The *Chandra* ABC Guide to Pileup

Chandra X-ray Science Center - Science Data Systems

June 17, 2005

Abstract

The purpose of this document is to provide a *Chandra-ACIS* specific overview of pileup - i.e., the phenomena of two or more photon events overlapping in a single detector frame and being read as a single event. We discuss the definition of pileup and describe its effects on detected spectra and variability. We outline methods for avoiding pileup, and also discuss when not to avoid pileup. We describe several ways the degree of pileup can be estimated when planning an observation. Methods of mitigating the effects of pileup in real data are outlined. This document is meant as a 'living resource'. As knowledge of how to detect, assess, and correct for the effects of pileup improves, we will update and expand the procedures described below.

1 Pileup Basics

1.1 Definitions

Pileup is a phenomenon that is inherent to CCD detectors, such as those that comprise the *ACIS* instrument on-board *Chandra*, which 'under-sample' the mirror point spread function (PSF). Simply put, it occurs whenever two or more photons are detected as a single event, and thus it represents a loss of information from these events. The degree to which this information can be 'recovered' is described below. Any corrections, however, are necessarily imperfect. Thus, it is often desirable to choose instrumental set-ups that minimize the occurrence of pileup.

The likelihood of pileup occurring is significant whenever source flux levels are high enough such that there is a reasonable probability of two or more photons arriving within the same detector region within a single *ACIS* frame integration time (or CCD row readout time, for continuous clocking mode). The charge from a single photon event is typically read out from a 3×3 pixel island; therefore, the relevant 'detector region' referred to above is larger than a single pixel. Charge clouds from neighboring events can overlap and cause events centered several pixels away from each other to become piled (see Davis 2001, for a more thorough description).

The detected energy of a piled event is approximately equal to the sum of the energies of the individual photon events of which it is comprised. If the summed energy of the piled event exceeds the on-board spacecraft threshold (typically 15 keV), it is rejected by the spacecraft software. For sufficiently bright sources, this can lead to a visible 'hole' in the source image, as we illustrate in Fig. 1.

Piled events also suffer from 'grade migration'. All events detected by *ACIS* are assigned grades based upon the shape of their charge cloud distributions in a 3×3 pixel island. These grades are used to determine whether the detected event is from a real photon or from a background event, such as a cosmic ray hit. As the number of photon events making up a piled event increases, it is more and more likely that the grade assigned to this piled event will 'migrate' to a value inconsistent with a real photon. The piled event thus will be rejected either by spacecraft software or during subsequent analysis on the ground. This effect of grade migration also contributes to the detection hole illustrated in Fig. 1.

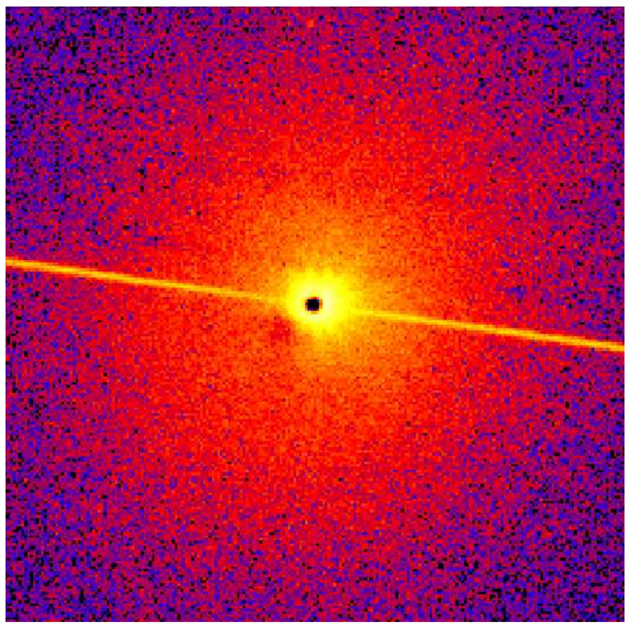

Figure 1: 0th order image from an *ACIS-HETG* observation of a bright X-ray binary. Here, the count rate is sufficiently high that most piled events at the center of the point spread function (PSF) exceed the threshold energy and/or are assigned bad grades, yielding a hole in the image. In addition to the image hole caused by pileup, there is a readout streak (photons collected during the 41.04 msec required to transfer an image frame to the readout buffer), as well as asymmetry in the wings of the PSF. This latter effect is due to the effects of Charge Transfer Inefficiency (CTI), which similar to pileup, affects the grades of photon events. For CTI, photons closer to the chip readout (right side of the image) are less likely to have their grades migrate to bad values, and hence are less likely to be rejected.

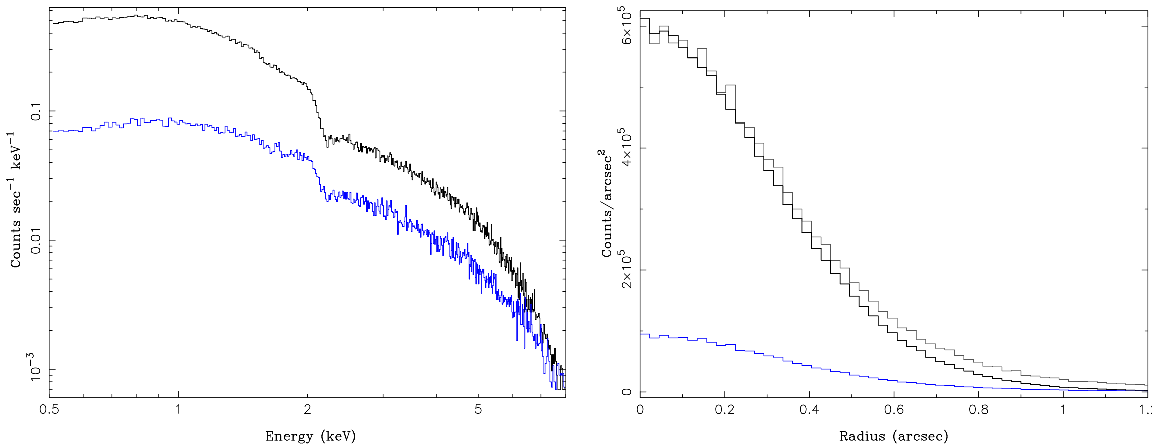

Figure 2: *Left: MARX* simulations of a piled up power-law spectrum. The black (upper) line shows the detected spectrum, presuming no pileup. The blue (lower) line shows the detected spectrum presuming a 3.24104 s frame time (i.e., a 3.2 s integration plus a 41.04 ms frame transfer time) and a grade migration parameter of $\alpha = 0.5$. *Right:* Radial point spread function (PSF) associated with each spectrum. The grey (upper most) line shows the piled PSF (blue, lowest line) rescaled to the amplitude of the unpiled PSF. Note that pileup makes the PSF appear wider.

As a simple empirical description of this process of grade migration, one can assign a probability, α, that for each photon event beyond the first, the piled event retains a grade consistent with a real photon. Thus, in this simple model, the probability that a piled event is *retained* as a 'real photon' is $\alpha^{(N-1)}$, where N is the number of photons comprising the piled event. It is very important to note here that this is an empirical description of grade migration that has been found useful in some situations (examples of which will be described below). As such, α is an uncalibrated quantity, and is likely unsuited for some applications. Grade migration is a complex phenomenon, which in reality will depend upon details of the detector, the incident spectrum, etc. We have found, however, that within the confines of our current understanding of the physics and calibration of the detector, more complex grade migration schemes are not yet warranted.

1.2 Pileup Fractions and Their Effects

Pileup's two major effects are energy migration (photon energies sum to create a detected event with higher energy) and grade migration (event grades migrate toward values inconsistent with real photon events). In combination these lead to *all* of the following effects occurring in a piled source. There is a net decrease in the total observed count rate, as well as a net decrease in the fractional root mean square (rms) variability of the lightcurve. The detected spectral shape from the source is distorted, with there being a net loss of photons. The peak amplitude of the observed source point spread function is decreased, and the PSF shape is distorted. To what degree these effects of pileup can be tolerated in any given observation depends, of course, upon the specific science goals one wishes to achieve.

It is very important to remember the energy dependence of the pileup effects. Whereas there is a net decrease in total count rate, it is possible for pileup to yield a net increase in restricted high energy bands. Likewise, the energy dependent effects on the detected PSF can be very complex. In Fig. 2, we show *MARX* simulations of these effects. Because of the energy dependence of pileup, it is impossible to arrive at a single diagnostic to indicate its severity. For example, although the overall pileup in a given observation might be mild and hence tolerable for broad energy band variability studies, it might be considered detrimental for science that focuses on any hard X-ray tail. This is because such tails typically have photon power law indices with $\Gamma \geq 1.7$ (i.e., photon count flux proportional to photon energy $E^{-1.7}$). A fractionally small number of photons from near 2-3 keV might pile with themselves, with the piled events being detected at 4-6 keV. In this energy range, between the decreased effective area of the detector and the fall-off in the intrinsic spectrum, the piled events might comprise a fractionally large portion of the detected 4-6 keV events.

