ALTSched: Improved Scheduling for Time-domain Science with LSST

Daniel Rothchild[1] , Christopher Stubbs[2] , and Peter Yoachim[3]

[1] Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, USA; drothchild@berkeley.edu

[2] Department of Physics, Center for Astrophysics, Harvard University, Cambridge, MA, USA

[3] Department of Astronomy, University of Washington, Seattle, WA, USA

Abstract

Telescope scheduling is the task of determining the best sequence of observations (pointings and filter choices) for a survey system. Because it is computationally intractable to optimize over all possible multiyear sequences of observations, schedulers use heuristics to pick the best observation at a given time. A *greedy* scheduler selects the next observation by choosing whichever one maximizes a scalar merit function, which serves as a proxy for the scientific goals of the telescope. This sort of bottom-up approach for scheduling is not guaranteed to produce a schedule for which the sum of merit over all observations is maximized. As an alternative to greedy schedulers, we introduce ALTSched, which takes a top-down approach to scheduling. Instead of considering only the next observation, ALTSched makes global decisions about which area of sky and which filter to observe in, and then refines these decisions into a sequence of observations taken along the meridian to minimize airmass. We implement ALTSched for the Large Synoptic Survey Telescope (LSST) and show that it equals or outperforms the baseline greedy scheduler in essentially all quantitative performance metrics. Due to its simplicity, our implementation is considerably faster than OPSIM, the simulated greedy scheduler currently used by the LSST Project: a full 10-yr survey can be simulated in 4 minutes, as opposed to tens of hours for OPSIM. LSST's hardware is fixed, so improving the scheduling algorithm is one of the only remaining ways to optimize LSST's performance. We see ALTSched as a prototype scheduler that gives a lower bound on the performance achievable by LSST.

Key words: methods: observational – telescopes

Online material: color figures, supplementary file

1. Introduction

The Large Synoptic Survey Telescope (LSST) is a large ground-based survey system scheduled to begin operation in 2022. With the capability to survey the entire southern sky in six bands about once a week, LSST will enable time-domain astronomy at an unprecedented scale. The telescope was designed with four primary science goals in mind: to understand dark matter and dark energy; to catalog the solar system; to study the structure and formation of the Milky Way; and, perhaps most uniquely suited to LSST's particular characteristics, to explore the large frontier of time-domain astronomy. These science goals motivate a range of competing technical metrics that LSST should optimize. Presenting a full list of science cases for LSST and accompanying technical metrics is beyond the scope of this paper; we refer readers to the LSST observing strategy whitepaper (LSST Science Collaborations et al. 2017). Here we present only a few representative

examples (with science cases in parentheses), which we believe are especially sensitive to scheduling decisions:

long observing seasons (parallax measurements, AGN variability, few-months-long transients) ⇔ high cadence (faster transients, better characterization of light curves)

large survey area (large-scale structure, weak lensing, galaxy/star surveys) ⇔ higher coadded depth (studying fainter objects)

few filter changes (maximize survey efficiency) ⇔ many filter changes (obtain nightly colors for variables and transients)

large overlap between neighboring observations (rapid revisits for very fast transients) ⇔ small overlap between neighboring observations (cover a larger area per unit time)

multiple observations per field per night (link asteroids, probe ∼hour-scale variability) ⇔ one observation per field per night (transients that do not change in ∼1 hr)

With the LSST Project well into the construction phase, the system's hardware characteristics (aperture, field size, slew rate, field of view, sensor quantum efficiency, etc.) are fixed. Apart from actively engineering the weather in Chile, the only remaining opportunities to extract additional performance from the LSST system are in the scheduler and in the data reduction pipeline. LSST's existing scheduler, called OPSIM, is a greedy

algorithm: it chooses the next field to observe by maximizing a scalar merit function intended as a proxy for the scientific merit of observing the field. This approach is appealing in its apparent simplicity, but as we show below, even after over a decade of development, OPSIM underperforms on many science metrics, particularly in the time-domain science so critical to LSST's mission. We argue here that the fault lies not in any implementation detail, but rather in the fundamental nature of greedy algorithms: because they make scheduling decisions only one or a few observations in advance, the parameters of the algorithm give scientists little or no control over many 10-yr global properties of the schedule.

To remedy this, we introduce and analyze an alternative scheduling algorithm, ALTSched, which makes scheduling decisions with a top-down hierarchical approach, instead of using a purely bottom-up local merit function. We defer a full description of ALTSched to Section 3, but simply put, our algorithm first decides which region of sky to observe (nightly), then which particular area and filter to use (hourly), and only then which individual pointings to observe (minute scale). In contrast, OPSIM chooses every next observation ab initio, with no planning horizon longer than 30 s and no global strategy, except what is encoded indirectly through the merit function.

One major advantage of ALTSched is that the parameters of the algorithm directly control final properties of the schedule. For example, ALTSched observes either the northern or southern sky each night. Because LSST can observe about half of the sky each night, we can obtain a universal 2-day revisit cadence simply by assigning even-numbered nights to the northern sky and odd-numbered nights to the southern sky. Other cadences (e.g., with some 1- and 3-day revisits) can be achieved by partitioning the nights between north and south differently. Importantly, there is no known way to similarly modify this distribution of revisit times in OPSIM, except by adjusting the weights and penalties of the merit function specifically to mimic the behavior of ALTSched.

As of the writing of this paper, the LSST Project is planning to run a suite of OPSIM simulations in order to choose a final observing strategy for LSST. One point of this paper is to argue that optimizing over the parameters of OPSIM in this way will not yield an optimal survey strategy. Taking the example above one step further before giving a fuller analysis in Section 4, the gaps between observations to a particular sky pixel resemble an exponential distribution for every OPSIM run we have analyzed, over a wide range of OPSIM's parameters. An exponential distribution of revisit times is consistent with the timing of observations following a random process. In ALTSched, the shape of this distribution can be easily controlled. Instead of running more and more OPSIM simulations, we therefore advocate exploring how OPSIM can be made to reproduce the survey strategy of ALTSched, and how ALTSched's cadence can be further improved.

In the sections that follow, we first summarize the LSST baseline scheduler and the tools used to assess the scientific performance of various alternatives. We then describe the implementation of our alternative scheduler, ALTSched, and make quantitative performance comparisons to the LSST baseline. We close with a discussion and some suggestions for future work.

2. OPSIM

To provide context for ALTSched, the algorithm we propose below, we first introduce LSST's current baseline scheduling algorithm.[4]

LSST's current scheduler is part of a package called the Operations Simulator (OPSIM; Delgado & Reuter 2016). OPSIM uses a greedy algorithm to choose fields based on a proposal system, where abstractions of different scientific proposals give a score for each candidate field, and the scheduler chooses the field with the highest combined score, or merit. OPSIM considers a number of criteria in its merit function, including time since last observation, coadded depth achieved thus far, airmass and sky brightness at the proposed pointing, slew time to reach the pointing, etc. For a full description of the complex merit function used for MINION_1016, we refer readers to Delgado et al. (2014, Section 5). Once an observation is chosen and executed, the merit scores are recalculated for every candidate field, and the process repeats. The parameters of OPSIM are the weights and penalties associated with the various inputs to the merit function. Although easily interpretable in the context of local decision making, these parameters have no simple connection to many of the scheduling decisions we actually want to make: the average and distribution of season lengths and of revisit times; the colors (or lack thereof) obtained during a single night; the uniformity of light curve sampling over time; etc. In contrast, as we show below, ALTSched's parameters directly control these properties.

Included with OPSIM is a module that simulates the system hardware, weather, seeing conditions, and downtime. For fairness of comparison, ALTSched uses the exact same calculations, though we reimplement the simulator to improve computational efficiency. In particular, we obey all physical constraints on the telescope, including making no observations below LSST's minimum elevation or in the zenith avoidance area, accounting for readout time, shutter time, slew speed, settle time, optics correction time, and filter change time, only using five of LSST's six bandpass filters per night, etc. Taking slew time as an example: both OPSIM and ALTSched use the same slew calculation, described in Delgado et al. (2014, Section 6). In short, the slew time is calculated assuming

---

4 The LSST scheduler is undergoing continual development, and the version of OPSIM described and compared with in this article has been superseded, but it was the baseline plan at the time this work was completed.

uniform accelerated motion of the telescope mount and dome, plus some settle time.

To compare simulated surveys, the LSST Project has developed a useful suite of analysis tools, the Metrics Analysis Framework (MAF), which can evaluate candidate schedules by computing various performance metrics of interest (Jones et al. 2014). Examples include the number of Type Ia supernovae (SNe) light curves that meet certain criteria, the distribution of coadded depths over the observed region, and the anticipated uncertainty in parallax and proper motion measurements. This framework is critical for the fair evaluation of simulated surveys.

