Kepler: A Search for Terrestrial Planets

Algorithm Theoretical Basis Document
for the
Science Operations Center

Jon M. Jenkins, Douglas A. Caldwell and Ron Gilliland

KSOC-21008-001
Friday, July 30, 2004

NASA Ames Research Center
Moffett Field, CA. 94035

Document Control

Ownership

This document is part of the Kepler Project Documentation that is controlled by the Kepler Project Office, NASA/Ames Research Center, Moffett Field, California.

Control Level

This document will be controlled under KPO @ Ames Configuration Management system. Changes to this document **shall** be controlled.

Physical Location

The physical location of this document will be in the KPO @ Ames Data Center.

Reference Documents

KKPO-16001 Data Release and Publication Policy

Distribution Requests

To be placed on the distribution list for additional revisions of this document, please address your request to the Preparer:

Jon Jenkins
Kepler Analysis Lead (Co-Investigator)
M/S 244-30
NASA Ames Research Center
Moffett Field, CA 94035

E-mail: jjenkins@mail.arc.nasa.gov

DOCUMENT CHANGE LOG

CHANGE NUMBER	CHANGE DATE	PAGES AFFECTED	CHANGES / NOTES
001	July 30, 2004	All	Initial Release

Approved by: ______________________ Date 7/30/04
[Larry Webster, Deputy Project Manager]

Approved by: William J. Borucki Date 8/2/2004
[William Borucki, Principal Investigator]

Approved by: D Koch Date 7/29/04
[David Koch, Deputy Principal Investigator]

Approved by: Janice Voss Date 7/28/04
[Janice Voss, Science Director]

Approved by: ______________________ Date 7/29/04
[Jeffrey Garside, Signal Processing Scientist]

Approved by: ______________________ Date 8/2/04
[Jeff Shapiro, SOC Lead Engineer]

Prepared by: ______________________ Date 7-29-04
[Jon Jenkins, Lead Signal Processing Scientist (Co-Investigator)]

Preface

This volume is intended for scientists and engineers with an interest in the processing of the *Kepler* science data stream. The goal of this document is to describe the physical processes and instrumental characteristics of the CCD data collected by *Kepler*, and the scientific processing applied to the measurements leading to the detection and characterization of planets in the data set. In addition, the theoretical bases and rationale are given for the non-pipeline software: target list management, the Quick Look software overview, the management of the onboard data compression parameters, onboard cosmic ray detection parameters, and End-To-End Model description. Finally, numerous appendices contain code and relevant derivations.

Contents

Chapter 1

Introduction

The *Kepler* Mission is designed to characterize the frequency of Earth-size planets in the habitable zones of solar-like stars in the solar galactic neighborhood. *Kepler's* approach is to observe $>$100,000 main-sequence stars in a $>$100 square degree field of view (FOV) centered on 70° galactic longitude, +5° galactic latitude in the constellation named Cygnus. As given in Table 1.1, there are about 200,000 *main sequence* stars in *Kepler's* FOV to m_R=15.0. The Stellar Classification Program (SCP) is underway to identify these stars among the total of roughly 500,000 stars to that magnitude. *Kepler* will search for evidence of transiting planets in the FOV by monitoring the brightnesses of the chosen target stars. In addition, *Kepler* will search for evidence of modulation of stellar flux due to Close-in Extrasolar Giant Planets (CEGPs).

Transiting planets exhibit drops in brightness proportional to the ratio of the area of the planet to the area of its parent star. An Earth-sized planet transiting a solar-like star generates a transit depth of $\sim$80 ppm, and for a 1 AU orbit, each transit lasts as long as 13 hours, depending on the exact inclination angle. The transit of a Jupiter-sized planet in front of a solar-like star, however, is $\sim$1% deep, since Jupiter's radius is about 0.1 times that of the Sun. CEGPs, like HD209458, take only $\sim$3 hours to transit their stars. Thus, in searching for transiting planets, *Kepler* is interested in finding negative pulses lasting from a few hours to as long as $\sim$16 hours (corresponding to a central transit for a 2 AU orbit).

The signature of the reflected light component of CEGPs is not as well-constrained as that of a transiting planet. The detailed shape and amplitude of the reflected light depend a great deal on the properties of the planet and its atmosphere, in particular, the size of the planet and distance from the star. Atmospheric composition and cloud structure are also important. Nevertheless, the existing models predict that the reflected light component should have a peak-to-peak amplitude of 20-100 ppm for Jovian-sized planets within 0.1 AU. In searching for such planets, *Kepler* will be seeking evidence of quasi-sinusoidal signals with periods as long as 7 days. Beyond this orbital range, the amplitude of the reflected light signature drops precipitously, since it is inversely proportional to the square of orbital radius.

In order to detect such small signals, the photometric precision of measurements must be exquisite compared to that routinely obtained by ground-based observations. The Combined Differential Photometric Precision (CDPP) for a G2V m_R=12 star in the FOV should be no more than 20 ppm, including stellar variability. Moreover, the nature of the signals *Kepler* seeks requires extensive observations. *Kepler* will continuously image the FOV over its primary mission of 4 years, and for as long as 6 years, if the extended mission is approved.

The primary role of the Science Operations Center (SOC) is to produce detrended normalized relative flux time series for each target star, and to search through these light curves for evidence of transiting planets and/or modulation of reflected light by CEGPs. The target stars are to be selected by the Science Office (SO) using the *Kepler* Input Catalog and additional relevant stellar catalogs. The original data consists of pixels containing each target star from the 42 CCDs in the focal plane of the photome-

Table 1.1: Modeled Number of Main-Sequence Stars in *Kepler's* Field of View

				Spectral Type				Cumula-
m_R	B5	A5	F5	G5	K5	M5	All	tive Sum
9.5	151	299	200	86	20	0	756	756
10.5	481	838	706	358	80	0	2463	3219
11.5	1002	2181	2248	1300	242	7	6979	10198
12.5	1832	5004	7037	4189	991	46	19098	29296
13.5	3051	10245	19796	13379	3271	167	49909	79205
14.5	4498	18142	51098	42035	10969	611	127353	206558
Total	11014	36708	81085	61347	15573	831	206558	——

ter, together with collateral data to correct for systematic effects incurred in operating the CCDs, and ancillary data comprised of a subset of engineering data for parameters that may be used as diagnostics or for identifying and correcting systematic effects in the photometric data.

The original data are downlinked from the *Kepler* spacecraft through the Deep Space Mission System (DSMS) to the Mission Operations Center (MOC), that performs completeness checking and formulates re-transmit requests for data packets that are corrupted or missing at the end of each pass. A SOC-provided First Look Platform examines the photometric data received during each pass and makes a number of statistical checks to verify that the science data meets requirements in terms of quality. The results are reported to a web page accessible by all *Kepler* elements. Complete photometric data sets are transported to the Data Management Center (DMC) at the Space Telescope Science Institute (STScI), along with a subset of the engineering data called 'ancillary data'.

The DMC performs pixel-level calibrations, and then passes the calibrated pixels on to the SOC where the photometric light curves are extracted and the planet searches are conducted. Almost all the stars are to be sampled every 15 minutes, with individual exposures co-added together within each 15-minute block, with occasional Full Field Images (FFIs) being provided. A small subset of stars (512) will be sampled at a much higher rate of 1 min^{-1}. Co-I's Ron Gilliland (STScI) and Tim Brown (High Altitude Observatory – HAO) will lead the effort to investigate and study pressure mode (p-mode) oscillations of bright target stars (> 10^{12} e^{-} month^{-1}). These 'short cadence' targets will also be used for stars identified with high SNR transits of giant planetary companions, to enhance the science return for such objects. In generating the calibrated light curves, the SOC must specify the pixel level calibrations to be performed at the DMC, and must develop software to combine target star pixels together into raw fluxes, and calibrate the stellar fluxes to remove systematic effects such as residual spacecraft pointing offsets. The calibrated stellar light curves will be archived at the DMC for eventual release to the general community following the Data Release and Publication Policy (KKPO-16001) adopted by the *Kepler* Mission. Any candidates identified by the search must be validated by estimating the statistical confidence in the detections, and the possibility that nearby, background variable stars could be the source of the photometric variations.

In addition to the science processing, the SOC must carry out a number of tasks related to target and photometer management. These include preparing target definitions for the photometer and specifying other photometer operating parameters such as integration time and focus position. While we do not expect to exercise all of these options in the post-commission phase, the SOC needs to provide the tools and data analyses to support the SO in making decisions regarding operations such as re-focus, decontamination, and annealing the focal plane. The

SOC also has the responsibility of tracking the actual CDPP and estimating the theoretical CDPP for each target star. The SOC must provide Huffman coding tables for the onboard compression of the CCD pixels and be able to update these as necessary to maintain adequate compression rates for the Solid State Recorder (SSR).

This Algorithm Theoretical Basis Document (ATBD) can be broken into two main parts: pipeline and non-pipeline processing.

The science pipeline processing details the steps in processing the original data downlinked from the spacecraft to generate calibrated stellar light curves. This part includes the algorithms required to search for planetary signatures in the light curves, and the means used to validate any candidates using Kepler Mission System (KMS) data from the spacecraft. Many of the algorithms baselined for these tasks have been developed and prototyped in the course of pathfinder activities such as the Vulcan Camera search for 51 Peg-like planets and the Kepler Tech Demo. As such, the theoretical bases for many of these algorithms have been published in several peer-reviewed journal articles. Indeed, many of these algorithms are in current use either as part of Vulcan, the KTD or as part of simulation efforts such as the End-To-End Model (ETEM), a MATLAB Monte Carlo model of the *Kepler* photometer.

The second part of this ATBD describes the methodology for developing algorithms to manage the Target List and the Photometer. Tasks falling into this category include monitoring and predicting the CDPP, managing the Huffman compression tables, selecting aperture masks for the target stars. Most of these tasks have not been carried through to the point of producing prototype software. Therefore, the description of these algorithms is more preliminary and less specific than for the first part.

Part I

Science Pipeline Processing

The organization of Part I is as follows: Chapter 2 presents a general, high-level overview of the processing steps from photoelectrons to detecting transiting planets and CEGPs. Chapter 3 lists and discusses the steps necessary to calibrate the original data set consisting of CCD pixel values transmitted from the spacecraft. This includes black level subtraction, nonlinearity correction, flat field correction, smear and dark subtraction, and background flux removal. Chapter 4 describes the process of extracting photometric light curves from the calibrated pixel data set, including the tasks of forming photometric apertures, determining ensemble stars, and removing residual systematic errors. A one-dimensional example is provided. This is followed by Chapter 5 which describes difference image analysis, a tool shown to provide excellent performance for stellar photometry on time-series based data. Specific topics include mean image creation and registration, PSF changes, difference images, false positive elimination, and hot pixel tracking. Chapter 6 presents an overview of detection theory with an emphasis on its application to detecting planets. Chapter 7 discusses the use of the DIARAD/SOHO data along with adaptive wavelet-based matched filters to detect transit candidates. It addresses the assessment of statistical confidence levels via a bootstrap approach. Chapter 8 specifically details the detection algorithms intended for use in identifying CEGPs by reflected light. This chapter also provides sections detailing statistical tests to be used to establish confidence in the candidates (i. e., rejecting false positives due to statistical fluctuations in the light curves). Part I concludes with chapter 9. It covers the establishment of statistical confidence in detections, describes how centroids can be used to reject confusion posed by background eclipsing binaries or variable stars, discusses the development of crowding parameters, and breifly touches on the assessment of the physical parameters of a detected planet.

Chapter 2

Overview of Science Processing

Figure 2.1 shows the primary activities to be carried out in the Science Operations Center (SOC) and the Data Management Center (DMC) during the *Kepler* Mission. The tasks are broken into the following parts: 1) Aperture Photometry, 2) Difference Image Analysis, 3) Single Event Statistics / CDPP, 4) Transiting Planet Search, and 5) Reflected Light Search. The ATBD will concentrate on further defining and providing theoretical support for those tasks.

Kepler's focal plane is populated with 42 CCDs, each of dimension 2200 columns by 1044 rows, with 27-μm wide pixels. Each pixel subtends $\sim$3.98 arc seconds of sky on a side, and the point spread function (PSF) is approximately 5 pixels wide (at the 95% level), so that each star will illuminate approximately 25 pixels. For design purposes, we assume that 32 pixels will be specified per star on average, and that the pixels will be co-added for 15 minutes before being stored in the solid state recorder (SSR) for downlink to Earth. A small number (512) of stars will be sampled every minute, but this detail does not radically alter the algorithms to be used in processing the data. Although there are about 100 million pixels in the FOV, only the pixels of interest containing target stars, or pixels that can be used to remove systematic errors, such as black level, smear and dark current, and background flux are downlinked. During the first year of operation, $\sim$6 million pixels will be downlinked for each 15 minute cadence.

Once the data arrive at the DMC, they are first corrected at the pixel level for those systematic errors that can be estimated directly from the CCD frames themselves, such as black level (bias), dark current, and the effects of shutterless readout (smear). The SOC will identify and track hot and/or dead pixels and perioidically providing updates to the DMC. The remainder of the processing is conducted at the Science Operations Center (SOC) at NASA Ames Research Center in Moffett Field, California. Here we will correct for background flux, extract photometric light curves for each star, correct for residual systematic errors, and conduct searches for transiting planets and for the reflected light from Close-in Extrasolar Giant Planets (CEGPs).

Once candidates are identified, a series of statistical tests will be conducted to assess the confidence level of each candidate, as well as to determine if the planet-like photometric variations are due to a nearby background eclipsing binary or variable star. This latter test will be conducted by examining the centroid time series for the candidates and correlating it with the photometric time series. If there is a strong correlation between the centroid motion and the photometry, then it is highly likely that the photometric variations are due to a dim background star located in the target star aperture but offset from it by $>$1 arcsec. If a candidate has an adequate confidence level and does not show evidence of correlated centroids, then it is subjected to further scrutiny by the Follow-up Observations Program (FOP) which will consist of a set of ground-based and perhaps space-based observations to rule out other sources of confusion. The physical parameters of viable candidates will be extracted from the photometry using detailed models of transit shape and stellar limb darkening. That is, we will determine the best-fit period, duration and depth of candidate transiting planets from the observations, together with error bars on the retrieved quantities.

The remainder of this part describes the various processing steps in detail, as well as furnishing fundamental theoretical discussions and background material germane to the selection of algorithms.

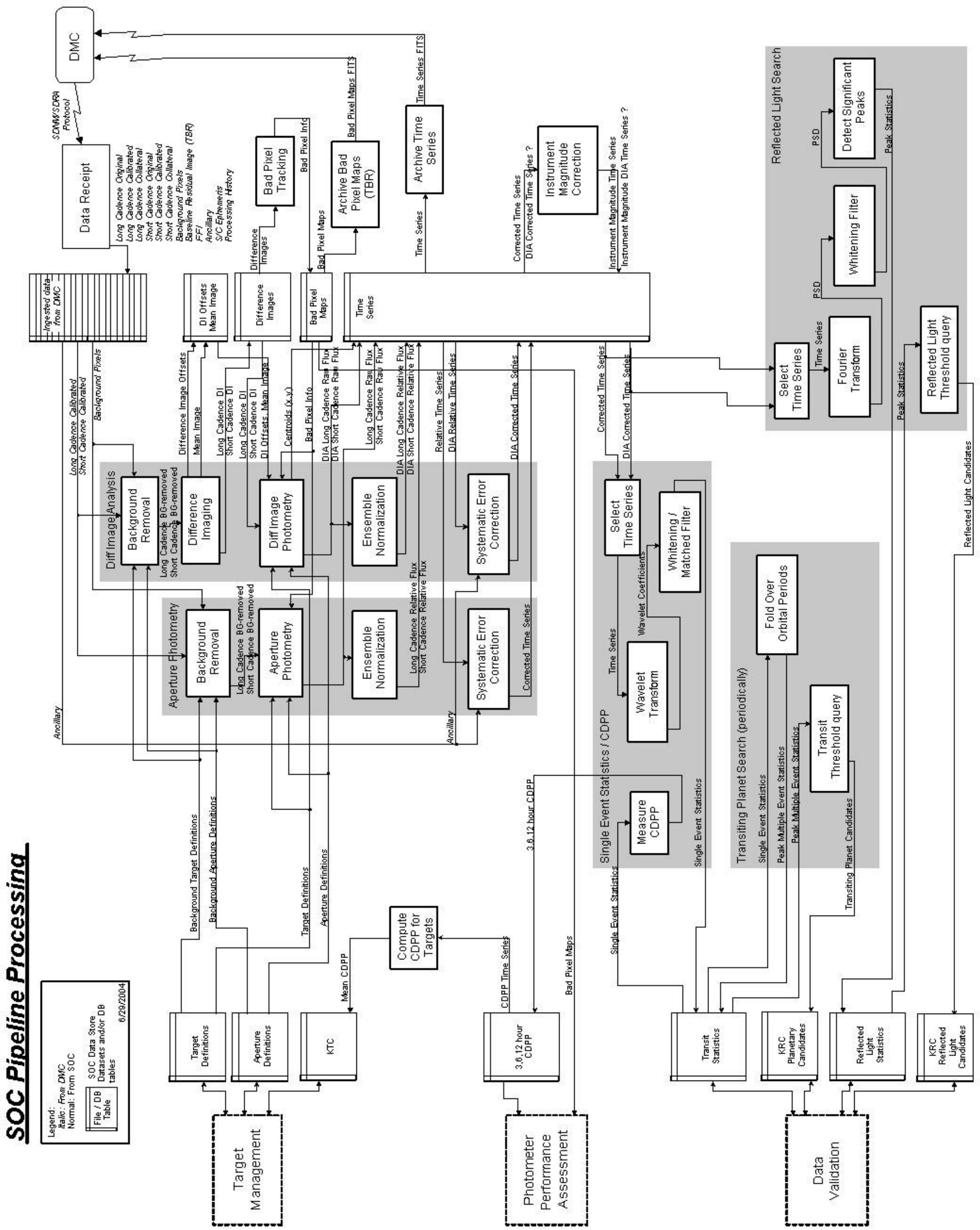

Figure 2.1: The data flow for processing the science data obtained from *Kepler*.

Chapter 3

Pixel-Level Calibrations

This chapter details pixel-level corrections made to the original data set to obtain calibrated pixel values. These corrections counter certain systematic artifacts of the CCD operation, such as black level (bias), dark current, and smear from shutterless operation. We also discuss pixel-level corrections that may be applied to the data, such as flat field and nonlinearity corrections, that may not be supported during mission operations. Most of these calibrations are performed at the DMC including black-level subtraction, and smear subtraction. The SOC identifies and tracks hot/dead pixels for use in generating calibrated light curves.

Figure 3.1 shows the layout of each of *Kepler's* 42 science CCDs. Each CCD consists of 2200 columns by 1044 rows of physical pixels, with two readout amplifiers located at the bottom corners of the CCD. The bottom 20 rows are covered with aluminum to allow for an estimate of the dark and the smear charge to be made for each column on each exposure. These pixels are not suitable for imaging target stars and are excluded from the usable FOV area. As the data received by the DMC are compressed, the first task is to reconstruct the 15-minute pixel values from the compressed data stream. Chapter 12 describes the baseline compression scheme in detail, but we summarize the process here.

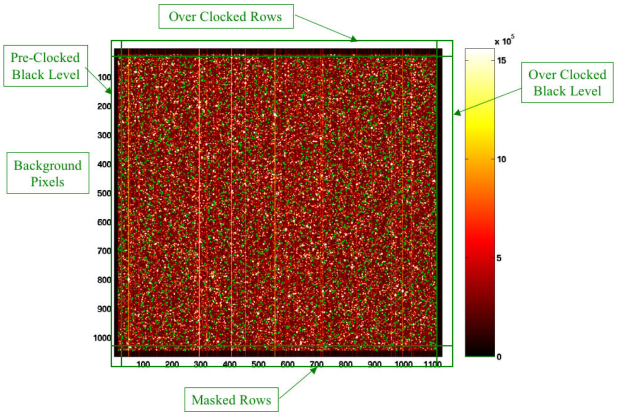

Figure 3.1: The layout of *Kepler's* science CCDs is given, with the positions of the various collateral pixel data indicated by the labels in the figure.

In order to increase the effective storage capacity of the SSR, and to reduce the time required to downlink the photometric data from *Kepler*, the baseline compression scheme performs two tasks: First, it requantizes the data to make the ratio of the quantization noise to the inherent measurement uncertainty uniform over the dynamic range of the observations. Second, it reduces temporal redundancies in the pixel time series arising from the nature of the observations. The first step reduces the size of the word required to store each pixel value from 23 bits to 16 bits, while the second step results in a further reduction to ∼4.5 bits, on average. This level of compression is achievable because the photometer is imaging the same stars on the same pixels continuously for each 90 day segment, so that most of the expected variations in the pixel values are due to the sub-pixel pointing offsets at the 15-minute-to-15-minute level. Some variations are expected due to intrinsic stellar variability of stars contained in the target star apertures (including background stars) on timescales comparable to the stellar rotation periods. For solar-like stars of greatest interest to *Kepler*, the stellar rotation periods will be $\geq$14 days so that stellar variability should not dominate the variability of the observed pixel values at the 15-minute level. Of course, we do expect that there will be target stars and background stars that exhibit strong variations on timescales less than 2 weeks, but these should represent a small fraction of the stars observed by *Kepler*.

Once every 24 hours, or suitable interval specified by the Ground Segment (GS), the RAD750 computer stores a requantized baseline image on the SSR. The baseline pixel value is subtracted from each of the next 95 requantized pixel values for each Pixel of Interest (PoI), and the residuals are entropically encoded and stored on the SSR. The baseline images allow the compression scheme to track changes due to intrinsic stellar variability (or other sources) on timescales longer than one day. For robustness, the PCE also entropically encodes the difference between successive baseline images. The presence of the baseline-to-baseline residuals in the data stream means that three distinct pieces of information must be lost in order to make it impossible to reconstruct the residual pixel values in a particular 24-hour interval: The baseline pixel, the difference from the baseline to the next baseline, and the difference from the previous baseline to the current one must all be lost.

Over the lifetime of the mission, the average fraction of data expected to be downlinked intact to the ground on the *first* pass will be $\geq$95%, with the 5% loss occurring mainly in large 'chunks' due to weather, equipment failures and DSN operator errors. The bit error rate during nominal communication will be much smaller, $\sim 10^{-5}$. Thus, for DSN passes with good links, the chance of losing all three pieces of information should be less than 10^{-12}. The SSR is capable of holding $\sim$18 days of data for 170,000 targets, so that there will be $\geq$4 chances to achieve a good link to send down the data packets lost during previous passes. Note that the 95% first pass success rate indicates a less than 1 in 10^5 chance that all four passes will fail to deliver a good enough link to successfully downlink any packet. In total, we would expect to lose 22.6 Mb out of 3.44 Tb of data over the course of the mission, or 6×10^{-6} of the entire data set. In practice, if there is more than one 'bad' pass, additional DSN resources will likely be brought to bear in order to retrieve the necessary data.

Figure 3.2 presents a flowchart for the sequence of steps required to obtain fully calibrated pixels from the bitstream downlinked by the spacecraft. The top flow represents the decompression procedure. In order to reconstruct a pixel value from the data stream, the DMC must first decode the bitstream to obtain a pixel residual, $\delta\tilde{p}$, add the corresponding baseline value, $\langle\tilde{p}\rangle$, and then map the requantized pixel value back onto the linear scale in ADUs. The reconstructed pixel values are then ready to be corrected for on-chip systematic effects. These are detailed in the following sections.

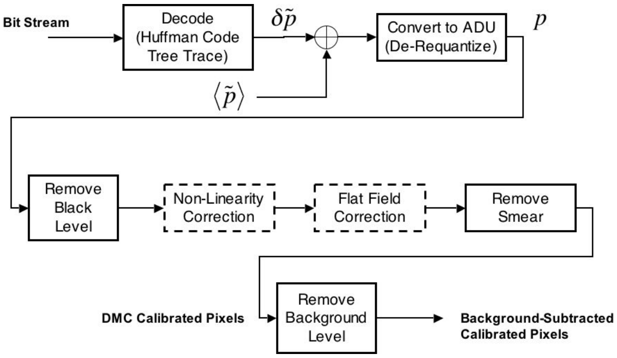

Figure 3.2: Flowchart for the sequence of steps applied to the raw data stream to obtain calibrated pixel values. The top flow shows the steps required to decompress the data, which occurs at the DMC. The middle flow shows the sequence of pixel level corrections planned to occur at the DMC, with "optional" corrections indicated by dashed borders. Whether these corrections are made will be determined by the ability to characterize the flat field and the transfer function of the instrument. The last step, estimating and removing the sky background, will occur at the SOC.

3.1 Black Level Subtraction

The CCDs in *Kepler's* focal plane are essentially analog detectors. Although they count photoelectrons, at the end of the exposure an analog voltage is reported for each pixel, and this is digitized by a 14-bit Analog to Digital Converter (ADC). The analog CCD voltage is biased ∼5% above the minimum of the ADC input voltage range to prevent clipping of low input signals. Similarly, the maximum voltage read out from a CCD is set ∼5% below the maximum of the input voltage range of the ADC to prevent clipping of high pixel signals. Thus, the full range voltage swing of the CCD signal covers ∼90% of the ADC input range.

Estimates of the black level, or the digital count value corresponding to 0 V and hence, 0 e^-, are obtained from virtual pixels read out either prior to or after a physical CCD row has been read out. The baseline design is to pre-read 12 pixels and overclock 20 pixels for each physical CCD row. No flux is accumulated in the serial register during readout. Consequently, these pixels measure the zero point of the CCD electronics chain, and can be used to estimate the 'black level' or 'bias' of the CCDs. Thus, there are a total of 32 columns that can be potentially combined together to form the black level estimate for each readout row. Not all of these are expected to be useful for estimating the black level. Pixels at the edge of each virtual segment may be corrupted by systematic effects in operating the CCDs. The exact combination of pixels to use shall be determined by characterization of the CCD electronics chains during test and integration. In the baseline design, whatever subset of pixels are chosen to form the black level estimate for each row are simply summed together prior to being stored on the SSR. Thus, the DMC will need to divide each black level value by the number of pixels that were summed together to generate it. Note that the black level estimates are not requantized prior to being encoded, since the black level itself determines the zero-point for the requantization. The PCE will need to use an estimate of the black level to implement the requantization of pixels that receive flux.

3.2 Nonlinearity Correction

Once the black level has been removed, it is appropriate to correct the pixel values for known nonlinearities in the transfer function relating photoelectrons to ADUs. Assuming that the transfer function is well-behaved (i. e., it is monotonic and smoothly varying apart from the discontinuities introduced by the ADC), it is relatively straightforward to correct the pixel values. Nonlinearities fall into two different categories for *Kepler*: those that we intentionally introduce to realize improved performance, and those that are not introduced intentionally. In both cases a characterization effort may be required in order to correct for the effect.

One source of nonlinearity is intentionally introduced into the measurements. It was dealt with as part of the decompression process. The requantization step used by the flight software to set the level of the quantization noise maps linear, digitized ADC counts onto a nonlinear scale. The requantization is in effect a non-uniform ADC in which the step sizes get larger towards the upper end of the possible input values. For example, the maximum counts reported in a 15 minute interval with 300 co-adds (corresponding to 2.5 s integration intervals with 0.5 s readout intervals) is 4,914,900 ADU. The shot noise at this level is ∼20,000 e^- or 141 ADU. If the difference between successive levels in the output of the requantizer at this level is set equal to the shot noise, then the magnitude of the quantization noise would be 29% of that of the shot noise. This is a trivial source of nonlinearity to deal with since it occurs in the digital domain and we specify it, so that mapping the reconstructed, requantized values back into ADU is a matter of knowing where the 'steps' occur in the requantization scheme.

3.3 Flat Field Correction

Another area of consideration is the multiplicative effect known as the flat field correction. This has an impact on how the error estimates are treated. The flat field effect is a result of the quantum efficiency (QE) of the CCD pixels not being uniform. Values of 0.5% are typically observed for the RMS pixel-to-pixel sensitivity of CCDs. In addition, there are often large scale variations across a CCD, and there is also the effect of the vignetting of the optics, which produces a similar effect. It is likely that of the short-scale pixel-to-pixel rms variability will be determined quantitatively during pre-flight characterization of the detectors. The large-scale QE variations, together with the details of the vignetting, may not be known at the requisite level of detail until *Kepler* is in orbit. One Full Field Image (FFI) may be sufficient to permit extraction of this information, assuming that the Stellar Classification Program (SCP) delivers sufficiently accurate stellar magnitudes transformed onto the *Kepler* instrument magnitude scale.

3.4 Smear and Dark Subtraction

The *Kepler* photometer has no shutter, so that the images will contain vertical streaks due to star light accumulating in the pixels along each column during readout. This represents a systematic error source unique to *Kepler* that must be estimated and removed. Moreover, there is no capability to take dark frames or flat frames, as is customary for ground-based photometric observations. The risk associated with a mechanical shutter is too large to justify having one, given the benign operating environment for *Kepler*. However, estimates of the average dark current per pixel and the effects of the shutterless readout-induced smear can be estimated from different measurements made with the CCDs.

A set of 20 rows at the bottom of each CCD are masked over to block out starlight. During each exposure, these pixels accumulate dark current, but do not accumulate star flux. During readout, the charge packets that are clocked into the CCD to replace the masked-over pixels accumulate starlight as they are clocked through the FOV. Thus, these masked rows measure the smear and the dark current directly. In addition, a set of 20 rows are to be clocked out following the readout of the physical CCD, much as the 12 pre-clocked and 20 over-clocked columns are generated. These 20 rows only exist during readout and accumulate the same smear flux as the physical pixels do, but accumulate dark current only while they exist (nominally 0.5 s). Assuming that the readout is 0.5 s, the integration interval is 2.5 s, that there are 1132 columns and 1064 rows read out each time, the masked pixels contain

$$b_{mask} = b_{smear} + 3\,i_{dark}, \tag{3.1}$$

where b_{mask} is the masked pixel flux accumulated in 3 s, b_{smear} is the smear flux each pixel picks up, and i_{dark} is the dark current (in $\mathrm{e^{-}s^{-1}}$). Likewise, the flux in each overclocked row, $b_{virtual\ row}$ is given by

$$b_{virtual\ row} = b_{smear} + 0.5\,i_{dark}. \tag{3.2}$$

Solving Eqs. 3.1 and 3.2 yields

$$i_{dark} = \frac{1}{2.5}\,(b_{mask} - b_{virtual\ row}), \tag{3.3}$$

and

$$b_{smear} = b_{mask} - 3\,i_{dark}. \tag{3.4}$$

In practice, the constants in Eqs. 3.1 – 3.4 are established by the detailed timing control of the CCDs, which will be documented by the Flight Segment (FS).

Still a third approach exists for estimating and correcting for smear. If saturated charge is conserved, then the smear can be estimated by summing up the flux accumulated in the physical pixels comprising each column. Properly scaled, this summed flux should be a high fidelity estimate of the smear flux, since it has so much less fractional shot noise compared to either the masked rows or the overclocked rows. The details of exactly what measurements will be used to correct for smear may not be fully worked out until data is returned on flight, but pre-flight test of the flight electronics will provide useful information to constrain the possible solutions. We note that by operating at -95°C, there will be virtually no dark current, so that correcting for the dark current should not be a concern. These pixel-level corrections will be performed at the DMC at the Space Telescope Science Institute (STScI) located in Baltimore, Maryland.

3.5 Background Flux Removal

There are two major sources of background flux in the FOV: zodiacal light and dim background stars.

The zodiacal light is solar flux that is scattered from dust grains in and above the ecliptic plane into the Photometer's aperture. Beyond a certain magnitude, every pixel will contain at least one dim star, and the dimmer the star, the denser their concentration. At this point, the flux from these stars is so diffuse as to present a smoothly varying background as individuals cannot be detected in the actual images. The background flux from these sources will be estimated in each CCD output by monitoring 4500 dim pixels throughout the image. It is likely that a low order two dimensional polynomial surface will be fit to the pixel measurements and then subtracted from each target star pixel.

Chapter 4

Extracting Photometric Light Curves

This chapter describes the steps necessary to transform calibrated pixel measurements into detrended, normalized, relative light curves.

4.1 Optimal Pixel Weighting: Forming Photometric Apertures

A major task for the SOC is to determine the photometric aperture to be used for generating the calibrated light curve for each target star. This is distinct from the task of choosing an aperture for the purposes of selecting which pixels are returned to the ground. Each of these tasks will be treated in turn.

Motions of stellar images over a finite photometric aperture cause apparent brightness changes (even with no intra- or inter-pixel sensitivity variations). The wings of any realistic PSF cause these motion-induced brightness variations, as they extend outside of any reasonable photometric aperture. Pixel-to-pixel variations generally exacerbate motion-induced brightness variations as well as causing apparent changes in the PSF. In addition, changes in platescale and focus also induce apparent brightness changes in measured stellar fluxes. Figure 2 of Koch et al. (70) presents an example from the Kepler Testbed. Several possible remedies to these problems exist: 1) Calibrate the response of each star's measured brightness as a function of position and focus and use this information to correct the measured pixel fluxes. 2) Regress the lightcurves against the measured motion and focus or other correlated quantity to remove these effects. 3) Calculate the stellar fluxes using weighted sums of the aperture pixels in such a way as to reduce the sensitivity to the typical image motion and focus changes.

The first solution requires detailed knowledge of the 3-D correction function for each star, and must be applied on timescales short enough so that the change in position and focus is small compared to the full range of motion and focus change. This solution is equivalent to PSF-fitting photometry. For the Kepler Mission, the attitude control system operates on timescales much shorter than 15 minutes, so that the motion becomes decorrelated after about 25 seconds. Long term components of focus and platescale change will occur due to the apparent 1 degree per day rotation of the sun about the spacecraft and differential velocity aberration. Changes on timescales this long can be neglected for the purposes of transit photometry, so long as the amplitudes are not large enough to move the target stars by significant fractions of a pixel. The short coherence time of the spacecraft jitter would necessitate the application of the flux corrections after one or several readouts, which is impractical.

The second solution has been previously demonstrated in obtaining 10^{-5} photometry for front-illuminated CCDs (100) and for back-illuminated CCDs (61). Our modification of this method is presented in Section 4.6.

In contrast to the first approach, the third solution is feasible if the image motion and focus changes are approximately wide-sense stationary random processes (i.e. the statistical distributions of changes in position and focus are constant in time (50)). Strict wide-sense stationarity is not required, however, it simplifies the implementation of the method, as updates in the pixel weights would not be required in between spacecraft rotations (which occur every 3 months). What remains is the problem of designing the pixel weights themselves. Section 4.2 follows the derivation of a formula for obtaining the optimal pixel weights, and gives examples of their effectiveness in reducing sensitivity to motion.

4.2 Theoretical Development

We wish to derive an expression for the optimal pixel weights minimizing the combination of sensitivity of the flux of a star to image change and the effects of the weights on shot noise and background noises. This approach is motivated by a signal processing perspective in which aperture photometry is viewed as applying a finite-impulse response (FIR) filter to a temporally-varying 2-D waveform. In the 1-D signal-processing analog, the desire is to shape the frequency response of a FIR filter to reduce or enhance the sensitivity of the filter to energy in certain wavelength regions. In the problem at hand, the desire is to use the free parameters available (the pixel weights) to minimize the response in the flux measurement to image motion and PSF changes. The following assumptions are made: 1) the PSF and its response to image motion and focus change are well-characterized, 2) the distribution of background stars is known, and 3) the background noise sources are well-characterized. Consider a set of N images of a single target star and nearby background stars consisting of M pixels ordered typographically (i.e. numbered arbitrarily from 1 to M). Assume that the range of motion and focus change over the data set are representative of the full range of motion and focus changes. Define the error function, E, as the combination of the mean fractional variance between the pixel-weighted flux and the mean, unweighted flux and a second term accounting for the effect of the pixel weights on the shot and background noise:

$$E \equiv \frac{1}{N}\frac{1}{\overline{B}^2}\sum_{n=1}^{N}\left(\overline{B}-\sum_{j=1}^{M} w_j b_{n,j}\right)^2 + \frac{\lambda}{\overline{B}^2}\sum_{j=1}^{M} w_j^2\left(\overline{b}_j+\sigma_j^2\right) \tag{4.1}$$

where $b_{n,j}$ is the j^{th} pixel value at timestep $n, n = 1,\ldots,N; j = 1,\ldots,M$; w_j is the weight for pixel $j, j = 1,\ldots,N$; $\overline{b}_j$ is the mean pixel value for pixel j; $\overline{B}$ is the mean flux with all weights set to 1; σ_j^2 is the background noise variance for pixel j; and all quantities are expressed in e^-. Here we take the shot noise to be due entirely to the star itself, and the background noise to be a zero-mean process which includes such noise sources as readout noise and dark current. This implies that the images have been corrected for all non-zero-mean noise sources such as dark current and the readout smear flux. We further assume that the background noise sources are uncorrelated from pixel to pixel. If this is not the case, the second term of 4.1 can be augmented to account for the correlation. The scalar $\lambda \in [0,\infty)$ determines the balance between the desire to minimize the difference between the flux estimate and the mean flux value, and the desire to minimize the accompanying shot noise and the background noise. For this situation, we would normally set $\lambda = 1$.

The error function in Eq. 4.1 is quadratic, and therefore admits a closed-form solution in matrix form:

$$\mathbf{w} = [\frac{\mathbf{1}}{\mathrm{N}}\mathbf{B}^{\mathbf{T}}\cdot\mathbf{B}+\lambda\mathbf{D}]^{\mathbf{-1}}\cdot\overline{\mathbf{b}}\,\overline{\mathrm{B}}, \tag{4.2}$$

where

$$\begin{aligned}\mathbf{B} &\equiv \{b_{n,j}\}, n = 1,\ldots,N; j = 1,\ldots,M \\ \mathbf{D} &\equiv \{D_{i,j}\} = \overline{b}_i+\sigma_i^2, i = j = 1,\ldots,M \qquad (4.3) \\ \mathbf{b} &\equiv \{\overline{b}_j\}, j = 1,\ldots,M.\end{aligned}$$

Throughout this paper, boldface symbols represent column vector or matrix quantities. For real data with noise-corrupted images, the scalar λ should be adjusted to prevent over-fitting. If enough data is available, λ will be essentially 0.

An alternative iterative scheme can be used that is based on the popular NLMS (normalized least mean square error) algorithm for adaptive filtering (50). The chief advantage of such an algorithm is that the pixel weights can be designed 'in place,' and can be updated as necessary. This algorithm adjusts the pixel weight vector by an incremental vector opposite the direction of an estimate of the gradient of the error function. Taking the expression

$$\begin{aligned}\hat{E(n)} &= (\overline{B}-\mathbf{b_n^T}\cdot\mathbf{w_n})^2 && (4.4)\\ &= [\overline{B}-(\overline{\mathbf{b}}+\Delta\mathbf{b_n})^{\mathbf{T}}\cdot\mathbf{w_n}]^2 && (4.5)\\ &= (\overline{B}-\overline{\mathbf{b}}^{\mathbf{T}}\cdot\mathbf{w_n}-\Delta\mathbf{b_n^T}\cdot\mathbf{w_n})^2 && (4.6)\\ &= (\Delta\mathbf{b_n^T}\cdot\mathbf{w_n})^2 && (4.7)\end{aligned}$$

as the error estimate at time n, where $\Delta\mathbf{b_n}$ is the difference between the average pixel fluxes and those at the n^{th} time step, the update to the weight vector at time step n is given by

$$\begin{aligned}\mathbf{w_n} &= \mathbf{w_{n-1}} \quad -\mu\frac{\nabla E(n)}{\Delta\mathbf{b_n^T}\cdot\Delta\mathbf{b_n}+\epsilon} \\ &= \mathbf{w_{n-1}} \quad -\mu\frac{\mathbf{b_n^T}\cdot\mathbf{w_{n-1}}}{\Delta\mathbf{b_n^T}\cdot\Delta\mathbf{b_n}+\epsilon}\Delta\mathbf{b_n^T},\end{aligned} \tag{4.8}$$

where μ is a positive scalar that controls the rate of adaptation (and convergence in the case of stationary noise and motion) and ϵ is a small positive number to ensure stability in cases where the instantaneous error is 0. Note that the term for shot and background noise does not appear here. It is not necessary as μ can be adjusted to prevent over-fitting the noise, and the algorithm is mainly of interest in the case of noise-corrupted images.

In terms of considering implementation on the *Kepler* spacecraft, Equation 4.2 may be preferred, as it reduces the computations required. This approach would require the collection of adequate samples of star pixels to recover well-sampled (super-resolution) scenes for each target star. This might be avoided with proper calibration of the optics and CCDs along with a high-resolution catalog of stars in the FOV. If necessary, the adaptive scheme of (57) could be implemented, with proper choice of scheduling and for the value of μ to insure that the adaptation takes place over time scales significantly longer than a transit.

4.3 A One-Dimensional Example

In this section we provide a 1-D example to examine various properties of optimal pixel weighting. Figure 4.1a shows the average 'image' of a Gaussian with a full width half max (FWHM) of 4 pixels on an 11 pixel aperture, nominally centered at -0.2 pixels. The integration time corresponded to 3 minutes for a 12th magnitude star for the Kepler photometer, yielding an average flux of $5.5x10^7$ e^- at each timestep. This PSF was moved over a 101-point grid in space of ±0.5 pixels from its nominal location at -0.2 pixels and integrated over each pixel in the aperture to form a set of images. Figure 4.2b illustrates the response of the flux signal to image motion over the data set for optimal pixel weights corresponding to various signal-to-noise ratios (SNRs) and for unweighted aperture photometry. Here, SNR is defined as the ratio of the mean flux to the root sum

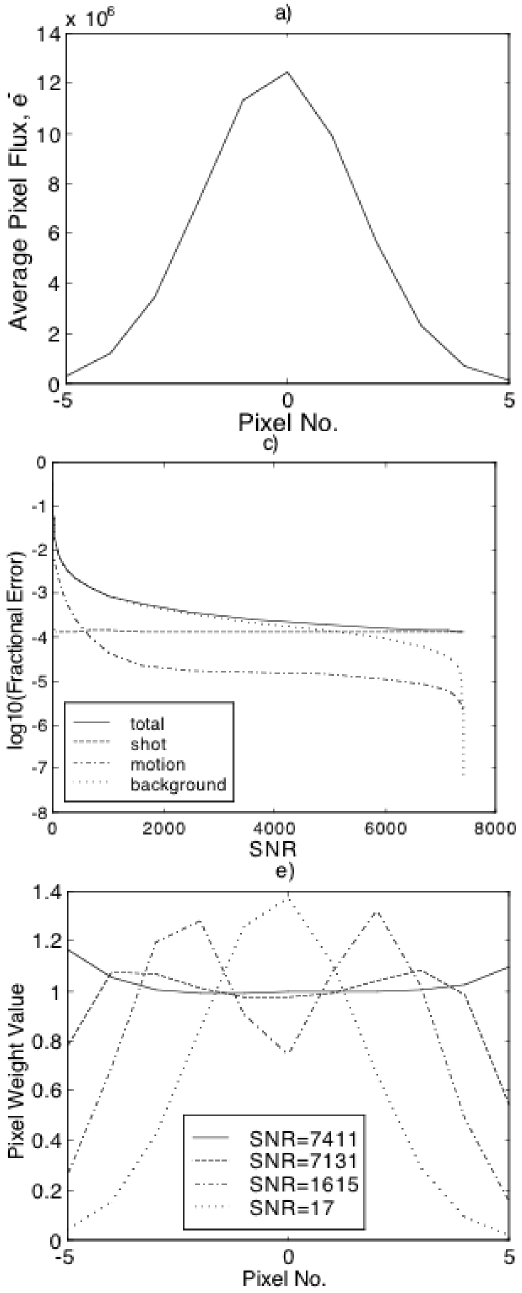

Figure 4.1: Slides a, c, and e

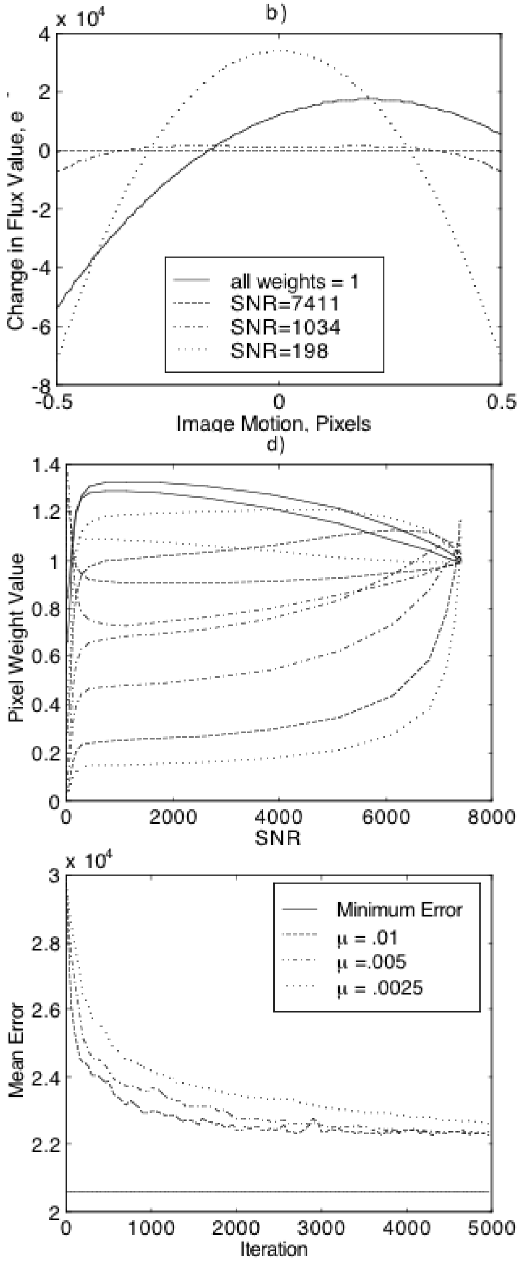

Figure 4.2: Slides b, d, and f

square (RSS) combination of shot noise and background noise. Note that the full range of brightness variations is 1.3% for unweighted pixels, and that this is reduced to $1x10^{-5}$ at an SNR of 7,411. At low SNRs, the background noise dominates, and the pixel weights adjust to minimize the increased noise due to background, rather than to motion. However, the response to motion is made symmetric by the pixel weights even at low SNR so that the motion will more easily average out over timescales much longer than the coherence scale of the motion. Figure 4.1c presents the total expected fractional error and its three components, shot noise, motion error, and background noise, as functions of SNR. The pixel weights confine the motion error to well below the unweighted case over the range of SNRs presented here. Figure 4.2d shows the evolution of the optimal pixel weights as a function of SNR, while Figure 4.1e shows the profiles of the pixel weights at four different SNR values. As the SNR deteriorates, the profile of the optimal pixel weights looks more and more like the original star profile. The final Figure, 4.2f, illustrates the application of 4.9 to an online adaptive solution for the pixel weights. A total of 5,000 images along with shot noise and background noise of 6,310 e^{-}/pix were presented to the algorithm, which was initialized with all weights equal to 1. The pixel weights converge after a few thousand iterations, corresponding to a few days of adaptation. Better initialization would result in faster convergence. We note that the excess error, or misadjustment of the weights is rather small, about 10% of the theoretical minimum error. Once convergence is achieved, the adaptation rate can be reduced so that the algorithm tracks changes in the mean image position and PSF shape over timescales much longer than transits, preserving any transits in the resulting flux time series.

4.4 Selecting Pixel Masks

As in conventional differential aperture photometry, optimal pixel weighting benefits from pre-masking of the pixels containing each star image to consider only those pixels with significant stellar flux content. The advantages are two-fold. First, design of optimal pixel weights for dim pixels from actual images is problematic whether the weights are generated using Equation 4.2, or whether an adaptive algorithm is applied online. A great deal of data is required to reduce the uncertainties in the corresponding pixel weights to acceptable levels. Second, reducing the number of pixels for which weights are sought reduces the amount of data required for the pixel weight design. Various schemes for identifying the photometric aperture diameter and shape appear in the literature (57) and (30). Here we present the method applied to the laboratory data to identify pixels allowed to participate in flux measurement for each star. The pixels are first listed in order of descending brightness. Next the cumulative SNR for the pixel list is calculated according to:

$$SNR(i) = \frac{\sum_{j=1}^{i} \overline{b}_j}{\sqrt{\sum_{j=1}^{i} \overline{b}_j + i\sigma^2}}, i = 1, \ldots, M, \qquad (4.9)$$

where all units are in e^-. The function $SNR(i)$ will increase as more pixels are added until the point at which the pixels are so dim as to detract from the information content of the flux estimate. All pixels beyond the point at which the maximum is attained are masked out. Figure 4.3 shows the cumulative SNR for star S12d, a 12^{th} magnitude smear star in the laboratory demonstration. $SNR(i)$ peaks at 32 pixels. Figure 4.4 shows the pixels selected for inclusion in the flux estimate for this star in white, while masked-out pixels are in black. The border of the mask is roughly circular, as expected.

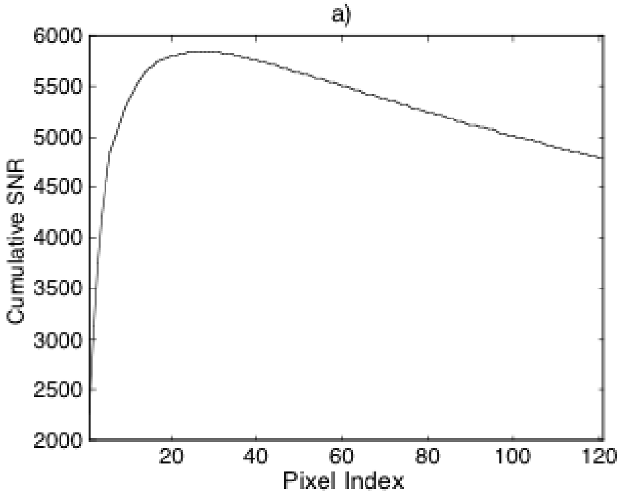

Figure 4.3: Cumulative SNR for star S12d shows that only 32 pixels contribute meaningful information about the star's flux.

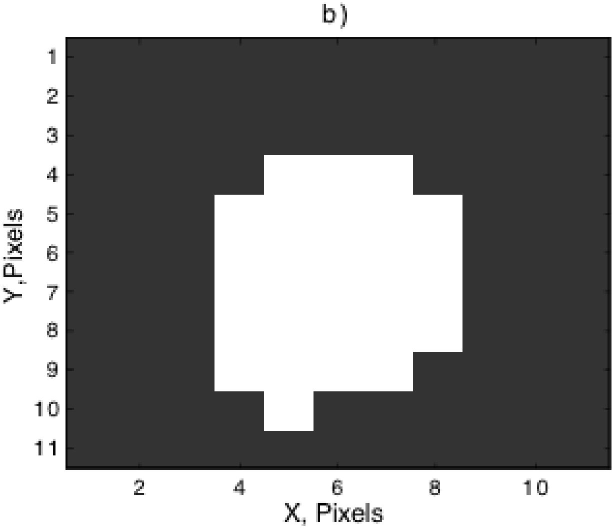

Figure 4.4: The pixels used for flux calculations for this star are shown in white.

4.5 Ensemble Photometry and Common-Mode Noise Rejection

Ground-based photometry suffers from having to correct large brightness changes that occur over various time scales. Time varying extinction is the largest of these, resulting in 10–20% changes in star brightness over the course of a night. Night–to–night variations in atmospheric transparency cause longer term errors in star brightness measurements. While Kepler will not have these problems, there are other common-mode errors that can affect photometric accuracy, for example, temperature related gain changes. These and other multiplicative common-mode errors can largely be compensated by dividing out an appropriate mean signal from the measurement of an individual star. The mean signal appropriate for a given target star is determined by constructing an ensemble of stars whose response to the error inducing process is similar to that of the target. For example, in ground-based observations an ensemble is usually constructed from stars of a similar color to the target because they are affected in the same way by wavelength dependent extinction or transparency changes.

Kepler photometry is not expected to suffer from large changes in sensitivity or gain; however, if changes are seen in the data, ensemble photometry may be used to mitigate them. The selection of an ensemble will depend in part on the cause of the error signal. Because each CCD output is processed through a separate amplifier, the largest pool for an ensemble would be the set of stars on the same output amplifier. In the event of thermally induced gain changes caused by the amplifier heating up during readout, the ensemble would be chosen from stars in a region around the same row as the target. In this case, the ensemble might be chosen from stars whose raw flux time series is highly correlated with that of the target star. Highly variable stars are excluded from the ensemble. Variables can either be known in advance, for example known eclipsing binaries, or detected during photometry. To detect new variables, we first perform ensemble photometry and then select out stars whose relative flux time series varies more than some predefined threshold. The ensemble normalization is then redone excluding these stars. In the absence of large gain or sensitivity changes, the method to remove systematic errors described below will offer superior performance.

4.6 Removing Systematic Noise

This section describes the approach to be used to estimate and remove residual systematic errors. The one systematic error that is expected to be of some concern is pointing errors on time scales longer than the photometry sampling period. These can cause photometric variations that are highly correlated from star to star, although they are not a common mode noise term. The preferred method for removing systematic noise that is highly correlated across the target stars is to apply singular value decomposition (SVD) analysis to direct measurements of the suspected systematic noise sources, such as focal plane temperatures and photometer pointing offsets. This results in a set of vectors that best explain the signatures of the various systematic noise sources. Once these SVD components are removed from the lightcurves, an SVD analysis shall be conducted for the target stars in each channel to determine if other systematic errors might be present. If so, then there will be a small set of SVD vectors with large singular values that are not correlated with any of the previously identified systematic sources. An attempt should be made to correlate these residual SVD vectors with existing ancillary data measurements other than those representing the known systematic sources. If additional systematics can be identified with direct measurements, then these should be added to the list of known systematics and the first step of the process should be repeated with the new set of systematic sources. If no additional systematics can be identified in the ancillary data, then the residual SVD components should be removed from the stellar lightcurves prior to attempting to detect planetary signatures.

To develop this idea further, the singular value decomposition of a real $m \times n$ matrix A is the factorization,

$$A = U\Sigma V^{'}. \tag{4.10}$$

The matrices in this factorization have the following properties: $U_{m\times m}$ and $V_{n\times n}$ are orthogonal matrices. The columns u_i of $U = [u_1, ..., u_m]$ are the *left singular vectors*, u_k, and form an orthonormal basis, so that $u_i \cdot u_j = 1$ for $i = j$, and $u_i \cdot u_j = 0$ otherwise. The rows of $V' = [v_1, \dots, v_n]$ contain the elements of the *right singular vectors*, v_k, and form an orthonormal basis. $\Sigma_{m\times n}$ is a diagonal matrix with entries $(\sigma_1, \dots, \sigma_n)$ is a real, nonnegative, and diagonal matrix. Its diagonal contains the so called *singular values* σ_i, where $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$.

The singular vectors form orthonormal bases, and the important relation $A \cdot v_i = \sigma_i \cdot u_i$ shows that each right singular vectors is mapped onto the corresponding left singular vector, and the "magnification factor" is the corresponding singular value.

Every $m \times n$ matrix has a singular value decomposition. The sum of singular values of A equals the Frobenius norm of a matrix A, defined as the square root of the sum of the squares of all its entries and in this sense represents energy.

Any real matrix $A_{m\times n}$ (where m > n) can be written as the sum of n rank-one matrices. A low rank approximation of A defined as $A_{low\,rank} = \sum_{j=1}^{j=r} \sigma_j u_j v_j^T$ where $r < n$ captures as much of the energy (in the 2-norm or Frobenius norm sense) of A as possible and expresses matrix A as a sum of r rank-one matrices of decreasing importance, as measured by the singular values. On the other hand, $A_{low\,rank} = \sum_{j=r}^{j=n} \sigma_j u_j v_j^T$ where $1 < r < n$ ignores the dominant component(s) contributing to the energy while retaining subtler variations. This is the basis for applying SVD to a matrix containing light curves and creating a low rank approximation which retains only the residual flux variations that are unique to each stars including the transits.

By visual inspection of the singular values, one may identify large singular values that should be discarded to reduce the effects of trends in data caused by systematic variations.

4.6.1 Simulation Study

To demonstrate the potential of SVD to eliminating correlated noise from a grouping of star flux time series values, a simple simulation was developed. Star flux data from the photometer was created as the sum of shot noise, poisson noise, star field flux, flux decrease due to planetary transits, and the trend introduced by long term systematic variations. Everything beyond the flux mean, the flux random noise, and the transit offset is considered to be correlated signals that SVD should be able to eliminate from the individual light curves if operated on collectively.

Follows is a description of the data generated in this simulation.

1. Each of 50 star field mean flux variations is drawn from a uniform distribution varying between 0 and 10^7.

2. Shot noise is modeled as as a gaussian with a zero mean and a standard deviation 10^3.

3. Poisson Noise is modeled as gaussian with a zero mean and a standard deviation equal to the square root of the mean star flux.

4. Two trends are introduced: (1) a nonlinear trend, one cycle of sinusoidal variation over the entire observation period with a peak magnitude 10^4, and (2) a minuscule trend, proportional to the flux mean, increasing over the first half of the observation period and decreasing over the second half.

5. Transits corresponding to a dimming of 2% of star flux values and with a duration of 5 hours (equivalent to 20 observations) are superimposed on light curves.

A single example of these curves is shown in Figure 4.5. Star data A is a matrix of size mxn where m is time steps and n is number of stars.

Given these 50 light curves, SVD is then applied. Figures 4.6 through 4.10 show what SVD extracts as the first five singular values. Note that these represent common variation in the data in descending oder of magnitude. The relative magnitude of the singular values are shown in Figure 4.12.

After singular value decomposition, a residual matrix is formed as

$$A_{low\,rank} = \sum_{j=r}^{j=n} \sigma_j u_j v_j^T \tag{4.11}$$

where $r < n$. The significant (and common to all light curves) singular values (here taken to be four) are removed from the data and the resulting light curves are reconstructed. The result is what can be thought of as the individual star's light curve without instrumental variation. An example of this is shown in Figure 4.11. Note how easy it is to visually identify the transit when compared to the original data in Figure 4.5.

