > 1$ and assuming that $M + P \bmod P = 0$, we accumulate the slot counts of the P adjacent bins to form a "superslot" count vector $\mathbf{z}$ of length $\frac{M+P}{P}$, i.e. $z_j = \sum_{i=0}^{P-1} y_{jP+i}$. The correlation-superslot method then forms the integer and fractional estimates of the timing offset as:

$$\widehat{k}_{\text{corr-ss}} = P \arg\min_j z_j. \tag{15}$$

and

$$\widehat{\epsilon}_{\text{corr-ss}} = P\frac{z_{\widehat{k}-1} - z_{\widehat{k}+1}}{\frac{N}{M}PK_s}. \tag{16}$$

Finally, we also consider a hybrid scheme which forms the integer estimate from the correlation-superslot method, and the overall estimate using the ML estimate for that particular bin:

$$\widehat{k}_{\text{hybrid}} = \lfloor \widehat{\tau}_{\text{corr-ss}} \rfloor. \tag{17}$$

and

$$\widehat{\tau}_{\text{hybrid}} = \widehat{\tau}_{\text{ML},\widehat{k}}. \tag{18}$$

We will observe the performance of these schemes for two scenarios, the parameters of which are described in Table 1. The first scenario utilizes a low order PPM with moderate signal flux, while the second scenario utilizes higher order PPM with low signal flux and comparatively high background flux. These scenarios simulate high-rate near- Earth, and low-rate deep-space channels, respectively.

The performance of the three schemes in terms of the RMSE is shown in Figure 4 for Scenario 1. Data is generated using the Poisson model outlined in (1) for random timing offsets τ. As can be seen, the performance of the hybrid scheme approaches that of the ML with increasing flux. Depending on the target RMSE, however, one may find one method preferable to another. A target of 10^{-1} slot for instance, may be chosen for its projected impact of less than 0.5 dB in capacity loss [12]. At this target RMSE, the ML scheme requires one-third as much signal flux as the hybrid scheme, and less than one-tenth as much signal flux as the correlation-superslot method.

Table 1. The example scenarios for which the timing estimation performance is simulated

	Scenario 1	Scenario 2
M	16	128
P	4	32
N	10^5	1.25×10^5
T_{slot}	0.5 ns	5 ns
K_s	0.25	0.09
K_b	0.00005	0.001

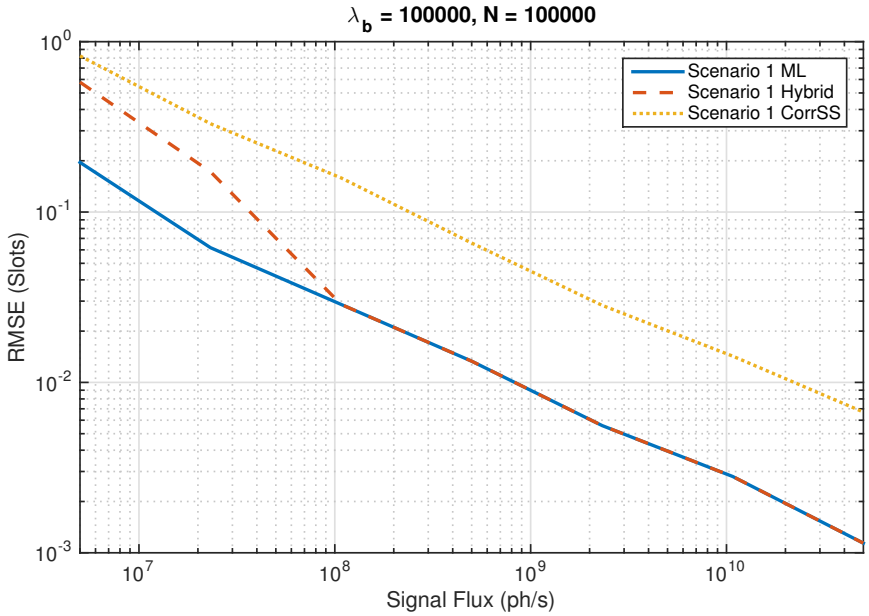

Figure 4. Root mean square error performance of the three estimation schemes for Scenario 1 in the presence of Poisson distributed data as in Equation (1). For this scenario, the Hybrid scheme can be seen matching the ML.