A number of OPSIM runs, or simulated surveys, have been released by the Project. At the time this work was carried out, the baseline simulated survey was MINION_1016, and we compare our results to those of MINION_1016. Later versions of OPSIM have improved on a number of metrics, but all runs we have analyzed since MINION_1016 suffer from poor time-domain performance. For example, none of the cadences released with the recent whitepaper call recover even half as many well-sampled SNe Ia without eliminating visit pairs or changing the exposure time (where "well-sampled" is as defined below). And to our knowledge, no OPSIM or feature-scheduler run at all has exceeded ALTSched in this metric. Results for MINION_1016 are presented in more depth in Section 4.

Several alternatives to OPSIM have also been proposed. Naghib et al. (2016) frame the scheduling problem as a Markovian decision process. Ridgway (2015) proposes scheduling LSST by dividing observing into blocks that are observed (and then reobserved) in an optimal manner. Ridgway's proposal is similar in spirit to the algorithm developed and described in this paper.

3. ALTSched

In this section, we introduce ALTSched, an alternative scheduling algorithm for LSST. In the next section, we introduce a number of metrics, and demonstrate quantitatively that ALTSched equals or outperforms the existing OPSIM baseline MINION_1016 on these metrics.

3.1. ALTSched Algorithm

ALTSched is a deterministic scheduling algorithm: in a given night, it does not adapt to current weather or seeing conditions. Although, in principle, adjusting a schedule to take prevailing weather conditions into account should only improve a scheduler, we justify our decision to use a nonadaptive algorithm in three ways: (1) maintaining a consistent cadence —i.e., avoiding long observation gaps—is difficult to do if the scheduler aggressively avoids regions with poorer observing conditions, since by random chance, some regions will go unobserved for a long time; (2) the best seeing tends to occur

on the meridian (at minimum airmass), which is where ALTSched observes anyway; (3) one strength of ALTSched is that it achieves such high performance even without dynamically reacting to observing conditions. We only expect our performance to improve with the judicious addition of poor-weather avoidance.

The algorithm itself is remarkably simple. Each night, it chooses whether to observe the northern or southern sky based on which region has received fewer visits so far, and then during the night it scans north and south of the meridian, taking 30 s exposures and then slewing by approximately one field width. LSST is on an alt/az mount, and so has a zenith-avoidance area. To observe the region of sky that passes over zenith, we therefore periodically scan to the east and west over that region. Each N/S scan is repeated twice in order to obtain two observations per night, separated by roughly 30–60 minutes. Before repeating a scan, we change filters so that every field is visited in two bands per night. We refer readers to the video in the supplementary material; the scanning strategy is much more easily explained visually than by text.

The exact pointings used are drawn from a fixed tiling that is randomly rotated by much more than a field size each night. If a pointing is not used in a given night, it is saved for use in a later night, so that gaps do not persist. The fixed tiling is chosen as a solution to the Thomson charge-distribution problem for $N = 3500$ (Thomson 1904).

3.2. Moon Avoidance

One deficiency in ALTSched is that it contains no provision for avoiding the Moon. MINION_1016 uses a conservative avoidance radius of 30°. This corresponds to 7% of the celestial sphere, so about 7% of observations should fall within this avoidance zone. However, 30° is likely too conservative in times near a new moon and in the redder filters. We therefore expect only a few percent of observations to be problematically close to the Moon. For observations that fall too close to the Moon, the simulated sky brightness is very high, leading to a low limiting depth for those observations. So any metric depending on depth should only improve by adding a Moon-avoidance module to ALTSched. Figure 1 shows histograms of the angular distance between ALTSched's observations and the Moon. Note that ALTSched avoids observing in the u and g bands at all when the Moon is up and bright, so there are few or no observations in u or g closer than 30° to the Moon.

3.3. Filter Allocation

Most simulated surveys of LSST assume that the telescope will visit each field twice per night, with a separation of ∼30–60 minutes, in order to link asteroid observations. To improve the cadence in each band, ALTSched usually carries out these two visits in different filters. To accomplish this with a minimal number of filter changes, we divide observations into
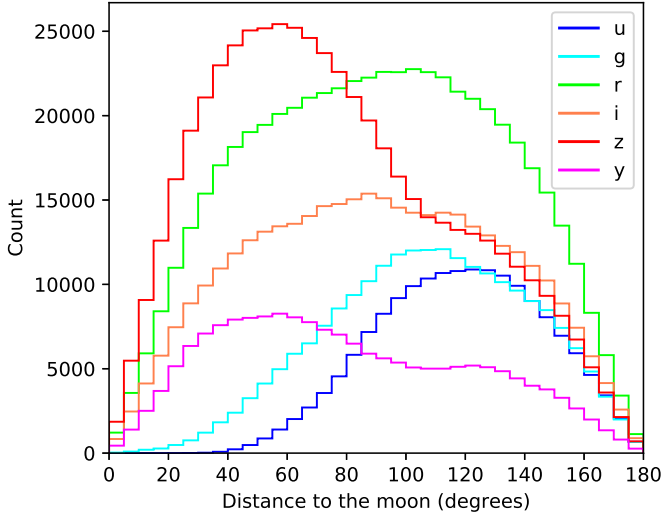
Figure 1. Histogram of angular distance between ALTSched's observations and the Moon. Only a small fraction of observations fall closer than 30° to the Moon, and of those, a number occur when the Moon is only slightly illuminated.

(A color version of this figure is available in the online journal.)

blocks that take ∼30–60 minutes to observe, and we visit each block twice back-to-back, changing filters before revisiting a block (but not between the revisit of a block and the first visit to the subsequent block). The video in the supplementary material (stacks.iop.org/PASP/131/115002/mmedia) demonstrates this filter allocation strategy. On average, we execute 11.4 filter changes per night, compared to MINION_1016's 4.3. This adds 1.3 s to our average slew time over MINION_1016, and allows us to double the cadence in each band.

Our filter allocation strategy is designed such that, in theory, visits to every sky pixel will cycle through the six filters in order. Assigning numbers 1 through 6 to LSST's six bandpass filters (*ugrizy*), and using arithmetic modulo 6, we start a night in some filter *i*, and during the night, before every revisit block, we switch from filter *k* to filter *k* + 1. The next night starts in filter *i* + 1, and the process repeats, ensuring that visits to each pixel cycle through all six filters. In the ideal case, every sky pixel gets a visit in every band once every six nights (two bands every two nights, since we alternate observations between the northern and southern sky).

However, many factors cause us to deviate from this ideal strategy. First, we do not want a uniform distribution of total number of visits over filters, so we replace some observations in, say, the *y* band, with more in *r* instead. Second, we observe *y* and *z* preferentially in twilight, since these bands are less sensitive to high sky brightness. Third, we avoid using *u* or *g* when the Moon is up and bright. And, finally, the filter changer can only fit five of LSST's six filters, so only five filters can be used in a single night. In each of these cases, if a filter is "not allowed" during a time when the cyclic allocation strategy

would have scheduled it, we simply use some other filter instead. Besides these intentional deviations, weather and downtime also cause us to deviate from this ideal strategy in ways that are less predictable.

Overall, ALTSched's filter allocation strategy ensures that the per-band cadence is much more regular than in MINION_1016 (as shown below), but there is likely much room for improvement by designing a strategy that more intelligently takes the four deviations from the ideal described above into account.

4. Metrics and Results

As laid out in Section 1, the science drivers of LSST motivate a wide range of technical metrics, many of which are in tension with each other. Broadly speaking, LSST's science goals fall into two categories: those that depend mainly on the final coadded images (static science), and those that depend mainly on the temporal distribution of visits throughout the survey (time-domain science). A full list of results on a variety of metrics is available at http://altsched.rothchild.me:8080. Here we highlight a small subset of these metrics that are particularly sensitive to scheduling decisions, especially those where ALTSched and MINION_1016 achieve different results.

4.1. Static Science

A number of science drivers (e.g., galaxy/star surveys, large-scale structure measurements) use coadded images, and are largely insensitive to the exact distribution of visits over time. A wide range of static science is enabled by achieving a higher final coadded depth in each band. We therefore use coadded depth as a chief metric for static science. Single-visit depths and full 10-yr coadded depths are shown for both schedulers in Table 1. Since different survey strategies may allocate visits across filters differently, we summarize the total coadded performance across bands with the *effective survey time* metric T_{eff}. Given design limiting depths M_f for a 30 s exposure in filter $f \in \mathscr{F}$, the effective survey time of a series of exposures $\mathscr{E} = (m_i, f_i)$ that achieved a 5σ limiting depth m_i in filter f_i is given by

$$T_{\mathrm{eff}} = \sum_{(m_i, f_i) \in \mathscr{E}} 30\ \mathrm{s} \times 10^{0.8(m_i - M_{f_i})}.$$

The 5σ limiting depth of an exposure is computed taking into consideration the sky brightness, seeing, and airmass, as described in Equation (6) of Ivezic et al. (2009). Design depths M_f for LSST are shown in Table 1. These design depths assume an airmass of 1, *r*-band seeing of 0″.7 (FWHM), and *r*-band sky brightness of 21 mag arcsec^{-2}. In practice, observations are taken under worse conditions, so the total T_{eff} is surprisingly low for a 10-yr survey.

