PROCEEDINGS OF SPIE

Photon counting and precision photometry for the Roman Space Telescope Coronagraph

Nemati, Bijan

SPIE.

Photon Counting and Precision Photometry
for the Roman Space Telescope Coronagraph

Bijan Nemati[a]

[a]University of Alabama in Huntsville, 301 Sparkman Dr., Huntsville, AL 35899, U.S.A.

ABSTRACT

The Nancy Grace Roman Space Telescope will include, as one of its two instruments, the highest contrast coronagraph ever attempted, with sensitivity down to Jupiter class planets. With flux ratio below 1e-8, these planets will be exceedingly dim, so that signal rates are as low as 0.01 electrons per second at the imaging detector. These rates necessitate ultra-low noise detectors and methods. For its science imaging camera, the Roman Coronagraph will employ an electron multiplication CCD (EMCCD), to achieve near-zero read noise. EMCCD's, however, deliver the low read noise at the cost of amplification of all other noise, because of the stochastic nature of the electron multiplication process. To circumvent this next-order challenge, a thresholding technique called photon counting can be used. The resulting image has no read noise and no excess noise factor (ENF). The remaining challenge, for precision photometry, is to account for the undercount and overcount effects inherent to photon counting. These arise primarily from the inefficiency of thresholding itself, and coincidence loss, where multiple-electron events are not distinguished from single-electron ones. Here we present a detailed description of the photon counting algorithm and the corrections necessary to achieve photometric accuracy below 0.5%.

Keywords: photon counting, photometry, CCD, EMCCD, coronagraph, exoplanet

1. INTRODUCTION

High contrast imaging in space will take a major leap forward with the Coronagraph Instrument (CGI) on the Nancy Grace Roman Space Telescope.[1] Equipped with deformable mirrors, and focus and tip tilt active control, this technology demonstration instrument will achieve at least two orders of magnitude higher contrast than any existing instrument, and will potentially have sensitivity to image and obtain spectra, in the visible band, from Jupiter class planets. High contrast imaging coronagraphs offer versatility in targeting and longer mission life, since they do not require formation flying as do external occulters.[2] Along with this comes a disadvantage of generally lower throughput, particularly when the telescope has significant obscurations as is the case for Roman. A Jupiter class planet, depending on its albedo and phase function, will have a flux ratio around $5 \cdot 10^{-9}$ (5 ppb), or a delta-magnitude of ~ 21. The typical target star brightness will be around 6 mag in the visible band, so that the planet will be at a relative magnitude of about 27. In direct imaging, this results in signal rates in the 10 milli-photon per second regime, necessitating ultra low noise sensors. In this paper we will describe the low noise sensor approach used by Roman CGI, the EMCCD, and derive the image processing steps need to provide precision (0.5%) photometry of the low noise images.[3]

2. THE EMCCD

The best CCD's currently achieve about 3 electrons (e^-) of read noise, and even at this rate, read noise becomes by far the dominant noise source. Since the instrument is being operated in the space environment, with significant rates of cosmic rays, frame times must be kept relatively short. Integration with higher frame rate makes the per-read classes of noise, chief among them read noise, more important. Electron multiplication CCD's (EMCCD's) can dramatically reduce read noise by amplifying the signal through successive impact ionization events. Each pixel's charge packet in these devices is clocked out as usual but in the last stage, it is passed

---

Further author information, send correspondence to:
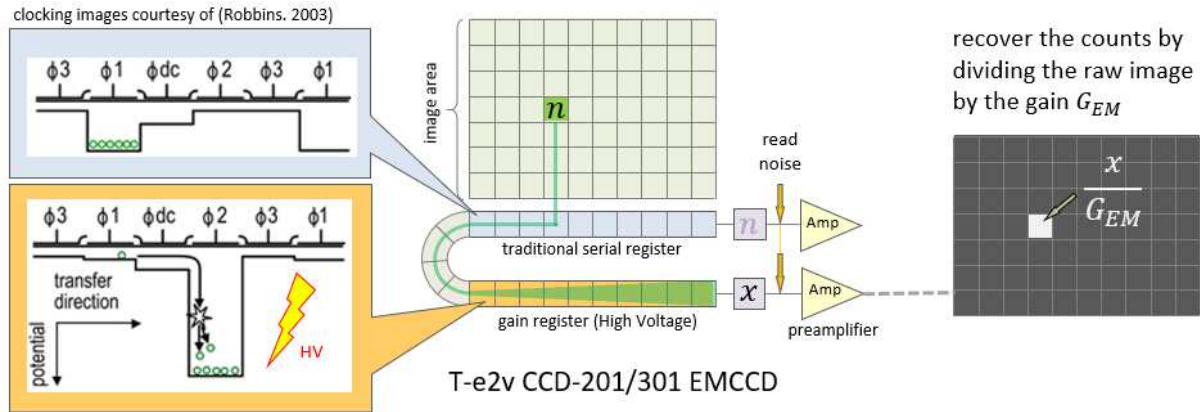Bijan Nemati: bijan.nemati@uah.edu, Telephone: 1 256 824 2525

Figure 1. The basic architecture of the Teledyne e2v CCD-201(or 301) EMCCD. Charges parallel clocked into the serial register can be clocked serially either right to a standard, low noise amplifier, or left through a gain register, which incorporates a "high voltage" clock phase, into a high-bandwidth amplifier. The gain of the signal effectively reduces read noise in proportion. The output image is divided by the mean gain to obtain the estimate of the pre-gain image.

through a serial *gain register* prior to read out. Roman CGI uses a mission-optimized version of the Teledyne e2v CCD-201 EMCCD, designated as the CCD-301, featuring mitigation for charge traps and cosmic ray tails. In the CGI's two cameras, this EMCCD will be operated with gains as high as 7500. The amplified image, sometimes also called the 'analog' or 'proportional' image, is enhanced by a factor of the EM gain, G_{EM}, so that the best estimate image is obtained by dividing every pixel by the EM gain (a single scalar number for the whole image).