IV. Timing Estimate Cramér-Rao Bound

To quantify how optimal these estimators are with respect to mean square error, we can derive the Cramér-Rao lower bound (CRB) on the estimation variance. We note that strictly speaking, the non-differentiability of the likelihood function means the CRB does not exist. However, because it is differentiable at non-integer values and the probability of observing an integer-valued timing offset is zero, it may still prove useful as a performance metric. We begin by calculating the Fisher information:

$$\mathcal{I}(\tau) = -\mathbb{E}\left[\frac{\partial^2}{\partial\tau^2}\ell(\tau;\mathbf{y})\right] \tag{19}$$

$$= -\mathbb{E}\left[\frac{\partial^2}{\partial\tau^2}\sum_{m=0}^{M+P-1}\log\left(P_{Y_m|\tau}(y_m|\tau)\right)\right]. \tag{20}$$

The second derivative of the likelihood function is non-zero for two values of m: $m_1 = \lfloor\tau\rfloor$ and $m_2 = \lfloor\tau\rfloor + M$ (where again these indices are taken modulo $M+P$). Then the Fisher information may be written:

$$\mathcal{I}(\tau) = \mathbb{E}\left[\frac{K_s^2 y_{m_1}}{(K_s(1-\epsilon)+K_bM)^2} + \frac{K_s^2 y_{m_2}}{(K_s\epsilon+K_bM)^2}\right] \tag{21}$$

$$= \frac{K_s^2\left(\frac{N}{M}K_s(1-\epsilon)+NK_b\right)}{(K_s(1-\epsilon)+K_bM)^2} + \frac{K_s^2\left(\frac{N}{M}K_s\epsilon+NK_b\right)}{(K_s\epsilon+K_bM)^2} \tag{22}$$

$$= \frac{K_s^2\left(\frac{N}{M}K_s+2NK_b\right)}{\epsilon(1-\epsilon)K_s^2+MK_sK_b+M^2K_b^2}. \tag{23}$$

Finally, the CRB is defined as:

$$\mathrm{CRB}(\tau) = \mathcal{I}(\tau)^{-1} \tag{24}$$

$$= \frac{\epsilon(1-\epsilon)K_s^2+MK_sK_b+M^2K_b^2}{K_s^2\left(\frac{N}{M}K_s+2NK_b\right)}. \tag{25}$$

It is worth noting that the CRB of the timing estimator decreases as $\mathcal{O}(K_s^{-1})$ and $\mathcal{O}(N^{-1})$, in concordance with similar estimates in the AWGN channel, which fall as $\mathcal{O}(\mathrm{SNR}^{-1})$ and $\mathcal{O}(N^{-1})$ [13]. As can be seen in Figure 5, the maximum likelihood scheme is asymptotically efficient. Since the CRB is dependent upon the timing offset (specifically, it is dependent upon the fractional component of the timing offset), a non-random value ($\tau = 4.5$) was chosen for the results in Figure 5 to make a meaningful comparison.

V. Data-Driven PPM Synchronization

When the sequence of transmitted pulses is unknown or is not deterministically uniform, the aggregated slot counts are no longer Poisson distributed. This is the case when the data sequence is used for synchronization without additional pilots, as is sometimes done in RF OFDM communications by exploiting the cyclostationarity of the cyclic prefix [14].

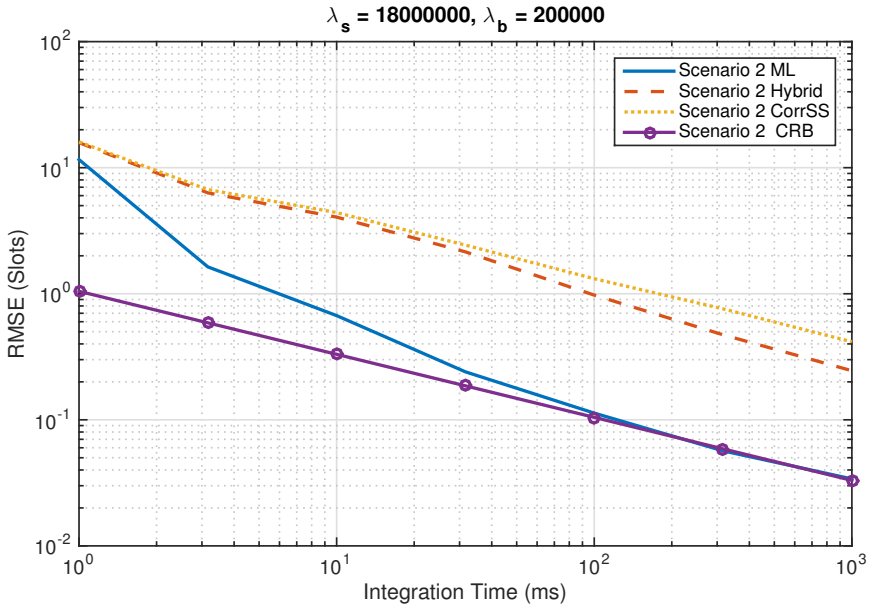

Figure 5. Root mean square error performance of the three estimation schemes with Scenario 2 and the Cramér-Rao bound. In this scenario the ML scheme achieves the CRB with significantly less integration time (with sufficiently long integration time, the Hybrid scheme will match the ML). These simulations used a timing offset $\tau = 4.5$.