Table 1
Design Specification for Single-visit 5σ Limiting Depths Under Ideal Observing Conditions and Median Single-visit and Coadded Depths Achieved by MINION_1016 and ALTSched for LSST's Six Broadband Filters

	u	g	r	i	z	y
Design (Ideal) Single-Visit Depths	23.9	25.0	24.7	24.0	23.3	22.1
Median Single-Visit Depth (MINION_1016)	23.09	24.51	24.05	23.45	22.71	21.78
Median Single-Visit Depth (ALTSched)	23.21	24.50	23.93	23.49	22.90	21.91
Median Coadded Depth (MINION_1016)	25.48	27.02	27.03	26.46	25.65	24.73
Median Coadded Depth (ALTSched)	25.61	27.03	27.04	26.35	25.93	24.31

Note. The first row shows the design specifications for single-visit 5σ limiting depths (in magnitudes) for LSST's six broadband filters under ideal observing condition, assuming an airmass of 1, r-band seeing of 0.7″ (FWHM), and r-band sky brightness of 21 mag arcsec^{-2}. Subsequent rows show the actual median single-visit depths achieved by MINION_1016 and ALTSched, and the median (over sky pixels) 10-yr coadded depths.

Two related metrics are the *average slew time* (including filter changes), and the *open-shutter fraction* (OSF), defined as

$$\mathrm{OSF} = \frac{T_{\mathrm{exp}}}{T_{\mathrm{exp}} + T_{\mathrm{slew}} + T_{\mathrm{shutter}} + T_{\mathrm{readout}}},$$

where T_{exp} is the exposure time, and T_{readout} consists of any intermediate readout time between back-to-back "snaps" during the same visit (the readout after the last snap is included in the slew time). Both MINION_1016 and ALTSched divide 30 s visits into two 15 s snaps for cosmic-ray rejection.

Maximizing T_{eff} is one motivation for ALTSched's meridian-scanning strategy, since observing fields at their minimum airmass yields the highest 5σ depth. Results for these three metrics are shown in Table 2. Despite achieving a lower OSF and higher average slew time, ALTSched reaches approximately the same T_{eff} as MINION_1016, since MINION_1016 observes off the meridian, as shown in Figure 2. In particular, in LSST's wide–fast–deep region, MINION_1016 achieves a mean (median) airmass of 1.22 (1.21) compared to ALTSched's 1.12 (1.09). MINION_1016's mean (median) normalized airmass—i.e., the airmass of an observation divided by the minimum airmass that field could have been observed at —is 1.16 (1.14) compared to 1.05 (1.01) for ALTSched. ALTSched suffers from a higher slew time for three reasons: first, because we change filters much more often than MINION_1016 in order to obtain same-night colors for nearly every visit; second, because our scanning strategy is simple and could be optimized for faster slews; and third, because our sky tiling is spaced farther apart than the tiling used in OPSIM.

Another important consideration for static science is that of survey uniformity. In particular, weak lensing and large-scale structure measurements depend sensitively on the uniformity of coadded depth across the sky (Awan et al. 2016). OPSIM uses a set of fixed field centers and attempts to mitigate the resulting imprint on the coadded depth maps by dithering around those field centers. However, every dithering strategy tried thus far still leaves a discernible imprint on the final coadded depth maps. Instead of dithering, ALTSched eliminates fixed fields entirely. We use a fixed tiling for the entire survey, but every

Table 2
T_{eff}, Open-shutter Fraction, and Average Slew Time of MINION_1016 and ALTSched

	T_{eff}	OSF	Avg. Slew
MINION_1016	333 days	0.72	7.4[a]
ALTSched	329 days	0.69	11.1

Notes. ALTSched matches MINION_1016's T_{eff} despite higher slew times because we observe on the meridian, boosting the S/N of each observation.
[a] The average slew time reported elsewhere for MINION_1016 is 6.8 s. However, the particular version of the LSST software stack used throughout this article produces a value of 7.4s, which is the value that is comparable to the 11.1s we report for ALTSched.

night, we randomly rotate this tiling by much more than a field size. Pointings not visited in a night are scheduled in a subsequent night, so gaps in the tiling pattern do not persist. To measure survey uniformity, we include angular power spectra of the number of visits to a sky pixel, shown in Figure 3. ALTSched's tiling strategy reduces power at most angular scales, including around $\ell \approx 150$, where cosmological probes are particularly sensitive.

Unlike any strategy using fixed fields with dithering, our method admits simple analysis that yields an expression for the uniformity in the number of visits to a sky pixel. For the fixed tiling used throughout the survey, let $\Omega_{0,1,2}$ be the areas of sky covered by 0, 1, and 2 pointings, respectively. Assume no sky pixels are observed more than twice, so the total area covered by the tiling $\Omega = \Omega_0 + \Omega_1 + \Omega_2$. Then the standard deviation of number of visits to a sky pixel using our strategy—i.e., after applying the randomly rotated tiling n times—is

$$\sigma = \sqrt{n\left(\frac{\Omega_1 + 4\Omega_2}{\Omega} - \left(\frac{\Omega_1 + 2\Omega_2}{\Omega}\right)^2\right)}.$$

See Appendix C for a brief derivation. Note that this expression does not depend on the tiling itself—only on the Ω_i. In order to minimize slew time while maintaining an even cadence, we choose a tiling with the fields evenly spaced and with $\Omega_2 \approx 0$.
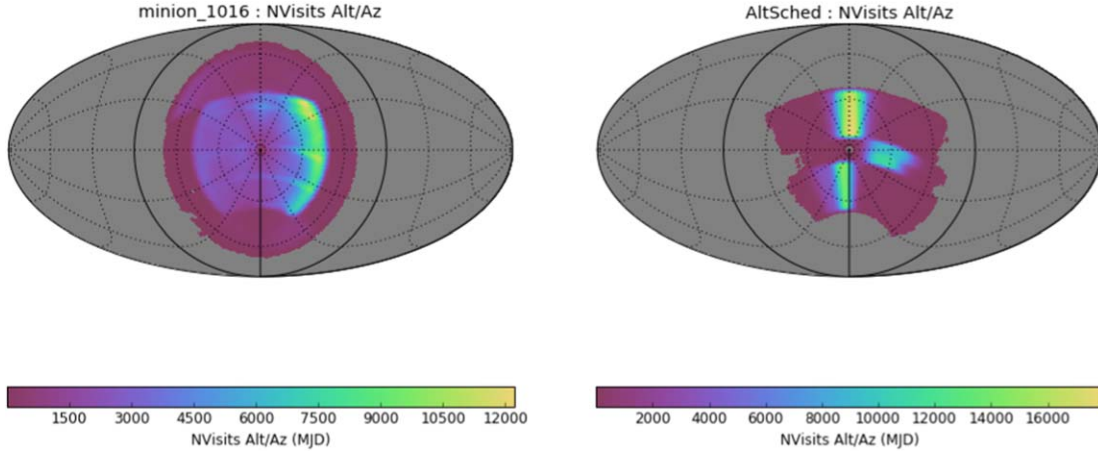
Figure 2. Number of visits as a function of alt/az (north is up, east is right, zenith is center, and the horizon is shown as a bold line). MINION_1016 exhibits an east bias, where observations are preferentially taken at high airmass in the east. ALTSched stays close to the meridian except near azimuth, where LSST's alt/az mount prevents observations directly on the meridian.

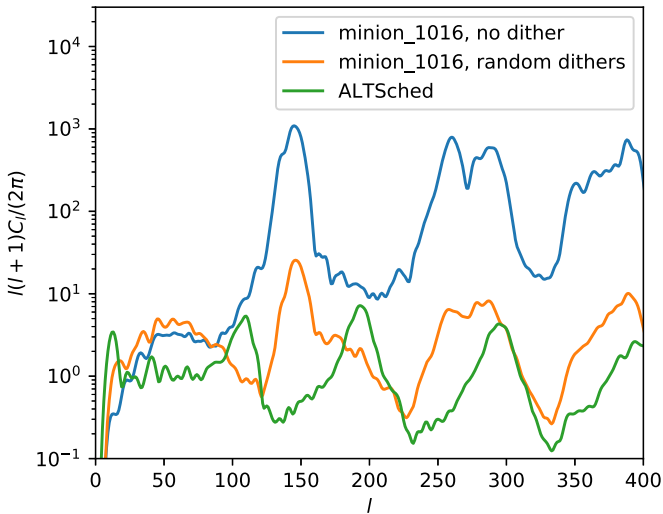(A color version of this figure is available in the online journal.)