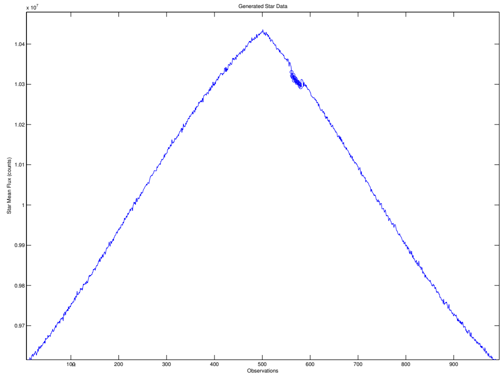

Figure 4.5: Generated Star Data includes a mean flux value, a sinusoidal component, a drifting trend, random noise (shot and Poisson), and a transiting event.

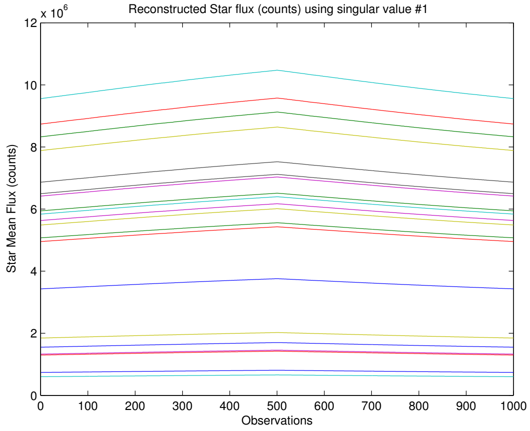

Figure 4.6: Reconstructed Star Flux (in counts) using only the first singular value.

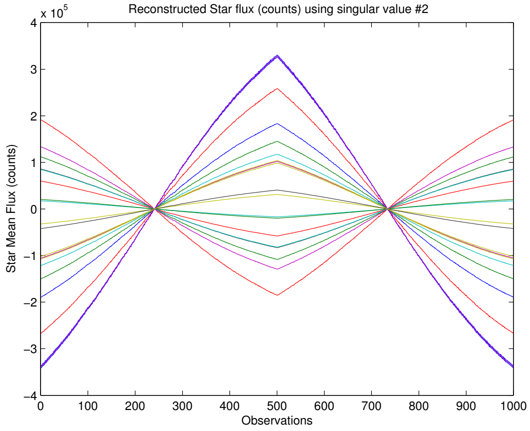

Figure 4.7: Reconstructed Star Flux (in counts) using singular value #2.

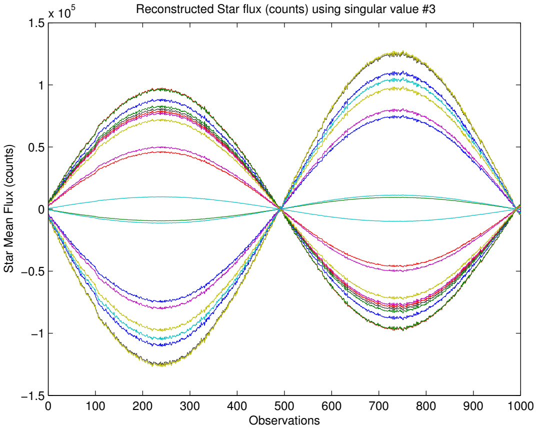

Figure 4.8: Reconstructed Star Flux (in counts) using singular value #3.

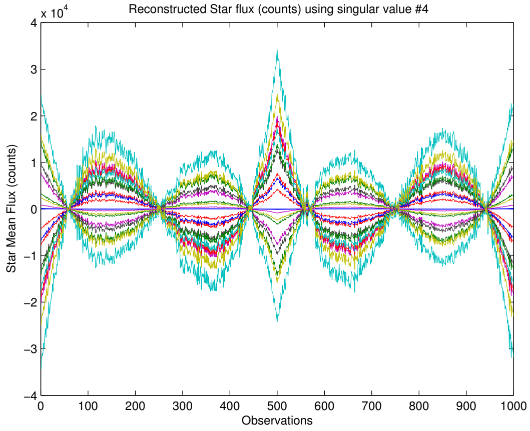

Figure 4.9: Reconstructed Star Flux (in counts) using singular value #4.

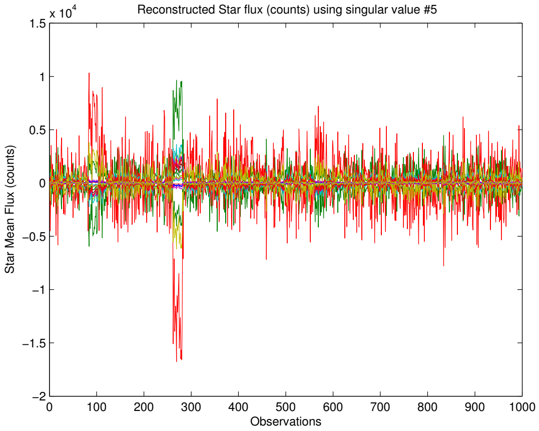

Figure 4.10: Reconstructed Star Flux (in counts) using singular value #5. Notice that most of what appears to be modeled is random fluctuations about the mean and possibly one or two transits. At this point, removing the singular value from the data set would hurt the transit detection ability.

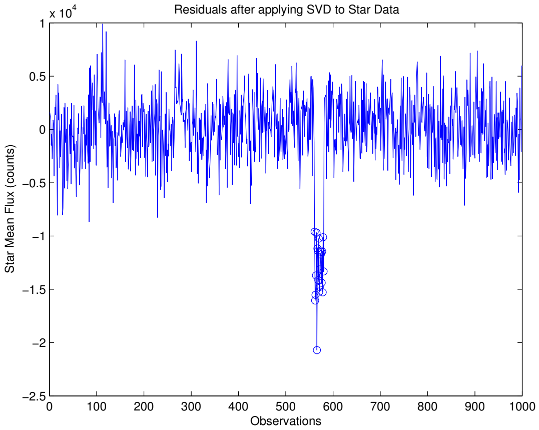

Figure 4.11: Residuals after applying SVD to the original Star Data.

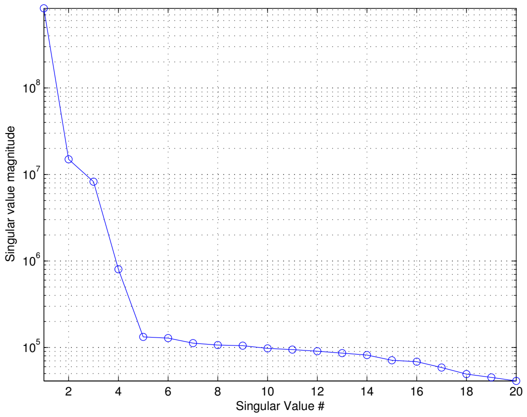

Figure 4.12: The relative sizes of singular values. Notice that there is a knee in the curve between the fourth and fifth singular values. Only singular values to the left of this knee should be eliminated from the data.

Chapter 5

Difference Image Analysis

Difference image analysis has been shown to provide excellent performance for stellar time-series photometry in a number of contexts over the past five years. Ground-based data in very crowded stellar fields, collected for the purpose of detecting microlensing events, is now routinely analyzed with difference image analysis. Data acquired with the CCD cameras on HST in crowded stellar fields has provided best results using difference image analysis, for both faint and bright star applications, whether working in limiting cases of low signal-to-noise on detection of ancient supernovae, or in high signal-to-noise applications for the detection of planets via transits.

For the *Kepler* project, the use of difference image analysis also enables checks on the positional coincidence of differential transit signatures in comparison with the direct image as a means of eliminating a significant fraction of false positives arising from background, diluted eclipsing binaries. Hot pixel development within stellar apertures can best be tracked using difference image analysis. Difference image analysis may provide a competitive basis for the generation of extracted time series from the image level data for *Kepler*.

5.1 Introduction

The state-of-the-art crowded field, time-series photometry involves creation of difference images (e.g., Alcock et al 1999; Alard 1999), where for well-sampled, ground-based CCD data excellent gains over classical point spread function (PSF) fitting in direct images are realized. With good difference images non-variable objects are removed (except for residual, unavoidable Poisson noise), leaving any variables clearly present as isolated (positive or negative) PSFs even if the variable was badly blended with brighter stars in the direct images. Extraction of precise relative photometry changes for any star in a difference image can be handled with either aperture photometry or PSF fitting, and precise knowledge of the PSF is much less critical for the difference image analyses relative to attempting photometry on blended stars in the direct image via point spread function fitting.

As one might infer from the name, difference image analysis (DIA) involves the creation of individual images that are the simple difference between an observed image and a model, or appropriate time-averaged mean image. In DIA the primary challenge is the creation of a difference image for each image in a sequence, that for non-variable sources, results only in Poisson noise from the photon sources, plus any instrument noise associated with the data acquisition. Assuming the model image represents a good time-averaged mean, DIAs at the position of a variable source will show appropriate noise plus the differential image signature of the temporal variability. Creating good difference images requires that precise knowledge exists for registration changes frame-to-frame, as well as any changes in the point spread function with time. The primary effort in DIA is in determining and accounting for these changes. For ground-based data, variable seeing, coupled with differential atmospheric refraction, and changing color terms in extinction are the primary challenges that must be dealt with in creating good difference images. For space-based data, some of the ground-based complications go away, leaving registration and any changes in the PSF from the instrument as the primary challenges. Ground-based applications almost always start with data that is well sampled, i.e., there are more than two pixels spanning the full-width-half-maximum (FWHM) of PSFs, even for the best seeing images. For space-based data an additional challenge arises, since tradeoffs between field of view and pixel scale on the sky typically result in under-sampled data with sharp PSFs.

In ground-based applications it is typical to select several images acquired in times of best seeing, and low extinction, and then average these together, after interpolation to a common registration, to form a model image. Difference images are then formed by interpolating the mean image to the (field-dependent) position of individual images (or interpolating the individual images, with well-sampled data, either is fine), and convolving with a differential seeing kernal to match the PSFs of the model to the individual images, and then forming image differences as a simple pixel-by-pixel difference. For the ground-based data, scaling the individual images for extinction changes, including color terms, must of course be included.

For the creation of good difference images in space-based applications, the essential starting point for DIA-based work, can be expected to require careful attention to creation of an optimal reference image, and internally consistent knowledge of registration and PSF adjustments required to match the reference image to individual frames.

In an ideal experiment, where the guiding is perfect, and there are no changes of focus, or other sources of PSF changes, the DIA would be very simple. One would simply form a reference image as the mean over all available images, then for each individual image the difference image would be formed for each pixel by subtracting the reference.

Up to this point in the introduction, only creation of the difference images has been addressed, and this is intentional. Once the full process has been executed to arrive at excellent difference images the remaining steps by comparison are minor in practice. DIA is all about creating good difference images, although this document will include discussion of the remaining steps: using the difference images for relative photometry, and using the difference images for false positive rejection, and hot pixel detection.

The primary discussion here follows from analysis of four separate programs with HST; a brief synopsis of these are included here as a means of introducing publications with a significant technical description content. Gilliland *et al* (1995) provides a summary of analyses applied to a 40 hour time-series of near-UV observations of the core of 47 Tuc with the original WF/PC with the purpose of detecting δ Scuti oscillations in the Blue Straggler population. The primary tools underlying DIA (precise registration, building up an over-sampled mean image, and using these as integral parts of cosmic ray elimination) were developed and used for these analyses, although this terminology was not used at the time. Gilliland, Nugent, and Phillips (1999) applied DIA in comparing two epochs of HST observations of the Hubble Deep Field. The difference image used in this case was primarily just for comparing two epochs of averaged data, and thus illustrates the power of the technique in a simple application. (The Type Ia Supernova, 1997ff, detected with DIA at 27th magnitude remains the highest redshift object of its class and has contributed to fundamental advances in cosmology.) The data analysis discussion includes details of developing registration information, model image creation and hot pixel tracking that will remain directly relevant to *Kepler* applications. Gilliland *et al* (2000) discuss results for an HST-based search for extrasolar giant planets for which 34,000 stars in the globular cluster 47 Tuc were followed for 8.3 days with resulting precisions sufficient for detection of ‘Hot Jupiters’. The primary technical advance required in this case involved the need for PSF matching between the over-sampled model and individual frames. The last HST program for which DIA is being applied is from observations in February 2004 in which $>$ 100,000 stars in the galactic bulge (Kailash Sahu, PI) were monitored for 7 days, again to search for short-period, gas-giant planets. The DIA for these data are being conducted in parallel with drafting this report.

The remainder of this chapter will be as follows. Section 5.2 will detail the steps that are required to form internally consistent registration information, development of an over-sampled mean image, and use of the latter in forming difference images. Section 5.3 will discuss the complication of changing PSFs with time in creating the difference images. Section 5.4 will provide a description of options for deriving relative time-series photometry from a series of difference images. Section 5.5 will discuss application of the same difference images for false positive eliminations. Section 5.6 will discuss detection and tracking of hot pixels using difference images. Section 5.7 includes thoughts on unique complications, likely with the *Kepler* data and options for analyses.

5.2 Mean Image Creation and Registration

For the purposes of this section we assume that the PSF is perfectly stable in time, or at least we will ignore any variations of the PSF over the time interval of interest. What attributes are required of a 'mean image' for the purposes DIA? In the absence of image motion frame-to-frame, the mean image at each pixel would simply be the the sum divided by the number of frames. Then difference images would be formed by subtracting this mean from each individual frame. There will be motion between individual images, so we require that the mean image be formed in such a way that it can be evaluated at the position of any individual image. Essentially, the mean image in this context is intended to encapsulate information about what any individual frame would be at an arbitrary guiding position. Consider a single pixel of interest, that happens to be located just off the core where the intensity changes rapidly as a function of x, y offsets. Over a time interval of a few weeks appropriate for executing DIA, the pointing is assumed to provide a jitter ball in which the individual pointings define a more-or-less Gaussian distribution with a width that is small compared to an individual pixel scale. We now want the 'mean image' to capture the information about how the intensity changes detected by this pixel as a function of position offset. There are at least two ways to do this. One would be to form an over-sampled mean image at say a factor of 4 sub-pixel resolution, perhaps by averaging together somehow the set of individual pointings from the ensemble that fall closest to the 4×4 sub-pixel points to be sampled. If we had pointing errors (dithers) that spanned a full pixel, this would result in a nicely over-sampled mean image of the stellar scene. In the case of *Kepler*, with the pointing errors much smaller than a pixel scale the size of the 4×4 sub-pixel scale would be chosen to just span the realized dithering. The second approach is to define a mean image in such a way that the intensity response of a given pixel is captured as the terms for a function which best fits in a least-squares sense the surface $I(x, y)$, where I is the intensity, or number of counts expected per unit time for the pixel. Having developed the mean image in terms of this surface fit, difference images would then be formed by taking any individual image, evaluating its specific x, y offset within the ensemble of pointings, then evaluating the surface fit and subtracting, thus forming the difference for the target pixel. A difference image is the same operation repeated for all of the pixels in the image.

Following the discussion in Gilliland *et al* (1995) the surface fit representation at each pixel can be shown as:

$$I_{i,j}(t) = f(\delta x_{i,j}(t), \delta y_{i,j}(t)) \tag{5.1}$$

where t carries an implied mapping from $n = 1$ to N separate exposures to be analyzed. In practice I have set up the function f as a bi-cubic polynomial with the following basis terms (separately formed at each

i, j):

$$\begin{aligned}
p_{1,n} &= \delta x_n \\
p_{2,n} &= \delta y_n \\
p_{3,n} &= 1.5\delta x_n^2 - 0.5 \\
p_{4,n} &= 1.5\delta y_n^2 - 0.5 \\
p_{5,n} &= \delta x_n \delta y_n \\
p_{6,n} &= (2.5\delta x_n^2 - 1.5)\delta x_n \\
p_{7,n} &= (2.5\delta y_n^2 - 1.5)\delta y_n \\
p_{8,n} &= (1.5\delta x_n^2 - 0.5)\delta y_n \\
p_{9,n} &= (1.5\delta y_n^2 - 0.5)\delta x_n
\end{aligned} \tag{5.2}$$

The surface fit at each i, j pixel then is solved for as a least-squares solution for the coefficients a_0, a_m, $m = 1, \ldots, 9$, such that the weighted difference:

$$\chi^2 = \sum_{n=1}^{N} 1/\sigma_n^2 [I_n - (a_0 + \sum_{m=1}^{9} a_m p_{m,n})]^2 \tag{5.3}$$

is minimized. I reach a solution for the a_m using a multiple linear regression code (REGRES) from Bevington (1969) where the σ_n^2 factor is taken simply as Poisson noise (object plus sky) and detector readout noise

$$\sigma_n^2 = I_n + RO^2 \tag{5.4}$$

The solution for the surface fit is performed iteratively with the elimination of cosmic rays as points deviating by more than 3 – 4σ from the fit (see Gilliland *et al* 1995 for detailed comments on this step given under-sampled data). An intermediate data product consisting of either the data, $I_{i,j}$, or this value replaced by the surface fit (model) expectation $f(\delta x_{i,j}, \delta y_{i,j})$ is saved after the solution for all pixels.

As outlined above, each pixel of interest has it's own surface fit, $I(x,y)$ developed independent of other pixels. Input to the process of developing this fit is knowledge of the x, y offsets for all images in the stack to be analyzed. This requires that in a first attempt to develop the surface fits a reasonably accurate set of x, y offsets are available either from the pointing control system, or from separate analysis of the data, and these deltas apply reasonably well to the entire image. Once an initial set of surface fits are available, the registrations of individual frames can be improved in an iterative sense via direct use of the surface fits. For a registration model consisting of x, y zero point offsets, and small rotation and plate scale terms the registration can be iteratively improved via a least squares solution at each image for the coefficients (zero points: x_0, y_0, plate scales: psc_x, psc_y [deviations from unity], and rotation term: rot) that minimize:

$$\chi^2 = 1/\sigma_{i,j}^2 [I_{i,j} - f(\delta x_{i,j}, \delta y_{i,j})]^2 \tag{5.5}$$

where $f(\delta x_{i,j}, \delta y_{i,j})$ is the Legendre polynomial surface fit at each pixel, and

$$\delta x_{i,j} = x_0 + (x_{i,j} - x_c)psc_x + (y_{i,j} - y_c)rot \tag{5.6}$$

$$\delta y_{i,j} = y_0 + (y_{i,j} - y_c)psc_y - (x_{i,j} - x_c)rot \tag{5.7}$$

where x_c, y_c are simply the mid-points of the x, y ranges respectively. The solution for improved registrations, and improved surface fits is cycled through a few times and has always converged well when given a good starting point.

Once the iteration cycle on surface fits at each pixel, coupled with data replacement by model values when cosmic rays are detected, and the registration improvement is finished, then the difference image, $dI_{i,j}$, over all pixels of interest is simply set as:

$$dI_{i,j} = I_{i,j} - f(\delta x_{i,j}, \delta y_{i,j}). \tag{5.8}$$

5.3 PSF Changes in DIA

With observations from HST the point spread function changes throughout the orbital period of about 96 minutes as the telescope flexes due to changing thermal conditions. This should be much less of an issue for the *Kepler* data, although in the days after a 90 degree roll, or on time scales of months the *Kepler* PSFs may be expected to change as well. The PSF scales as measured in pixels are quite comparable for HST and *Kepler*. One simple measure of PSF variability for HST is to track the relative change of intensity in the central pixel (for stars well centered on a pixel), compared to the total intensity (as say summed over a 5×5 pixel domain). In the HST projects with nearly continuous observations over 7-8 days designed for gas-giant transit detection, the central pixel intensity changes by typically 20% peak-to-peak over the course of 96 minute orbits. Since these projects were aimed at detecting 1–2% transit depths, with expected time series precisions down to 0.2–0.4% for the brighter stars, and the fields are quite crowded, accounting for these PSF changes becomes the dominant challenge in forming good difference images. For HST data, modelling the PSF changes is the most computationally intensive step in the DIA, and the one that is most tricky to set up. Although not expected to be needed for *Kepler*, it will be informative for risk mitigation in dealing with unexpected analysis needs to describe the HST experience in detail.

In the previous section we described development of a model for each pixel of interest, that captures how the intensity of the pixel is expected to change in response to arbitrary x, y offsets within the range spanned by typical frame-to-frame guiding errors. The solution for this model was done separately for each pixel by performing a least-squares solution of the surface fit $I(x,y)$ with basis functions being bi-cubic Legendre polynomials in two dimensions. Bringing in PSF variations changes the character of the pixel-intensity model dramatically, since the solution can no longer be localized to a single pixel. The PSF changes can be modelled as a convolution of a representative image with a convolution kernal. Thus the correspondence between observed intensity and a model representation as in Equation 5.1 becomes:

$$I_{i,j}(t) = f(\delta x_{i,j}, \delta y_{i,j}) + psf_{l,m}(t) \otimes f(\delta x_{i,j}, \delta y_{i,j}) \tag{5.9}$$

where $psf_{l,m}$ would be an appropriately sized grid of values as required to capture the changing PSF. For the HST case the convolution needs to be over ∼3.5 pixels.

For the ground-based micro lensing projects, where DIA techniques were initially applied, it was common to select a subset of the observations with the best seeing (sharpest PSFs), and form an average, best-seeing model image from these. The convolution kernal to account for matching the PSF of individual frames would then always represent a smearing, i.e., the convolution kernal, represented as a discrete set of pixel values in an l×m matrix would have a central value less than unity, with positive power in neighboring pixels. For the HST data I have experimented with using the full data set to create a reference image, the convolution kernal must then account for cases in which the individual image is blurred compared to the model (central value of kernal less than unity, with positive wings), and cases with sharper individual images (central value greater than unity with net negative power in the wings).

With under-sampled images, and with a set of image-to-image offsets that samples well (and redundantly) the full sub-pixel phase space, I have found that evaluating the differential convolution kernal at a factor of two over-sampling works best. I have adopted a brute-force, least-squares solution for a differential PSF convolution kernal as a 7×7 matrix $psf[l,m]$, by solving for the 49 separate values that minimize:

$$\chi^2 = \sum_{i,j} 1/\sigma^2_{i,j}[I_{i,j} - psf \otimes g(\delta x_{i,j}, \delta y_{i,j})]^2 \tag{5.10}$$

where $g(\delta x_{i,j}, \delta y_{i,j})$ is an image at factor of two over-sampling developed by evaluation of $f(\delta x_{i,j}, \delta y_{i,j})$ at the nominal registration $(\delta x_{i,j}, \delta y_{i,j})$ at each pixel, plus ±0.5 pixel positions. For these fits, and the ones referred to earlier to improve registration, I have found it prudent to eliminate inclusion of variable stars. This has been accomplished by taking a cut above 3σ in a map of *rms* per pixel from the surface fit step of Equation 5.3, and defining such pixels to either be coincident with variable stars, or bad pixels. Also a down-selection is made to only include pixels in the fit that carry significant information, e.g. pixels that seem to represent sky background would not be useful for these fits. In practice I include pixels that never saturate, and that have coefficients a_m, $m = 1, \ldots, 9$ that are significantly non-zero.

With the HST instruments there is mild field dependence of the PSF changes. This has been captured by performing the solution for psf on separate 5×5 domains of the full field, then smoothing the results by fitting with a 2–D, quadratic Legendre polynomial.

The effect of a changing PSF is isolated by generating a 'differential convolution image', $pI_{i,j}$ defined as:

$$pI_{i,j} = f(\delta x_{i,j}, \delta y_{i,j}) - psf \otimes g(\delta x_{i,j}, \delta y_{i,j}) \qquad (5.11)$$

For an individual image $I_{i,j}$ that is blurrier than the average (model) image, $pI_{i,j}$ shows a signature at each point source that is negative in the core with positive wings. For $I_{i,j}$ sharper than average, cores of stars in this 'differential convolution image' are positive with negative wings. A difference image could now be formed as:

$$dI_{i,j} = I_{i,j} - f(\delta x_{i,j}, \delta y_{i,j}) - pI_{i,j}. \qquad (5.12)$$

The rationale for carrying this special image is that in doing so the effects of PSF changes frame-to-frame can now be taken into account during the critical step of forming the surface fits in Equation 5.1, and in particular allowing for robust elimination of cosmic rays. Consider a case in which most images have similar PSFs, but a small subset are blurrier. Then in the approach outlined in the previous section the wings of stars in the subset of blurrier images would often be flagged as cosmic rays, and therefore have their values replaced by the model. This is of course a disaster for photometry in these images, since the higher values in star wings resulted from spatial rearrangement of flux, and clipping these values would therefore result in smaller than real fluxes when sums over the full PSF associated with a star are formed. If the $I_{i,j}$ of Equation 5.1 are temporarily subtracted by the $pI_{i,j}$, then the resulting vector of intensity values is free of the effects of PSF changes, and a proper surface fit intended only to be able to capture the effects of image motion, can be formed. This brings up a critical point which intentionally was not made explicit in the earlier discussion: *Reaching a full DIA solution involves a number of coupled iterations.* And often the starting point for solutions is not well posed.

For example, consider the case in which only a few frames have significantly blurrier PSFs. A simple exercise of the surface fits of §5.2 would result in having the wings of stars thrown away as cosmic rays in these images. If this happened, then later performing the solution for a 'differential convolution kernal' would not capture the effect of the blurred PSF since we would have thrown away the stellar wings. The solution to this involves yet another iteration cycle at the original surface fit (Equation 5.1) stage, in which a solution is made in which a much higher penalty is temporarily adopted for frames which show too many cosmic rays as having been eliminated. In the most recent DIA case I have formed a mask for pixels that are on stars with good, strong, but unsaturated pixels, and a separate mask that corresponds to pixels at the level of sky background. The number of cosmic rays flagged on stars should be $\leq$ the number flagged on sky, adjusted for the relative numbers of pixels in the two masks. For those frames in which relatively more cosmic rays had apparently been detected on stars, than sky, the cosmic ray elimination threshold has been raised as necessary to drop the number of eliminated pixels to a nominal level. Then, and only then, would the solution for the differential PSF for that frame be well posed. Once a decent estimate for the effects of changing PSFs has been isolated in $pI_{i,j}$, then the threshold for cosmic ray elimination in these frames can be lowered to the standard value used generally. But it takes a 'bootstrap' approach to arriving at a full solution. The details of this are not important here, since major adjustments would certainly be required for use with the *Kepler* data.

In Fig. 5.1 the fourth panel from the left contains the 'differential convolution image' as defined in Equation 5.11. Comparing this image to a direct image in the leftmost panel, one can see that structure is reproduced near each stellar source. The structure accounts for the unique shape of the PSF in this individual frame compared to the average over the full set of images.

5.4 Photometry from Difference Images

In Fig. 5.1 a difference image section is reproduced as the rightmost panel. High-amplitude variable stars are trivial to pick out in a movie of difference images – appearing as stellar PSFs that go from positive to negative and back. Some of the variable stars

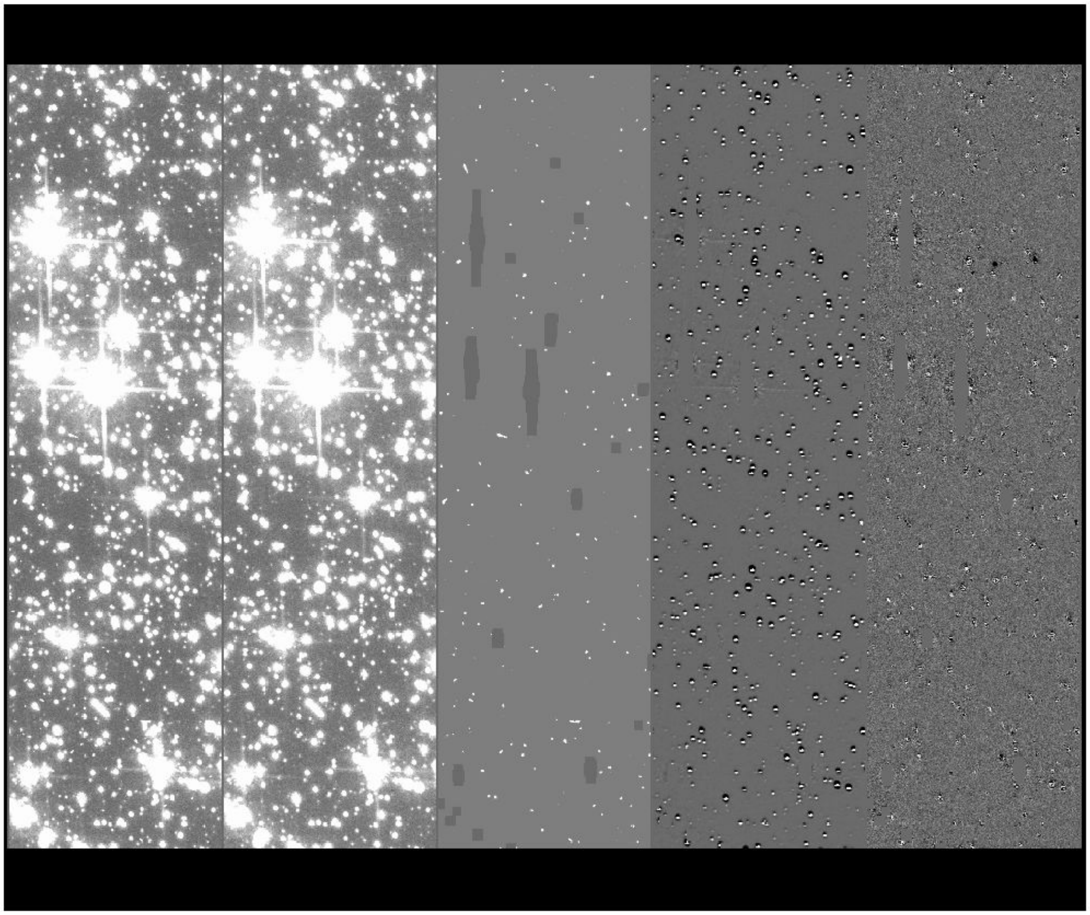

Figure 5.1: Panels from left to right are: 1) original image, 2) same with cosmic rays replaced with estimated data values, 3) the cosmic rays, 4) a differential convolution image, and 5) the corresponding difference image. See the text for additional discussion.

in this region are within strong blends in the direct image. In the difference image the blends have gone away, leaving just the residual positive or negative image at the position of a variable star, depending upon whether it was brighter or fainter than average respectively in this individual frame. The advantage of working with difference images should be clear in this context. If attempting to extract photometry for a strongly blended star in the left panel, one would need to either: (1) use multiple PSFs fit simultaneously to the several blended stars, or (2) use an aperture large enough to encompass the full blend. The drawback with multiple PSF fits is that one must know the shape of the PSF extremely well, and correctly evaluate it for dithered, under-sampled data in order to obtain a good result. With aperture photometry on a blended star much excess noise will be included through having to use a large aperture.

For PSF fitting, or aperture photometry in the difference images it is necessary to know the accurate positions of all stars for which differential intensity estimates are desired. I assume that an excellent star list consisting of intensities and good (x, y) positions are available from separate analysis. Aperture photometry can then be obtained by centering a circular aperture of radius R_{ap} on the nearest integer position of the star, and forming a sum in the difference image:

$$dI_{ap} = (\sum_{r<R_{ap}} dI_{i,j})/I_{norm} \qquad (5.13)$$

thus providing a fractional measure, dI_{ap}, via summing the difference image counts and ratioing this to the sum over the direct model image. The normalization to expected direct counts is: $I_{norm} = \sum_{r<R_{ap}} f(\delta x_{i,j}, \delta y_{i,j})$. If a very small aperture is used, then it would be advisable to use adjustments for partial pixel inclusion at the outer radius.

I usually carry several aperture photometry measures using a range of aperture sizes, and then down-select on a star-by-star basic to the one that provides the smallest noise level.

A PSF fit can be obtained as:

$$dI_{psf} = (\sum_{r<R_{ap}} w_{i,j} psf_{i,j} dI_{i,j} / \sum_{r<R_{ap}} w_{i,j} psf_{i,j}^2) / I_{norm} \tag{5.14}$$

where the weights, $w_{i,j}$ are set simply as the inverse total variance as in Equation 5.4.

The $psf_{i,j}$ can be adopted from an analytic approximation to the general PSF, but needs to be carefully evaluated for the precise position of the star to be fit using a combination of x, y information from the master star list and the unique registration offsets for each frame. In practice, for these HST data that do not show huge field-dependent PSFs, I have taken a single, bright and unsaturated star near field center, and used its model evaluated from Equation 5.1 to define the PSF. In this approach one must properly take into account the relative sub-pixel shifts of the fiducial PSF star, and the target star to be fit. (This is one of several items that should be done better, at least in principle.)

5.5 False Positive Elimination from Difference Images

A primary reason for carrying DIA is that it is likely to provide an invaluable tool for eliminating a significant fraction of false positives that arise from the diluted signal of much fainter, background eclipsing binaries. Using the appropriate sums over difference images to isolate the variability signal can be used to accurately determine the position of the intrinsic variable. Consider the simple case of having detected a weak sinusoidal signal in a star of interest. The problem at hand is to develop a test to determine if the apparent signal more likely results from a background, previously unidentified variable (thus a false positive). The approach here is to make optimal use of the apparent signal characteristics to maximize contrast in a sum over individual difference images. For the sinusoidal signal case this could be done by forming two summed images: (1) all of the difference images at times within 10% of the phase of peak intensity, and (2) all of the difference images at times within 10% of the minimum intensity. Then the highest contrast version of the variable star would simply be obtained as the difference of these two sums. If the position of the resulting summed difference image signal is spatially offset in a statistically significant way from the centroid of the direct image (formed as a simple average over all of the images used in the difference of difference images), then the variability is not coincident with the bright star, and a false positive has been eliminated.

The conceptual advantage to working in difference image space, as compared to comparing sums over direct images, is that in DIA the position of the differential signal will be coincident with the background object. Working in direct image space the bright image would be pulled toward the source of variability only by an amount equal to their separation multiplied by the relative intensity. Since we care about cases in which the background star may be 10,000 times fainter than the star of interest, with a separation of 8 arcsecs (as an example), the variable would pull the centroid by 0.0008 arcsecs, or 0.0002 pixels in the direct image during peak variability. In the summed difference images the variability signal will be offset from the primary star by 8 arcsecs, or two full pixels. While in principle the same information content may formally exist in the two cases, it will surely be easier to recognize the reality of a false positive in the DIA where the location of the putative contaminating star is directly indicated. For an example of DIA in providing the location of a faint, background variable that was not detected until pursuing false positive elimination tests, see Fig. 5.2 (also Figure 1 of Edmonds *et al* 2002). In the original HST data on 47 Tuc with WFPC2 the star corresponding to the variable is not visible in direct, averaged images, but in the sums of difference images chosen at specific phases to accentuate the variability the faint variable is very obvious. In this case there are three stars with a total intensity of over 200 times that of the variable within a radius of $0\farcs2$, and within $0\farcs5$ 13 stars total 800 times that of the mean intensity of the variable. The PSF full-width-half-maximum in these images is about 1.5 pixels, or $0\farcs7$.

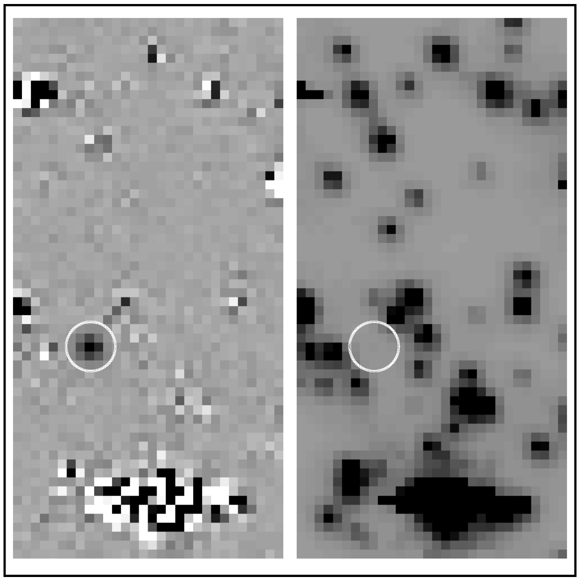

Figure 5.2: Illustration with HST WFPC2 data from the 47 Tuc observations to detect 'Hot Jupiters'. The left panel shows the sum of several difference images selected to be near the peak of variability, minus the sum of an equal number near minimum for this variable. The circle indicates a diameter of 5.5 pixels, or $0\farcs25$. The right panel is an average of the same direct images (with intensities scaled down ×1000 relative to the difference image), the circle shows the same region as in the left panel. The variable is obvious in difference images, not visible in direct images.

For the case of pursuing false positive elimination on candidate transit signals the procedure would be to sum over all of the difference images from within the time span of the candidate transits (and take the negative of this in order to deal with a positive signal). The nominal positional error for x, y centroids via PSF fits is the characteristic PSF scale divided by the photometric signal-to-noise (King 1983). For Kepler photometry the characteristic PSF scale is about one pixel, and the smallest signal that will be believed as a candidate transit (averaged over 3 – 4 individual transits) is about 8σ, therefore the error on positions will be (to within factors of less than two) $\leq$ 1/8 pixel. In general terms it should therefore be possible to detect position shifts not consistent with zero, when the offsets are greater than 1/2 pixel. Therefore, to this first level of approximation, only background stars within about the central pixel area of typically 20 pixels summed for the time series, cannot be sieved for false positives. Once the background variable is well off the peak of the bright star, the signal-to-noise of the differential photometric, or relative positional offset signal will be higher than the value contrasted to the full signal of the primary target, thus further improving the power of this approach.

In general it will be necessary to perform the DIA test for false positives by doing a full, properly weighted PSF fit in which intensity and x, y are directly solved for with proper error estimation. There is a class of potential signals in target stars from 'background' objects that DIA will likely not be useful for. If a nearby very bright star, say 7^{th} magnitude for *Kepler*, that we do not derive photometry for has a low amplitude eclipse, then the wings of its PSF could supply sufficient signal to nearby stars that we do follow, to yield an apparent transit in the target star. Since the contamination is spread fairly uniformly over our small aperture, the centroid analysis on the suspect transit signal would not pick up a shift. We might detect this via correlated (false) signals in a group of stars near the bright contaminating stars. We might also trust that we could know whether any stars in the *Kepler* field of view brighter than those we will follow (8^{th} magnitude if we include the faintest subset of saturated stars) are short period binaries capable of showing contaminating events.

5.6 Hot Pixel Tracking

Experience from HST indicates that false positives arising from hot pixels, or significantly increased noise from a hot pixel within a photometric aperture will be rare (see Gautier and Gilliland 2004).

We have tentatively decided not to track hot pixels, at least not in the sense of trying to feed the results back for potential use in reprocessing of the image-level data at the Data Management Center. Nonetheless, the SOC will surely want to invoke some level of hot pixel tracking to serve two purposes: (1) Having an easy means at hand for testing for false signals from a hot pixel with a pathological temporal behavior which could mimic isolated transits, and thus in conjunction with noise events lead to false positives. (2) Maintaining a detailed general analysis of the behavior and growth rate of hot pixels that would be used to make decisions regarding annealing if this capability is maintained in hardware.

In general terms I would envision the following types of hot pixel tracking (more generally ‘hot’ might be replaced by ‘bad’ here):

1. To track the general build up of hot pixels in time a sum of difference images can be formed that consists of the final 24 hours within a quarterly pointing minus a sum of 24 hours from early in the roll period. On a pixel-by-pixel basis it will be possible to form the ratio of late minus early mean intensity ratioed to the noise level, pixels at several sigma levels of significance likely represent a hot pixel that has turned on and stayed on as a result of radiation damage at some point in the intervening three months. Should it be of interest to attempt correcting for this, an edge detection algorithm could be run over the time series of intensity for the single pixel in question from the difference images to isolate when it turned on.

2. Similarly, a ‘chi-squared’ map can be made from the difference images over the full quarter by evaluating the pixel-by-pixel standard deviation and ratioing this to the expected noise on each pixel. This will turn up some variables, and flaky or hot pixels.

3. In the same sums discussed in §5.5 intended to provide a sum of differences at phases to accentuate the candidate signal, an inspection should be performed to test against the unlikely prospect that the candidate signal arises from pathological behavior of a hot pixel that turns on and off to mimic a transit. Since in this case the resulting signal would be isolated to one pixel (moderated by charge diffusion), it should be easy to flag such cases (sensitivity is compromised in this step by having a strongly under-sampled PSF, in the case of wanting to discriminate against the possibility that the central pixel is at fault).

5.7 Unique Complications of the *Kepler* Data

In comparison to HST data, the *Kepler* data are expected to provide a generally well-posed basis for DIA. However, there are a number of areas in which the 7 – 8 day HST-based observation sets do not provide good analogues.

One certain complication is that some parts of the *Kepler* focal plane will experience drifts from differential velocity aberration within the three month rolls that are larger than the expected scale of 15-minute to 15-minute integration jitter. A difference image can best be formed from using a large ensemble of frames at pointings that span the position of the individual image; drift in time will always compromise this to some degree at the extrema in time. It might be best to break the data into multiple segments, ideally maintaining a ‘rolling center’ for the bulk of the data. For example it might well be reasonable to form difference images for one week intervals using all the images from three week periods centered on the target week, and advancing the block one week at a time in a rolling fashion. An alternative would be to treat the full quarter in the surface fits of Equation 5.1 building in enough terms to track both the intensity changes associated with minor offsets integration-to-integration, as well as from the larger scale drifts from differential velocity aberration.

If the PSF changes in a significant way as a function of time in the *Kepler* data it may be more difficult to deal with, than with the HST data. With the HST data we always had dithers that provided good sampling of the full sub-pixel phase space of x, y offsets. With *Kepler* we do not expect to have anything remotely close to full sub-pixel phase space coverage. This means that it will not be possible to define a simple over-sampled image, and it would therefore be potentially much harder to do something reasonable along the lines of PSF compensation as discussed in §5.3. Since the only PSF changes anticipated for *Kepler* are slow drifts, we can assume that we will not need to design for active PSF compensation as was done for HST data.

Chapter 6

An Introduction to Detection Theory

This chapter introduces basic detection theory from the standpoint of testing a simple binary hypothesis in the presence of White Gaussian Noise (WGN). The concept of a detection statistic is introduced and the properties of the simple matched filter are explored. The problem of setting a threshold is discussed and an empirical approach is put forth to handle the case of detecting transiting planets. The problem of detecting a deterministic signal in colored Gaussian noise is then described, and the problem of designing a whitening filter is discussed.

6.1 Simple Binary Hypotheses for WGN

In this section we introduce the simple matched filter as the solution to binary hypothesis problems for additive WGN. Throughout this discussion we will assume that the data consists of measurements $x(n)$ for $n = 1, \ldots, N$, and that there are two possibilities denoted $H0$ and $H1$. Under the null hypothesis, $H0$, the data consists solely of noise, $w(n)$, while under the alternative hypothesis, $H1$, the data consist of a combination of noise and the signal of interest, $s(n)$,

$$\begin{aligned} H0 &: x(n) = w(n) \\ H1 &: x(n) = w(n) + s(n), \end{aligned} \qquad (6.1)$$

where $w(n)$ is zero-mean WGN with variance σ_w^2.

The task before us is to design an algorithm that will detect the presence of the signal in the observations. What is a detection algorithm? Essentially it is a set of mathematical computations that transform the data set, $x(n)$, into a scalar value, T, called the detection statistic, which is compared against a threshold, η to detect s. Given the properties of the noise and the transformation yielding T, it is possible to determine the distribution of T and assess the significance of any value observed for T. Note that T is a random variable, since we are dealing with a stochastic process, $w(n)$, embedded in the data set. Additionally, T should be a scalar; that is, T should summarize all the available information about the phenomenon of interest and answer the question: Is s present in x or not? In this sense, $x(n)$ is not restricted to being a time series: it could be a combination of measurements from different instruments or bandpasses. What is important is that the computation of T take into account all the relevant information available with which to make the decision. Once T is determined, it is compared to a threshold η, and if T exceeds η, we accept $H1$ and say that we've detected the signal of interest. On the other hand, if T is less than the threshold, we reject the alternative hypothesis and say that s is not present in x. A major problem aside from determining the mapping from x to T is determining the appropriate threshold. This depends on the desired false alarm rate, which depends on the statistical distribution of T in the absence of a signal. The simple case considered here has a closed form solution for both the optimal detector and for setting the threshold.

A well-known result from detection theory (see, e. g., 68) is that if $s(n)$ is known, the optimal detector is a simple matched filter of the form

$$T = \frac{\mathbf{x}^{\mathrm{T}}\mathbf{s}}{\sigma_w \sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}}, \qquad (6.2)$$

where we've used vector notation to denote the time series [i. e., $\mathbf{x}$ in place of $x(n)$]. In terms of the vector space underlying the observations, T is the dot product of the data vector with the signal vector, normalized by the product of the standard deviation of the measurement noise with the magnitude of the signal vector. Now the question is how to interpret T? T is a linear combination of Gaussian random variables, hence it also is Gaussian (88). We need only specify the mean and standard deviation of T under each hypothesis in order to fully characterize the performance of the detector. Under the null hypothesis, $x(n)$ is composed entirely of noise, so that the expected value of T is given by

$$\begin{aligned}\langle T\rangle_{H0} &= \left\langle \frac{\mathbf{w}^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}}\right\rangle \\ &= \frac{\langle \mathbf{w}^{\mathrm{T}}\rangle\,\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}} = \frac{\mathbf{0}^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}} = 0,\end{aligned} \tag{6.3}$$

where $\langle\cdot\rangle$ is the expectation operator. The variance of T under $H0$ is given by

$$\begin{aligned}\left\langle \left(T-\bar{T}\right)^2\right\rangle_{H0} &= \left\langle \left[\frac{\mathbf{w}^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}}\right]^2\right\rangle \\ &= \frac{\mathbf{s}^{\mathrm{T}}\left\langle \mathbf{w}\,\mathbf{w}^{\mathrm{T}}\right\rangle\mathbf{s}}{\sigma_w^2\,\mathbf{s}^{\mathrm{T}}\mathbf{s}} \\ &= \frac{\sigma_w^2\,\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w^2\,\mathbf{s}^{\mathrm{T}}\mathbf{s}} = 1.\end{aligned} \tag{6.4}$$

Under $H1$, the expected value of T is

$$\begin{aligned}\langle T\rangle_{H1} &= \left\langle \frac{(\mathbf{w}+\mathbf{s})^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}}\right\rangle = \frac{\left(\langle\mathbf{w}\rangle+\mathbf{s}\right)^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}} \\ &= \frac{\mathbf{0}^{\mathrm{T}}\mathbf{s}+\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}} = \frac{\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}}{\sigma_w}.\end{aligned} \tag{6.5}$$

The term $\langle T\rangle_{H1}$ is often called the Signal to Noise Ratio (S/N) of the signal s, and together with the noise distribution, determines the detectability of s. Similarly, the variance of T under $H1$ is given by Equation 6.6.

$$\begin{aligned}\left\langle \left(T-\bar{T}\right)^2\right\rangle_{H1} &= \left\langle \left[\frac{(\mathbf{w}+\mathbf{s})^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}} - \frac{\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}}{\sigma_w}\right]^2\right\rangle \\ &= \left\langle \left[\frac{(\mathbf{w}+\mathbf{s})^{\mathrm{T}}\mathbf{s}}{\sigma_w\sqrt{\mathbf{s}^{\mathrm{T}}\mathbf{s}}}\right]^2\right\rangle - \frac{\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w^2} \\ &= \frac{\left\langle \left(\mathbf{w}^{\mathrm{T}}\mathbf{s}+\mathbf{s}^{\mathrm{T}}\mathbf{s}\right)^2\right\rangle}{\sigma_w^2\,\mathbf{s}^{\mathrm{T}}\mathbf{s}} - \frac{\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w^2} \\ &= \frac{\mathbf{s}^{\mathrm{T}}\left\langle \mathbf{w}\,\mathbf{w}^{\mathrm{T}}\right\rangle\mathbf{s}+2\,\langle\mathbf{w}\rangle^{\mathrm{T}}\mathbf{s}\,\mathbf{s}^{\mathrm{T}}\mathbf{s}+\left(\mathbf{s}^{\mathrm{T}}\mathbf{s}\right)^2}{\sigma_w^2\,\mathbf{s}^{\mathrm{T}}\mathbf{s}} \\ &\quad - \frac{\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w^2} \\ &= \frac{\sigma_w^2\,\mathbf{s}^{\mathrm{T}}\mathbf{s}+2\mathbf{0}^{\mathrm{T}}\mathbf{s}\,\mathbf{s}^{\mathrm{T}}\mathbf{s}+\left(\mathbf{s}^{\mathrm{T}}\mathbf{s}\right)^2}{\sigma_w^2\,\mathbf{s}^{\mathrm{T}}\mathbf{s}} - \frac{\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w^2} \\ &= \frac{\sigma_w^2+\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w^2} - \frac{\mathbf{s}^{\mathrm{T}}\mathbf{s}}{\sigma_w^2} = 1.\end{aligned} \tag{6.6}$$

Thus, T is of unit variance under either $H0$ or $H1$, and the two corresponding distributions of T are separated by the S/N of the signal to be detected. The distance between the two distributions determines the detectability of s. Figure 6.1a shows the probability density distributions under $H0$ and $H1$ for a signal with an SNR of 4σ. Each time we test for the presence of s, we are drawing a random number from either the distribution governed by $H0$, or the one governed by $H1$. The false alarm rate for our detector is the area under the curve of the PDF for $H0$ to the right of η, while the detection rate is the area under the curve of the PDF for $H1$ to the right of η. The higher the mean S/N of s, the higher the probability of detecting it for a given threshold. Both the false alarm rate P_{FA} and the detection rate, P_{D} are functions of η. We can plot P_{D} versus P_{FA} to examine the relationship between these two quantities as a function of S/N, as is shown in Fig. 6.1b.

For problems such as transit detection, where the probability that the desired signal is present in the data is not known, or is poorly constrained, the most common method for establishing the detection threshold is the Neyman Pearson criterion. The trick is to choose a value for η that maximizes the detection rate while achieving the desired false alarm rate.

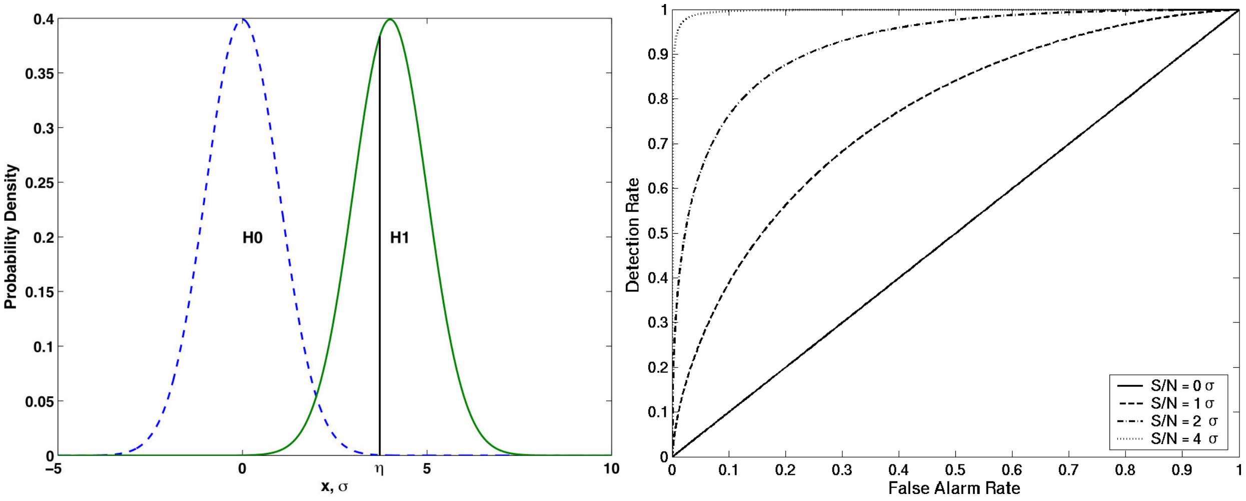

Figure 6.1: Panel a) Probability density distributions for the null statistics and detection statistics of a 4σ signal. The threshold, η determines the false alarm rate and the detection rate of the detector. Panel b) Receiver operating curves (ROCs) for the binary hypothesis problem for additive WGN. The curves for signal S/N's of 0σ, 1σ, 2σ and 4σ are shown by the solid, dashed, dot-dashed, and dashed, respectively. As the S/N increases, we are able to achieve higher detection rates for a given false alarm rate.

With Equations 6.2-6.6 in hand, we are in a position to specify the detection threshold, η. Since T is unit-variance and Gaussian under both $H0$ and $H1$, we can easily calculate both the false alarm rate and the detection rate. Under $H0$, the chance that $T \geq \eta$ is given by

$$P_{FA} = \frac{1}{\sqrt{2\,\pi}} \int_{\eta}^{\infty} \exp(-y^2/2)\, dy, \qquad (6.7)$$

while the probability that $s(n)$ will be detected if present is given by

$$P_D = \frac{1}{\sqrt{2\,\pi}} \int_{\eta-\langle T\rangle}^{\infty} \exp(-y^2/2)\, dy. \qquad (6.8)$$

The threshold is simply selected to achieve the desired false alarm rate. For example, given a signal with an SNR of 4 σ and a desired false alarm rate of 1×10^{-4}, $\eta = 3.72\,\sigma$, and so $P_D = 0.61$. If the location of $s(n)$ within the data stream is unknown, then the matched filter of equation 6.2 can be implemented by correlating a normalized version of $s(n)$ with the input data stream. Another name for this implementation is a correlation receiver. For the case of WGN, the resulting matched filter has a constant false alarm rate (CFAR), which is a desirable property for detectors. In searching for transiting planets we are faced with the same problem of not knowing when the transits occur, and so must apply the matched filter at all time steps. The analysis is complicated, however, by the desire to search for periodically spaced transits. Fortunately, there are Monte Carlo analysis techniques to establish the effective number of independent statistical tests conducted in searching a light curve for transiting planets over a fixed period range (62). This allows us to set an appropriate threshold to control the total number of false alarms, as desired. This topic is further explored in section 6.3.

6.2 Colored Gaussian Noise

In this section we provide generalizations of the results from the previous section (6.1). We still have a binary hypothesis (for each test), but $w(n)$ is no longer restricted to be white, although we assume that it is Gaussian. Colored Gaussian noise can be modeled as the result of filtering WGN through a linear but possibly time-varying filter (50). The filtered noise process may possess an auto-correlation matrix with non-zero off-diagonal entries. In this case, a matched filter provides the optimal detector, but it has a different form from that of equation 6.2:

$$T = \frac{\mathbf{x}^{\mathrm{T}}\,\mathbf{R}^{-1}\mathbf{s}}{\sqrt{\mathbf{s}^{\mathrm{T}}\,\mathbf{R}^{-1}\mathbf{s}}} \qquad (6.9)$$

where $\mathbf{R}$ is the autocorrelation matrix of the noise, $w(n)$. In order to interpret equation 6.9, note that since R is non-singular and symmetric, it possesses a square root, so that Eq. 6.9 can be rewritten as

$$T = \frac{\left(\mathbf{R}^{-1/2}\mathbf{x}\right)^{\mathrm{T}}\left(\mathbf{R}^{-1/2}\mathbf{s}\right)}{\sqrt{\left(\mathbf{R}^{-1/2}\mathbf{s}\right)^{\mathrm{T}}\left(\mathbf{R}^{-1/2}\mathbf{s}\right)}} = \frac{\tilde{\mathbf{x}}^{\mathrm{T}}\tilde{\mathbf{s}}}{\sqrt{\tilde{\mathbf{s}}^{\mathrm{T}}\tilde{\mathbf{s}}}}, \qquad (6.10)$$

where $\tilde{\mathbf{x}} = \mathbf{R}^{-1/2}\mathbf{x}$ and $\tilde{\mathbf{s}} = \mathbf{R}^{-1/2}\mathbf{s}$ are 'whitened' versions of the data and signal vectors. Thus, the optimal detector consists of the cascade of a whitening filter with a matched filter. The difficulty lies in designing the whitening filter itself, as the correlation matrix $\mathbf{R}$ is often unavailable. It is important to note that the whitening filter may distort $\mathbf{s}$, and that the resulting detector seeks the transformed version of $\mathbf{s}$. The results of §6.1 hold, however, with respect to the detectability of $\tilde{\mathbf{s}}$.

If the mean value and the correlation structure of the noise process are stationary (i. e., constant in time) and certain additional mild conditions are met, equation 6.9 can be expressed in the frequency domain as per Kay (67):

$$T = \int_{-\pi}^{\pi} \frac{X(\omega)S^{*}(\omega)\,d\omega}{P(\omega)} \Big/ \sqrt{\int_{-\pi}^{\pi} \frac{S(\omega)S^{*}(\omega)\,d\omega}{P(\omega)}}, \qquad (6.11)$$

where $X(\omega)$ and $S(\omega)$ are the Fourier transforms of the data and signal, respectively, '$*$' denotes complex conjugation, and $P(\omega)$ is the power spectrum of the noise. Kay (67) suggests an adaptive matched filter based on equation 6.11 using a smoothed periodogram to estimate $P(\omega)$. The main difficulty with designing the whitening filter in the frequency domain is that the statistics of $w(n)$ may not be stationary. Indeed, the solar irradiance clearly exhibits nonstationary behavior over the solar cycle. We can expect that other solar-like stars will exhibit nonstationary behavior. In Chapter 7 we detail a wavelet-based, adaptive matched filter that constructs a time-varying whitening filter by analyzing the noise power in each orthogonal channel of a filterbank implementation. This approach is similar to the familiar graphic equalizer of a stereo system where the user can adjust the volume of each band independently of the others. In our case, the 'equalizer' scales the input of each channel so that the outputs have equal power density. Additional details are deferred until chapter 7. The next section discusses a solution for establishing the threshold for transit photometry campaigns.

