

PROCEEDINGS OF SPIE

[SPIDigitalLibrary.org/conference-proceedings-of-spie](https://spiedigitallibrary.org/conference-proceedings-of-spie)

ExEP yield modeling tool and validation test results

Rhonda Morgan, Michael Turmon, Christian Delacroix, Dmitry Savransky, Daniel Garrett, et al.

Rhonda Morgan, Michael Turmon, Christian Delacroix, Dmitry Savransky, Daniel Garrett, Patrick Lowrance, Xiang Cate Liu, Paul Nunez, "ExEP yield modeling tool and validation test results," Proc. SPIE 10400, Techniques and Instrumentation for Detection of Exoplanets VIII, 104001K (12 September 2017); doi: 10.1117/12.2274468

SPIE.

Event: SPIE Optical Engineering + Applications, 2017, San Diego, California, United States

ExEP Yield Modeling Tool and Validation Test Results

Rhonda Morgan^a, Michael Turmon^a, Christian Delacroix^{b,c}, Dmitry Savransky^{b,c}, Daniel Garrett^b, Patrick Lowrance^d, Xiang Cate Liu^d, and Paul Nunez^a

^aJet Propulsion Laboratory, California Institute of Technology

^bSpace Imaging and Optical Systems Laboratory, Cornell University

^cCarl Sagan Institute, Cornell University

^dIPAC, California Institute of Technology

ABSTRACT

EXOSIMS is an open-source simulation tool for parametric modeling of the detection yield and characterization of exoplanets. EXOSIMS has been adopted by the Exoplanet Exploration Programs Standards Definition and Evaluation Team (ExSDET) as a common mechanism for comparison of exoplanet mission concept studies. To ensure trustworthiness of the tool, we developed a validation test plan that leverages the Python-language unit-test framework, utilizes integration tests for selected module interactions, and performs end-to-end cross-validation with other yield tools. This paper presents the test methods and results, with the physics-based tests such as photometry and integration time calculation treated in detail and the functional tests treated summarily.

The test case utilized a 4m unobscured telescope with an idealized coronagraph and an exoplanet population from the IPAC radial velocity (RV) exoplanet catalog. The known RV planets were set at quadrature to allow deterministic validation of the calculation of physical parameters, such as working angle, photon counts and integration time. The observing keepout region was tested by generating plots and movies of the targets and the keepout zone over a year. Although the keepout integration test required the interpretation of a user, the test revealed problems in the L2 halo orbit and the parameterization of keepout applied to some solar system bodies, which the development team was able to address. The validation testing of EXOSIMS was performed iteratively with the developers of EXOSIMS and resulted in a more robust, stable, and trustworthy tool that the exoplanet community can use to simulate exoplanet direct-detection missions from probe class, to WFIRST, up to large mission concepts such as HabEx and LUVOIR.

Keywords: exoplanets, yield modeling, software testing, EXOSIMS, Design Reference Mission simulation, high contrast imaging, coronagraph, starshade

1. INTRODUCTION

EXOSIMS¹ is a modular, open-source² simulation tool for modeling of the exoplanet space missions, including detections, characterizations, and expected yield. EXOSIMS has been adopted by the Exoplanet Exploration Programs Standard Definition and Evaluation Team (ExSDET) as a common mechanism for comparison of various exoplanet mission concept studies, with yield results reported for Decadal Study concepts³ and WFIRST.^{4,5}

To build confidence in the quantitative predictions of the tool, we developed a three-level validation test plan that leverages the Python-language unit-test framework at the basic functional level, utilizes integration tests for selected module interactions, and performs end-to-end cross-validation with other yield tools. This paper presents the test methodology and results, with the physics-based tests such as photon counts and integration times treated in detail and the functional tests treated summarily. Section 2 discusses the principles and operation of EXOSIMS. Section 3 discusses unit testing of particular functional units. Section 4 discusses integration tests in which multiple EXOSIMS functional units operate. Section 5 discusses end-to-end testing and section 6 concludes the paper.

Correspondence to R.M.: Rhonda.Morgan@jpl.nasa.gov, or by post, 4800 Oak Grove Drive, Pasadena, CA, 91109.

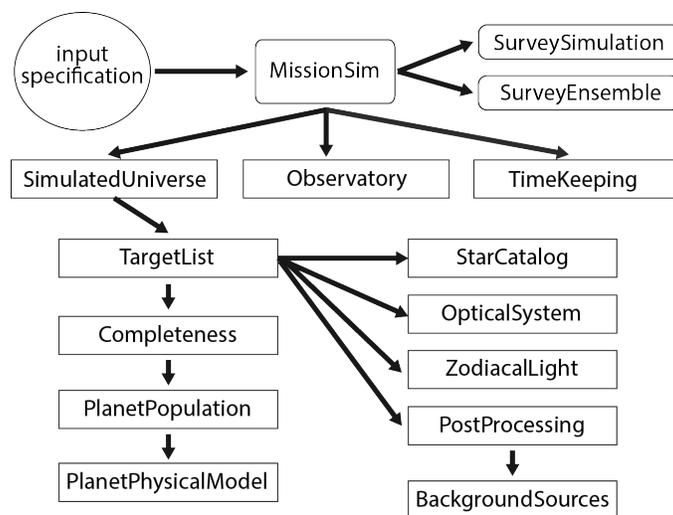


Figure 1. EXOSIMS simulation hierarchy. Reading an input specification file sets up an object hierarchy covering all aspects of a time-stepped mission simulation. Different specifications may result in somewhat different object hierarchies.