Figure 3. Angular power spectra of the number of visits to a sky pixel for a region of LSST's wide–fast–deep region that excludes any deep drilling fields and the galactic plane. Even after applying random dithers to OPSIM's fixed fields, ALTSched increases uniformity over MINION_1016 at most angular scales.

(A color version of this figure is available in the online journal.)

Even with $\Omega_2 \approx 0$, we achieve some fast revisits due to pointings held over from previous nights, which will overlap randomly with the current night's pointings. To make the fields "evenly spaced," we draw pointings from a solution to the Thomson charge-distribution problem for $N = 3500$ (Thomson 1904). See Figure 4 for a visualization of our tiling. Adjusting N changes the density of the tiling, which controls the tradeoff between the frequency of rapid revisits and the area observed per night.
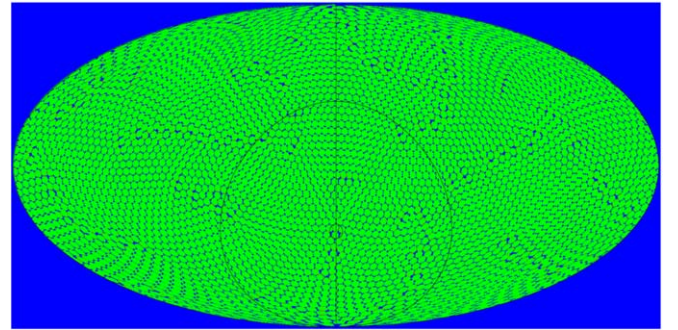

Figure 4. Tiling used by ALTSched projected using the Mollweide (equal-area) projection. Blue indicates no visits to that sky pixel, green indicates a single visit, and red indicates two visits. Note that, since the tiling is randomly rotated each night, the gaps in the tiling do not persist across nights.

(A color version of this figure is available in the online journal.)

4.2. Time-domain Science

Optimizing time-domain science is much more about managing tradeoffs between different science cases than maximizing certain quantities. We argue in this section that (1) the parameters of ALTSched directly control these tradeoffs, whereas in OPSIM, the tradeoffs are difficult or impossible to control, and (2) the ALTSched simulation analyzed in this paper enables a wider range of science cases than MINION_1016.

Most existing simulations assume that LSST will carry out pairs of visits separated by ~30–60 minutes (to link asteroid observations), and will take 30 s exposures during each visit. Some OPSIM simulations have been run to explore deviating from these assumptions, but here we take them as given. In this section, we explore how the distribution of visits over time
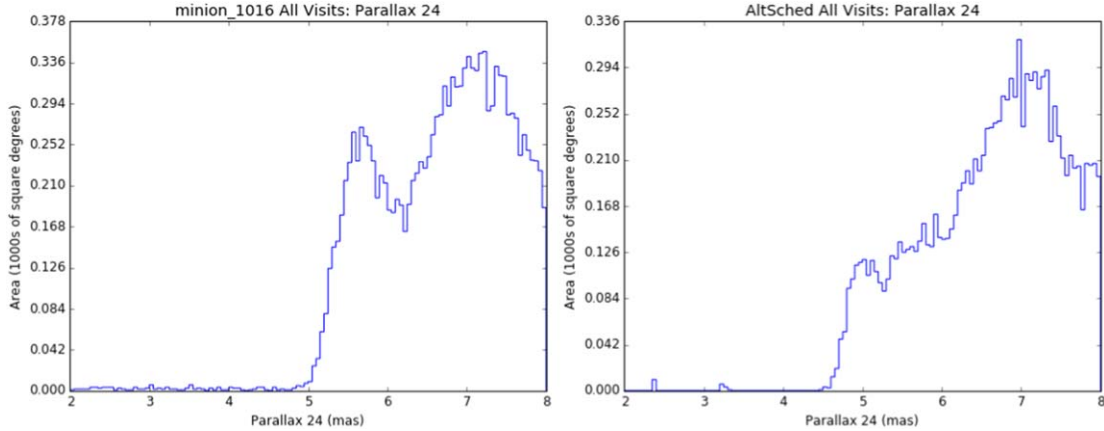
Figure 5. Parallax precision for an $r = 24$ mag star (without refraction), using either MINION_1016 (left) or ALTSched (right). Lower is better. (A color version of this figure is available in the online journal.)

affects time-domain science tradeoffs and describe how these tradeoffs can be controlled in ALTSched and OPSIM.

One fundamental tradeoff in time-domain science is between the mean season length and the mean inter-night gap (or mean cadence), which is the mean duration between consecutive visits to a sky pixel. Controlling this tradeoff with ALTSched is simple: instead of scanning along the meridian, the scheduler can start the night either east (shorter season/higher mean cadence) or west (longer season/lower mean cadence) of the meridian, and slowly move west/east over the course of the night. In contrast, there is no direct way to control this tradeoff with a greedy scheduler like OPSIM. The simplest way would be to relax the airmass/hour angle penalty and hope that the scheduler makes full use of the additional area it can use. However, this is exactly what MINION_1016 does, and instead of increasing the season length, it simply observes at a higher, but mostly fixed, airmass (see Figure 2). Alternatively, one could adjust the penalty on the hour angle throughout the night in order to persuade the scheduler to start observing in, say, the east, and then move over to the west by the end of the night. This might achieve the desired result, but would require tuning of weights just in order to reproduce the simple behavior already achieved by ALTSched.

In practice, we observe along the meridian throughout the night, since we do not see an advantage in changing the season length/mean cadence at the expense of lowering S/N by observing at higher airmass. Because MINION_1016 observes at a wider hour angle range, and therefore has a slightly longer season length, one might worry that ALTSched suffers in parallax performance. However, this is not the case, as shown in Figure 5.

Although the mean inter-night gap is largely determined by observing efficiency and the season length, as described above, the actual distribution of inter-night gaps, measured with an *inter-night gap histogram*, is much more sensitive to the scheduling algorithm. This metric, though often disregarded, has a large impact on the quality of the light curves LSST will measure. To get an intuitive sense for how the scheduling algorithm can affect the inter-night gap histogram, consider Figure 6, which shows, for a randomly chosen sky pixel, the cadences achieved by MINION_1016 and ALTSched. Notice that MINION_1016 often observes this pixel many times in quick succession, and then goes many days without any reobservations. In contrast, ALTSched observes this pixel with a much more regular cadence. This intuition is quantified with histograms of the gaps between consecutive visits to a sky pixel, shown in Figures 7 and 8.

Controlling the inter-night gap histogram with ALTSched is also simple. Changing the mean season length/mean cadence as described above yields a simple scaling of the histogram in the x-axis. And the location of the peak can be tuned either by changing the number of visits to a field per night or by employing a rolling cadence. Both options are directly controllable in ALTSched. The sharpness of the peak can also be controlled, simply by judiciously choosing which nights the telescope observes north versus south. In the default version of ALTSched, the scheduler observes north and south on alternating nights, yielding a sharp peak in the inter-night gap histogram at 2 days. However, choosing a repeating sequence such as N N S N S S N S would yield a flatter peak at 2 days, with half the histogram mass at 2 days, and the other half distributed between 1 and 3 days. In contrast, every inter-night gap histogram we have measured from an OPSIM simulation (even for simulations released after MINION_1016) has looked roughly exponential. Note that an exponential distribution of inter-night gaps is consistent with an unconstrained stochastic process about the mean. We infer from this that none of the tunable parameters in OPSIM explored so far have a discernible impact on the shape of the inter-night gap histogram.
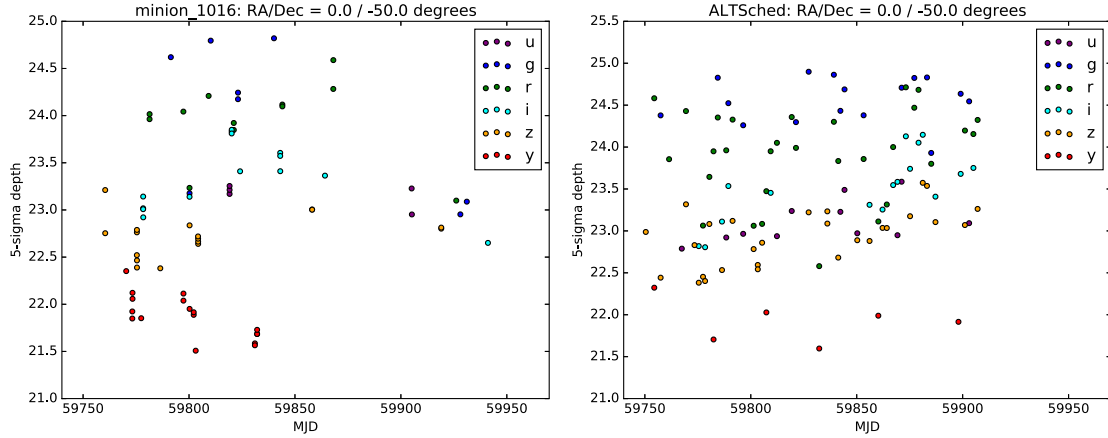
Figure 6. 5σ depth vs. time for an arbitrarily chosen R.A./decl. for MINION_1016 (left) and ALTSched (right). These plots are typical for each scheduler for LSST's wide–fast–deep region. The two plots have approximately the same number of visits (94 for MINION_1016 vs. 97 for ALTSched), but ALTSched spreads the visits more uniformly over time.
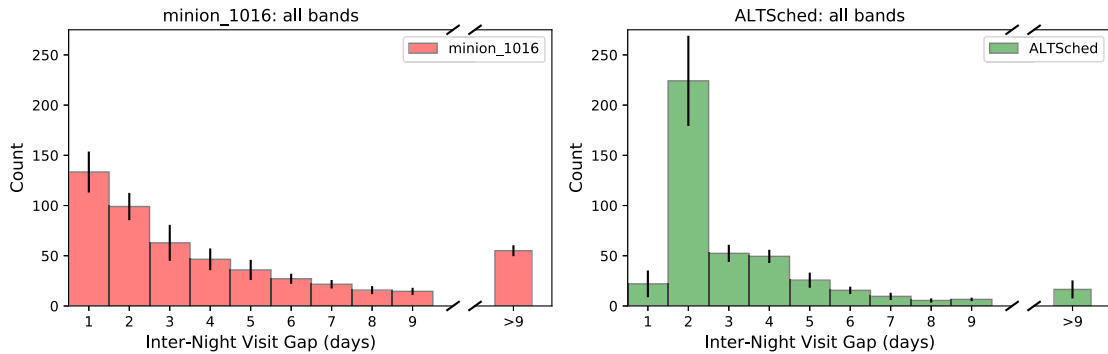(A color version of this figure is available in the online journal.)