6.3 Setting Thresholds for Transit Searches

In this section we discuss the problem of determining the equivalent number of statistical tests conducted in searching a photometric data set for transiting planets. A similar problem is encountered in detecting sinusoidal signals in noise-corrupted time series. Horne & Baliunas (56) proposed a Monte Carlo technique for determining the effective number of independent frequency bins in the Lomb-Scargle periodogram of a time series, an essential step in determining an appropriate detection threshold and for assessing the statistical significance of any peak in the periodogram. Here we propose an analogous approach for the transit detection problem. To open the discussion we review some basic detection theory relevant to the problem and then illustrate various facets of the problem for non-Gaussian noise. We provide an argument supporting the validity of the results derived for white Gaussian noise to more general cases of colored non-Gaussian noise. We proceed with the case of white Gaussian observation noise, giving a prescription for determining the effective number of independent tests. This is followed by several examples drawn from actual or anticipated observations.

If we wish to detect a deterministic signal in a noisy data set where the noise is Gaussian (colored or white), the optimal detector consists of a prewhitening filter followed by a matched filter detector (c. f., 68). For the transit detection problem, a whitening filter can be thought of in terms of detrending the light curve to make it possible for a simple matched filter to detect a transit. Simple matched filters do *not* take into consideration points 'out of transit'. Thus, if the transits are superimposed upon a slowly-varying background with large excursions compared to the depth of transit, and if no prewhitening is performed, the matched filter will have a difficult time distinguishing transits from negative excursions occurring on longer timescales. The details of implementing a whitening filter depend a great deal on the specific observation characteristics: the contiguity of the data set, the uniformity of the sampling, etc. All whitening filters represent an attempt to use 'out of transit' points to predict the flux 'in transit', i.e., whitening filters presuppose a knowledge of the correlation structure of the observation noise. Here we will assume that the noise is white or has been whitened. Now if the noise is not Gaussian, this detector may not be optimal. However, well-sampled photometric observations are often moderately characterized as Gaussian once outliers caused by cosmic rays and poor observing conditions are removed. In any case, time domain matched filters or their equivalent are the dominant detection strategies employed in this area. Thus, it is fruitful to consider this model given its popularity. We will further assume that the data has been treated in such a way that the transit pulse shapes are well preserved, or that the effects of the pre-whitening filter on the shape of the "whitened" transit are known. The search for transits of a given star's light curve, then, consists of convolving the light curve with a sequence of model transit pulses (distorted in the case of a pre-whitener that does not preserve transit shape) spaced by each trial orbital period. Equivalently, the light curve may be convolved with a single model transit pulse and then folded at each trial period. The resulting detection statistics are examined for large positive values, the location of which gives the orbital period and phase of candidate planets. Equation 6.12 provides the formulation for a simple matched filter:

$$l = \frac{\mathbf{b} \cdot \mathbf{s}}{\sigma\sqrt{\mathbf{s} \cdot \mathbf{s}}} = \frac{1}{\sigma}\mathbf{b} \cdot \hat{\mathbf{s}}, \tag{6.12}$$

where **b** is the data vector, **s** is the signal to be found, and σ is the standard deviation of the zero-mean, white Gaussian noise. Note that this is simply the length of the projection of the data vector along the direction of the signal vector. Under the null hypothesis (no transits), l is a zero-mean, unit-variance Gaussian random variable. Likewise it can be shown under the alternative hypothesis of **s** being present that l is a unit variance Gaussian random variable with a mean equal to $\sqrt{E_s}/\sigma$. Here, $E_s = \sum_i s_i^2$ is called the energy of **s**. For transits consisting of rectangular pulse trains, equation (6.12) collapses into the square root of the number of points in transit times the mean data value during transit divided by the standard deviation of the observation noise.

In applying the detection algorithm one will in practice construct a rather large number of detection statistics in order to densely sample the region of the parameter space of interest. For example, suppose we have 6 weeks of data from a ground-based program at a resolution of 4 hr^{-1} and 12–hours of observations each night and search for transiting planets with periods between 2 and 7 days. The step size in phase should be about $1/4$ a transit duration, or 45 min. The step size in trial period should be set so that the furthest transits from a fixed central one do not shift more than about half a transit duration from those for the previous trial period. The outermost transit pulses shift by one half the number of periods multiplied by the change in period. The average step size in period for this case is (3 hours/2)/(6 weeks/2/4.5 days) = 19 minutes, giving $\sim$ 373 trial periods. The average number of tests at any period is 4.5 days/(3 hours/4) = 144 tests. Thus, there are roughly 53,000 test statistics required per star to retain good sensitivity to all possible period/phase combinations. For 5,000 stars, then, there are $\sim 3 \times 10^8$ test statistics constructed. The tests for each star are not independent, however, as every trial period will test for a transit at a given point in time for some trial phase. Thus the set of detection statistics for such a search is highly correlated and possesses a complex web of correlations.

This is illustrated by the following example. Consider star Cyg1433 from the NASA Ames' Vulcan Survey. Vulcan 1433 is a binary consisting of two late-F dwarfs undergoing grazing eclipses (Caldwell, Borucki, & Lissauer 2000). This star exhibits a transit-like feature with a depth of 3.19%, a duration of 3.36 hours and a photometric period of 1.957 days (the orbital period is twice this value). The folded light curve for this star is displayed in Figure 6.2a with the phase normalized such that the 'transits' occur at a normalized phase of 0.25. By conducting a search for planets with orbits between one and seven days on a grid with 7.5-min spacing, we test 885,504 different models against the light curve. Figure 6.2b shows the maximum detection statistic obtained for each period sought for 2.5-hour transits. The maximum statistic obtained is 27.7 σ at a period of 1.96 days. Strong peaks are observed at rational harmonics of the fundamental photometric period, and the curve is elevated well above that of the bottom curve in the figure, which is the result for Cyg1433's light curve once the transits are removed from the data. The multiple peaks in the top curve, which might be confusing at first sight, actually provide confirmation that the signal being picked up is caused by a periodic set of pulses of comparable depth. For most of the searches discussed in the remainder of this paper we set up a nonuniform grid with respect to orbital period based on the following criterion. The correlation coefficient between a test at a given phase and period and the highest-correlated test at the next largest period is no less than 0.75. This dictates the step size in period for a given period and number of transits observed, and yields a maximum reduction in apparent SNR of only 12.5%.

We define the quantity $\mathbf{l_{max}}$ as the maximum detection statistic over all tests of a light curve:

$$\mathbf{l_{max}} = max_i\{l_i\}. \qquad (6.13)$$

The complementary cumulative distribution function (CCDF), $\overline{F}_{l_{max}}(x) = 1 - F_{l_{max}}(x)$ of $\mathbf{l_{max}}$ interests us here [1]. $\overline{F}_{l_{max}}(x)$ is the false alarm rate of a single search as a function of the detection threshold, x. The question is, how many independent tests, N_{EIT}, were effectively conducted in performing the search? By this we mean, how many independent draws from a N(0,1)[2] process are required in order for the distribution of the maximum of the N_{EIT} draws to match the distribution of $\mathbf{l_{max}}$ over some given range of the x–axis containing the desired false alarm rate? We call this process $\mathbf{N_{max}}$ and the corresponding distribution, $F_{N_{max}}(x; N_{\mathrm{EIT}})$, and density, $f_{N_{max}}(x; N_{\mathrm{EIT}})$. We do not require that the two distributions match over the entire x–axis, just over the portion of interest.

The domain of interest warrants further discussion. The goal of this endeavor is to choose an appropriate threshold for individual tests. Strictly speaking, if the observation noise is WGN, the complementary distribution $\overline{F}_{l_{max}}(x)$ provides this information directly; the value, x, of $\mathbf{l_{max}}$ for which the sample CCDF $\overline{F}_{l_{max}}(x) = N_{\mathrm{FA}}/N_{stars}$ is the appropriate single-test threshold, where N_{FA} is the total number of false alarms. We note that in searching N_{stars} light curves for planets we are performing N_{stars} independent searches. (If the searches are not independent, then something has gone wrong with the processing of the photometric data, as the resulting light curves should not be correlated, and hence, under the assumption that the observation noise is normal, the searches must be independent.) If we restrict the single search false alarm rate to be $N_{\mathrm{FA}}/N_{stars}$, the total expected false alarms is constrained to be equal to the desired N_{FA}. This reasoning can be extended to individual tests as well. If the distribution $\bar{F}_{l_{max}}$ can be approximated by the distribution $\overline{F}_{N_{max}}(x; N_{\mathrm{EIT}})$ in the region near $N_{\mathrm{FA}}/N_{stars}$, then it is sufficient to choose the single-test false alarm rate to be $N_{\mathrm{FA}}/N_{stars}/N_{\mathrm{EIT}}$ using the actual single test statistics. Thus the region of interest is centered on $\overline{F}_{l_{max}} = N_{\mathrm{FA}}/N_{stars}$.

[1] Throughout this paper the term density refers to the probability density function of a random variable. That is, given a random variable $\mathbf{y}$ (denoted by boldface type), the density or probability density function (PDF) is the function defined as the probability that an instance of $\mathbf{y}$ is confined to an infinitesimal interval about x : $f_y(x) = \lim_{\Delta x \to 0}\{P(x \leq \mathbf{y} \leq x + \Delta x)/\Delta x\}$. The term distribution refers to the cumulative probability distribution function (CDF), $F_y(x)$, where $F_y(x) = P(\mathbf{y} \leq x)$. The term complementary cumulative distribution function (CCDF) refers to $\overline{F}_y = 1 - F_y(x)$.

[2] An $N(\mu, \sigma)$ distribution is defined as normal (i.e. Gaussian) with mean μ and variance σ^2.

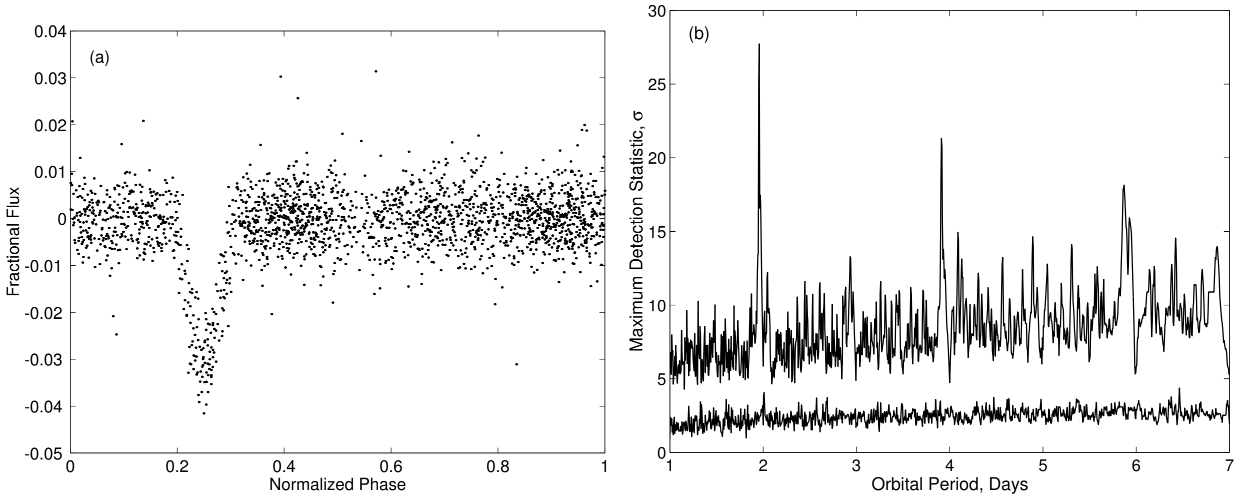

Figure 6.2: Folded light curve for star Cygnus 1433 from the Vulcan campaign (a) and maximum detection statistics for a search for planets with orbits between 1 and 7 days for this star (b). The light curve is folded so that the transit-like feature occurs at a normalized phase of 0.25. The maximum detection statistic obtained for 2.5-hour transits is plotted for each period for the original light curve (top curve) and for the light curve obtained by removing the transits from the light curve (bottom curve). Note the sharp peaks appearing at multiples of the fundamental period 1.96 days. The top curve is elevated above the bottom curve because there is some phase for each period sought corresponding to a model light curve with transits overlapping at least one of the features in the original light curve.

Now, to derive the distribution $\overline{F}_{N_{max}}(x;N_{\mathrm{EIT}})$, we recall that the joint density of N_{EIT} independent Gaussian variables $X = \{x_i\}_{i=1,\dots,N_{\mathrm{EIT}}}$ is

$$f(x_1,x_2,\dots,x_{N_{\mathrm{EIT}}}) = \prod_{i=1}^{N_{\mathrm{EIT}}} g(x_i), \qquad (6.14)$$

where

$$g(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}x^2) \qquad (6.15)$$

is the PDF of an N(0,1) process (88). The density of $\mathbf{N_{max}}$ can be obtained by noting that the probability of the maximum of N_{EIT} draws from an N(0,1) process attaining a value, x, is the probability of any one of the draws being equal to x times the probability that the remaining draws are less than or equal to x. As the draws are independent, we can write the density of $\mathbf{N_{max}}$ by inspection:

$$f_{N_{max}}(x;N_{\mathrm{EIT}}) = N_{\mathrm{EIT}}\, g(x) G(x)^{N_{\mathrm{EIT}}-1}, \qquad (6.16)$$

where

$$G(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-\frac{1}{2}y^2)\, dy \qquad (6.17)$$

is the CDF of an N(0,1) process. The distribution of $\mathbf{N_{max}}$ is simply the distribution of an N(0,1) process raised to the N_{EIT}^{th} power:

$$F_{N_{max}}(x) = G(x)^{N_{\mathrm{EIT}}}. \qquad (6.18)$$

Thus, if the CCDF $\overline{F}_{l_{max}}(x) = N_{\mathrm{FA}}/N_{stars}$ at $x = \eta$

$$N_{EIT} \approx \log\left\{1 - \frac{N_{\mathrm{FA}}}{N_{stars}}\right\} / \log\{G(\eta)\}. \qquad (6.19)$$

If the joint distribution of the tests were known, the distribution of $\mathbf{l_{max}}$ could be found analytically or numerically, at least in principle. Given the correlation matrix, $\mathbf{C}$, for the tests, the joint characteristic function is $\Phi(\Omega) = \exp\{-\frac{1}{2}\Omega\mathbf{C}\Omega^t\}$, but the joint density requires the inverse correlation matrix $\mathbf{C}^{-1}$ (88). We note that the detection statistics are drawn from a N_{points}–dimensional space, where N_{points} is the size of the data set. Hence, there can be no more than N_{points} linearly independent tests performed over the data set. However, the parameter N_{EIT} of the process, $\mathbf{N_{max}}$, may be much larger than the number of observations, N_{points}, for a given sampling and planetary search, as will emerge from the examples considered later on. This underscores the fact that statistical independence of the tests conducted over a search space is separate from the linear independence of the signals considered as vectors in the underlying observation space. For the 6-week-long observations considered above, there are only $\sim 2,000$ observations, with $\sim 53,000$ tests applied to these points. Moreover, since there are more tests than points, $\mathbf{C}$ must be singular, and thus, there doesn't appear to be a closed-form expression for the joint density of the tests. In any case, given the large size of the correlation matrix, integrating the joint density or joint characteristic function either analytically or numerically is impractical. Below we advocate the study of the distribution of $\mathbf{l_{max}}$ through Monte Carlo experiments.

Here we argue that the equivalent number of independent tests conducted per star, N_{EIT}, is not determined by the distribution of the observation noise, and is not strongly influenced by the presence of (red) colored noise. Appendix E provides a proof that the distribution of the observation noise does not affect the value of N_{EIT}. Although the algorithm we provide in this section to estimate N_{EIT} is not affected by the actual noise distribution, the single-test threshold must be established by considering the actual distribution for the detection statistics. We first note, however, that even if the observational noise is not Gaussian, we require that it be of bounded variance and that the light curves have been cleaned of strong, isolated outliers. Thus, the observation noise density should be well confined, even if the tails are longer than that for a Gaussian process with the same standard deviation. Second, we note that each detection statistic is a linear combination of several samples of observation noise. In most practical situations many samples 'in transit' will be obtained simply by the fine sampling grid applied to ensure good sensitivity to the edges of transit events. For instance, the examples from the ground-based program we draw upon feature sampling at ≥ 4 hr^{-1}, giving at least 8 points per transit for transits longer than two hours. Furthermore, we require in general that several (≥ 3) transits be observed. By the central limit theorem (88), the density of the detection statistics may be well or moderately characterized as being Gaussian even in the event that the observation noise on individual data points is not. For example, let the observation noise $w(n)$ be white and drawn from the mixed Gaussian distribution with density

$$f(x) = \sqrt{\frac{5}{8}}\left[g\left(\sqrt{\frac{5}{2}}x\right) + \frac{1}{2}g\left(\sqrt{\frac{5}{8}}x\right)\right] \quad (6.20)$$

and with corresponding distribution

$$F(x) = \frac{1}{2}\left[G\left(\sqrt{\frac{5}{2}}x\right) + G\left(\sqrt{\frac{5}{8}}x\right)\right]. \quad (6.21)$$

In this case, $w(n)$ is a zero-mean, unit-variance process, but is distinctly non-Gaussian. Now consider 1) single transit statistics, l_1, for three-hour transits (12–point pulses) and 2) three transit statistics, l_3, for three three-hour transits (36 samples from the mixed distribution). Figure 6.3 shows the CCDFs for $w(n)$, for l_1, for l_3, and for an N(0,1) process. Note that the single transit and three-transit statistics are well-modeled as being drawn from a N(0,1) process even though w(n) is not a N(0,1) process.

For the case of red noise, if the correlation length of the noise were comparable to the length of a transit, we would expect N_{EIT} to be less than for the case of white noise. Consider a colored noise process generated by passing a WGN process through a low pass filter with impulse response $\mathbf{h}$: $\mathbf{w_c} = \mathbf{w} * \mathbf{h}$. For this example, assume $\mathbf{h}$ is a rectangular pulse of length 3 hours. In applying a simple matched filter for single 3-hour transits, we convolve the observed noise, $\mathbf{w_c}$, with the unit-energy signal,

$$\hat{\mathbf{s}} : l_c(t) = k\,\mathbf{w_c} * \hat{\mathbf{s}} = k\,\mathbf{h} * \mathbf{w} * \hat{\mathbf{s}} = k\,\mathbf{h} * \mathbf{l}(t), \quad (6.22)$$

where $k = 1/\sqrt{E_{\hat{\mathbf{s}}*\mathbf{h}}}$ is a scale factor chosen to ensure that $l_c(t)$ is an N(0,1) process under the null hypothesis. The last term in the equality defining $l_c(t)$ shows that it is the (scaled) moving average of the single event statistic $l(t)$ for white noise. The correlation length of $l(t)$ is half that for $l_c(t)$. Thus, as a time series, $l(t)$ has twice as many independent samples as does $l_c(t)$. Hence, we should anticipate that the expected maximum value for $l_c(t)$ is less than the expected maximum value for $l(t)$ for the same

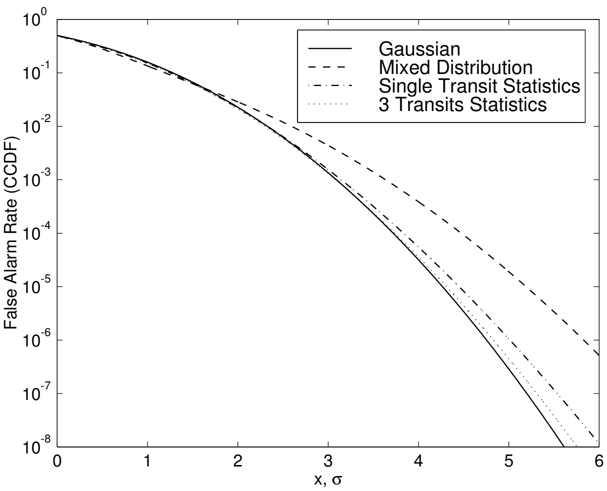

Figure 6.3: Sample and theoretical CCDF's (false alarm rates) as a function of threshold, x, for N(0,1) Gaussian noise (solid line), the mixed Gaussian distribution in the example in the text (dashed line), detection statistics for a single transit, l_1, in noise from the mixed distribution(dash-dotted line), and detection statistics for three transits, l_3, in noise from the mixed distribution (dotted line). The distributions from which l_1 and l_3 are drawn are 12-point and 36-point averages of samples from the mixed distribution, respectively. As more points in the mixed distribution are combined, the resulting distribution becomes more similar to a Gaussian one.

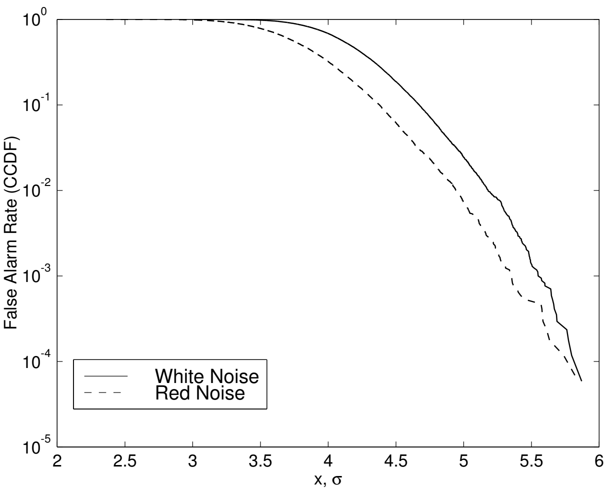

Figure 6.4: Sample CCDF's for search statistics for white and red Gaussian observational noise. The false alarm rate for red noise (dashed line) falls significantly faster than for white noise (solid line) as the threshold is increased. Thus, there are effectively fewer independent statistical tests conducted in searching the red noise sequence for transits than there are in searching the WGN sequence.

length observation. In fact, this should be true of any search for multiple transits as well, since multi-transit statistics are linear combinations of single-transit statistics. This is borne out by a numerical example in which a 4-week observation is considered with a sampling rate of 4 hr^{-1} and a search for three–hour transits with periods between 2 and 7 days is conducted. Figure 6.4 shows the CCDFs for both the red and white noise cases, demonstrating that the equivalent number of independent tests in conducting a full search is smaller for red colored noise than for white noise. That is not to say that it is easier to detect transits in colored noise. Although $N_{\rm EIT}$ is smaller, the scale factor k in effect reduces the SNR of a single transit by the same factor, making it more difficult to detect transits in colored noise with a correlation length comparable to a transit than it is for white noise.

As the assumption of white noise provides a conservative estimate for N$_{\rm max}$ in the case of red noise, let us consider WGN noise for the remainder of this section. Given the number of stars, N_{stars}, and the desired total number of false alarms, $N_{\rm FA}$, we set the threshold so that the single test false alarm rate is equal to $N_{\rm FA}/(N_{stars}\, N_{\rm EIT})$. Let us consider some limiting cases for the complementary distribution of the maximum test statistic. Suppose there is a signal $\hat{s}$ we test for in data set b such that $b = A\hat{s}$ and $\sigma = 1$. It follows that $l_{max} = A^{-1} b \cdot b = A^{-1}\sqrt{\sum_i b_i^2}$. This will be the case, or nearly so, if we test for all possible signals or for a large number of signals that are dense on the N_{points}–dimensional unit hypersphere underlying the observations. Consequently, the distribution of $\mathbf{l_{max}}$ would approach a χ–distribution with N_{points} degrees of freedom. This is the distribution for an incoherent matched filter or 'energy' detector (68) and explains its poor performance in comparison with a true matched filter. On the other hand, since the set of detection statistics for most transit searches is a complete set of vectors in the linear algebra sense, the distribution $\overline{F}_{l_{max}}(x)$ is bounded below by $\overline{F}_{N_{max}}(x;N_{points})$. The search for planetary transit trains in most cases, however, is a rather restricted class of possible signals compared to the set of all possible signals. We should expect it to asymptotically approach the distribution for $\overline{F}_{N_{\rm EIT}}(x)$ for some $N_{\rm EIT} > N_{points}$. While we do not supply a proof, we give several examples that demonstrate that $\mathbf{N_{max}}$ does, indeed, provide a good model for the distribution of $\mathbf{l_{max}}$ in the region of interest.

The algorithm for determining $N_{\rm EIT}$ is as follows

1. For the distribution of observational time steps, construct a synthetic data sequence composed of independent, identically distributed (i.i.d.) points drawn from a zero-mean unit variance WGN process.

2. Examine the maximum detection statistic obtained from this synthetic data set by applying the simple matched filter algorithm of eq 6.12: $l_{max} = \max_i \{x \cdot \hat{s}_i\}$ over the desired grid in the region of the period-phase duration parameter space of interest.

3. Repeat steps 1 and 2 a large number of times, at least several tens of the number of stars in the target sample.

4. Determine the number $N_{\rm EIT}$ of i.i.d. draws from a WGN process so that the complementary distribution of $\overline{F}_{N_{\rm EIT}}(x)$ matches the sample complementary distribution function of the set $\{l_{max}\}$ determined above at the point of interest $N_{\rm FA}/N_{stars}$ (equation 6.19).

Note that it is not necessary to determine the value of $N_{\rm EIT}$ to exquisite precision as the CCDF of $\mathbf{N_{max}}$ falls rapidly at the false alarm rates of interest to transit photometry campaigns. Even relative uncertainties of 50% can be tolerated in the estimate of $N_{\rm EIT}$. The remainder of this section is devoted to several examples drawn from actual or anticipated observations.

6.3.1 NASA Ames Vulcan Camera Observations

We first consider the case of collecting data for a ground-based system similar to the NASA Ames Vulcan Camera where 12 hours of data are obtained per night at 4 $\rm hr^{-1}$ over several weeks. Figure 6.5 shows the results of conducting the Monte Carlo experiment above on 1-, 3- and 6-week long sets of data, searching for planets with periods between 2 and 7 days. Over 10^5 trials were conducted for each data set. Taking $N_{\rm FA}$=1 and a sample of 5,000 stars, $N_{\rm EIT}$ is approximately 1,900, 24,000, and 79,000, for 1 week, 3 weeks and 6 weeks of data, respectively. Figure 6.6 shows how $N_{\rm EIT}$ evolves as a function of $\overline{F}_{l_{max}}$ for each case. Although the search space is the same for all three data sets, the longer the baseline, the greater the resolution in terms of discriminating between planets with similar periods, and hence, the greater the number of effective independent statistical tests. I.e., for longer data sets the correlation coefficient between one particular planetary signature and a second one drops off more rapidly as a function of period and phase as the parameters of the latter are varied from those of the former. Thus, the CCDFs, $\overline{F}_{l_{\rm max}}(x)$, for 1-week and 3-week long data sets 'roll over' at smaller values of x than does the CCDF for the 6-week long data set. Not only are the values of $N_{\rm EIT}$ smaller for shorter data sets, the threshold required for the same false alarm rate is smaller as well. The sample CCDFs appear quite 'ragged' at small values of $\overline{F}(x)$ because there are only a few samples available to estimate the behavior in the tail of the distribution. The number of trials performed to estimate the distribution must be high enough so that a reliable estimate for $N_{\rm EIT}$ can be obtained at the relevant single-search false alarm rate.

6.3.2 Multiple Season Observations

We next examine $N_{\rm EIT}$ for two 12–week seasons of Vulcan data and the Hipparcos data for HD 209458. The Hipparcos data consist of 89 points over 3 years' time, which is much sparser than the sampling for the Vulcan camera (> 2000 points per season). Figure 6.7 illustrates the difference in the behavior of

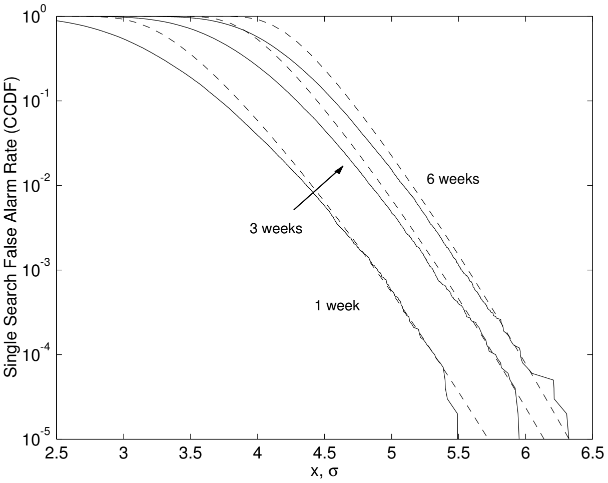

Figure 6.5: Sample CCDFs for planetary searches through 1, 3 and 6 weeks of Vulcan Camera data (solid curves) along with the theoretical curves for i.i.d. draws from a Gaussian process (dashed curves) that best match the empirical curves near a single search false alarm rate of 1 in 5,000. The effective number of independent tests performed in searching through data sets of these lengths, N_{EIT}, is approximately 1,900, 24,000, and 79,000, respectively.

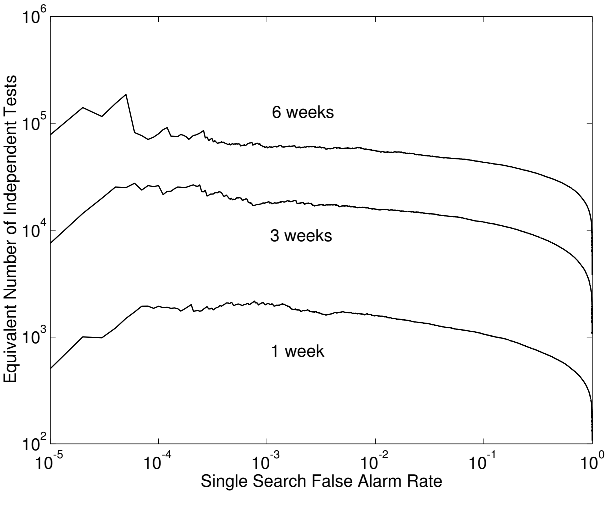

Figure 6.6: Equivalent number of independent tests for data similar to that collected by the Vulcan Camera as a function of the single search false alarm rate for observations lasting 1 week, 3 weeks and 6 weeks.

$\overline{F}_{l_{max}}$ for each data set. Over 10^6 trials were performed in each analysis. The Hipparcos data are so sparse that in searching for planets with periods from 2 to 7 days, the sample complementary distribution $\overline{F}_{l_{max}}$ is matched over a much shorter interval by $\overline{F}_{N_{\text{EIT}}}$ compared to the two seasons of Vulcan data (Fig. 6.7a). This is illustrated in panel Fig. 6.7b, where N_{EIT} is plotted versus the false alarm rate. At a single-search false alarm rate of 1/10,000, N_{EIT} is 110,000 for HD 209458, and is 790,000 for the Vulcan data. The Hipparcos data are so sparse that the signal space covered by the transit search is a significant fraction of the total surface of the 89-dimensional hypersphere underlying the signal vector space. Thus the CCDF rolls off much slower than that for $\overline{F}_{l_{max}}$ until rather small false alarm rates are reached.

6.3.3 The Proposed *Kepler Mission*

The proposed Discovery-class *Kepler Mission* would observe $> 100,000$ target stars in the Cygnus constellation continuously for at least 4 years at a sampling rate of 4 hr^{-1} (11). The goal of the mission is to determine the frequency and orbital characteristics of planets as small as Earth transiting Sun–like stars. The range of periods of greatest interest is from a few months to 2 years, with a range of transit durations from ~ 5 hr to 16 hr for central transits of planets with periods over this orbital range. The average transit duration is 8 hr over these periods, assuming a uniform distribution of periods. (Note that since the average chord length of a circle of unit diameter is $\pi/4$, the average duration of a transit is $\pi/4$ times the duration of a central transit which is 13 hours long at a period of 1 year.) We applied the N_{EIT} algorithm to examine the statistics of $\mathbf{l_{max}}$ for this experiment and to estimate N_{EIT}. Figure 6.8 shows the result for over 10^6 searches for 8–hr transits for orbital periods between 90 days and 2 yr, yielding $N_{EIT} \sim 1.7 \times 10^7$ for a single search. This agrees with the estimate obtained using Kent Culler's approach discussed in the introduction, and is no surprise as the assumptions for his method are met by this experiment. There is strong agreement between the theoretical curve and the empirical distribution of

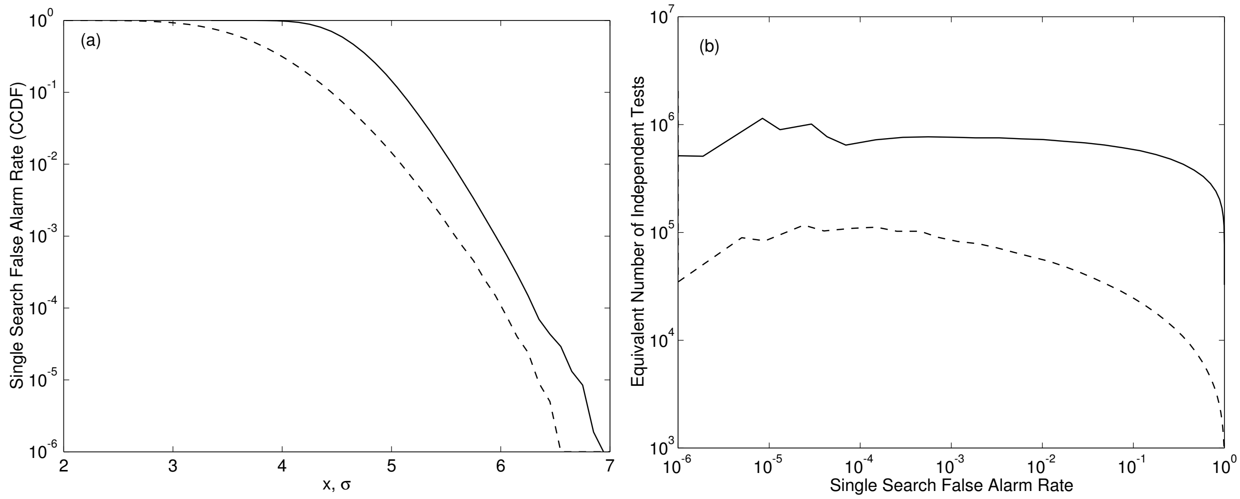

Figure 6.7: Analysis of multiple year data sets. Panel a) displays the CCDF's for the HD209458 data set and for two 12-week observations with the Vulcan Camera spaced one year apart. Although the Vulcan data consist of 4,000 points, while the HD 209458 data consist of only 89 points, the effective number of independent statistical tests conducted in searching two seasons of Vulcan data set is only 8 times more than that for the HD 209458 data set for a false alarm rate of 1 in 10^4. Panel a) illustrates that the slope of the CCDF for the Hipparcos data set (dashed curve) is much different than that for the Vulcan data (solid curve). Panel b) shows the evolution of N_{EIT} with false alarm rate corresponding to the Vulcan data (solid curve) and the Hipparcos data (dashed curve).

$\mathbf{l_{max}}$, even for false alarm rates as high as 0.1. Thus, we estimate that there are $\sim 1.7 \times 10^{12}$ independent statistical tests required in performing the desired search over 100,000 stars. The corresponding requisite single-test threshold is 7.1σ for no more than one expected false alarm for the entire campaign. The close agreement between the theoretical and the empirical curves most likely stems from the fact that the signals we are searching for are quite sparse on the unit-hypersphere underlying the 14,000-dimensional signal vector space for the simulations.

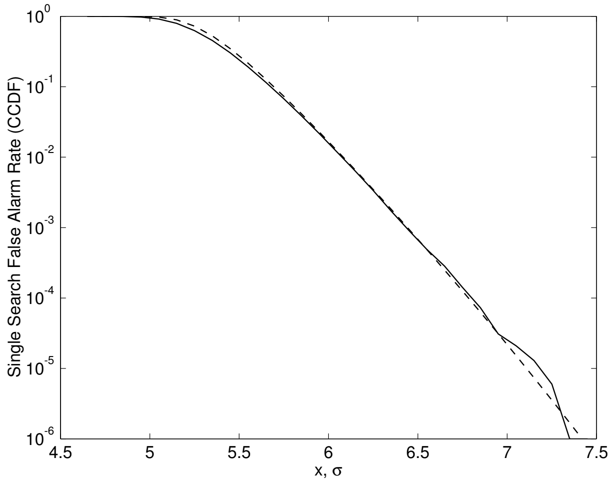

Figure 6.8: The sample CCDF for a 4–yr *Kepler Mission* searching for 8–hr transits for planets with orbital periods between 90 days and 2 yr (solid curve), along with the theoretical curve for the maximum of 17 million draws from an N(0,1) process (dashed curve).

Chapter 7

Detecting Transiting Planets

This chapter draws heavily on Jenkins (60), 'The Impact of Solar-Like Variability on the Detectability of Transiting Terrestrial Planets.' First the intrinsic variability of the Sun is examined and seen to be non-white and time-varying. This motivates the development of a wavelet-based, adaptive matched filter, for which the MATLAB source code is included in Appendix G. Folding the single event statistics to obtain multiple event statistics is described and FORTRAN source code to implement this is given in Appendix C. Monte Carlo techniques for estimating confidence levels in candidates whose detection statistics exceed the detection threshold are described, along with prototype MATLAB or FORTRAN code in Appendix F.

7.1 The DIARAD/SOHO Observations

In order to motivate the development of the adaptive matched filter will be discussed in §7.2, we describe the behavior of the Sun revealed by measurements made by the DIARAD instrument aboard the SOHO spacecraft. While we expect to observe significant diversity in stellar variability, we take the Sun's behavior as a proxy for all solar-like stars. DIARAD is a redundant, active-cavity radiometer aboard SOHO that measures the white-light irradiance from the Sun every 3 minutes (41). The second cavity is normally kept closed and is opened occasionally to calibrate the primary cavity, which ages throughout the mission with exposure to the Sun. The instrumental noise for a single 3 minute measurement is 0.1 W m^{-2} (Steven Dewitte 1999, personal communication). The DIARAD measurements considered here consist of 5.2 years of data that begin near solar minimum in January, 1996 and extend to March, 2001, just past solar maximum.

The data are not pristine: there are gaps in the data set, the largest of which lasts 104 days, and there are obvious outliers in the data. In particular, a set of 10 or 11 consecutive, anomalous points appears almost every 60 days. Each set begins with a point several W m^{-2} below the trendline, with the remaining 9 or 10 points lying approximately 6 W m^{-2} above the trend line. Nevertheless, the DIARAD time series is the most uniformly-sampled, lowest noise data set available. We've taken the liberty of removing the obvious outliers such as the ones occurring every 60 days, and a small number of isolated outliers that appear to occur randomly. We have not removed some of the data segments that appear to be corrupted in more subtle ways. An example of these is given by data on the edges of gaps in the data set, which often have atypically large slopes. Fully 83% of the data samples are available (62% of the missing points are represented by the three largest data gaps). For our purposes, a contiguous, completely sampled data set is highly desirable. This is mainly for computational convenience (to avoid division by 0 errors), and the filled-in points are largely neglected in addressing the detectability of transits against stellar variability. To that end, the missing points have been filled in by reflecting a segment on either side of each gap across the gap. We combine the two segments by taking the sum of each multiplied by a linear taper directly proportional to the distance from the closest edge of the gap. This procedure naturally preserves

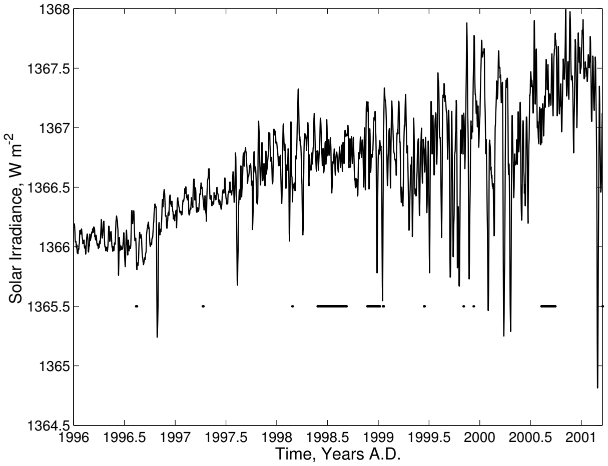

Figure 7.1: The time series of solar irradiance as measured by the DIARAD instrument aboard SOHO from January 1, 1996 through March, 2001, binned to 1 hr. Gaps of a day or longer are denoted by the horizontal segments at 1365.5 W m^{-2}.

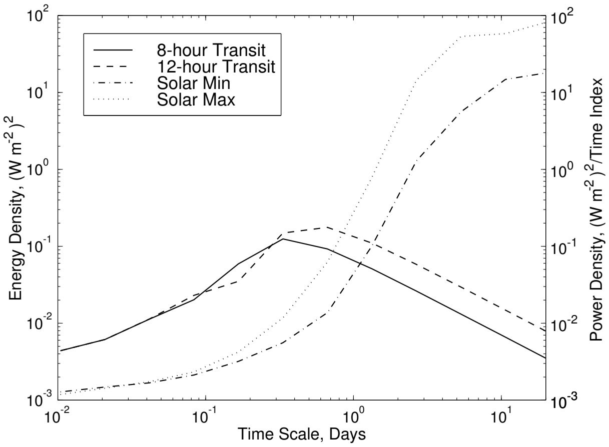

Figure 7.2: Distribution of power as a function of timescale from a wavelet analysis of the time series of solar irradiance as measured by the DIARAD instrument aboard SOHO for the years of 1996, near solar minimum (dash-dotted curve), and for 2000, near solar maximum (dotted curve). The timescale labeling is approximate, as no unique definition for it exists. The distribution of energy with timescale is also plotted for an Earth-size, 8-hr transit (solid curve) and for a 12-hr transit (dashed curve). The area under the transit curves and above the solar variability curves indicates that the transits are readily detectable against the solar variations.

continuity of the data, and preserves the correlation structure to a large degree. Some smoothing of the small scale structure occurs, however, as the procedure takes the average of two segments of a noise process. We've adjusted the filled-in data to reduce the amount of smoothing using a technique described in Jenkins (60).

Figure 7.1 shows the DIARAD time series, binned to 1 hr. Filled-in gaps of at least a day in duration are denoted by the horizontal line segments at 1365.5 W m^{-2}. The average solar flux during the 5.2 yr of observation is 1366.6 W m^{-2}. Note that on this scale, an Earth-sized transit (84 ppm) is 0.115 W m^{-2}. The sample standard deviation of the data set is 0.5 W m^{-2}. This would seem to imply that detecting Earth-sized transiting planets might be a terribly difficult, if not impossible task. The solar variability is not a white noise process, however, and most of the noise power occurs on very long timescales compared to the duration of a central transit of planets with orbital periods up to 2 yr about a solar-like star (2-16 hr). This is made clear by Figure 7.2 which exhibits the power of the DIARAD time series as a function of timescale near solar minimum (1996) and near solar maximum (2000) along with the energy at each timescale for Earth-size, 8-hr and 12-hr transits. These curves were obtained by a wavelet analysis described in §7.2. Note that at time scales shorter than 1 day, the ratio of the transit energy to the power of the solar time series is much greater than 1. This indicates that transits of Earth-sized planets are highly detectable against solar-like variability, with low-intrinsic noise, space-based observations.

Two important qualities are revealed by this examination of the DIARAD/*SOHO* observations: i) solar variability is *not* white, and ii) solar variability is *not* stationary. Any detection scheme which has pretenses of "optimality" or "efficiency" must take these two crucial characteristics into account or face suboptimal performance, and possibly outright failure. The approach baselined for detection of transiting planets in *Kepler's* data set is an adaptive, wavelet-based matched filter. This detector performs a joint time-frequency decomposition of the data to estimate the properties of the noise as a function of time, and then applies a matched filter to the "whitened" data in the wavelet domain, taking into account the effect of the whitening filter on the shape of a transit pulse. This filter is developed in detail in the following section.

7.2 An Adaptive Wavelet-Based Matched Filter

Chapter 6 introduced detection theory for the problem of detecting a known signal in additive WGN, and then discussing the issues of non-white, non-stationary Gaussian noise. There it was shown that for non-white, *stationary* noise, an optimal detector can be formed in the frequency domain using an estimate of the PSD of the measurement noise (including intrinsic stellar variability), as per Kay (67). Here we argue that for time-varying noise, especially noise processes with steep spectral slopes, an explicit time-frequency representation is desirable.

Kay (67) suggested an adaptive matched filter based on equation 6.11 using a smoothed periodogram to estimate $P(\omega)$. This approach is fine for noise processes that are weakly-colored or white, but not for 1/f-type processes such as solar variability. Simply smoothing the periodogram with a moving average filter tends to reduce the apparent spectral slope of these processes significantly, yielding an inaccurate power spectrum estimate. Alternatively, Kay's method may be modified by using multitaper spectrum approaches to estimate the noise power spectrum, minimizing the "leakage" of the effective data window. Several choices for tapers are available, including sinusoidal families (97) which approximate optimal tapers minimizing the asymptotic bias of the estimate. Alternatively, prolate spheroidal sequences are widely acknowledged to yield optimal spectrum estimates minimizing the spectral leakage outside a given resolution bandwidth, and have been used with great success to examine p-mode oscillations in the solar power spectrum (see, e. g., 109). While good results can be obtained using a modification of Kay's approach, there are computational issues to consider. The length of the window used to estimate the periodogram must be chosen in some way, as well as the number of adjacent data segments to be used to provide additional smoothing of the power spectrum estimate. Moreover, the sensitivity of the detector to a transit-like signal depends on the location of the transit pulse within the window. It would seem that for the best results, a periodogram centered at each possible transit location needs to be computed, further increasing the computational burden. We propose a wavelet based approach using an overcomplete wavelet transform (OWT) of the data and the signal to be detected. The wavelet domain is a natural one for designing time-varying filters since it *is* a joint time-frequency representation of a waveform. In addition, the overcomplete wavelet expansion admits a filterbank implementation with a direct interpretation in terms of equation 6.11. As such, the properties of Kay's adaptive detector should hold for the detector described here; namely that the detector would be asymptotically efficient (ideal) if an independent realization of the noise process were available.

First, let's review wavelets briefly. A wavelet transform is similar to the Fourier transform in that the wavelet coefficients are the projection of the original data vector onto a set of basis functions. In the case of wavelets, however, the basis functions are concentrated in time as well as in frequency. Moreover, unlike the Fourier basis, there is an infinity of possible choices for wavelet bases that trade off resolution in frequency for resolution in time. (This also implies that there is not a unique definition of the term "time scale" for wavelet transforms as there is for "frequency" for the Fourier transform.) The first orthogonal non-trivial wavelets were obtained by Debauchies (29) who was interested in obtaining a continuous wavelet transform through iterations of a discrete time algorithm. Somewhat earlier, however, Smith & Barnwell (106) succeeded in designing critically sampled, perfect reconstruction, octave band filter banks. Debauchies' wavelets are special cases of those filters meeting the conditions specified by Smith and Barnwell, such that the limiting process is a continuous time wavelet transform. The methodology we adopt is based on a filterbank implementation of an overcomplete discrete-time wavelet transform. Hence, we'll approach the subject from the viewpoint of filterbanks as per Vetterli & Kovačević (108).

Figure 7.3 shows a dyadic, critically sampled filter bank. In the first stage in the process, the time series $x(n)$ is separated into two channels by filters with responses $H_L(\omega)$ and $H_H(\omega)$. Each filtered signal component is then downsampled by a factor of two (essentially, every other sample is discarded). The highpass signal, $x_1(n)$ is not subjected to further filtering in the analysis section. The lowpass signal, however, is treated in an identical manner as its predecessor, $x(n)$, and the process is iterated $M-1$ times, for a total of M output channels. For our purposes, all we need to know is that $H_L(\omega)$ is a lowpass filter and that $H_H(\omega)$ is a highpass filter, and that these filters isolate complementary frequency components of the time series $x(n)$. Corresponding to $H_L(\omega)$ and $H_H(\omega)$ are reconstruction filters $G_L(\omega)$ and $G_H(\omega)$ such that the signal $x(n)$ is exactly equal to its reconstruction $\hat{x}(n)$. The equivalent filter for each channel in Figure 7.3 can be determined explicitly, and we'll refer to these filters from the highest center frequency to the lowest as $\{h_1(n), h_2(n), \ldots, h_M(n)\}$. The output signals corresponding to these filters will be designated $\{x_1(n), x_2(n), \ldots, x_M(n)\}$, respectively.

Figure 7.4 shows the frequency response of each filter in a filterbank implementation of a discrete-time wavelet expansion of a time series out to M=16. Panel *a* shows the frequency axis on a linear scale, while panel *b* is plotted with a log scale for the frequency axis. The filters enjoy a "constant-Q" property. That is, the quality factor (Q) defined to be the ratio of the center frequency of a bandpass filter to its full width at half maximum, is constant for all but the final filter. In the following analysis, we omit the decimation operators ('$\downarrow 2$') in Figure 7.3a, and replace each filter following a decimation operator with the result of upsampling it by 2 (i. e., we retain the same effective filters as those of the critically sampled filter bank). This leads to an overcomplete wavelet expansion of a filtered time series. The price we pay is that the representation is highly redundant, increasing the computational burden, since we must now filter the samples discarded in the critically sampled implementation. The gain achieved is the shift invariance of the OWT of a time series. Therefore, the OWT of the convolution of two time series is the same as convolving the OWT coefficients of one time series at each scale with the corresponding coefficients of the other time series. This is not the case for the critically sampled WT. To make this explicit, let

$$\mathbb{W}\{x(n)\} = \{x_1(n),\, x_2(n),\, \ldots,\, x_M(n)\}, \tag{7.1}$$

be the overcomplete wavelet transform of $x(n)$ where

$$x_i(n) = h_i(n) * x(n),\; i = 1,\, 2,\, \ldots,\, M, \tag{7.2}$$

and '$*$' denotes convolution. Then we have

$$\mathbb{W}\{x(n) * y(n)\} \;=\; \{x_i(n) * y_i(n)\}_{i=1,\ldots,M}. \tag{7.3}$$

A remark is in order regarding the implementation of the decimated, discrete time-wavelet transform. Normally, in order to ensure that the number of output points equals the number of input points, the convolutions performed on the data set are circular. In other words, the signal vector is treated as if it were periodic with period N, the length of the data set. If N is a power of 2, then the convolutions can be performed efficiently with FFTs. We adopt this convention as well, applying it to the overcomplete discrete-time wavelet transform such that each $x_i(n),\; i = 1,\, \ldots,\, M$ is an N-point sequence. Moreover, we will not distinguish between circular and non-circular convolution unless there is a reason to do so (i. e., a relationship holds for one but not the other).

One additional property is required before we can obtain a wavelet-based expression for a matched filter. We need to be able to express the dot product between two vectors in the wavelet domain. For an overcomplete, dyadic wavelet expansion, the following relationship holds:

$$x(n) \cdot y(n) = \sum_{i=1}^{M} 2^{-\min(i,M-1)}\, x_i(n) \cdot y_i(n), \tag{7.4}$$

where $x(n)$ and $y(n)$ are time series. The restriction of the power of 2 in equation 7.4 is necessary because

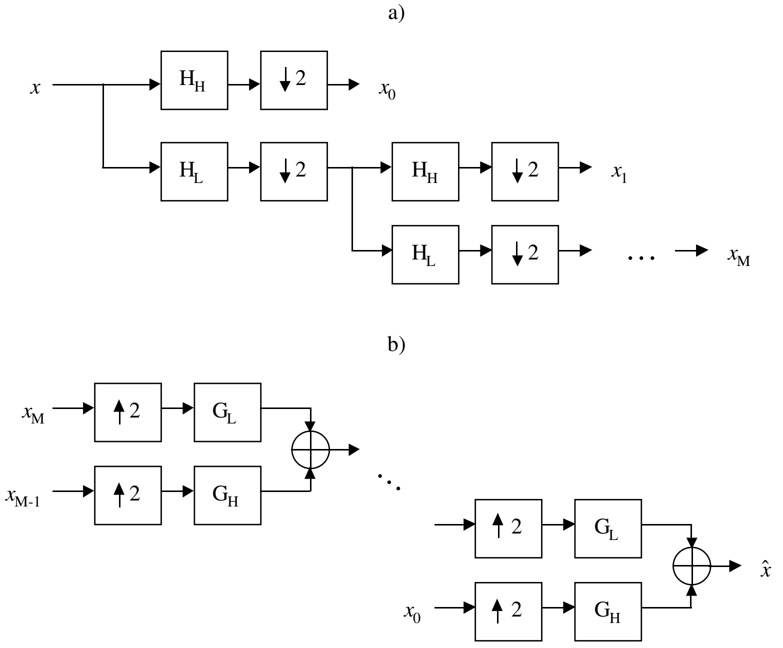

Figure 7.3: Block diagram of a filterbank implementation of a critically-sampled, discrete-time wavelet expansion of a time series. Panel *a* shows the analysis section which partitions a time series into different channels with complementary passbands. Panel *b* illustrates the synthesis section which reconstructs the original time series from the set of channels.

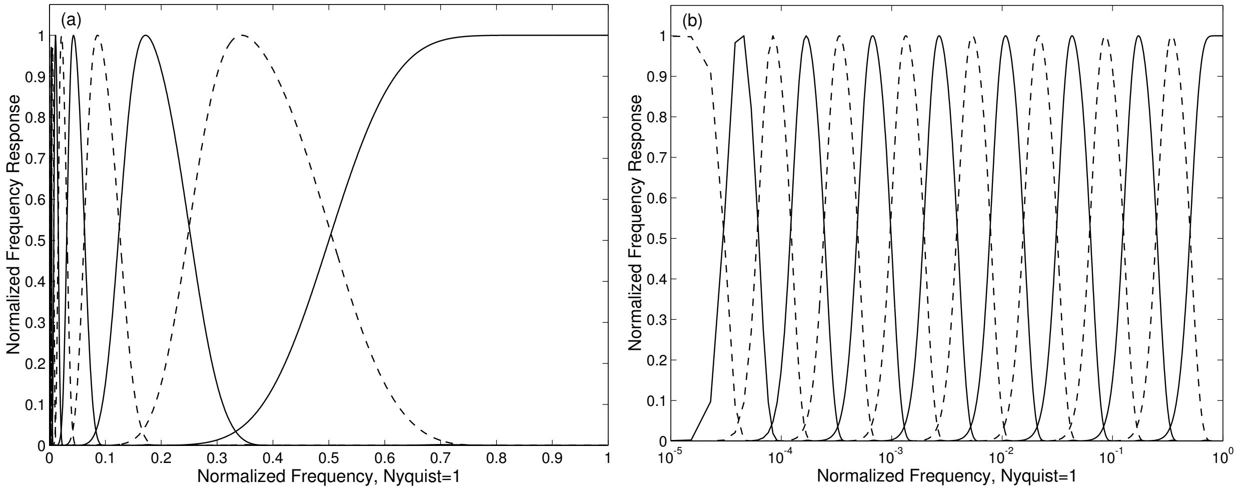

Figure 7.4: Frequency response of the filters in a filterbank implementation of a discrete time-wavelet expansion of a time series using Debauchies' 12-tap filter. Panel *a* shows the frequency responses on a linear frequency scale. Panel *b* has a logarithmic frequency scale, illustrating the "constant-Q" property of an octave-band wavelet analysis.

the last two channels of the OWT have the same bandwidth. Equation 7.4 can be established from Parseval's relation for tight frames (108). This result, in turn, should agree with our intuition, as each time we iterate the dyadic filterbank of Figure 7.3, we double the number of samples representing the lowpass channel output from the previous iteration.

We are now in the position to recast equation 6.11 in terms of the overcomplete wavelet expansion. The whitening filter is implemented by simply scaling each channel of the filter bank by a time-varying value inversely proportional to the local standard deviation of the data in that channel. The bandwidth in the channel helps determine the time frame over which the standard deviation is estimated. If the window is K points long for the smallest scale, then it should be $2^{i-1}K$ for the i^{th} channel. The window should also be much longer than the transit duration of interest so that it will not itself be perturbed by a transit, thereby reducing the detectability of transits. On the other hand, the window should be kept short enough to track changes in the statistics of the underlying observational noise. Empirically we find that a window length 10 times the duration of a transit works well. The detection statistic, then, is computed by multiplying the whitened wavelet coefficients of the data by the whitened wavelet coefficients of the transit pulse, and then applying equation 7.4:

$$\begin{aligned} T &= \frac{\tilde{\mathbf{x}}\cdot\tilde{\mathbf{s}}}{\sqrt{\tilde{\mathbf{s}}\cdot\tilde{\mathbf{s}}}} \\ &= \frac{\sum_{i=1}^{M} 2^{-\min(i,M-1)} \sum_{n=1}^{N} [x_i(n)/\hat{\sigma}_i(n)]\,[s_i(n)/\hat{\sigma}_i(n)]}{\sqrt{\sum_{i=1}^{M} 2^{-\min(i,M-1)} \sum_{n=1}^{N} s_i^2(n)/\hat{\sigma}_i^2(n)}}. \end{aligned} \tag{7.5}$$

The time-varying channel variance estimates, σ_i^2, are given by

$$\hat{\sigma}_i^2(n) = \frac{1}{2^i K+1} \sum_{k=n-2^{i-1}K}^{n+2^{i-1}K} x_i^2(k),\; i=1,\;\ldots,\;M, \tag{7.6}$$

where each component $x_i(n)$ is periodically extended in the usual fashion and $2K+1$ is the length of the variance estimation window for the shortest time scale.

The structure of the OWT is exceptionally convenient as it permits the efficient calculation of T for a transit pulse at any location. Note that equation 7.6 implies that the whitening coefficients are determined solely by $x(n)$, regardless of the assumed location of a transit signal. Thus, to compute T for a given transit pulse centered at all possible time steps, we simply "doubly whiten" $\mathbb{W}\{x(n)\}$ (i. e., divide it point-wise by $\hat{\sigma}_i^2(n)$), correlate the results with $\mathbb{W}\{s(n)\}$, and apply the dot product relation, performing the analogous operations for the denominator, noting that $\hat{\sigma}_i^{-2}(n)$ is itself a time series:

$$\begin{aligned} T(n) &= \frac{\mathbb{N}(n)}{\sqrt{\mathbb{D}(n)}} \\ &= \frac{\sum_{i=1}^{M} 2^{-\min(i,M-1)}\,[x_i(n)/\hat{\sigma}_i^2(n)] * s_i(-n)}{\sqrt{\sum_{i=1}^{M} 2^{-\min(i,M-1)}\,\hat{\sigma}_i^{-2}(n) * s_i^2(-n)}}. \end{aligned} \tag{7.7}$$

Note that the '$-$' in $s_i(-n)$ indicates time reversal. The terms $\mathbb{N}(n)$ and $\mathbb{D}(n)$ are introduced for convenience later on.