Nevertheless, the concept of a 'pileup fraction' can be useful in preliminary assessments. Below we discuss several different possible definitions of 'pileup fraction'. For simplicity, we will be assuming Poisson statistics with a fiducial incident (mean) count rate, Λ. This rate is to be interpreted as the counts per detector region per frame time that would occur in the absence of pileup. Again, the detector region of interest is larger than a single pixel, and is approximately a 3×3 pixel region, and represents the majority of source counts *for on-axis sources only*. Within this framework, we further will assume a 'perfect PSF' such that in any frame with two or more photon events, all photon events will be piled up. Thus, the following estimates become increasingly inaccurate as the source is moved off-axis. Finally, we will adopt the simple empirical α-model for grade migration.

Given the above assumptions, one can describe the fraction of *detected* events, f_e, that are in fact piled events. The total rate of detected events is given by $\sum_{n=1}^{\infty} \alpha^{n-1} \Lambda^n \exp(-\Lambda)/n!$, while the rate of *single event frames* (i.e., the only unpiled events) is given by $\Lambda \exp(-\Lambda)$. Thus the pileup fraction f_e is given by:

$$f_e = 1 - \frac{\Lambda \exp(-\Lambda)}{(\Lambda + \alpha\Lambda^2/2! + \alpha^2\Lambda^3/3! + \ldots)\exp(-\Lambda)} = 1 - \frac{\alpha\Lambda}{\exp(\alpha\Lambda) - 1} \quad . \tag{1}$$

This is essentially the definition of pileup fraction used by *CIAO* spectral fitting tools, i.e., the `jdpileup` model in *Sherpa* (where pileup fraction is accessed via the `show pileup` command) and by the `pileup` kernel in *ISIS* (where pileup fraction is accessed via the `print_kernel` command).

An alternative definition of pileup fraction is f_f, the fraction of frames that have detected events that contain two or more events, *with both quantities calculated in the absence of pileup*. This is given by

$$f_f = 1 - \frac{\Lambda \exp(-\Lambda)}{(\Lambda + \Lambda^2/2! + \Lambda^3/3! + \ldots)\exp(-\Lambda)} = 1 - \frac{\Lambda}{\exp(\Lambda) - 1} \quad . \tag{2}$$

This is essentially the definition used by the web interface to *PIMMS*[1], with the further caveat that in calculating f_f, *PIMMS* multiplies the above expression by 0.866, and also multiplies the total count rate by this same factor. This additional weighting represents the fraction of flux in the central 3×3 pixel island when the mirror assembly was tested on the ground. (The mirror performance is improved in the weightlessness of space, and a larger fraction of the count rate is contained within the central 3×3 pixel island.) Note that this definition does not account for the effects of grade migration.

For variability studies, a more useful definition might be the fraction of the expected count rate lost due to pileup. Including the detected piled events into the observed count rate, this fraction is given by

$$f_r = 1 - \frac{(\Lambda + \alpha\Lambda^2/2! + \alpha^2\Lambda^3/3! + \ldots)\exp(-\Lambda)}{\Lambda} = 1 - \frac{[\exp(\alpha\Lambda) - 1]\exp(-\Lambda)}{\alpha\Lambda} \quad . \tag{3}$$

To lowest order, the mean count rate is reduced by a factor of $(1 - \beta\Lambda)$, with $\beta \equiv (2 - \alpha)/2$. One can also show that to lowest order, the fractional variation of a lightcurve, defined as *observed* root mean square variability divided by *observed* mean count rate, is reduced by this same factor when compared to expectations from the unpiled lightcurve fractional rms. That is, even though the mean count rate is reduced, positive fluctuations above this mean are slightly more reduced, while negative fluctuations below this mean are slightly less reduced. Hence, piled lightcurves appear to have less fractional variability than they should.

[1] `http://http://cxc.harvard.edu/toolkit/pimms.jsp`

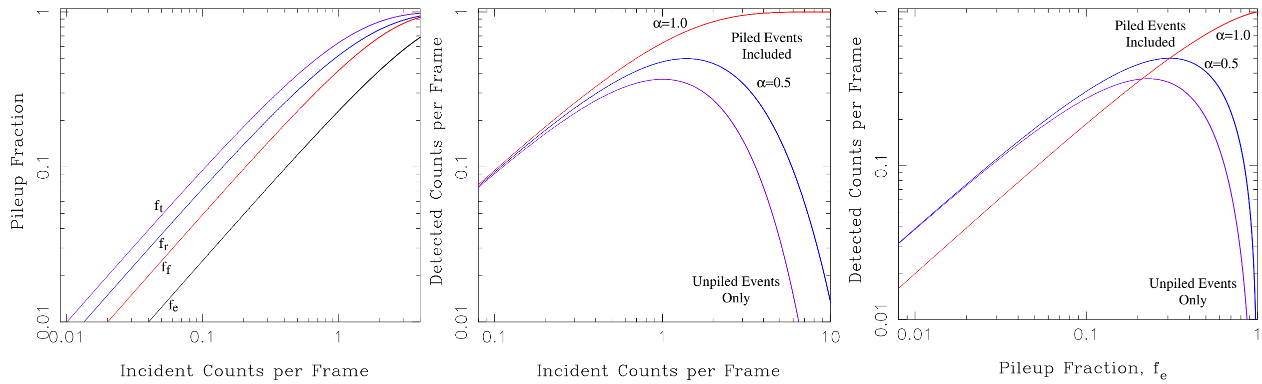

Figure 3: *Left*: Various definitions of pileup fraction vs. the incident expected (unpiled) count rate per frame per detector region, assuming, where relevant, a grade migration parameter of $\alpha = 0.5$. Curves, from bottom to top are as follows. Black: f_e, the fraction of detected events that are in fact piled events. Red: f_f, the fraction of frames with events that have two or more events. Blue: f_r, the fraction of the expected count rate lost due to pileup. Purple: f_t, the total fractional count rate loss, due to both energy and grade migration. *Middle*: Detected count rate per frame per detector region vs. the incident, unpiled count rate per frame per detector region. From bottom to top, the curves are as follows. Purple: Total detected *unpiled* count rate (independent of α). Blue: detected count rate, including piled events, for $\alpha = 0.5$. Red: detected count rate, including piled events, for $\alpha = 1$. *Right*: Identical to the middle figure, except here we plot detected count rate vs. pileup fraction, f_e, *which depends upon the grade migration parameter* α.

Finally, if one had a means of distinguishing which detected events were in fact piled events, a useful pileup fraction is the total fraction of events lost – whether it is due to grade or energy migration – compared to the expected rate. This can be expressed as

$$f_t = 1 - \frac{\Lambda \exp(-\Lambda)}{\Lambda} = 1 - \exp(-\Lambda) \ . \tag{4}$$

This definition is useful in application to the gratings, at least for first order spectra, since *order sorting* has the effect of removing events that are piled into a higher energy. (These events, however, appear in the higher orders of the detected spectrum.)

To illustrate these different definitions, in Fig. 3 we plot pileup fraction vs. incident (i.e., expected, unpiled) count rate per frame per detector region. We also plot detected count rate vs. incident count rate, as well as detected count rate vs. pileup fraction. Where relevant, we have assumed $\alpha = 0.5$; however, we also show $\alpha = 1$ for two cases. Note that these figures follow exactly the above definitions. *They have not been calibrated to actual* Chandra *data.* They are meant to serve as a guide, rather than be used for quantitative analysis.

1.3 Avoiding Pileup

Given the deleterious effects of pileup upon spectra, images, and lightcurves, it is usually advisable to search for ways to minimize it. There is essentially one goal, achievable with a variety of strategies: one must reduce the counts per frame per pixel. Thus the strategies involve combinations of spreading the signal out over more pixels (via offset pointing, defocusing, or inserting the gratings), and reducing the integration times (via subarrays and turning off detector chips, or implementing continuous clocking mode). Defocusing is *not* a recommended strategy. Below, we outline the pros and cons of some of the better methods for minimizing pileup. Note that many of the methods can be combined (for example, performing a gratings observation with a subarray implemented).

- Offset Pointing: Placing a source several arcminutes away from on-axis pointing serves to both reduce the effective area of the mirrors, as well as broaden the point spread function. Offset pointings therefore reduce the counts per frame per pixel. There are several disadvantages to this approach, however. Aside from the obvious disadvantage that *Chandra* was designed for high resolution imaging and that it would be unfortunate not to utilize this capability, it should be kept in mind that calibration is best understood and described for on-axis pointings. Furthermore, the advantage that can be gained by this strategy is limited to factors of several, which means that the brightest sources must be handled by other means.

- Short Exposures: The nominal frame time for a 6 chip, full-frame *ACIS* observation is 3.24104 sec – a 3.2 sec integration followed by a 41.04 msec frame transfer to readout. Frame integration times as short as 0.2 sec can be chosen without any loss of detector area. The tradeoff, however, is that approximately 3.2 sec is still required to read the image from frame storage into memory. Thus, such observations are highly inefficient in terms of effective exposure time, with the exposure efficiency being approximately the chosen integration time divided by 3.2 sec. A factor of 16 reduction in counts per frame can be achieved at the price of a 94% loss of efficiency.