Figure 7. Histograms of inter-night visit gaps to a sky pixel in MINION_1016 and in ALTSched, for visits in any filter. The mean inter-night gap is roughly conserved, since both schedulers have a similar total number of visits. But by alternating between observing the northern and southern skies, ALTSched suppresses 1-night revisits, which significantly reduces longer gaps. The difference between the two schedulers is even more striking than these plots indicate, since most of the remaining long gaps in ALTSched are attributable to downtime, weather, and season gaps.
(A color version of this figure is available in the online journal.)

Many science cases for LSST rely on approximately month-long transients (supernovae, kilonovae, tidal disruption events, etc.). For these science cases, we expect that multi-week-long gaps will severely reduce the quality of light curves, without a commensurate increase in quality from the higher rate of sampling over small sections of the light curve. This intuition is borne out in simulations, run by Nicolas Regnault, Philippe Gris, and the Paris Supernovae Cosmology Team, of the number of well-sampled SNe Ia different simulated schedules would obtain (personal communication). As shown in Table 3, ALTSched achieves an eightfold increase in the number of well-sampled SNe Ia, and also an increase of 0.07 in the maximum redshift at which a typical SN Ia will be well-sampled. Although these simulations are for SNe Ia in particular, we expect similar results to hold for other transients. For example,

Cowperthwaite et al. (2019) find that ALTSched achieves a nearly $2\times$ increase in the number of serendipitous kilonovae discoveries compared to MINION_1016. Similarly, Goldstein et al. (2019) find that ALTSched discovers lensed supernovae earlier than MINION_1016, enabling faster spectroscopic follow-up.

ALTSched achieves more favorable inter-night gap histograms than MINION_1016 for two main reasons. The band-agnostic gaps (Figure 7) are improved because ALTSched alternates between observing the northern and southern skies each night, suppressing 1-day revisit gaps. Because the mean cadence is approximately conserved across schedulers with similar numbers of total visits, this by necessity eliminates most of the long gaps, which are so detrimental for transient characterization. The per-band gaps (Figure 8) are improved since we carry out the two visits each
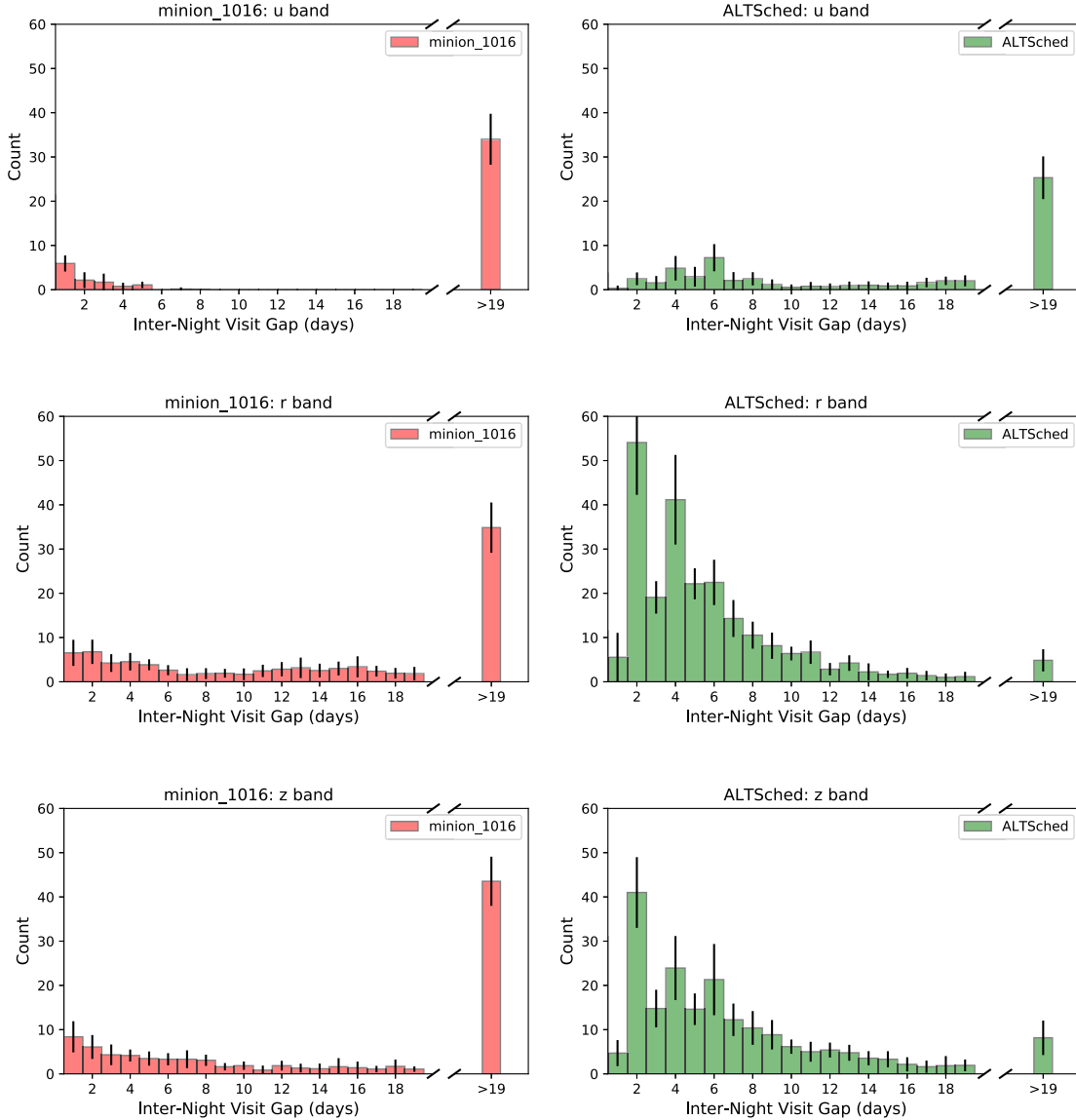
Figure 8. Per-band inter-night visit gap histograms for MINION_1016 and ALTSched, for the u, r, and z bands. Since both ALTSched and MINION_1016 defer u observations to dark time, we only see a small gain relative to MINION_1016 in the u band. Histograms for g, i, and y are similar and are deferred to Appendix B. (A color version of this figure is available in the online journal.)

sky pixel receives per night in different filters, doubling the cadence in each band (see Figure 9).

Although in this paper we advocate looking at the entire inter-night gap histogram, it is more common within LSST scheduling to present histograms of the median inter-night gap. We present these histograms in Appendix A.

So far, we have considered only the inter-night gaps between visits. However, the histogram of gaps between observations to a sky pixel within a given night is also an important metric. In order to sample transients at a large range of timescales, both ALTSched and MINION_1016 revisit each field after 30–60

minutes. In addition, both schedulers carry out some number of "rapid revisits," which are spaced less than a minute apart. MINION_1016 carries out more rapid revisits than ALTSched, and the frequency can be adjusted in either scheduler by changing the field tiling density: a tiling with more overlaps between adjacent pointings yields more ∼30 s revisits. We therefore omit further analysis of rapid revisits.