Recall at this point the form of Kay's adaptive detector (equation 6.11), and the partitioning of power in each channel by the filterbank implementation of the OWT (Figure 7.4). Rather than estimating the power spectrum of the noise with a uniform moving average, equations 7.5 and 7.7 estimate $P(\omega)$ by partitioning the frequency domain into non-uniform intervals that increase in width logarithmically from the baseband. They then average the power in each channel over a time interval proportional to the inverse of the width of the channel. Clearly, an analogous operation could be carried out using periodograms rather than a wavelet transform. The efficiency of the structure of the OWT, however, provides a compelling reason not to do so. Moreover, the OWT allows one to estimate the channel variances with windows of differing lengths, an option not available with periodograms. Equation 7.7 forms the basis for the adaptive matched filter applied throughout the remainder of this paper. For the purposes of examining the detectability of transits against solar-like variability, however, we need only compute the expected detection statistic $< T >$

or the S/N via

$$\langle T(n)\rangle = \sqrt{\sum_{i=1}^{M} 2^{-\min(i,M-1)}\ \hat{\sigma}_i^{-2}(n) * s_i^2(-n)}, \qquad (7.8)$$

which holds so long as the analysis windows used to estimate $\hat{\sigma}_i^{-2}(n)$ are sufficiently long. This can be verified by examining the change in the detection statistics when $\left\langle x_i^2(n)\right\rangle$ is calculated with and without the presence of transits.

Figure 7.5 shows a flowchart for the construction of the single event statistics using the wavelet-based matched filter.

Finally, we note how to combine the components of individual detection statistics to form multiple event statistics. Suppose we wish to test for transits at widely spaced locations $A \subset \{1,\ldots,N\}$. The total detection statistic is given by:

$$T_A = \sum_{i\in A} \mathbb{N}(i) \Big/ \sqrt{\sum_{i\in A} \mathbb{D}(i)}, \qquad (7.9)$$

where $\mathbb{N}$ and $\mathbb{D}$ are as in equation 7.7. Hence, T_A can be determined from the components of the single transit statistics at each individual transit location. The next section presents the results of our analysis of the DIARAD data set using this analysis technique.

7.3 Performance Prediction Results

In this section we present the results of using the DIARAD/SOHO data to predict the expected performance of *Kepler*, a recently selected Discovery Mission designed to detect Earth-size planets orbiting solar-like stars in the circumstellar habitable zone. *Kepler* will observe $> 100,000$ target stars in the Cygnus constellation continuously for at least four years at a sampling rate of 4 hr^{-1} (11). For detecting Earth-size planets, the spectral types of the target stars span F7 through K4. The range of planetary periods of greatest interest is from a few months to 2 years, with a corresponding range of central transit durations from $\sim$ 5 hr to 16 hr. The average transit duration is 10.1 hr for a uniform distribution of orbital periods over this range. (Note that since the average chord length[1] of a circle of unit diameter is $\pi/4$, the average duration of a transit is $\pi/4$ times the duration of a central transit, which is 13 hours at an orbital period of one year. The average central transit duration over these periods happens to be $\sim$13 hours, too. Moreover, 50% of transits are longer than 11.3 hours.) The total number of effective independent tests to be performed in searching the light curves of 100,000 stars for transiting planets with orbital periods in this range is $\approx 2\times10^{12}$ (62). Assuming Gaussian statistics, a detection threshold of $\sim 7\sigma$ is required to control the total number of expected false alarms below 1 for the entire experiment. At this threshold, if the mean S/N of a set of transits is $\geq 8\sigma$, a detection rate of $\geq$84% will be achieved. As the total S/N is proportional to the square root of the number of transits, a single event S/N of 4σ suffices for each of a set of four transits (for a one year orbit). This is a conservative requirement. It can easily be argued that the 50% detection rate achievable at a single event S/N of 3.5σ would yield a statistically significant sample of detections (or non-detections) given 100,000 target stars in the survey.

Kepler's aperture is 0.95 m allowing $5.75\times10^9\ e^-$ to be collected every 6.5 hr for a G2, $m_v = 12$ dwarf star for a shot noise of 13 ppm. The instrument noise should be $\sim$ 6 ppm over this same duration. This value is based on extensive laboratory tests, numerical studies and modeling of the *Kepler* spacecraft and photometer (70; 64; 94). The values in Table 3 of Koch et al. (70) support this level of instrumental noise from a high-fidelity hardware simulation of *Kepler's* environment, while the numerical studies of Remund et al. (94) are based on a detailed instrumental model. This model includes terms such as dark current, read noise, amplifier and electronics noise sources, quantization noise, spacecraft jit-

[1] Here we must be clear about how "random" chords are generated. For circular orbits, the sole parameter determining whether a planet transits or not is orbital inclination, i. Assuming that i is uniformly distributed implies that the distance of chords from the center of the stellar disc for transiting planets, a, is also uniformly distributed. The average chord length, $\bar{c}$, of chords constructed in this manner for a unit-diameter disc, is then $\int_0^{1/2} 2\sqrt{1/4-a^2}\ da \big/ \int_0^{1/2} da$ or $\pi/4$, giving the ratio of $\bar{c}$ to the maximum chord length, 1, as $\pi/4$.

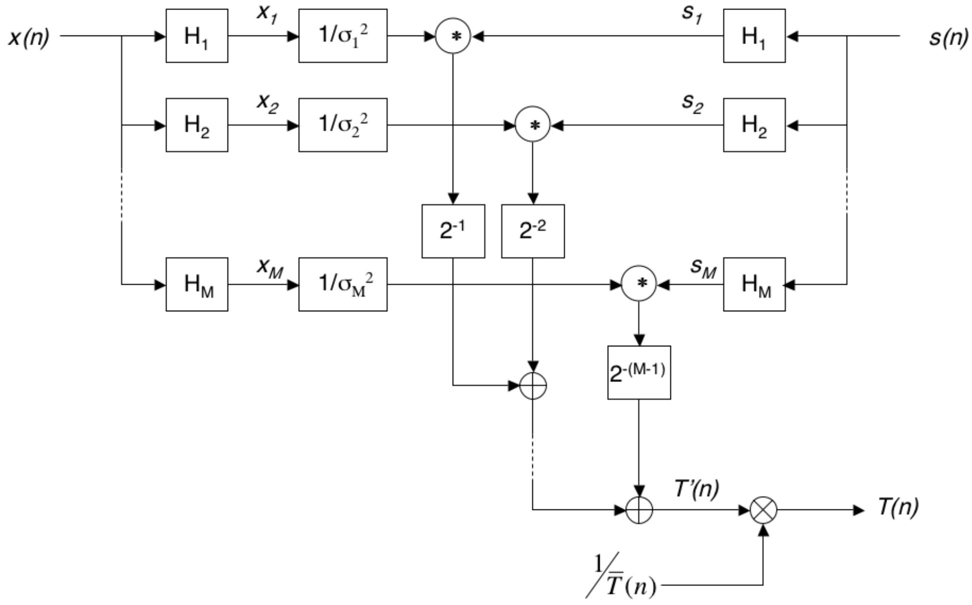

Figure 7.5: Flowchart showing the construction of single event statistics for the wavelet-based, matched filter presented in the text.

ter noise, noise from the shutterless readout, and the effects of charge transfer efficiency. To simulate the combined effects of the shot noise and instrumental noise for *Kepler*, a WGN sequence was added to the DIARAD time series with a standard deviation equal to the square root of the combined shot and instrumental variance for an m$_v$=12 star less the square of the DIARAD instrumental uncertainty ($0.1\ W\ m^{-2}$ in each 3 min DIARAD measurement). The DIARAD instrumental variance is $\sim 1/4$ the combined shot and instrumental variance for one of *Kepler's* m$_v$=12 stars. Prior to applying the techniques of Chapter 6, it was necessary to extend the length of the time series to a power of two (from $\sim 2^{17.47}$ to 2^{18} points). The time series was 'periodically' extended by reflecting segments at the beginning and end of the original time series across the imaginary gap from the end to 2^{18}. Both reflected segments were tapered and added together much in the same fashion as the missing points were filled in as described in Jenkins (60). In addition, to compensate for the smoothing nature of the fill-in procedure, we computed the critically sampled WT of the extended time series and examined the local variances of the wavelet coefficients. The variances of the filled-in points were adjusted to match the variances of the points at the edges of the gaps, with a linear transition from one value to the next. This procedure was applied to each wavelet scale so long as the mean variance of the filled-in points was significantly below that of the original points. These procedures minimize edge effects attendant in performing a circular WT of a time series containing data gaps. In an actual search, care needs to be exercised near the edges of any data gaps. Any candidates with transits near data gaps should be scrutinized carefully to eliminate false positives due to edge effects.

The *Kepler* Mission should not suffer from large time gaps. Roll maneuvers are planned about every 90 days to reorient the sunshade and the solar panels as the Sun would otherwise appear to revolve about the spacecraft every year. A twenty-four hour period has been budgeted for thermal stability to be achieved after each roll and for nominal science operations to re-commence. We assume that transits cannot be found within 12 hours prior to the roll maneuver and for 12 hours after thermal stability is achieved. The lost data amounts to $\sim$2% of the total, implying that about 2% of all transits occurring during the mission will be missed. This represents an insignificant impact on the science return as the detection of a planet does not depend on observing a set of consecutive transits. Moreover, the missing phase space can be filled in by extending the mission by about 2% or one month beyond the nominal four years.

The OWT of the extended synthetic time series and that of a single transit were computed using Debauchies' 12-tap discrete wavelet filter (29). Equation 7.8 was applied to transits of 6.5-hr duration and 13-hr duration with depths of 84 ppm (0.115 $W\ m^{-2}$) corresponding to an Earth-size transit of a solar-like star. Note that we have not included limb-darkening in the simulated transits: they are simply rectangular pulses. This is a conservative approach. Limb-darkening increases the depth of non-grazing transits, providing higher total signal energy for transits with duration longer than 82% of a central transit (which holds for more than 57% of all transits). Also, limb-darkening concentrates the energy of a transit into a shorter time period. Both of these effects increase the S/N of a transit signal and increase its detectability against solar-like variability, which exhibits less power at shorter time scales. Throughout this discussion we ignore S/N's calculated for filled-in points in the DIARAD data or from points within a day of gaps at least as long as a day. Filled-in points do influence the results of nearby non-filled-in points since they are included in the calculation of local variance estimates of other points (see equation 7.6). Their influence is reduced by the compensation scheme described earlier. Figure 7.6 shows the results as a function of time throughout the 5.2-year DIARAD data record. Note that the S/N of a 13-hr transit is significantly higher than that of a 6.5-hr transit at the beginning of the data record near solar minimum ($\sim 5.7\sigma$ vs. $\sim 4.9\sigma$), but that it is nearly the same at the end of the record near solar maximum ($\sim 4.25\sigma$ vs. $\sim 4\sigma$). This is a consequence of the movement of noise power towards shorter time scales as solar max is approached (see figure 7.2). Another way to interpret the S/N's plotted in this figure is to examine the equivalent total noise, or combined differential photometric precision (CDPP), in a time interval equal to the duration of the transit. This is easily computed by dividing the transit depth (84 ppm) by the S/N. Figure 7.7 shows the result of this calculation for the 6.5-hr and 13-hr transits. As the desired total noise for *Kepler* is to have no more than 21 ppm for the total noise budget at 6.5 hr (for an m_v=12 star), it's clear that this requirement is met with significant margin over most of the data record. Since transit photometry campaigns search for sequences of transits, it is the mean S/N that is of interest, not the S/N of any particular transit. These calculations were extended to cover transits of durations .25 hr to 20 hr. Figures 7.8 and 7.9 present contour maps of the S/N and equivalent total noise over the course of the DIARAD observations with instrumental and shot noise expected for *Kepler*. The S/N's allow *Kepler* to detect Earth-size planets exhibiting four transits longer than $\sim$ 5 hr for m_v=12 stars.

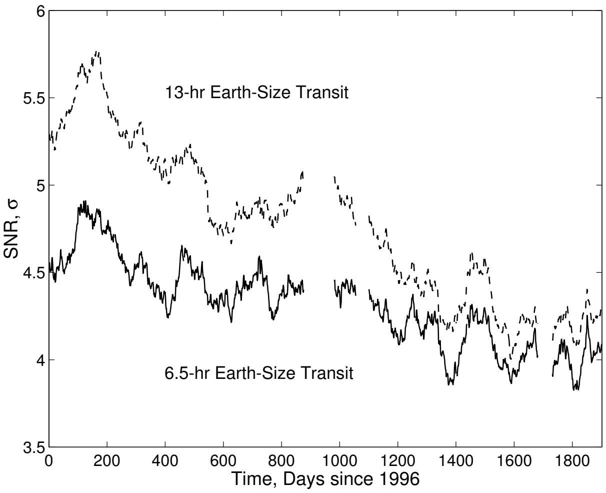

Figure 7.6: Estimated S/N's (in σ) for 6.5-hr transits and 13-hr transits of Earth-size planets orbiting Sun-like stars over half a solar-like cycle. Values of S/N greater than 4σ indicate a detection rate exceeding 84% for four transits.

We note that minimum detectable planet radius is not particularly sensitive to the single event S/N as this is proportional to the square of the planetary radius. To illustrate this, we extend the calculations above to stars of magnitude other than m_v = 12. The uncertainty of the DIARAD time series is equivalent to the combined shot and instrumental noise of a m_v = 10.4 star. To simulate data from stars brighter

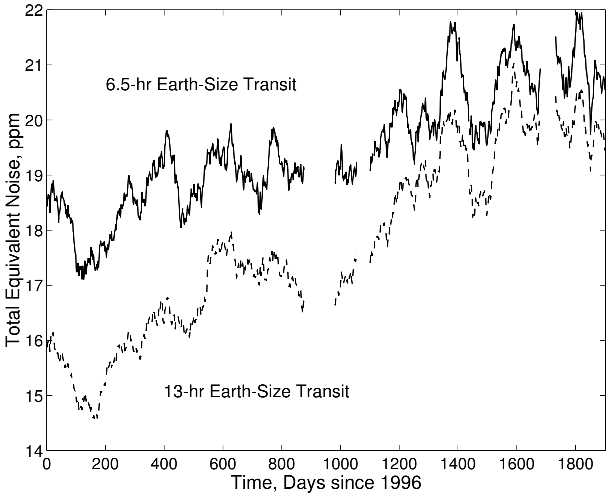

Figure 7.7: Equivalent total noise (in ppm) for 6.5- and 13-hr transits of Earth-size planets orbiting Sun-like stars over half a solar-like cycle. *Kepler's* total noise budget is set to no more than 21 ppm at a time scale of 6.5 hr, including stellar variability (i. e., 4σ for an 84 ppm Earth-size transit). This requirement is met with significant margin on average for the noise environment expected for *Kepler*.

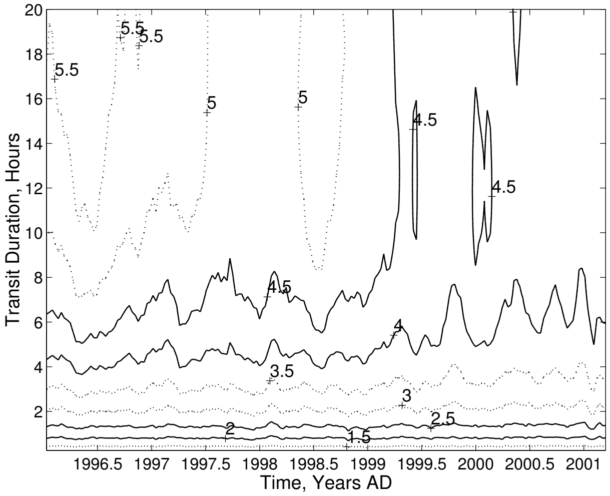

Figure 7.8: Contour map of estimated S/N's (in σ) for single transits of Earth-size planets orbiting Sun-like stars with durations from 0.25 to 20 hours. Four or more Earth-size transits longer than $\sim$ 5 hr are detectable $\geq$84% of the time.

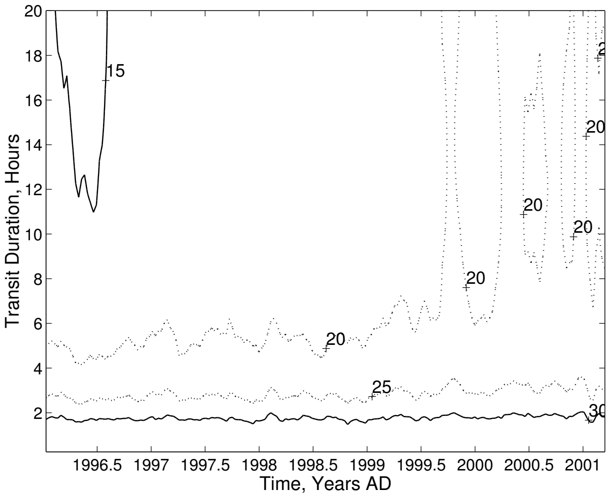

Figure 7.9: Contour map of the equivalent total noise (in ppm) as a function of transit duration (or time interval) for Earth-size transits with shot and instrumental noise appropriate for the *Kepler* Mission.

than this required "denoising" the DIARAD time series to remove the instrumental noise. To do this, we multiplied each channel of the decimated WT of the 15-min binned DIARAD time series by a scalar equal to the square root of the ratio of the sample variance less the DIARAD instrumental variance to the sample variance, and then transformed the result back into the time domain. This operation is essentially a Wiener filter implemented in the wavelet domain. Noise sequences representing a combination of shot noise and *Kepler* instrumental noise were then added to the "denoised" time series to simulate data from stars of different magnitudes. The sample variances of the first few channels are actually slightly less than the reported measurement uncertainties. We believe that this is likely the result of the measurement-replacement procedure we used. Alternatively, it may be due in part to an overly conservative estimate of the instrument sensitivity by the DIARAD science team. In any case, the difference between the reported variance and the actual sample variance is small. At the point design for a m_v = 12 star, the difference is relatively insignificant since the shot noise for such a star is well above the reported DIARAD measurement uncertainty. For the first sev-

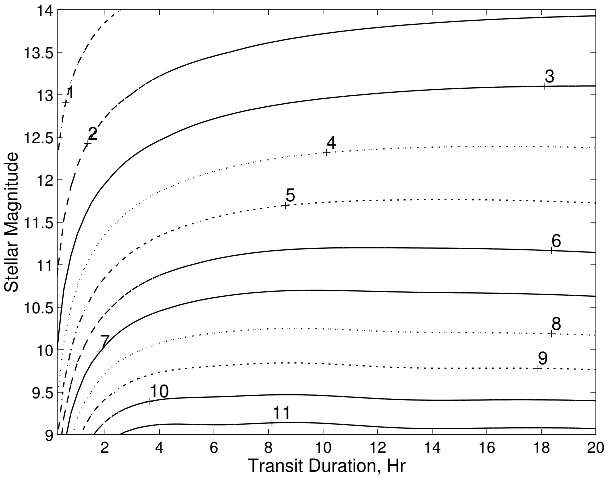

Figure 7.10: Contour map of the Earth-size, single transit S/N (in σ) as a function of stellar magnitude and transit duration. The range of stellar magnitudes corresponds to the range for *Kepler's* target stars.

eral channels (short time scales), then, we simply set the scalar to zero when the operation given above yielded an imaginary number. This is in one respect a conservative approach as it places more noise in these channels than in the original time series for a given magnitude star.

Figure 7.10 shows a contour map of the Earth size, single transit S/N as a function of stellar magnitude and transit duration. We obtain S/N's as high as $11\,\sigma$ for $m_v = 9$ stars while S/N's as low as $1\,\sigma$ are obtained at $m_v = 14$ for transits longer than 2.5 hr. Values for the minimum detectable planetary radius at an 84% detection rate for four and for six transits are given in the contour maps of Figure 7.11. This figure demonstrate that planets significantly smaller than Earth can be found by *Kepler*. For example, at $m_v = 10$ and for four transits, planets with radii as small as 0.7 $R_\oplus$ are detectable (0.5 Earth areas). With six transits, planets with radii as small as 0.6 $R_\oplus$ (0.36 Earth areas) are detectable. Additionally, for cases exhibiting six transits, planets as small as 1.0 $R_\oplus$ can be detected orbiting stars as dim as $m_v = 12.7$. Keep in mind that this is for a detection rate of 84%. Planets smaller than these are still detectable at lower detection rates.

Finally, we use the DIARAD time series to esti-

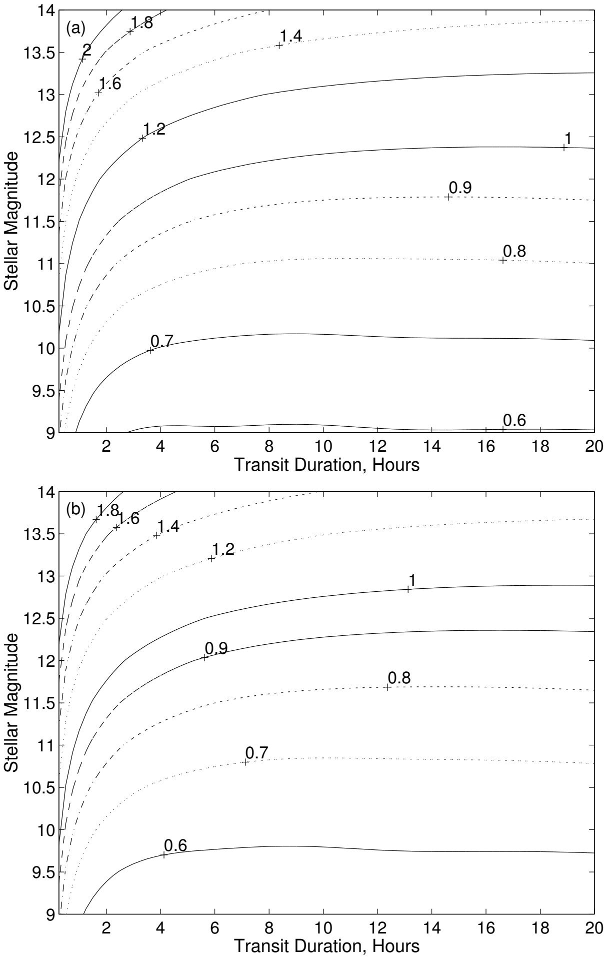

Figure 7.11: Contour map of the minimum detectable planetary radius ($R_\oplus = 1$) at the 84% detection rate as a function of stellar magnitude and transit duration for planets exhibiting four transits (panel *a*) and for six transits (panel *b*). At $m_v = 10$ and for four transits, planets with radii as small as 0.7 $R_\oplus$ are detectable (0.5 Earth areas). With six transits, planets with radii smaller than 0.6 $R_\oplus$ (0.36 Earth areas) are detectable.

mate the effect of stellar rotation period on the detectability of terrestrial planets. Batalha, et al. (7) estimate that 65% of *Kepler's* target stars (F7-K9) are sufficiently old to have spun down to rotation periods $\geq$ 20 days. The question is, how is the detectability of transits affected by rotation periods experienced by the majority of these target stars? Ground-based observations show that solar-type stars rotating faster than the Sun are more magnetically active, increasing the photometric variability over a range of time scales. These observations provide an indication of the appropriate scaling relation to use on time scales $>$ 1 day. Figure 7 of Radick et al. (93) indicates that photometric variability, σ_{phot}, on time scales shorter than a year is related to the chromospheric activity level parameter, $\mathrm{R}'_{\mathrm{HK}}$, by a power law with exponent 1.5. Other observations (84) suggest that $\mathrm{R}'_{\mathrm{HK}}$ is approximately inversely proportional to stellar rotation period, P_{rot}, so that

$$\sigma_{phot} \approx P_{rot}^{-1.5}. \tag{7.10}$$

What these ground-based studies do not provide, however, is the relation between rotation period and photometric variability on time scales shortward of a few days. The DIARAD measurements represent a means by which the time scale-dependent response of solar-like stars to increased magnetic activity can be estimated. At solar maximum (with high magnetic activity levels), variability at long time scales increases significantly relative to solar minimum, while it remains comparatively constant at time scales of hours (see Figure 7.2). To generate a synthetic time series for an arbitrary rotation period, then, we first scale the variances of the OWT of the filled-in DIARAD time series (binned to 15 min) according to equation 7.10 and the ratio of the curves in Figure 7.2, so that the scaling ramps from a factor of 1 at the shortest time scale up to the value given by equation 7.10 by the 9th time scale ($\approx$ 2.66 days). Next, the inverse OWT is performed, and the resulting time series is resampled by linear interpolation onto the appropriate time grid. Finally, *Kepler's* combined shot and instrumental noise for m_v = 12 stars is added to the resampled time series. This procedure represents our best estimate of how stellar rotation period should affect the photometric variability of solar-like stars. We do not expect this model to be accurate over a wide range of stellar types. It probably is only indicative of the expected effects over stellar types near the Sun (G1–G4). Earlier type stars generally exhibit less spotting and consequently, lower σ_{phot}, while later type stars exhibit more spotting and higher σ_{phot} for a given P_{rot} (see, e. g., 82). Earlier type stars, however, are larger, requiring a larger planet to achieve the same S/N for a given photometric variability, while later type stars are smaller, mitigating the increased variability for a given size planet to some degree. This analysis does not include the effects of flare events, which exhibit transient signatures on time scales of minutes (more frequently) to a few hours (more rarely), the frequency of which increases significantly for rapid rotators.

Keeping these limitations in mind, we investigated rotation periods from as short as one tenth to as long as twice that of the Sun, where we adopt a mean projected solar rotation period, $P_{\odot}$, of 26.6 days. Figure 7.12 shows the power density as a function of time scale for m_v=12, solar-like stars with $0.5P_{\odot} \leq P_{rot} \leq 2.0P_{\odot}$, along with the energy density of a 10-hr, Earth-size transit. As P_{rot} decreases, more transit energy is masked, decreasing the detectability. On the other hand, as P_{rot} increases, more transit energy leaks through the background noise, aiding in detection. Figure 7.13 shows the mean S/N determined over rotation periods between $0.1P_{\odot}$ and $2.0P_{\odot}$ and as a function of transit duration from 0.25 to 20 hours. The single transit S/N exceeds 4σ for transits longer than 7 hours and $P_{rot} \gtrsim 21$ days, giving a detection rate $\geq$ 84% for four or more such transits. (We note that applying the scaling relation of equation 7.10 to all time scales uniformly results in a value of 3.5σ for similar duration transits and rotation periods, yielding a 50% detection rate.) Figure 7.14 shows contour plots of the minimum detectable planet radius at the 84% detection rate for four transits (panel *a*) and for six transits (panel *b*) as functions of transit duration and stellar rotation period. Six 3-hr-long or longer transits are sufficient to detect an Earth-size planet for $P_{rot} \gtrsim 16$ days. *Kepler* stands a good chance of detecting planets at least

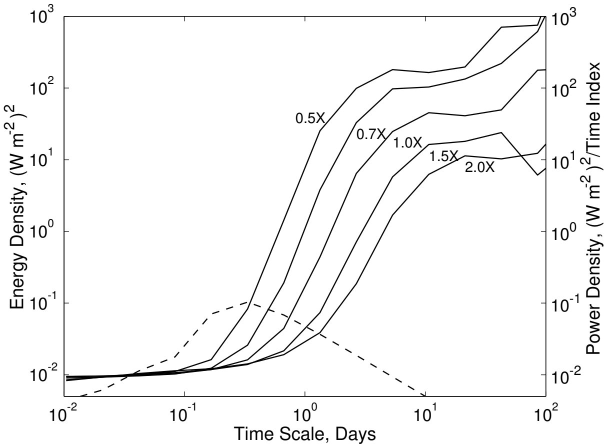

Figure 7.12: Distribution of power as a function of time scale from wavelet analyses of simulated time series of solar-like stars rotating both faster and slower than the Sun. The labeled solid lines are for stars with rotation periods between $0.5P_\odot$ and $2.0P_\odot$ ($P_\odot$ = 26.6 days), while the dashed curve shows the energy density of a 10-hr, Earth-size transit. As the rotation period decreases, the power spectrum shifts left towards shorter time scales, and upwards as well, due to increased photometric variability, and hence, 'swallows' more transit energy. Earth-size transits remain detectable for stars rotating as much as twice as fast as the Sun, so long as a sufficient number of transits ($\sim$ 7) are observed.

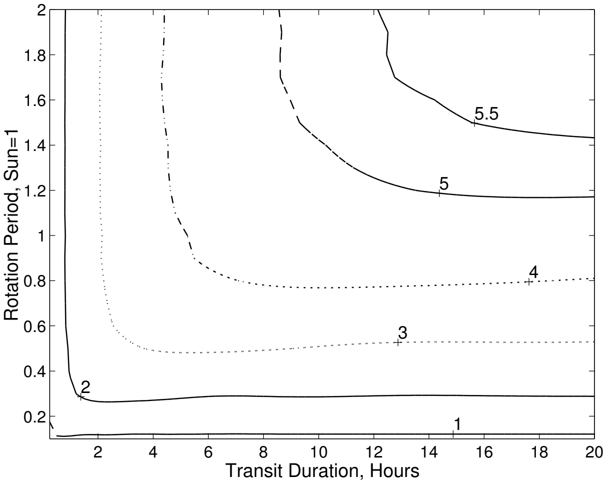

Figure 7.13: Contour map of the mean Earth-size, single transit S/N (in σ) as a function of transit duration and stellar rotation period for m$_v$=12, G2 stars in *Kepler's* FOV. Mean S/N's exceeding 4σ indicate a detection rate of at least 84% for four or more transits.

as small as Earth orbiting stars with rotation periods 40% shorter than that of the Sun.

7.4 Assessing Statistical Confidence in Transiting Planet Candidates

The interpretation of the S/N's obtained in Jenkins (60) in terms of detection probability depend on the distribution of the null statistics. If the observation noise is significantly non-Gaussian, equation (6.7) may underestimate the false alarm rate for a given threshold, and so, the detection rate may be lower than that indicated by equation (6.8) once a reasonable threshold is determined. In this section we characterize the distribution of null statistics for simulated *Kepler* data. We then assess its similarity to a Gaussian distribution in terms of the threshold required for a given false alarm rate. We note first that even if the distribution of the individual null statistics is significantly non-Gaussian, the distribution of the null statistics for multiple transits may be approximately Gaussian. This is due to the tendency of linear combinations of random variables to approach a Gaussian distribution (88). To address this question, we apply a bootstrap approach similar to that described in Jenkins, Caldwell, & Borucki (62). The modified algorithm is described in the appendix.

One might wonder whether solar-like variability produces transit-like features that might be confused with actual transit events. It is a curious characteristic of random processes that they can, indeed, produce any given feature if observed for a sufficient length of time. The DIARAD data set is no exception. There are several transit-like features over the 5.2 yr data set. The S/N of these features is no more than 5σ, and only a handful exhibit detection statistics larger than 4σ. The number of such events is somewhat higher than one would expect from Gaussian noise. The average Earth-size transit yields a detection statistic of $\sim 8\sigma$ against this noise. Thus, even though there are some transit-like features, they are individually much less significant than an Earth-sized transit event would be. The question to answer is: how great is the likelihood that a number of such features would occur with a purely periodic separation, so that the total S/N exceeds the detection threshold? To answer this question, we examine the bootstrap distribution of the null statistics of searches for sets of four 8-hr transits in the DIARAD data set.

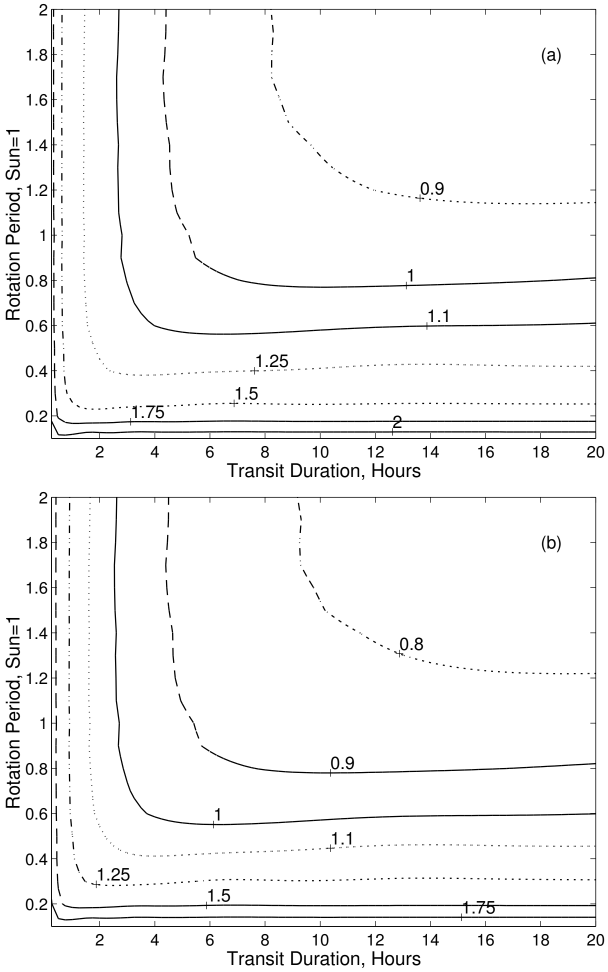

Figure 7.14: Contour maps of the minimum detectable planetary radius ($R_\oplus = 1$) at the 84% detection rate as a function of transit duration and stellar rotation period for planets exhibiting four transits (panel *a*) and six transits (panel *b*). Instrument and shot noise appropriate for m_v=12, G2 stars in *Kepler's* FOV is included in the analysis. Transiting Earth-size planets exhibiting six transits are detectable around stars with rotation periods 40% shorter than that of the Sun ($\mathrm{P}_{rot} \approx 16$ days).

Figure 7.15 shows the false alarm rate as a function of detection threshold for the bootstrap statistics for the bare DIARAD data, along with those for simulated *Kepler* data for an m_v=12 star, and for that expected for Gaussian noise. The range of false alarm rates extends from 10^{-10} to 10^{-15}. At the required false alarm rate of 10^{-12} for *Kepler*, the curves indicate thresholds of 7.04, 7.18, and 7.52σ, respectively, for Gaussian noise, for noise appropriate for a m_v=12 *Kepler* star, and for DIARAD data with no instrumental or shot noise added. Thus, to reach a false alarm rate appropriate for *Kepler*, we would need to increase the detection threshold above that for Gaussian noise by only 0.14σ for a m_v=12 star, and by $\sim 0.5\sigma$ for very bright stars ($m_v \leq 10.4$). This reduces the detection rate to 80% at m_v=12. At $m_v \leq 12$, however, the detection rate is reduced by an insignificant amount as the S/N for four Earth-sized transits is $\sim 16\sigma$ at these stellar magnitudes, which is much higher than the revised detection threshold of 7.5σ. Therefore, even though solar-like stars may exhibit occasional transit-like features (as would any random process), the frequency and strength of such features does not significantly increase the detection threshold that is required to limit the total number of false alarms over the entire campaign to no more than one. Thus, natural solar-like variations pose no threat to the ability of transit photometry to detect planets as small as Earth, assuming that a sufficient number of transits is observed.

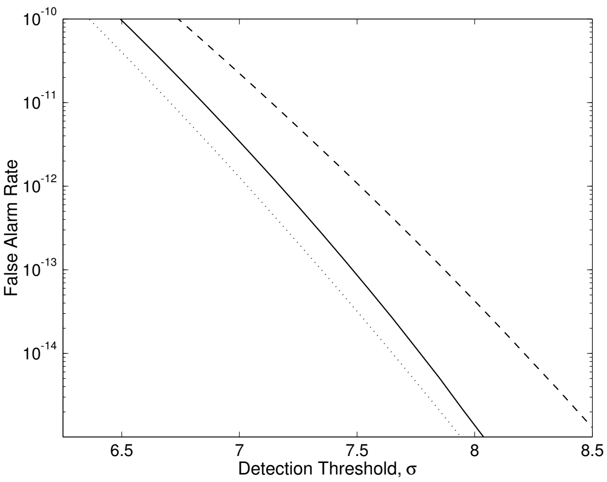

Figure 7.15: Graph of the false alarm rate as a function of detection threshold for a search for four 8-hr transits in the DIARAD data. The dotted line is for Gaussian noise, the solid line is for the DIARAD data plus shot and instrumental noise appropriate for an m_v=12 star, and the dashed curve is for the DIARAD data with no additional noise. Although the null statistics of the DIARAD data are significantly non-Gaussian, the combination of statistics for searches for 4 or more transits results in a distribution that is fairly well characterized as Gaussian. When the additional shot and instrumental noise for an m_v= 12 star is included, the resulting distribution is nearly Gaussian.

7.5 A Modified Bootstrap Algorithm for Determining the Distribution of the Null Statistics for a Transit Search

Here we outline the computational algorithm used to explore the bootstrap statistics of a search for several transits, given a time series representing observational noise. This is a necessary step in determining an appropriate detection threshold for a photometric transit campaign. The goal is to determine what the distribution of the null statistics is for multiple transits from a knowledge of null statistics corresponding to single transit events. A direct examination of the multiple event statistics for a data set such as from DIARAD is numerically prohibitive. Jenkins, Caldwell, & Borucki (62) provide a Monte Carlo approach for examining such distributions which can be computationally quite intensive. The approach given here allows one to concentrate efficiently on the upper tail of the distribution, which is often of greatest interest. First, assume that the single event statistics have been computed and that they have been sorted in descending order. Further assume that the numerator and denominator from equation (7.5) have been preserved, so that multiple event statistics can be computed from the components of the single event statistics. Now the bootstrap statistics for a search for L transits consist of forming the multiple transit statistics for all possible combinations of L events. For the DIARAD data set, there are $\sim 150,000$ time steps, for a total of over 4×10^{20} possible combinations for four transits. Clearly, forming the sample distribution for such a large number of points is out of the question. We can, however, sort the single event statistics and sample the distribution of interest in a practical manner, obtaining a histogram at any desired resolution.

Note that there is no natural *a priori* ordering for multiple event statistics in terms of the component single event statistics due to the manner in which the former are formed from the latter. However, the higher multiple event statistics will tend to be produced by combinations of high single events. Thus, it is possible to examine the bootstrap distribution of the multiple event statistics roughly from highest to lowest over a given range of values. We give the example for four transits, but the algorithm can be easily generalized to any number of transits. Begin with a counter set at [1,1,1,1]. This indicates the combination of four transits each identical to the event with the largest single event statistic. Here we assume a lower threshold of 6σ for the range of statistics of interest and a given bin size ($\ll 1$). The multiple event statistic corresponding to this combination of the ordered single event statistics is formed, and the histogram bin containing this statistic is incremented by one (the number of ways to draw this combination of statistics at random). The counter is incremented by one to [1,1,1,2], the corresponding statistic is formed and the corresponding histogram bin incremented by

4, the number of permutations of this set of digits. This procedure is continued until a statistic is encountered that is below the lower threshold (of 6σ for this example). At this point, the 2nd digit (from the right) of the counter is incremented to 2, the 1st is set to 2: [1,1,2,2], and the procedure is continued. At any point that a statistic is encountered below 6σ, the next higher digit from the one that was previously incremented is itself incremented. This criterion prevents the algorithm from needlessly considering multiple event statistics below the range of interest ($< 6\sigma$ here). Additionally, the monotonicity of the counter digits is preserved with every increment. In this way, assuming no lower threshold for skipping combinations, all possible combinations would be considered. At the termination of the algorithm, the number of events in each bin are divided by the total possible number of combinations of events to form a histogram of the probability density distribution above 6σ. Note that the resulting histogram will not be accurate in the neighborhood of the lower threshold, as many statistics that somewhat exceed this bound are not considered, due to the lack of a natural *a priori* ordering for the multiple event statistics. Hence, the lower threshold should be set conservatively below the actual range of interest. For the DIARAD data, reliable results are obtained above $\sim 6.25\sigma$. The false alarm rate as a function of threshold is obtained by taking 1 less the cumulative sum of the density histogram, and noting that the threshold is the left edge of each histogram bin.

This procedure may still be too taxing in computational terms. For example, assume that the lower threshold is 6σ and that there are 146,000 single events. Gaussian statistics imply that events greater than this threshold occur with frequency 10×10^{-10}. So we would expect the procedure above to terminate after approximately 4.5×10^{11} iterations. In this case, the procedure can be sped up by sampling, either deterministically or randomly. For deterministic sampling, instead of incrementing the counter by 1, it can be incremented by a fixed value greater than 1, say 100. Alternatively, the counter can be incremented by a discrete positive random deviate with a mean of 25, for example. Such deviates can be obtained simply by taking the nearest integer larger than the product of a uniform random deviate in the interval [0,1] and twice the desired mean increment. The resulting histogram must be multiplied by the mean increment value to account for the missing values. For the examples discussed in §7.5, the counter was randomly incremented with a mean increment of 25 and a histogram bin size of 0.1σ.

Chapter 8

Detecting Close-In Extrasolar Giant Planets by Reflected Light

This chapter draws heavily on Jenkins and Doyle (2003) 'Detecting Reflected Light from Close-In Extrasolar Giant Planets with the *Kepler* Photometer'. The nature of expected reflected light signatures from CEGPs is described, and the problem of determining an optimal detection algorithm is described. A practical generalized likelihood ratio test is discussed that searches for CEGP signatures in periodograms of stellar light curves. The task of setting an appropriate detection threshold is discussed. Prototype MATLAB source code is given in Appendix C for the detection algorithms and for setting the detection threshold.

8.1 The Reflected Light Signature

The reflected light signature of an extrasolar planet appears uncomplicated at first, much like the progression of the phases of the moon. As the planet swings along its orbit towards opposition, more of its star-lit face is revealed, increasing its brightness. Once past opposition, the planet slowly veils her lighted countenance, decreasing the amount of light reflected toward an observer. As the fraction of the visible lighted hemisphere varies, the total flux from the planet-star system oscillates with a period equal to the planetary orbital period. Seager et al. (102) showed that the shape of the reflected light curve is sensitive to the assumed composition and size of the condensates in the atmosphere of a CEGP. While this presents an opportunity to learn more about the properties of an atmosphere once it is discovered, it makes the process of discovery more complex: The reflected light signatures are not as readily characterized as those of planetary transits, so that an ideal matched filter approach does not appear viable. The signatures from CEGPs are small ($<$100 ppm) compared to the illumination from their stars, requiring many cycles of observation to permit their discovery. This process is complicated by the presence of stellar variability which imposes its own variations on the mean flux from the star. Older, slowly rotating stars represent the best targets. They are not as active as their younger counterparts, which are prone to outbursts and rapid changes in flux as star spots appear, evolve, and cross their faces. In spite of these difficulties, a periodogram-based approach permits the characterization of the detectability of CEGPs from their reflected light component.

Our study of this problem began in 1996 in support of the proposed *Kepler Mission*[1] to the NASA Discovery Program (12), (36). That study used measurements of solar irradiance by the Active Cavity Radiometer for Irradiance Monitoring (ACRIM) radiometer aboard the *Solar Maximum Mission* (*SMM*) (112), along with a model for the reflected light signature based on a Lambert sphere and the albedo of Jupiter. Here we significantly extend and update the previous preliminary study using measurements by the Dual Irradiance Absolute Radiometer (DIARAD), an active cavity radiometer aboard the Solar

[1] www.kepler.arc.nasa.gov

Heliospheric Observatory (*SOHO*) (41) along with models of light curves for 51 Peg b–like planets developed by Seager et al. (102). For completeness, we include Lambert sphere models of two significantly different geometric albedos, p=0.15 and p=2/3. The *SOHO* data are relatively complete, extend over a period of 5.2 years, are evenly sampled at 3 minutes, a rate comparable to that for *Kepler's* photometry (15 minutes), and have the lowest instrumental noise of any comparable measurement of solar irradiance. Seager et al. (102) provide an excellent paper describing reflected light curves of CEGPs in the visible portion of the spectrum. However, they do not consider the problem of detecting CEGP signatures in realistic noise appropriate to high precision, space-based photometers.

8.2 Detection Approach

The detection of reflected light signatures of non-idealized model atmospheres such as those predicted by Seager et al. (102) is more complicated than for the signature of a Lambert sphere. The power spectrum of any periodic waveform consists of a sequence of evenly spaced impulses separated by the inverse of the fundamental period. For a Lambert sphere, over 96% of the power in the reflected light component is contained in the fundamental (aside from the average flux or DC component, which is undetectable against the stellar background for non-transiting CEGPs). Thus, detecting the reflected light signature of a Lambert sphere can be achieved by forming the periodogram of the data, removing any broadband background noise, and looking for anomalously high peaks. In contrast, the power of the Fourier expansions of Seager et al.'s model CEGP light curves at high orbital inclinations is distributed over many harmonics in addition to the fundamental due to their non-sinusoidal shapes (see Fig. 8.1 and 8.2). How does one best search for such a signal? [2]

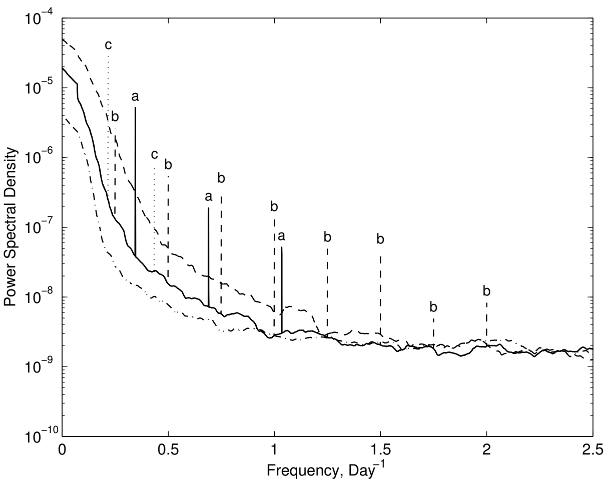

Figure 8.1: Power spectral density (PSD) estimates for solar-like variability and signatures of three extrasolar giant planets. The figure displays Hanning-windowed periodograms for a combination of the first 4 years of the DIARAD data set and three reflected light CEGP signatures. The three planetary signatures are for 1.2 $\mathrm{R_J}$ planets with atmospheres composed of 1.0 μm particles in a 4 day orbit, a planet with 0.1 μm particles in a 2.9 day orbit, and a 4.6 day, albedo $p = 2/3$, Lambert sphere. The planetary signatures consist of impulse trains with their harmonic components denoted by 'a's, 'b's and 'c's, respectively. The noise fluctuations in PSD estimates are quite evident.

As in the case of a pure sinusoid, a Fourier-based approach seems most appropriate, since the Fourier transform of a periodic signal is strongly related to its Fourier series, which parsimoniously and uniquely determines the waveform. Unlike the case for ground-based data sets that are irregularly sampled and contain large gaps, photometric time series obtained from space-based photometers like *Kepler* in heliocentric orbits will be evenly sampled and nearly complete. This removes much of the ambiguity encountered in power spectral analysis of astronomical data sets collected with highly irregular or sparse sampling. Thus, power spectral analyses using Fast Fourier Transforms (FFTs) simplify the design of a detector. For the sake of this discussion, let $x(n)$ represent the light curve, where $n \in \{0,\ldots,N-$

[2] A key point in searching for arbitrary periodic signals, or even pure sinusoids of unknown frequency is that no optimal detector exists (68). The most prevalent approach is to use a generalized likelihood ratio test which forms a statistic based on the maximum likelihood estimate of the parameters of the signal in the data. Such a detector has no pretenses of optimality, but has other positive attributes and often works well in practice.

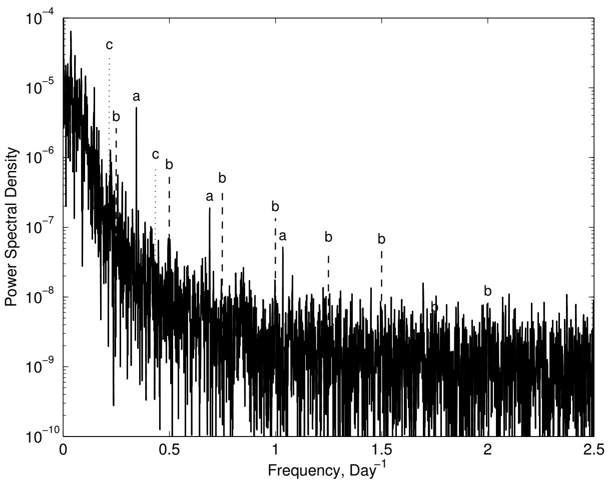

Figure 8.2: Three solar-like PSDs are displayed in the figure, along with a combination of these same planetary signatures and a 26.6 day period, solar-like star. The stellar PSDs have been smoothed by a 21-point moving median filter (0.015 Day^{-1} wide) followed by a 195-point moving average filter (0.14 Day^{-1} wide) to illustrate the average background noise. This is the procedure used by the proposed detector to estimate the background stellar PSDs prior to whitening the observed periodograms. The solid curve corresponds to the DIARAD data (P_{rot} = 26.6 days), while the dashed and dash-dotted curves are for solar-like stars with rotation periods of 20 and 35 days, respectively, demonstrating the dependence of stellar variability on stellar rotation period. Three harmonic components of the planet with 0.1 μm particles (solid lines topped with 'a's) are visible above the noise, while seven components of the planet with 1.0 μm particles are visible (dashed lines topped with 'b's). Only two components (dotted lines topped with 'c's) of the $p = 2/3$ Lambert sphere are visible. Thus, it should be possible to constrain the particle size distribution and composition of a CEGP atmosphere by the number of detected Fourier components. On this scale, the planetary signatures appear as vertical line segments, though they are actually distributed over a few frequency bins.

$1\}$ is an N-point time series with a corresponding discrete Fourier transform (DFT) $X(k)$, $\omega = 2\pi k/N$ is angular frequency, and $k \in \{0, \ldots, N-1\}$). The phase of the light curve is a nuisance parameter from the viewpoint of detecting the planetary signature and can be removed by taking the squared magnitude of the DFT, $P_X(k) = |X(k)|^2$, which is called the periodogram of the time series $x(n)$. In the absence of noise, if the length of the observations were a multiple of the orbital period, T_p, then the periodogram would be zero everywhere except in frequency bins with central frequencies corresponding to the inverse of the orbital period, $f_0 = T_p^{-1}$, and its multiples. If the length of the observations is not an integral multiple of the orbital period, the power in each harmonic is distributed among a few bins surrounding the true harmonic frequencies, since the FFT treats each data string as a periodic sequence, and the length of the data is not consonant with the true orbital period. The presence of wide-band measurement noise assures that each point in the periodogram will have non-zero power. Assuming that the expected relative power levels at the fundamental and the harmonics are unknown, one can construct a detection statistic by adding the periodogram values together that occur at the frequencies expected for the trial period T_p, and then threshold the summed power for each trial period so that the summed measurement noise is not likely to exceed the chosen threshold. The statistic must be modified to ensure that it is consistent since longer periods contain more harmonics than shorter ones, and consequently, the statistical distribution of the test statistics depends on the number of assumed harmonics. This is equivalent to fitting a weighted sum of harmonically related sinusoids directly to the data. Kay (68) describes just such a generalized likelihood ratio test (GLRT) for detecting arbitrary periodic signals in WGN assuming a generalized Rayleigh fading model.[3]

The approach we consider is similar; however, we assume the signals consist of no more than seven Fourier components, and we relax the requirement that the measurement noise be WGN. This is moti-

[3] In the Rayleigh fading model for a communications channel, a transmitted sinusoid experiences multipath propagation so that the received signal's amplitude and phase are distorted randomly. A sinusoid of fixed frequency can be represented as the weighted sum of a cosine and a sine of the same frequency, with the relative amplitudes of each component determining the phase. If both component amplitudes have a zero mean, Gaussian distribution, then the phase is uniformly distributed and the amplitude of the received signal has a Rayleigh distribution. The generalized Rayleigh fading model consists of a set of such signals with harmonically related frequencies to model arbitrary periodic signals.

vated by the observation that the model light curves developed by Seager et al. (102) are not completely arbitrary and by the fact that the power spectrum of solar-like variability is very red: most of the power is concentrated at low frequencies. At low inclinations, the reflected light curves are relatively smooth and quasi-sinusoidal, exhibiting few harmonics in the frequency domain. At high inclinations, especially for the $\bar{r}$=1.0 μm model, the presence of a narrow peak at opposition requires the presence of about seven harmonics in addition to the fundamental (above the background solar-like noise). Another GLRT approach would be to construct matched filters based directly on the atmospheric models themselves, varying the trial orbital period, inclination, mean particle size, etc. A whitening filter would be designed and each synthetic light curve would be "whitened" and then correlated with the "whitened" data.[4] We choose not to do so for the following reason: These models reflect the best conjectures regarding the composition and structure of CEGP atmospheres at this time, with little or no direct measurements of their properties. A matched filter approach based on these models could potentially suffer from a loss in sensitivity should the actual planetary atmospheres differ significantly from the current assumptions. On the other hand, the general shape and amplitude predicted by the models are likely to be useful in gauging the efficiency of the proposed detector.

Our detector consists of taking the periodogram as an estimate of the power spectral density (PSD) of the observations, estimating the broadband background power spectrum of the measurement noise, 'whitening' the PSD, and then forming detection statistics from the whitened PSD. We first form a Hanning-windowed periodogram of the N-point observations. For convenience, we assume the number of samples is a power of 2. For *Kepler's* sampling rate, f_s = 4 hr^{-1}, N = 2^{17} points corresponds to 3.74 years (or about 4 years). The broadband background, consisting of stellar variability and instrumental noise, is estimated by first applying a 21-point moving median filter (which replaces each point by the median of the 21 nearest points), followed by applying a 195-point moving average filter (or boxcar filter). The moving median filter tends to reject outliers from its estimate of the average power level, preserving signatures of coherent signals in the whitened PSD. The length of 195 points for the moving average corresponds to the number of frequency bins between harmonics of a 7 day period planet for the assumed sampling rate and length of the observations. Both of these numbers are somewhat arbitrary: wider filters reject more noise but don't track the power spectrum as well as shorter filters do in regions where the PSD is changing rapidly. This background noise estimate is divided into the periodogram point-wise, yielding a 'whitened' spectrum as in Figures 8.3 and 8.4. The advantage of whitening the periodogram is that the statistical distribution of each frequency bin is uniform for all frequencies except near the Nyquist frequency and near DC (a frequency of 0), simplifying the task of establishing appropriate detection thresholds. The whitened periodogram is adjusted to have an approximate mean of 1.0 by dividing it by a factor of 0.6931, the median of a $\chi_2^2(2x)$ process. (This adjustment is necessitated by the moving median filter.) Finally, the value 1 is subtracted to yield a zero-mean spectrum. [The distribution of the periodogram of zero-mean, unit-variance WGN is $\chi_2^2(2x)$ (see, e. g., 88).] Finally, the detection statistic for each trial period $N/(Kf_s)$ is formed by adding the bins with center frequencies iKf_s/N, $i = 1, \ldots, M$ together, where $M \leq 7$, as in Figure 8.5. The trial periods are constrained to be inverses of the frequency bins between 1/2 and 1/7 days^{-1}.

This procedure was applied to each of 450 model reflected light curves spanning inclinations from 10° to 90°, orbital periods from 2 to 7 days, plus stellar variability for stars with P_{rot} between 5 and 40 days and instrumental and shot noise corresponding to apparent stellar brightnesses between R=9.0 and

[4] For Gaussian observation noise and a deterministic signal of interest, the optimal detector consists of a whitening filter followed by a simple matched filter detector (68). The function of the whitening filter is to flatten the power spectrum of the observation noise so that filtered data can be characterized as white Gaussian noise. Analysis of the performance of the resulting detector is straightforward. For the case of non-Gaussian noise, the detector may not be optimal, but it is generally the optimal linear detector, assuming the distribution of the observation noise is known, and in practice often achieves acceptable performance.

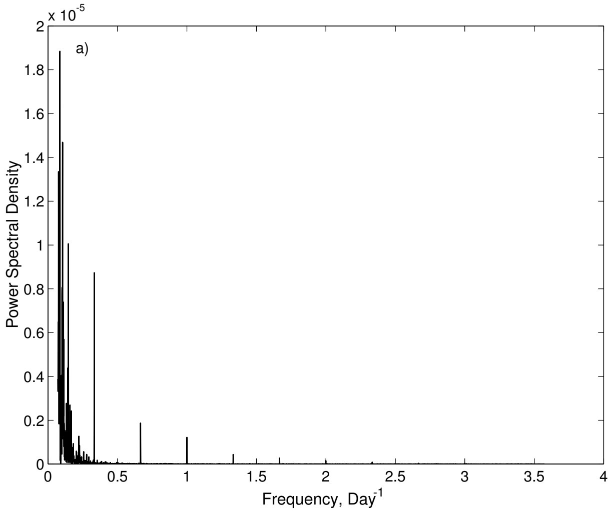

Figure 8.3: The process of applying the proposed detector to photometric data is illustrated by the periodogram of synthetic stellar variability for a solar-like star with a solar rotation period of 26.6 days, m_R=12 and an orbiting 1.2 R_J planet with an orbital period of 3 days.