Assuming the pulses are uniformly randomly distributed in the M signal-containing slots, we can calculate the aggregate slot count pmf of the signal component explicitly. Let $K_m \sim \text{Binomial}(N, \frac{1}{M})$ be a random variable denoting the number of times a signal pulse appears in the mth slot over N symbols. Then

$$\begin{aligned} P_{Y_m|\tau}(y_m|\tau) &= \mathbb{P}\left(\sum_{j=0}^{N-1} X[m+jN] = y_m \middle| \tau \right) \\ &\overset{(a)}{=} \mathbb{P}\left(\sum_{j=0}^{K_m-1} X[m+jN] + \sum_{j=K_m}^{N-1} X[m+jN] = y_m \middle| \tau \right) \\ &\overset{(b)}{=} \sum_{k=0}^{N} \mathbb{P}(K_m = k) \mathbb{P}\left(\sum_{j=0}^{k-1} X[m+jN] + \sum_{j=k}^{N-1} X[m+jN] = y_m \middle| \tau \right) \end{aligned} \tag{26}$$

where in (a) we have utilized the fact that, without loss of generality, we may assume the K_m signal-containing slots to be in the first K_m symbols, and in (b) we have utilized the law of total probability. By definition, the Binomial random variable yields

$$\mathbb{P}(K_m = k) = \binom{N}{k} \left(\frac{1}{M}\right)^k \left(1 - \frac{1}{M}\right)^{N-k} \tag{27}$$

and the Poisson variable is

$$\mathbb{P}\left(\sum_{j=0}^{k-1} X[m+jN] + \sum_{j=k}^{N-1} X[m+jN] = y_m\right) = \frac{(kK_{s,m}(\tau) + NK_b)^{y_m} \exp(-(kK_{s,m}(\tau) + NK_b))}{y_m!}. \tag{28}$$

We note that even though the individual slot counts are Poisson distributed (for a given τ), the distribution of aggregate slot counts is not Poisson since the mean and variance are not the same. From the law of total variance, the variance of this distribution can be calculated explicitly:

$$\begin{aligned}
\text{var}[Y_m|\tau] &= \mathbb{E}_K\left[\text{var}_{Y_m|K}\left[Y_m|K,\tau\right]\right] + \text{var}_K\left[\mathbb{E}_{Y_m|K}\left[Y_m|K,\tau\right]\right] \\
&= \mathbb{E}_K\left[KK_{s,m}(\tau) + NK_b\right] + \text{var}_K\left[KK_{s,m}(\tau) + NK_b\right] \\
&= \frac{N}{M}K_{s,m}(\tau) + NK_b + K_{s,m}(\tau)^2\frac{N}{M}\left(1 - \frac{1}{M}\right).
\end{aligned} \tag{29}$$

The full distribution of $\mathbf{Y}|\tau$ can be derived by introducing a latent random variable $\mathbf{Z}$ which contains the number of signal pulses in each aggregate signal slot. $\mathbf{Z}$ is multinomially distributed with N trials and parameters $(p_0 = 1/M, \ldots, p_{M-1} = 1/M)$. However, to utilize this latent variable requires conditioning over a set which grows factorially, since

$$P_{\mathbf{Y}|\tau}(\mathbf{y}|\tau) = \sum_{\mathbf{z}|z_0+\cdots+z_M \;\; _1=N} P_{\mathbf{Y}|\tau,\mathbf{Z}}(\mathbf{y}|\tau,\mathbf{z})P_{\mathbf{Z}}(\mathbf{z}) \tag{30}$$

This distribution is much less tractable than the Poisson distribution, and a closed-form solution of the ML scheme is unlikely. Instead of deriving another ML scheme for the above distribution, we will inspect the performance of the ML scheme derived in Section III in the presence of aggregate slot counts given by the actual distribution in (26). We note the resulting RMSE performance of this mismatched scheme in Figure 6. The RMSE exhibits an "error floor" behavior in which increasing the signal flux does not decrease the MSE of the timing estimator. This is in sharp contrast to the ideal Poisson distributed counts, in which increasing the signal flux results in better estimator performance.