2. SCOPE AND OPERATION OF EXOSIMS

EXOSIMS defines key parameters in a script file and uses them to instantiate a mission simulation object (see Figure 1⁶), which contains representations of a simulated universe of host stars and planets, an observatory and its orbital position, an optical system including detectors and coronagraphs/starshades, and parameterized background sources such as zodiacal light. A mission is simulated by iterating through a time-stepped series of observations, each of which may result in missed detections, false alarms, correct detections, or spectral characterization. The end result is a list of such observations, which can be extracted and processed to determine yield for that particular mission simulation. Building atop a single mission simulation, an ensemble of independently-drawn Monte Carlo simulations can be used to determine expected yields, averaging over realizations of planet occurrences in the simulated universe, as well as detection statistics.

Default choices are offered for the fundamental, required objects shown in Figure 1. EXOSIMS refers to this tree of objects as the simulation “backbone,” and we will find it convenient to abbreviate any such object as an EBO (EXOSIMS backbone object) to distinguish it from a generic object within the larger object-oriented programming (OOP) system. In addition to the default (or *prototype*) EBOs, any EBO can be overridden with a specialized EBO that implements the correct interface, in typical OOP fashion. So, for example, the default (or *prototype*) `Observatory` EBO can be replaced with a *specialized* `WFIRSTObservatoryL2` EBO which is aware of the geometrical constraints of observing from an L2 orbit. This provides significant flexibility, particularly regarding observatories, planet populations, and scheduling of observations. The JSON-encoded script file (<http://json.org>) is the primary means of control over the characteristics of the simulation. It allows arbitrary configuration of numerical parameters such as planet occurrence rates and detector sensitivity. This file also selects which specific EBOs are to be used in the simulation.

EXOSIMS is a modest-size object-oriented package consisting (as of June 2017) of about 3850 source lines of code (SLOC) in the Python language. The SLOC metric excludes comment lines and the conventional triple-quoted Python documentation strings. As noted above, the code has an imposed object-oriented structure which naturally facilitates unit testing of each EBO, both prototypes and specialized. These unit tests are the foundation of the test and validation scheme we developed, and they are addressed in the next section. The distinctive feature of EXOSIMS is the diversity and complexity of the real-world entities that it models, notably:

Universe An observable universe consisting of star systems, with all stars in known positions and with their planets in given orbits and with specified albedo and mass/radius relationship.

Table 1. Snapshot of EXOSIMS unit test coverage relative to the June 2016 version.

Module	Statements	Missing	Coverage
MissionSim.py	124	15	88%
BackgroundSources.py	20	10	50%
Completeness.py	20	4	80%
BrownCompleteness.py	182	71	61%
Observatory.py	254	13	95%
WFIRSTObservatory.py	44	0	100%
OpticalSystem.py	203	12	94%
KasdinBraems.py	32	1	97%
Nemati.py	46	36	22%
WFIRSTOpticalSystem.py	26	0	100%
PlanetPhysicalModel.py	26	0	100%
FortneyMarleyCahoyMix1.py	92	3	97%
PlanetPopulation.py	87	3	97%
EarthTwinHabZone1.py	18	0	100%
EarthTwinHabZone2.py	14	0	100%
KeplerLike1.py	60	0	100%
KnownRVPlanets.py	80	1	99%
PostProcessing.py	32	3	91%
SimulatedUniverse.py	102	22	78%
KeplerLikeUniverse.py	27	0	100%
KnownRVPlanetsUniverse.py	51	0	100%
StarCatalog.py	32	3	91%
EXOCAT1.py	40	1	98%
SurveyEnsemble.py	5	0	100%
SurveySimulation.py	327	240	27%
TargetList.py	157	24	85%
KnownRVPlanetsTargetList.py	45	0	100%
TimeKeeping.py	59	0	100%
ZodiacalLight.py	42	13	69%
Stark.py	46	37	20%
util/deltaMag.py	4	0	100%
eccanom.py	24	1	96%
get_module.py	92	9	90%
keplerSTM.py	84	4	95%
statsFun.py	38	8	79%
Cumulative	2538	534	79%

Keypoint A solar system with bright bodies in position according to known ephemeris, an observatory in a given orbit, and keepout regions determined according to the resulting geometry constraints at the simulation's current observation time.

Photometry A telescope with a starlight suppression system (coronagraph or starshade) and an optical path with detector noise model having a given sensitivity and resolution, for both detection and spectral characterization.

We devoted special attention to developing integration tests for these three particular functional areas. They involve cross-function and cross-EBO interactions, so they are not appropriate for unit tests. The integration tests for these three functional areas are discussed in section 4.

3. UNIT TESTING

As noted, the EXOSIMS code consists of 3850 SLOC spread over 208 functions. We have written unit tests for 79% of the code, measured by line count, referenced to the version of June 2016. We have released these unit tests as part of the standard source code distribution. The module-by-module coverage is detailed in Table 1.

In this table, the backbone objects (EBOs) are shown un-indented, and specialized EBOs are indented beneath their base class. Several new specialized EBOs have been developed since these unit tests were written, and this new code has little test coverage.

Unit tests can employ several strategies, depending on the structure of the code under test. For instance, the known radial-velocity planets universe (`KnownRVPlanets`) module, as implemented in EXOSIMS, is largely driven by a stored tabulation of radial velocity (RV) planets from the IPAC catalog.⁷ The companion unit test checks exhaustively against an independently-downloaded catalog of the same RV stars and planets (584 planets and about 35 attributes, including semi-major axis, mass, and eccentricity). The Stark zodiacal light module (`Stark.py`) implements the Leinert *et al.*⁸ zodiacal light model from Stark *et al.*⁹ The corresponding unit test checks against an independent tabulation the team created that covers 40 sources at two wavelengths. Similarly, the stellar proper motion function (in `Observatory`) is point-checked against an independent calculation.