- Subarrays and Turning Off Chips: The minimum frame integration time can be shortened by reducing the portion of an *ACIS* chip that is to be read out and/or by shutting off unwanted chips. A sub-array as small as 1/8 of the chip can be chosen, which reduces the nominal frame integration time from 3.2 sec to 0.8 sec and 0.7 sec for six chip *ACIS-I* and *ACIS-S* observations, respectively. By further limiting the observation to a single chip, one can achieve frame times of 0.5 sec and 0.4 sec, respectively. Thus, up to a factor of 8 reduction in the counts per frame rate can be achieved; however, one is obviously forgoing the opportunity to study spatial structure beyond the borders of the chip and subarray selections. Again, this strategy is limited in how much pileup can be reduced. For example, if one wanted to observe a source described by a $\Gamma = 2$ power-law with a neutral hydrogen column of $10^{21}\ \mathrm{cm}^{-2}$, yet limit the event pileup fraction, f_e, to less than 10%, the observations would be restricted to sources with absorbed 0.5-8 keV fluxes of approximately less than $7 \times 10^{-12}\ \mathrm{ergs\ cm^{-2}\ s^{-1}}$, assuming that $\alpha = 0.5$. Note that for this case, the *total* pileup fraction, f_{t}, is $\approx 40\%$.

- Gratings: The *HETG* or *LETG* gratings can also be used to reduce pileup. They act as a filter and produce a reduced count rate 0^{th} order, CCD quality spectrum. Simultaneously, the gratings disperse the spectrum into multiple arms, which both reduces the total count rate compared to the absence of gratings and spreads the spectrum out over a much larger range of pixels. The advantage compared to a subarray is that 0^{th} order can image a larger region of the sky for the same reduction in pileup, while the gratings arms simultaneously produce higher resolution spectra. The disadvantage is that the signal-to-noise is reduced, and thus longer integration times are required. How much longer these integration times need to be depends upon the source spectrum, but generally they are greater for softer sources and are less severe for harder sources. For example, if one wished to use *ACIS-HETG* to observe the 2-8 keV spectrum of a $\Gamma = 2$ power law source, the 0^{th} order spectrum count rate is reduced by approximately a factor of 6 compared to that without the gratings, while the summed count rate in the four gratings arms is also reduced by a factor of 6 compared the CCD count rate without the gratings in place. Thus, in the 0^{th} spectrum alone one can achieve a pileup reduction comparable to a 1/8th subarray (without gratings), at the price of requiring an approximately 3 times longer integration time to achieve the same signal-to-noise in the summed observation (i.e., 0^{th} order plus the four gratings arms). Softer sources, especially those with significant flux below 2 keV, are more reduced in total count rate and hence require even greater increases in integration times.

  Although the flux limits for a given pileup fraction in the 0^{th} order spectrum are comparable to the 1/8th subarray, the dispersed spectrum can tolerate substantially greater fluxes without becoming piled up (see below). Typically, sources with 0.5-8 keV fluxes less than $\approx 10^{-9}\,\mathrm{ergs\,cm^{-2}\,s^{-1}}$ have *total* pileup fractions, f_t, that do not exceed $\approx 10\%$ in any region of the gratings arms. This latter statement, of course, is dependent upon the shape of the incident spectrum.

- CC-Mode: In continuous clocking (CC) mode, CCD rows are read out one row every 2.85 msec. Although this is more than 1100 times faster than the full frame readout time of 3.2 s, a 1000 times increase in flux tolerance for avoiding pileup is not obtained. Again, grades are assigned in 3×3 pixel islands, and this is true even in CC-mode where 'virtual frames', consisting of 512 rows that were consecutively read out, are created to assign grades. Thus at a minimum one needs to consider the effective readout time for assessing the possibility of pileup to be at least three times longer than the nominal 2.85 msec. (That is, one needs to consider the integration times for the rows adjacent to any detected event.) Additionally, CC-mode entails a higher background, and does not allow one to perform any CTI correction. (The fact that a photon event could potentially pileup with the 'trailed charge' from transfer inefficiency also might lead one to assign an 'effective integration time' for assessing pileup that is even greater than three times the row readout time.) For a given pileup tolerance, CC-mode therefore allows one to observe sources perhaps 50 times brighter than is achievable for a single chip, 1/8 subarray observation. Using the example of the $\Gamma = 2$ power law source with a $10^{21}\,\mathrm{cm^{-2}}$ column, for event pileup fractions, f_e, less than 10% (assuming $\alpha = 0.5$), one is restricted to 0.5-8 keV fluxes approximately $< 4 \times 10^{-10}\,\mathrm{ergs\,cm^{-2}\,s^{-1}}$. (Again, the *total* pileup fraction is larger, at $f_t \approx 40\%$.) This limiting flux is slightly less than the allowed limits when inserting the gratings; however, higher signal-to-noise (but obviously worse spectral resolution, and only one dimension of spatial information) is obtained for a given integration time.

- Piled Gratings Spectra: For sources with 0.5-8 keV fluxes approximately $> 10^{-9}\,\mathrm{ergs\,cm^{-2}\,s^{-1}}$, even the spectra in the gratings arms can become significantly piled. (Note that the 0^{th} order spectrum becomes severely piled at substantially lower flux levels.) Again, the relative figure of merit for determining the degree of pileup is counts per frame per pixel. From this point of view, gratings pileup is typically worst in the MEG as opposed to the HEG gratings. The MEG gratings have slightly larger effective area and half the spectral resolution, and thus more than twice the counts per pixel (compared to HEG) near the peak of their effective area curves. The MEG effective area is largest between 6 and 8 Å (1.5-2 keV), so this is often the spectral regime most effected by pileup. This statement, of course, depends upon the incident spectrum.

  Aside from choosing a subarray (typically one can choose a 1/2 subarray without significant loss of spectral information from the gratings arms) or CC-mode, it may be possible to further mitigate spectral pileup in the gratings by looking at higher order photons. This suggestion comes with an important caveat: it is possible to show that to simplest approximation the fractional loss term in *each* spectral order is identical to eq. (4) with the count rate, Λ, being the *total* count rate at the detector location associated with the wavelength and order of interest. That is, if 6 Å first order photons are piled up, 2 Å third order photons at that same location are equally piled. Furthermore, these third order events are possibly contaminated by piled events from first and second order combining to yield a false third order event. Choosing a higher order only avoids pileup if the wavelength of interest, at higher order, comes from a low observed count rate portion of the detector. (These points are illustrated further in Fig. 4.)

  The combination of the gratings plus CC-mode allows for the highest limiting fluxes for a given pileup fraction.

We again note that the above mitigation methods can be combined. Perhaps the most extreme example is the *Chandra* observation of Sco X-1 (Observation ID 3505). Sco X-1 is usually the brightest X-ray source

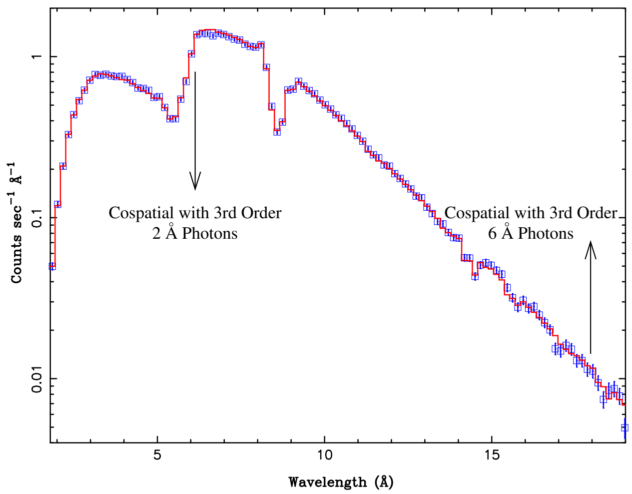

Figure 4: A count rate spectrum (counts s^{-1} Å^{-1}, which is proportional to counts per frame per pixel) of an X-ray binary observed with *ACIS-HETG*. This spectrum corresponds to MEG+1 order, which dominates the total count rate in the positive orders of the MEG arm. The peak of the observed count rate is near the peak of the effective area at ≈ 6Å. Estimations are that in this detector location, the total pileup fraction, f_t, is $\approx 10 - 15\%$. (This observation used a 1/2 subarray, and thus had a frame time of 1.74104 sec.) The positive third order 2 Å photons also come from this peak count rate location, and hence are actually more piled up than first order 2 Å photons. On the other hand, third order 6 Å photons are co-spatial with 18 Å first order photons, and come from a low count rate region of the detector. Thus, the third order 6 Å photons are less piled up than their first order counterparts.

in the sky (aside from the Sun), so this particular observation employed an offset pointing, the gratings, CC-mode, *and* the subsequent analysis involves choosing higher order photons for certain wavelengths of interest! Which method, or combination of methods, is best for any particular observation will of course depend upon both the incident spectra and the desired science goals.