The readout amplifier at the end of the gain register has relatively high read noise, of order $100 \ e^-$, due to the high clock rate at which it is operated. However, this read noise is reduced proportionately when gain is applied, relative to the image electrons. That is, if the operating gain is 5000, then the effective read noise would be $0.02 \ e^-$. The mean EM gain, which comes about from successive impact ionization opportunities, is related to the per-stage probability p_s according to:[4]

$$g = (1 + p_s)^{n_g} , \tag{1}$$

where n_g is the number of gain stages in the gain register. For the CCD-201/301, the number of gain elements is $n_g = 604$. Thus, if, for example the probability for impact ionization is 1.4%, the resulting mean EM gain is $g = 4435$. Changing the high voltage phase changes the gain to be suitable to the input flux and integration time. With read noise reduced, the next highest sources of detector noise are usually dark current (typically $\sim 1 \ e^-/\text{pix/hr}$) and *clock-induced charge* (CIC, typically $\sim 0.01 \ e^-/\text{pix/fr}$). CIC is in fact present in all CCD's but it is always negligible relative to a CCD's read noise. Only when it is amplified by EM gain, as it is in an EMCCD, does it become a noticeable noise source. In fact the amplification causes *all* noise to be enhanced, except read noise. This is due to the stochastic nature of the gain process, and has been described in detail elsewhere.[4] This *excess noise factor*, or ENF, is a function of gain but quickly approaches its asymptotic value of $\sqrt{2}$ for gains above ~ 20. Robbins (2003) derive the dependence of ENF on the EM gain:[4]

$$\text{ENF} = \frac{1}{g} + 2 \ (g - 1) \ g^{-(n_g+1)/n_g} . \tag{2}$$

Thus, with EM gain, read noise is dramatically reduced, but all other noise, is *increased* by roughly a factor of $\sqrt{2}$.

A number of approaches have been proposed for optimizing the signal to noise ratio of EMCCD images.[5] One method that keeps the advantage of eliminating read noise without incurring ENF, is called *photon counting*. If the frame rate is high enough, the expectation value of counts per pixel per frame becomes low enough that most frames have zero photo-electrons, and occasionally have a single photo electron, and rarely more than one.

In this regime, the amplified image is discriminated, pixel by pixel, by applying a specific threshold. Pixels with counts below the threshold are designated as having zero photo-electrons, and those above are designated as having one photo-electron. the frames are then co-added to create a summed image which has virtually no read noise, and no ENF.
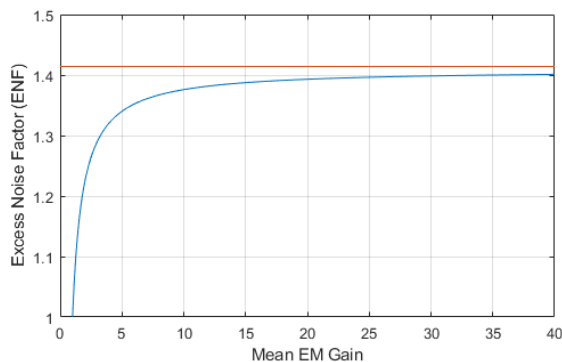

Figure 2. Excess noise factor as a function of EM gain for the T-e2v EMCCD, which has 604 gain elements. The line at $\sqrt{2}$ represents the asymptotic value at higher gains.

These improvements come at the cost of three effects: 1) threshold loss, where a real photo-electron event is rejected because it failed the threshold cut; 2) coincidence loss where, a 2-count event is called a single-count event, and 3) leakage from detector effects.

Overall, each of the steps outlined here, from using EMCCD's to employing photon counting, solves a bigger problem at the expense of smaller ones. Photon counting of EM images represent a nearly ideal image sensing approach, but in many precision applications, precision photometry is also important. The Roman CGI error budget calls for photometry errors below 0.5%.

In this paper we review the basics of photon counting and develop a correction algorithm that yields photometric correction accuracy below 0.5%. We conclude by outlining future possible improvements.