Tests regarding exoplanet populations (`PlanetPopulation`) and physical properties (`PlanetPhysicalModel`) use extensive numerical checks versus tabulated data. For instance, the `FortneyMarleyCahoyMix1` planet physical model implements relationships among planet mass, size, and albedo that were described in several related papers. The unit tests for this module check against full tabulations of planet properties extracted from the relevant papers. Geometric albedos of model planets, as a function of semi-major axis and metallicity, as computed by EXOSIMS are exhaustively tested against values independently extracted from tables in the digital version of the companion article.¹⁰ The EXOSIMS radius-mass (density) relationships for giant planets are tested by comparison to values independently extracted from the digital version of Table 4 of the relevant article.¹¹ The EXOSIMS radius-mass relationships for smaller planets are tested by comparing against a re-implementation of the published relationships.^{11,12}

In all the above cases, the test takes the form of either exhaustive or point checks of computed results from EXOSIMS against tabulated data. In other cases, numerical checks against another implementation are used, which allows probes of code functionality for many values without the need for an offline tabulation. This type of test works well for general-purpose subroutines. For instance, EXOSIMS determines solar system geometry, which influences keepout regions, using one of two methods: propagation of the solar system's Keplerian parameters, or the forward-propagated positions encoded in a JPL SPICE kernel.¹³ The corresponding unit tests use a separate implementation that accesses the SPICE ephemeris and validates predicted positions at a selection of times. On the other hand, determination of the observing geometry for exosystems by EXOSIMS uses propagation of the Keplerian parameters relevant to that system (`keplerSTM.py`). The corresponding unit test uses a separate implementation of a nine-body solar system with diverse parameters (eccentricity, mass, orbital inclination, etc.).

As a final example of detailed probes as a testing tool, EXOSIMS pervasively uses the notion of an *interpolant*, a 1-d or 2-d interpolating function which may be defined by tie points read from a file. Interpolants are used to define many key optical system parameters such as wavelength and working-angle dependent contrast and throughput. (See Subsection 4.3 for examples.) The optical system unit tests exercise this interpolant functionality by probing the given functions at a series of values, as well as by supplying erroneous tables. Similar exhaustive probes are used for other geometric, statistical, and numerical utilities in EXOSIMS.

In other cases, non-quantitative functionality checks are used where no numerical test is applicable. For instance, object constructors in the EXOSIMS prototype classes (base classes) must set all properties (as many as 30). In general, the unit tests for prototype classes such as `PlanetPhysicalModel`, `PlanetPopulation`, and `Observatory` verify that the object type, units, size, and range matches constraints. Some planet populations and parameters are generated by random sampling, and in this case random number distribution checks are used to verify the correct distributional form, even though strict numerical checks are inapplicable.

In general, we found that completing these unit tests:

- Identified places where code should be divided into smaller functional units so that unit testing was possible. The resulting code factorization improved code quality in many places.
- Identified places where interfaces and logic should be simplified, typically to reduce cyclomatic complexity in the form of nested if/else constructs, to make it possible to test all execution paths.

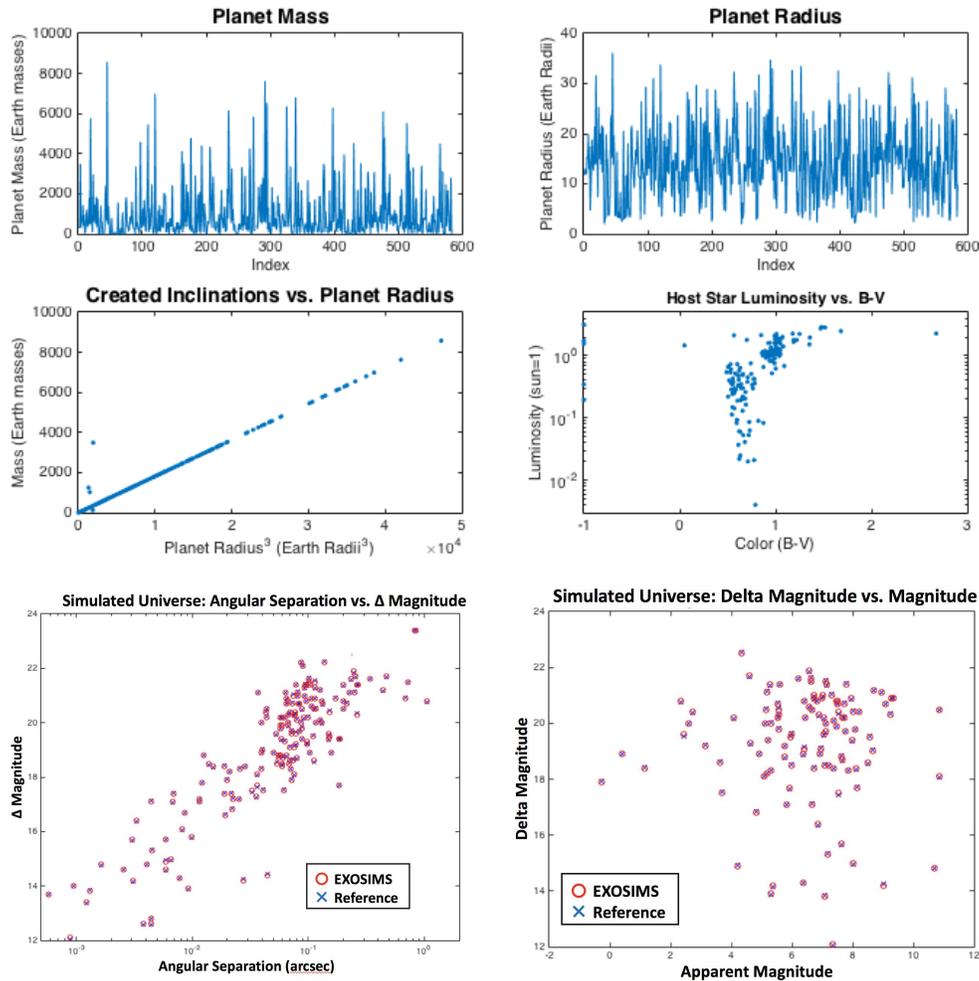


Figure 2. Planet properties within the simulated universe.