5. Schedulers Used for Other Sky Survey Projects

The scheduling principles we describe in this paper apply to any ground-based telescope able to image a large sky area per
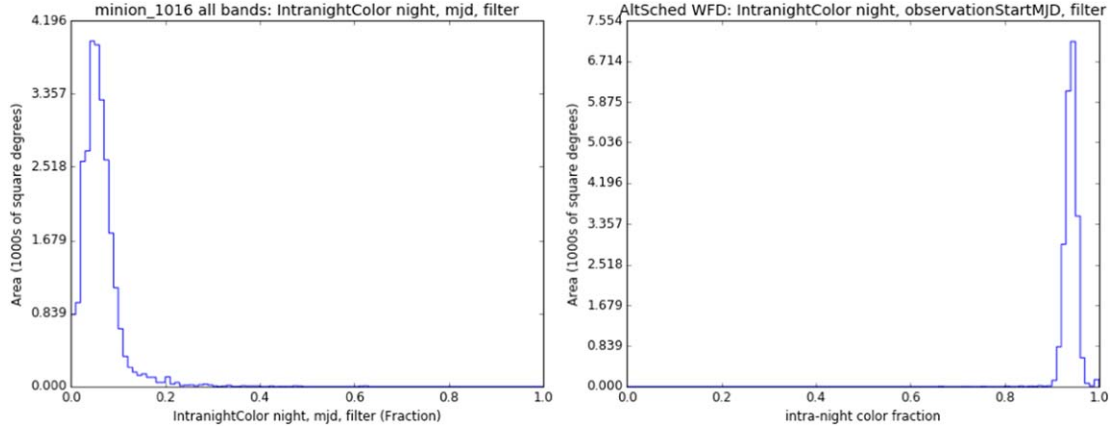
Figure 9. Histogram of the fraction of nights in which a sky pixel was visited in at least two different bands. Only nights when two or more exposures were achieved at a sky pixel are included in the calculation. By observing nearly all fields in two bands instead of only one in a given night, ALTSched doubles the observing cadence in each band.

(A color version of this figure is available in the online journal.)

Table 3
Results from Simulations Run by Nicolas Regnault, Philippe Gris, and the Paris Supernova Cosmology Team (Personal Communication)

	N_{SNe}	Avg. z_{max}
MINION_1016	47,000	0.30
ALTSched	366,000	0.37

Note. N_{SNe} is the number of well-sampled SNe Ia in the redshift-limited sample. Here, an SN is well sampled if the light curve has (considering only *griz* observations): (1) $\geqslant 1$ visit every 4 days; (2) >1 visit in $[-20, -10]$ days (rest frame); (3) >1 visit in $[+35, +45]$ days (rest frame); and (4) $\sigma_{color} < 0.04$. z_{max} is the maximum redshift at which a typical SNIa meets these cuts, and the average in "Avg. z_{max}" is taken over sky pixels.

unit time. Such telescopes include LSST (Ivezic et al. 2009), Palomar Transient Factory (PTF; Rau et al. 2009), Zwicky Transient Facility (ZTF; Smith et al. 2014), Pan-STARRS1 (Chambers et al. 2016), SkyMapper (Keller et al. 2007), the Dark Energy Survey (DES; The Dark Energy Survey Collaboration 2005), and others. Scheduling software has been developed for each of these telescopes, but often only limited information about these schedulers can be found in the literature. Broadly speaking, most schedulers are greedy algorithms like OPSIM: they choose each observation as the night progresses based on where the telescope is currently pointing and on current or predicted observing conditions. In general, greedy algorithms are guaranteed to maximize merit in the long term in only the simplest of problems, and telescope scheduling is not one of those problems. To see why, consider a scenario where every field north of the telescope has a merit score of, say, 5, and every field south of the telescope has a merit score of 4. Fields directly overhead have very low merit. If the telescope happens to be pointing in the south and long

slews are penalized, then a greedy algorithm will call for observing in the south the entire night, even though the globally optimal policy would be to slew to the north at the very beginning and observe there for the rest of the night.

The schedulers used for PTF (Law et al. 2009) and Pan-STARRS1 (Chambers & Denneau 2008) are all greedy algorithms. Similarly to LSST, these surveys include a wide-area component that receives multiple epochs in several bands, and we suspect that the challenges in time-domain science faced by greedy algorithms for LSST apply to these surveys as well. ZTF uses a different scheduling algorithm, which schedules an entire night at a time by solving an integer program designed to maximize observing efficiency, subject to constraints on when and how often each field needs to be observed (Bellm et al. 2019). Las Cumbres Observatory uses a similar algorithm (Saunders & Lampoudi 2014). For LSST, this approach is challenging for a few reasons: the overhead for each observation varies considerably (a few seconds to 2 minutes); in order to achieve a more uniform sky coverage, LSST may not use fixed fields at all, and the large number of fields and observations per night may render the integer program intractable to solve repeatedly during a night when weather changes.

6. Conclusions and Future Directions

As we have described in this paper, ALTSched gives survey designers much more control over *global* survey characteristics than a greedy algorithm. However, certain limitations caused by our algorithm's simplicity leave room for improvement in the more *local* scheduling decisions: ALTSched does not avoid taking exposures near the Moon; it does not avoid poor seeing conditions or clouds; and it makes slews that are longer than necessary. These problems are not insurmountable, and because they strictly worsen ALTSched's performance on metrics, we

view these limitations optimistically as opportunities to extract even more science performance from LSST.

Between now and the beginning of full operations, the LSST Project plans to find an optimal survey strategy by running a large number of OPSIM simulations and choosing whichever yields the best compromise between science cases. We want to stress that the best survey strategy that can be found with this procedure is likely to underperform on a number of time-domain metrics, since the parameters of OPSIM do not give much control over the survey's cadence.

Instead, we advocate for combining the advantages of ALTSched (global schedule characteristics) with those of OPSIM (sensible local scheduling decisions). We see two ways to combine forces. The simplest is to simulate surveys that switch back and forth between the two algorithms. When ALTSched tries to observe the Moon, switch to a greedy algorithm that knows not to; at the end of every month, spend a day or two using a greedy algorithm to even up coadded depth across the sky; when clouds are present, use the greedy algorithm exclusively. Strategies like these should have minimal impact on the favorable properties of ALTSched, while also largely rectifying some of its limitations. Another way forward is to combine the two algorithms directly, by using a greedy scheduler whose merit function has been specifically engineered to reproduce ALTSched for large-scale decisions (which region of the sky to observe; which filter to use), but where more local scheduling decisions can be made by lower-order terms in the merit function.

In parallel, we encourage the LSST science community to continue using ALTSched as a tool to demonstrate the minimum performance that LSST is capable of. We developed most of ALTSched over a few months, and many opportunities for improvement remain: finding a scanning pattern that minimizes slew times; making more intelligent filter choices; adapting throughout the night to avoid deviating from the meridian;

recovering more gracefully from downtime; using separate tilings per filter to increase homogeneity; etc. Making improvements like these, and combining ALTSched with OPSIM as described above, is what we believe will find the best survey strategy available in the limited time remaining before full operations commence.

Appendix A
Median Inter-night Gaps

Here we present histograms of the median inter-night gap between consecutive visits to a sky pixel. Because MINION_1016 and ALTSched have a similar total number of visits available per unit area, the mean (and therefore, roughly speaking, the median) inter-night gap should be fixed (Figure 10). We expect a twofold reduction in the per-band median inter-night gaps (Figures 11 and 12) because ALTSched observes each field in two filters per night instead of MINION_1016's single filter.

We see additional improvement beyond these predictions because MINION_1016 sometimes observes the same field more than twice per night, thus incurring additional long inter-night gaps. MINION_1016 also has a slightly longer season length since it observes at a wider range of airmasses. For the u band, the improvement is actually less than twofold because both ALTSched and MINION_1016 cluster observations in the u band around times with low lunar brightness.
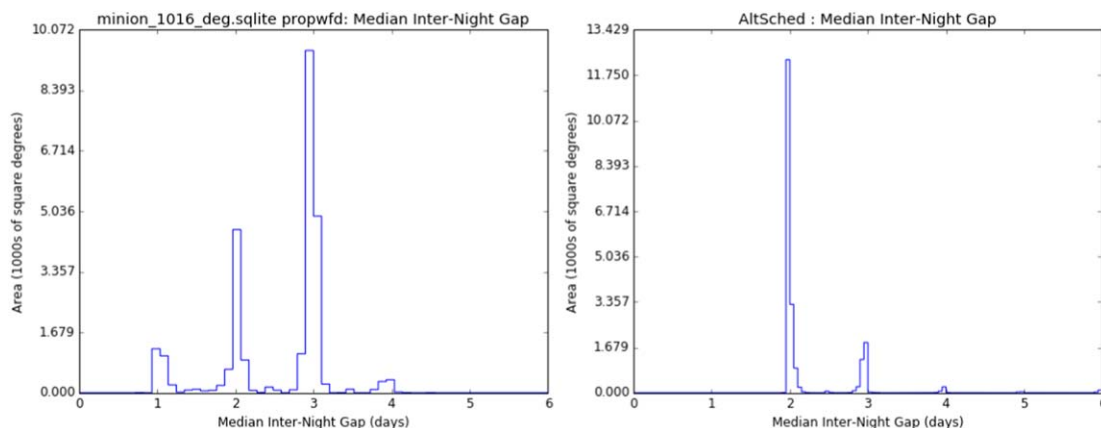