3. PHOTON COUNTING

Incident photons within the response region of a pixel create Photo-electrons as random, independent events following Poisson statistics. For a photon rate r_{ph} incident on a pixel, the mean expected count rate per frame in time t_f is given by:
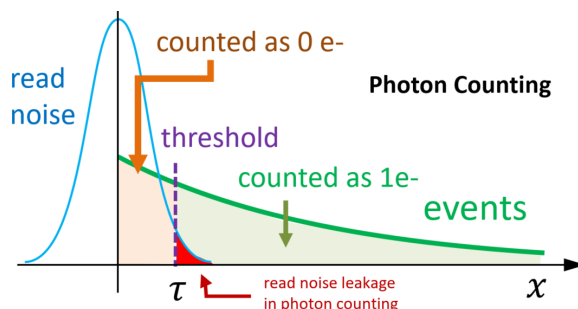
$$\lambda = r_{ph} \; \eta \; t_f \,, \tag{3}$$

Figure 3. Photon Counting rejects read noise and eliminates ENF at the expense of some efficiency loss.
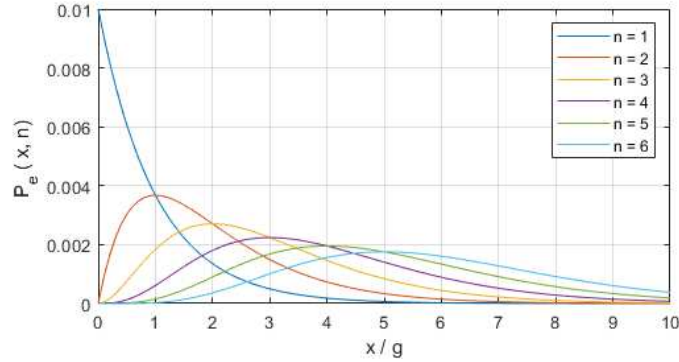
Figure 4. EM gain probability distribution $P_e(x|n)$ as a function of gain-normalized output electrons x for the first few numbers of input electrons n. The mean x/g for each distribution is n.

where η is the pixel's quantum efficiency. The actual number of electrons, n, in a given frame is a random instance of the Poisson probability distribution function (PDF), given by:

$$P_p(n|\lambda) = \frac{\lambda^n\ e^{-\lambda}}{n!}\ . \tag{4}$$

For a given photon rate r_{ph}, if t_f is short enough, λ can be made small. Table 1 lists the probabilities for getting n from 0 to 4 for three small values of mean expected value λ.

Table 1. Poisson probabilities $P_p(n|\lambda)$ to observe n counts when the mean expected value is λ, for a few values of λ. In photon counting, typically frame rates are increased to reduce λ to approximately 0.1.

n	0	1	2	3	4
$P_p\ (n|\ 0.05)$	95.1%	4.8%	0.12%	$2.0 \cdot 10^{-5}$	$2.5 \cdot 10^{-7}$
$P_p\ (n\ |\ 0.1)$	90.5%	9.0%	0.45%	0.015%	$4 \cdot 10^{-4}$
$P_p\ (n\ |\ 0.5)$	60.7%	30.3%	7.6%	1.3%	0.16%

The image photo electrons then go through the gain register and multiplied in number. Basden et al. show in Ref. 6 that the probability distribution function for the number of generated electrons, x, given that n electrons entered the gain register is given by the discrete gamma probability distribution function:

$$P_e(x|g,n) = \frac{x^{n-1}\ e^{-x/g}}{g^n\ (n-1)!}\ . \tag{5}$$

For $n = 0$ no output electrons are generated. When $n = 1$, the PDF reduces to the exponential distribution:

$$P_e(x|g,1) = \frac{e^{-x/g}}{g}\ . \tag{6}$$

Photon counting begins with an EM-gain amplified *analog* image, and applies a threshold t to the counts x in each pixel to assign a value of $n = 0$ or $n = 1$ to the number of input electrons depending on whether $x < \tau$ or $x \geq \tau$. The thresholding necessarily carries with it an inefficiency. If all the non-zero events had $n = 1$, the thresholding efficiency would be given by integrating Eq. 6 from τ to infinity, to obtain:

$$\epsilon_{th}^{(1)} = e^{-\tau/g}\ . \tag{7}$$

The superscript "(1)" is a reminder that this is a first order approximation, and not exact. For sufficiently small mean expected values λ, n is rarely above 1, so this is a reasonable approximation. But when n is above 1, the efficiency for that fraction of the cases is underestimated, as can be seen from comparing the distributions in 4. This is an "overcount."

Besides overcounts due to underestimating the threshold efficiency, the procedure has also makes "undercount" error, assigning $n = 1$ for cases where in reality $n = 2$ or higher. These "coincidence" cases, coming from high-multiplicity Poisson events, are the cause of one type of inefficiency in photon counting, called coincidence loss.[7] It is shown in Appendix A that the efficiency after coincidence loss is given by:

$$\epsilon_{CL} = \frac{1 - e^{-\lambda}}{\lambda} \, . \tag{8}$$

The placement of the threshold depends on the scene and sensor conditions, but in general is done to maximize signal to noise ratio (SNR) over the course of the measurement. The photometric SNR is given by:

$$\mathrm{SNR} = \frac{S}{\sqrt{S + B}} \, , \tag{9}$$

where S is the signal and B is the background plus detector noise. In photon counting, every noise source acts as a background contributing shot noise. For example, read noise, which is Gaussian random in nature, and is characterized by its standard deviation, σ_r, contributes false positives as it "leaks" past the threshold, as shown in Fig. 3. The leakage rate is the probability of a false positive per pixel per frame, and is given by:

$$r_l = \frac{1}{2} \, \mathrm{erfc}\left(\frac{\tau}{\sqrt{2} \, \sigma_r} \right) \, . \tag{10}$$

Typically a threshold of $\tau = 5\sigma_r$ is used, but the best strategy is to flesh out Eq. 9, including the leakage rate from E. 10 and determine an optimum.