- Introduced more automation into file- and table-driven data sources within EXOSIMS, because test development required automation of table generation.
- Provided a stimulus to establish end-to-end code repeatability, by standardizing on a single random number generator and setting the random number seed in a repeatable way.

Familiar technical obstacles involved in unit testing of OOP codes were overcome. Many methods under test, such as those in the simulated universe and in the optical system, hold considerable state, which must be set up consistently in order to perform a test. It was typically possible to instantiate these objects with a subset of the overall system initialization script before performing the test. Several components use randomization (calls to a system-provided random number generator) inside their methods, and in these cases the randomization was typically circumvented by patching in order to perform a repeatable test. Mock objects were created to provide needed interfaces to objects under test.

4. INTEGRATION TESTING

We describe test results for the EXOSIMS simulated universe, keepout, and photometry.

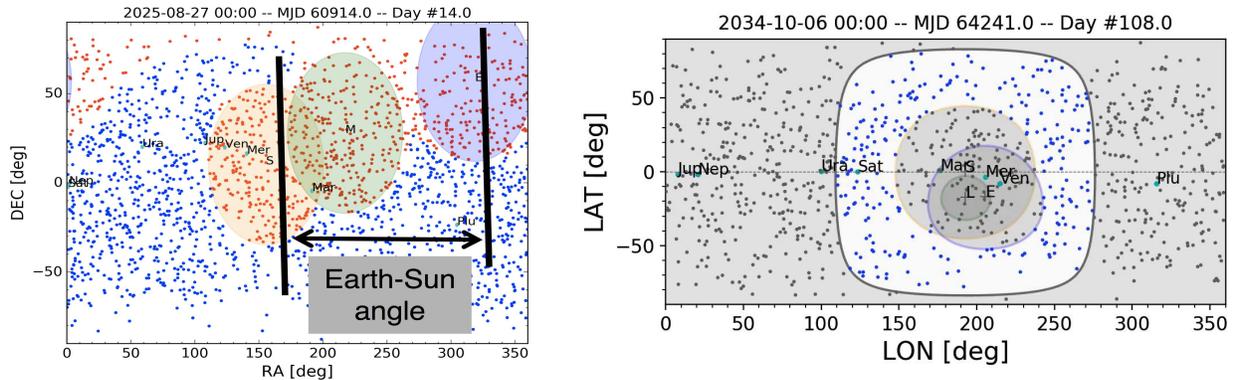


Figure 3. Target keepout regions for an observatory in a WFIRST-type L2 halo orbit. Both panels show color-coded host system targets (blue: visible; red/gray: kept out) in sky coordinates at one particular simulation epoch. Left panel: diagnostic plot from an early `Observatory` module with excessive Earth-Sun separation due to an incorrect observatory orbit. (Circular keepout regions on this plot are approximations.) Right panel: keepout after validation of the observatory geometry, in heliocentric longitude/latitude coordinates. Potential glint from a starshade keeps out the vast majority of targets (outer gray area), and the sun, earth, and lunar avoidance (gray overlays) block other targets. The keepout regions in the right-hand plot are not circular due to the map projection.

Video 1: The two-year keepout movie corresponding to the right-hand panel: <http://dx.doi.org/DOI10.1117/12.2274468>

4.1 Simulated Universe

We ensured that stellar and planetary parameters are correctly handled by EXOSIMS. We used the known RV planets universe, described above, for these tests. After filtering against reasonable target criteria for planet delta-magnitude and working angle, the original population of 584 planets was reduced to about 178 planets spread over 114 star systems. These planets span a wide range of mass/radius, semi-major axis, and luminosity. See Figure 2. We checked that these attributes are correctly computed by EXOSIMS, and checked key derived quantities such as working angle and delta magnitude. In turn, these elementary physical properties formed the basis of the photometric tests described below.

4.2 Keepout

The keepout set is those targets that are not observable at the current simulation epoch due to geometric constraints on telescope pointing. In general, keepout is a function of the target list and the observatory position in relation to bright objects like the Earth and Sun. For telescopes at L2, the observing geometry is strongly dependent on the halo orbit used. Telescopes with starshades require a maximum sun angle constraint to avoid glint off the occulter. The upshot of these component interactions is that keepout determination must be tested using the `MissionSimulation` object (the highest level of Figure 1), which encompasses the simulated universe and the target list (`SimulatedUniverse`, `TargetList`), as well as the observatory and its orbit (`Observatory`), and the current simulation epoch (`TimeKeeping`). The object and its children are instantiated with an appropriate script specifying an observatory and a target list, and the simulation can then be directed to any time desired, and the keepout computed. The most effective test proved to make a movie of kept-out versus visible targets in a map format across heliocentric latitude and longitude, as well as per-target summary statistics such as cumulative visibility.