Figure 10. Histogram of median inter-night visit gaps to a sky pixel in MINION_1016 and in ALTSched, for visits in any filter. MINION_1016 has slightly longer observing gaps because it effectively has a longer season duration and because it observes some fields more than twice per night.
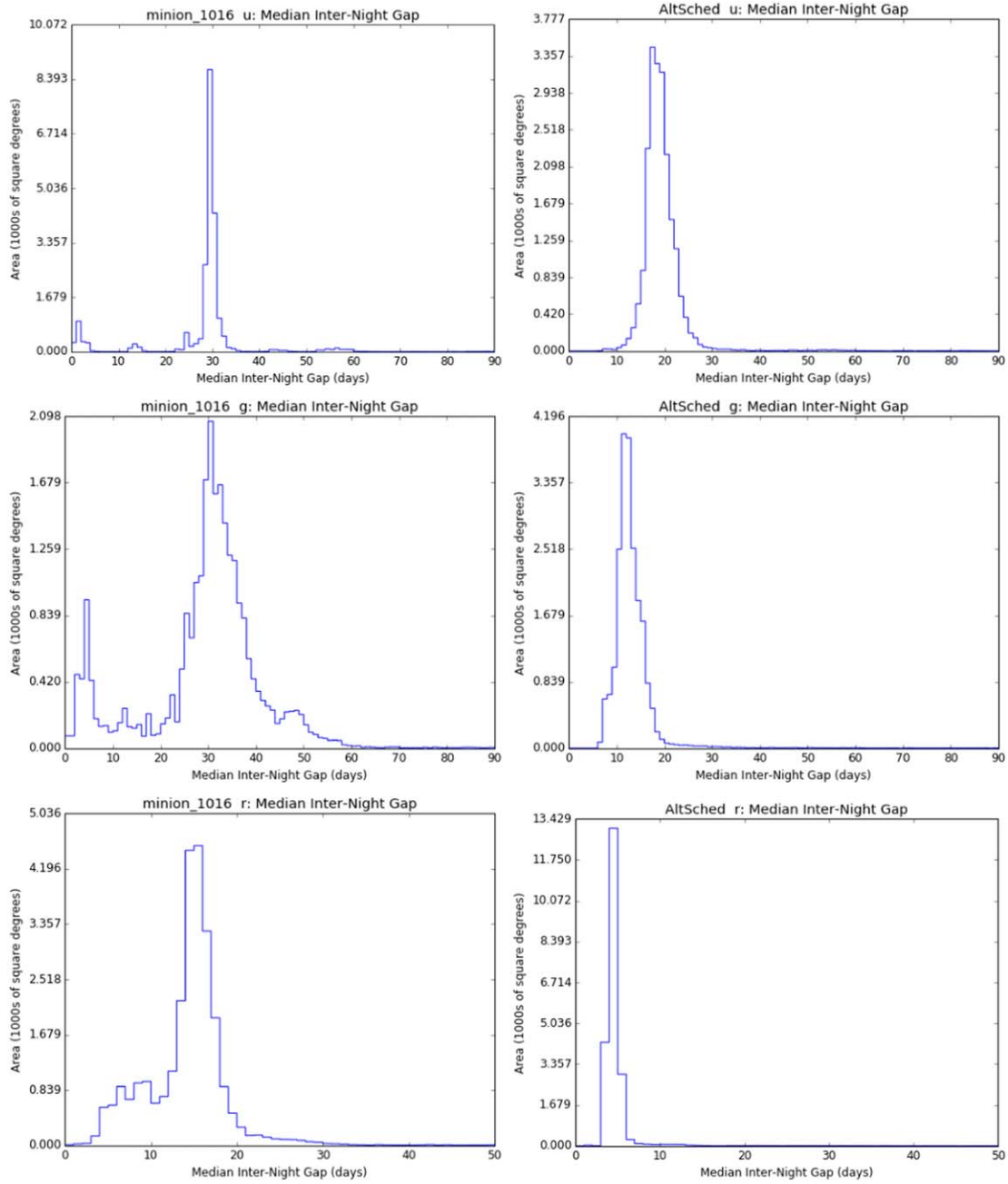(A color version of this figure is available in the online journal.)

Figure 11. Per-band median inter-night visit gaps for MINION_1016 and ALTSched, for the *u*, *g*, and *r* bands. ALTSched achieves considerably better median gaps, primarily because it executes pairs of observations taken in a single night in different filters, effectively doubling the per-band cadence in each band.
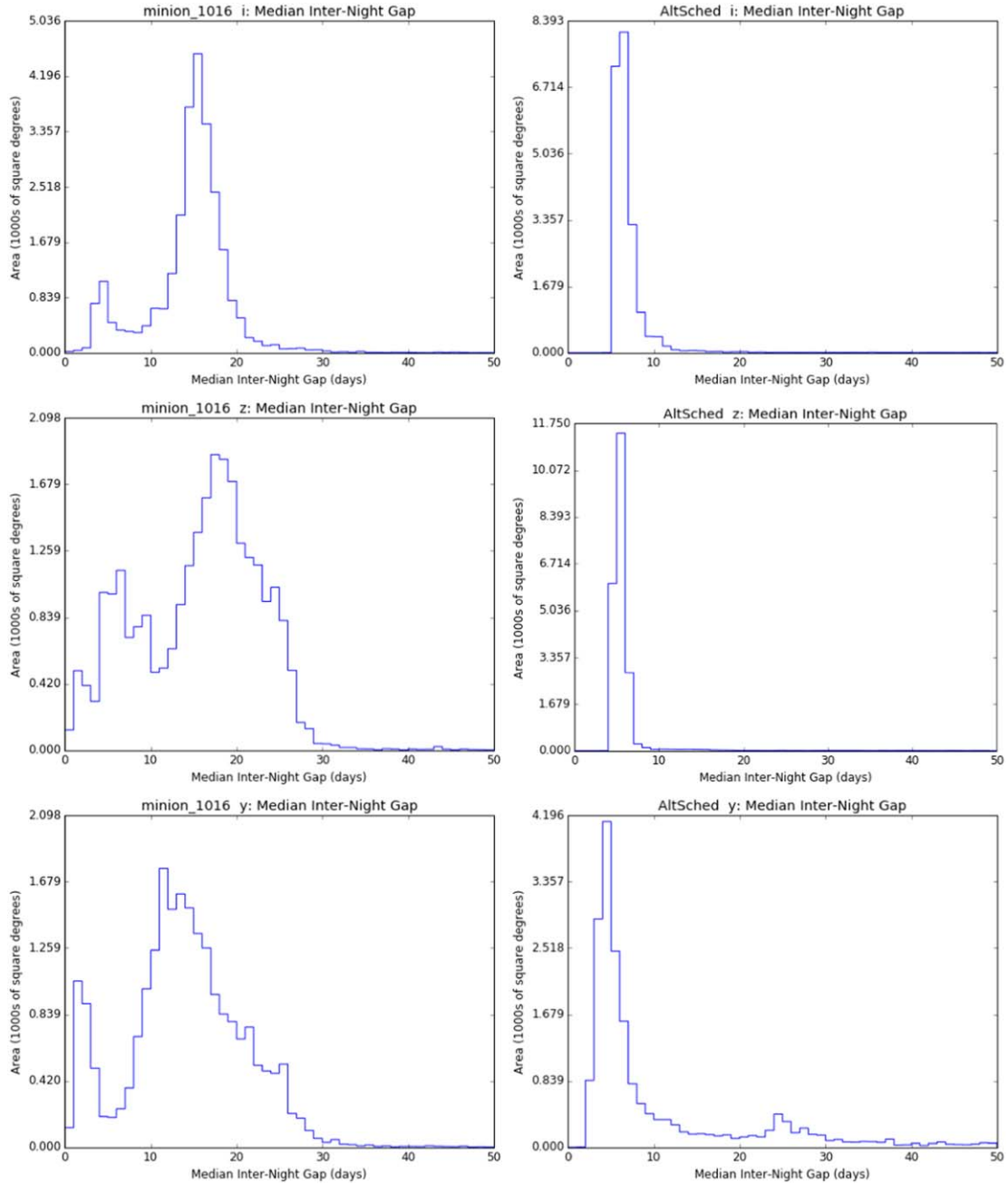(A color version of this figure is available in the online journal.)