1.4 Not Avoiding Pileup

There are times when one knows full well ahead of time that some portions of an observation will be piled up, yet the chosen instrument configuration is the best for the desired science goals. There are a number of obvious examples. The X-ray image of a globular cluster might subtend an entire *ACIS* chip and produce fluxes and spectra for a hundred sources. Only a small handful – the very brightest – of these sources might exhibit detectable pileup. Subarrays or CC-mode would forgo valuable information from the majority of sources to improve the spectra of the minority of sources. Likewise, X-ray binary surveys of external galaxies will often have a small fraction of their sources that are piled up. For example, the nucleus and several ultra-luminous X-ray (ULX) sources might be mildly piled. Again, accepting mild pileup in a small fraction of sources might be considered a fair trade-off for measuring spectra in a larger field of view.

There also are instances when one might accept pileup in gratings spectra. Gratings observations of arc-minute scale dust scattering halos in front of a central point source might be performed best without the use of CC-mode. This is especially true if one wants to be able to spatially model the effects of CTI on the point spread function (see Fig. 1). There are also cases where imaging mode is required to spatially separate nearby sources. As an example, Observation ID 4572, a gratings observation of an ‘accretion disk corona’ source, reveals two bright X-ray sources separated by 2.8 arcsec. This is far apart enough that the resulting gratings spectra are easily distinguished, although the MEG spectra exhibit mild pileup.

The common feature of all of the above examples is that for each observation one is accepting a degree of pileup in exchange for enhanced imaging coverage and information. There are two 'best advice' strategies for minimizing pileup in an observation that scientifically requires imaging information. The first strategy is to limit the integration time per frame as much as possible by choosing the smallest subarray that still contains the extended regions of interest. (As mentioned above, for a gratings observation one can often choose a 1/2 subarray without loss of spectral information from the gratings arms.) However, one is obviously forgoing the opportunity for 'serendipitous science' in the regions excluded by the subarrays. The second strategy is to insert the gratings. The drawback is that an increased integration time (a factor of three, or more for soft sources) is required to achieve the same signal-to-noise.

The truth is that the only *Chandra* CCD observations that are *not* affected by pileup are ones where both the source is faint enough and the observation is short enough that one does not statistically expect there to be any frames (or row readout times) where two photons have arrived in the same detector region of interest. Pileup is present, to greater and lesser degrees, in almost all *Chandra* observations. We therefore devote the rest of this guide to: estimation of the degree and effects of pileup for purposes of proposal planning, detection of pileup in existing observations, and mitigation of pileup effects in observed sources.

2 Pileup Estimation

For purposes of planning an observation, there are a number of means for estimating the severity and effects of pileup on an observation. In order of increasing levels of sophistication and detail, one can: perform analytic estimates, run a *PIMMS* simulation, simulate a piled spectrum within an X-ray spectral package, or run a *MARX* simulation. The first two methods are the most straightforward, but provide the least amount of information. These methods will only provide estimates of piled fractions, as defined in §1.1. If one wishes to quantitatively assess pileup effects on spectral shape, one must simulate a spectrum using an X-ray spectral fitting package (*Sherpa*, *ISIS*, or *XSPEC*), or run a *MARX* simulation. If one wishes to assess the effects of pileup on an image, currently the only available method is to run a *MARX* simulation. *MARX* simulations will also provide an accurate estimate of the effects of pileup on a mean (i.e., white noise) lightcurve. (As of yet, there is no simulation tool for assessing the effects of pileup on a variable lightcurve, e.g., one with red noise or quasi-periodic features, although such a tool is under consideration for development). Below, we discuss each of these simulation methods in turn.

2.1 Analytic Estimates

It is useful to run through simple analytic estimates in order to develop some intuition as to the degree of pileup one might expect. The goal is to arrive at a pileup fraction as defined in §1.1 for either the case of a CCD or a gratings observation by determining an expected count rate per frame per detector region of interest. The peak effective area of the *ACIS* CCD detectors occurs between 1 and 2 keV, and is approximately 600 cm^2. Thus, assuming an average photon energy of $2\,\mathrm{keV} = 3.2 \times 10^{-9}\,\mathrm{erg}$, a 10^{-12} $\mathrm{erg\ cm^{-2}\ s^{-1}}$ source has a count rate per 3.2 sec frame of approximately 0.6. Unless the observation is an offset pointing, most of these photons will land in the same 3×3 region of interest and contribute to pileup. Thus, one should expect a pileup fraction of $f_e = 14\%$ (eq. 1; and Fig. 3), assuming a grade migration parameter $\alpha = 0.5$. This might seem like a mild pileup fraction; however, using a more stringent definition of pileup fraction yields $f_t = 45\%$, i.e., 45% of the expected count rate will be subject to grade and energy migration (eq. 4; and Fig. 3). More sophisticated simulation techniques may be warranted for fractions this large.

For gratings observations, a natural flux unit to consider is $\mathrm{counts/cm^2/s/Å}$, since the wavelength scale is linear with the dispersion distance along the gratings arms. The MEG peak effective area is approximately 80 cm^2 near 6 Å, and the MEG wavelength grid is 0.011 Å/pixel (i.e., twice that of the HEG). Thus, in a

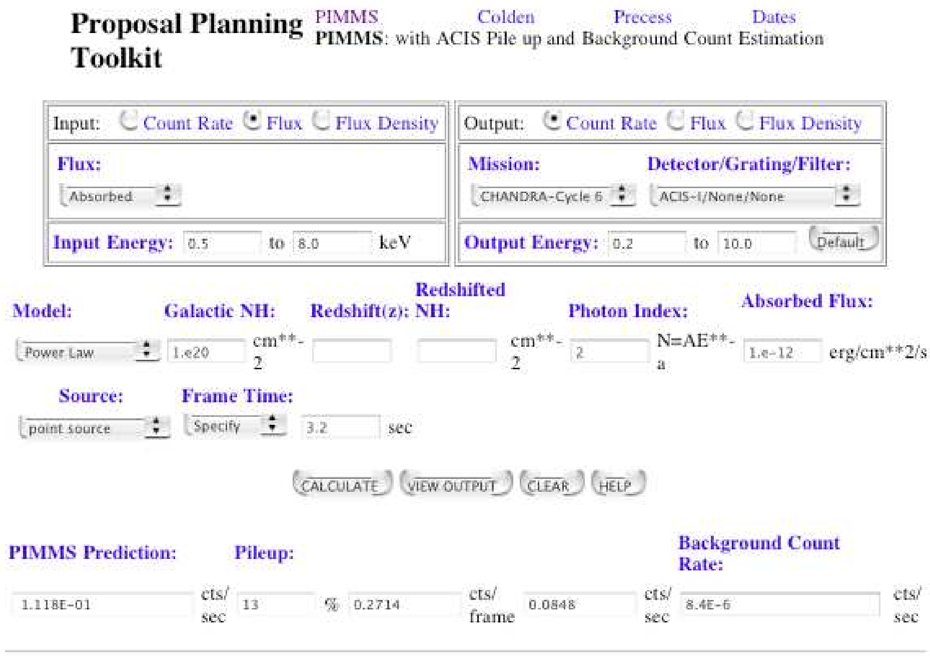

Figure 5: A screen capture of the *PIMMS* session described in the text.

3.2 sec frame time in a 3×3 pixel region near 6 Å, one expects 0.8 counts/frame for a source with a flux of 0.1 counts/cm^2/s/Å (≈ 0.3 counts/cm^2/s/keV, at 6 Å). Thus, under a stringent definition of pileup fraction, this yields $f_t = 55\%$. Again, this is likely large enough to warrant a more careful treatment.

2.2 PIMMS

Compared to the above simple estimates, *PIMMS* is performing a more careful calculation of the expected count rates. It is properly convolving the assumed spectral shape with a model of the detector response in order to predict a total count rate. The pileup calculation, however, is not any more sophisticated than the analytic estimates outlined above and in §1.2. As discussed in §1.2, the *PIMMS* pileup fraction is given by f_f, eq. 2, with a multiplicative factor of 0.866 applied to both the count rate and the overall pileup fraction. This is the fraction of frames that have two or more events in the central detection region, presuming that 86.6% of the count rate occurs in the central detection region.

As a concrete example, consider a $\Gamma = 2$ power-law absorbed by a neutral hydrogen column of $10^{20}\,\mathrm{cm}^{-2}$, with a 0.5–8 keV flux of $10^{-12}\,\mathrm{erg\,cm^{-2}\,s^{-1}}$. Assuming a 3.2 sec frame time observation performed with *ACIS-I*, *PIMMS* predicts 0.112 counts/sec in the 0.2–10 keV band (i.e., within a factor of two of the very simple estimates above). The predicted pileup fraction is 13%, which is consistent with the estimate given by f_f when one employs the factors of 0.866. The more stringent pileup fraction estimate, f_t, yields a fraction of 24%, where again we multiply both the total count rate and the pileup fraction by a factor of 0.866. This is completely consistent with the *PIMMS* simulation, which predicts that 24% of the count rate is lost due to pileup (i.e., the predicted 0.112 counts/sec rate becomes an observed 0.085 counts/sec). A screen capture of the *PIMMS* session described above is shown in Fig. 5.