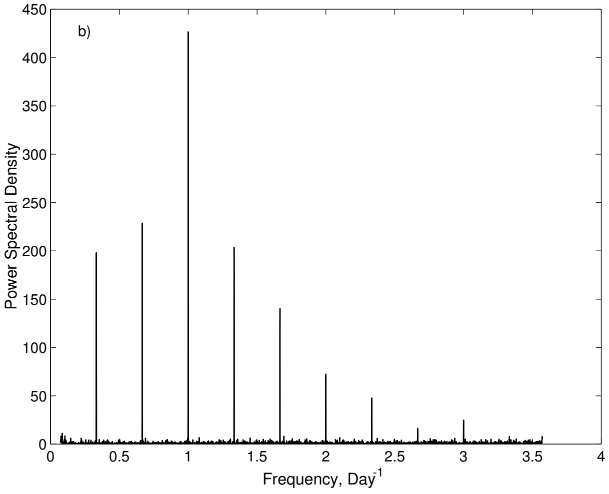

Figure 8.4: The process of applying the proposed detector to photometric data is illustrated by the "whitened" periodogram. The components of the signal due to the planet appear at multiples of 1/3 day^{-1}. The fundamental is not the strongest component in the whitened spectrum, as it would be for the case of white observational noise.

R=15.0. The combinations of these parameters generated a total of 21,600 synthetic PSDs for which the corresponding detection statistics were calculated. The number of assumed Fourier components was varied from $M = 1$ to $M = 7$. Some results of these numerical trials are summarized in Figure 8.6, which plots the maximum detectable orbital period, $P_{\max}$, for $M = 1$ at a detection rate of 90% against I, for P_{rot}=20, 25 and 35 days, for Sun-like (G2V) stars with apparent stellar magnitudes m_R=9.5, 11.5 and 13.5. Detection thresholds and detection rates are discussed in §8.3.

For $\bar{r} = 0.1$ μm clouds (Fig. 8.6a), planets are detectable out to $P = 4.75$ days for P_{rot}=35 days, out to $P = 3.7$ days for $P_{rot} = 25$ days, and out to $P = 3.1$ days for $P_{rot} = 20$ days. The curves are rounded as they fall at lower inclinations, and planets with I as low as 50° are detectable for all the curves, while planets with $I > 20°$ are detectable only for stars with $P_{rot} = 35$ days. For clouds consisting of $\bar{r} = 1.0$ μm particles (Fig. 8.6b), the curves of $P_{\max}$ are more linear, extending to orbital periods as long as 6 days for $P_{rot} = 35$ days, as long as 4.8 days for $P_{rot} = 25$ days, and to > 3 days at high inclinations for stars brighter than m_R=14. The detectability of both of these models at high orbital inclinations would be improved by searching for more than one Fourier component, (i. e., choosing a higher value for M). This is a consequence of the larger number of harmonics in the reflected light signature. Although the power is distributed among more components, as the orbital period increases, the signal is less sensitive to the low frequency noise power due to stellar variability, which easily masks the low frequency components of the signal. The behavior of the maximum detectable planetary radius for a Lambert sphere with $p = 0.15$ (Fig. 8.6c) is very similar to Seager et al.'s $\bar{r} = 0.1$ μm model. A Lambert sphere with $p = 2/3$ outperforms all the other models, as expected due to its significantly more powerful signal. Planets in orbits up to nearly 7 days can be detected for Sun-like stars with rotation periods of 35 days. For Sun-like stars with rotation periods of 25 and 20 days, planets are detectable with orbital periods up to 5.4 and 4.6 days, respectively. The Lambert sphere model PSD's contain only two Fourier components. Con-

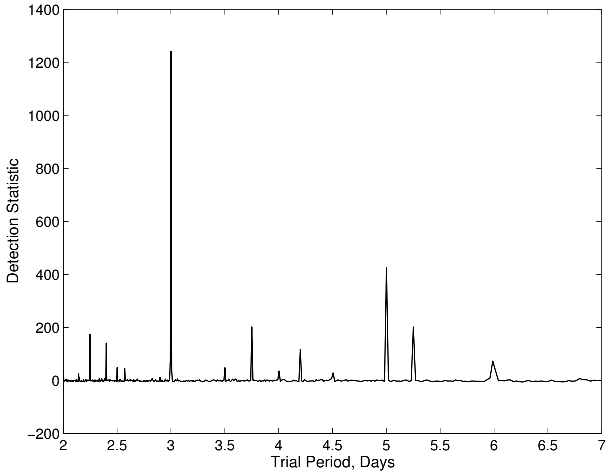

Figure 8.5: The co-added spectrum corresponding to the time series in Fig. 8.3 and 8.4 is shown. The periodogram has been co-added to itself so that the components of a periodic signal appear in the same bin, and thus, dramatically increase the chance of detection. Note the strong peak at 3 days, corresponding to the period of the signal in the time series. This may not always be the case as it depends on the strength of the fundamental compared to the background stellar and instrumental noise. In any case, the presence of many strong peaks at rational harmonics of the actual fundamental provide additional confidence that a periodic signal has been detected, and their spacing dictates the fundamental period.

sequently, the detectability of such signatures is not improved significantly by choosing $M > 1$.

Now that we have specified the detector, we must analyze its performance for the stellar population and expected planetary population. We should also determine the optimal number, M_{opt}, of Fourier components to search for, if possible. The value of doing so cannot be overstated: higher values of M require higher detection thresholds to achieve a given false alarm rate. If too large a value for M is chosen then adding additional periodogram values for $M > M_{\mathrm{opt}}$ simply adds noise to the detection statistic. This will drive down the total number of expected detections. On the other hand, if too small a value for M is chosen, then the sensitivity of the detector to CEGP signatures would suffer and here, too, the number of expected detections would not be maximized. The first step is to determine the appropriate threshold for the desired false alarm rate as a function of M. This is accomplished via Monte Carlo runs as presented in §8.3. To determine the best value of M, we also need a model for the population of target stars, which defines the observation noise, and a model for the distribution of CEGPs. We use the Besançon galactic

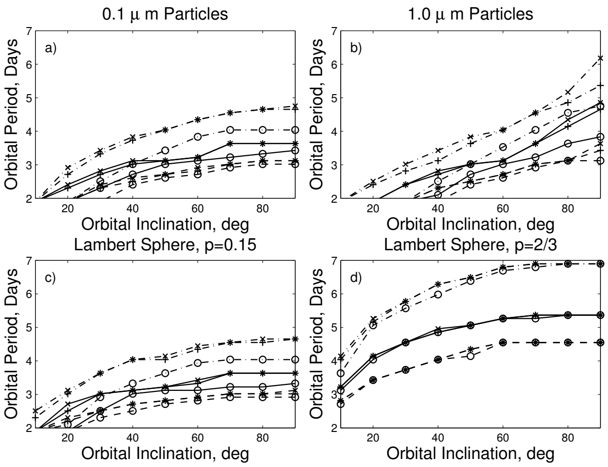

Figure 8.6: The maximum detectable planetary period at a detection rate of 90% vs. orbital inclination for various stellar brightnesses and rotation periods and 4 years of data are plotted for: a) Seager et al.'s $\bar{r} = 0.1$ μm particle model, b) Seager et al.'s $\bar{r} = 1.0$ μm particle model, c) a Lambert sphere with geometric albedo $p = 0.15$, and d) a Lambert sphere with $p = 2/3$. The number of assumed Fourier components, M, is set to one here. Stellar rotation periods of 20 days, 25 days and 35 days are denoted by dashed lines, solid lines and dash-dotted lines, respectively. Stellar magnitudes m_R=9.5, 11.5 and 13.5 are denoted by 'x's, crosses, and open circles, respectively. The first three models yield comparable numbers of expected CEGP detections. Seager et al.'s $\bar{r} = 1.0$ μm particle model is easier to detect at longer periods at high orbital inclinations relative to the $\bar{r} = 0.1$ μm particle model or the $p = 0.15$ Lambert sphere model. This is due to the greater number of Fourier components, which can compensate for red noise from stellar variability that can mask lower frequency harmonics.

model to characterize the target star population. The distribution of CEGPs with orbital period can be estimated from the list of known CEGPs. Moreover, we need a method for extrapolating solar-like variability from that of the Sun to the other spectral types. Two methods are considered. In the first, the stellar variability is treated strictly as a function of stellar rotation period, so that the detection statistics are adjusted for the varying stellar size. In the second, it is assumed that the mitigating effects of decreasing (increasing) the stellar area towards cooler (warmer) late-type stars are exactly balanced by an increase (decrease) in stellar variability. Hence, no adjustment is made to the detection statistics as a function of spectral type. Given this information, we can then determine which value of M maximizes the number of expected CEGP detections for a particular atmospheric model.

We found that the optimal value of M depends a great deal on the assumed stellar population, and the distribution of CEGPs with orbital period. If the rotation periods of *Kepler's* target stars were evenly distributed, then optimal values for M varied from $M = 1$ to 5, depending on the atmospheric model and method for extrapolating stellar variability across spectral type. Adopting a realistic distribution of stellar rotation period and spectral type produced a surprising result. We found that $M = 1$ yielded the highest number of detections assuming all four of the atmospheric models considered were equally likely. The number of detections for each atmospheric model as a function of M, and the average number of detections across all four atmospheric models are given in Table 8.1. The results of both methods for extrapolating stellar variability across spectral type are averaged together for this exercise. The effects of setting M to 1 were not strong for Seager et al.'s $\bar{r}$=1.0 μm model where M_{opt} exceeded 1. In this case, $M = 2$ or 3 was optimal, depending on how stellar variability was extrapolated. Up to 6% fewer CEGPs would be detected using $M = 1$ rather than $M = 3$ (174 vs. 185 total detections). For Seager et al.'s $\bar{r}$ =0.1 μm model and both Lambert sphere models, $M = 1$ was optimal, although the average number of detections drops slowly with M.

8.3 Monte Carlo Analysis

In order to determine the detection thresholds and the corresponding detection rates, we performed Monte Carlo experiments on WGN sequences. Much of this discussion draws on that of Jenkins, Caldwell, & Borucki (62), which concerns the analogous problem of establishing thresholds for transit searches. Each random time series was subjected to the same whitening, and spectral co-adding as described in §8.2. Two statistical distributions produced by these Monte Carlo trials are of interest: that of the null statistics for a single trial period, and that of the maximum null statistic observed for a search over all the trial periods. The former defines in part the probability of detection for a given planetary signature and background noise environment, since the distribution of the detection statistic in the presence of a planet can be approximated by shifting the null distribution by the mean detection statistic. The latter dictates the threshold necessary to control the total number of false alarms for a search over a given number of stars.

Let $l_{1,0}(M)$ denote the random process associated with the null statistics for a single trial period, and assumed number of Fourier components, M. Likewise, let $l_{\max,0}(M)$ denote the random process corresponding to the null statistics for a search of a single light curve over all trial periods. The corresponding cumulative distribution functions are $P_{l_{1,0}}(x,M)$ and $P_{l_{\max,0}}(x,M)$, respectively.[5] For N_* stars, the thresholds, $\eta(M)$, that yield a false alarm rate of $1/N_*$ for each search are those values of x for which

$$Q_{l_{\max,0}}(x,M) = 1 - P_{l_{\max,0}}(x,M) = 1 - 1/N_* \qquad (8.1)$$

and hence, deliver a total expected number of false alarms of exactly one for a search of N_* light curves. For a given threshold, η, and mean detection statistic, $\bar{l}_1(M)$, corresponding to a given planetary signature the detection rate, $P_D(M)$, is given by

[5] In this discussion, the cumulative distribution function of a random variable y is defined as the probability that a sample will not exceed the value x: $P_y(x) = P(y \leq x)$. The complementary distribution function, $1 - P_y(x)$ will be denoted as $Q_y(x)$.

Table 8.1: Number of Expected Detections vs. Assumed Number of Fourier Components

	Atmospheric Model				
M	$\bar{r} = 1.0^a$	$\bar{r} = 0.1^a$	$p = 2/3^b$	$p = 0.15^b$	Average
1	173.7	168.7	738.0	158.9	309.8
2	184.7	155.3	736.6	146.9	305.9
3	183.8	140.4	719.7	130.8	293.7
4	175.0	126.7	706.6	117.6	281.5
5	165.8	116.1	693.6	107.7	270.8
6	159.1	108.6	683.2	101.0	263.0
7	152.9	102.5	675.6	96.0	256.8

[a] Atmospheric models from Seager et al. (102) with mean particle radii $\bar{r}$ in microns.

[b] Lambert sphere models with the given geometric albedos, p.

$$P_D(M) = P_{l_{1,0}}(\bar{l}_1 - \eta, M), \qquad (8.2)$$

where the explicit dependence of $\bar{l}_1$ and η on M is suppressed for clarity.

Figure 8.7a shows the sample distributions for $Q_{l_{1,0}}(x,M)$ resulting from 619 million Monte Carlo trials for $M = 1$, 3, 5, and 7. This represents the single test false alarm rate as a function of detection threshold. Figure 8.7b shows $Q_{l_{\max,0}}(x,M)$ resulting from 1.3 million Monte Carlo runs, for the same values of M. This represents the single search false alarm rate as a function of detection threshold for each value of M. Error bars denoting the 95% confidence intervals appear at selected points in both panels.

It is useful to model $P_{l_{1,0}}$ and $Q_{l_{\max,0}}$ analytically. If the whitening procedure were perfect, and assuming that the observation noise were Gaussian (though not necessarily white), $l_{1,0}$ would be distributed as a χ^2_{2M} random variable with a corresponding distribution $Q_{\chi^2_{2M}}(2x+2M)$. Figures 8.7a and 8.7b show the sample distributions for $l_{1,0}$ resulting from 619 million Monte Carlo runs. Higher values of M require higher thresholds to achieve a given false alarm rate. We fit analytic functions of the form

$$Q_{l_{1,0}}(x,M) \approx Q_{\chi^2_{2M}}(Ax+B) \qquad (8.3)$$

to the sample distributions $Q_{l_{1,0}}(x,M)$, where parameters A and B allow for shifts and scalings of the underlying analytical distributions. Two methods for determining the fitted parameters are considered. In the first, we fit the analytic expressions directly to the sample distributions, including the uncertainties in each histogram bin. The resulting fit is useful for estimating the detection rate as a function of signal strength above the threshold, but may not fit the tail of the distribution well. In the second method, the log of the analytic function is fitted to the log of the sample distributions in order to emphasize the tail. The fitted parameters are given in Table 8.2. Regardless of whether the sample distribution or the log sample distribution is fitted, the values for A are within a few percent of 2 and the values of B are no more than 14% different from $2M$, indicating good agreement with the theoretical expectations.

To determine the appropriate detection thresholds, we need to examine the sample distributions $Q_{l_{\max,0}}$. These are likely to be well-modeled as the result of taking the maximum of some number, N_{EIT}, of independent draws from scaled and shifted χ^2_{2M} distributions. Here, N_{EIT} is the effective number of independent tests conducted in searching for reflected light signatures of unknown period in a single light curve. We take the values for A and B obtained from the fits to the log of $Q_{l_{1,0}}(x,M)$ and fit the log of the analytic functions of the form

$$Q_{l_{\max,0}(x,M)} \approx 1 - P^{N_{\text{EIT}}}_{\chi^2_{2M}}(A,x+B) \qquad (8.4)$$

to the log of the sample distributions $Q_{l_{\max,0}}(x,M)$.

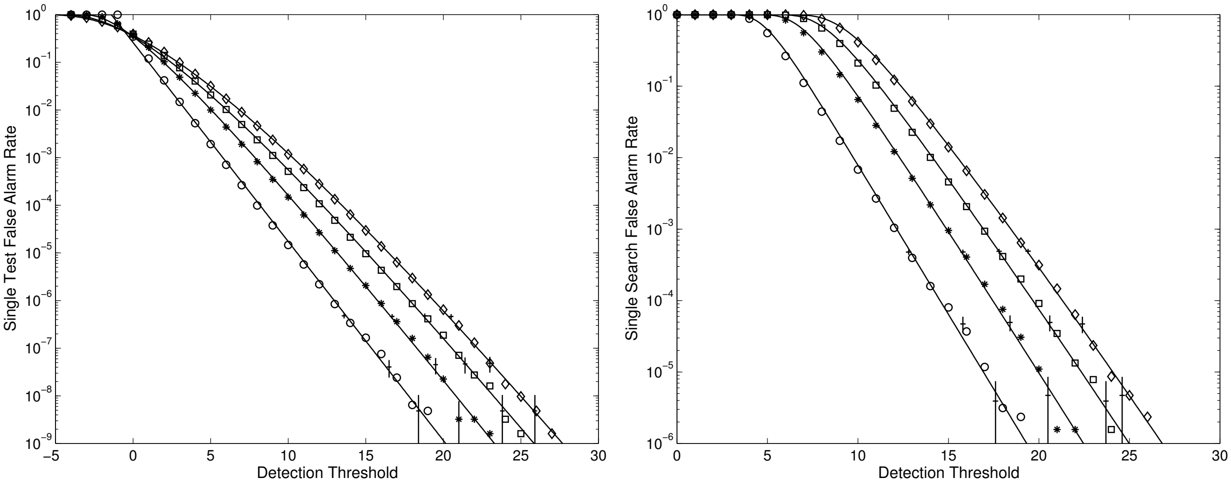

Figure 8.7: The single test and search false alarm rates as functions of detection threshold for the proposed detector. The number of assumed Fourier components, M=1, 3, 5 and 7, are denoted by circles, asterisks, squares, and diamonds, respectively, for the sample distributions. For clarity, only every fifth point of each sample distribution is plotted. The solid curves indicate the least-squares fits to the log of the sample distributions, emphasizing the upper tail in the fit. Error bars for 95% confidence intervals are denoted by vertical line segments crossed by horizontal line segments at various locations in each sample distribution. The single test false alarm rates can be used to estimate the detection rates for a given CEGP signal (see Fig. 8.8), while the single search false alarm rates determine the detection threshold for a given number of target stars and desired total number of false alarms. Determining the optimal value of M is important, given that higher values of M require correspondingly higher detection thresholds, which drives down the number of detections if the chosen value of M is too high.

Table 8.2: Analytical Fits to Monte Carlo Null Distributions

	Fit to Single Test				Fit To	
	Direct Fit[a]		Fit to Tail[b]		Single Search[c]	
M	A	B	A	B	N_{EIT}	Threshold
1	2.110	2.114	1.923	2.691	451.81	16.9
2	2.106	4.231	1.936	4.911	429.73	18.8
3	2.104	6.346	2.001	6.738	462.57	20.0
4	2.104	8.460	1.995	9.002	463.56	21.3
5	2.103	10.574	2.006	11.082	469.40	22.3
6	2.103	12.688	1.980	13.548	459.68	23.5
7	2.104	14.801	2.037	15.170	476.03	24.1

[a] The fit is of the form $P_{l_{1,0}}(x,M) \approx P_{\chi^2_{2M}}(Ax+B)$

[b] The fit is of the form$P_{l_{\max,0}}(x,M) \approx P^{N_{\mathrm{EIT}}}_{\chi^2_{2M}}(Ax+B)$, where A and B are fits to the tail of the single test distributions.

[c] Threshold for a false alarm rate of 1 in 10^5 searches of stellar light curves.

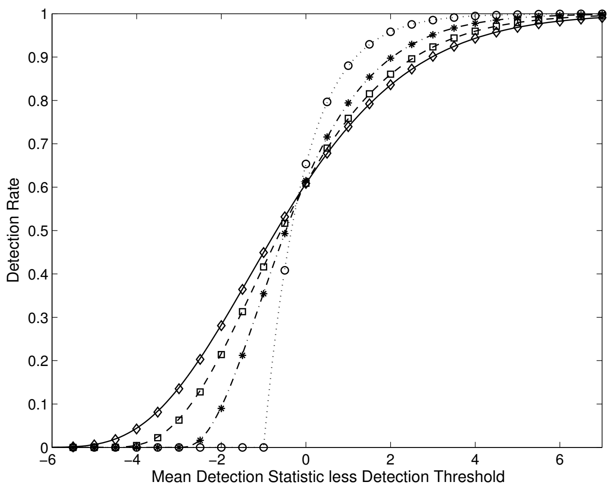

Figure 8.8: The detection rate as a function of the signal strength above the detection threshold (various symbols) along with analytic expressions (various curves) fitted to the empirical distributions. The number of assumed Fourier components, M=1, 3, 5 and 7, are denoted by circles, asterisks, squares and diamonds, respectively for the sample distributions. The corresponding analytical fits are denoted by dotted, dash-dotted, dashed and solid curves, respectively. For clarity, only every 5th point is plotted for the sample distributions. At the threshold, the detection rate attains ∼60%. This is due to the asymmetry of the distribution of null statistics. On this scale, the empirical distribution functions and the analytic expressions appear identical.

The values for N_{EIT} are given in Table 8.2 and fall between 430 and 476. For the length of data considered, there are ∼490 frequency bins corresponding to periods between 2 and 7 days. Thus the whitening and spectral co-adding operations apparently introduce some correlation among the resulting detection statistics, somewhat reducing the total number of independent tests conducted per search.

In determining the expected number of CEGPs whose reflected light signatures *Kepler* will likely detect, we average the detection rates from §8.2 over all inclinations and over the distribution of planetary periods of known CEGPs. The former can be accomplished by noting that inclination for randomly oriented orbits is distributed according to the sine function. Table 8.3 contains the average detection rates for 1.2 R_J planets orbiting Sun-like stars as functions of stellar rotation period and apparent magnitude for all four atmospheric models for a detector with M = 1. These results correspond to a false alarm rate of 1 in 10^5 light curve searches. The detection rate falls more rapidly with decreasing stellar rotation period than it does with increasing apparent stellar magnitude for the range of magnitudes and rotation periods considered here. The atmospheric models predicted by Seager et al. (102) are sensitive to the planet-star separation and are not likely to be accurate for planets well within 0.04 AU or planets much beyond 0.05 AU. Most of the planets making up our assumed planetary orbit distribution function fall within or close to these limits. Thus, we do not believe that departures from the simple scaling suggested by Seager et al. (102) are important in estimating the number of CEGPs that *Kepler* will detect. The detection rate is zero for stars with rotation periods shorter than 20 days for all save the $p = 2/3$ Lambert sphere model which can detect planets orbiting stars with P_{rot} as short as 15 days.

8.4 Potential Sources of Confusion and Methods of Discrimination

Detection algorithms detect all signals of sufficient amplitude with features that are well matched to the shape of the signal of interest.[6] Thus, not all signals yielding detection statistics above the detection threshold need be signatures of CEGPs. Indeed, several potential sources of confusion exist that might inject signals similar to reflected light signatures of CEGPs. These include intrinsic photometric variability of target stars themselves, and dim background variable stars within the photometric apertures of target stars. Such variations include those produced by star spots, eclipsing or grazing eclips-

[6] An exception to this rule is provided by the incoherent matched filter or “energy detector” that thresholds the variance of a time series. This detector is not sensitive to the shape of the input signal, and consequently, suffers inferior performance relative to a matched filter when the shape of the target signal is well defined (see, e. g., 68).

Table 8.3: Average Detection Rate for 1.2 R_J planets Orbiting Sun-Like Stars,(%)

	Apparent Stellar Magnitude					
P_{rot}	(m_R)					
(Days)	9.5	10.5	11.5	12.5	13.5	14.5
	$\bar{r}$=1.0 μm Particles					
5	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0
20	12.2	12.0	11.8	10.8	8.2	2.6
25	36.0	35.7	34.6	31.8	24.0	8.2
30	49.6	48.7	47.4	43.5	33.2	13.3
35	59.3	58.2	55.3	53.0	40.8	15.9
40	66.5	65.9	64.4	56.6	44.6	16.8
	$\bar{r}$=0.1 μm Particles					
5	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0
20	10.8	10.6	10.3	9.9	5.1	0.0
25	36.5	36.3	35.7	34.0	25.8	5.0
30	53.5	53.2	51.6	48.3	39.2	9.5
35	62.9	62.1	60.4	58.2	46.9	10.0
40	72.0	71.5	68.8	64.4	51.1	10.2
	Albedo $p = 0.15$ Lambert Sphere					
5	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0
20	6.8	6.7	6.3	4.9	1.0	0.0
25	38.6	38.4	37.5	34.0	25.4	1.2
30	56.6	56.4	55.9	52.7	42.4	4.6
35	67.3	67.1	65.6	61.2	50.0	5.6
40	75.6	75.4	74.4	70.1	54.7	5.9
	Albedo $p = 2/3$ Lambert Sphere					
5	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0
15	39.0	39.0	39.0	38.9	38.8	37.5
20	67.1	67.1	67.0	66.9	66.3	64.3
25	82.4	82.4	82.4	82.3	81.9	78.8
30	84.1	84.1	84.1	84.1	83.6	80.9
35	93.9	93.9	93.8	93.4	92.4	84.6
40	97.3	97.3	97.2	96.4	95.6	89.1

ing binaries, or intrinsic stellar pulsations. Section §8.4.1 describes each of these classes of variability along with an assessment of the likelihood they pose as sources of confusion. Section §9.2 presents a robust method for rejecting confusion from blended, variable background stars in a target star's photometric aperture.

8.4.1 Potential Sources of Confusion

Sources of stellar variability that might be mistaken for reflected light signatures of CEGPs include stellar pulsations, star spots, and photometric variability induced by binarity. These phenomena can occur in the target star or in a blended background star, but the amplitudes of concern are different since the magnitude of the variations of a blended background star will be diluted by the flux of the target star. In addition, non-reflected light signatures of CEGPs might be present, confounding the isolation and detection of the reflected light signature. In this section we discuss these sources of photometric variability and assess the likelihood that each poses as a source of confusion.

CEGPs can induce periodic photometric variations other than that due to reflected light. Doppler modulation of the host stellar spectrum via reflex motion of the host star about the system barycenter modulates the total flux observed in the photometer's bandpass. Loeb & Gaudi (79) estimate the amplitude of this effect and conclude that Doppler-induced photometric variations for Jupiter-mass planets orbiting solar-type stars in periods less than 7 days are about 20 times fainter than the reflected light signature of Jupiter-sized, $p = 2/3$ Lambert spheres. The Doppler-induced photometric signal is 90° out of phase with that of the reflected light component from a CEGP. Hence, rather than making it more difficult to detect a CEGP, the combination of the two signatures makes it easier to detect one since the power from orthogonal signals add constructively in the frequency domain. Radial velocity measurements should help distinguish between the two signatures in the case of non-transiting CEGPs.

Stellar pulsations can cause strictly periodic photometric variations. Acoustic waves traveling in the Sun resonate at specific frequencies with characteristic periods on the order of 5 minutes and typical amplitudes of $\sim$10 ppm. The coherence lifetime for these so-called p-mode oscillations is approximately a month, beyond which the sinusoidal components drift out of phase (34). Buoyancy waves (also called gravity waves) should have much longer periods of 0.28-2.8 hours along with correspondingly longer coherence timescales. To date, no one has observed the signatures of g-modes in the Sun. The VIRGO experiment aboard *SOHO* has placed upper limits of 0.5 ppm on the amplitudes of solar g-modes (4), which is in line with theoretical predictions (3). It does not appear that pulsations of solar-like stars could present major problems: the coherence timescales are short and the amplitudes are significantly smaller than those due to the reflected light component from CEGPs. Moreover, the amplitudes preclude stellar pulsations of background blended stars from being confused with signatures of CEGPs due to dilution.

Long-lived star spots or groups of spots can produce quasi-sinusoidal photometric signatures. Some individual starspot groups of F, G, and K dwarfs have been known to last for months-to-years and cover an appreciable fraction of the star's surface (20-40% in extreme cases, 28), with the starspot cycles themselves lasting from a half to several decades for nearby solar-type stars (5). Contributions to solar variability at tens of minutes come from granulation and are present in only a few tens of ppm, while sunspots contribute a variation of about 0.2% over days or weeks. Faculae can also contribute variations of about 0.1% over tens of days and last longer than individual sunspots, because differential rotation distributes these over the whole solar disc (58). It is difficult to imagine that star spots on solar-like single stars could be easily confused with CEGPs. On the Sun, for example, individual sunspots evolve and change continuously on timescales comparable to the mean solar rotation period (26.6 days). Thus, the photometric signatures of sunspots vary from rotation to rotation so that the photometric dips due to spots do not repeat with a great degree of precision. In the Fourier domain it can be difficult to identify the fundamental associated with the solar rotation period: the peak is extremely broad. Of more concern, then, are photometric variations from dim background late-type binaries, such as BY Dra or RS CVn variables.

The BY Draconis variables are dKe and dMe stars with typical differential amplitudes of 0.2 magnitudes and periods of a few days. For example, in photometric observations of CM Draconis (M4 + M4, 1.27 day period), Lacy (77) noted a $\sim$0.01 mag sinusoidal feature he attributed to a long-lived, high latitude spot group that persisted for years. RS CVn stars are generally eclipsing binaries consisting of at least one subgiant component. These stars display nearly sinusoidal variations of up to 0.6 mag. The photometric variations are due to an uneven distribution of cool spots in longitude that rotationally modulate the apparent flux. Fortunately, one way of distinguishing these variations from the phase variations of CEGPs is the fact that starspot activity of these stars varies with phase over time. Kozhevnikov & Kozhevnikova (76) found that the quasi-sinusoidal starspot variation of CM Draconis had shifted by 60 degrees in phase over a two decade period and had increased in amplitude (to $\sim$0.02 mag). The eponymous BY Dra (M0 Ve + M0 Ve) has a mean photometric period of 3.836 days, and can demonstrate rather fickle photometric behavior: the nearly sinusoidal variations discovered by Chugainov (24) nearly disappeared by mid-1973. The light curves for several BY Dra and RS CVn stars can be explained by the presence of two large spots on one of the stellar components. As the spots evolve and migrate in longitude, the photometric variations change significantly (see, e. g., 101). Some RS CVn systems with orbital rotation periods of several days exhibit remarkable photometric variations over timescales of months. The RS CVn binary V711 Tau (K0 V + B5 V), for example, has an orbital period of 2.84 days, and migration of spot groups in longitude leads to changes in its "photometric wave" including the exhibition of double peaks, nearly sinusoidal variations, and rather flat episodes (6). Starspot-induced variations do not seem likely candidates for being mistaken for reflected light signatures of CEGPs, even for binary systems.

Ellipsoidal variables [e. g., o Persei (B1 III + B2 III), period = 4.42 days, differential amplitude 0.07 magnitudes in V] are non-eclipsing binaries that display photometric variations due to the changing rotational aspect of their tidally elongated shapes (107). These stars' light curves exhibit two maxima and two minima per orbital period, and one minimum can actually be significantly deeper than the other. Thus, we do not expect that ellipsoidal variables will be mistaken for CEGPs as the shape of the variations is significantly different from that expected for CEGPs.

It is unlikely that photometric variations of binary target stars will be confused with CEGPs. The *Kepler Mission* will be preceded by ground-based observations to characterize all the stars in the FOV with $m_R \leq 16$. These observations should be able to detect almost all of the short period binaries. Moreover, ground-based, follow-up observations should be able to detect any of these types of variable stars in the cases where one might have been mistakenly classified. These follow-up observations should help discriminate between planetary and stellar sources for any candidate signatures of CEGPs. Nevertheless, we should examine the frequency of such binary systems in the photometric apertures of target stars, and *Kepler's* ability to distinguish between photometric variability intrinsic to a target and that due to blended background variables.

In a study of the light curves of 46,000 stars in the cluster 47 Tuc, Albrow et al. (1) identified 71 likely BY Dra stars that exhibited photometric variations as high as 0.2 magnitudes. The fraction of stars that are in binary systems is significantly lower in 47 Tuc ($\sim$14%) than it is in the galactic disc ($\sim$65%, as per 38). The peak-to-peak amplitudes of the CEGP reflected light curves considered here are between 20 and 60 ppm, so that background BY Dra binaries would need to be $\sim$8 magnitudes dimmer than a particular target star to exhibit photometric variations of the appropriate amplitude. We determined the distribution of late-type (G, K and M) stars with m_R=17.0 to 23.0 corresponding to the range of apparent magnitudes for *Kepler* target stars using the Besançon galactic model. The number of binary systems with rotation periods between 2 and 7 days can be estimated using the Gaussian model of Duquennoy & Mayor (38) for the distribution of binaries as a function of the log period. According to this distribution, $\sim$1.75% of binaries in the galactic disc should have periods in this range. Table 9.1 gives the number of background binaries with periods in this range consisting of at least one dwarf G, K or M star in each aperture of a *Kepler* target star. The apertures vary from 400 square arcsec for m_R=9.5 stars, to 200 square arcsec for m_R =14.5 stars, with a corresponding number of background binaries varying from 13 to 69, respectively. Even if such a system appears in the photometric aperture of a target star, it is likely that it can be detected by observing the centroid of the brightness distribution over time (Ron Gilliland 2001, personal communication), as discussed in §9.2.

Chapter 9

Data Validation

This chapter discusses the process of validating data. It includes a short section on establishing statistical confidence in detections. A second section discusses problems of discriminating against background variable stars as the source of the planet-like photometric signatures in a candidate light curve. The effects of such a background star on the centroid of a planetary target star are described and then used to develop a means of detecting whether a background star is the source of the photometric variations. Limitations to the effectiveness of this technique are explored as as function of the apparent magnitude of the targets star. Finally, the Ξ^2 fitting of light curve data to estimate various transit parameters.

9.1 Establishing Statistical Confidence in Detections

Consider any estimation problem in which parameters get estimated (along with appropriate error estimates on the parameters). A normal follow-up question is 'how sure can I be that there is not a much better fit in some other corner of the parameter space?' Given that merit functions often do not have a single, global minimum, what process stopped this fit from converging to a local minimum rather than the global minimum? Such questions are generally very difficult to answer.

If you happen to know the actual distribution law of your measurement errors, then you can use Monte Carlo simulations to create synthetic data sets. These data sets can be subjected to the same fitting procedure as the original data. This allows for a determination of the probability distribution of the χ^2 statistic and the accuracy with which the model parameters are reproduced by the fit.

This approach was used in §7.5 to explore the bootstrap statistics of a search for several transits, given a time series representing observational noise. This was a necessary step in determining an appropriate detection threshold for a photometric transit campaign. The goal was to determine what the distribution of the null statistics was for multiple transits from a knowledge of null statistics corresponding to single transit events.

This approach was also used in §8.3 to determine the detection thresholds and the corresponding detection rates for detection of extrasolar planets using reflected light signatures.

In both instances the bootstrap approach seemed to give good results. The reader is referred to these sections for details.

9.2 A Method to Mitigate Confusion from Blended Background Stars for CEGPs

Since *Kepler* will return target star pixels rather than stellar fluxes to the ground, it will be possible to construct centroid time series for all the target stars. This represents a robust and reliable means to discriminate between sources of variability intrinsic to a target star and those due to background variable stars situated within the target stars' photometric aperture. Suppose that the background variable located at $\mathbf{x_2}$ is separated from the target star located at $\mathbf{x_1}$ by $\Delta\mathbf{x} = \mathbf{x_2} - \mathbf{x_1}$, and that its brightness changes by δb_2 from a mean brightness of $\bar{b}_2$, while the target star's mean brightness is $\bar{b}_1$. Then the change in the photometric centroid position $\delta\mathbf{x_c}$ with respect to the mean position is given by

$$\delta\mathbf{x_c} = \delta b_2\ \Delta\mathbf{x}/(1+\bar{b}_1/\bar{b}_2). \qquad (9.1)$$

Thus, a background star 8 magnitudes dimmer than the target star separated by 1 arcsec and exhibiting a change in brightness of 10% will cause the measured centroid to change by 63 μas. The uncertainty in the centroid, however, is determined largely by the Poisson statistics of the stellar flux signal and the random noise in each pixel. For *Kepler's* Point Spread Function (PSF), the uncertainty of the centroid of an m_R=9.5 star measured over a 24 hr interval is $\sim$16 μas (on a single axis). At a magnitude of m_R=13.5, the corresponding uncertainty is $\sim$118 μas. Note, however, that we are not limited to the resolution of a centroid over a short interval: Equation 9.1 implies that the time series of the displacements of the target star's centroid will be highly correlated with the photometric variations if the latter are caused by a variable background star offset sufficiently from the target star. For detecting CEGPs by reflected light, this fact implies that the centroid time series of a star can be subjected to a periodogram-based test to determine if there are statistically significant components at the photometric period.

We performed numerical experiments with the PSF for *Kepler* and the expected shot and instrumental noise to determine the radius to which background variables can be rejected at a confidence level of 99.9% for four years of observation. The expected accuracy of the centroids given above assumes that errors in pointing can be removed perfectly by generating an astrometric grid solution for *Kepler's* target stars. At some magnitude, systematic errors will become significant. Here, we assume that the limiting radius inside which we cannot reject false positives is 1/8 pixels, or 0.5 arcsec. Better isolation of background binaries might be obtained in practice for stars brighter than $m_R = 14.0$. The relevant figures for these calculations are given in Table 9.1, showing that *Kepler* should be able to reject almost all such false positives for m_R <14.0. A significant number (28) of false positives might occur for target stars with $14.0 < m_R < 15.0$. These would require further follow-up observations to help discriminate between background variables and signatures of CEGPs. We note, however, that this assumes that the background variables display periodic signatures that retain coherence over several years. As discussed in §8.4.1, this is generally not the case.

9.3 The Effect of Dim Variable Background Stars on Target Star Centroids

An important problem for *Kepler* is the need to distinguish variable background stars from planets, as the former can inject transit-like signatures into the photometric apertures of target stars. For example, if an edge-on eclipsing binary lies within 2 arcsec of a target star and is $\sim$9.25 mags fainter than the target, the depth of the eclipse signal would be 1×10^{-4} that of the total target star flux. This should not be a major problem for transiting giant planets as the SNR is large enough to allow for detailed studies of the shape of the transit light curve to reject such confusion. In the case of transiting terrestrial planets, however, the SNR will most likely not allow for adequate discrimination power, unless the planet is in an extremely short period orbit. However, we have access to more than the light curve to mitigate this problem. So long as the variable background star is offset sufficiently ion the sky from the target, the centroid time history of the target star can reveal confusion. Identifying confusion for transiting planets is more challenging than that for reflected light candidates, since the signal only exists during the transits.

To get a handle on this problem, let us assume the following simple analytical model: Let the brightness of the target star be B, the brightness of the variable star be b, and let the stars be offset by Δx. Further, assume that the variable star's brightness changes by δb during the transit-like features. The change in the centroid position during 'transit' from

Table 9.1: Number of Background Binaries Not Excluded by Astrometry for Reflected Light Searches

	Apparent Stellar Magnitude					
	(m_R)					
Parameter	9.5	10.5	11.5	12.5	13.5	14.5
Number of Background Binaries in Target Apertures[a]	3	18	85	296	903	2405
Centroid Rejection Radius (arcsec)[b]	<0.5	<0.5	<0.5	<0.5	<0.5	0.7
Aperture Size (square arcsec)	400	384	352	288	240	192
Number of Potential False Alarms[c]	0	0	0	1	3	18

[a] The background binaries of concern have periods between 2 and 7 days and are 8 magnitudes fainter than the target stars.

[b] Background variables can be rejected outside this radius with a confidence level of 99.9%.

[c] These are the expected numbers of background variables that cannot be rejected simply by examining *Kepler* data. Follow-up observations may be necessary to distinguish them from CEGPs if the objects display coherent, periodic light curves over the 4 year duration of *Kepler's* observations.

the baseline is given by

$$\delta x = \frac{b\Delta x}{B+b} - \frac{(b-\delta b)\,\Delta x}{B+b-\delta b}, \qquad (9.2)$$

which simplifies to

$$\delta x = \frac{\delta b \Delta x}{B}, \qquad (9.3)$$

for $B \gg b$. Thus we can determine Δx from δx by dividing the latter by the fractional change in brightness observed in the light curve.

For a star with a Gaussian profile it can be empirically established that the uncertainty in δx is given approximately by the ratio of the Full Width Half Max (FWHM) to twice the S/N of the stellar flux signal (Dave Monet, personal communication):

$$\sigma^2_{\delta x} \approx \frac{\mathrm{FWHM}}{2\left(B/\sqrt{B+\sigma^2_{bg}}\right)}, \qquad (9.4)$$

where σ^2_{bg} is the variance of the background noise. Applying the standard propagation of errors to Eq. 9.3, we find the uncertainty in an estimate of the actual offset of the background star to be

$$\sigma^2_{\Delta x} \approx \left(\frac{\delta b}{B}\right)^{-2} \left[\sigma^2_{\delta x} + \Delta x^2 \sigma^2_{\frac{\delta b}{B}}\right]. \qquad (9.5)$$

The uncertainty in $\delta b/B$ is simply the photometric precision of the light curve on intervals equal to the transit duration, so long as we assume that there is no significant contribution from the baseline values for either the target star flux or position.

Note that we do not need to estimate Δx in order to detect confusion: it is sufficient to obtain $\delta x \gg \sigma^2_{\delta x}$. How large must Δx be for this to be true? Since we must estimate the centroid offset by combining the offsets in both axes $\sqrt{\delta x^2 + \delta y^2}$, we are dealing with χ process with 2 degrees of freedom. Choosing a threshold of $3\sigma_{\delta x}$ yields a 99% confidence that the apparent centroid offset is due to a background star and not to stochastic noise. The case of most interest is that for $\delta b/B \approx 1 \times 10^{-4}$, where $B = 5 \times 10^9$, corresponding to an $m_R = 12$ G2 star for a 6.5-hour interval. Adopting a fractional S/N of 20 ppm gives $\sigma^2_{bg} = 5 \times 10^9$. The FWHM of *Kepler* photometer is about 1.33 pixels or 5.29 mas so that $\sigma_{\delta x} = 52.9$ mas. Thus, we can reject background eclipsing binaries as the source for four transit-like features when $\Delta x > 0.794$ arcsec.

9.4 Development of Crowding Parameters

There is interest in developing metrics to describe not only the number of stars in the neighborhood of a target of interest, but also the relative magnitude and the distance from the target of interest. This gives a feel for how much the light from the target star is due to the flux from neighboring stars. This metric is called the Crowding Parameter.

As such, let the target star of interest be denoted by i. The crowding parameter for target i is, C_i. It is then computed by

$$C_i = \frac{F_i}{\sum_{j=1}^{N} F_j} \tag{9.6}$$

where F_i is sum of the total flux of star i in the its aperture, F_j is the flux from star j that is seen in the aperture for star i, and index j sums over the N stars whose flux contributes to the total flux seen in the aperture of the i^{th} star. Note that in this section the term *aperture* refers specifically to those pixels used to compute the target star's flux time series, not necessarily all surrounding pixels encompassing the aperture that is transmitted from *Kepler*.

This crowding parameter, C_i, is, under this definition, a ratio of the amount of light in a given aperture that is due to the target star as compared to the total light. It may be advantageous to represent this number as a percentage, in which case the formula in Equation 9.6 is multiplied by 100%.

At this point a decision may be required as to a star's fitness for inclusion in the *Kepler* catalogue due to its crowding parameter. If the parameter is too low, the star's flux will be too easily confused with that of nearby stars and therefore will be too noisy (in some sense) for a reliable target search algorithm.

Obviously other parameters can easily be envisioned combining the distance and magnitude of nearby stars into a measure of some sort. There may be utility in these measures. If so, they can be developed at that time.

Finally, it should be noted that the crowding parameter is computable given an input catalog and a representative PSF. All of this information could be developed prior to launch to facilitate initial target selection. Standard software routines in the IDL library exist (for example, DAOPHOT - Type Photometry Procedures are available at $IDLastro.gsfc.nasa.gov/contents.html$).

9.5 Assessing Physical Parameters

Upon successful identification of a transit event, a whole host of follow on questions arise as to the nature of the planet in question. What distance is it from the target star? What size is it? Does it have a period conducive to water existing in a liquid state? The list goes on. Based upon the limited data available from the light curve, some estimates can be made on the following parameters:

1. Transit Depth,
2. Transit Duration,
3. Transit Period,
4. Transit Phase, and (from other sources)
5. Stellar Type,
6. Stellar Distance,
7. Stellar Radius,
8. Stellar Mass.

Identification of the parameters does not end with the computation of the parameters themselves. It is important that this process is followed by estimating the error in the computations. Finally, it is generally a task to statistically measure the goodness of fit of the parameters to the physical model to assess the confidence in the match between the data and the model.

The actual mechanics of identifying and bounding the parameters will done via a χ^2 fitting process. Performing χ^2 fits in estimating errors in the fitted parameters are standard science activities. Many numerical analysis libraries return the standard errors along with the fitted parameters. Specific implementation of these efforts are beyond the scope of this document, but will be defined for the SOC at a future time.

Finally, it is expected that all available information associated with a target possessing a potential transit will be displayed in some sort of published report format that will be determined by the SOC.

Part II

Non-Pipeline Science Processing

Part II summarizes the non-pipeline science processing of the *Kepler* data. It begins with Chapter 10 which discusses the management of the target lists and their associated parameters, focusing on the photometer coordinate system transformations, and concluding with a discussion of aperture selection. Chapters 11 provides an overview of the expectations for some of the SOC software that will be used upon receipt of data from the DSN. Chapter 12 discusses the compression algorithms used to achieve lossless and reasonable downlink rates, including entropic encoding. It also discusses the mission's sensitivity to data loss. Chapter 13 covers a potential method for on-board detection and correction of cosmic ray events. This methodology will need to be adapted to ground-based cosmic ray detection and rejection. Finally, Chapter 14 describes the End-To-End-Model that has been developed to facilitate investigation of algorithms and engineering decisions associated with the *Kepler* Mission.

Chapter 10

Target List Management

This chapter discusses the tasks associated with managing the Target List. Such topics include monitoring CDPP, centroid motion, and updating the photometric aperture mask ID and target definitions for command build at the MOC.

10.1 Photometer Coordinate System Discussion

In order to locate and identify celestial targets in the *Kepler* data, it is necessary to specify the transformations between equatorial (*right ascension, declination*) and *Kepler* focal plane pixel coordinates (module, output, CCD row, CCD column). An initial transformation will be defined before launch to determine what targets will be in the field of view in order to generate an initial target list and to allow Guest Observers to prepare proposals. After launch, the coordinate conversion will be refined using the measured positions of known stars to solve for the transformation coefficients. The two different cases will be treated separately below. The coordinate system used by the *Kepler* photometer and the layout of the modules and CCD is described in section 10.2. Because of the quarterly 90° spacecraft rolls, there are four orientations of the focal plane on the sky, thus an integer number of 90° rotations about the photometer axis is added to the transformation depending on the season. The individual transformations used to determine a target's location on the focal plane are described in section 10.3. The post-launch method for refining the the coordinate transformation is described in § 10.4.

10.2 Photometer Coordinate System

There are separate coordinate systems for the spacecraft and for the photometer. The orthogonal spacecraft coordinates are defined as

+X the photometer axis with positive (+) pointing out of the photometer to the sky

+Y pointing out of the center of the solar array

+Z completing the right hand coordinate system.

The origin of the spacecraft coordinate system is near the spacecraft–launch vehicle interface.

The focal plane coordinates are defined as

+X′ coincides with spacecraft +X

+Y′ is 13.0 degrees (+ rotation about +X) offset from +Y

+Z′ is 13.0 degrees (+ rotation about +X) offset from +Z

The origin is near the surface of the center CCD. A detailed description of the focal plane coordinates is given in the systems engineering report Kepler.SER.FPA.006. Figure 10.1 shows the layout of the focal plane coordinates, modules, and CCD outputs. During operations there are four orientations of the focal plane corresponding to the quarterly 90° spacecraft rolls. With the exception of the central module (number 13) the layout of the CCDs is 90° rotationally symmetric.

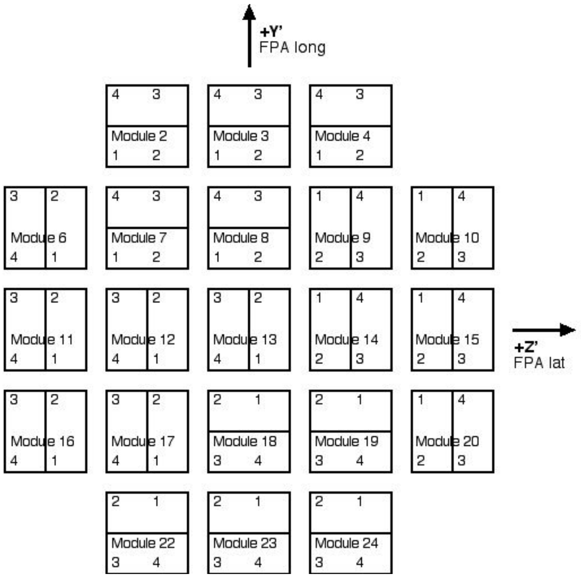

Figure 10.1: The layout of the *Kepler* focal plane with the module numbers (2–24) and output numbers (1–4 on each module). The focal plane Y′ and Z′ axes are indicated. Modules 1, 5, 21, and 25 correspond to the fine guidance sensors at the four corners of the focal plane.

There are two CCDs within each module. Each CCD detector chip has two output amplifiers. The outputs are numbered 1–4 on each module. The detector chips have a serial register on the long edge of the chip, with the amplifiers on the corners. The row number increases along the short edge of the chip from each amplifier. The column number increases along the long edge of the chip from both amplifiers towards the middle. The active imaging rows are numbered 1-1024, columns from 1 to 1100. Therefore, a location on the focal plane is uniquely specified by module number, output number, row, and column (e. g., module 13, output 2, row 256, column 625). Dark pixels and over clocked pixels (bias and smear) are described in Chapter 3, Pixel Level Calibrations. Figure 10.2 illustrates the orientation of the CCD rows and columns within a module.

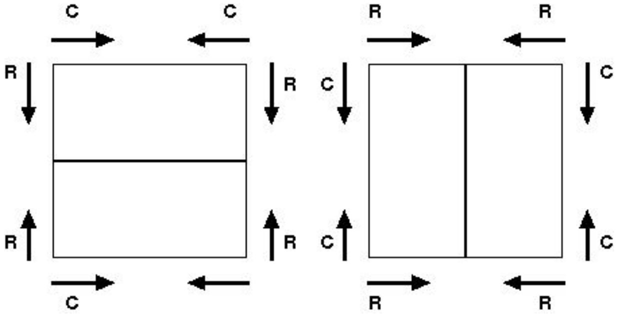

Figure 10.2: The row (R) and column (C) orientation for CCDs within a module. Each CCD chip has two readout amplifiers located at opposite ends of the serial register along the long edge of the chip. The rows and columns are numbered starting from 1 in each corner.

10.3 Transformation from Equatorial Coordinates to Focal Plane Pixels

Prior to launch, a transformation is needed that is sufficiently accurate to determine what targets fall within the *Kepler* field-of-view (FOV). The transformations described herein follow the method developed in Koch (73). The coordinate transforms, excluding optical and velocity aberrations, are coded in Koch (74). The transformation can be done in a series of steps: 1) a rotation to transform from *RA, Dec* to the center of the FOV, 2) a rotation to transform from the center of the FOV to the center of the CCD, 3) a transformation to correct for optical distortions, 4) a transformation to correct for differential velocity aberration across the focal plane, and 5) a conversion to module, CCD row, and column number, with field flattener correction. Each of these steps will be treated in detail below.

10.3.1 3–2–1 Transformation: Equatorial to Center of FOV

The first step is to convert from right ascension and declination (α, δ) to the center of the FOV. The transformation can be represented as a single rotation around a specific or *eigen* axis, or a series of three rotations in an orthogonal coordinate system. Following the notation in Wertz, ‘Spacecraft Attitude Determination and Control’ Appendix E, a 3-2-1 transformation is performed. That is, the spacecraft XYZ coordinates are initially aligned with the celestial sphere having the +X-axis at $\alpha = 0^{\mathrm{h}}$, $\delta = 0$, +Z-axis is at the north pole and +Y completes the right-handed coordinate system (at $\alpha = 6^{\mathrm{h}}(90^\circ)$, $\delta = 0$). 3-2-1 refers to rotations about the spacecraft +Z, +Y and then +X axes respectively.

The (current) selected FOV is at $\alpha = 19^{\mathrm{h}}35^{\mathrm{m}}50^{\mathrm{s}}$ $\delta = 34^\circ 40' 0''$. So the 3-rotation is 293.95833°, followed by a 2-rotation of −34.66667°. Finally to align the gaps with the bright stars a 1-rotation of 119.50000° is required. This last rotation is dependent on the observing season and will vary by multiples of 90° corresponding to the quarterly spacecraft roll maneuvers. The Euler angle rotation matrix or direction cosine matrix is given in Wertz (110) Table E-1.

To transform from RA and Dec to focal plane array (FPA) coordinates, the RA and Dec are converted to direction cosines:

$$\begin{aligned} \cos a &= \cos(\alpha) * \cos(\delta) \\ \cos b &= \sin(\alpha) * \cos(\delta) \\ \cos g &= \sin(\delta) \end{aligned} \tag{10.1}$$

These are then multiplied by the direction cosine matrix for the transformations, yielding the direction cosines in the transformed system $\cos a'$, $\cos b'$, $\cos g'$. Using the inverse of Equations 10.1 yields what we will call the longitude and latitude in the new (spherical) coordinate system with origin at the center of the FPA.

10.3.2 Optical Transformation

The transformation due to the *Kepler* optics can be broken into two parts: an axially symmetric transformation due to the Schmidt corrector and the primary mirror, and a position shift within a module due to the field flattener lens. The latter transform will be discussed in section 10.3.5. Aberrations from a Schmidt camera result in an axially symmetric redistribution of the light from a star, causing the measured centroid of the star to shift relative to that from an ideal imager. The change in position depends on the details of the optical design, but will be radial; that is, a star that would have appeared at some distance r_0 from the center of the FOV will appear at $r_0 + \Delta r$. The radial distance from the center of the FOV to a given target and the angle measured from the Y' axis are

$$\begin{aligned} r_0 &= \sqrt{Y'^2 + Z'^2}, \text{ and} \\ \tan\phi &= Z'/Y', \end{aligned} \tag{10.2}$$

respectively. The radial distance change will be of the form

$$\Delta r = a r_0^{\alpha}, \tag{10.3}$$

where a, and α are parameters that depend on the details of the Schmidt optics. The nominal plate scale for Kepler is 3.″98/pixel or 0.″1474/μm.

10.3.3 Transform Center of the FOV to CCD

The next step in the transformation is to determine which CCD chip the target coordinate falls on. As a first approximation, assuming that the modules are on a 2.8600° grid is sufficient to locate the center of the module on the sky. Once the distortions Schmidt optics have been characterized, the angular position of the center of each module relative to the center of the FOV will be substituted for the above approximation. Having found the appropriate module for a given target, a second 3–2–1 rotation from the center of the FPA to the center of the module is performed. The final rotation contains a term of 0°, 90°, 180°, or 270° to align the CCD chip rows with latitude and columns with longitude. The coefficients for this transform depend on the module and are tabulated in Koch (74). See Fig 10.2 for the orientation of the rows and columns within a module.

10.3.4 Velocity Aberration

The finite velocity of light causes the position of a star as seen from a moving observatory to differ from that seen by an observatory at rest. The effect is known as the aberration of light *(not to be confused with optical aberrations!)*. An object whose position makes an angle θ' to the moving observer’s velocity vector (see Fig. 10.3) will be seen at an angle θ in the rest frame given by

$$\tan\theta = \frac{\sin\theta'\sqrt{1-\beta^2}}{\cos\theta' + \beta} \qquad (10.4)$$

where, $\beta = V/c$ and c is the velocity of light (96). The effect is largest at $\theta' = 90°$.

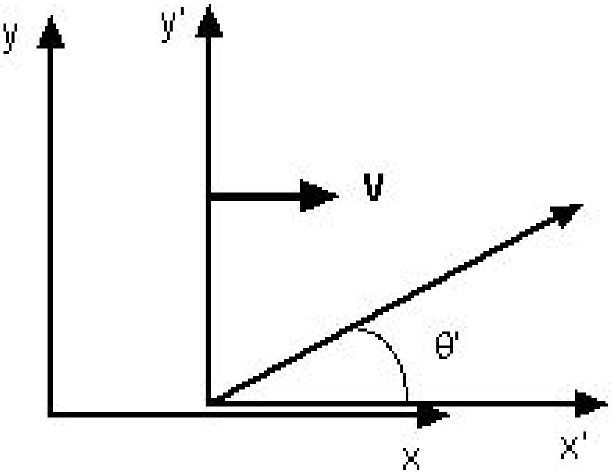

Figure 10.3: The coordinate systems used to define the position change of an object observed from a moving platform.

The apparent position shift (apparent - true) of a star at right ascension and declination (α, δ) seen by an observatory moving with a barycentric velocity with components parallel to equatorial rectangular axes given by $(\dot{X}, \dot{Y}, \dot{Z})$ is given to first order in V/c by (55)

$$\begin{aligned} \cos\delta\Delta\alpha &= -\tfrac{\dot{X}}{c}\sin\alpha + \tfrac{\dot{Y}}{c}\cos\alpha \\ \Delta\delta &= -\tfrac{\dot{X}}{c}\cos\alpha\sin\delta - \tfrac{\dot{Y}}{c}\sin\alpha\sin\delta + \tfrac{\dot{Z}}{c}\cos\delta \end{aligned} \qquad (10.5)$$

The maximum shift in the apparent position of a star as seen from the moving Earth is ±20″ due to this effect. While the magnitude of the shift will be similar for *Kepler*, it will be taken out by the guidance system for the center of the FOV, assuming the guidance system is designed to minimize the change in position of stars on the four fine-guidance CCDs. Therefore, it is only the differential shift between the center of the FOV and an offset target that must be taken into consideration. The differential shift will vary throughout *Kepler's* orbit as the angle between the spacecraft velocity vector and the FOV varies. The differential position shift at the corner of the focal plane will be approximately ±2″, or approximately one half of a pixel. The differential shift for a target at (α, δ) measured with respect to the center of the field of view (α_0, δ_0) can be found from Eq. 10.5 using

$$\begin{aligned} d\alpha &= \Delta\alpha - \Delta\alpha_0 \\ d\delta &= \Delta\delta - \Delta\delta_0. \end{aligned} \qquad (10.6)$$

These offsets can be calculated for the mid-point time of a roll period in order to choose the target apertures. The offsets are small enough so that target apertures will not need to be updated between roll maneuvers. The offset calculated from Eq. 10.6 can be added to a target's (α, δ) to determine the precise sub-pixel location at a given time using the steps in section 10.3.5.