3.1 Photon Counting Steps with Background Subtraction

Roman CGI direct images are created from stacks of frames taken from the scene, corrected for bias from various detector effects. Focusing on a single pixel, we outline here a general model and procedure for obtain a corrected count. All of the pixels are corrected the same way, so the procedure readily applies to the entire frame of pixels as well. For the pixel in consideration, one can model the post-gain readout q as

$$q = b + P_n(\sigma_r) + P_e(x|n, g) \, . \tag{11}$$

where b is a time-dependent bias (including a fixed pattern component if need be) and $P_n(\sigma_r)$ is the normal Gaussian probability distribution with standard deviation equal to the read noise σ_r. Assuming a photon flux into the pixel (measured in ph/s/pix) of ϕ, pixel quantum efficiency η, dark current i_d (e/pix/s) and some CIC (measured in e/pix/fr) the distribution of possible n, i.e., photo-electrons in the image area, is given by:

$$n = P_p(n \mid \phi \, \eta \, t_f + i_d \, t_f + \mathrm{CIC}) \, . \tag{12}$$

If we were *not* doing photon counting, after collecting a number of such frames and averaging, the resulting mean count for this pixel would be given by:

$$\langle q \rangle = \langle b \rangle + g \, \langle \phi \rangle \, \eta \, t_f \; + g \, \langle i_d \rangle \, t_f + g \, \langle \mathrm{CIC} \rangle \, . \tag{13}$$

Note that the read noise mean is zero and does not appear in the mean signal equation.

When we photon count, we apply a threshold $\tau + b$ to each frame. Since the bias b can be separately calibrated and is readily handled in this way, henceforth we omit it in the discussion for simplicity, by assuming it is zero. The actual threshold is, to be remembered, always $\tau + b$. Once we threshold, the factor g disappears, but photon

counting related efficiencies appear. We then transition from the raw readout q to collected *counts* (image area photoelectrons) c, so that the post-thresholding version of the equation becomes:

$$\langle c \rangle = (\langle \phi \rangle \; \eta \; t_f \; + \langle i_d \rangle \; t_f + \langle \text{CIC} \rangle) \cdot \epsilon_{th} \; \epsilon_{CL} \,. \tag{14}$$

We now note that the quantity in the parentheses is the expected counts λ (of Eq. 4), and label it as such:

$$\lambda_{br} = \langle \phi \rangle \; \eta \; t_f \; + \langle i_d \rangle \; t_f + \langle \text{CIC} \rangle \,. \tag{15}$$

The "br" subscript indicates that this is the mean expected counts for "bright" (br) frame, i.e. one where the flux ϕ is non-zero. In terms of λ, Eq. 14 becomes:

$$\langle c \rangle = \lambda_{br} \cdot \epsilon_{th} \; \epsilon_{CL} \,. \tag{16}$$

Assuming slowly varying conditions, "dark" (dk) frames, with the same frame time t_f, can be taken (ideally before and after the bright frames) and averaged. The mean expected counts will be the same as given by Eq. 14, but without the first term, involving the flux. Thus the dark-subtracted counts are given by:

$$\lambda_{br} - \lambda_{dk} = \langle \phi \rangle \; \eta \; t_f \,. \tag{17}$$

What is ultimately of interest is $\langle \phi \rangle$, and this is obtained by dividing the left hand side of Eq. 17 by the quantum efficiency and the frame time, all of which are either known or can be obtained from calibrations. This procedure, applied to EM "analog" frames, produces an accurate estimate of the photoelectrons and then the flux.

We now offer a set of steps for photon counting with the aim of obtaining λ_{br} and λ_{dk}, so that we can eventually get $\langle \phi \rangle$ via Eq. 17. For photon counted estimates, the steps are:

1. shorten the single-frame exposure time t_f until most (e.g. $\sim 90\%$) of the pixels have 0 photo-electrons

2. choose a threshold τ for photon counting, such that the SNR is maximized

3. collect N_{br} bright frames

4. threshold each frame, setting the count at each pixel to 1 if the analog counts are above τ, otherwise, 0

5. co-add the bright frames to get a single photon counted frame for the full exposure time $t = N_{br}t_f$

6. repeat with the dark frames, using the same threshold τ and frame exposure time t_f

7. apply photometric correction

For the dark frames, choosing the same frame exposure time t_f ensures that read noise and CIC and read noise leakage are correctly subtracted. When possible, using substantially more dark frames than bright frames reduces the shot noise in the dark-subtracted frame after scaling.

In the co-added bright frame, considering a single pixel, the counts observed are:

$$N_{br} = \lambda_{br} \; N_{fr} \; \epsilon_{th} \; \epsilon_{CL} \,. \tag{18}$$

Our objective, therefore, is to obtain a bias-free solution of Eq. 18 in terms of λ_{br}. We need, per the above procedure, a solution for each of the bright and dark cases so the dark subtraction in Eq. 17 can be accomplished. In the following discussion, we may occasionally drop the br subscript, but it is implied. Also implied is the need to repeat the procedure for dark frames to get λ_{dk}.
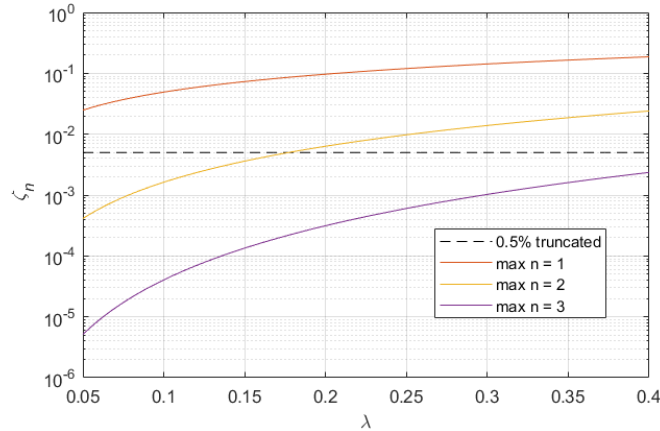
Figure 5. Fraction of Poisson events truncated by only considering event multiplicities up to n, as a function of λ. The truncated fraction is defined in Eq. 20. The dashed line, shown as a guide, corresponds to 0.5% being truncated.