This example illustrates several important points. First, whereas a predicted fraction of 13% might sound moderate, the reality of 24% of the expected events being subjected to grade and energy migration gives a truer sense of the severity of pileup. Second, *PIMMS* is implicitly assuming a grade migration parameter of $\alpha = 0$, i.e., all piled events are lost to bad grades. This is reflected in the estimated count rate provided by *PIMMS*. Third, and most important, *PIMMS* is not providing any information as to how the spectrum is being distorted by the effects of pileup.

For gratings observations, *PIMMS* only provides a useful estimate of the pileup fraction for the 0^{th} order spectrum. Employing the $10^{-12}\,\mathrm{erg\,cm^{-2}\,s^{-1}}$ power law source model discussed above and considering the *HETG* gratings, *PIMMS* predicts count rates of 0.018 cps, 0.016 cps, and 0.006 cps in the 0^{th} order, MEG first order, and HEG first order spectra, respectively. (This is consistent with our earlier discussion that for a hard source, inserting the gratings reduces the total count rate by a factor of approximately three.) The predicted pileup fractions, f_f, are 2%, 2%, and 1%, respectively. Only the first of these numbers, that for the 0^{th} spectrum, has any accuracy. The latter two pileup fractions are calculated presuming that the entire first order count rate falls within a region comparable in size to a point source imaged without the gratings. For gratings observations, and to assess the effect of pileup on spectral shapes, one needs to turn to more sophisticated simulations.

2.3 Spectral Simulations

If one wishes to determine the effect of pileup on spectral shapes, as well as the effect on the count rate, one needs to perform a spectral simulation. The simplest way to do this for an *ACIS* (non-gratings) observation is to use one of the X-ray spectral fitting packages – *Sherpa*, *ISIS*, or *XSPEC*– to create a fake spectrum with the pileup model of Davis (2001) applied. Compared to *PIMMS*, this has the additional advantages of allowing one to explore a far wider range of spectral models and to explore the effect of the grade migration parameter, α. Threads for creating simulated data sets with *Sherpa*, *ISIS*, and *XSPEC*, respectively, can be found on the web at:

```
http://asc.harvard.edu/sherpa/threads/fakeit/
http://space.mit.edu/cxc/isis/examples/ex_fakeit.sl
http://space.mit.edu/cxc/isis/examples/ex_pileup.sl
http://heasarc.gsfc.nasa.gov/docs/xanadu/xspec/manual/node28.html
```

In Appendix A and B, we present scripts to use within *Sherpa* and *ISIS* to create a fake spectrum of a piled up, absorbed power-law, as observed by the *ACIS-S* detector. These scripts make use of effective area files and response matrices for simulating *Chandra* data that can be obtained from:

```
http://asc.harvard.edu/caldb/prop_plan/index.html
```

Results of these simulations are shown in Fig. 6.

The simulations shown in Fig. 6 are for a full frame (3.24104 sec) observation of a mildly absorbed ($N_{\rm h} = 10^{20}\,\mathrm{cm^{-2}}$), $\Gamma = 2$ power-law with a 0.5–8 keV flux of $10^{-12}\,\mathrm{erg\,cm^{-2}\,s^{-1}}$. We have chosen a grade migration parameter of $\alpha = 1$, i.e., *all* piled events are retained as 'good events'. *This is likely to be an unrealistic assumption*; however, it was chosen to accentuate the effects that pileup possibly can have at high energy. Note that for these parameters, the overall count rate is reduced, but the high energy count rate is increased. This particular simulated spectrum has a pileup fraction (accessed via the `show pileup` command in *Sherpa*, or the `print_kernel()` command in *ISIS*) of $f_e = 23\%$.

To highlight the spectral effects of pileup in these simulations, we fit the piled spectrum with an unpiled power law. In order to explain the reduced count rates at low energy, and the increased counts at high energy, the fitted power-law photon index is $\Gamma = 1.46 \pm 0.04$. Fitting the fake data with a piled power law, but letting the fitted grade migration parameter also be a free parameter, yields $\Gamma = 1.82 \pm 0.12$ and $\alpha = 0.26 \pm 0.06$. (That is, this fitted model predicts a harder intrinsic power-law, but fewer piled events being retained as 'good events'.) This fitted photon index is closer to the 'true' value, but does not agree with it. This emphasizes an important point, which we will discuss in greater detail in §4: there are many degeneracies in models of piled spectra, and it is extremely difficult to unambiguously model all of the

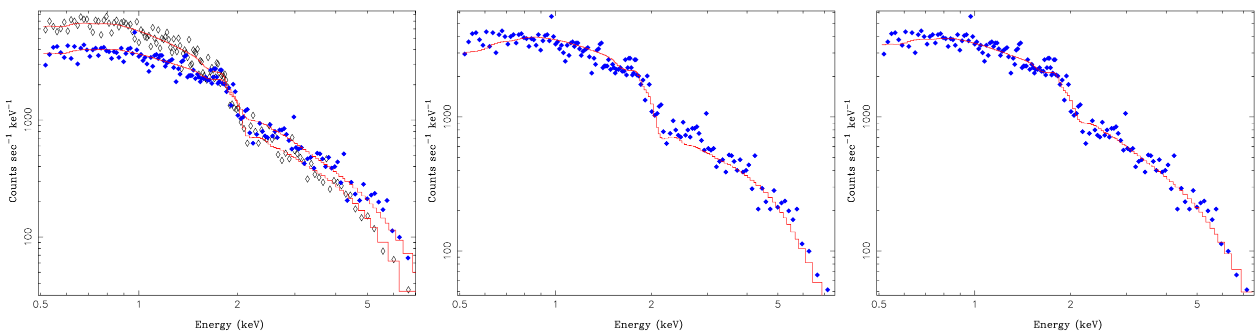

Figure 6: *Left*: Simulations of a 50 ksec *ACIS-S* observation of an absorbed power-law source with $N_{\rm h} = 10^{20}\,{\rm cm}^{-2}$, $\Gamma = 2$, and 0.5–8 keV flux of $10^{-12}\,{\rm erg\ cm}^{-2}\,{\rm s}^{-1}$. The spectrum represented by the hollow diamonds does *not* have the effects of pileup simulated. The spectrum represented by the filled blue diamonds has pileup simulated with a grade migration parameter of $\alpha = 1$. *Middle*: The same piled spectrum as on the left; however, it has been fit with an absorbed power-law model *without* assuming pileup. The derived photon index is $\Gamma = 1.46 \pm 0.04$ (90% confidence level). *Right*: The same piled spectrum as on the left, fit with an absorbed power-law and using the pileup model. The fit finds $\Gamma = 1.82 \pm 0.12$ and $\alpha = 0.26 \pm 0.06$ (90% confidence level; see text).

effects of pileup. This is especially true of 'featureless' spectra such as a power-law. As shown in Fig. 6, a piled power-law appears as an unpiled power-law with a different slope[2].

Here we used a value of $\alpha = 1$ in these simulations. This is clearly an unrealistic value (i.e., all piled events are retained as valid photon events). What value should users employ in their simulations? As of yet, there has been no systematic study of what values of α yield the most realistic results in actual observations. Our experience has been that values of α near the extremes of 0 or 1 are unrealistic. A reasonable suggestion is to use the mid-value of $\alpha = 0.5$[3]. For users who are especially concerned about the effects that different choices of α might have, it is suggested that they systematically set and freeze α to a variety of values in both their simulations and their fits, and then explore the effects of these choices on the other fit parameters of interest. As regards the other parameter choices available in the spectral-fitting package versions of the Davis (2001) pileup model, these are discussed in greater detail in §4.

For the case of pileup correction in a gratings observation, there are no models standardly available as part of any of the major spectral fitting packages. (A model is under development, and will be discussed in future versions of this document.) However, a simple `s-lang` user model, `simple_gpile`, has been developed for use in the *ISIS* spectral fitting package. It is meant to model pileup in *first order spectra only*. A copy of the code is found in Appendix C and is described in further detail in §4. This model is meant for cases of very mild gratings pileup, i.e., where the peak pileup fraction for *total* event loss, f_t is approximately $< 30\%$. This model uses the fact that incident counts/sec/Å is proportional to incident counts per frame per pixel, and that the detected first order count rate should be exponentially decreased by a factor that is proportional to this latter quantity. The normalization factor to be used is in fact a parameter of the model, `pile_frac`. The maximum decrease in first order events (occurring usually, but not always, near the peak of the effective area curve for the observation in question) in this model is given by $\exp(-\texttt{pile\_frac})$. The 'best choices' for the `pile_frac` parameter are described in §4. Given this model, however, one can simulate gratings spectra in a manner similar to that described above for CCD spectra.

[2]These points are discussed further at `http://space.mit.edu/cxc/analysis/PILECOMP/index.html`

[3]We have identified several piled observations, however, which exhibit more than 0.5 detected counts per frame, which is only possible for $\alpha > 0.5$ (see eq. 4 and Fig. 3). Thus, there is some evidence that values of $\alpha \sim 0.7$ might be reasonable. This is an issue that will be explored further in future revisions of this document.