Figure 12. Same as Figure 11, but for the *i*, *z*, and *y* bands.
(A color version of this figure is available in the online journal.)

Appendix B
Inter-night Gaps

In Figure 13, we include histograms of the typical inter-night gap between observations to a sky pixel in the *g*, *i*, and *y* bands (deferred from the text).
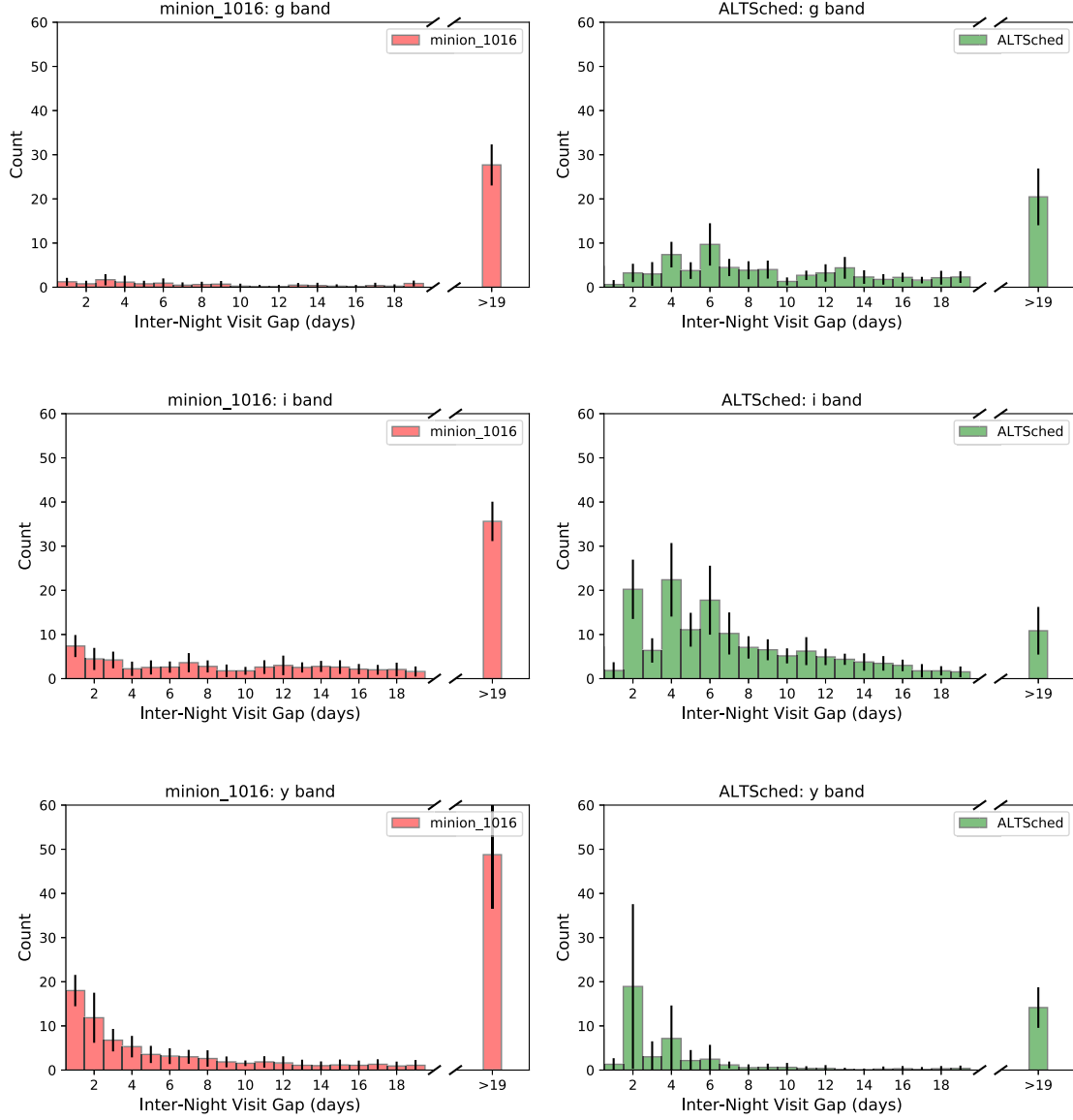
13

Figure 13. Per-band inter-night visit gap histograms for MINION_1016 and ALTSched, for the *g*, *i*, and *y* bands.
(A color version of this figure is available in the online journal.)

Appendix C
Uniformity Derivation

Consider a survey that observes only at pointings drawn from some fixed sky tiling. Let the area not covered by any pointing be Ω_0, the area covered by exactly one pointing be Ω_1, and the area covered by exactly two pointings be Ω_2. Assume no part of the sky is covered by three or more fields—i.e., that the total sky area $\Omega = \Omega_0 + \Omega_1 + \Omega_2$. If this tiling is observed once, the mean number of times a sky pixel will be observed is

$$\mu_f = \frac{\Omega_1 + 2\Omega_2}{\Omega}$$

and the rms fluctuation in number of visits to a pixel is

$$\sigma_f = \sqrt{\frac{1}{\Omega}(\Omega_0(\mu_f)^2 + \Omega_1(1 - \mu_f)^2 + \Omega_2(2 - \mu_f)^2)}$$

$$= \sqrt{\frac{\Omega_1 + 4\Omega_2}{\Omega} - \left(\frac{\Omega_1 + 2\Omega_2}{\Omega}\right)^2}.$$

If the tiling is observed *N* times without rotation or dithering, the final average and standard deviation in number of visits will be $N\mu_f$ and $N\sigma_f$.

To reduce the standard deviation, we can simply rotate the tiling by some random amount each time it is observed. If we

do this N times, then by the central limit theorem, the probability distribution of number of visits to each sky pixel is normal with mean

$$\mu_r = N\mu_f = N\left(\frac{\Omega_1 + 2\Omega_2}{\Omega}\right)$$

and standard deviation

$$\sigma_r = N\frac{\sigma_f}{\sqrt{N}} = \sigma_f\sqrt{N} = \sqrt{N\left(\frac{\Omega_1 + 4\Omega_2}{\Omega} - \left(\frac{\Omega_1 + 2\Omega_2}{\Omega}\right)^2\right)}.$$

ORCID iDs

Daniel Rothchild ⦿ https://orcid.org/0000-0002-4605-0949
Christopher Stubbs ⦿ https://orcid.org/0000-0003-0347-1724
Peter Yoachim ⦿ https://orcid.org/0000-0003-2874-6464

References

Awan, H., Gawiser, E., Kurczynski, P., et al. 2016, ApJ, 829, 50
Bellm, E., Kulkarni, S., & Graham, M. 2019, PASP, 131, 068003
Chambers, K. C., & Denneau, L. J. 2008, PS1 Design Reference Mission, Zenodo, doi:10.5281/zenodo.199860
Chambers, K. C., Magnier, E. A., Metcalfe, N., et al. 2016, arXiv:1612.05560
Cowperthwaite, P. S., Villar, V. A., Scolnic, D. M., & Berger, E. 2019, ApJ, 874, 88
Delgado, F., & Reuter, M. A. 2016, Proc. SPIE, 9910, 991013
Delgado, F., Saha, A., Chandrasekharan, S., et al. 2014, Proc. SPIE, 9150, 915015
Goldstein, D. A., Nugent, P. E., & Goobar, A. 2019, ApJS, 243, 6
Ivezic, Z., Tyson, J. A., Abel, B., et al. 2009, AAS Meeting 213, 460.03
Jones, R. L., Yoachim, P., Chandrasekharan, S., et al. 2014, Proc. SPIE, 9149, 91490B
Keller, S. C., Schmidt, B. P., Bessell, M. S., et al. 2007, PASA, 24, 1
Law, N. M., Kulkarni, S. R., Dekany, R. G., et al. 2009, PASP, 121, 1395
LSST Science Collaborations, Marshall, P., Anguita, T., et al. 2017, Science-driven Optimization of the LSST Observing Strategy, Zendo, doi:10.5281/zenodo.842713
Naghib, E., Vanderbei, R. J., & Stubbs, C. 2016, Proc. SPIE, 9910, 991011
Rau, A., Kulkarni, S. R., Law, N. M., et al. 2009, PASP, 121, 1334
Ridgway, S. T. 2015, An Optimized Cadence for LSST: The Optimum Unit Method, Tech. Rep. 17818, LSST
Robitaille, T. P., Tollerud, E. J., Greenfield, P., et al. 2013, A&A, 558, A33
Saunders, E., & Lampoudi, S. 2014, in The Third Hot-wiring the Transient Universe Workshop, ed. P. R. Wozniak et al. (Santa Fe, NM), 117
Smith, R. M., Dekany, R. G., Bebek, C., et al. 2014, Proc. SPIE, 9147, 914779
The Dark Energy Survey Collaboration 2005, arXiv:astro-ph/0510346
Thomson, J. J. 1904, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 7, 237