10.3.5 Pixel Identification

At this point the angles are small enough ($< 1°$) that the spherical latitude value is taken to be equal to a linear conversion to a column (3.980000 arcsec/pixel). The spherical longitude value has a cosine(latitude) correction included, however this is still very small (0.5″ = 1/8 pixel) at the extreme corner. *NOTE: DAC gets larger errors when using a linear conversion (∼ 3 pixels at the corners of a chip) We need to confirm the above assertion.*

Field Flattener Lens Corrections

The field flattener lenses on each module will cause optical distortions that can shift the centroid of a target. It is expected that these distortions will be very small and can be represented by a low order transformation in pixel offsets:

$$\begin{aligned} x' &= a_1 + a_2 x_0 + a_3 y_0 + \dots \\ y' &= a_4 + a_5 x_0 + a_6 y_0 + \dots, \end{aligned} \qquad (10.7)$$

where (x_0, y_0) are the un-aberrated pixel positions. The parameters of the transformation (a_i) will be characterized during test and commissioning.

Alignment Offsets

During assembly of the focal plane, there is no shimming or alignment process for the CCD detector chip assemblies (DCAs) and modules. Thus each DCA has some small amount of alignment offset in position and rotation from the nominal desired alignment. Once on-orbit the precise (to about $1/10^{\text{th}}$ pixel) relative orientation of each DCA will be measured based on known stellar catalog positions. The coordinate transformation routine (74) has made accommodations for incorporating these alignment offsets. DCA rotations will be taken into account when transforming from the center of the FPA to the center of each module, then a column and row offset will be added. For each DCA, a 39 pixel row offset is already incorporated to account for the central gap between the DCAs on each module. The uncertainty due to tolerance buildup is expected to be on the order of ±5 pixels in row and column locations.

Once the as-built offsets have been incorporated, a target's pixel location on the output is recalculated, since the exact location of the position relative to the split in the middle of the chip is now known.

10.4 Post-Launch Coordinate Transformation

After launch, we can measure the positions of known stars over the *Kepler* focal plane, allowing us to refine the focal plane to sky coordinate conversion. The refinements will occur at the module level; that is, the transformation to the center of the field of view will still follow the method of section 10.3.1. The center of each module on the sky (modulo 90° for the quarterly rolls) will be determined from the observed positions of known stars on the module, and the transformation to the observed center of each module will be as described in section 10.3.3. Within a module the transformation will be treated in two steps (48):

1. a projection from the spherical equatorial coordinate system to a cartesian set of "standard coordinates" (ξ, η) aligned with north and east,
2. a transformation from standard coordinates to pixel coordinates (x, y) taking into account rotations, alignment offsets, optical distortions, etc.

The transformation to standard coordinates for a star at (α, δ) on a module centered at (α_0, δ_0) is given by

$$\xi = \frac{\sin(\alpha - \alpha_0)}{\sin\delta_0 \tan\delta + \cos\delta_0 \cos(\alpha - \alpha_0)} \tag{10.8}$$

and

$$\eta = \frac{\tan\delta - \tan\delta_0 \cos(\alpha - \alpha_0)}{\tan\delta_0 \tan\delta + \cos(\alpha - \alpha_0)}. \tag{10.9}$$

The equatorial coordinates should be corrected for differential velocity aberration before applying Eqs. 10.8 & 10.9. That is, the catalog positions should be corrected to the apparent position at the time of the observation. The transformation from standard to pixel coordinates will be of the form

$$\begin{aligned} \xi - x &= c_1 + c_2 x + c_3 y + c_4 x^2 + c_5 y^2 + c_6 xy + \ldots \\ \eta - y &= d_1 + d_2 x + d_3 y + d_4 x^2 + d_5 y^2 + d_6 xy + \ldots. \end{aligned} \tag{10.10}$$

The coefficients c_i and d_i are solved for in a least-squares sense using the known equatorial and pixel positions of several stars per module. In practice the solution is generally iterative in that the location of the module center (α_0, δ_0) and the choice of reference stars can be adjusted to optimize the solution. The iteration steps involve determining c_i and d_i from a set of reference stars, comparing the measured and calculated positions of the reference stars, adjusting (α_0, δ_0), removing reference stars with high-σ position errors, and re-determining c_i and d_i. Once determined for a module, the coefficients should be stable barring thermally induced focus changes.

10.5 Aperture Selection for Data Transmission

The selection of target star apertures for data transmission is a key step in the data stream. The difficulty in aperture selection comes from the competing goals of minimizing the quantity of transmitted data while at the same time ensuring sufficient data to meet the optimal pixel weighting as described in

§4.2. Ideally only those pixels contained in the optimal set would be downloaded. However, this set of pixels is likely to change over the duration of the mission due to the variation in the absolute location of the starfield with respect to the pixels on which it is imaged (due to, for example, velocity aberration). This section describes a systematic approach to determining an aperture that will guarantee transmission of at a minimum the optimal pixels.

10.5.1 Aperture Selection Development Methodology

The following steps describe the basic steps to find the (almost) optimal set of pixel apertures.

1. Develop / obtain the PSF model that includes blurring in the corners of the arrays. Include blooming effects for $M_v \leq 9$ stars.

2. Develop / obtain a galactic model of the expected stars that will be seen in the *Kepler* field of view (FOV).

3. Develop / obtain a model of the expected variations in the centroid of the PSFs as a result of spacecraft motion.

4. Simulate the PSFs/starfield/CCD output.

5. Obtain a list of the optimal pixels for each star.

6. Apply the variations expected.

7. Obtain a list of the additional optimal pixels needed for each star.

8. Find a set of apertures that contains all of the optimal pixels described in step 7.

The method of accomplishing step #8 above is non-trivial and may require significant computational time. Exhaustive search methods may not be achievable. However, other approaches leading to sub-optimal results are possible. For example, Genetic Algorithms (GAs) have been shown to be relatively efficient at finding near-optimal results in multi-dimensional space searches. The key to a successful GA is to define the genes and the fitness function well in order to have the resulting solution be valid. Discussions as to the specific methodologies associated with GAs are beyond the scope of this document, but are mentioned here as one of several possible solutions to the aperture selection problem.

Possible fitness functions and values of interest include:

$$E = \sum_i P_i \tag{10.11}$$

$$E = \sum_i P_i \cdot f_i \tag{10.12}$$

$$E_{theoretical\,min} = \sum_i P_i^{orig} \cdot f_i \tag{10.13}$$

$$E_{practical\,min} = \sum_i P_i^{motion} \cdot f_i \tag{10.14}$$

where P_i is the number of pixels in aperture i, f_i is the (expected or actual) frequency of use for aperture i, P_i^{orig} is the number of pixels in the original optimal list, and P_i^{motion} is the number of pixels in the motion-included optimal list. The first equation would simply minimize the number of pixels in all apertures. The second equation would minimize the number of pixels transmitted to the ground for a specified aperture set. The third equation would express the optimal number of pixels transmitted given no spacecraft jitter. The fourth equation would express the theoretical minimum number of pixels transmitted given worst-case spacecraft jitter.

On a side note, it may be necessary to have certain relatively unique types of apertures defined. These seem to fall broadly into three categories. First, a long, column-wise aperture is needed to capture data associated with saturated pixels (the so-called blooming pixel data). Secondly, a generic, rectangular (square?) area to capture a relatively large portion of the starfield may be required. This may be useful in determining/estimating certain parameters like smear pixels and other virtual pixels. Finally, a generic perimeter region of space may be requested for any number of reasons, not the least of which is the opportunity for secondary observers.

Finally, there is a host of *a priori* information about the downlink apertures that should significantly assist in any efforts associated with them. For example, the set of 1024 downlink apertures have the following attributes:

1) Every aperture does not contain any "holes" or "branches", that is it is a convex, closed shape (as nearly as a sampled square grid can approximate such an underlying continuous shape).

2) Each aperture is symmetric about some axis (not necessarily the x or y axis) prior to quantization/sampling effects.

3) Each aperture has a corresponding aperture that is its mirror image in both dimensions (note that this means there are only 256 unique apertures because each one has 3 other ‘mirrors’ associated with it).

4) Each aperture has a corresponding rotation of 90, 180, and 270 degrees. By a similar argument as in #3 above, this means there are only 64 unique apertures.

5) The overall size of the aperture (total number of pixels) is dependent upon the total flux received from the corresponding target star.

This information, combined with the various opportunities to simulate a set of downlink apertures, should allow for the identification of a set of downlink apertures that can be used to reduce the amount of data that is transmitted from *Kepler* to the DSN without loss of information. This data reduction is more thoroughly addressed in Chapter 12 Management of Compression Parameters.

10.5.2 Aperture Selection Simulated Validation

This section will be completed at a later time.

Chapter 11

Quick Look Software

This chapter contains an overview of what functionality will continue in what was originally designated the ‘Quick Look’ software. Quick Look itself, as it was originally conceived, will no longer be implemented. Instead, its functionality will be moved to the SOC Science Processing Pipeline and designated as ‘First Look’.

11.1 General Quick Look Overview

There are any number of attributes that can broadly be applied to the various Quick Look routines. Probably the most important of them involves data being obtained from the DSN/MOC every 4 days. Once this data is received (and maybe it is incomplete, containing missing packets) an automated analysis should begin. This analysis will nominally take place on the most recent data as well as data that is up to 30 days old as a comparison baseline. The analysis routines should be performed on a group of fiducial stars of long cadences types which are designated prior to the beginning of the mission. The results should be compared to nominal parameter values and subsequently written to logs as well as displayed in a report. Such a report will be available on the web for viewing. Prudent precautions for access and security are assumed to be in place. Additionally, relative importance of the processing results should be assigned, allowing a follow-up engineer to quickly assess the presence or absence of a critical event or situation associated with *Kepler’s* performance parameters.

Initially, the Quick Look program was intended to run at the MOC. As a consequence there were a number of tables that were assumed to be provided to the MOC by the SOC. These tables included:

1. Target Definition Table
2. Photometric Aperture Definition Table
3. Photometric Operation Parameter Table
4. Requantization Table
5. Cosmic Ray Huffman Encoding/Decoding Table
6. Science Data Huffman Encoding/Decoding Table

Even though the SOC will now perform this analysis without the MOC, this same information needs to be available to the analysis routines that will be developed. For the purposes of this chapter, it is assumed that these tables are present and in a readily-accessible format and location.

11.2 Roll Maneuver Routines

The Quick Look-R software package will be designed to run after each roll maneuver by the FS, who will be responsible for writing and maintaining the code. It will assist in analysis of the attitude of the spacecraft. Because of the movement of the starfield, post-roll target definitions will need to be verified for correctness. Potentially, new target definitions may need to be generated. This information will be based upon the attitude quaternion (both intended and actual). The results of the analysis should be summarized in a report (written/saved to disk text and published/graphed on the web).

Because of the uncertainty in target positioning after a roll maneuver, it will probably be necessary to execute the Quick Look-R software on a full field image (FFI).

11.3 Focal Plane Analysis

The Quick Look-F software is tasked with analyzing the focus condition of the spacecraft. It also will be written, run, and maintained by the FS. The focus is pre-set before launch and is not expected to need modification. However, unknown conditions provide an impetus to at a minimum analyze it in the unlikely event that *Kepler* needs to adjust the focus (is able to re-focus).

The image quality is based largely upon the ability of the optics to focus the image, in this case the 112 square degrees of stars in the constellation Cygnus. One measure of this focus is called the Full Width Half Maximum (FWHM) of the PSF. Measuring the FWHM of a point source can be done in any number of ways. Here we choose to fit the data to a waveform that is representory of the PSF and find the width of the curve at one-half of its peak value (hence the term full width half max). This process is displayed in Figure 11.1. It is expected that detailed textual and graphical reports about the focus will be generated for each fiducial target, each module, and the *Kepler* FOV as a whole.

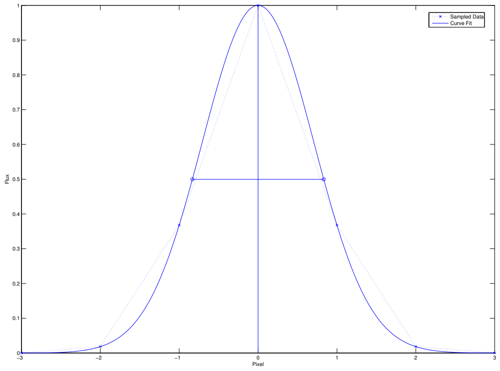

Figure 11.1: A one dimensional representation of the star flux data, the curve-fitted function, and the subsequent measurement of FWHM. Here the solid line is the curve that is a Gaussian with a peak of 1.0, a mean of 0.0, and a standard deviation of 1.0 is the 'best fit' for the underlying sampled data represented by a dotted line. The FWHM of the curve is measured as 2.0 (the horizontal line).

11.4 General Data Analysis

The vast majority of the processing functionality that was Quick Look has survived in the Quick Look-D program. Quick Look-D monitors fiducial targets for unexpected or significant changes in the following metrics:

Table 11.1: Metrics for Quick Look-D.

Parameter Name	**Possible Metric(s)**
Brightness	Absolute or Relative Change in Target Flux Value
Centroid	Absolute Location, Relative Movement over Time
Encircled Energy	Total Value, RMS Value, Drift Percentage
Background Pixels	Total Flux, RMS Flux
Black Level	Absolute Deviation in Black Level from 0
Smear	RMS Change in Total Smear Value
Trends	Significant Trends (Correlated or Uncorrelated to Ancillary Data)
Image Quality	Absolute and Changes in PSF on Local and Global Scales
Plate Scale	Stability of Plate Scale Parameters
Cosmic Ray Hits	Number of Hits per Unit Time, Contribution to CDPP

Chapter 12

Management of Compression Parameters

The flight software uses a simple algorithm to compress the pixel measurements and the cosmic ray counts for each pixel. The baseline algorithm is to remove the first measurement of 96-sample block from each pixel sample during that interval, and to Huffman code the residuals. A fixed Huffman code will be used and the code table must be delivered by the SOC to the MOC for upload. Similarly, the baseline approach to compressing the cosmic ray counts is to run length encode them and then Huffman code the run-length encoded counts. A Huffman code table for the purposes of coding the run-length encoded cosmic ray counts must also be provided by the SOC. Over time, the statistical properties of the pixel residuals or the cosmic ray counts may drift, necessitating updates to the onboard Huffman coding tables. The SOC must track the performance of the onboard compression algorithms and determine updates to the Huffman tables.

The chapter is organized as follows. In §12.1 we discuss fundamental limits to the compressibility of data and describe general compression processes. Autoregressive predictor filters are described in §12.2, which are then applied to study the compressibility of *Kepler*-like pixel time series in §12.3. The task of entropy encoding a residual pixel time series is described in §12.4, which focusses on Huffman codes. Finally, in §12.5 we consider the effects of data loss on the ability to reconstruct pixel time series from the coded bitstream.

12.1 Compression of Digitized Data

Seminal studies on the capacity of communication channels were performed in the 1920's by Nyquist (85, 86) and Hartley (49). However, Shannon (105) was the first to systematically study the capacity of noisy communication channels and the compressibility of signals and is considered the father of information theory. Shannon showed that there are fundamental limits to the compressibility of digitized data, and that a statistical analysis of digitized data is sufficient to determine the compressibility of such data. The number of bits required to represent a time series depends on the desired level of quantization noise, the ability to predict the process sampled by the time series, and the distribution of observation noise. For this discussion, we'll adopt the model for a compression algorithm given in Figure 12.1. Here, $\tilde{p}_n$ represents the raw co-added pixel values flowing from the focal plane electronics. The raw time series is first requantized, yielding the time series p_n, from which a predicted value, $\hat{p}_n$, is subtracted, yielding a residual, δp_n, which is entropically encoded and stored in the solid state recorder (SSR). If the requantized pixel time series can be predicted well, then the residuals should be near zero and have the same distribution as that of the observation noise. In the absence of observation noise (and quantization noise), then the data stream could be compressed into a very small package indeed, since no information need be transmitted aside from the initial timesteps necessary to initiate the perfect prediction process. In reality, the process $p(n)$ will not be perfectly predictable and the distribution of $\delta p(n)$ will depart somewhat from that of the observation noise.

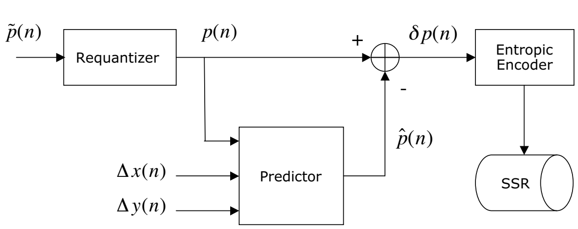

Figure 12.1: A flowchart for the process of requantizing, compressing and coding pixel time series onboard *Kepler*. The raw, 15-min co-added pixel time series $\tilde{p}(n)$ is requantized to control the quantization noise to a fixed fraction of the measurement uncertainty, yielding the time series $p(n)$. A filter which can take previous measurements into account as well as ancillary data such as pointing offsets $\Delta x(n)$ and $\Delta y(n)$ provides a predicted value $\hat{p}(n)$ which is subtracted from p(n), yielding the residuals $\delta p(n)$. These residuals are entropically encoded and then stored in discrete blocks in the solid state recorder (SSR).

The step of requantization is the only "lossy" one. The remainder of the steps are constrained to yield a coded bit stream from which the requantized pixel values, $p(n)$, can be perfectly reconstructed. Quantization noise is unavoidable in applications like *Kepler*, but its importance in terms of the total noise budget can be controlled in various places in the signal processing chain. The initial injection of quantization noise occurs with the digitization of the CCD pixel values from analog volts to digital counts by the analog to digital converter (ADC). The baseline design consists of 14-bit ADCs together with CCDs with well capacities of 1,000,000 e$^-$.[1] This implies that the minimum change in pixel brightness that can be resolved in a single measurement by the system is $1\times10^6/(2^{14}-1)$, or ∼61 e$^-$ per digital count. Now the quantization noise is the error between the analog value entering the ADC and the resulting quantized value when the latter is transformed back into e$^-$. The quantized value $p(n)$ is obtained effectively by rounding the analog value $\tilde{p}(n)/61$ to the nearest integer. Hence, quantization noise, the difference $\tilde{p}(n)-61\,p(n)$ can be modeled as a uniform random process with a minimum value of -30.5 e$^-$ and a maximum value of 30.5 e$^-$. The variance of this process is $61^2/12$ (see, e. g., 88). Note that at the top of a CCD pixel well, the root mean square (RMS) shot noise is ∼1,000 e$^-$, which is much larger than the initial quantization noise of 17.6 e$^-$. It makes no sense to retain a resolution of 61 e$^-$ for nearly saturated pixels with an inherent uncertainty of ∼1,000 e$^-$. Therefore, the proposed algorithm first requantizes the pixel time series so that the quantization noise is a small, fixed fraction of the inherent uncertainty in a pixel measurement.

To what value should we control the quantization noise? This question can be answered in part by considering the increase in the total noise budget due to quantization. Normalizing the measurement uncertainty, $\sigma^2_{measured}$ to one, we have

$$\sigma^2_{total} = \sigma^2_{measured} + \Delta^2_Q/12, \qquad (12.1)$$

where σ^2_{total} is the combination of the measurement uncertainty, $\sigma^2_{measured}$ and the quantization transition level, Δ_Q, normalized to the measurement uncertainty. We will assume that the time series under consideration has an observation noise characterized as a zero-mean, white Gaussian noise process. We note that Shannon (105) showed that for a given variance value, Gaussian noise has a higher entropy than any other distribution, and hence, requires more bits to code. In this sense, our assumption is conservative, but it is likely to be a good approximation in practice. The co-adding process will tend to produce time series that have Gaussian noise distributions, according to the central limit theorem (88). Equation 12.1 allows us to determine the impact of quantization noise on the total noise budget for a given Δ_Q.

A fundamental result due to Shannon (105) is that the compressibility of a message or data set is deter-

[1] An ongoing trade study is being conducted to investigate methods for preserving the full range of the CCD wells while improving the quantization noise at the dim end of the target star range. One possibility is to use a dual slope amplifier to increase the effective number of bits in the ADC. Another potential scheme is to use a second ADC that maps the full well depth of the CCD pixels following one which restricts its attention to the bottom portion of the well. In the event the first ADC "maxes out", the second ADC is called upon to digitize the pixel voltage.

mined entirely by its entropy, H, defined as

$$H = -\sum_{i=1}^{N} f_i \, \log_2 f_i, \tag{12.2}$$

where $\{f_i\}_{i=1,\ldots,N}$ are the relative frequencies of each symbol in a set of N symbols. For a zero-mean, σ^2-variance WGN process, the frequencies f_i can be evaluated as

$$\begin{aligned} f_i &= \int_{i-1/2}^{i+1/2} \sqrt{\frac{\Delta_Q}{2\pi}}\, e^{-x^2\,\Delta_Q^2/2}\, dx \\ &= \mathrm{erf}^*\left[\left(i+1/2\right)\,\Delta_Q\right] - \mathrm{erf}^*\left[\left(i-1/2\right)\,\Delta_Q\right], \end{aligned} \tag{12.3}$$

for $i = \ldots, -2, -1, 0, 1, 2, \ldots$, where

$$\mathrm{erf}^*(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-y^2/2}\, dy, \tag{12.4}$$

is a modified version of the standard error function, $\mathrm{erf}(x)$.

Figure 12.2 shows the entropy of a WGN process as a function of the relative increase in the total noise due to the quantization level, Δ_Q. Accepting modest amounts of quantization noise can significantly reduce the entropy, and hence, the number of bits required to represent a measurement. For example, if the quantization noise is allowed to increase the total noise budget by 10%, the entropy is only 1.5 bits. For a quantization level that inflates the total noise budget by 1%, the entropy is still a modest 3.1 bits. Table 12.1 lists various quantities of interest over a range of total noise budgets increases from 1% to 10%. The listed parameters include the total noise (relative to the original measurement noise), the quantization noise, the quantization level (relative to the measurement noise), and the entropy of the quantized WGN. These entropy values represent the best case values for a perfect predictor. Nevertheless, information theory provides the astonishing fact that experimental data can be quantized to as little as 1.5 bits per measurement for quantization noise that increases the total noise budget by only 10%. For *Kepler* this means that one day's worth of data for 200,000 stars might be compressed to as little as 0.67 Gbit. (For the sake of the discussion, we'll use the following assumptions in estimating onboard storage requirements: 200,000 target stars, 25 pixels per star, 96 measurements per day.) Thus, *Kepler* could conceivably store up to 95 days' worth of data on its baseline 64 Gbit SSR.

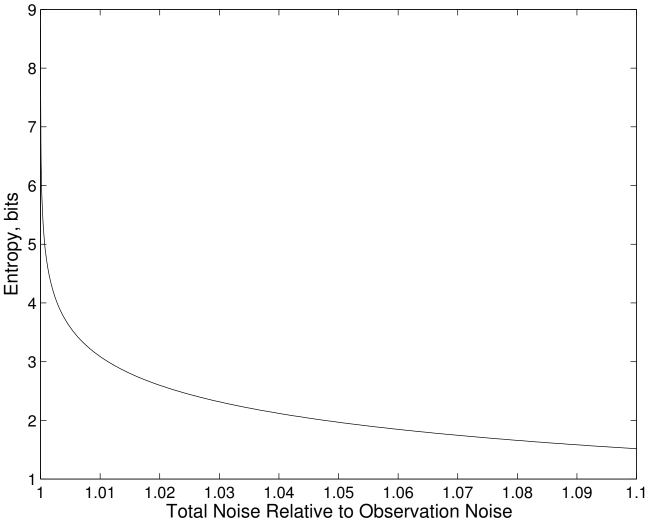

Figure 12.2: The entropy of a perfectly predictable digitized process as a function of the combination of observation noise and quantization noise. The total noise is the RSS of the observation noise and the quantization noise, normalized by the observation noise. An increase in the noise of the time series of 1% requires at least 3.09 bits per measurement, while an increase in the total noise by 10% requires only ∼1.5 bits to code each measurement. Table 12.1 tabulates various parameters for a range of total noise values from 1.01 to 1.1.

Requantization can be implemented simply by constructing a table of reconstructed pixel values with the jumps between neighboring values being dictated by the shot noise and the desired quantization noise. A lookup table would be used onboard that maps each possible pixel value to the index corresponding to the closest 'allowed' pixel value in the table of reconstructed pixel values. Assuming 312 exposures per 15 minutes, and setting the quantization noise to one quarter the shot noise, there would be a total of 40,722 possible reconstructed pixel values spanning counts from 0 e$^-$ to the maximum, 312×10^6 e$^-$. Thus, the initial step of requantization achieves a modest degree of compression in itself by

Table 12.1: Compression Parameters for Predictable Processes

Total Noise	Quantization Noise	Quantization Level	Entropy Bits
1.01	0.1418	0.4911	3.0874
1.02	0.2010	0.6963	2.5980
1.03	0.2468	0.8549	2.3160
1.04	0.2857	0.9895	2.1188
1.05	0.3202	1.1091	1.9682
1.06	0.3516	1.2179	1.8468
1.07	0.3807	1.3186	1.7456
1.08	0.4079	1.4131	1.6592
1.09	0.4337	1.5024	1.5838
1.10	0.4583	1.5874	1.5171

reducing the word size from 23 bits to avoid overflow to 15.3 bits to represent the requantized pixel values. (If the exposure times were 4 seconds with a 0.5 second readout, then there would be 200 co-adds per 15-min frame and 32,592 values in the lookup table, reducing the word length to 15 bits to represent a requantized pixel value.) The following section describes a realizable predictor for *Kepler*-like pixel time series.

12.2 Predicting Pixel Time Series

In this section we discuss the problem of developing a predictor filter for the purposes of compressing *Kepler* pixel time series. As discussed previously, the operating environment for *Kepler* should provide for very little change in the telescope attitude over time scales of days. The changes in pixel brightness due to low frequency jitter should be small, but may be significant in terms of achieving optimal data compression. Intrinsic stellar variability is a concern, but not for stars exhibiting solar-like variability, which is quite small on timescales of interest for compression. To address these sources of variability we'll introduce the following model for the generation of a pixel time series as a function of pointing offsets in x and y denoted by Δx and Δy, respectively. Additionally, we'll assume that the stellar variability is predictable in the sense that a current measurement can be predicted from a linear combination of M previous time samples:

$$p(n) = \sum_{i=1}^{M} a_i\, p(n-i) + b\,\Delta x(n) + c\,\Delta y(n) + w(n), \tag{12.5}$$

where a_i, $i = 1,\ldots,M$, b and c are real scalars defined by the dependence of $p(n)$ on its previous samples and the pointing offsets, and $w(n)$ is the observation noise. Aside from the terms involving $\Delta x(n)$ and $\Delta y(n)$ Equation 12.5 possesses the familiar form of an auto regressive (AR) model for a stochastic process (see, e. g., 50).[2] This is a highly flexible model for stochastic processes modeled as the result of passing noise through a linear filter. Such AR models have been used extensively and successfully in the analysis of a host of natural signals such as human speech and seismological signals. The power of this representation is that the AR model coefficients can be adjusted to match any arbitrary PSD, if M is sufficiently high. Additionally, AR models can

[2] For the purposes of this discussion we'll limit our attention to zero-mean, wide sense stationary (WSS) processes. The pixel time series under consideration can be modified to meet the first condition by subtracting a suitable estimate of the average value. They are also likely to be well-modeled as WSS over timescales of days. A stochastic process is WSS if its mean and variance are fixed quantities.

match narrowband PSDs with relatively few coefficients, compared to other modeling approaches. The AR approach seeks to determine the values of the parameters $a_i, i = 1, \dots, M$, b and c based on observations of $p(n)$, $\Delta x(n)$ and $\Delta y(n)$. As an example, consider the periodic signal of an eclipsing binary with a photometric period of ∼2 days modeled as a clipped sinusoid:

$$p(n) = 100 \min \left[cos \left(2\pi / 200n \right), 0 \right] + w(n), \quad (12.6)$$

where $w(n)$ is zero-mean, unit-variance, WGN. Were it not for the clipping operation and measurement noise, $p(n)$ could be represented exactly with 2 AR coefficients. Thus, in principle, such a sequence could be compressed to just four values: the two AR coefficients, and the first two data samples. The clipping operation in Equation 12.6 implies that there are significant harmonics above the fundamental frequency of 0.48 day^{-1}. However, the strength of the harmonics drops rapidly so that good prediction might be possible with only a few coefficients especially in the presence of measurement noise. Fitting a 2-parameter AR model to Equation 12.6, we find $a_1 = 1.356$ and $a_2 = -0.357$. The values for the coefficients are a strong function of the level of the observation noise. Were $w(n)$ absent, we would have $a_1 = 1.979$ and $a_2 = -0.98$. Suppose we wish to constrain the total noise to no more than 1.03 that of the observation noise. The entropy of the original sequence with observation noise and quantization noise is 5.27 bits, while that of the prediction residuals is 3.2 bits using 2 AR coefficients. Using more AR coefficients can improve the compression. For example, using 10 AR coefficients drops the entropy of the residuals to 2.8 bits.

Consider the unclipped version of this signal, that is, let $p(n) = 100 \cos \left(2\pi / 200n \right)$. Then the entropy of the original time series (quantized to a resolution of $\sqrt{12}/4$) is 7.5 bits, while that of the residual with $M = 2$ is 3.4 bits, and for $M = 10$, the entropy is 2.6 bits. Figure 12.3 shows the entropy of each time series as a function of the number of AR coefficients. Note that little improvement is obtained for $M \geq 10$. The difference in the performance of the AR modeling for the two sequences is due to the fact that the clipped sinusoid is not as well modeled as a linear filtered noise process: the clipping is nonlinear. Hence, the predictive filter has difficulty in predicting the values of the time series in the neighborhoods of the boundaries of the clipped portions of the time series. In the Fourier domain, this implies that a significant fraction of the power is distributed among the harmonics (∼16%) relative to the fundamental impulse at the true period (∼84%). Figure 12.4 shows the time series and the residual time series for these two signals for $M = 2$.

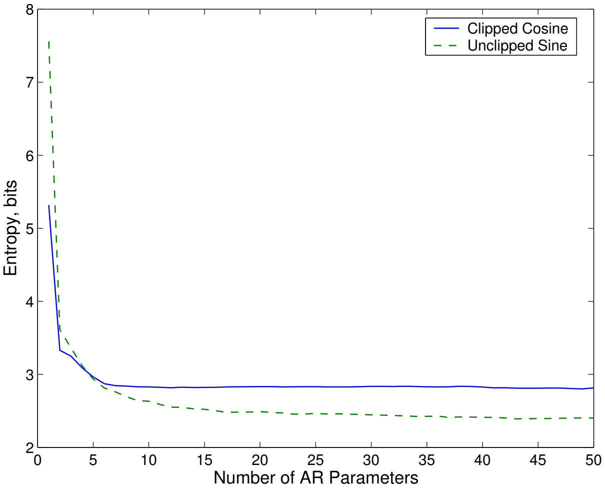

Figure 12.3: The entropy of the residuals of a clipped cosine with an amplitude of 100 and a period of ∼2 days, and that of an unclipped cosine of the same amplitude and period as functions of the number of AR parameters used to predict each time series. The quantization resolution is set to $\sqrt{12}/4$, which yields a quantization noise of 1/4 that of the observation noise. The entropy, or number of bits required to represent each time series, initially falls dramatically as the number of AR parameters is increased from $M = 0$, but levels off by $M \approx 10$ for the clipped cosine, and by $M \approx 20$ for the unclipped cosine.

This simple example demonstrates that AR modeling can significantly improve the compression of pixel time series which vary over long time scales. The predictive filter described in Equation 12.5 is simply an extension of the standard AR approach for time series in which the current value of a time series is a linear combination of previous values added

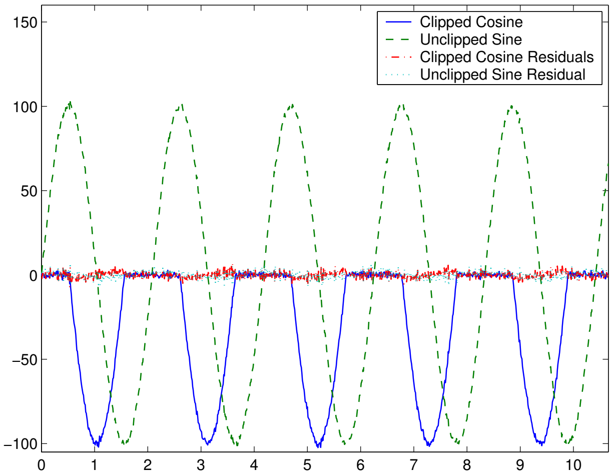

Figure 12.4: Time series consisting of a clipped cosine of amplitude 100 and a period of ∼2 days in unit variance observation noise, and an unclipped sinusoid of the same period and amplitude, along with the residuals of filtering each time series with 2-parameter predictive filter obtained from an autoregressive analysis. Since the residuals are confined to a small region about zero, they are easier to compress compared to the original time series.

to a random increment. The coefficients of Equation 12.5 can be determined nearly as easily as can those for a pure AR model. To make this explicit, consider the error signal, $e(n)$, given as

$$e(n) = p(n) - \hat{p}(n), \tag{12.7}$$

and consider the mean square error

$$\begin{aligned} E &= \left\langle e(n)^2 \right\rangle \\ &= \left\langle [p(n)-\hat{p}(n)]^2 \right\rangle \qquad (12.8) \\ &= \left\langle \left[p(n) - \sum_{i=1}^{M} a_i\, p(n-i) - b\Delta x(n) - c\,\Delta y(n)\right]^2 \right\rangle, \end{aligned}$$

where $< \cdot >$ is the expectation operator. Taking the derivative of Equation 12.9 with respect to a_k and setting it to zero, we have

$$\begin{aligned} 0 &\equiv \partial e(n)/\partial a_k \\ &= \left\langle 2\left[p(n) - \sum_{i=1}^{M} a_i\, p(n-i) - b\,\Delta x(n) - c\,\Delta y(n)\right] p(n-k)\right\rangle \\ &= \langle p(n)\,p(n-k)\rangle - \sum_{i=1}^{M} a_i \langle p(n-i)\,p(n-k)\rangle - \\ &\qquad\qquad b\,\langle \Delta x(n)\,p(n-k)\rangle - c\,\langle \Delta y(n)\,p(n-k)\rangle \qquad (12.9) \\ &= R_p(k) - \sum_{i=1}^{M} a_i\, R_p(k-i) - b\, R_{p\,\Delta x}(k) - c\, R_{p\,\Delta y}(k), \end{aligned}$$

where R_p, $R_{p\,\Delta x}$ and $R_{p\,\Delta y}$ are the autocorrelation function of $p(n)$, the cross correlations between $p(n)$ and $\Delta x(n)$ and $\Delta y(n)$, respectively, and we assume that the observation noise and the jitter offset time series $\Delta x(n)$ and $\Delta y(n)$ are uncorrelated with each other. Similarly we obtain the following equations for the partial derivatives of E with respect to the jitter offsets

$$\begin{aligned} 0 &\equiv \partial e(n)/\partial b \qquad (12.10) \\ &= R_{p\,b}(0) - \sum_{i=1}^{M} a_i\, R_{p\,\Delta x}(i) - bR_{\Delta x}(0) - cR_{\Delta x\,\Delta y}(0) \end{aligned}$$

and

$$\begin{aligned} 0 &\equiv \partial e(n)/\partial c \qquad (12.11) \\ &= R_{p\,c}(0) - \sum_{i=1}^{M} a_i\, R_{p\,\Delta y}(i) - bR_{\Delta x\,\Delta y}(0) - cR_{\Delta y}(0), \end{aligned}$$

where $R_{\Delta x\,\Delta y}$is the crosscorrelation of $\Delta x(n)$ and $\Delta y(n)$. Combining Equations 12.10, 12.11 and 12.12 together into a matrix form, we have

$$\mathbf{R\,a} = \mathbf{r}, \tag{12.12}$$

where the matrix $\mathbf{R}$ is given by

$$\begin{bmatrix} R_p(0) & \dots & R_p(M-1) & R_{p\,\Delta x}(1) & R_{p\,\Delta y}(1) \\ R_p(1) & \dots & R_p(M-2) & R_{p\,\Delta x}(2) & R_{p\,\Delta y}(2) \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ R_p(M-1) & \dots & R_p(0) & R_{p\,\Delta x}(M) & R_{p\,\Delta y}(M) \\ R_{p\,\Delta x}(1) & \dots & R_{p\,\Delta x}(M) & R_{p\,\Delta x}(0) & R_{\Delta x\,\Delta y}(0) \\ R_{p\,\Delta y}(1) & \dots & R_{p\,\Delta y}(M) & R_{\Delta x\,\Delta y}(0) & R_{p\,\Delta y}(0), \end{bmatrix} \tag{12.13}$$

$\mathbf{a}$ is given by

$$\mathbf{a} = \begin{bmatrix} a_1 & a_2 & \dots & a_M & b & c \end{bmatrix}^{\mathrm{T}}, \tag{12.14}$$

and

$$\mathbf{r} = \left[R_p(1) \quad \ldots \quad R_p(M) \quad R_{p\Delta x}(0) \quad R_{p\Delta y}(0)\right]^{\mathrm{T}}. \tag{12.15}$$

The solution to Equation 12.12 is trivial given the cross correlations R_p, $R_{p\Delta x}$, etc. Fortunately these can be estimated directly from the time series $p(n)$, $\Delta x(n)$ and $\Delta y(n)$ themselves. An alternate perspective views this problem as one of regressing the time series $p(n)$ in terms of delayed versions of itself and the time series $\Delta x(n)$ and $\Delta y(n)$. Defining the design matrix $\mathbf{A}$ as

$$\begin{bmatrix} p(1) & \ldots & p(M) & \Delta x(0) & \Delta y(0) \\ p(2) & \ldots & p(M+1) & \Delta x(1) & \Delta y(1) \\ p(3) & \ldots & p(M+2) & \Delta x(2) & \Delta y(2) \\ \vdots & \ldots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{12.16}$$

and the vector $\mathbf{p}$ as

$$\mathbf{p} = \left[p(0) \quad p(1) \quad \ldots\right]^{\mathrm{T}}, \tag{12.17}$$

we find the familiar least-squares solution

$$\mathbf{a} = \left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\right)^{-1}\left(\mathbf{A}^{\mathrm{T}}\,\mathbf{p}\right). \tag{12.18}$$

We note that numerically efficient on-line methods for determining the AR coefficients exist. Perhaps the most appropriate would be an adaptation of the Least Mean Square (LMS) algorithm, which has enjoyed much success and popularity in a variety of adaptive filtering applications (see, e. g., 50). This algorithm would allow the spacecraft to both learn and update the AR parameters efficiently without intervention by the Ground Segment, and without the need to compute inverses of correlation matrices. Of course, in this case the AR parameters would need to be downlinked along with the compressed pixel time series every time they were updated. Next we'll examine the compressibility of simulated *Kepler* data.

12.3 Compressing Simulated *Kepler* Data

This discussion draws significantly on analysis of synthetic pixel time series generated by modeling software called simkepccdpoly, which efficiently generates realistic, simulated data for one CCD channel. A collaboration between the author and Daniel Peters of Ball Aerospace & Technologies Corporation has led to the development of simkepccdpoly. We won't describe this software in detail, as it is described elsewhere, but will summarize its salient features (see, e. g., 63; 94). Simkepccdpoly incorporates realistic characteristics of the *Kepler* photometer, including pointing jitter, flight point spread functions (PSFs), CCD operating parameters, such as dark current, pixel size, charge transfer efficiency, exposure time, readout time, read noise, etc. The software also incorporates realistic astronomical information such as the density of stars with as a function of apparent magnitude down to 26th magnitude, and zodiacal light. Cosmic rays can also be injected based on a radiation environment analysis by Ball Aerospace & Technologies Corporation (Neil Nickles, personal communication). In the simulations discussed in this section, cosmic rays have been added to the pixel times series, but they have *not* been detected and removed either at the 15 minute level or at the individual exposure level. An important parameter for determining onboard storage requirements is the average number of pixels downlinked per star. Figure 12.5 shows the number of pixels per target star as a function of apparent magnitude for the required jitter PSD and the best focus PSF. The average number of pixels for stars brighter than 14th magnitude is only 15.2 pixels. The average number of pixels for stars brighter than 15th magnitude is 12.26 pixels. We'll adopt a conservative value of 20 pixels for our calculations.

We'll consider two different jitter PSDs in investigating the compressibility of *Kepler* pixel time series: the required forward sum PSD, and the predicted performance PSD. Throughout this discussion we'll assume that the requantization has been performed to limit the quantization noise to 1/4 that of the shot noise, so that the combination of shot noise and quantization noise is about 3% larger than that due to shot noise alone. Note that this does not include the effects of stellar variability.

First we'll examine a data set with target stars from 9th to 16th magnitude for jitter with a power

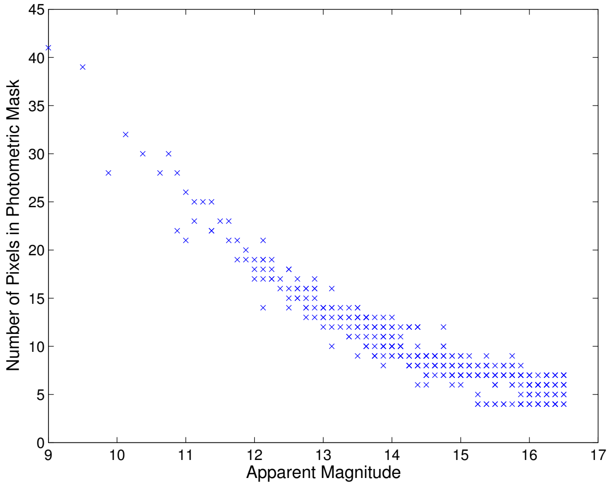

Figure 12.5: The number of pixels required per target star as a function of the apparent magnitude. The best focus PSF and the required jitter PSD were used to generate the synthetic data. An algorithm which performs a signal to noise ratio analysis determined the pixels to be used in constructing flux estimates for each target star. The average number of pixels per target star for the magnitude range between 9th and 16th magnitude is 10.4 pixels.

spectrum matching the required jitter PSD envelope. Figure 12.6 shows the entropy of target star pixels from this run ordered from the brightest to the dimmest pixel. The ensemble entropy of the requantized target star pixels in the absence of a predictive filter is 5.5 bits. With a predictive filter that has the form given in Equation 12.5 with M=0 (i. e., no dependence on previous samples other than removing an average value), the ensemble entropy is 2.6 bits. This is close to the theoretical minimum of 2.3 bits for the chosen level of quantization, and increases the data storage capability of the SSR by 220%. Figure 12.7 shows the corresponding pixel entropies for the performance jitter PSD. In this case, the difference between the entropies of the requantized pixel time series and those of the residual time series are less stark: the ensemble entropy of the former is 3.1 bits, while that of the latter is 2.6 bits. If attitude information is not used to predict and compress pixel time series, then the required jitter PSD is less amenable to compression than the performance jitter PSD. These simulations, however, do not include the effects of stellar variability, which might force us to use higher orders of AR modeling to obtain good compression ratios.

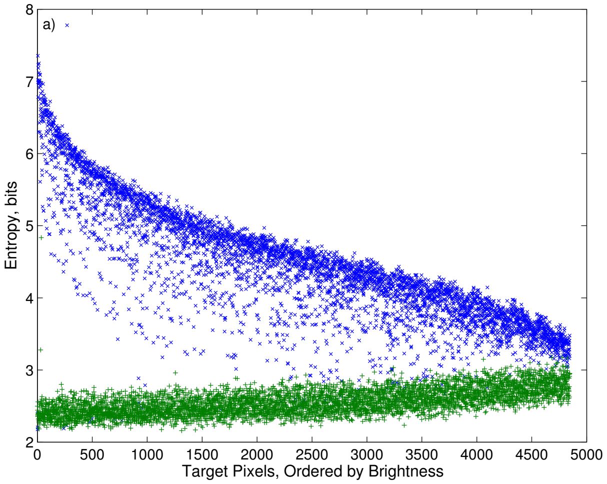

Figure 12.6: The entropy of pixel time series for *Kepler* target stars for the required jitter PSD. There are 500 stars spanning apparent brightnesses from 9th to 16th magnitude. The ‘x’s denote the entropy of each requantized pixel time series, while the crosses denote the entropy of the residual time series obtained by removing the dependence of the pixel brightnesses on the jitter. For the required jitter, the ensemble entropy of the pixel time series is 5.5 bits, while that of the residuals is 2.6 bits. For the performance jitter, the ensemble entropy of the pixel time series is 3.1 bits, while that of the residuals is 2.6 bits. In both cases the entropy of the residual pixel time series is close to the theoretical minimum of 2.3 bits for the chosen level of requantization.

We investigated the effect of stellar variability on the compressibility of *Kepler*’s pixel time series by adding segments of DIARAD/*SOHO* observations of the Sun from January 1996 into March 2000 to the pixel time series before quantization. For a description of these time series see e. g., Jenkins (60). We also scaled the amplitude of the solar variability by scale factors ranging from as low as 1/100 to as high as 100. Figures 12.8 and 12.9 show the entropy as a function of the scale factor used to amplify the solar variability segments for four different predic-

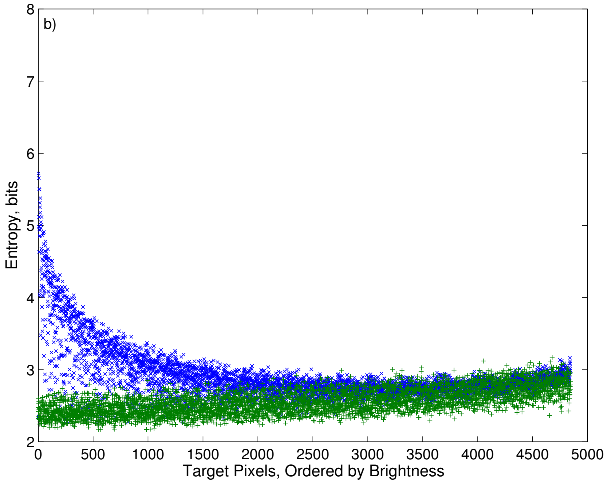

Figure 12.7: The entropy of pixel time series for *Kepler* target stars for the performance jitter PSD.

tion filter schemes. The first filter is a first difference filter incorporating no jitter information. This is essentially an M=1, $c = d = 0$ AR filter and hence, can track time-varying processes since it is a simple high pass filter. It can also amplify high frequency noise, though. The second filter is an $M = c = d = 0$ filter: only the average of each pixel time series has been removed. Its only ability to track time-varying processes is provided by the block size used to partition the data for downlink. The third filter is an $M = 0$, $c = d = 1$ filter that accounts for linear terms in jitter, and for which the average value of each pixel time series has been removed, as in the second filter. The fourth filter has $M = 5$, and $c = d = 1$. For scale factors from 0 to 2 or 3 solar, the latter two filters yield comparable results that are significantly better than those for the first two filters. As the solar amplification factor increases, the entropies of the residual pixel time series for all four filters and for both jitter PSDs increase. As expected, the fourth filter outperforms the other three for all values of solar variability. The key result is that a filter can be found that limits the entropy of the residual time series to no more than ~5 bits for $M \leq 5$. Higher values of M would result in lower entropies. It is likely that the vast majority of target stars will exhibit photometric variability comparable to that of the Sun. Taking an amplification factor of 10 times solar as a conservative limit, we can expect to compress *Kepler* target star pixel time series to no more than ~3.5 bits, or 70 bits per star per 15-min period (for 20 pixels per star). Thus, for 200,000 stars, a day's worth of 15-min measurements should require no more than 1.25 Gbit, so that the SSR can hold 51 days of data.

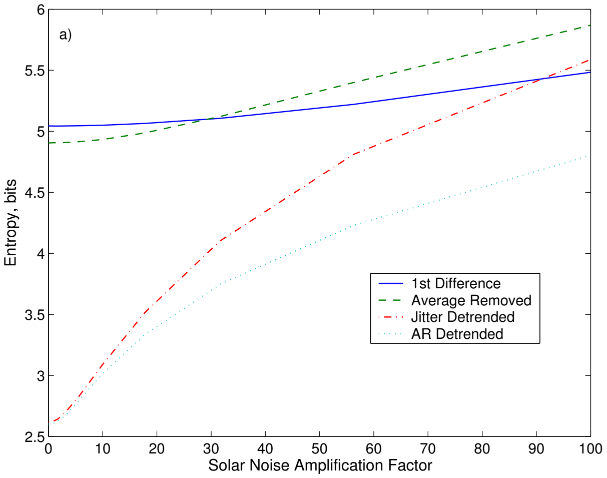

Figure 12.8: The entropy of pixel time series for *Kepler* target stars including the effects of amplified solar-like variability. The solar-like variability has been scaled from 0.01 to 100 times that of the original DIARAD/*SOHO* time series and added to each of 5,000 synthetic *Kepler* pixel time series for 500 target stars spanning apparent brightnesses from 9th to 16th magnitude. Four different predictive filters have been used to improve the compressibility of the residual time series. The solid curves denotes the entropy of the residuals obtained by applying a first difference filter to the requantized pixel time series. The dashed curves denotes the entropy resulting from merely removing the average values from the time series. The dash dot curves results from removing the average and linear terms in jitter offsets from each time series. The dotted curves denotes the entropy resulting from applying a 5-parameter AR filter to each time series including linear terms in jitter offsets. The required jitter PSD is included in the results displayed.

How would *Kepler* be able to implement AR predictive filters? Given that $M = 1$ or $M = 0$ AR filters with $c = d = 0$ result in admirable compression

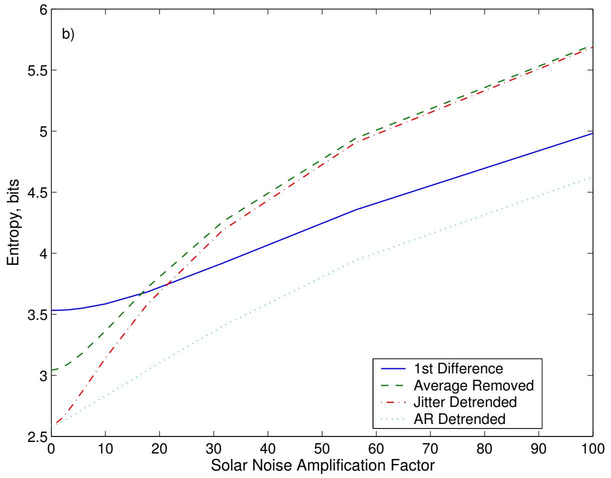

Figure 12.9: The performance jitter is included in the results shown. For the required jitter, the entropies range from ∼5 to ∼6 bits for the first two filters over the entire range of scaled solar-like variability. The entropy for the two filters which take into account jitter attain entropies of ∼2.6 bits for solar-like variability, and rise beyond 3 bits only for solar-like variability scaled up by a factor of 10. Clearly, the compressibility of *Kepler*-like pixel time series is dominated by jitter for the required jitter PSD. In contrast to the case of the required jitter PSD, the entropies for the performance jitter in b) for all four filters are at or below ∼3.5 bits for solar-like variability, rising to ∼ 5 bits only for solar-like variability scaled by at least a factor of 60 above that of the Sun. The compressibility of *Kepler*-like pixel time series is dominated by jitter for the performance jitter PSD for solar-like variability, but this dominance relaxes for stellar variability far higher than that of the Sun.

rates, I would suggest that the flight software be initialized in one of these two configurations. Once several weeks of data have been obtained, it would be possible to estimate the AR parameters from the downlinked pixel time series and upload them to the spacecraft. Given the DIARAD*SOHO* observations of the Sun, and less precise observations of variable stars from ground-based instruments, I believe that the AR parameters for most stars would be slowly-varying. Indeed, the effects of differential velocity aberration may force the updating of the AR parameters rather than changes in the behavior of individual stars. The jitter information can be supplied either by the ADCS or by centroiding of bright fiducial target stars in the FOV.

It is important to note that a great deal of flexibility exists for implementing the AR filters. Since the outputs are restricted to be integer time series, the arithmetic need not be floating point. The goal is not to perfectly predict the time series, but rather to predict them well enough so that small residuals are obtained. We note that the AR predictor proposed here is relatively insensitive to errors in the AR coefficients. For solar-like variability, the ensemble entropy increases by only 0.1 bits for a predictor with 10% errors in the AR parameters, and by 0.9 bit with 50% errors in the AR parameters. For 10× solar variability the increase in entropy is 0.2 bits and 1.2 bits for 10% and 50% errors in the AR parameters, respectively. The predictions could also be used to identify and remove cosmic rays at the 15-min level if that were desirable. It cannot be overemphasized that the only penalty incurred by using suboptimal AR coefficients or by drifting AR coefficients is that the level of compression will be somewhat degraded relative to the achievable level. Data will not be lost in any case due to less efficient prediction. The validity of any implemented predictive filter will be monitored closely by the Ground Segment. Indeed, any updates to the predictive filter tables can only be generated from an analysis of downlinked pixel time series. This task is straightforward and requires only modest computing resources. We turn briefly to entropic encoding before concluding this article.

12.4 Entropic Encoding of Digital Data

Once an appropriate predictive filter has been specified, it is necessary to consider the task of encoding the residual pixel time series for storage and later transmission to the ground station. This is the job of an entropic encoder which takes advantage of the non-uniformity in the distribution of a class of data to realize the promise of the compression implied by its entropy. We will briefly describe the classic Huffman coding scheme, which is but one of several possible entropic encoders.

A Huffman code examines the frequency of all possible symbols and designs bit strings for each one in such a way so that the most likely symbols are assigned the shortest bit strings and the least frequent symbols are assigned the longest bit strings (see, e. g., 59; 91). To do this, a tree is constructed that contains each symbol as a leaf. The two least frequent symbols are first combined to form a node on the tree, and their relative frequencies are summed. At the next step, the two least frequent nodes (including leaves, since they are nodes, too) are combined in the same way and the process continues until only two nodes remain, which are combined into the root of the tree. To determine the code for a particular symbol, the tree is traced from that leaf up to the root, and a bit is assigned at each branching node, with left branches assigned a '0' and right branches assigned a '1' (or vice versa). The bit string assigned a symbol is the string of branching labels tracing the route from the root to the leaf. The least frequent symbols have the longest journey to the root, so they are assigned the longest bit strings. The most frequent symbols have the fewest branch nodes between them and the root and so receive the shortest bit strings. To decode a message, one begins at the root, traveling down the tree as bits are read in (branching left for a '0' and right for a '1') until a leaf is encountered, which determines the next decoded symbol that was transmitted. The pointer is reset to the root and the process begins anew for the next bit.

The Huffman code achieves the lowest bit rate on average of all possible coding schemes. Figure 12.10 shows the length of each codeword as a function of both the requantized pixel value and the residual pixel value for simulated *Kepler* pixel time series for the required jitter PSD. Huffman coding is fairly efficient. The actual average number of bits necessary to implement the Huffman code was 4.97 bits for the requantized values and 2.65 bits for the residual values obtained with $M = 0$ and $c = d = 1$. These entropies are very close to the theoretical values based on the distributions: 4.93 bits and 2.61 bits, respectively. A linux box at Ames codes each 16-bit, 1027 by 1024 difference frame in 0.53 sec and decodes each difference frame in 0.44 sec. There are alternatives to Huffman coding. Perhaps the most likely one is arithmetic coding, which represents each message as a real number between 0 and 1.

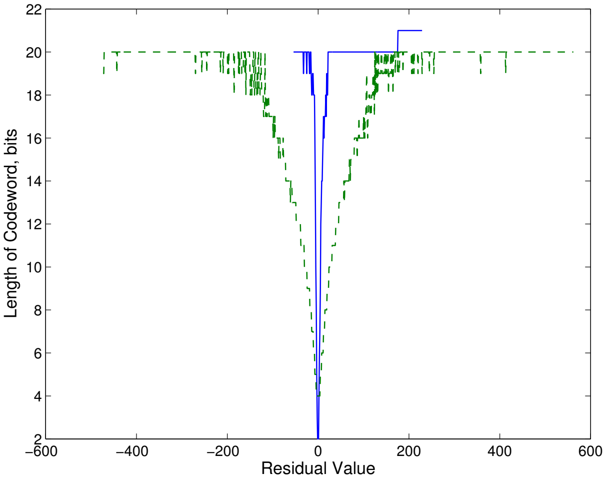

Figure 12.10: The lengths of codewords for Huffman codes for simulated *Kepler* data for the required jitter PSD. The solid curve shows the codeword lengths for the residual pixel time series having jitter removed, while the dashed curve shows the codeword lengths for the requantized pixel time series. The range of pixel values is smaller for the residuals than it is for the requantized values so that there are fewer codewords. The asymmetry in the ranges about zero are due to undetected cosmic rays, which can only inject positive charge. Note that the realized entropies are 4.97 and 2.65 bits for the requantized and residual values, respectively. The corresponding theoretical values are 4.93 and 2.61 bits, based on a histogram analysis of the synthetic data sets.

Note that any message of finite symbols can be represented this way if the number is represented in the radix corresponding to the number of possible symbols. For example, if there were 10 possible symbols, ('0'-'9'), the message 0.01243259 would represent the sequence '01243259'. In arithmetic coding, the unit interval is broken into subintervals corresponding to each symbol of length proportional to its relative frequency. The first symbol is encoded by choosing a number lying in its corresponding interval. The second symbol is encoded by break-ing the first interval into a set of subintervals in the same way as the interval $[0,1)$ was partitioned, and choosing a number within the sub-subinterval corresponding to the second symbol. So if the interval of the first symbol was $[0.1,0.2]$, and the interval of the second symbol was [0.9,0.91), then the message would be bounded on the bottom by $0.1+(0.2-0.1).9 = 0.19$, and on top by $0.1+(0.2-0.1).91 = 0.191$. The compression is achieved by noting that smaller intervals take more digits to represent than larger ones. As symbols are coded, the number of significant digits which must be kept around grows, but as the interval containing valid messages shrinks, the number of digits that change diminish, too, so that the message-number can be left shifted out of the register as the process continues. A message is decoded in a similar fashion, digits are read in until a unique symbol interval is assured (because of the bounds on the message read in so far), the symbol is added to the decoded message and the process continues. Both Huffman coding and arithmetic coding are standard entropy encoders and everyone who uses a browser or image display program has certainly used one or the other, since both coding schemes are part of the JFIF standard for JPEG encoders/decoders.