To confirm this effect and to dispel any suspicion that this error floor is merely an artifact of simulation, we derive a closed form expression of the MSE of the correlation-superslot method using the bias-variance decomposition:

$$\text{MSE}(\widehat{\tau}) = \text{var}(\widehat{\tau}) + \text{bias}(\widehat{\tau})^2. \tag{31}$$

In particular, we assume we are operating in a high $\frac{K_s}{K_b}$ regime, such that correct detection of the ISGT is possible (i.e. $\widehat{k} = k$). The correlation-superslot method can be easily shown to

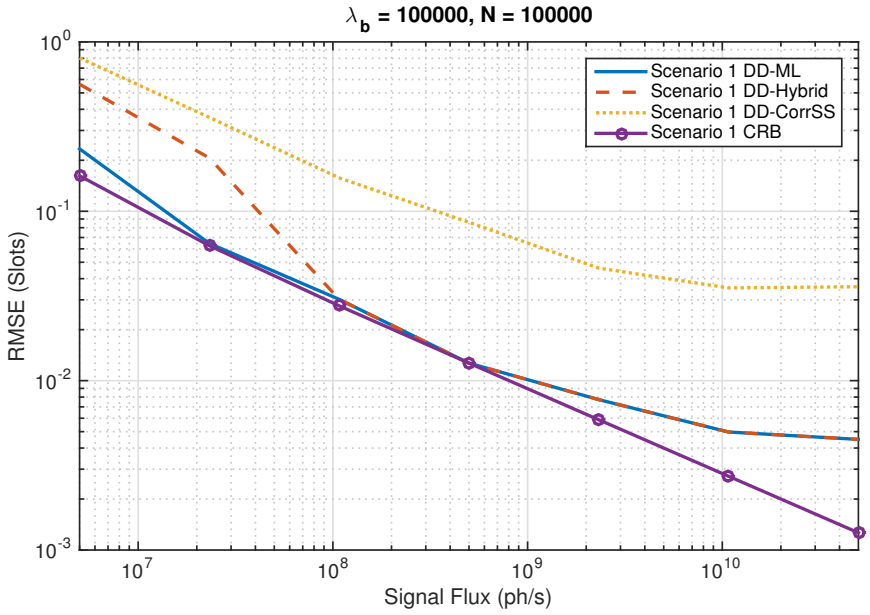

Figure 6. Performance of the three schemes with Scenario 1 when using the mismatched pmf corresponding to unknown, uniformly distributed PPM pulses. The saturation of the RMSE occurs at high signal flux, but can be combatted with longer integration times.

be unbiased:

$$\begin{aligned}\mathbb{E}[\widehat{\epsilon}] &= \mathbb{E}\left[\frac{Y_{k-1}-Y_{k+1}}{N/MK_s}\right] \\ &= \frac{\frac{N}{M}K_s + NK_b - (1-\epsilon)\frac{N}{M}K_s - NK_b}{\frac{N}{M}K_s} \\ &= \epsilon. \end{aligned} \tag{32}$$

Thus the MSE of the estimator is equal to its variance. The variance can be similarly found:

$$\begin{aligned}\mathrm{var}(\widehat{\epsilon}) &= \mathbb{E}\left[\frac{(Y_{k-1}-Y_{k+1})^2}{(\frac{N}{M}K_s)^2}\right] - \epsilon^2 \\ &= \frac{1}{\frac{N^2}{M^2}K_s^2}\big(\mathrm{var}(Y_{k-1}) + \mathbb{E}[Y_{k-1}]^2 + \mathrm{var}(Y_{k+1}) \\ &\quad + \mathbb{E}[Y_{k+1}]^2 - 2\mathbb{E}[Y_{k-1}]\mathbb{E}[Y_{k+1}]\big) - \epsilon^2. \end{aligned} \tag{33}$$

When the aggregate slot counts are Poisson distributed (as in Section III, in which a pilot sequence is used), then

$$\mathrm{MSE}(\widehat{\epsilon}_{\mathrm{pilot}}) = \frac{M(2-\epsilon)}{NK_s} + \frac{M^2K_b}{NK_s^2}. \tag{34}$$

By contrast, when the observed sequence is unknown and the observed counts are subject to the additional variance derived in (29), the MSE is

$$\mathrm{MSE}(\widehat{\epsilon}_{\mathrm{data}}) = \frac{M(2-\epsilon)}{NK_s} + \frac{2M^2K_b}{NK_s^2} + (1+(1-\epsilon)^2)\frac{M-1}{N}. \tag{35}$$

Scheme	RMSE Scenario 1	Computation Time Scenario 1	RMSE Scenario 2	Computation Time Scenario 2
CorrSS [2]	0.0658 slots	30.2 μs	1.32 slots	36.3 μs
Hybrid	0.0133 slots	58.2 μs	0.976 slots	81.9 μs
ML	0.0127 slots	167 μs	0.113 slots	8150 μs
DD-ML	0.0127 slots	167 μs	0.113 slots	8150 μs

Table 2. Performance and running time of the considered timing estimation schemes. Computation does not include the integration time, and should be considered only for relative scale in a software implementation, not an indication of hardware-optimized performance.