3.2 First Order Photometric Solution

Using Eq. 18, with expressions for the efficiencies from Eq. 8 and Eq. 7, we find:

$$\lambda_1 = -\ln\left(1 - \frac{N_{br}/N_{fr}}{e^{-\tau/g}}\right).$$

(19)

This is a straight-forward solution to the photon counting photometry problem, and is accurate at the percent level for typical scenarios. It is a first order approximation, and hence bears the subscript 1 to remind us of this. But the Roman CGI error budget allocation is only 2% for all calibration errors combined. The error budget is based on observing a companion with flux ratio of 50 ppb, so this amounts to an allocated error of 1 ppb in flux ratio noise units. Furthermore, photometric correction is one of many calibration terms, and is thus allocated only 0.5% . Thus, for greater accuracy we need to go back and improve on our approximations.

3.3 Next-Level Approximation

Whereas the coincidence loss efficiency given by Eq. 8 is exact, the thresholding efficiency given by Eq. 7 was made under the assumption that the random counts, when they occur, always manifest as a single count ($n = 1$). For these, the EM gain process produces the exponential PDF, per Eq. 6, we integrated that equation to arrive at Eq. 7. We can calculate what fraction of events 'missing' from accounting when we keep truncate the Poisson distribution function in this way. We note first that in photon counting, for a given pixel, the frames that contribute to the total count are the non-zero frames. From Eq. 4, we see that the fraction of zero-count events is $e^{-\lambda}$, so all the rest add up to $1 - e^{-\lambda}$. The truncated fraction, for a given maximum n considered, is then given by:

$$\zeta_n = 1 - \frac{\sum_{i=1}^{n} P_p(i|\lambda)}{1 - e^{-\lambda}}.$$

(20)

We plot this for n values 1 through 3, in Fig. 5. We see that the first order approximation, effected by truncating past $n = 1$ (our first-order approximation) misses to account for roughly 5% of the cases at $\lambda = 0.1$. For these missing cases, the EM gain PDF (Eq.5) is different from that for $n = 1$, and the threshold efficiency is *higher* than given by Eq. 7 (see Fig. 4), so the use of this approximated value results in an over-estimation of λ in Eq. 19.

The truncation fraction does not correspond one-to-one to the amount of overestimation, but it does give a rough guide. Since in a single frame, different pixels have varying λ, the photometric correction distorts the image to some small level. To keep the distortion below the required value of 0.5%, we infer from Fig. 5 that

going out to $n = 3$ should more than satisfy the 0.5% Roman CGI requirement. In general, the 'truncated' PDF that includes terms out to a maximum n is given by summing over the included n values:

$$P_n(x|\lambda) = C(n, \lambda) \cdot \sum_{i=1}^{n} P_e(x|g, i) P_p(i|\lambda). \tag{21}$$

where C is a normalization factor needed to ensure this PDF integrates to unity over all x:

$$C(n, \lambda) = \left(\int_0^\infty P_n(x|\lambda) \, dx \right)^{-1}. \tag{22}$$

The thresholding efficiency corresponding to this PDF is given by integrating it from τ to infinity. Note in Eq. 21 the summation starts at 1 and not 0, since the 0-count frames do not contribute to the total. For $n = 3$, following Eq. 21, we have:

$$P_3(x|\lambda) = C(3, \lambda) \cdot \left[\lambda \, e^{-\lambda} \, \frac{e^{-x/g}}{g} \left(1 + \frac{\lambda x}{2g} + \frac{\lambda^2 x^2}{12 g^2} \right) \right]. \tag{23}$$

To get $C(3, \lambda)$, we integrate this function, and using Eq.22 obtain:

$$C(3, \lambda) = \frac{6 \, e^\lambda}{\lambda \left(6 + \lambda(3 + \lambda) \right)}. \tag{24}$$

Thus the third order approximation to the threshold efficiency, obtained by integrating $P_3(x|\lambda)$ from τ to infinity, is given by:

$$\epsilon_{th}^{(3)} = e^{-\tau/g} \cdot \left(1 + \frac{\tau^2 \lambda^2 + 2g\tau\lambda(3 + \lambda)}{2g^2(6 + 3\lambda + \lambda^2)} \right). \tag{25}$$

While adding more terms makes the correction more accurate, it increases the complexity of the equation. In fact adding only a single additional term, i.e. going to $P_2(x|\lambda)$ results in not being able to solve Eq. 18 in closed form. However, an iterative solver, like the well-known Newton's method, can provide the needed accuracy usually in 1 to 2 iterations.[8] The Newton method finds the nearby root of a function through successive improvements to an initial guess:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}. \tag{26}$$