2.4 MARX Simulations

MARX is the most sophisticated means of simulating observations with pileup; it is perhaps the best method for simulating pileup in gratings observations; and it is the only method for simulating pileup in imaging observations. As described in the *MARX* manual and web pages[4], *MARX* implements a version of the pileup model of Davis (2001), where users can choose instrumental parameters (e.g., integration frame time, offset pointing angles. the presence of gratings), as well as pileup parameters (i.e., the grade migration parameter, α). The resulting output can be extracted and analyzed very much in the same manner as real *Chandra* observations. For simple point source spectra without the use of gratings, it is likely that the simulation scripts presented in the appendices will suffice. However, if one wishes to determine whether implementing an offset pointing will be sufficient to reduce pileup in a point source, then *MARX* simulations are the best means of doing this. In this case, one can use *Sherpa*, *ISIS*, or *XSPEC* to create the input flux spectra (i.e., unfolded spectra) for the *MARX* simulation. (This is described in the *MARX* manual.)

MARX is also the preferred method for simulating pileup in gratings spectra. Although one can use the `simple_gpile` model (Appendix C) to create fake spectra in *ISIS*, the pileup amplitude is an assumed input parameter (albeit a parameter for which one can make educated estimates; see §4). *MARX* will yield a more self-consistently calculated pileup amplitude. Additionally, *MARX* is the only means of assessing the effects of pileup on higher order gratings spectra.

Although pileup over extended regions is rare (gratings observations of the Crab nebula are notable examples exhibiting extended pileup), *MARX* is the only means of simulating its effects. A perhaps more common case of 'imaging' pileup assessment is to study the pileup-induced distortion of the PSF (see Fig. 2). Again, *MARX* is the only means of performing such simulations. For further information creating simulations of piled spectra, users are referred to the *MARX* manual.

3 Pileup Detection

Currently, there are no tools for the automatic detection of pileup in a *Chandra* observation. A number of ideas are currently being researched (for example, using the measured PSF of a source to look for signs and assess the degree of pileup). As such methods are developed and tested, they will be described in future versions of this document. In the meantime, perhaps the best quick indicator of the presence of pileup is to employ a count rate selection criterion. As shown in Fig. 3, the pileup fraction is an increasing function of incident count rate, but a double valued function of *detected* count rate. Point sources with approximately less than 0.1-0.2 detected counts per frame integration time will either be on the branch of solutions with mild to moderate pileup, or very heavy pileup. The latter likely will be obvious from a very distorted shape of the PSF. Sources with approximately greater than 0.1-0.2 detected counts per frame are likely to be affected strongly by pileup. Again, as always, whether any given pileup fraction should be considered 'significant' will depend upon the scientific questions being addressed by the data.

4 Pileup Mitigation

Currently, the only available methods for pileup mitigation are for application to spectral observations. There are no standard methods for 'correcting' images or lightcurves (although see Tomsick et al. 2004 for a discussion of pileup effects in an example of a *Chandra* lightcurve). For observations without the gratings, *Sherpa*, *ISIS*, and *XSPEC* have implementations of the model of Davis (2001). We will describe the 'practical use' of this model below. For observations with the gratings, one can use the `simple_gpile` model of Appendix C, use of which will also be described below. An extension of the model of Davis (2001)

[4]See: `http://space.mit.edu/ASC/MARX/` and `http://space.mit.edu/ASC/MARX/examples.html`.

to the case of gratings observations, as described in Davis (2003), is in development, and will be described in later revisions of this document.

4.1 Correcting Imaging Observations

By 'correcting' or 'mitigating' pileup, what we actually mean is applying the effects of pileup to a model spectrum, and then forward folding this model through a detector response. This forward folded model prediction is then compared to to the observed spectrum[5]. This process is described in detail in Davis (2001). Here we describe the parameters of this model (common to its implementations in the three major spectral fitting packages), and give practical advice for this model's use.

There are a number of assumptions that go into the pileup model that must be kept in mind during its use, both in the data preparation phase and in the model fitting phase. The model presumes that simple counting statistics for a *steady source* apply. Therefore one first needs to examine the lightcurve and only extract spectra from intervals with comparable count rates. If one does wish to derive an 'average' spectrum for a variable source, the best procedure is still to separate the data into periods of nearly uniform rate, but then perform a joint fit to the resulting spectra (perhaps in the fit tying together the same parameter, aside from normalizations, from one spectrum to another). It is difficult to come up with an *a priori* number for the amount of variability within a lightcurve that one can tolerate in a pileup fit. The general rule of thumb, however, is that the greater the pileup fraction, the greater the need for uniformity in the lightcurve.

The pileup model also presumes that one has extracted all the good grade events, and hasn't unnecessarily removed data. The pre-`CIAO 3.2` 'afterglow detection' routine, `acis_detect_afterglow`, could, for bright, piled sources, accidentally remove events associated with source photons[6]. This tool has now been replaced with `acis_run_hotpix`, which does not suffer from this problem and is the norm for 'standard processing' of current data. For older data, one must make sure that afterglow correction has not been applied with `acis_detect_afterglow`, and has been removed if that tool has been used.

Spectra representing extended emission can also be analyzed with the pileup model. Similar to the restrictions on lightcurve variability, the model presumes spatial uniformity. Thus, one must make sure to only extract extended regions of similar count rate and spectral shape. As discussed below, the model contains a parameter for the number of independent regions (approximately 3×3 pixel islands) contained within the extraction region.

Also as discussed below, the model contains a parameter for the fraction of the spectrum that is within the central, piled, portion of the PSF (as opposed to the remaining fraction of the spectrum, assumed to be in the unpiled wings of the PSF). The default value of this parameter is 95%, i.e., 95% of the spectrum is subject to pileup, and 5% is left unpiled. This fraction is approximately correct if one extracts counts from an on-axis source from a circular region of 2 arcsec radius (≈ 4 pixel radius). That is, for that extraction radius, approximately 95% of the incident counts arrive within the central 1 arcsec radius, and are likely subject to pileup (for their given incident rate). Note that ideally, this fraction should be energy dependent, and this is another simplification introduced by the model. If one needs to extract a smaller region (for example, due to a neighboring source), one might expect the fraction subject to pileup correction to increase. If one extracts a larger region, or moves off axis, one might expect the fraction subject to pileup to decrease. One might also expect this fraction to be greater for very soft sources as opposed to very hard sources. Given that for all energies below 6.4 keV 85% *or more* of the photons arrive within the central 1 arcsec, one should always expect this parameter value to be > 0.85.

[5]This brings up a subtle point. The spectral fitting packages compare the model to the data presuming the usual counting statistics (or other statistics defined by the user). This is in contrast to *MARX*, where the counting statistics are first calculated and then the pileup is applied. The latter is more correct. However, given the likely large uncertainties in the pileup model (e.g., the grade migration parameter), the former is deemed acceptable in the context of these models.

[6]See the thread at `http://cxc.harvard.edu/ciao/threads/acisdetectafterglow/`.

To reiterate, in preparing the data for use with the pileup model, one should extract events that: have uniform rates in both time and across extended regions (if applying the pileup model to an extended source), do not have afterglow correction applied to them with the `acis_detect_afterglow` tool, and come from an approximately 4 pixel radius region (if extracting an on-axis point source, without neighboring sources closer than 4 pixels). In actually applying the pileup model, there are up to 7 parameters: `nregions` (`n/nregions`), `g0` (`g0/g0`), `alpha` (`alpha/alpha`), `psfrac` (`f/psfrac`), `nterms` (`/max_ph`), and `fracexpo` (`fracexp/`) The names here refer to the parameter names under *ISIS*, with the latter two being set via the `set_kernel` command rather than in the fit function. The parameter names in parentheses are those under *Sherpa*/*XSPEC*, respectively. Additionally, *Sherpa* and *XSPEC* have an explicit parameter for frame time, `ftime`/`fr_time`. (In *ISIS*, the frame time is automatically read from the data file header, but can be overridden with the `set_frame_time` command.) We describe these parameters below.

- `nregions (n/nregions)`: Divide the model counts among `nregions` regions, to which the pileup model will be applied independently. This should be approximately the number of 3×3 pixel islands in the extracted spectra. For point sources, it is unity. In either case, it should remain a frozen parameter in the fits.

- `g0 (g0/g0)`: Grade correction for single photon detection. I.e., a fraction `g0` of single photon events will be retained as good grades. In practice, this should be frozen to unity in any fit.

- `alpha (alpha/alpha)`: The grade migration parameter, such that the probability of `n` events piled together in a single frame being retained as a 'good grade' is $\mathtt{alpha}^{\mathtt{n}-1}$. This parameter can range from 0 to 1, and is the parameter most likely to be allowed to vary in a fit.

- `psfrac (f/psfrac)`: The fraction of events in the source extraction region to which pileup will be applied. A typical value is ≈ 0.95, and it should always be > 0.85. This is the second most likely parameter to be allowed to vary in a fit.

- `nterms (/max_ph)`: This is the maximum number of photons considered for pileup in a single frame. For practical purposes, this should be left frozen at the maximum value of 5.