The difference between these two regimes is readily apparent; whereas the pilot-driven MSE follows

$$\lim_{K_s \to \infty} \mathrm{MSE}(\widehat{\epsilon}_{\mathrm{pilot}}) = 0, \tag{36}$$

the data-driven MSE tends toward

$$\lim_{K_s \to \infty} \mathrm{MSE}(\widehat{\epsilon}_{\mathrm{data}}) = (1 - (1-\epsilon)^2)\frac{M-1}{N}. \tag{37}$$

However, the MSEs in both cases are proportional to N^{-1}, meaning that this "error floor" can be overcome with longer integration times. In practice, the error floor is only apparent at high signal powers, and the low-flux scenarios in which PPM would be utilized will not be limited by this phenomenon.

A summary of the RMSE performance and computational complexity of the proposed schemes can be found in Table 2. Computation time is calculated as the average time the estimation functions take in MATLAB, and should be considered for relative scale, not indicative of hardware-optimized performance.

VI. MMSE Timing Estimation

While we have shown that the ML algorithm achieves the lowest possible mean square error of an unbiased estimator as the signal flux or integration time tend to infinity, we have not directly attempted to optimize our key metric, the mean square error. In this section we discuss a method which directly minimizes the mean square error, and tends to slightly outperform the ML method, but which has an undesirable computational cost. As such, this estimator is unlikely to be of practical interest. The ML scheme treats the timing offset as a deterministic unknown quantity, but for the sake of curiosity we may view the estimation of τ from a Bayesian perspective. In this case, a point estimate which directly minimizes the mean square error is known as the minimum mean square error (MMSE) estimator. Under some mild regularity conditions, it is defined as

$$\widehat{\tau}_{\mathrm{MMSE}}(\mathbf{y}) = \mathbb{E}[\tau | \mathbf{Y} = \mathbf{y}]. \tag{38}$$

Since this estimator requires knowledge of the distribution of $\tau|\mathbf{Y}$, it is typically considered computationally intractable. The rationale follows from the use of Bayes' Rule:

$$f_{\tau|\mathbf{Y}}(\tau|\mathbf{y}) = \frac{p_{\mathbf{Y}|\tau}(\mathbf{y}|\tau)f_{\tau}(\tau)}{\int_0^{M+P} p_{\mathbf{Y}|\tau}(\mathbf{y}|\widetilde{\tau})f_{\tau}(\widetilde{\tau})d\widetilde{\tau}}. \tag{39}$$

We may assume the distribution of τ is uniform over $[0, M+P)$. The normalization factor $\int_0^{M+P} p_{\mathbf{Y}|\tau}(\mathbf{y}|\widetilde{\tau})f_{\tau}(\widetilde{\tau})d\widetilde{\tau}$ is typically difficult to compute, and must be re-computed for each realization of $\mathbf{Y}$. However, a number of sampling methods which rely only on a function proportional to $f_{\tau|\mathbf{Y}}(\tau|\mathbf{y})$ are used in Bayesian computational methods, allowing one to forgo the repeated numerical integration. The function proportional to the true distribution in our case is $b(\tau) = \prod_{i=1}^{M+ISGT} \lambda_m(\tau)\exp(-\lambda_m(\tau)) \propto p_{\mathbf{Y}|\tau}(\mathbf{y}|\tau)f_{\tau}(\tau)$. Among the more popular methods which exploit this approach is what is known as Markov Chain Monte Carlo (MCMC), and in particular, the Metropolis-Hastings algorithm [15].

Metropolis-Hastings is an iterative sampling algorithm which attempts to construct a Markov Chain with a steady-state probability distribution equal to that of a desired, difficult to calculate distribution. The algorithm is well-covered in standard textbooks [16], though for completeness we outline the approach in Algorithm 1, where we utilize the notation $\mathcal{N}(\mu, \sigma^2)$ to denote generating a Gaussian random variable with mean μ and variance σ^2, and $\mathcal{U}(a, b)$ to denote generating a uniform random variable on the interval $[a, b]$. We also use the function $f(\cdot|\mu, \sigma^2)$ to denote the probability density function of a Gaussian random variable with mean μ and variance σ^2.