This integration test proved to be a powerful diagnostic, identifying several issues related to observation geometry. An example is shown in Figure 3. The left panel contains a sky coordinate map of one particular simulation epoch, corresponding to one frame of the corresponding keepout movie. The target stars are shown as blue (observable) and red (kept out) points in RA/DEC coordinates. Solar system bodies are shown as text tags, and the keepout region due to each is shown schematically as a shaded circle in the map plane. (The circle indicates region scale, but causes distortions, because the true map-plane regions are non-circular projections of a given solid angle onto the map. This is a limitation of the plot, not EXOSIMS.) In this plot, the Earth and Sun appear nearly in opposition relative to the observatory, indicating a bug which turned out to be due to improper positioning of the L2 halo orbit. The keepout region was tested by generating a movie of the targets

Table 2. Mission and Telescope Parameters for Photometric Integration Testing

Astrophysical	Zodiacal surface brightness	23.34 mag/arcsec ²
	Exozodiacal surface brightness	23.34 mag/arcsec ²
	Albedo	0.5
Mission	$F_0(\lambda = 550 \text{ nm})$	1e4 photon/m ² /nm/s
	Telescope diameter	4.0 m
	Obscuration factor	0
	Central wavelength	550 nm
	IWA@550 nm	70.9 mas
	OWA@ 550 nm	7.094 arcsec
	$\Delta\text{mag floor}$	26
Telescope	Postprocessing, f_{pp}	0.1
	Throughput	0.6333
	Ω	$\pi (1/\sqrt{2} \cdot \lambda/D)^2$
	Quantum efficiency	0.91
	Dark current	5e-4 e ⁻ /s
	Clock-induced charge	3e-3 e ⁻ /s
	Read noise	3 e ⁻ /frame
	Frame rate	0.001 Hz
	Excess Noise Factor (ENF)	1
	Detection	SNR
Bandwidth		20%
Coronagraph throughput		0.3
Coronagraph Contrast		1e-10
Characterization	SNR	10
	Spectral resolution R_s	70
	Coronagraph throughput	0.4
	Coronagraph Contrast	1e-10

in the keepout zone over a year. Though this integration test for keepout requires the interpretation of a user, it proved to give great insight to observing geometry and the overlap of the various keepout regions.

A frame from the finished keepout movie for a starshade scenario is shown in the right frame of Figure 3. The circular keepout regions of solar system bodies now appear in addition to a larger keepout region (shown in a tan color) due to the need to avoid solar glint from the starshade. Again, observable stars are in blue. This plot offers a way to verify that keepout (in this scenario, different for Earth, Moon, Sun, and other planets) is computed correctly, and that observing geometry is correct.

4.3 Photometry

We thoroughly tested the photometric calculations performed by EXOSIMS. Referring to Figure 1, the key components are the optical system (`OpticalSystem`), the targets (`TargetList`, `StarCatalog`), and interference calculated by `ZodiacalLight`. Observing geometry is also important due to its effect on working angle (WA). To encompass these component interactions, integration tests for photometry are done from the `SimulatedUniverse` object level.

This test case utilized an IPAC-derived radial velocity (RV) exoplanet catalog containing 177 exoplanets that satisfied magnitude filters. These known RV planets were set at quadrature to allow for deterministic validation of the calculation of physical parameters, such as working angle, photon counts and integration time. In the end, 84 exoplanets remained within the acceptable working angle range of the 4m unobscured telescope, $0.0709 \text{ arcsec} \leq WA \leq 0.7094 \text{ arcsec}$. We tested photometric parameters for these 84 exoplanets in each of three *observing modes*: detection at 550nm and 750nm, and spectral characterization at 750nm. Many photometric characteristics are wavelength-dependent, so use of two different wavelengths allows testing of these behaviors.

Spectral characterization uses a different $\Delta\lambda$, which is derived from the specified spectral resolution. As such, it was important to test both detection and characterization observing modes.

Photometric calculations follow the approach of Nemati¹⁴ as parameterized by Delacroix et al.⁵ Important parameters are in Table 2. The key parameter of integration time is defined with reference to an interplay between wavelength-dependent planet and background count rates, $C_p(\lambda)$ and $C_b(\lambda)$. Here

$$C_p(\lambda) = f_{\text{core}}(\lambda) C_{F0}(\lambda) 10^{-0.4(m_V + \Delta mag)} \quad (1)$$

where f_{core} is the fraction of light contained within the photometric aperture of the PSF (usually the FWHM), and with a spectral flux density electron count rate given by

$$C_{F0}(\lambda) = \epsilon(\lambda) A \Delta\lambda T(\lambda) F_0(\lambda) \quad (2)$$

where ϵ is quantum efficiency, A is pupil area, $\Delta\lambda$ is the bandwidth, $T(\lambda)$ is the optical system throughput, and $F_0(\lambda)$ is the spectral flux density function, given empirically¹⁵ by $\log_{10} F_0(\lambda) = 4.01 - (\lambda - 550)/770$ for wavelength in nm. The competing background noise for detection is

$$C_b(\lambda) = \text{ENF}^2 [C_{sr}(\lambda) + C_z(\lambda) + C_{dc} + C_{cc}] + C_{rn} \quad (3)$$

where the excess noise factor (ENF) is 1 for CCD or $\sqrt{2}$ for EM-CCD, suppressed starlight residuals are C_{sr} , zodiacal light $C_z(\lambda)$ is a sum of a local- and an exo-zodiacal term, and C_{dc} , C_{cc} , C_{rn} are dark current, clock-induced charge, and readout noise count rates. For spectral characterization, the photon noise of the planet is included in the background noise:

$$C_b(\lambda) = \text{ENF}^2 [C_p(\lambda) + C_{sr}(\lambda) + C_z(\lambda) + C_{dc} + C_{cc}] + C_{rn} \quad (4)$$

The count rate due to residual stellar speckle is

$$C_{sr}(\lambda) = I \frac{\Omega}{\theta^2} C_{F0}(\lambda) 10^{-0.4m_V} \quad (5)$$

where I is the normalized stellar PSF intensity per pixel, Ω is the solid angle of the photometric aperture, and θ is the pixel scale. Finally, the composite zodiacal light rate is

$$C_z(\lambda) = C_{F0}(\lambda) \Omega \tau_{\text{sky}} (f_z + f_{ez}) \quad (6)$$

where f_z is the solar system zodiacal flux, and f_{ez} is the flux from the zodiacal light of the host system, and τ_{sky} is the sky transmission through the coronagraph. Given these rates, the integration time to achieve a desired signal-to-noise ratio (SNR) is

$$t_{\text{int}}(\lambda) = \frac{C_b(\lambda) \text{SNR}^2}{C_p(\lambda)^2 - (\text{SNR} f_{pp} C_{sr}(\lambda))^2} \quad (7)$$

where f_{pp} is the noise reduction due to postprocessing. Integration time for characterization is determined the same way, with a narrower $\Delta\lambda$ in eq. (2), calculated as $\Delta\lambda = \lambda/R_s$ where R_s is spectral resolution.