We can thus form an objective function from Eq. 18 such that its roots in λ are the solutions to that equation. This is simply:

$$f(\lambda) = \lambda \, N_{fr} \, \epsilon_{th}(\lambda) \, \epsilon_{CL}(\lambda) - N_{br}. \tag{27}$$

Equation 28 and Eq. 8 inserted into Eq. 27 provides $f(\lambda)$. Its derivative $f'(\lambda)$ is:

$$\begin{aligned} f'(\lambda) = \frac{e^{-\tau/g - \lambda} N_{fr}}{2g^2(6 + 3\lambda + \lambda^2)^2)} \cdot (& 2g^2(6 + 3\lambda + \lambda^2)^2 \quad + \quad t^2\lambda(-12 + 3\lambda + 3\lambda^2 + \lambda^3 + 3e^\lambda(4 + \lambda)) \\ & + \quad 2gt(-18 + 6\lambda + 15\lambda^2 + 6\lambda^3 + \lambda^4 + 6e^\lambda(3 + 2\lambda))). \end{aligned} \tag{28}$$

The starting guess is well supplied by λ_1 from Eq. 19. Appendix B provides a Matlab example listing for a function that implements this photometric correction.

4. TESTING THE ALGORITHM

To check the results, we performed three tests. In the first test, we simulated a large number of frames being co-added, with a particular value of λ, via Eq. 11, except we set the bias b to zero, and with n generated by the Poisson PDF $P_p(n|\lambda)$. We did this for a number of λ values. In Fig. 6, we show the results for $\lambda = 0.10$. 3000 Monte Carlo trials, each with 100,000 frames (a single pixel in each frame) with $\lambda = 0.1$ were co-added, after having each been applied the Poisson and EM randomizations and read noise was also added to the pre-thresholded frames. The 0th order estimate is simply N_{br}/N_{fr}. Since in the 0th order estimate there are no
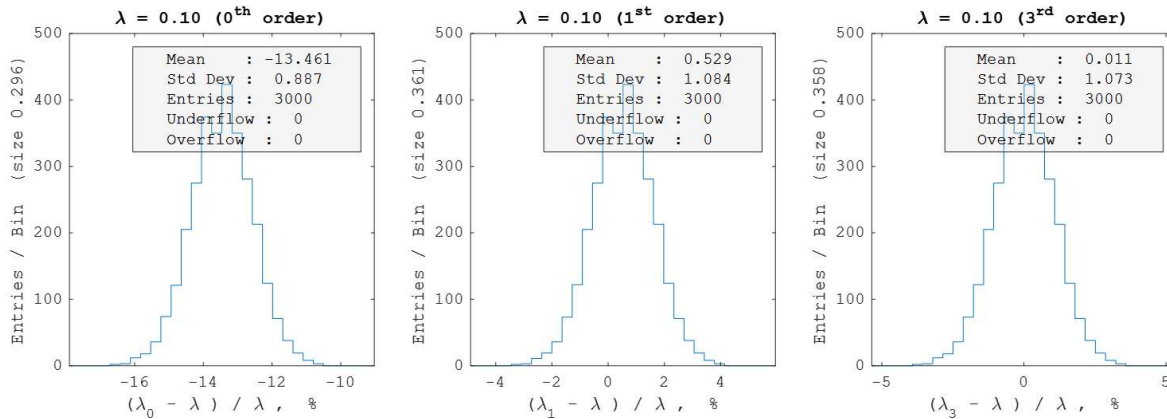
Figure 6. Testing the photometric correction algorithm on simulated pixel readouts. Each plot is a histogram of the fractional error in the estimate. On the left is the 0th order estimate N_{br}/N_{fr}; in the middle is the first order estimate given by Eq. 19, and on the right is the 3rd order estimate using $\epsilon_{th}^{(3)}$ and Newton's method. The means errors are seen to be 13.5%, 0.53%, and 0.01%, respectively.
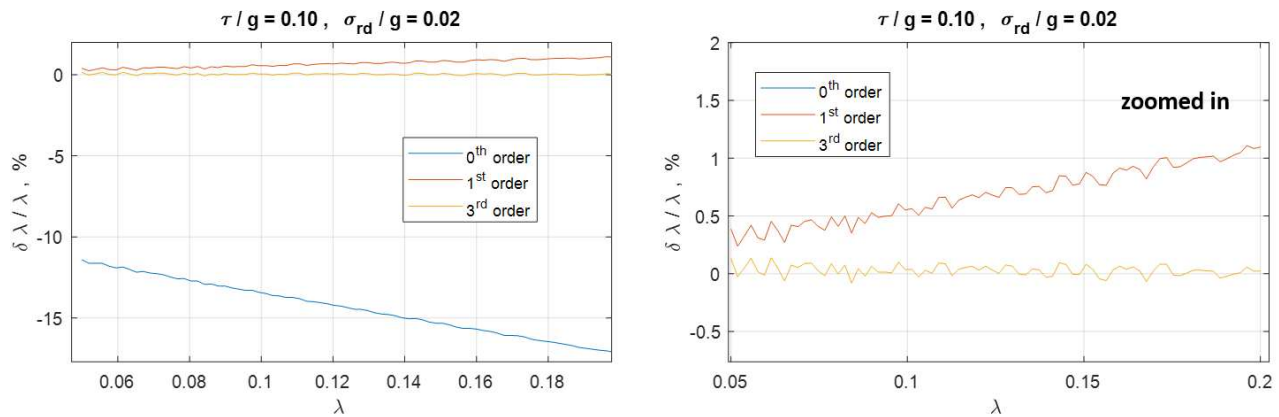