Algorithm 1 Markov Chain Monte Carlo Sampling

$\tau_0 \leftarrow \mathcal{U}(0, M+P)$
for $i = 1$ to num_iterations **do**
  $X \leftarrow \tau_{i-1} + \mathcal{N}(0,1)$
  $U \leftarrow \mathcal{U}(0,1)$
  if $\frac{b(X)f(X|\tau_{i-1},1)}{b(\tau_{i-1})f(\tau_{i-1},X,1)} > U$ **then**
    $\tau_i = X$
  else
    $\tau_i = \tau_{i-1}$
  end if
end for

The first samples produced by MCMC are typically not representative of the steady-state distribution of the Markov Chain, and are discarded as part of the "burn-in" period. The remaining samples should have the desired distribution, from which the mean of Equation (38) can be computed by averaging the samples produced by Algorithm 1.

The resulting performance is shown in Figure 7 for $\tau = 4.5$. We note that the MMSE-MCMC algorithm is able to marginally improve upon the RMSE of the ML scheme, though the added computational complexity makes the approach essentially impractical in nature. Additionally,

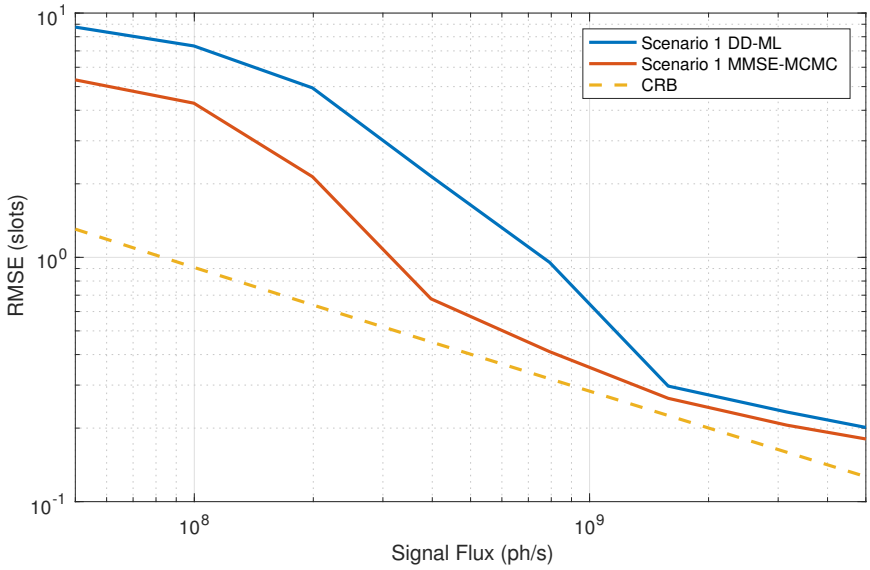

Figure 7. RMSE performance of the MMSE-MCMC scheme compared to the DD-ML method under Scenario 1. In this case 10^4 iterations were utilized with a 10^3 length burn-in period.

as the signal flux or integration time increases, numerical stability becomes an issue since $b(\tau)$ is composed of a product of increasingly larger numbers with small numbers.

VII. Closed Loop Timing Tracking

Estimating a static timing offset is rarely sufficient for a practical system; the timing offset between the transmitter and receiver varies dynamically due to clock drift and doppler. Various methods for tracking the offset over time exist, most of which involve observing a series of timing offset estimates and inferring the rate of phase change (i.e. the frequency offset). For latency-tolerant communications, block-based methods such as least-squares fitting may be appropriate. In this case, the existence of outliers may artificially skew the frequency offset, so outlier pruning [17] may be necessary (alternatively, one may use a cost function which is more robust than the square-error metric, such as the Huber penalty function [11]).

For online methods, tracking loops have a long history of success in RF receivers. A phase locked loop (PLL) can directly exploit the timing estimators discussed in this article. A PLL is composed of three primary components; a phase detector, a loop filter, and a controllable oscillator. Any of the timing algorithms derived in this paper may be used as the phase detector. Standard second and third order loop filters may be designed to track a time-varying phase (i.e a frequency offset) and a time-varying frequency offset (i.e. a frequency drift), respectively. Many standard texts provide insights on how such filters are designed [18]. The oscillator may be implemented either as a voltage-controlled or numerically-controlled oscillator, or if processing purely on digital samples, as a direct digital synthesizer. The latter