- `fracexpo (fracexp/)`: This parameter is a fraction ≤ 1 that multiplies the frame exposure time to create a shorter, effective frame exposure time. This can be used to model, for example, deadtime effects (e.g., from cosmic ray events). Important note: the default *Sherpa* value is such that when it is combined with *Sherpa*'s default frame time (3.24104 sec), the effective exposure per frame is 3.2 sec. Unless one is explicitly attempting to model novel deadtime effects, *Sherpa* users should set and freeze `fracexp` to unity, and set the frame exposure time via the `ftime` parameter only.

- Frame Time `(ftime/fr_time)`: In *ISIS*, the frame time is automatically read from the data file header (or defaulted to 3.2 sec if a frame time cannot be read). This value can be overridden with the `set_frame_time` command. In *Sherpa* and *XSPEC*, the frame time is a parameter of the fit function, and should be set to the good exposure time per frame (presuming that `fracexp` has been set to unity). This should be equal to the `EXPTIME` keyword in the header (the `TIMEDEL` keyword includes an additional 41.04 msec, for the readout time per frame).

Although there are seven potential parameters of the model, three should almost never be changed (`g0`, `nterms`, `fracexpo`), two should be frozen to values based upon the observations (`nregions`, and frame time), leaving only two to potentially be used as fit parameters (`alpha`, `psfrac`). In terms of 'practical advice', even these two should be left frozen in initial fits to the data. We suggest that `alpha` be frozen to a value between 0.5-0.7, and `psfrac` be frozen to 0.95 at the start of the fitting process. One should first explore the parameters of the model that is being piled, especially as regards model normalization.

As shown in Fig. 3, the pileup model can be 'double valued' in terms of incident count rates, vs. detected count rates. I.e., for a given detected count rate, there is a high flux solution and a low flux solution. With the pileup parameters initially frozen, the user should determine which branch of solutions is most correct for the observation in question. Additional constraints on likely normalizations can be obtained by looking at other information from the data set. Can an incident count rate be determined from a readout streak? Are there enough counts in the wings of the PSF to make an estimate? Are there archival or concurrent observations that can help one to estimate the expected incident count rate? Referring to Fig. 3, one sees that the low and high flux solutions are often separated by factors of three or more in incident flux. If one can reasonably estimate which branch the fit properly belongs to, model normalization parameters should be constrained so as not to allow the fit to 'wander' to the other branch.

Note, however, that the curve of detected counts vs. incident counts is fairly flat in the regime of ≈ 0.5 detected counts per frame. (This is assuming `alpha` $= 0.5$, but the result is not very different for nearby values of `alpha`.) This will be an intrinsically very difficult regime to fit with the pileup model.

Once an initial fit has been obtained, and the model normalization parameters have possibly been constrained to avoid an unwanted (low or high flux) branch of solutions, one can explore fits with `alpha` and `psfrac` thawed. At this point, there potentially might be numerous 'local minima' in the fitting process. To reduce the number of local minima, it is again advantageous to freeze or constrain other fit parameters, if possible. Archival or contiguous observations, from *Chandra* or other observatories, may prove useful. If one is fitting the pileup model to a $0^{\rm th}$ order gratings spectrum, fits to the dispersed spectrum will provide a useful guide. As another example, simultaneous *RXTE* observations might constrain a power law component that extends to higher energy. Archival *ASCA* data might constrain a lower energy component, if one has reason to believe that it is not time variable. Surveys might suggest reasonable limits for any neutral column in the fit.

In addition to limiting the fit ranges of other model parameters, it is also useful to use several different fit methods. *ISIS* offers four (`lmdiff`, `minim`, `levenberg-marquardt` and `subplex`), while *Sherpa* offers a number of methods (`levenberg-marquardt`, `simplex`, `powell`, among others, as well as 'Monte Carlo' versions of these methods). Under *ISIS* we suggest that both `lmdiff` and `minim` should be tried in any pileup fit. Under *Sherpa*, `simplex` and `powell` both are recommended to be tried in any pileup fit. (In CIAO 3.2, the `levenberg-marquardt` method has encountered some problems in pileup fits on some operating systems. This is expected to be rectified in a subsequent CIAO release.) Given possibly very long run times, the 'Monte Carlo' versions of these fit methods are not recommended for pileup use in *Sherpa*. Instead, for both *Sherpa* and *ISIS*, searching for lower χ^2 minima can often be accomplished by initiating error searches on a number of the fit parameters, and then refitting if the error search finds a new χ^2 minimum. This procedure of fitting, error searching, then refitting, often can be faster than using one of the 'Monte Carlo' fit methods. (Additionally, it is simple enough to implement the fit-error search-refit paradigm via a `s-lang` script in either *Sherpa* or *ISIS*.)

Finally, it is worth noting that even when all of the above suggestions are employed, the pileup fit might produce a range of fits with only minimal differences in χ^2. This can be especially true for the fit parameter `alpha`, with a wide range of `alpha` producing fits of comparable χ^2. In such cases, it is suggested to freeze `alpha` at a range of values and at each fit the spectrum and determine error bars for the other model parameters. The variations in the model parameters with `alpha` can then be treated as systematic errors.

4.2 Correcting Dispersed Gratings Observations

The pileup correction for CCD spectra, given a presumed or fitted α parameter, incorporates a scheme for calculating a pileup fraction that is internally self-consistent, given the assumptions of the model. There currently is no such similar model for gratings spectra. One is under development, and it is expected to be incorporated into a future CIAO release.

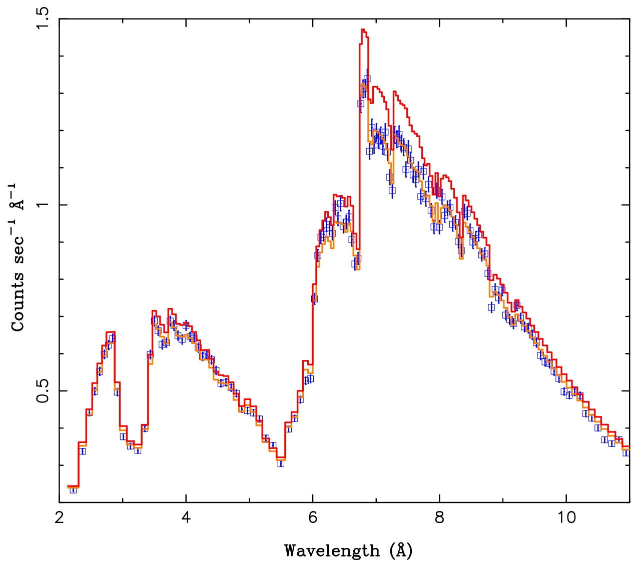

Figure 7: A mildly piled gratings spectrum, from the MEG -1 order. The orange line (passing through the data) is a fit utilizing the `simple_gpile.sl` model, with a peak pileup fraction of `pfrac`=0.1. The red line (extending above the data) is the same model with the pileup fraction set to `pfrac`=0.

A simple `s-lang` based gratings pileup correction/estimation model, `simple_gpile.sl`, has been developed for use in ***ISIS***. The model code is presented in Appendix C. Unlike the CCD pileup correction, here the pileup fraction is a *parameter* of the model. It can be left unfrozen and 'fit'; however, it is instead envisioned that in most circumstances this parameter will be analytically estimated and left frozen, or perhaps varied within a very limited range of values. Furthermore, the model is only applicable to *first order* gratings spectra (although it can also employ information from the 2nd and 3rd orders).

The model uses the concept that for first order gratings spectra, pileup results in a wavelength-dependent exponential loss of flux, based upon the counts per frame per pixel (see eq. 4). For the gratings, counts per frame per pixel is proportional to counts per second per Å. Given that the response matrix for the gratings is nearly diagonal, this latter rate is nearly equal to the computed model flux multiplied by the detector effective area (i.e., the 'arf' for the observation). Thus the `simple_gpile.sl` model is implemented as a simple convolution model. The unpiled model flux is calculated, multiplied by the detector effective area, and then scaled to a given pileup fraction. An exponential of this scaled model count rate is taken, and then multiplied by the unpiled model. The default is to use only the first order count rate, but the second and third order count rates can be used as well.

The chief drawback of this approach is that one must assume a value of the pileup fraction, f_t, rather than calculate it from first principles. (The under-development gratings pileup model, however, will automatically estimate the resulting pileup fraction.) Reasonable guesses for the value of f_t can be made analytically. Typically, the greatest degree of pileup occurs in the MEG gratings, near the peak of its effective area at approximately 6 Å. (This statement is, of course, dependent upon incident spectrum.) At this wavelength, the MEG has larger effective area than the HEG, and has pixels that cover twice the wavelength range (i.e., it has half the spectral resolution of the HEG). Thus the peak pileup fraction in the MEG tends to be approximately two times larger than the peak pileup in the HEG, and this can often be assumed when employing the `simple_gpile.sl` model.