Figure 7. Testing the sensitivity of the different approximations to the actual value of λ at a given threshold. A threshold to gain ratio of 0.1 was used, while the read noise was 5 times smaller ($\sigma_{rd}/g = 0.02$), amounting to a threshold at $5\,\sigma_{rd}$. The plots are in percent fractional error, and the right plot is a zoomed-in version of the left plot, showing that the third order solution shows no visible dependence, while the first order solution does.

efficiency corrections, it underestimates significantly, by 13% for $\lambda = 0.1$. The first-order estimate is based on Eq. 19 and is nearly good enough, at just over 0.5%. However, for larger λ the error would grow. The third-order estimate is seen to have a mean error of only 0.01%, well below the required value.

Another important test is the sensitivity at a given threshold to λ. This is a measure of the brightness-dependent distortion caused by the algorithm. Figure 7 shows the results from running, for each of a series of λ values, a Monte Carlo with 3000 instances where 10,000 frames were co-added to compute λ, keeping the thershold fixed. The percent error in estimating λ is plotted. While the 1st order solution, which neglects $n > 1$ PDF's and their differing threshold efficiencies, shows a significant λ dependence, the third order solution shows no discernible dependence.

Finally, we switched the second test to examine the threshold dependence at a particular value of λ. At low threshold, read noise leakage is expected, but past $\sim 4\,\sigma_{rd}$ the leakage becomes negligible and the dependence is only due to estimation error. Again, while the first order correction shows a distinct threshold dependence the third order solution shows no discernible dependence.
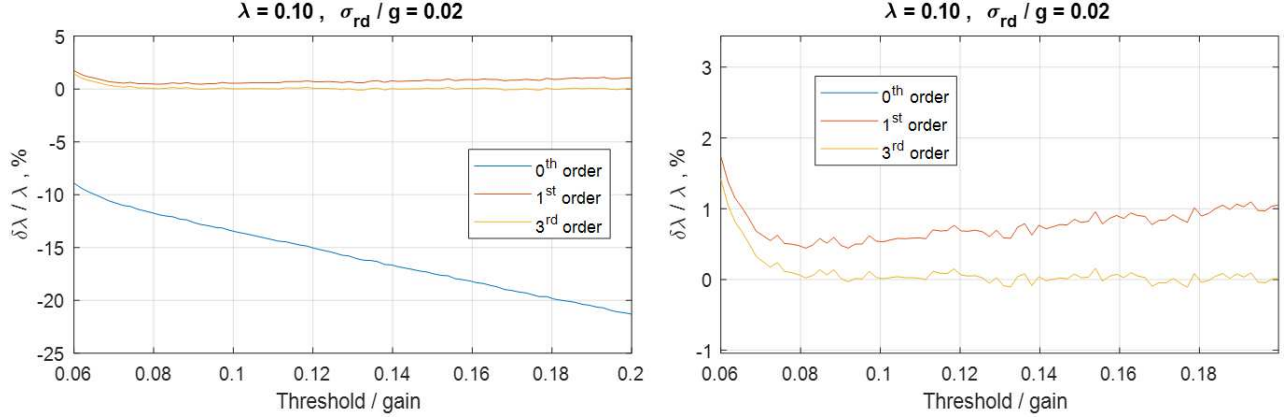
Figure 8. Threshold sensitivity of the various order approximations for estimating λ is shown, with the zoomed-in version on the right. The effect of read noise leakage is clearly seen for $\tau < \sim 4\,\sigma_{rd}$. Beyond, the third order shows no threshold dependence.

5. SUMMARY AND CONCLUSION

We have described the photon counting algorithm as planned for the Roman Space Telescope Coronagraph instrument, and discussed in some detail the photometric accuracy of the thresholding approach. We have shown there exists a simple, closed form, first order solution (Eq. 19) to the mean expected count (photoelectron) rate, λ per frame, but this solution provides only $\sim 1\%$ level of photometric accuracy. We found that going to the third order correction, which accounts for Poisson distributed high multiplicity events out to $n = 3$ electrons, provides an excellent approximation, good to less than 0.1% bias for the reasonably expected range of thresholds and pixel brightness λ. This more than satisfies the requirements of the Roman CGI calibration budget. Finally we have provided a Matlab function that applies this correction in the appendix.

APPENDIX A. COINCIDENCE LOSS

In photon counting, the simple, single threshold that discriminates between the $n = 0$ and $n = 1$ cases does not take into account the $n > 1$ cases. For the latter there is an undercount. Here we derive the efficiency of this undercount, called coincidence loss. Suppose the mean expect rate of counts per frame for a given pixel is λ and we read N such frames. The mean expected total for N frames is $N_{tot} = N\lambda$. From Eq. 4, of the N co-added frames, there are on average $Ne^{-\lambda}$ frames that have $n = 0$ and $N_1 = N\lambda e^{-\lambda}$ that have $n = 1$. The number of events that have $n > 1$ is then given by:

$$N_{>1} = N(1 - (1 + \lambda)e^{-\lambda}). \tag{29}$$

The procedure ascribes to these each a single photo electron. The efficiency factor for this undercount is given by the fraction of the expected electrons $N\lambda$ that have been accounted for, using this scheme:

$$\epsilon_{CL} = \frac{N_1 + N_{>1}}{N_{tot}} = \frac{N\lambda e^{-\lambda} + N(1 - (1 + \lambda)e^{-\lambda})}{N\lambda} = \frac{1 - e^{-\lambda}}{\lambda}. \tag{30}$$