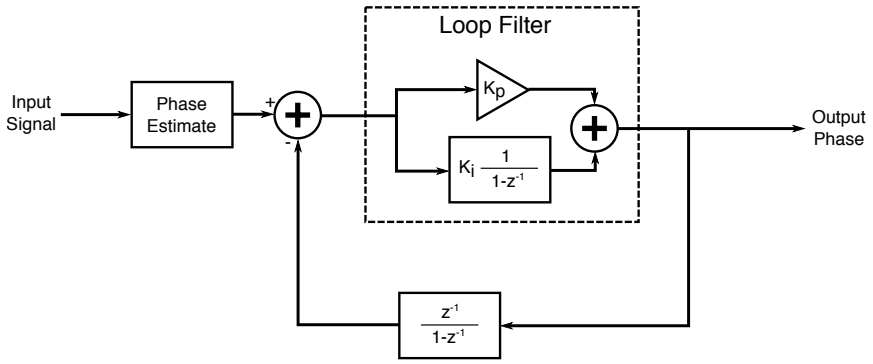

Figure 8. Second order phase locked loop block diagram.

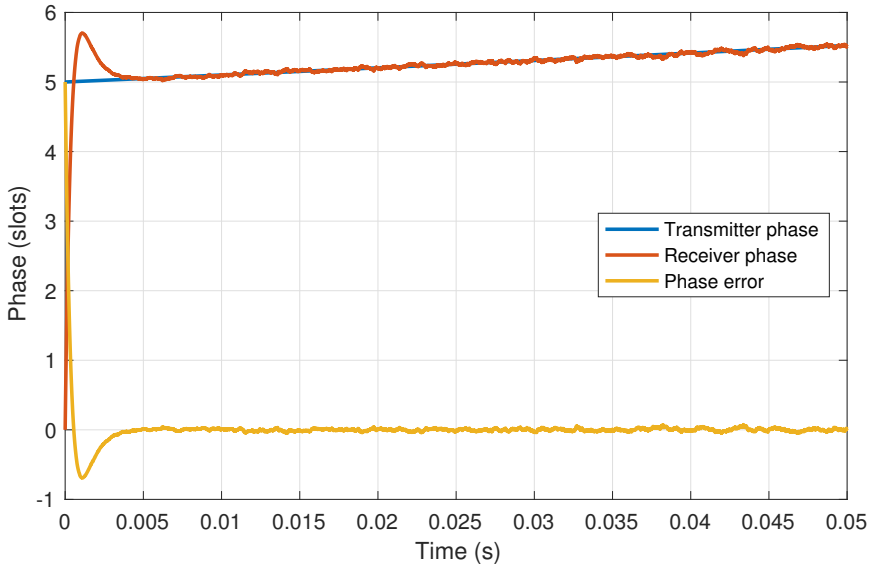

Figure 9. Phase locked loop performance of Scenario 1 with a 500 slot integration time.

has been implemented as a software routine, as shown in Figure 8.

Utilizing Scenario 1, we develop a second order proportional and integral critically-damped loop filter with 10 kHz loop bandwidth. This loop filter has two free parameters: the gain of the proportional controller ($K_p = 1.80 \times 10^{-3}$) and the gain of the integral controller ($K_i = 8.10 \times 10^{-7}$). Each phase detection is performed over 500 PPM symbols using the ML timing estimator. We introduce a frequency offset of 100 parts per million, and show in Figure 9 how the PLL is able to track the frequency offset through a time series. For Scenario 2, Figure 10 shows how we are able to reduce the steady-state phase error far below that of a single phase estimate, in which we reduce the phase error of a single low flux estimate from roughly 0.6 slots RMSE to 0.01 slots RMSE.