Exhaustive tests of the EXOSIMS calculations of these quantities were done against a reference implementation in Matlab. Selected results are plotted in Figure 4. Due to source magnitude differences, $C_p(\lambda)$ varies over four orders of magnitude, and agreement is obtained across the range. $C_b(\lambda)$ has an effective floor due to zodi and detector noise, and agreement is obtained across a range of two orders of magnitude. Agreement of integration times holds to within a small fraction of one percent. Similar agreement was obtained for all rates described above. Specific numerical results for the RV planets examined by Traub¹⁵ are shown as a reference for each of the three observing modes in Tables 5, 6, and 7, and all of the above-described rates.

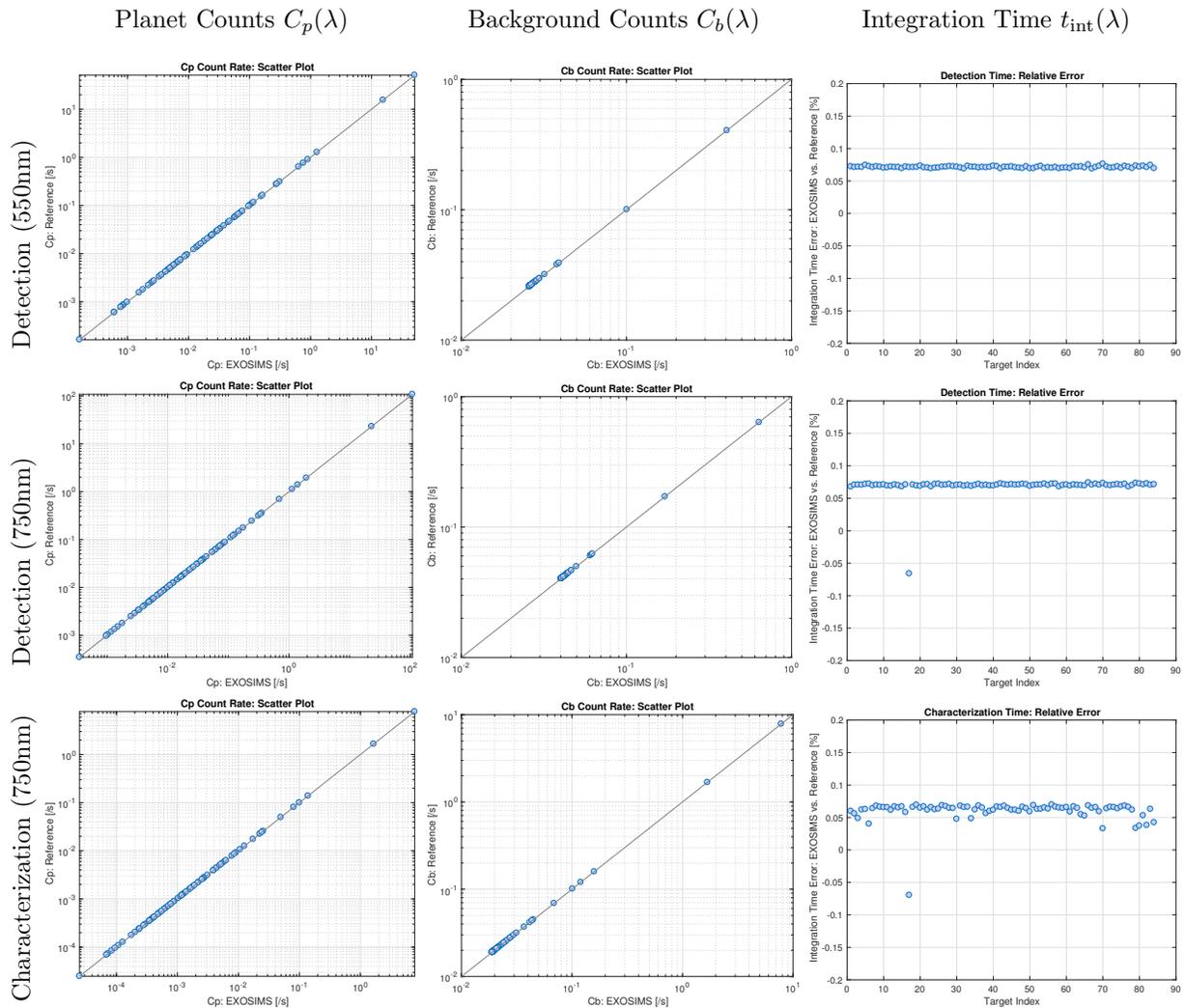


Figure 4. Photometric agreement over three scenarios: detection at 550nm (top), at 750nm (middle), and spectral characterization at 750nm (bottom). Relative errors are expressed in percent.

Table 3. End-to-End Cross Validation of ExoEarth Detection Yield.