APPENDIX B. A MATLAB FUNCTION FOR COMPUTING λ

Here we provide a function for iterative solution to λ, in Matlab.

```matlab
function lam_est = photCorrPC (nobs, nfr, t, g)
% This function computes the expected per-frame value, lambda, given the total, nobs,
% seen in nfr frames, given that the EM gain is g and the threshold is set at t
%
% B. Nemati -UAH- 7-Nov-2020

% start with a good first estimate and iterate a couple of times
lam_est = -log(1 - (nobs/nfr) * exp(t/g) );  % first order estimate
lam_est = lam_est - deltaLam (lam_est, t, g, nfr, nobs);
lam_est = lam_est - deltaLam (lam_est, t, g, nfr, nobs);

end

function dlam = deltaLam (lam, t, g, nfr, nobs)
% Photon counting solve funciton for n=3 correction
% evaluates the function   lam * nfr * coincidenceEff * threshEff - nobs
% where:
% lam  = mean expected rate per pixel per frame
% t    = threshold chosen for photon counting
% g    = EM gain
% nobs = sum of counts across all frames after thresholding
%
% func = lam * nfr * coincidenceEfficiency * thresholdingEfficiency - nobs
% fprime = d func / dlam  at this lambda
%
% output:
%   dlam  = func / fprime  at this lambda
%
% B. Nemati -UAH- 7-Nov-2020


epsThr3 = ...
    exp(-t/g) * ( t^2 * lam^2  +  2*g*t*lam*(3+lam) + 2*g^2*(6+3*lam+lam^2) ) ...
    / ( 2*g^2 * (6 + 3*lam + lam^2) );

epsCL = (1-exp(-lam))/lam;

func = lam * nfr * epsThr3 * epsCL - nobs;


dfdlam = (1/(2 * g^2 * (6 + 3*lam + lam^2 )^2) ) * ...
    exp(-t/g - lam) * nfr * ...
    ( ...
    2*g^2*(6+3*lam+lam^2)^2 + ...
    t^2 * lam * (-12 + 3*lam + 3*lam^2 + lam^3 + 3*exp(lam)*(4+lam)) + ...
    2*g*t *(-18 + 6*lam + 15*lam^2 + 6*lam^3 + lam^4 + 6 * exp(lam) * (3+2* lam)) ...
    );


dlam = func / dfdlam ;

end
```

Listing 1. Matlab Example

ACKNOWLEDGMENTS

REFERENCES

[1] Mennesson, B., Debes, J., Douglas, E., Nemati, B., Stark, C., Kasdin, J., Macintosh, B., Turnbull, M., Rizzo, M., Roberge, A., Zimmerman, N., Cahoy, K., Krist, J., Bailey, V., Trauger, J., Rhodes, J., Moustakas, L., Frerking, M., Zhao, F., Poberezhskiy, I., and Demers, R., "The WFIRST coronagraph instrument: A major step in the exploration of sun-like planetary systems via direct imaging," in [*Proceedings of SPIE - The International Society for Optical Engineering*], **10698** (2018).

[2] Ruane, G., Mawet, D., Mennesson, B., Jewell, J., and Shaklan, S., "Vortex coronagraphs for the Habitable Exoplanet Imaging Mission concept: theoretical performance and telescope requirements," *Journal of Astronomical Telescopes, Instruments, and Systems* **4**(01), 1 (2018).

[3] Harding, L. K., Demers, R. T., Hoenk, M., Peddada, P., Nemati, B., Cherng, M., Michaels, D., Neat, L. S., Loc, A., Bush, N., Hall, D., Murray, N., Gow, J., Burgon, R., Holland, A., Reinheimer, A., Jorden, P. R., and Jordan, D., "Technology advancement of the CCD201-20 EMCCD for the WFIRST coronagraph instrument: sensor characterization and radiation damage," *Journal of Astronomical Telescopes, Instruments, and Systems* **2**(1), 011007 (2015).

[4] Robbins, M. S. and Hadwen, B. J., "The noise performance of electron multiplying charge-coupled devices," *IEEE Transactions on Electron Devices* **50**(5), 1227–1232 (2003).

[5] Rousset, N., Villeneuve, J., Fournier-Lupien, J.-H., Attiaoui, A., Taillon, G., Francoeur, S., and Daigle, O., "Comparison of EMCCD post-processing methods for photon counting flux ranges," *High Energy, Optical, and Infrared Detectors for Astronomy VI* **9154**, 91540F (2014).

[6] Basden, A. G., Haniff, C. A., and Mackay, C. D., "Photon counting strategies with low-light-level CCDs," *Monthly Notices of the Royal Astronomical Society* **991**, 985–991 (2003).

[7] Daigle, O., Gach, J.-L., Guillaume, C., Lessard, S., Carignan, C., and Blais-Ouellette, S., "CCCP: a CCD controller for counting photons," *Ground-based and Airborne Instrumentation for Astronomy II* **7014**, 70146L (2008).

[8] Galántai, A., "The theory of Newton's method," *Journal of Computational and Applied Mathematics* (2000).