12.5 Sensitivity to Data Loss

The approach to data compression detailed in the previous sections does not take into account the effects of data loss on the ability to reconstruct the requantized pixel sequences. In this section we argue that careful design of the "packaging" of the entropy encoded pixel time series can mitigate propagation of data loss from packets lost in transmission to other packets successfully transmitted. Additionally, compression of the data minimizes exposure to data dropouts during transmission and reduces the time required to successfully transfer a data set completely intact. First, however, we'll discuss the philosophy of data compression versus data transmission over a noisy communication channel.

The goal of an ideal data acquisition instrument is to accumulate non-redundant information about some phenomenon or process. Indeed, a huge amount of effort is placed on isolating independent measurements from redundant, non-independent measurements in remote sensing and other applications. For *Kepler* the ideal instrument would be one that analyzed the photons entering the aperture and transmitted the identities of stars harboring planets, together with the planetary orbital periods, transit depths, etc. This would require much more processing power than *Kepler* is capable of, not to mention much more faith than the science team is willing to place on the design of a flight version of the data analysis pipeline. The goal of detection and error correction (DEC) coding is to add bits to each block of data so that transmission errors can be detected and corrected. In mitigating the effects of a noisy channel note that DEC requires just as many additional bits to protect a block of redundant data as it does to protect a block of non-redundant data. Data which have been losslessly compressed and DEC-coded can be transmitted in far less time over the noisy channel, minimizing the exposure to transmission errors. Further, a smaller data set allows for more powerful DEC-coding so that successful transmission can occur with smaller gain margins.

For the *Kepler* data stream, let us assume that each pixel is compressed to 3.5 bits on average, and that there are 200,000 stars consisting of 20 pixels on average. In this case, four days' of data can be coded in 5 Gbits. The current assumption is that the 64 Gbit SSR holds 9 days' of data, and that 5 days' of data can be transmitted in 6 hours. This implies a data transmission rate of 5.92 Gbit hr^{-1}, so that four day's of compressed data can be transmitted in 41 min. Assuming that the data link allows 90% of the packets to be successfully downlinked in a single transmission, there is time to transmit the 4-day block of data 8.8 times in a single Deep Space Network (DNS) pass. Assuming the data dropouts are independent, then rather than losing 10% of the data, only 1.29×10^{-9} of the data is lost. As there are $\sim 5.4 \times 10^{9}$ bits of data, that implies 7 bits are lost. This is not the optimal approach to minimize data loss, however. If selective retransmission is available, then only lost packets need be retransmitted so that a loss rate of 1×10^{-20} can be realized in

45 min, neglected the time needed to request the retransmission of selected packets. Some combination of selective retransmission and multiple transmission would optimize the time to reach any given loss rate. For example, suppose that the spacecraft is 0.25 AU away so that the light time delay is 4 min from the time a request is made to when the selected data is received at the DSN. The initial transmission of 4 days of data takes 41 min followed by 4 min to request and begin receiving the first selected lost packets. At this point it probably makes most sense to request retransmission only once more for the 1% of the data that failed to make it through, and to repeat the transmission of these packets 17 times. Each transmission of these packets takes $\sim$4 s, for a total of 59 min $(41+4+4.1+4+.41\cdot 17)$. Compressing the data makes it easier to successfully downlink a completely intact set of data.

Although the foregoing discussion illustrates why compression is a good idea, we still need to consider the effect of lost bits on the ability to reconstruct intact neighboring packets. For AR predictors with $M = 0$, this is not an issue since reconstruction requires knowledge of only the residual, the offsets $\Delta x(n)$ and $\Delta y(n)$ for that timestep (except for $c = d = 0$, in which case the attitude offsets are unnecessary), and the average pixel value, $\bar{p}$, that was removed from the entire block of data. It's true that the 'full word' or average that was removed is essential, but this can be mitigated by also transmitting the difference between the last pixel value in the block of data and the next $\bar{p}$ value to be used. That way, if the 'full word' is lost for any reason, the full word for the previous or the following block can be used to reconstruct the data samples. This idea carries over to AR predictors with $M \geq 1$. Rather than transmitting one requantized pixel, a set of M values are transmitted. At the next block boundary, the backwards residual from the last pixel value to the next set of M initial pixel values is transmitted as well, to insure that the pixel time series can be reconstructed backwards from the following block in case a gap occurs. In case $c \neq 0$ and $d \neq 0$, it will be important to pay special attention to successfully transmitting $\Delta x(n)$ and $\Delta y(n)$, however they are constructed on board.

This leads to a consideration of how big to make the blocks. For M=0 predictors the block size determines the adaptability of the system to stellar variability. For solar-like stars most of the variability occurs on timescales $\gg$1 day. Having a block size of 1 day then, should be adequate to track changes in stellar variability for such stars. This is supported by the simulations discussed in §12.3. For $M = c = d = 0$, a block size of 1 day implies $\sim$1% overhead for the compressed data (prior to DEC coding). In these cases, the number of data dropouts doesn't matter in the sense that so long as $\bar{p}$ (and perhaps $\Delta x(n)$ and $\Delta y(n)$ are transmitted), lost data packets don't impede the ability to properly interpret intact data packets. For $M \geq 1$, however, this is not the case. The AR predictors described here can reconstruct pixel values either forwards or backwards to a gap from the initial pixel values at a block boundary, but not across a gap. The block size in this case must be chosen so that it is highly unlikely that more than one gap in the data stream will occur. This is a straightforward engineering task, so long as the data channel is properly characterized, but is beyond the scope of this document. Given that the size of the data sets to be downlinked are rather small, this issue should not be a problem.

Chapter 13

Management of Cosmic Ray Rejection Parameters

This chapter discusses the management of the cosmic ray rejection software parameters for the flight software. The cosmic ray rejection algorithm for the flight software attempts to identify cosmic ray hits, which deposit photoelectrons in affected pixels in the FOV. The anticipated flux rate of cosmic rays is 5 cm^{-2} s^{-1}, so that on average, each 27 μm pixel receives a direct hit $\sim$3 times per day. However, cosmic ray events deposit charge over a range of pixels depending on the inclination angle of the ray to the CCD. The information available in flight is restricted to the current pixel measurement and the previous 15-minute average of the pixel value. This is sufficient to design an optimal detection threshold, assuming the distribution of the cosmic ray-injected photoelectrons is known, and that the individual exposures are dominated by shot noise and a fixed read noise. This is not the case, however, as pointing offsets on the 3-sec timescale are likely to be important for a majority of the pixels of interest. The task is to define a practical threshold based only on the difference between the current pixel measurement and the average value from the previous 15-min frame. This task appears to be manageable. The SOC shall provide the detection thresholds based on modeling efforts preflight. These shall be updated once actual flight data is acquired.

This chapter was written and included prior to the baseline change of removing the cosmic ray detection and rejection from being an on-board processing step. At this time the cosmic ray detection and rejection processing is expected to be done on the ground. Because this is a recent change, very little work has been done to modify the contents of this chapter to make it directly applicable to this new processing venue.

The chapter is organized as follows: In §13.1 we discuss the energy distribution of the cosmic ray flux and the method by which we transformed this distribution into the distribution of charge deposited per pixel per event. A summary of the results and a set of recommendations is set forth in §13.2.

Although the detection and removal of cosmic rays on board the *Kepler* spacecraft is no longer planned, the method described in this chapter is extendable to a ground-based detection and removal methodology and will likely be developed for inclusion in the SOC Pipeline or other processing venue.

13.1 The Cosmic Ray Flux

The cosmic ray flux environment has been of great concern to almost all space missions with CCDs that are sensitive to cosmic rays. The actual flux experienced by a device depends a great deal on the exact orbit, that is, is the spacecraft in low Earth orbit (LEO), or is it in deep space? The flux also depends on the shielding and configuration of the detectors within the spacecraft, which can affect the generation of secondaries from primary events. In any case, the *Kepler* Mission has adopted a flux rate of 5 cm^{-2} s^{-1} based on previously flown missions in similar orbits, such as *SOHO*.

A study was conducted at Ball Aerospace Technologies Corporation (BATC) to derive the distribution of total charge deposited into a CCD for each cosmic ray hit (Neil Nickles, personal communication). The results are displayed in figure 13.1, which shows a mode of ∼2500 e⁻, little or no charge below ∼2000 e⁻, and a long upper tail trailing out to at least 100,000 e⁻. We note that 90% of events deposit less than 6,200 e⁻ into a CCD. To put this into perspective, note that an m_R=12 star occupies about 25 pixels, and that over 6.5 hours, about 4×10^9 e⁻ accumulates in its aperture. The shot noise for such a star will be 63,245 e⁻. Now, 25 pixels receive a cosmic ray flux rate of 21.3 per 6.5 hr interval. Since the occurrence of cosmic rays follows a Poisson distribution, then the standard deviation of the number of events in the star's aperture is 4.55. Assuming that 2,500 e⁻ are deposited with each event, then the noise from the cosmic rays in a 6.5-hr interval is 4.55 times 2,500 or 11,384 e⁻. This most likely is too low, since the energy deposited by a single cosmic ray hit varies over such a large range. If we generate random deviates using the transformation method from the distribution given in Figure 13.1, then we obtain a standard deviation of 24,500 e⁻ for the charge deposited in each 6.5-hour interval, or about 6 ppm relative to stellar flux. Thus, the cosmic ray flux is expected to increase the noise budget by about 5% if nothing is done to detect and remove charge injected by cosmic ray events.

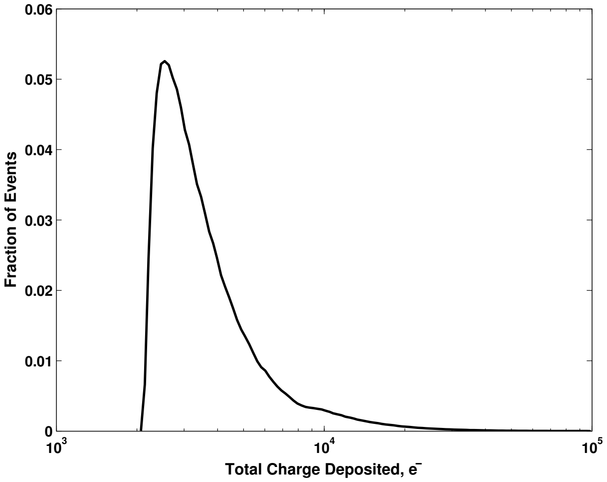

Figure 13.1: The distribution of the total charge deposited into a *Kepler* CCD per cosmic ray event. This distribution resulted from modeling taking into account the expected cosmic ray environment for *Kepler's* orbit, and a detailed structural model for the spacecraft, the instrument, and any planned radiation shielding (Neil Nickles, personal communication).

The point in time at which cosmic rays are most readily identified and removed is when each CCD pixel is read out, and before it is added to the running 15-min sum. *Kepler's* CCDs are 1132×1066 devices when the 32 virtual columns and 20 virtual rows are accounted for. For a 0.5 second readout time, then there is ∼400 ns or about 20 clock intervals to read out each pixel, detect cosmic rays, correct for any detected cosmic ray, and add the current value to the running sum. Furthermore, the tasks associated with operating the CCDs from the readout to the accumulation of 15 minutes of data will be implemented using FPGAs (field programmable gate arrays). Thus, the operations that can be performed are not only limited by the time interval available, but in the type of operation that can be performed, since FPGAs are not general purpose computers (i. e., no floating point operations).

Another limitation is that the only information available to detect cosmic rays (aside from the current pixel value), is the value for the previous 15-min co-add interval. The difficulty this presents is that there is not even an estimate of the variance of each pixel at the single exposure timescale, let alone knowledge of the distribution of the flux accumulated per pixel per exposure. Ideally, one would analyze the distribution of the pixel values for both cases of cosmic rays present and no cosmic rays, which together with the cosmic ray flux rate would allow for the determination of an optimal detection threshold for each pixel. So the major task to be addressed is whether a practical and effective cosmic ray rejection algorithm can be fashioned that uses only a moving 15-min average pixel value for both detrending and for determining the threshold value. In the absence of residual pointing offset errors by the Attitude Determination and Control System (ADCS), this would not be a problem. In that case, the statistics of each pixel's flux time series would be dominated by the Poisson noise associated with counting photons. The ADCS is not perfect, however, and the 3-sec to 3-sec variations in flux due to modulation of pixel brightness by pointing drifts on these short time scales might exceed or dominate those due to shot noise.

To study the issue of rejecting cosmic rays in individual exposures we need to know what the distribution is for the charge injected into individual pixels by cosmic ray events. To transform the distribution in Figure 13.1 to the desired one, we applied the following assumptions. 1) The total charge deposited is uniformly distributed over the path traveled by the the cosmic ray as it traverses the CCD slab. 2) The charge deposited by the cosmic ray diffuses the same way as is charge from actual photons. Given the geometry of the CCDs (27 μm × 27 μm × 16 μm), we traced random rays through a 13 by 13 pixel region of a CCD, distributing the charge in each pixel encountered by a ray according to the assumptions above. The CCD pixels were divided into 13×13 sub-pixels for the purposes of the numerical calculations. We amassed a catalog of 6,097 cosmic ray trails, normalized so that the sum of each trail was unity. Each trail, then, could be scaled by a random deviate drawn from the total charge distribution to model the effect of a single cosmic ray.

With this library of cosmic ray trails and the total charge distribution in hand (and the cosmic ray flux rate), we are in a position to simulate the effects of cosmic rays on a CCD image of any given exposure time. (We note that cosmic rays are accumulated during readout, too, so that physical pixels experience cosmic rays for the full exposure plus readout interval. We've ignored the fact that the diffusion of the charge for a cosmic ray even during readout might differ from that for one experienced during an exposure, as well as the differences in cosmic rays for virtual pixels, which only exist during readout.) While this information can be used to generate cosmic ray events with 'realistic' spatial features, there is no way to incorporate spatial correlations into the cosmic ray detection algorithm for the flight system. Thus, rather than using the 2-D library and the total charge distribution, we can simply histogram the charge injected into the pixels of a CCD for a given time interval (without any other flux), and then generate a distribution for the charge injected per pixel per unit time, independent of the spatial correlations. Simulated cosmic ray events can then be generated more rapidly from a single distribution using the transformation method than can be accomplished for the 2-D library and the total charge distribution. Figure 13.2 shows the distribution of charge from cosmic ray events accumulated per pixel per 3 sec interval.

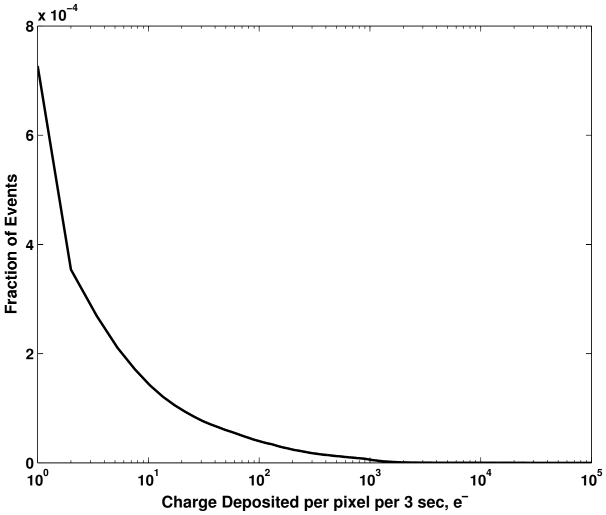

Figure 13.2: The distribution of the charge deposited into each *Kepler* CCD pixel per cosmic ray event per 3 sec.

The first question that can be answered from this distribution, is what is the noise added per pixel per 3 sec interval? The answer is ∼22 e^-, so that the noise for a m_R=12 star occupying 25 pixels over a 6.5-hr interval is 9714 e^-, or 2.4 ppm relative to the stellar flux, assuming that the noise introduced by cosmic rays is independent from pixel to pixel. This value lower than the value obtained by looking at the charge accumulated in an aperture by a factor of 2, indicating that on average, 4 pixels are affected by a given cosmic ray. This correlation must be considered when calculating noise in a given aperture from noise on individual pixels. The question to be answered now, is what can be done to reduce the amount of noise introduced by cosmic rays by working at the single exposure timescale?

To answer this question, we performed a set of simulations using the End To End Model (ETEM) developed by a collaborative effort by J. Jenkins (SETI Institute) and Dan Peters (BATC). We used the forward sum required ADCS jitter Power Spectral Density (PSD) developed for the Concept Study Report (CSR). While the current required jitter PSD is significantly lower than that of the CSR, our aim was to obtain a conservative estimate of the power to discriminate against cosmic rays in the face of 3-sec to 3-sec image motion. It would be entirely appropriate to perform the calculations outlined in this paper to simulations obtained using either the performance jitter PSD or the refined and improved required jitter PSD. The simulations generated pixel time series for over 4500 pixels corresponding to the pixels of interest for 495 target stars in a synthetic star field on a single CCD channel. Figure 13.3 shows a histogram for the ratio of the standard deviation of the synthesized pixel time series due to pointing offsets (i. e., without any stochastic noise added) to the shot noise for the target star pixels on time intervals of 3 sec.

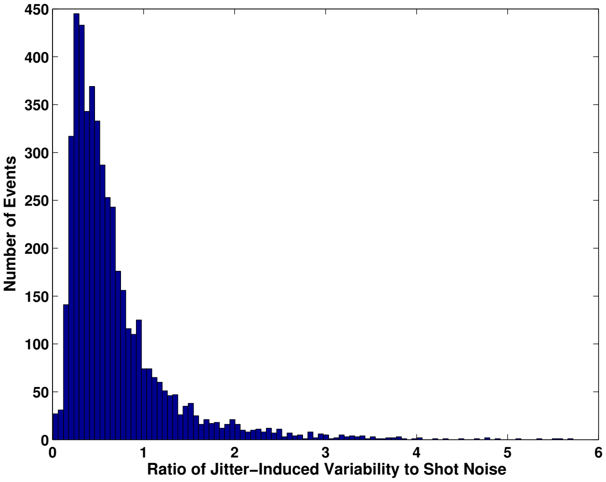

Figure 13.3: A histogram of the ratio of the variability of target star pixel time series due to pointing offsets to the expected shot noise on 3 sec intervals.

The apparent variability of target star pixel time series can vary by as much as 6 over that expected from shot noise, as seen in Figure 13.4. The challenge, then, is to provide a detection threshold that is high enough to prevent the detector from being overwhelmed by false positives for those pixels most sensitive to image motion, while maintaining a relatively good detection rate for those pixels that are relatively insensitive to motion. The appropriate criterion to use in setting the threshold is to pick that value that minimizes the root mean square error between a pixel time series without cosmic rays, and the same pixel time series after cosmic rays have been added and then 'cleaned.' This corresponds to choosing a threshold to minimize the total number of expected errors if either kind (false alarms or missed detections).

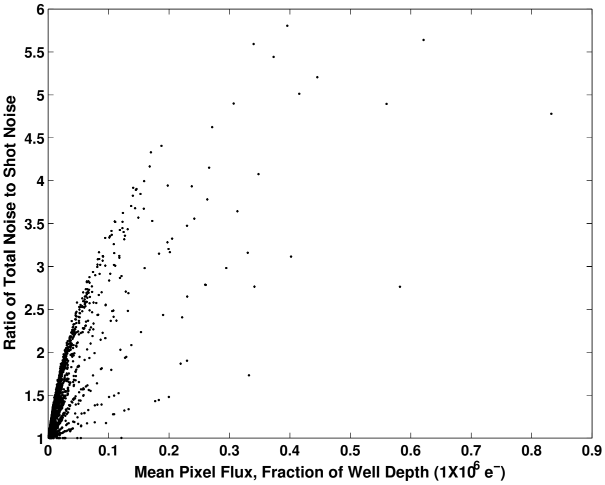

Figure 13.4: The ratio of the total noise of 3 sec pixel time series to the expected shot noise.

For this set of calculations, we used the baseline algorithm: subtract off the previous 15-min pixel value scaled to 3-sec from the current 3-sec measurement, threshold it, and if it exceeds the threshold, replace it with the scaled 15-min average. Figure 13.5 shows the optimal threshold determined for 1100 pixels, along with a threshold that was chosen to track the maximum of the 'envelope' of the optimal threshold. We note that in terms of the sample standard deviation of each pixel time series, the optimal thresholds were tightly distributed about 4.3 σ with a standard deviation of 0.28σ. The deviations of the pixel times series are a combination of shot noise and pointing offset-induced variations. Figure 13.6 shows the ratio of the optimal and the max-envelope thresholds relative to the shot noise for these pixels. There is a clear relationship between the max-envelope threshold and mean pixel brightness, since the latter is the square of the shot noise. The RMS errors between “cleaned” cosmic ray-corrupted pixel time series and the original time series is given in figure 13.7, showing that thresholds can be chosen based on mean pixel brightness that result in reasonable detector performance. Figure 13.8 shows the ratio of the RMS error to the shot noise for both optimal and max-envelope thresholds, showing that the baseline cosmic ray detection algorithm can limit the effects of cosmic rays to below one tenth that of the shot noise across the dynamic range of the CCDs. The effect is even more pronounced at pixel fluxes less than about 2% well depth, where the effect can be limited to as little as 0.06 that of the shot noise. The root sum square (RSS) combination of cosmic rays and shot noise is less than 1% greater than shot noise alone across the dynamic range, and is as little as 0.4% greater at less than 2% well depth (where we’ve assumed that the spatial correlations will “double” the square root statistics of individual pixels).

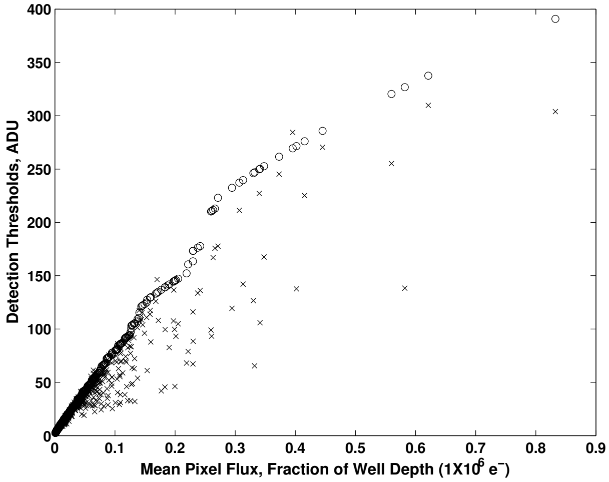

Figure 13.5: The optimal (crosses) threshold for each pixel time series that minimized the chosen error function along with a threshold (circles) that tracks the upper bound of the “envelope” of the optimal thresholds.

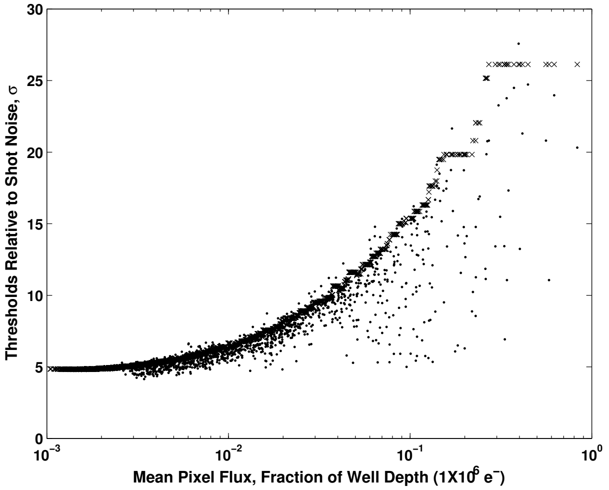

Figure 13.6: Ratios of the optimal (dots) threshold to the shot noise for each pixel time series that minimized the chosen error function along with the ratio of a threshold (crosses) to the shot noise where the threshold tracks the upper bound of the “envelope” of the optimal thresholds.

13.2 Discussion

The results of the numerical simulations described in the preceding sections demonstrates that we should be able to effectively detect and reject cosmic ray events using a very simple detection algorithm. In order to implement the algorithm, however, it is essential that the variability of target star pixels be determined at the timescale of the individual exposure sample intervals. This might be accomplished through modeling efforts, with a knowledge of the ADCS performance and detailed characterization of the optics and the starfield. It could be accomplished much more easily however with direct measurements of pixel time series at the single exposure level. Such measurements would not only allow us to estimate the sensitivity of pixels to motion, but would allow us to better determine the distribution of cosmic ray-induced charge events. I propose that we explore the possibility of specifying a subsection of the full 84 channels for acquisition during FFI mode. Thus, we could for example request a series of 84 FFI’s of which only data for one channel is stored on the SSR, so that the data set would take up only as much

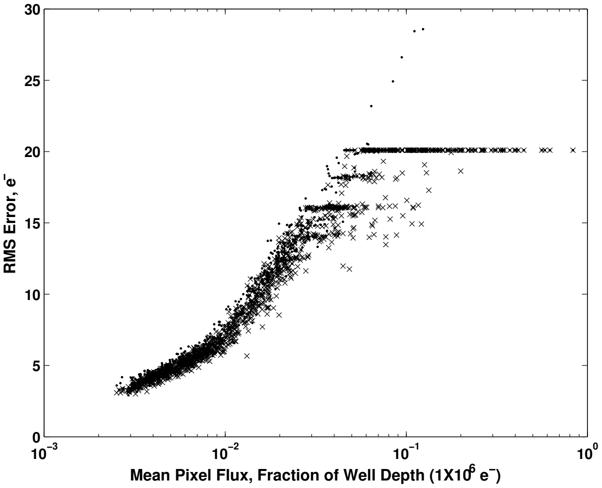

Figure 13.7: The RMS error between “cleaned” pixel time series and pristine pixel time series (i. e., without cosmic rays added) for both the optimal detection thresholds (crosses) and for the maximum envelope-tracking threshold (dots).

memory as a full FFI for the entire set of >100 million pixels. Of course, we would only want to collect such data during the commissioning phase, and perhaps during roll maneuvers, but it doesn't appear to require new capabilities on the part of the hardware, and only slightly more flexibility in terms of the software. It could provide other un-anticipated diagnostic functions, such as the ability to acquire high-rate images from a flaky CCD channel.

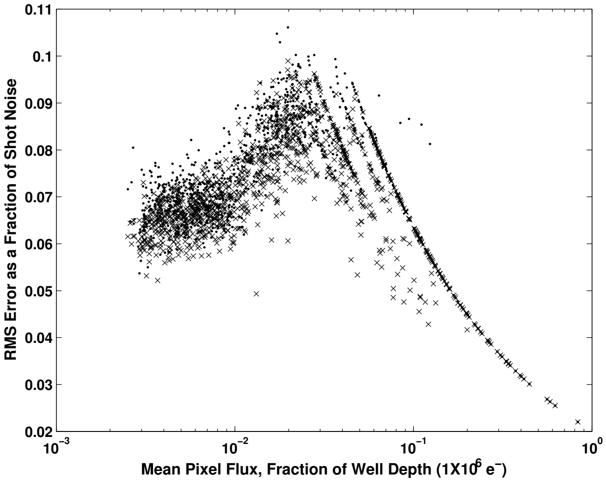

Figure 13.8: The ratio of the RMS error between “cleaned” pixel time series and pristine pixel time series (i. e., without cosmic rays added) for both the optimal detection thresholds (crosses) and for the maximum envelope-tracking threshold (dots) to the shot noise for each target star pixel.

Chapter 14

A Description of the End-To-End Model

This chapter describes the algorithms behind the End-To-End Model (ETEM) and the methodology behind them. Emphasis is given to discussing the limitations of the model and the properties of the various operating modes.

14.1 Analytical Tools

As part of the development effort, three analytical tools have been constructed to aid in the design process. The first tool is the Combined Differential Photometric Precision (CDPP) spreadsheet, which tracks the CDPP for a 6.5-hour transit for a G2V star as a function of the apparent stellar magnitude and the set of mission design parameters. The quantity CDPP is the effective white noise standard deviation in a 6.5-hour interval that determines the S/N of a 6.5-hour transit of a given depth. For example, a CDPP of 20 ppm for a star with a planet exhibiting 84 ppm transits lasting 6.5 hours leads to a single transit S/N of 4.1σ.

The second tool is the *Kepler* Merit Function, which assesses the value of the science return in terms of the number of expected planetary detections for an assumed planetary population distribution (including both habitable and non-habitable planets) and in terms of the number of stars for which pressure mode (p-mode) oscillations can be studied.

The third tool is the End-To-End Model, which is a Monte Carlo simulation of the *Kepler* Photometer producing synthetic data time series at the pixel level. These three tools are supported by a Noise and Error List that tracks the effects and magnitudes of $\sim$75 separate stochastic and systematic noise sources that potentially affect *Kepler* photometric performance.

The first two tools do not produce time series but rather perform analyses combining the expected noise and signal properties or predicting the science return given these properties. As such, ETEM is extremely valuable in assessing the effects of noise terms which are not subject to closed form analysis, such as the effects of pointing 'jitter' or the effects of residual cosmic ray events on the CDPP. This suite of complementary analytical tools allows us to predict the performance of *Kepler* and to measure the effect of design choices on the scientific return.

The goal of this chapter is to describe the methodology behind the current version of ETEM and to give examples of the investigations it has enabled. ETEM has been under development since 1995 as a collaboration between *Kepler* team members at both NASA Ames Research Center and at Ball Aerospace & Technologies Corporation (BATC). ETEM began as a FORTRAN program that generated synthetic photometric data for a single star at the pixel level given a Point Spread Function (PSF), characterization of observational noise sources such as sky background, read noise, and dark current, and a sample pointing offset time series. This simple model of an isolated star was later incorporated into a larger program to simulate 100 target stars together with the effects of shutterless operation as part of the *Kepler* Technology Demonstration (KTD) funded by the NASA Discovery Program (64; 95). Generating synthetic photometric data is a necessary component of assessing the expected photometric performance. The back end of ETEM consists of analyzing the output data to construct light curves and measure the photometric precision achieved for the input design. This analysis uses algorithms similar to those baselined for the actual data processing and analysis system. These differences exist because the output of ETEM has been until recently restricted to producing rather short data sets of up to a few days for relatively few stars compared to the actual mission data. Over the past two years we have redesigned ETEM in such a way as to permit the modeling of millions of background stars together with up to 2000 target stars on a single CCD readout area for artificial data runs of up to three months. In addition to its significantly improved efficiency, the current ETEM incorporates a great deal more realism accumulated as the spacecraft and photometer design have matured.

Figure 14.1 depicts the logical flow of ETEM beginning with the steps required to set up the model for the given system design parameters and ending with the generation of synthetic photometer data. The top line of the logic flow contains the sequence of steps necessary to prepare ETEM for generating synthetic photometer data. These steps incorporate all the phenomena relevant to the quantification of the photoelectronic image read out from a CCD prior to digitization by the Analog to Digital Converter (ADC), and the addition of stochastic noise. The second row of the flow consists of exercising the model to produce synthetic images, add various random noise sources such as shot noise and read noise, digitize the result, and finally, write the results to disk. This sequence of steps is repeated until the requested data set is completed.

The major improvements to ETEM reported in this paper lie in the numerical approach taken to model the response of the CCD pixels to motion of the stellar images. The perturbations to the image position due to tracking errors of the Attitude Determination and Control System (ADCS) and to astrophysical effects are either small enough to be well modeled by polynomial expansions of the pixel values in terms of the $(\Delta x, \Delta y)$ offsets, or occur on timescales much longer than that of a transit duration, and hence, are unimportant with respect to determining S/N of a transit. The principal purpose for ETEM is to analyze noise sources impacting the detectability of transits, although future enhancements envisioned include modeling of long term astrophysical effects such as differential velocity aberration to better understand the complications these might pose with respect to the data processing and target management.

In order to achieve this efficiency, some simplifying assumptions were made. The stars are assumed to move together, that is, no provision is made for parallax, proper motion, or second order effects of differential velocity aberration. As argued above, these effects do not significantly affect the S/N of a transit, but we do need to understand how they might effect operations and analysis of the data. Intrinsic stellar variability is also not currently modeled for the target or background stars. Stellar variability for old, main-sequence stars like the Sun occurs on timescales much longer than the duration of a transit of a terrestrial planet. The results from ETEM and the CDPP spreadsheet have been incorporated into detailed studies of the effects of solar-like variability on the detectability of transiting Earth-like planets (60). Provisions have been made in ETEM to allow for the inclusion of stellar variability into the simulations. Doing so would necessarily reduce the efficiency of ETEM. It is unclear whether this is necessary. To date, ETEM is used mainly to determine the contribution of noise sources not amenable to analysis. As such, once a noise term is characterized by ETEM, its effects are then incorporated into the CDPP spreadsheet and into the Merit Function. Although we plan to evolve ETEM to incorporate more realism over time, it is likely that *Kepler* will continue to require and exercise several numerical models during development.

This paper is organized as follows. Section 14.2 describes the steps taken to develop polynomial representations for a CCD readout area. Section 14.3 details the generation of synthetic CCD data including the addition of stochastic noise to the frames.

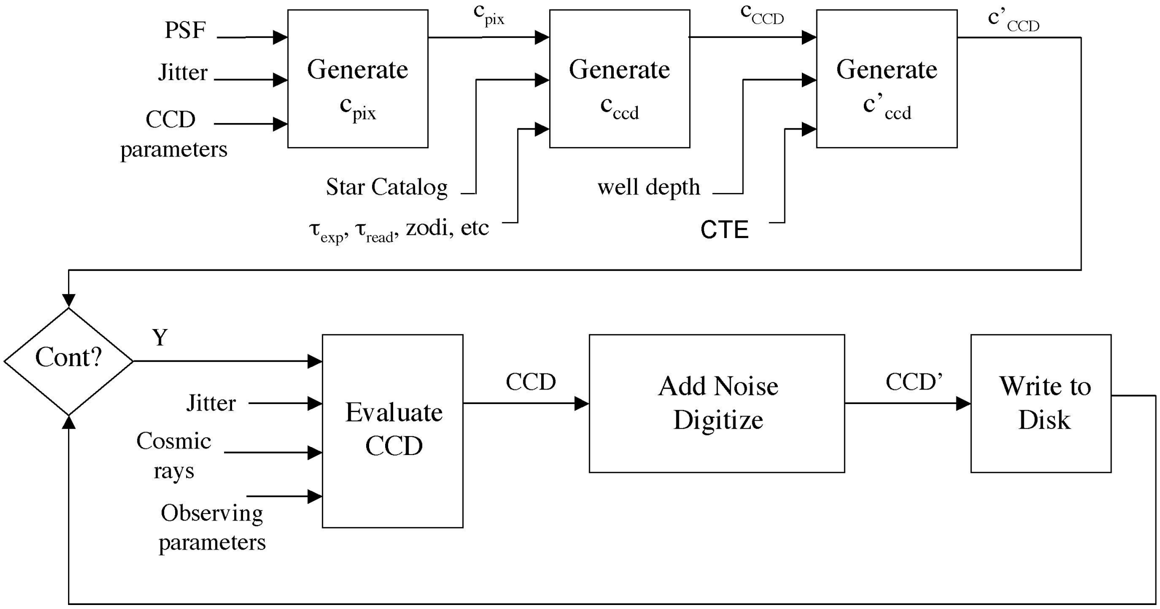

Figure 14.1: Logical flow chart for ETEM. The top flow consists of the sequence of steps necessary to set up ETEM to generate synthetic images, incorporating the relevant design parameters such as PSF, CCD dimensions, pixel-to-pixel sensitivity, intrapixel sensitivity, integration time, readout time, etc. The bottom flow consists of generating synthetic noise-free images, adding stochastic noise, digitizing the result and writing the data to disk. This is repeated until the requested data set is completed.

14.2 A Polynomial Representation for the *Kepler* Photometer

This section details the phenomena incorporated into the two boxes labeled 'Generate $\mathbf{c}_{CCD}$' and 'Generate $\mathbf{c}'_{CCD}$' in Fig. 14.1. With the exception of the effect of spilling of saturated charge, all the phenomena modeled in ETEM for generating synthetic CCD images are linear, so that they can be directly incorporated into a polynomial representation for the response of a CCD to image motion. Nevertheless, most pixels' behavior is well modeled by a polynomial representation, and those few pixels that are not, can be handled separately.

14.2.1 Response of Pixels to Image Motion

As in previous versions of ETEM, the first step is to determine how the pixels under a stellar image respond to image motion. For a star of brightness I_0 located at (x_0, y_0), the charge that is developed on the CCD is

$$I_\lambda = [I_0 \, PSF_\lambda(\Delta x_0, \Delta y_0) \, S_\lambda(x,y)] * D_\lambda(x,y), \quad (14.1)$$

where λ is the wavelength, $PSF_\lambda(x,y)$ is the PSF, $\Delta x_0 = x - x_0$ and $\Delta y_0 = y - y_0$, $S_\lambda(x,y)$ is the sensitivity function of the CCD, $D_\lambda(x,y)$ is the diffusion kernel, and '$*$' denotes the convolution operator. The optical PSF is derived from the optical design of the photometer using a raytracing algorithm (ASAP) for each of 21 wavelengths across the *Kepler* bandpass (420–860 nm)(72).

For $S_\lambda(x,y)$, we take the results reported by Jorden (65), which are only reported at two wavelengths, 600 and 850 nm. Below 600 nm, the variation of S_λ is quite small, while at 850 nm, the peak-peak variation is $\sim$10%. Fortunately, models for $D_\lambda(x,y)$, developed at BATC for the *Kepler* flight CCDs indicate that diffusion is important only for wavelengths shortward of 600 nm, and is not apparent at longer wavelengths (90). This is due to the fact that the longer wavelength light travels through the entire CCD thickness and is absorbed in or very near the active region, so that there is little opportunity for diffusion from the absorption site before readout. At these wavelengths, S_λ exhibits variations that are consistent with the physical gate structure of the CCD. Conversely, this also explains the relative unimportance of intrapixel sensitivity variations for short-wavelength light, which is absorbed above the active region and must diffuse down into it prior to readout. The 'blue' light never sees the gate structure, which has the opportunity to scatter the 'red' light.

Note that the apparent complementarity of D_λ and S_λ implies that I_λ can be expressed as a cascade of convolutions involving PSF_λ, S_λ restricted to a single pixel, and either D_λ or S_λ. Once I_λ is determined at all 21 wavelengths, it can be weighted by the stellar spectrum and photometer bandpass response and summed over λ to determine the total charge intensity for each pixel as a function of position. In the current version of ETEM, we have modeled the process using a total optical PSF over the solar spectrum and photometer bandpass, and an effective charge diffusion kernel prior to consideration of the CCD pixel sensitivity. We have performed analyses to show that this approach is conservative, but are working to improve the fidelity of this step as per the discussion above.

In previous versions of ETEM, the importance of charge diffusion was not recognized and hence, was ignored. This is actually a conservative assumption in that the charge diffusion blurs the optical PSF and reduces somewhat the sensitivity of the pixel values to motion. The values of each pixel as a function of (Δx, Δy) offsets from a nominal position were determined by scaling a tabular representation of the optical PSF (on a 5 by 5 pixel region) with 13 by 13 subpixel resolution, scaling it by the intrapixel sensitivity and then integrating over each pixel region. This response was evaluated at a particular image offset for a given jitter time series by bilinear interpolation over the tabulated values. The process of interpolation is numerically quite intensive especially when using cubic or spline interpolation. We note that this process yielded small but not insignificant modeling errors as the bilinear interpolation actually used did not preserve flux for a perfectly uniform CCD response. The most significant improvement in computational efficiency for ETEM lies in recognizing that for *Kepler*, the expected perturbations to the CCD images due to pointing 'jitter', thermal drifts and astrophysical effects such as differential velocity aberration are quite small over timescales of seconds to several days. For example, the pointing offset 'jitter ball' is required to be no larger than 0.1 arcsec (or 2.5 mpix), 3 σ, and is expected to be much smaller in practice. For such small pointing offsets, the response of the pixels to image motion is smooth and well represented by low-order, two-dimensional polynomials.

The current version of ETEM takes advantage of this fact and incorporates a polynomial fit to the response of each pixel to motion of a stellar image over a fine grid containing the 'jitter ball', resulting in pixel polynomial coefficients, c_{pix}. For any given pointing offset within the design region, each pixel value can then be determined by evaluating the corresponding polynomial for a given pointing offset pair (Δx, Δy) and simply scaling the result to an intensity appropriate for a given magnitude star. Figure 14.2 shows the fitting error between the polynomial representation and the cubic-spline interpolation of the pixel response for the required jitter Power Spectral Density (PSD). For the required Attitude Determination and Control System (ADCS) performance, we find that 3rd order polynomials adequately represent the pixel response to motion. The polynomial is of the following form

$$\begin{aligned} p(\Delta x, \Delta y) \;=\; & c_{00} + c_{10}\,\Delta x + c_{01}\,\Delta y + \\ & c_{20}\,\Delta x^2 + c_{11}\,\Delta x\,\Delta y + \\ & c_{10}\,\Delta y^2 + c_{30}\,\Delta x^3 + \\ & c_{21}\,\Delta x^2\,\Delta y + c_{12}\,\Delta x\,\Delta y^2 + c_{03}\,\Delta y^3. \end{aligned} \tag{14.2}$$

Higher order polynomials can be applied to provide better fits or to allow for a larger design range of jitter. Once the polynomials are determined, an entire CCD frame can be populated with stars using a realistic stellar distribution.

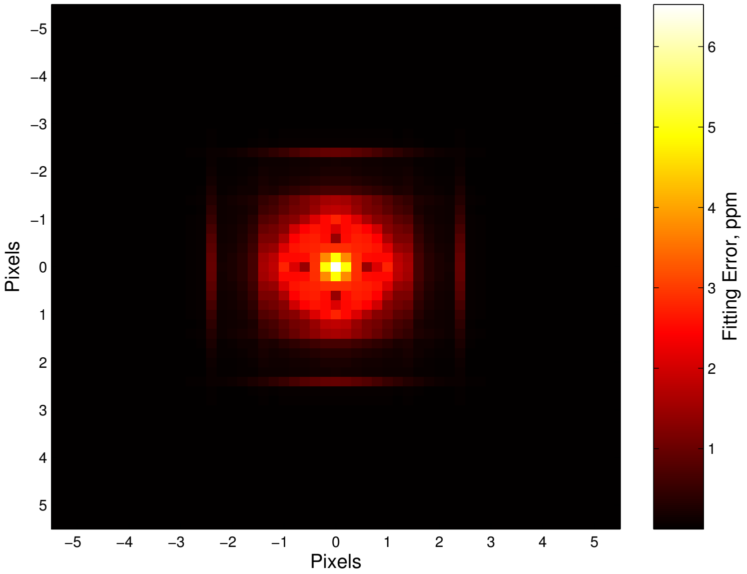

Figure 14.2: The rms fitting error between a cubic spline interpolated representation of response of a pixel to image motion of a stellar PSF, and a 3rd order polynomial representation. The greatest errors occur near the PSF core but at the 10^{-5} level are not significant.

14.2.2 Stellar Population of a CCD Frame

A synthetic star catalog is used to populate a single CCD readout channel consisting of 1100 columns by 1024 rows.[1] The polynomials for each pixel in a CCD can be determined by simply adding together the pixel polynomials for all the stars whose images fall on a given pixel.

Following Batalha, et al. (8), we make use of galactic models made publicly available by the Observatoire de Besançon[2] (see, e. g., Robin & Crézé (98), Haywood, Robin, & Crézé (52), and Haywood, Robin, & Crézé (51)) to obtain expected star counts as a function of apparent magnitude, spectral type and age. The USNO-A2.0 database yields 223,000 stars to m_R=14.0 in the $\tilde{1}12$ square degrees of *Kepler's* FOV (71). This establishes an appropriate mean extinction of ∼1.0 mag kpc^{-1} for the Besançon model. We note, however, that the bandpass for *Kepler* extends from ∼0.45 to ∼0.85 μm, which is far wider than the bandpasses available for the Besançon models. For the purpose of counting stars, using the R band should reflect the number of stars of greatest interest, but may tend to undercount the number of late main sequence stars. Figure 14.3 shows the distribution of stars of all luminosity classes and spectral types predicted by the Besançon model for *Kepler's* FOV.

To construct the CCD polynomial c_{ccd}, we first generate c_{pix} for 25 different nominal centerings of stars within their central pixel, on a 5 by 5 subpixel grid. A synthetic star catalog is compiled by sampling the distribution provided by the Besançon model, drawing random coordinates for each star in the CCD's FOV, and partitioning the stars into 25 polynomial classes. For each polynomial class, each CCD coefficient frame is determined by adding the stellar intensity to the center pixel on a blank 1100 by 1024 pixel array, and then convolving this "impulse frame" with each 11 by 11 coefficient array for each of the 10 polynomial coefficient planes (assuming 3rd order polynomials). In this way, efficient

[1] The actual flight CCDs have 2200 columns and 1044 rows with dual readout amplifiers. The bottom 20 rows are masked to allow for estimation of and correction for the effects of shutterless readout.

[2] http://www.obs.-besancon.fr/modele/modele.ang.html

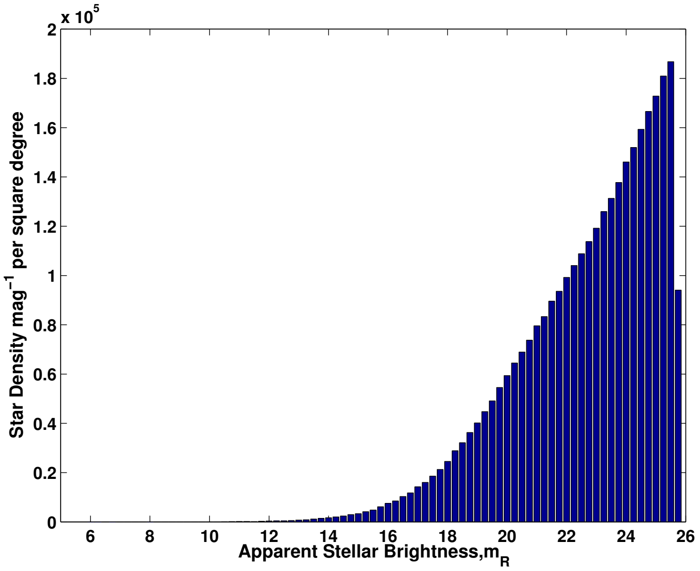

Figure 14.3: A histogram showing the distribution of the density of stars with apparent R magnitude. All luminosity classes and spectral types are represented.

Fast Fourier Transform (FFT) methods can be used to assemble each coefficient frame for each of the 25 stellar polynomial classes. The resulting polynomial frames are added together sequentially for each polynomial class as they are computed.

Given a pointing offset matrix A_{jit} and the CCD coefficients, $\mathbf{c}_{CCD}$[3], the charge deposited in a given interval of time is $CCD = A_{jit}\,\mathbf{c}_{CCD}$. In ETEM, $\mathbf{c}_{CCD}$ is scaled so that evaluation of the polynomial yields flux in e^{-} s^{-1}. Figure 14.4 shows a 2-D histogram of a realization of a jitter time series for the expected ADCS performance binned to 2 Hz sampling. We are now in a position to generate synthetic CCD images for a pointing offset time series.

14.2.3 Additional Imaging Phenomena

The polynomial representation $\mathbf{c}_{CCD}$ developed in the previous section allows us to evaluate the charge developed on a CCD for a given attitude, but it does not factor in all the relevant effects. In particular, we need to accommodate additive noise sources such as dark current, zodiacal light and the effect of the shutterless operation. Saturation effects must also be considered, along with Charge Transfer Efficiency (CTE), but these are the subjects of §14.2.4.

There are two purely additive fluxes that do not respond significantly to image motion: dark current and zodiacal light. Dark current accumulates during exposure and readout of the CCDs and is a strong function of the operating temperature of the CCD. Although the operating temperature of *Kepler's* focal plane is so cold ($< -90°$C) that the dark current is expected to be negligible, it is still accommodated in ETEM. The zodiacal light is solar flux that is scattered from dust grains in and above the ecliptic plane into the Photometer's aperture. Characterization of zodiacal light by the Hubble Space Telescope implies that the zodiacal background will inject the equivalent of an m_R=19 star in every CCD pixel (4

[3] Note that the polynomial coefficients are denoted by boldface. This is to indicate that the $\mathbf{c}_{CCD}$ is a matrix whose columns correspond to the polynomial coefficients, and whose rows correspond to each of the pixels under analysis. The evaluation of the CCD polynomial can then be expressed using matrix algebra, although the results must be reshaped to recover the original dimensions of the CCD.

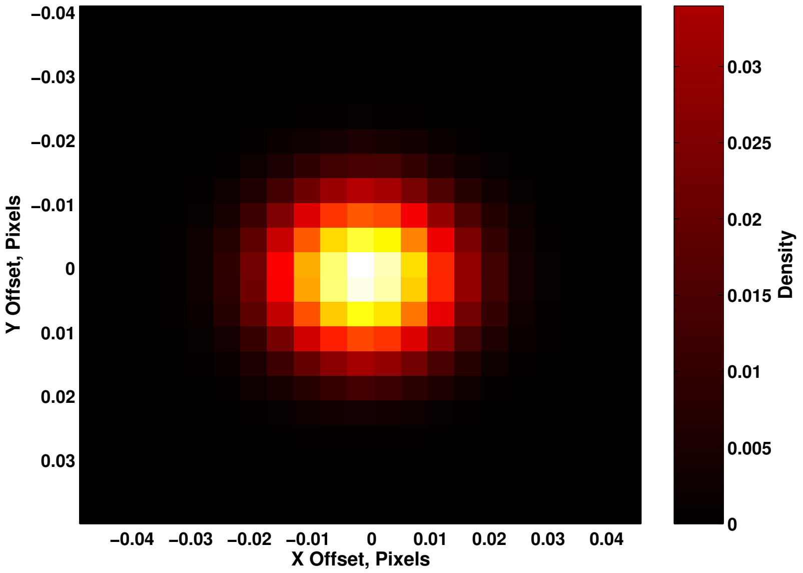

Figure 14.4: A 2-D histogram showing the distribution of pointing offsets for the expected ADCS performance for *Kepler* binned to 2 Hz sampling. The standard deviation of the pointing is 0.01 pix in each axis.

arcsec by 4 arcsec). This is much higher than the expected dark current. Neither dark current nor zodiacal light will vary with the expected pointing errors, although zodiacal light will vary smoothly over large spatial scales and on time scales of months. For the time scales of most interest to ETEM, these flux sources can be simply added to the constant term in $\mathbf{c}_{CCD}$.

At this point ETEM incorporates pixel to pixel sensitivity variations. This is particularly simple as it amounts to scaling each pixel polynomial by the relative sensitivity of the pixel. Most ETEM runs use a highly conservative value of 5% for the interpixel sensitivity variations and draw each pixel's relative sensitivity from a Gaussian distribution.

The fact that *Kepler* lacks a shutter has significant but mostly benign implications for the CCD images. During readout, each row is clocked down the CCD, passing under any stars falling on the CCD below their position during the exposure. At the same time, new rows are being read in from the top of the CCD and clocked down to their nominal locations for the next exposure, passing underneath stars above their exposure positions. The resulting images contain vertical streaks due to star light accumulating in the pixels along each column during readout. The smear component can be calculated by summing each frame of coefficients along the columns, scaling for the exposure time spent in each row. The smear polynomial only responds to image motion along the rows, except at the very edges of the CCD. Accounting for smear in $\mathbf{c}_{CCD}$ amounts to replicating the row polynomial for smear and adding it to each of the rows in $\mathbf{c}_{CCD}$. There is a provision for overclocking the CCDs by 20 rows for testing purposes, but also to allow for a separate estimate of smear. Such overclocked rows do not exist during the exposure, so while they pick up smear as they are clocked through the field, they do accumulate some dark current during readout.

Another source of flux exists: scattered light in the photometer. Studies have been performed to estimate the fraction of the focal plane that will be adversely affected by ghost images from the handful of $m_R \leq 6$ stars in the FOV. At this point, however, the design is not mature enough to quantify and model the effect in ETEM. The methodology used to model the star field applies to the ghosts and will be used to model the effect once sufficient data is collected during testing.

14.2.4 Saturation and Charge Transfer Efficiency

All the previous phenomena represented linear transformations of the polynomials representing the response of the CCDs to image motion. At this point it is necessary to include the nonlinear effect of charge saturation and charge transfer efficiency (CTE) in the simulation. We model saturation of a pixel as a process that conserves charge, but distributes it along the column containing the saturated pixel evenly in both directions. The former effect is supported by experiments performed with the Kepler Tech Demo and with HST (43). For the present purposes, it is not important to have a model for saturation that is realistic in all details. It is sufficient to have a model that is indicative of the difficulties pixel saturation may pose. Saturation will only affect a small handful of target stars in any event.

In ETEM, after the effects described in §14.2 are accounted for, a set of images is generated over a grid of offsets, much as for the calculation of the original pixel polynomials. For these images, pixels that exceed the specified CCD well depth are iteratively spilled up and down their columns until no pixels are saturated. The imperfect CTE is modeled at this point by noting that it can be expressed as a linear infinite impulse response (IIR) digital filter. Let b'_n be the pixel value read out from the CDD including the effects of CTE, and let $\{b_n, b_{n+1}, b_{n+2}, \dots\}$ be the pixel values in sequence of readout starting with pixel n before including the effects of CTE. We can express b'_n in terms of the $\{b_n, b_{n+1}, b_{n+2}, \dots\}$ as

$$b'_n = \alpha b_n + (1-\alpha)\, b_{n+1} + (1-\alpha)^2\, b_{n+1} + \dots, \quad (14.3)$$

where α is the fraction of charge in a pixel that is successfully clocked to the next row for a single clock cycle. Although the effective CTE filter is IIR, a typical value for α is 0.9996, so that $(1-\alpha)^m$ becomes insignificant for $m > 8$. The CTE filter is convolved with each column of the images for the parallel readout and with each row for the serial readout.

New CCD pixel polynomials are fitted to the set of images and the fitting residuals are examined for poorly behaved residuals. Saturated pixels and neighboring pixels that accept spilled charge are typically flagged, and the spill of saturated charge and CTE are modeled directly for these pixels and their neighbors. All other pixels' behavior is well represented by the new polynomials, $\mathbf{c}'_{CCD}$, since all the transformations, including the effect of CTE, are linear transformations of the original polynomials. At this point, ETEM is ready to generate synthetic photometric data for a specified run.

14.3 Running ETEM

This section describes the steps performed to generate synthetic CCD data once the development of the polynomial representation for the CCD response to motion is complete. To generate synthetic CCD data, ETEM evaluates the polynomial $\mathbf{c}'_{CCD}$, simulating spill of saturated charge and CTE for flagged pixels. Shot noise and read noise are added to the pixels, along with charge from cosmic ray events, if desired. The results are digitized, and are then written to disk and the process is repeated until the run is complete. There are two modes of operation for ETEM with respect to the generation of synthetic data, and these relate to operational constraints for *Kepler*.

To prevent saturation of target stars, the exposure time for the photometer is ∼3 seconds, so that each day approximately 29,000 images are acquired. There is not enough memory on the Solid State Recorder (SSR) onboard *Kepler* to keep all this data, so two lossy compression techniques are used to reduce the size of the data set. The first technique is to co-add the images for 15 minutes, reducing the total number of images stored on the SSR per day to 96. For the second technique only the pixels of interest are stored: those containing target stars and collateral pixels used to correct for CCD artifacts and other systematic errors, such as sky background, dark current and smear from shutterless operation. So, too, for ETEM there is no reason to generate data for pixels that won't be analyzed later. An analysis module examines the pixel content for each target star and determines the optimal photometric aperture in a similar manner to that described in Jenkins et al. (64).

The two modes relate to the generation of 15-minute frames, or long cadences, onboard *Kepler*. In the first mode, individual readouts are generated explicitly by evaluating $\mathbf{c}'_{CCD}$, adding stochastic noise and digitizing the results. These are co-added until the appropriate number have been summed to form a long cadence, then the results are written to disk. This mode is useful in examining phenomena that operate on timescales shorter than 15 minutes, such as analyzing the ability of *Kepler* to identify and deal with cosmic rays. The other mode of operation is to evaluate $\mathbf{c}'_{CCD}$ for an entire 15-minute interval, called the long cadence mode, add all the stochastic noise corresponding to that interval, and to model the effects of quantization by adding additional random deviates which are drawn from an appropriate distribution. For long runs of ETEM, the long cadence mode is preferred as it is ∼300 times less computationally intensive than the first.

The long cadence mode is enabled by the polynomial representation itself and the fact that the noise on a 15-minute frame can be analytically related to that at the single exposure level. Consider the process of co-adding a sequence of noise-free CCD images generated by evaluating $\mathbf{c}'_{CCD}$. Let $\mathbf{b}_n$ be a sequence of noise-free CCD frames constructed by evaluating polynomial $\mathbf{c}'_{CCD}$. For example, suppose we wish to bin the results by a factor of three, yielding $\tilde{\mathbf{b}}_\mathbf{n}$. This process can be written as

$$\tilde{\mathbf{b}} = BA_{jit}\,\mathbf{c}'_{CCD}, \tag{14.4}$$

where B implements the binning operation and is given by

$$B = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & \dots & 0 & 0 & 0 \\ & & & \vdots & & & \ddots & & \vdots & \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & 1 & 1 & 1 \end{bmatrix}. \tag{14.5}$$

Note that the associative property of matrix multiplication implies that A_{jit} can be pre-multiplied by the binning matrix B before it is multiplied by $\mathbf{c}'_{CCD}$. A significant reduction in processing time for the 15-minute mode is achieved relative to the single exposure mode by forming $\tilde{A}_{jit} = BA_{jit}$, and hence, binning the jitter matrix prior to evaluating $\mathbf{c}'_{CCD}$.

Whether or not ETEM is operating in the single exposure mode, once $\mathbf{c}'_{CCD}$ is evaluated, it is time to add shot noise and read noise. This is accomplished by adding Gaussian noise of appropriate variance to the noise-free polynomial values. In the long cadence mode, the specified single-exposure read noise is scaled by the square root of the number of exposures in a 15-minute integration. At this point, synthetic cosmic rays can be added to the images, if desired.