Scenario	AYO	EXOSIMS
4m, $2\lambda/D$	6.6	5.4 ± 1.5
4m, $2.5\lambda/D$	5.6	3.5 ± 0.7
6m, $2\lambda/D$	14.0	16.4 ± 1.4
6m, $2.5\lambda/D$	12.2	9.7 ± 1.0
8m, $2\lambda/D$	23.2	33.7 ± 1.9
8m, $2.5\lambda/D$	20.3	25.3 ± 1.8

5. END-TO-END SIMULATIONS

End-to-end simulations provided the highest-level functional test for the code, producing planet detection rates which were evaluated on their own as well as compared to other yield modeling codes. The end-to-end functional test with synthetic Earths was an important augmentation to the known RV planet testing, because the synthetic Earth path utilizes different specialized EBOs (`KeplerLikeUniverse`, `EarthTwinHabZone1`, and `BrownCompleteness`) than does the known RV planets path. In this manner, both main backbone paths of the code were exercised: generation of a pseudo-random population containing synthetic Earths, and adoption of RV planets from a catalog.

The end-to-end simulations were tested for self-consistency. A Monte Carlo ensemble of 1500 Design Reference Missions (DRMs, that is, a single simulated mission's record of time-tagged observations, false alarms, and detections) was run for an 8m unobscured aperture. Binomial detection statistics were computed to determine the $1\text{-}\sigma$ error bars on the histogram of unique planet detections (top panel, Figure 5). The error bars indicate that the apparent double peak has low significance. To establish variability of yield with growing ensemble size, the simulation ensembles were evaluated in groups, incremented by 100, up to the full set of 1500 DRMs. The mean and standard deviation of the groups of ensembles were seen to stabilize at about 800 DRMs. The ensembles were further subdivided into the first 700 vs. last 700 (A vs. B), which showed stabilization was reached more quickly by the second set. Similar tests were run for 4m and 6m diameter apertures with the result that the number of DRMs required is approximately 400 and 600 respectively. The number of required DRMs is driven by the number of stars in the target list, which in these tests was filtered by what would be detectable for 4, 6, and 8 m apertures.

Cross-validation was done against related results from the Altruistic Yield Optimization (AYO) algorithm,¹⁶ using unpublished results under similar conditions that were kindly communicated by the AYO authors.¹⁷ Starting from these matching astrophysical and instrument parameters (as in Stark *et al.* 2015¹⁸) we used EXOSIMS to compute yield for 4, 6, and 8m unobscured telescopes having a coronagraph with inner working angle (IWA) of $2\lambda/D$ and $2.5\lambda/D$. The results, presented in Table 3, agree to within 30%, which is remarkable considering the vastly different approaches of the codes. AYO creates a Monte Carlo cloud of planets around a star to calculate completeness, and then allocates observing time among targets to optimize cumulative completeness (yield). EXOSIMS synthesizes a universe and dynamically schedules observations under keepout constraints into a DRM, subsequently averaging over a Monte Carlo ensemble of DRMs. The discrepancy between the results is likely due to mismatched assumptions, such as overhead time allocation, star selection, coronagraph sky throughput, or any of 150 parameters used in the simulation. This initial comparison is an encouraging result which warrants in-depth follow-up, starting at the level of photometric comparison similar to Section 4.3.

6. CONCLUSION

We used a three-level approach in testing EXOSIMS: unit tests, integration tests, and end-to-end simulations. Unit tests did identify errors, but the integration tests were more powerful in this regard. The errors identified by the unit tests were narrow in scope, and probably the greatest value offered by the unit tests is that they increased repeatability of the code, improved separation of functionality, and led to improved automation of interfaces to externally-supplied tabular data. One key function of unit tests, is that they permit confident changes to the code base through regression tests, may be realized in future adoption of automated testing upon code checkin. Integration tests proved more time-consuming to implement, but had higher payoff in terms of identifying errors or modeling shortcomings. The integration tests of EXOSIMS were performed iteratively in collaboration with the developers of EXOSIMS as a mix of errors, documentation ambiguities, and new features to enable realistic validation were accommodated by the team. The multifaceted nature of the simulation tool, covering exoplanet catalogs, orbit geometry, optical path modeling, and other concerns, required several types of integration test. The end-to-end simulations illustrate the minimum feasible scale of the Monte Carlo ensemble needed for self-consistency. The end-to-end simulations are in family with results from another yield modeling code that uses a very different approach. The result of validation testing is a more robust, stable, and trustworthy tool that the exoplanet community can use for end-to-end simulations of direct detection missions.

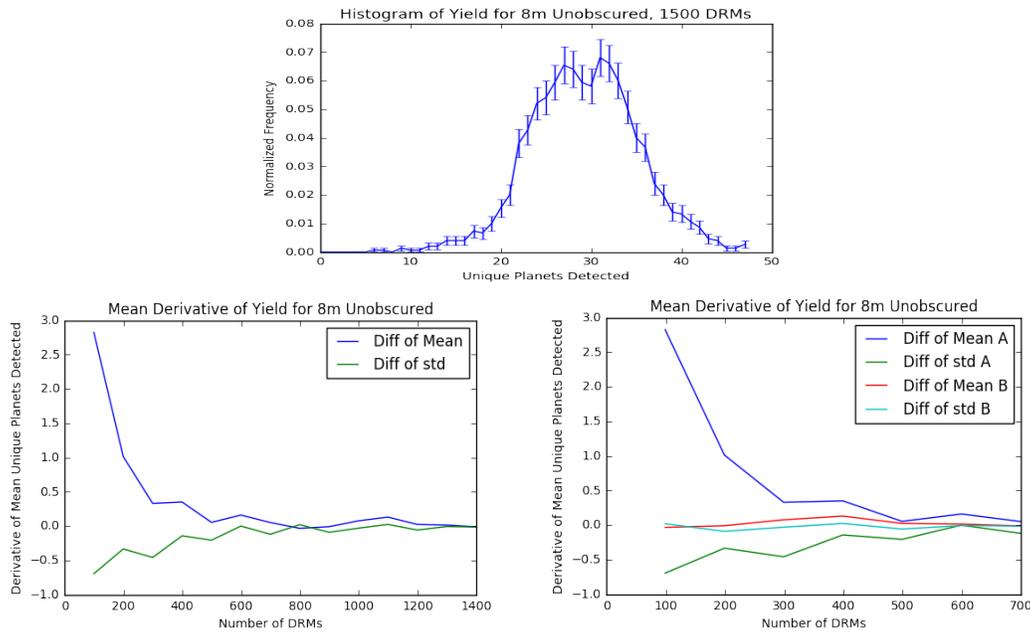


Figure 5. Binomial detection statistics on 1500 DRMs for an 8m unobscured aperture require 800 DRMs to reach stability.