As a rough estimate of the peak pileup fraction, one can scale from an estimated model flux. If the model flux at 6 Å (presumed to be the location of the peak MEG *observed count rate*) is given by, for example, 10^{-2} counts s^{-1} cm^{-2} Å, then multiplying by the approximate detector effective area of 80 cm^2, a 3.2 s frame time exposure, a 3 pixel long detector region, and the MEG resolution of 0.011 Å/pixel, we obtain an expected rate of 0.085 counts/three pixels/frame. Given that at this rate we are in the linear regime for estimating the pileup fraction, f_t, a good initial guess at the pileup fraction to use in the `simple_gpile.sl` model is 8.5% (i.e., set `pfrac`=0.085). These values can be scaled for other fluxes and frame times (e.g., if a sub-array has been chosen for the gratings observation).

Alternatively, one can plot the data in units of counts/s/Å, and multiply the peak value by 0.033×3.2 (for MEG data at the full frame time of 3.2 s) or 0.016×3.2 (for HEG data at the full frame time of 3.2 s). This then can serve as a useful lower limit for the value to assign to `pfrac`. Given the simplicity of the assumptions of this model, it is not suggested to be used on data that would require `pfrac` $>$ 0.2-0.3. In Fig. 7, we show this model applied to gratings data with `pfrac`=0.1.

5 Further Resources

This document has provided a rough overview of the effects of pileup in *Chandra* observations, and discussion of avoidance, detection, and mitigation strategies. For more detailed and mathematical discussions of the theoretical underpinning of pileup and its mitigation, the user is referred to the two articles Davis (2001) and Davis (2003). As noted earlier, a good discussion of some of the effects of pileup on timing analysis can be found in the article by Tomsick et al. (2004). Those users wishing to do the most careful simulations of pileup in *Chandra* observations, especially for imaging or gratings observations, are encouraged to consult the *MARX* manual. Users with questions not answered by this document should consult the *Chandra* help desk at: `http://asc.harvard.edu/helpdesk/`. Suggestions for improvements or additions to this document can be e-mailed to Michael Nowak at: `mnowak@space.mit.edu`.

References

Davis, J. E. 2001, ApJ, 562, 575

Davis, J. E. 2003, in X-Ray and Gamma-Ray Telescopes and Instruments for Astronomy. Proceedings of the SPIE, Volume 4851, ed. J. E. Truemper & H. D. Tananbaum, 101–111

Tomsick, J. A., Kalemci, E., & Kaaret, P. 2004, ApJ, 601, 439

Appendix A - Example Sherpa Script for Faking Piled Data

```
# Save in file ‘sherpa_script’, and run as: sherpa> use sherpa_script

# Load Cycle 7 ACIS-S3 (no gratings) arf and rmf obtained from
# http://cxc.harvard.edu/caldb/prop_plan/imaging/index.html

rsp[aciss](aciss_aimpt_cy07.rmf,aciss_aimpt_cy07.arf)
instrument = aciss

# Use an absorbed power-law model

source = xsphabs[a]*xspowerlaw[b]

# Neutral hydrogen column of 10ˆ20, temporary norm, photon index = 2

a.nh = 0.01
b.norm = 1.
b.PhoIndx = 2.

# Model flux in 0.5-8 keV energy band

modflux  = get_eflux(1,[0.5,8.])

# Reset the power-law normalization to yield 10ˆ-12 erg cmˆ-2 sˆ-1

newnorm = 1.e-12/modflux.value
plaw = get_par("b.norm");
plaw.value = newnorm
() = set_par(plaw)

# Set exposure time to 50 ksec and fake the data

fakeit time=5.e4
fakeit

# Write the fake data to a PHA file

write data fake_nopile.pha pha

# Create the pileup model and set frame time to 3.2 s,
# and fractional exposure to 1

jdpileup[jdp]
pileup = jdp
jdp.ftime = 3.2
jdp.fracexp = 3.2
```

```
# Fake the data, and write it to a pha file

fakeit
write data fake_piled.pha pha

# You now have two data sets saved on disk.  Use dmgroup to
# rebin.  Read in regrouped data sets, and analyze as usual!
```

Appendix B - Example ISIS Script for Faking Piled Data

```
% Run as: isis> () = evalfile(``isis_script.sl'');

% Load Cycle 7 ACIS-S3 (no gratings) arf and rmf obtained from
% http://cxc.harvard.edu/caldb/prop_plan/imaging/index.html

() = load_arf("aciss_aimpt_cy07.arf");
() = load_rmf("aciss_aimpt_cy07.rmf");
assign_arf(1,1);                       % Create an empty data set
assign_rmf(1,1);                       % by assigning an arf/rmf

require("xspec");                % Load the xspec models
fit_fun("phabs(1)*powerlaw(1)"); % Use an absorbed power-law model

set_par(1,0.01);  % Neutral hydrogen column of 10^20
set_par(2,1);     % Temporary power-law normalization
set_par(3,2);     % Power-law photon index

% Define a function to determine the flux in a given keV band

define kev_flux (id, kev_lo, kev_hi)
{
   % convert from wavelength grid [A] to energy grid [keV]
   variable m = _A(get_model_flux(id));

   % convert photons/sec => ergs/sec
   m.value *= 0.5 * (m.bin_hi + m.bin_lo) * 1.602e-9;

   % return integral over specified band [erg/sec]
   return rebin (kev_lo, kev_hi, m.bin_lo, m.bin_hi, m.value)[0];
}

() = eval_counts;                   % Evaluate the model
variable flux = kev_flux(1,0.5,8.); % Flux in the 0.5 to 8 keV band

set_par(2,1.e-12/flux);    % Renormalize to 10^-12 erg cm^-2 s^-1
set_arf_exposure(1,5.e4); % Set the exposure time to 50 ksec
fakeit;                   % Fake the data without pileup

set_fake(1,0); % Mark data as "real" so 'fakeit' won't overwrite

assign_arf(1,2);           % assign same arf and rmf to second set of
assign_rmf(1,2);           %  fake data that will have pileup applied
set_kernel(2,"pileup");    % assign pileup kernel to new data
set_frame_time(2,3.2);     % assign the nominal frame time to new data
fakeit;                    % Create fake data, with the pileup model,
                           % then analyze the data as usual!
```

Appendix C - The `simple_gpile.sl` Model for *ISIS*

```
% To run, save in a file, e.g., `simple_gpile_init.sl', and then in ISIS,
% isis> () = evalfile(``simple_gpile_init.sl'');

define simple_gpile_fit(lo,hi,par,fun)
{
   % Peak pileup correction goes as exp(log(1-pfrac)*[counts/max(counts)])
   variable pfrac = par[0];

   % Pileup scales with model counts from *data set* indx ...
   variable indx = typecast(par[1],Integer_Type);

   % ... but the arf index could be a different number, so get that
   variable arf_indx = get_data_info(indx).arfs;

   % Get arf information
   variable arf = get_arf(arf_indx[0]);

   % Rebin arf to input grid, and multiply by `fun' to get model
   % counts per bin
   variable mod_cts = fun*rebin(lo,hi,arf.bin_lo,arf.bin_hi,arf.value);

   % Go from cts/bin/s -> cts/angstrom/s
   mod_cts = mod_cts/(hi-lo);

   % Use 2nd and 3rd order arfs to include their contribution.
   % Will probably work best if one chooses a user grid that extends
   % from 1/3 of the minimum wavelength to the maximum, and has
   % at least 3 times the resolution of the first order grid.

   variable mod_ord;
   if(par[2] > 0)
   {
      indx = typecast(par[2],Integer_Type);
      arf = get_arf(indx);
      mod_ord = arf.value*rebin(arf.bin_lo,arf.bin_hi,lo,hi,fun);
      mod_ord = rebin(lo,hi,2*arf.bin_lo,2*arf.bin_hi,mod_ord)/(hi-lo);
      mod_cts = mod_cts+mod_ord;
   }
   if(par[3] > 0)
   {
      indx = typecast(par[3],Integer_Type);
      arf = get_arf(indx);
      mod_ord = arf.value*rebin(arf.bin_lo,arf.bin_hi,lo,hi,fun);
      mod_ord = rebin(lo,hi,3*arf.bin_lo,3*arf.bin_hi,mod_ord)/(hi-lo);
      mod_cts = mod_cts+mod_ord;
   }
```

```
   % In case something funky happens while wandering parameter space ...
   variable max_mod = max(mod_cts);
   if(max_mod <= 0){ max_mod = 1.;}

   % Scale maximum model counts to pfrac
   mod_cts = -log(1-pfrac)*mod_cts/max_mod;

   % Return function multiplied by exponential decrease
   fun = exp(-mod_cts) * fun;

   return fun;
}

add_slang_function("simple_gpile",
                   ["pile_frac","data_indx","arf2_indx","arf3_indx"]);
set_function_category ("simple_gpile", ISIS_FUN_OPERATOR);

define simple_gpile_defaults(i)
{
   switch(i)
   {case 0:
      return (0.1,1,0.,0.3);
   }
   {case 1:
      return (1,1,1,12);
   }
   {case 2:
      return (0,1,0,12);
   }
   {case 3:
      return (0,1,0,12);
   }
}

% The following is required only for ISIS versions > 1.2.9

set_param_default_hook ("simple_gpile", "simple_gpile_defaults");
```