14.3.1 Cosmic Ray Events

The cosmic ray flux environment has been of great concern to almost all space missions with CCDs since they are sensitive to cosmic rays. The actual flux experienced by a device depends a great deal on the exact orbit, that is, is the spacecraft in low Earth orbit (LEO), or is it in deep space? The flux also depends on the shielding and configuration of the detectors within the spacecraft, which can affect the generation of secondaries from primary events. In any case, the *Kepler* Mission has adopted a flux rate of 5 cm^{-2} s^{-1} based on previously flown missions in similar orbits, such as *SOHO*.

A study was conducted by BATC for *Kepler* to derive the distribution of total charge deposited into a CCD for each cosmic ray hit(83). The results are displayed in Fig. 14.5, which shows a mode of ∼2500 e$^-$, little or no charge below ∼2000 e$^-$, and a long upper tail trailing out to at least 100,000 e$^-$. We note that 90% of events deposit less than 6,200 e$^-$ into a CCD. To put this into perspective, note that an m_R=12 star occupies about 25 pixels, and that over 6.5 hours, about 4×10^9 e$^-$ accumulates in its aperture. The shot noise for such a star will be 63,245 e$^-$. Now, 25 pixels receive a cosmic ray flux rate of 21.3 per 6.5 hr interval. To compare this to the effect of uncorrected cosmic rays, we need to model the distribution of charge from cosmic rays at the pixel level.

To transform the distribution in Figure 14.5 to the

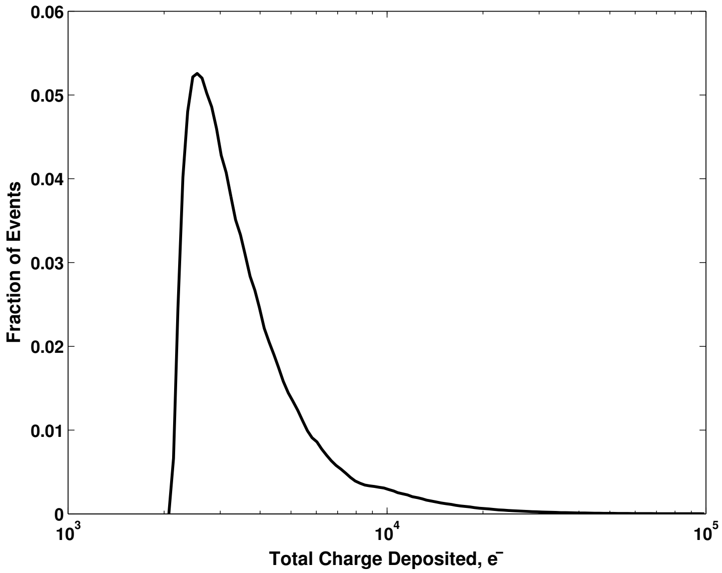

Figure 14.5: The distribution of the total charge deposited into a *Kepler* CCD per cosmic ray event. This distribution resulted from modeling taking into account the expected cosmic ray environment for *Kepler's* orbit, and a detailed structural model for the spacecraft, the instrument, and any planned radiation shielding(83).

desired one, we applied the following assumptions: 1) The total charge deposited is uniformly distributed over the path traveled by the cosmic ray as it traverses the CCD slab. 2) The charge deposited by the cosmic ray diffuses the same way as does charge from actual photons. Given the geometry of the CCDs (27 μm × 27 μm × 16 μm), we traced random rays through a 13 by 13 pixel region of a CCD, distributing the charge in each pixel encountered by a ray according to the assumptions above. The CCD pixels were divided into 13×13 sub-pixels for the purposes of the numerical calculations. We amassed a catalog of 6,097 cosmic ray trails, normalized so that the sum of each trail was unity. Each trail, then, can be scaled by a random deviate drawn from the total charge distribution to model the effect of a single cosmic ray. A Monte Carlo experiment using this model showed that the rms noise injected by cosmic rays in a 25-pixel aperture in a 6.5-hr interval is 21,171 e^-, or about 5 ppm relative to the stellar flux. This is not significant compared to the shot noise.

14.3.2 Digitization of the Synthetic, Noisy CCD Frames

After the stochastic noise has been added to the synthetic CCD frame, it can be digitized and either coadded to the running sum, or written to disk, in the long cadence mode. For this latter mode, the effect of quantization at the single exposure level can be modeled by adding zero-mean, White Gaussian Noise (WGN) with a standard deviation equal to $\sqrt{M}G/\sqrt{12}$ where G is the gain in $e^- ADU^{-1}$, and M is the number of co-adds. This does not accurately model extremely dim pixels whose exposure-to-exposure variations are less than 1 ADU, but these do not occur in target star pixels. In this mode, the final step is to normalize the pixel values by the gain to convert the scale to ADU from e^-. In the single exposure mode, the digitization can be performed explicitly. Note that for the single exposure mode, there is the opportunity to include the effects of nonlinearities in the analog signal processing chain before the quantization.

Figure 14.6 displays the average 15-minute frame for one run of ETEM, while Fig. 14.7 displays a single, 2.88 s exposure where only the pixels of interest have been calculated. The effects of the shutterless readout are evident as vertical streaks. In ETEM, the long cadence pixels of interest are written to disk and then subjected to analysis to determine the CDPP. By comparing the results of separate runs with individual noise sources toggled on and then off, it is possible to assess their contribution to the total CDPP budget.

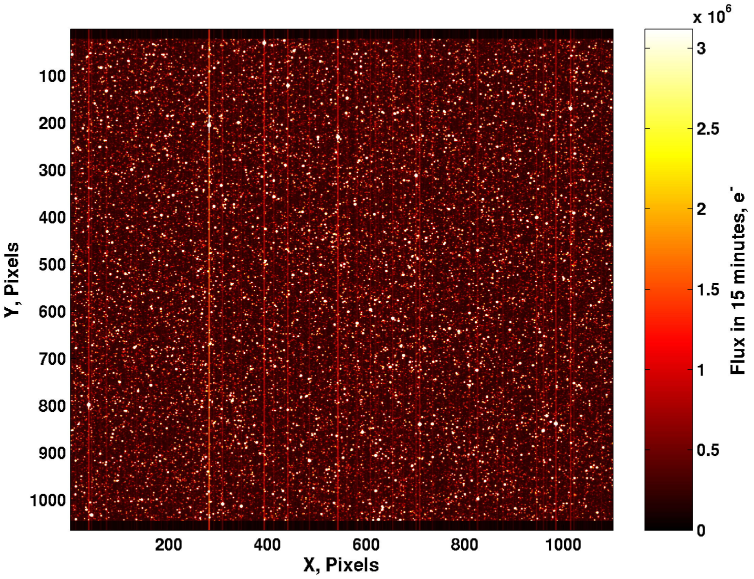

Figure 14.6: Synthetic accumulated CCD frame for *Kepler*. The image is the mean 15-minute frame for a synthetic stellar population generated by ETEM, clipped to 1% of the full range. Approximately 1×10^6 stars are simulated.

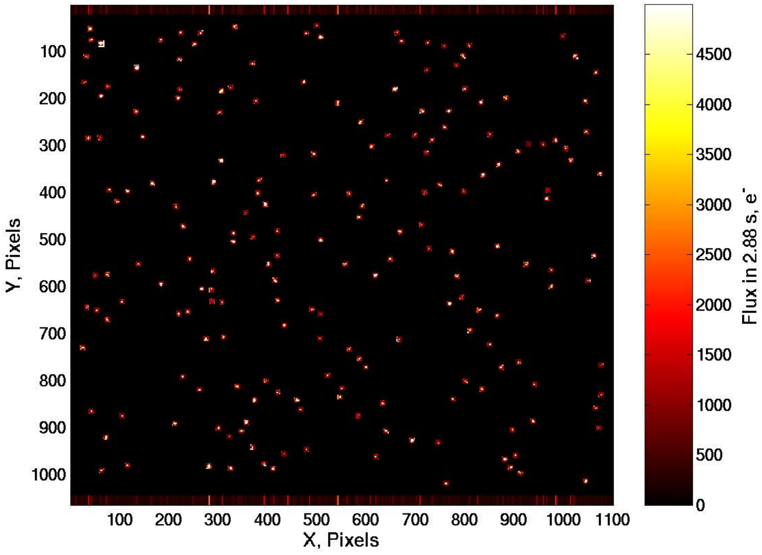

Figure 14.7: An image representing a single 2.88-s exposure generated during one run of ETEM, clipped to 0.5% of the full range. For efficiency, only pixels for those stars selected for study are calculated during the run, along with collateral pixels allowing for estimation and removal of dark current and shutterless smear.

Appendix A

Acronym List

Acronym	Definition
ADC	Analog to Digital Converter
ADCS	Attitude Determination and Control System
ADU	Analog Data Unit
AR	Auto-Regressive
ATBD	Algorithm Theoretical Basis Document
BATC	Ball Aerospace Technologies Corporation
CCD	Charge Coupled Device
FOV	Field Of View
CCDF	Complementary Cumulative Distribution Function
CDPP	Combined Differential Photometric Precision
CEGP	Close-in Extrasolar Giant Planet
CSR	Concept Study Report
CTE	Charge Transfer Efficiency
DEC	Detection and Error Coding
Dec	Declination
DIA	Difference Image Analysis
DMC	Data Management Center
DSMS	Deep Space Mission System
DSN	Deep Space Network
ETEM	End-To-End Model
FFI	Full Field Image
FOP	Follow-up Observations Program
FS	Flight Segment
FWHM	Full Width Half Max
GA	Genetic Algorithm
HAO	High Altitude Observatory
HST	Hubble Space Telescope
JFIF	JPEG File Interchange Format
JPEG	Joint Photographic Experts Group

Acronym	**Definition (continued)**
KMS	Kepler Mission System
KTD	Kepler Technology Demonstration
MATLAB	MATrix LABoratory
MOC	Mission Operations Center
OWT	Overcomplete Wavelet Transform
PoI	Pixel of Interest
PDF	Probability Density Function
PSD	Power Spectral Density
PSF	Point Spread Function
RA	Right Ascension
RMS	Root Mean Square
RSS	Root Sum Square
SCP	Stellar Classification Program
SO	Science Office
SOC	Science Operations Center
SNR	Signal to Noise Ratio
SSR	Solid State Recorder
STScI	Space Telescope Science Institute
SVD	Singular Value Decomposition
QE	Quantum Efficiency
WFPC	Wide Field Planetary Camera (HST)
WGN	White Gaussian Noise

Appendix B

Kepler SOC Algorithm List

0.
Algorithm Title: Example
Function: Summarizes the key features of the algorithm and what the input/output relationship is for the data that is processed by this algorithm.
Heritage: Describes any relevant precursors to the algorithm that were used as a theoretical basis for algorithms envisioned for use on the *Kepler* mission.
Description: Provides a pointer to the literature for a relevant paper describing the algorithm or further describes its functionality.

1.
Algorithm Title: Optimal Pixel Weighting
Function: The algorithm optimizes (in a least-squares sense) the weights for each pixel in the photometric aperture of a star in order to minimize the effect of image motion on the resulting summed aperture flux.
Heritage: Kepler Tech Demo
Description: See Jenkins, J. et al. 2000, 'Processing CCD Images to Detect Transits of Earth-sized Planets: Maximizing Sensitivity while Achieving Reasonable Downlink Requirements,' SPIE Conference 4013, p. 520.

2.
Algorithm Title: Pixel Mask Selection
Function: The algorithm selects a set of pixels around each star on which weighted aperture photometry is done. The pixels are selected by choosing only those that add information to the flux estimate.
Heritage: Kepler Tech Demo
Description: See Jenkins, J. et al. 2000, 'Processing CCD Images to Detect Transits of Earth-sized Planets: Maximizing Sensitivity while Achieving Reasonable Downlink Requirements,' SPIE Conference 4013, p. 520, and references therein.

3.
Algorithm Title: Ensemble Star Selection
Function: The algorithm selects a set of stars to be used for the ensemble average for normalizing a given target star. Dividing a target star's flux by the flux of an appropriately chosen ensemble removes common-mode noise.

Heritage: Vulcan
Description: Details of the ensemble selection are not yet known. Several methods have been used for Vulcan. At it simplest, all stars that were read out through the same CCD amplifier could be used as the ensemble for a given target. Some effort will likely be made to eliminate those that are known to vary, either intrinsically, or as a result of their location on the focal plane.

4.
Algorithm Title: Relative Flux Decorrelation
Function: The algorithm removes flux changes that are correlated over many stars. The ensemble average can remove only common-mode signals (e.g., all stars increase in brightness), whereas the decorrelation algorithm removes and signals that affect multiple stars on the same time scale, e.g., image motion, which may cause some stars to increase in brightness and others to decrease.
Heritage: Kepler Tech Demo, Vulcan
Description: See Jenkins, J. et al. 2000, ‘Processing CCD Images to Detect Transits of Earth-sized Planets: Maximizing Sensitivity while Achieving Reasonable Downlink Requirements,’ SPIE Conference 4013, p. 520.

5.
Algorithm Title: Adaptive Matched Filter
Function: The algorithm first ‘whitens’ the noise in relative-flux light curves. It uses the existing noise structure to detrend light curves and can adapt to non-stationary noise distributions. It then searches the whitened data for a predefined ‘matched’ test signal Ða planet transit. The algorithm returns the single-event statistics for each target star.
Heritage: Numerical simulations
Description: See, e.g., Jenkins, J. 2002, ‘The Impact of Solar-like Variability on the Detectability of Transiting Terrestrial Planets,’ Astrophysical Journal, 575, p 493; Van Trees, H. L. 1968, ‘Detection, Estimation, and Modulation Theory, Pt. 1,’ (New York: Wiley & Sons).

6.
Algorithm Title: Multiple-Event Statistics (Foldvec)
Function: The algorithm folds the single-event statistics output from the adaptive matched filter over a range of periods searching for the maximum multiple-event statistic. The algorithm returns the period, phase, and S/N of the maximum detected signal for each target star.
Heritage: Vulcan
Description: The single-event statistics are folded over the range of periods for which planets are being searched. The step size in period is such that the correlation between the test signal for two tests, at P and P+DP, is a specified amount, e.g., 0.75. The step size is therefore dependent on the period and duration of the transit. The maximum signal at each period is checked to see if it exceeds the current maximum value for the tests up to that point. At the end of the run, the period, phase and S/N of the maximum multiple-event signal is returned.

7.
Algorithm Title: Transit Confidence Estimation
Function: The algorithm estimates the significance of a detected multiple-event transit signal given the noise distribution in the light curve being examined and the effective number of independent tests performed during the transit search. The result of the algorithm is an estimate of how likely it is that a signal as high

as the one observed would occur by sampling the light curve randomly the appropriate number of times.
Heritage: Numerical simulations, Vulcan
Description: See, Jenkins, J., et al. 2002, 'Some Tests to Establish Confidence in Planets Discovered by Transit Photometry,' Astrophysical Journal, 564, p 495.

8.
Algorithm Title: Quick-look Data Quality Check
Function: The algorithm will be used at the Mission Operations Center to do quick-look testing of the data coming down from the spacecraft.
Heritage: Vulcan
Description: The specific checks to be performed have yet to be determined. They will likely include such things as image motion, focus, and large-scale variability for a specified set of test stars.

9.
Algorithm Title: Bias Correction
Function: The algorithm estimates and subtracts the CCD amplifier bias signal. Bias estimation and removal will be done at the DMC.
Heritage: Vulcan, Kepler Tech Demo, standard in CCD photometry
Description: A low-noise bias is estimated by over-scanning the readout amplifier by a number of rows. This estimate may be further filtered. The details of the estimator and filtering will likely be determined base on flight hardware performance.

10.
Algorithm Title: Non-linearity Correction
Function: The algorithm corrects for the non-linear response of the CCDs. Non-linearity correction will be done at the DMC.
Heritage: Kepler Tech Demo, Vulcan, standard in CCD photometry underlineDescription: The non-linear response of the CCDs is corrected after the bias is removed, likely by a simple table look-up. The details of the correction will be determined by measuring the response of the flight CCDs.

11.
Algorithm Title: Shutterless Readout Correction
Function: The algorithm corrects for the smear caused by shutterless readout of the CCDs. Smear correction will be done at the DMC.
Heritage: Kepler Tech Demo
Description: An un-illuminated region at the end of each CCD column is used to estimate the flux that is smeared into each pixel of a given column. See Jenkins, J. et al. 2000, 'Processing CCD Images to Detect Transits of Earth-sized Planets: Maximizing Sensitivity while Achieving Reasonable Downlink Requirements,' SPIE Conference 4013, p. 520.

12.
Algorithm Title: Background Estimation
Function: The algorithm estimates the contribution from background sources (zodiacal light, psf-wings from other stars) to the flux in the aperture of each target star. A robust estimate of the background is needed to accurately assess the characteristics of a given detection.
Heritage: Vulcan, Kepler Tech Demo, standard CCD processing

Description: The background flux for Kepler will primarily come from three(?) sources: zodiacal light, the wings of the point-spread-functions from other stars, and light scattered within the photometer. The background contribution from zodiacal light will vary with the orbital position of the spacecraft and should be largely independent of position within the field-of-view. The background from scattered light and other stars will depend on position on the focal plane and will have to be measured. By looking at the distribution of pixel brightness we can make a robust estimate of the background level. Several methods can be used, from the simple median, to a function fit to the distribution.

13.
Algorithm Title: Star Centroid
Function: The algorithm determines the centroid of the distribution of a star's flux.
Heritage: Kepler Tech Demo, Vulcan, standard stellar photometry
Description: A variety of methods are available from fitting the individual flux distributions, to a full-image model, depending on the positional accuracy needed.

Appendix C

FORTRAN Listing for Folding Single Event Statistics

This section contains a listing of a FORTRAN subroutine used to fold single event statistics that returns the maximum statistic observed for each trial period. The inputs are the single event correlations, the single event energies, the minimum period and the maximum period.

```
subroutine maxfoldvecm(lmax,jmax,a_mat,anorm_mat,size,minfold,maxfold)
integer size, minfold, maxfold
real*8  lmax(*), a_mat(*), anorm_mat(*), jmax(*)
real*8 li,lnormi
integer i,ii,j,k,ni

do i=1,size
  anorm_mat(i)=anorm_mat(i)*anorm_mat(i)
enddo

istart=0
do i=minfold,maxfold
   ii=i-minfold+1
   lmax(ii)=-1.d99
   jmax(ii)=1.d0
   ni=int(size/i)
   do j=1,i
      li=0.d0
      lnormi=0.d0
      k=j
      ni=int(size/i)
      do while (k .le. size)
 li = li + a_mat(k)
 lnormi = lnormi + anorm_mat(k)
 k=k+i
      enddo
      li = li/sqrt(lnormi)
      if(li.gt.lmax(ii)) then
 lmax(ii)=li
 jmax(ii)=float(j)
      endif
   enddo
enddo

do i=1,size
  anorm_mat(i)=sqrt(anorm_mat(i))
enddo

return
end
```

Appendix D

Summary of FORTRAN codes used for DIA.

To provide more explicit guidance in how DIA has been used for analysis of HST time-series photometric images, this appendix provides a rough “User’s Guide" to the several codes I use. All codes are available for reference, although these are “research level" and thus poorly documented, with obsolete sections, and other limitations.

I’ve broken the overall outline into three distinct stages: (1) production of a good mean over-sampled image, (2) difference image creation. In practice these stages are distinct only in the sense that I create an over-sampled image early, then don’t update it. Most of the work needed to create an over-sampled image has to be repeated in getting properly set up to create difference images. (3) Extracting stellar photometry values from the difference images and massaging these.

The flow here is a bit awkward in the sense that I’ve layered on new steps as needed over the years. A rewrite of the codes would almost certainly result in some modifications to the flow. (If I were starting over in a coherent way some whole codes might go away.)

```
(1) Initial processing, and production of first over-sampled image.

make.savef

  -- Startup routine that takes the calibrated, multi-group images
     from the archive (after conversion from fits to .c0h and .c0d
     format) and writes out in 4 separate image streams with 90
     degree rotations, application of delta-dark corrections,
     scaling by gain to yield e- units and storage as i*2.
     (This step is certainly unique to the HST data.)

make.codep

  -- Derives initial estimate of x,y offsets of each frame using PSF
     fitting to a few stars.  Used only once at beginning on full set
     of frames. (Likely to not be needed for Kepler data, or done
     differently.  An initial guess for offsets will be needed.)

make.skysub

  -- Derives a global sky zero point for each frame.  Used only once
     at beginning.
     Actually this is now used again after setting up the difference
     images to define the frame to frame sky changes (rather small
     in general compared to the constant background provided by the
     crowded field).

make.craye

  -- THE primary code for flagging cosmic rays and developing a polynomial
     fit based model of the stellar scene.  Largest and most important routine.
     Requires iteration with other routines.

make.image

  -- Used only for 'visualization' of the craye stacking result, derives
     a X2 oversampled average image of sky via evaluation of analytic
     expansions.
     (Note:  I tend to use "sky" to mean two different things.  Last it
     meant the distribution of light corresponding to the stellar scene.
     Sometimes it will mean a uniform background.)

make.fitpos1dr

  -- Used to refine estimate of x,y offsets, and rotation if desired
     by fitting for optimal shift
     of over-sampled image required for best fit to all individual frames.
     Requires iteration.

make.fitposps

  -- Used to include variation of plate scale in the registration model.

make.imgset4

  -- Used to define the final X4 over-sampled average image by second stacking
     process.  Used only once at end.
     This is the approach discussed in the Gilliland, Nugent, and Phillips
     (1999) paper.
     The 47 Tuc data is so extensive that each pixel can be forced to use
     all of the terms in the bi-cubic Intensity = f(x,y) expansion, even in
     regions of low signal.  Therefore the direct over-sampled image model
     can be used to define the over-sampled image.

Some of the above steps require iteration.  The following might be illustrative
of a full set of runs:

  0) savef       reformat frames, done only once
  1) codep       provides initial registration guess
  2) skysub      provides   "     sky background guess
  3) craye       derive first over-sampled model fit, presumably setting
 threshold for cosmic ray elimination high
  4) fitpos1dr   derive x,y zero point offsets and rotation for each frame
  5) fitposps    derive plate scale variations for each frame
  6) craye       repeat over-sampled model with improved registration input
  7) fitpos1dr   rederive registration
  8) fitposps    rederive registration
  9) craye       form new over-sampled model with updated registration
     [There is no well defined stopping point, can watch how much the
     registration coeff change, and if stable stop or do another cycle.]

 10) imgset4     evaluate over-sampled mean image for DAOPHOT analysis and
 star list definition
```

(2) Setting up the difference images. This section discusses new codes developed over the course of analyzing the 47 Tuc data and updated for the current (2004) analysis of bulge data with HST/ACS.

make.cntcry

-- Evaluates statistics on number of cosmic ray hits attributed to pixels on sky and on stars respectively. Provides a vector that can be used in future craye runs that raises the threshold for cosmic ray elimination for frames having too many rejected points on stars.

make.fitposxy25

-- This solves for a delta to the registration model (each individual frame against the current over-sampled model) by evaluating zero point x,y offsets over a 5x5 grid of areas for each frames. Then these are fit with quadratics and cubics in x,y for later use.

make.fitpsf25

-- This solves for a PSF at each of 5x5 areas over frames such that convolution of this PSF onto the over-sampled model image best represents each individual frame. This is the fundamentally new step developed to take into account focus variations frame-to-frame. The PSF is solved for in X2 over-sampled space on a 7x7 grid via brute force least squares iteration. This step probably requires 2/3rds of the overall processing time (but doesn't require a large memory machine).

make.difcon

-- This produces a "differential PSF image" (I'm not sure what terminology to really use) that is the difference for each frame of the model image convolved with the focus differential PSF and the model image simply evaluated at the position of each individual frame. If there were no frame-to-frame focus changes, then these frames would be zero. This isolates the changes at a given pixel due to focus changes relative to the mean value. These differential images are then used in cycling through the basic codes that build up both registrations and over-sampled models.

make.submodc

-- Using all of the registration information, information on frame-to-frame focus changes, and the over-sampled model the best model representation for each frame is subtracted from the frame. This produces the difference images.

Assuming that all of the steps in block (1) have been completed I have adopted the following as the overall iteration steps for the 47 Tuc analyses:

```
  1) fitposxy25  evaluate higher order registration model terms
  2) fitpsf25    evaluate the differential PSF for focus tracking
  3) difcon      set up frame set that isolates effects of focus changes
  4) submodc     create first set of decent difference images
  5) skysub      using difference images evaluate frame-to-frame background
 change of sky
  6) cntcry      analyze cosmic ray stats, adjust elimination threshold
  7) craye       evaluate model image with first use of focus compensation
  8) fitposxy25  reset registration using focus knowledge
  9) fitpsf25    reset differential PSFs
 10) difcon      reset differential PSF image frames
 11) submodc     new setting of diff images, just for verification
 12) skysub      verification step -- should be nulled out pretty well
 13) cntcry      analyze cosmic ray stats, adjust elimination threshold
 14) craye       iterate model evaluation
 15) fitposxy25  iterate registration
 16) fitpsf25    iterate PSF solution
 17) difcon      new setting of diff PSF images
 18) craye       final run of this -- turn on cosmic ray growth
 19) fitposxy25  final tweak of registration
 20) fitpsf25    final tweak of PSFs
 21) difcon      final setting of differential PSF images
 22) submodc     final production of difference images
   [As with the first block the precise point at which to stop is
   not well defined, but watching the level of changes on successive
   updates to registration and differential PSF kernal changes
   suggested the above as reasonable.]
```

(3) Intensity extractions over the full set of stars.

make.aperset

-- This is simply a tool used to analyze aspects of the star list as provided by DAOPHOT run on the over-sampled image. Allows estimation of contamination from neighbors. Imposes cuts in magnitude and color, staying away from edges, bad pixels, etc. [This should not be required for Kepler data.]

make.sortcon

-- Simply used to reorder and reformat output from aperset, ordered in terms of increasing level of contamination.
[This may not be needed for Kepler data.]

make.drpstr

-- Applies further cuts on the star list based on proximity to saturated pixels, vignetted regions etc.
[Again, not likely needed for Kepler since cuts on stars to be followed should have been done in setting up the observations.]

make.diffit

-- This is the code that makes use of the star list positions and performs both aperture and PSF fitting photometric extractions for intensity differences on the full set of difference images.
[A current known weakness, maybe not very important, is that the PSF is not carefully developed at the position of each star. This would be a disaster if fitting to direct images, not very important in difference images.]

make.psfdecrpcv

-- This code takes the intensity extractions for N stars on M frames and does some cleaning on the time series, e.g., enforces constancy of an ensemble mean and deprojects intensity changes versus a linear fit to x,y variation history. Output of this is ostensibly the final set of relative time series.
[I would expect the SOC to adopt their own approach to this.]

Assuming that things work right there isn't any iteration of steps in this 3rd block.
If actually adopting my codes for test purposes on existing HST data it would be advisable to have further notes detailing the parameters and files that need to be managed during the analyses.

Appendix E

Proof that the Distribution of Observation Noise Does Not Affect the Value of the Number of Independent Tests

Here we sketch a proof that the value for N_{EIT} does not depend on the noise distribution assumed for the observations. As discussed in the text, the distribution of the individual detection statistics may well be Gaussian even if the observation noise is not. For the purposes of the proof we will assume that the detection statistic, **l**, is a function of an N(0,1) process, **x**. Moreover, let us restrict **l** to be a zero–mean, unit variance random variable. Let $\mathbf{l} = h(\mathbf{x})$ establish the relationship between **x** and **l** and let $h(\mathbf{x})$ be strictly monotonic increasing: $h(x_1) < h(x_2)$ iff $x_1 < x_2$. This does not limit the variety of noise distributions that can be considered as a function h can always be found relating two given distributions (88). An example is $h(\mathbf{x}) = \mathbf{x}^3/\sqrt{15}$, whose corresponding density possesses extremely long tails in comparison with an N(0,1) process. Indeed, even the sum of 100 independent samples has significant tails compared to an N(0,1) process. Now, the properties of $h(x)$ imply that

$$F_l(y) = P\{\mathbf{l} \leq y\} = P\{\mathbf{x} \leq h^{-1}(y)\} = F_x(h^{-1}(y)). \tag{E.1}$$

Thus, there is a clear functional relationship between the distribution of **x** and the distribution of **l**. Additionally, this functional relationship carries over to the maximum detection statistic over a given search, $\mathbf{l_{max}} = \max_i\{l_i\}$ and the maximum of the corresponding Gaussian deviates, $\mathbf{x_{max}} = \max_i\{x_i\} : \mathbf{l_{max}} = h(\mathbf{x_{max}})$. Hence, $F_{l_{max}}(y) = F_{x_{max}}(h^{-1}(y))$. Let N_{EIT} be the effective number of independent tests performed in searching for transiting planets for the Gaussian detection statistics $\{x\}$:

$$\overline{F}_{x_{max}}(x) = N_{\mathrm{FA}}/N_{stars} \approx 1 - F_x(x)^{N_{\mathrm{EIT}}} \tag{E.2}$$

near $x = x_0$. But $x = h^{-1}(y)$ for some real number, y. So

$$\overline{F}_{x_{max}}(x) = \overline{F}_{x_{max}}(h^{-1}(y)) = \overline{F}_{l_{max}}(y), \tag{E.3}$$

and

$$F_x(x)^{N_{\mathrm{EIT}}} = F_x(h^{-1}(y))^{N_{\mathrm{EIT}}} = F_l(y)^{N_{\mathrm{EIT}}}. \tag{E.4}$$

Thus, $\overline{F}_{l_{max}}(y) \approx 1 - F_l(y)^{N_{\mathrm{EIT}}}$ near $y = h(x_0) = y_0$. Therefore, the distribution of $\mathbf{l_{max}}$ can be approximated by the distribution obtained from the process of choosing the maximum of N_{EIT} draws from the distribution of **l** in the region of interest, i. e., near $\overline{F}_{l_{max}} = N_{\mathrm{FA}}/N_{stars}$, which is the desired result.

Appendix F

FORTRAN Listing for Confidence Level Assessment

```
      program runcountlmult

c     Reads in file containing unnormalized single event statistics,
c     l1, the energy of each statistic E1 used to normalize multiple
c     event statistics c the specifications for the output histogram,
c     xmin, xmax and dx, and the c number of trials to run, ktrials,
c     and the number of transits in each c event to be computed,
c     ntrans

      integer NMAX,skip,CNTMAX
      real thresh
      parameter (NMAX=200000,skip=99,CNTMAX=4,thresh=6.)
      integer ktrials,nevents,ntrans
      real*4 xmin,xmax,dx,l1(NMAX),E1(NMAX)
      real*4 ltot,Etot,detstati,olddetstati,lmax
      character*20 outfile
      !integer idum, iran
      integer nhist,ihist
      real*8 hist(NMAX)
      !real*4 ran2
      integer ihiststat
      integer counter(CNTMAX),fact(CNTMAX),naddi,nadd
      integer oldcounter(CNTMAX)
      logical ex
      integer i,j

c     declarations for keeping track of state of ran2
      integer NTAB
      parameter(NTAB=32)
      integer idum2,iy,iv(NTAB)

      common /ran2blk/ idum2,iy,iv
```

```
c     Begin program

c     initialize fact
      fact(1)=1
      do i=2,CNTMAX
        fact(i)=i*fact(i-1)
      enddo

      !call readseed(idum2,iy,iv,idum)

c     open and read in setup.txt
      call readsetup(ktrials,xmin,xmax,dx,outfile,ntrans,nevents,
     .               l1,E1)
      !print *,’ktrials=’,ktrials

c     determine number of output histogram bins
      nhist=(xmax-xmin)/dx+1

c     initialize hist bins to 0
      do i=1,nhist
        hist(i)=0.
      enddo

c     check to see if the output file already exists
c     if it does, read it in and initialize hist to the
c     existing values, else set counter to [1,1,1,0]
      inquire(file=outfile, exist=ex)
      if (ex) then
        call readoutput(outfile,hist,nhist,counter,ntrans)
      else
        do i=1,ntrans-1
          counter(i)=1
        enddo
        counter(ntrans)=0
      endif


      print *,’xmin’,xmin,’xmax’,xmax
      print *,’nevents’,nevents

c     Start loop
      detstati=10.
      lmax=-10.
      i=0
      do while (counter(1).lt.nevents)
        i=i+1
        olddetstati=detstati
        do j=1,ntrans
          oldcounter(j)=counter(j)
        enddo
        call incrcounter(counter,ntrans,nevents)
        naddi=nadd(counter,ntrans,fact)
        !print *,’[’,(counter(j),j=1,ntrans),’]’;pause
        !print *,’naddi=’,naddi
        !pause
        ltot=0.
        Etot=0.
        do j=1,ntrans
          ltot=ltot+l1(counter(j))
          Etot=Etot+E1(counter(j))
        enddo
        detstati=ltot/sqrt(Etot)
        ihist=ihiststat(detstati,xmin,xmax,dx)
        hist(ihist)=hist(ihist)+naddi
        !lmax=max(lmax,detstati)
        if ((i/1000000)*1000000-i.eq.0)  then
          print *,i/1000000,detstati,
     .      ’[’,(counter(j),j=1,ntrans),’] naddi=’,naddi
          lmax=-10.
        endif

        if ((i/10000000)*10000000-i.eq.0) then
          call writeoutput(outfile,hist,nhist,xmin,dx,counter,ntrans)
          !call writeseed(idum2,iy,iv,idum)
        endif

        !if(olddetstati.lt.6.and.detstati.lt.6..and.
c    .      counter(2).gt.1) then
        ! print *,i,’l=’,detstati,’
c    .      cnt=[’,(counter(j),j=1,ntrans),’]’
        !endif

        if(detstati.ge.6.)
     .    counter(ntrans)=counter(ntrans)+skip

        if (detstati.lt.thresh) then
          j=2
          do while(j.le.ntrans)
            if(oldcounter(j).eq.nevents) counter(j-1)=nevents
            j=j+1
          enddo
          if(detstati.lt.6.) counter(ntrans)=nevents
        endif

        enddo !i

        call writeoutput(outfile,hist,nhist,xmin,dx,counter,ntrans)

c       call writeseed(idum2,iy,iv,idum)

      end

c     declarations for keeping track of state of ran2
      block data
      integer NTAB
      parameter(NTAB=32)
      integer idum2,iy,iv(NTAB)

      common /ran2blk/ idum2,iy,iv

      DATA idum2/123456789/, iv/NTAB*0/, iy/0/
      end

c**********************************************************************
      subroutine writeoutput(outfile,hist,nhist,xmin,dx,counter,ntrans)

      character*20 outfile
      integer nhist,counter(ntrans),ntrans
      real*8 hist(nhist)
      integer i
      real xmin, dx, xi

      open(unit=50,file=outfile,form='formatted',status='unknown')
      xi=xmin-dx
      do i=1,nhist
        xi=xi+dx
        write(50,*) xi,hist(i)
      enddo
      close(50)

      open(unit=50,file='counter.txt',form='formatted',
     .  status='unknown')
      write(50,*) (counter(i),i=1,ntrans)
      print *,(counter(i),i=1,ntrans)
      close(50)

      return
      end


c**********************************************************************
      subroutine readoutput(outfile,hist,nhist,counter,ntrans)

      character*20 outfile
      integer nhist,counter(4)
      real*8 hist(nhist)
      integer i
      real xi

      open(unit=50, file=outfile, form='formatted', status='old')

      do i=1,nhist
        read(50,*) xi,hist(i)
      enddo
      close(50)
      open(unit=50, file=’counter.txt’, form=’formatted’, status=’old’)
      read(50,*) (counter(i),i=1,ntrans)
      close(50)

      return
      end
c**********************************************************************
      subroutine readsetup(ktrials,xmin,xmax,dx,outfile,ntrans,nevents
     .                       ,l1,E1)

      integer ktrials,ntrans,nevents,NMAX
      parameter (NMAX=200000)
      real*4 xmin,xmax,dx,l1(NMAX),E1(NMAX)
      character*10 fname
      character*20 outfile,eventfile
      logical ex

c     This subroutine reads in run parameters for runbootseg

      fname=’setup.txt’
      inquire(file=fname, exist=ex)
      if (.not. ex) then
        print *, ’File SETUP.TXT does not exist!’
        pause
        stop
      else
        open(unit=50, file=fname, form=’formatted’, status=’old’)
        read(50, *) ktrials
        read(50, *) xmin
        read(50, *) xmax
        read(50, *) dx
        read(50, *) outfile
        read(50, *) ntrans
        read(50, *) eventfile
        close(50)
      endif

      inquire(file=eventfile, exist=ex)
      if (.not. ex) then
        print *, ’File ’,eventfile,’ does not exist!’
        pause
        stop
      endif
      open(unit=50, file=eventfile, form=’formatted’, status=’old’)
      nevents=0
  98  continue
      nevents=nevents+1
      read(50, *,END=99) l1(nevents), E1(nevents)
      goto 98
  99  continue
      close(50)
      return
      end


c***********************************************************************
      function ihiststat(detstati,xmin,xmax,dx)
      integer ihiststat
      real*4 detstati,xmin,xmax,dx
      if (detstati.lt.xmin) detstati=xmin
      if (detstati.gt.xmax) detstati=xmax
      ihiststat=int((detstati-xmin)/dx+.5)+1
      return
      end


c***********************************************************************
      FUNCTION ran2(idum)
      INTEGER idum,IM1,IM2,IMM1,IA1,IA2,IQ1,IQ2,IR1,IR2,NTAB,NDIV
      REAL ran2,AM,EPS,RNMX
      PARAMETER (IM1=2147483563,IM2=2147483399,AM=1./IM1,IMM1=IM1-1,
     *IA1=40014,IA2=40692,IQ1=53668,IQ2=52774,IR1=12211,IR2=3791,
     *NTAB=32,NDIV=1+IMM1/NTAB,EPS=1.2e-7,RNMX=1.-EPS)
      INTEGER idum2,j,k,iv(NTAB),iy
      !SAVE iv,iy,idum2
      common /ran2blk/ idum2,iy,iv
c     DATA idum2/123456789/, iv/NTAB*0/, iy/0/
      if (idum.le.0) then
      idum=max(-idum,1)
      idum2=idum
      do 11 j=NTAB+8,1,-1
        k=idum/IQ1
        idum=IA1*(idum-k*IQ1)-k*IR1
        if (idum.lt.0) idum=idum+IM1
        if (j.le.NTAB) iv(j)=idum
11      continue
      iy=iv(1)
      endif
      k=idum/IQ1
      idum=IA1*(idum-k*IQ1)-k*IR1
      if (idum.lt.0) idum=idum+IM1
      k=idum2/IQ2
      idum2=IA2*(idum2-k*IQ2)-k*IR2
      if (idum2.lt.0) idum2=idum2+IM2
      j=1+iy/NDIV
      iy=iv(j)-idum2
      iv(j)=idum
      if(iy.lt.1)iy=iy+IMM1
      ran2=min(AM*iy,RNMX)
      return
      END

c***********************************************************************
```

```
      subroutine readseed(idum2,iy,iv,idum)

c     This subroutine reads in seed parameters for ran2

      implicit none
      logical ex
      integer NTAB
      parameter(NTAB=32)
      integer idum,idum2,iy,iv(NTAB)

      inquire(file=’seed.txt’, exist=ex)
      if (ex) then
      open(unit=50, file=’seed.txt’, form=’formatted’, status=’old’)
      read(50, *) idum2
      read(50, *) iy
      read(50,*) iv
      read(50, *) idum
      else
      idum=-1 ! initialize
      endif
      close(50)
      return
      end

c***********************************************************************
      subroutine writeseed(idum2,iy,iv,idum)

c     This subroutine writes out seed parameters for ran2

      implicit none
      integer NTAB
      parameter(NTAB=32)
      integer idum,idum2,iy,iv(NTAB)

      open(unit=50,file=’seed.txt’,form=’formatted’,status=’unknown’)
      write(50, *) idum2
      write(50, *) iy
      write(50,*) iv
      write(50, *) idum
      close(50)

      return
      end

c***********************************************************************
      subroutine incrcounter(counter,ntrans,nevents)

      integer counter(ntrans),ntrans,nevents
      integer i,j

      counter(ntrans)=counter(ntrans)+1
      do i=ntrans,1,-1
      if (counter(i).gt.nevents) then
        counter(i-1)=counter(i-1)+1
        do j=i,ntrans
          counter(j)=counter(i-1)
        enddo
      endif
      enddo
      do i=2,ntrans
        if(counter(i).lt.counter(i-1)) counter(i)=counter(i-1)
      enddo

      return
      end

c***********************************************************************
      subroutine unique(vec,nvec,nuniq,nnuniq)

      integer vec(nvec),nvec,nuniq(nvec),nnuniq
      integer i,j,k

      do i=1,nvec
        nuniq(i)=0
      enddo

      i=1
      j=1
      k=1
      do while(j.le.nvec)
        if(vec(j).eq.vec(i)) then
          nuniq(k)=nuniq(k)+1
          j=j+1
        else
          k=k+1
          i=j
        endif
      enddo
      nnuniq=k

      return
      end

c***********************************************************************
      function nadd(counter,ntrans,fact)

      integer nadd,ntrans,nuniq(4),nnuniq,i,fact(ntrans)
      integer counter(ntrans)

      call unique(counter,ntrans,nuniq,nnuniq)

      nadd=fact(ntrans)
      i=1
      do while(i.le.nnuniq)
```

```
  nadd=nadd/fact(nuniq(i))
  i=i+1
enddo

return
end
```

Appendix G

MATLAB Listing for the Wavelet-Based Detector

```
function [l,lnorm]=qwavsearch(x,h0,tran,winlen)
% [l,lnorm]=qwavesearch(x,h0,tran,winlen)

if nargin<4, winlen=50; end

n=length(x);
n=2^floor(log2(n));
x=x(1:n)-median(x);
mtran=sum(tran~=0);

nh0=length(h0);
m=log2(n)-floor(log2(nh0))+1;

h1=flipud(h0).*(-1).^(0:nh0-1)’;
H0=fft(h0,n);
H1=fft(h1,n);

X=fft(x);
T=fft(tran,n);

l=0;
lnorm=0;

if n>2^14
h=waitbar(0,’Progress’);
end

k=winlen;

for i=1:m-1

% get wavelet coeffs at scale i for data and for transit pulse
Wxi=real(ifft(X.*H1));
Wtrani=real(ifft(T.*H1));
```

```
X=X.*H0;
T=T.*H0;
H0=[H0(1:2:end);H0(1:2:end)];
H1=[H1(1:2:end);H1(1:2:end)];

k=min(k*2,n);
Wstd_2=circshift(movcircstd(Wxi,k),-k).^-2;

SNRi=circfilt(flipud(Wtrani.^2),Wstd_2);
lnorm=lnorm+SNRi/2^i;

Li=circfilt(flipud(Wtrani),Wxi.*Wstd_2);
l=l+Li/2^i;

if n>2^14
waitbar(i/m)
end

end

Wxi=real(ifft(X));
Wtrani=real(ifft(T));

k=min(n,winlen*2^(m+1));
Wstd_2=movcircstd(Wxi,k).^-2;

Li=circfilt(flipud(Wtrani),Wxi.*Wstd_2);
l=l+Li/2^(m-1);

l=circshift(l,-mtran);

SNRi=circfilt(flipud(Wtrani.^2),Wstd_2);
lnorm=lnorm+SNRi/2^(m-1);
lnorm=circshift(lnorm,-mtran);
lnorm=sqrt(lnorm);

if n>2^14
waitbar(1)
close(h)
end

return
```

Bibliography

[1] Albrow, M. D., Gilliland, R. L., Brown, T. M., Edmonds, P. D., Guhathakurta, P., Sarajedini, A. 2001, ApJ, 559, 1060

[2] Andersen, B., Leifsen, T., Appourchaux, T., Frohlich, C., Jiménez, A., Wehrli, C. 1998, in Proceedings of the SOHO 6/GONG 98 Workshop, Structure and Dynamics of the Interior of the Sun and Sun-like Stars (Boston), 83

[3] Andersen, B. 1996, A&A, 312, 610

[4] Appourchaux, T. & 14 other authors 2000, ApJ, 538, 401

[5] Baliunas, S. L., Vaughan, A. 1985, Ann Rev Astron Ap, 23

[6] Bartolini, C., & 15 other authors 1983, A&A117, 149

[7] Batalha^a, N., M., Jenkins, J., Basri, G. S., Borucki, W. J., & Koch, D. G. 2002. Proceedings of the 1st Eddington Workshop: Stellar Structure and Habitable Planet Finding, Cordoba, Spain, in press

[8] Batalha^b, n, M., Jenkins, J. M., Basri, G. S., Borucki, W. J. and Koch, D. G. "Stellar variability and its implications for photometric planet detection with Kepler," in *ESA SP-485: Stellar Structure and Habitable Planet Finding*, B. Battrick ed., Noordwijk, Cordoba, pp. 35–40, 2002

[9] Borde, P., Rouan, D., & Léger, A. 2001, Comptes Rendus. Acad. Sci. Paris, serie IV, 2, 1049

[10] Borucki, W. J., Caldwell, D. A., Koch, D. G., Webster, L. D., Jenkins, J. M., Ninkov, Z., & Showen, R. L. 2001, PASP, 113, 439

[11] Borucki, W. J., Koch, D. G., Dunham, E. W., & Jenkins, J. M. 1997, in ASP Conf. Ser. 119, Planets Beyond the Solar System and The Next Generation of Space Missions, ed. David Soderblom (San Francisco, ASP), 153

[12] Borucki, W. J., Jenkins, J. M., Scargle, J., Koch, D., & Doyle, L. R. 1996, BAAS 28, 1115

[13] Borucki, W. J., Scargle, J. D. & Hudson, H. S. 1985, ApJ, 291, 852

[14] Borucki, W. J., & Summers, A. L. 1984, Icarus, 58, 121

[15] Brown, T. M., Personal Communication, 1999

[16] Brown, T. M., Charbonneau, D., Gilliland, R. L., Noyes, R. W., & Burroughs, A. 2001, ApJ, 552, 699

[17] Brown, T. M. & Charbonneau, D. 2000, in ASP Conf. Ser. 219, Disks, Planetesimals, and Planets, ed. F. Garzón & T. J. Mahoney (San Francisco, ASP), 54

[18] Butler, R. P., Marcy, G. W., Fischer, D. A., Brown, T. M., Contos, A. R., Korzennik, S. G., Nisenson, P., Noyes, R. W. 1999, ApJ, 526, 916

[19] Cameron, A. C., Horne, K., Penny, A., & James, D. 1999, Nature, 402, 751

[20] Castellano, T., Jenkins, J. M., Trilling, D. E., Doyle, L. R., & Koch, D. G. 2000, ApJ, 532, L51

[21] Chapa, J. O., & Rao, R. M. 2000, IEEE Trans. on Sig. Proc., 48, 3395

[22] Charbonneau, D., Brown, T. M., Noyes, R. W., Gilliland, R. L. 2002, ApJ, 568, 377

[23] Charbonneau, D., Brown, T. M., Latham, D. W., & Mayor, M. 2000, ApJ, 529, L45

[24] Chugainov, P. F. 1973, Izv Krym Astrofiz Obs, 48, 3

[25] Charbonneau, D., Noyes, R. W., Korzennik, S. G., Nisenson, P., Jha, S., Vogt, S. S., & Kibrick, R. I. 1999, ApJ, 522, L145

[26] Cochran, W. D., Hatzes, A. P., Butler, R. P., & Marcy, G. W. 1997, ApJ, 483, 457

[27] Cody, A. M., Sasselov, D. D. 2002, ApJ, 569, 451

[28] Cram, L. E., & Kuhi, L. V. 1989, FGK Stars and T Tauri Stars, NASA SP-502, (Washington, DC: US Gov. Printing Office)

[29] Debauchies, I. 1988, Comm. on Pure & Appl. Math., 41, 909

[30] Deeg, H. J., and Doyle, L. R., "VAPHOT - a package for precision differential aperture photometry," in Proceedings of the Photometric Extrasolar Planets Detection Workshop, NASA Ames Research Center, in press, 2000

[31] Deeg, H. J., Favata, F., & the *Eddington* Science Team 2000, in ASP Conference Series 219, Disks, Planetesimals, and Planets, ed. F. Garzón & T. J. Mahoney (San Francisco, ASP), 578

[32] Deeg, H. J., et al. 1998, A&A, 338, 479

[33] Defay, C., Deleuil, M., & Barge, P. 2001, A&A, 365, 330

[34] Deubner, F.-L., & Gough, G. O. 1984, Ann Rev Astron Ap, 22, 593

[35] Doyle, L. R., et al. 2000, ApJ, 535, 338

[36] Doyle, L. R., 1996, Photometric Spinoffs from the Kepler (Formerly FRESIP) Mission, SETI Institute whitepaper, March 27

[37] Dravins, D., Lindegren, L., Mezey, E., & Young, A. T. 1998, PASP, 110, 610

[38] Duquennoy, A., & Mayor, M. 1991, A&A, 248, 485

[39] Everett, M. E., Howell, S. B., van Belle, G. & Ciardi, D. 2002, PASP, in press

[40] Frandsen, S., Dreyer, P., & Kjeldsen, H. 1989, A&A, 215, 287

[41] Fröhlich, C., et al.1997, SoPh, 175, 267

[42] Frölich, C., 1987, JGR, 92, 796

[43] Gilliland, Ron, Personal Communication, 2003.

[44] Gilliland, R. L., et al. 2000, ApJ, 545, L47

[45] Gilliland, R. L., & Brown T. 1992, PASP, 104, 582

[46] Gray, D. F., 1992, The Observation and Analysis of Stellar Photospheres (2nd edition; Cambridge: Cambridge Univ. Press)

[47] Green, D., Mathews, J., & Seager, S. 2002, Presented at the Scientific Frontiers Conference in Research on Extrasolar Planets, Carnegie Institution, June 18-21

[48] Green, R. H., Spherical Astronomy, Cambridge University Press, Cambridge, 1985

[49] Hartley, R. V. L. July 1928, Bell Sys. Tech. Jour., 535

[50] Hayes, M. H., Statistical Digital Signal Processing and Modeling, John Wiley & Sons, New York, 1996

[51] Haywood, M., Robin, A. C. & Crézé , M. 1997, A&A, 320, 428

[52] Haywood, M., Robin, A. C. & Crézé , M. 1997, A&A, 320, 420

[53] Henry, G. W. 1999, PASP, 111, 845

[54] Henry, G. W., Marcy, G. W., Butler, R. P., & Vogt, S. S. 2000, ApJ, 529, L41

[55] Hohenkerk, C. Y., Yallop, B. D., Smith, C. A., & Sinclair, A. T. 1992, in Explanatory Supplement to the Astronomical Almanac, ed. P. Kenneth Seidelmann, University Science Books, Sausalito, CA, p. 95

[56] Horne, J. H., & Baliunas, S. L. 1986, ApJ, 302, 757

[57] Howell 1989, PASP, 101, 616

[58] Hudson, H. S., 1988, Ann Rev Astron Ap, 26, 473

[59] Huffman, D. A. 1952, Proc. Inst. Radio. Engineers, 40, 1098

[60] Jenkins, J. M., "The Impact of Solar-like Variability on the Detectability of Transiting Terrestrial Planets," *Astroph. J.* **575**, pp. 493-505, 2002

[61] Jenkins, J. M., Borucki, W. J., Dunham, E. W., & McDonald, J. S. 1997, ASP Conf. Ser., 199, 277

[62] Jenkins, J. M., Caldwell, D. A., & Borucki, W. J. 2002, ApJ, 564, 495

[63] Jenkins, J. M., Doyle, L. R., & Deeg, H. J. 2000, Acta Astronautica, 46, 693

[64] Jenkins, J. M., Witteborn, F., Koch, D. G., Dunham, E. W., Borucki, W. J., Updike, T. F., Skinner, M. A. & Jordan, S. P. 2000, Proc. SPIE, 4013, 520

[65] Jorden, P. R., Deltorn, J.-M., and Oates, A. P., 'Nonuniformity of CCDs and the effects of spatial undersampling,' in *Instrumentation in Astronomy VIII*, D. L. Crawford, and E. R. Craine, eds., Proc. SPIE, **2198**, pp. 836–850, 1994

[66] Kawaler, S. D. 1989, ApJ, 353, 65

[67] Kay, S. 1999, IEEE Trans. on Sig. Proc., 47, 10

[68] Kay, S. 1998, Fundamentals of Statistical Signal Processing: Detection Theory, (Upper Saddle River: Prentice-Hall PTR)

[69] Koch, D. G., Borucki, W., Webster, L., Dunham, E., Jenkins, J., Marriott, J. & Reitsema, H. J. 1998, Proc. SPIE, 3356, 599

[70] Koch, D. G., Borucki, W. J., Dunham, E. W., Jenkins, J. M., Webster, L., & Witteborn, F. 2000, Proc. SPIE, 4013, 508

[71] Koch, D. G., Personal communication, 2001

[72] D. Koch, W. Borucki, E. Dunham, J. Geary, R. Gilliland, J. Jenkins, D. Latham, E. Bachtell, D. Berry, W. Deininger, R. Duren, N. Gautier, L. Gillis, D. Mayer, C. Miller, D. Shafer, C. Sobeck, C. Stewart, M. Weiss, "Overview and status of the Kepler Mission", Proc. SPIE, **5497**, 2004.

[73] Koch, D. G., README for RA2pix.c, ARC009, 2004

[74] Koch, D. G., RA2pix.c source code, ARC009, v. 1.2, 2004

[75] Knollenberg, R. G., & Hunten, D. M. 1980, JGR, 85, 8036

[76] Kozhevnikov, V. P. and Kozhevnikova, A. V. 2000, IBVS, 5252

[77] Lacy, C. H., ApJ, 218, 444

[78] Laughlin, G. 2000, ApJ, 545, 1064

[79] Loeb, A., & Gaudi, B. S. 2003, astro-ph/0303212

[80] Mayor, M., & Queloz, D. 1995, Nature, 378, 355

[81] McDonough, R. N., and Whalen A. D., Detection of Signals in Noise, 2^{nd} Ed., Academic Press, San Diego, 1995

[82] Messina, S., Rodono, M., & Guinan, E. F. 2001, A&A, 366, 215

[83] Nickles, Neal, Personal Communication, 2003

[84] Noyes, R. W., Hartmann, L., Baliunas, S. L., Dunkan, D. K., & Vaughan, A. H. 1984, ApJ, 279, 763

[85] Nyquist, H., April 1924, Bell Sys. Tech. Jour. 324

[86] Nyquist, H. 1928, AIEE tran., 47, 617

[87] Olsen, E. H. 1977, A&A, 58, 217

[88] Papoulis, A. 1984, Probability, Random Variables, and Stochastic Processes, (New York: McGraw Hill)

[89] Perryman, M. A. C., 2000, Reports on Progress in Physics, 63, 1209

[90] Philbrick, Rob, Personal Communication, 2004

[91] Press, W. H., S. A. Teukolsky. W. T. Vetterling, B. P. Flannery 1992, Numerical Recipes in Fortran 77: The Art of Scientific Computing, 2nd Edition (New York: Cambrige University Press)

[92] Rabello-Soares, M. C., Roca Cortes, T., Jimenez, A., Andersen, B. N., Appourchaux, T. 1997, A&A, 318, 970

[93] Radick, R. R., Lockwood, G. W., Skiff, B. A., & Baliunas, S. L. 1998, ApJS, 118, 239

[94] Remund, Q. P., Jordan, S. P., Updike, T. F., Jenkins, J. M., & Borucki, W. J. 2001, Proc. SPIE, 4495, 182

[95] Remund, Q. P., Jordan, S. P., Updike, T. F., Jenkins, J. M., and Borucki, W. J., 'Kepler System Numerical Model for the Detection of Extrasolar Terrestrial Planets,' in *Instruments, Methods, and Missions for Astrobiology IV*, R. B. Hoover, G. V. Levin, R. R. Paepe, and A. Y. Rozanov, eds., Proc. SPIE, **4495**, pp. 182–191, 2002

[96] Resnick, R., Introduction to Special Relativity, John Wiley & Sons, New York, 1968

[97] Riedel, K. S., & Sidorenko, A. 1995, IEEE Trans. Sig. Proc., 43, 188

[98] Robin, A. C., & Crézé, M. 1986, A&A, 157, 71

[99] Robichon, N., & Arenou, F. 2000, A&A, 355, 295

[100] Robinson, Wei, Borucki, Dunham, Ford, & Granados 1995, PASP, 107, 1094

[101] Rodonó, M., et al.1986, A&A, 165, 135

[102] Seager. S., Whitney, B. A., & Sasselov, S. S. 2000. ApJ, 540, 505

[103] Schneider, J., et al.1998, in Origins, ASP Conf. Ser. 148, San Francisco, 298

[104] Schneider, J., & Chevereton, M. 1990, A&A, 232, 251

[105] Shannon, C. E. 1949, The Mathematical Theory of Communication, (Board of Trustees of the Univ. of Illinois: University of Illinois Press)

[106] Smith, M. J. T., & Barnwell III, T. P. 1984, Proc. IEEE Int. Conf. Acoust., Speech, and Sig. Proc, San Diego, CA

[107] Sterken, C., & Jaschek, C. 1996, Light Curves of Variable Stars (Cambridge U. Press, Cambridge)

[108] Vetterli, M., & Kovačević, J. 1995, Wavelets and Subband Coding, (Englewood Cliffs: Prentice-Hall PTR)

[109] Walden, A. T., Percival, D. B., & McCoy, E. J. 1998, IEEE Trans. Sig. Proc. 46, No. 12, 3153

[110] Wertz, J. R., Spacecraft Attitude Determination and Control, Kluwer Academic Publishers, Dordrecht, 1978

[111] West, R. A., Stobel, D. F., & Tomasko, M. G., 1986, Icarus, 65, 161

[112] Willson, R. C. & Hudson, H. S. 1991, Nature, 351, 42

[113] Young, A. T., et al. 1991, PASP, 103, 221