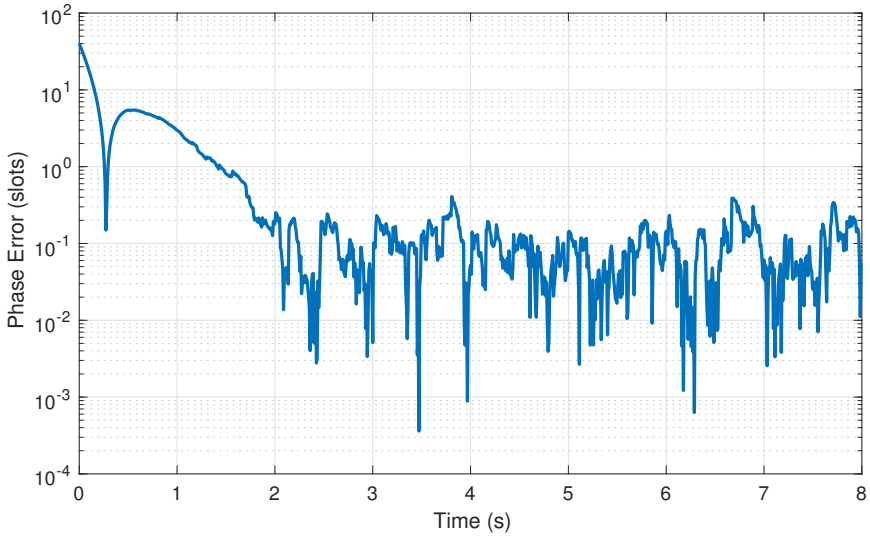

Figure 10. Phase locked loop performance of Scenario 2 with a 10000 slot integration time. The RMS error of the individual phase estimates is roughly 0.6 slots, whereas the RMS error of the phase-locked signal is less than 10^{-2} slots.

VIII. Conclusion

In this article we have developed several synchronization techniques for optical photon-counting detectors using pulse position modulation with inter-symbol guard times. We have shown the existence of a low complexity maximum likelihood estimator that outperforms previously known schemes with minor additional computational complexity, as well as a hybrid estimator which improves upon the prior art with a small constant-time additional complexity. As a largely academic exercise, we have also demonstrated how to generate the minimum mean square error estimator, though the complexity of its sampling scheme outweighs the minor additional improvement in RMSE. Finally, we have shown how any of the developed estimators may be utilized with well-known phase lock techniques for closed-loop digital frequency and phase tracking.

References

[1] H. Hemmati, *Deep Space Optical Communications*, ser. JPL Deep-Space Communications and Navigation Series. Wiley, 2006.

[2] K. Quirk, J. Gin, and M. Srinivasan, "Optical PPM synchronization for photon counting receivers," in *Military Communications Conference, 2008. MILCOM 2008. IEEE*, Nov 2008, pp. 1–7.

[3] R. Rogalin and M. Srinivasan, "Maximum likelihood synchronization for pulse position modulation with inter-symbol guard times," in *2016 IEEE Global Communications Con-*

ference (GLOBECOM), Dec 2016, pp. 1–6.

[4] V. A. Vilnrotter, E. R. Rodemich, and H. H. Tan, "A synchronization technique for optical PPM signals," *TDA PR 42-87, July-September 1986*, pp. 24–31, November 15, 1986.

[5] R. Velidi and C. N. Georghiades, "Symbol synchronization for optical multi-pulse pulse position modulation systems," in *Personal Wireless Communications, 1994., IEEE International Conference on*, Aug 1994, pp. 182–184.

[6] S. Patarasen and C. N. Georghiades, "Maximum-likelihood symbol synchronization and detection of OPPM sequences," *IEEE Transactions on Communications*, vol. 42, no. 6, pp. 2282–2290, Jun 1994.

[7] B. Erkmen and B. Moision, "Maximum likelihood time-of-arrival estimation of optical pulses via photon-counting photodetectors," in *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*, June 2009, pp. 1909–1913.

[8] Y. Fujiwara, "Self-synchronizing pulse position modulation with error tolerance," *IEEE Transactions on Information Theory*, vol. 59, no. 9, pp. 5352–5362, Sept 2013.

[9] R. Gagliardi and S. Karp, *Optical Communications*, ser. Wiley Series in Telecommunications and Signal Processing. Wiley, 1995.

[10] D. F. Yu and J. A. Fessler, "Mean and variance of single photon counting with deadtime," *Physics in Medicine and Biology*, vol. 45, no. 7, p. 2043, 2000.

[11] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[12] B. Moision, "Photon jitter mitigation for the optical channel," *IPN PR 42-171*, pp. 1–13, November 15, 2007.

[13] D. Rife and R. Boorstyn, "Single tone parameter estimation from discrete-time observations," *Information Theory, IEEE Transactions on*, vol. 20, no. 5, pp. 591–598, Sep 1974.

[14] J.-J. van de Beek, M. Sandell, and P. Borjesson, "ML estimation of time and frequency offset in OFDM systems," *Signal Processing, IEEE Transactions on*, vol. 45, no. 7, pp. 1800–1805, Jul 1997.

[15] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953. [Online]. Available: http://dx.doi.org/10.1063/1.1699114

[16] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice*, ser. Chapman & Hall/CRC Interdisciplinary Statistics. Taylor & Francis, 1995.

[17] K. J. Quirk and M. Srinivasan, "Optical PPM demodulation from slot-sampled photon counting detectors," in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, Nov 2013, pp. 1634–1638.

[18] F. Gardner, *Phaselock Techniques*. Wiley, 2005.