ACKNOWLEDGMENTS

Copyright 2017. All rights reserved. The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. U.S. Government sponsorship acknowledged; this work was partially supported by the Exoplanet Exploration Program office, and NASA funds to Cornell.

REFERENCES

- [1] Garrett, D. and Savransky, D., “Science yield modeling with EXOSIMS,” in [*AAS Meeting Abstracts*], **227**, 137.02 (Jan. 2016).
- [2] Savransky, D., “EXOSIMS: Exoplanet open-source imaging mission simulator.” <https://github.com/dsavransky/EXOSIMS> (2017).
- [3] Morgan, R., Lowrance, P., Savransky, D., and Garrett, D., “Exoplanet yield estimation for decadal study concepts using EXOSIMS,” in [*AAS Meeting Abstracts*], **227**, 305.01 (Jan. 2016).
- [4] Savransky, D. and Garrett, D., “WFIRST-AFTA coronagraph science yield modeling with EXOSIMS,” *JATIS* **2**, 011006 (Jan. 2016).
- [5] Delacroix, C., Savransky, D., Garrett, D., Lowrance, P., and Morgan, R., “Science yield modeling with the Exoplanet Open-Source Imaging Mission Simulator (EXOSIMS),” in [*SPIE Conf. Series*], *Proc. SPIE* **9911**, 991119 (Aug. 2016).
- [6] Delacroix, C., Garrett, D., and Savransky, D., “Exoplanet open-source imaging mission simulator (exosims) interface control document.” <https://github.com/dsavransky/EXOSIMS/ICD> (2017).
- [7] NASA Exoplanet Science Institute, “NASA exoplanet archive.” <http://exoplanetarchive.ipac.caltech.edu> (2017).
- [8] Leinert, C., Bowyer, S., Haikala, L. K., Hanner, M. S., Hauser, M. G., Levasseur-Regourd, A.-C., Mann, I., Mattila, K., Reach, W. T., Schlosser, W., Staude, H. J., Toller, G. N., Weiland, J. L., Weinberg, J. L., and Witt, A. N., “The 1997 reference of diffuse night sky brightness,” *Astron. Astrophys. Suppl.* **127**, 1–99 (Jan. 1998).
- [9] Stark, C. C., Roberge, A., Mandell, A., and Robinson, T. D., “Maximizing the ExoEarth candidate yield from a future direct imaging mission,” *Astrophys. J.* **795**, 122 (Nov. 2014).

Table 4. Star-Independent Photometric Parameters.

Scenario	$C_{\text{localzodi}}(\lambda)$ [s]	C_{dc} [s]	C_{cc} [s]	C_{rn} [s]
Detection at 550nm	0.0165	0.00855	5.13e-5	0.000684
Detection at 750nm	0.023	0.0159	9.54e-5	0.00127
Characterization at 750nm	0.00164	0.0159	9.54e-5	0.00127

- [10] Cahoy, K. L., Marley, M. S., and Fortney, J. J., “Exoplanet albedo spectra and colors as a function of planet phase, separation, and metallicity,” *Astrophys. J.* **724**(1), 189 (2010).
- [11] Fortney, J. J., Marley, M. S., and Barnes, J. W., “Planetary radii across five orders of magnitude in mass and stellar insolation: Application to transits,” *Astrophys. J.* **659**, 1661–1672 (Apr. 2007). One error later corrected.
- [12] Fortney, J. J., Marley, M. S., and Barnes, J. W., “Erratum: “Planetary radii across five orders of magnitude in mass and stellar insolation: Application to transits,” ApJ, 659, 1661 [2007],” *Astrophys. J.* **668**, 1267–1267 (Oct. 2007).
- [13] NASA Navigation and Ancillary Information Facility, “SPICE: An observation geometry system for space science missions.” <https://naif.jpl.nasa.gov/naif/index.html> (2017).
- [14] Nemati, B., “Detector selection for the WFIRST-AFTA coronagraph integral field spectrograph,” *Proc. SPIE* **9143**, 9143 (2014).
- [15] Traub, W. A., Breckinridge, J., Greene, T. P., Guyon, O., Jeremy Kasdin, N., and Macintosh, B., “Science yield estimate with the Wide-Field Infrared Survey Telescope coronagraph,” *JATIS* **2**, 011020 (Jan. 2016).
- [16] Stark, C. C., Roberge, A., Mandell, A., and Robinson, T. D., “Maximizing the ExoEarth candidate yield from a future direct imaging mission,” *Astrophys. J.* **795**(2), 122 (2014).
- [17] Stark, C. C. personal communication (Jan. 2017).
- [18] Stark, C. C., Roberge, A., Mandell, A., Clampin, M., Domagal-Goldman, S. D., McElwain, M. W., and Stapelfeldt, K. R., “Lower limits on aperture size for an ExoEarth detecting coronagraphic mission,” *Astrophys. J.* **808**, 149 (Aug. 2015).

Table 5. Photometric Parameters for Significant Exoplanets: Detection at 550nm.

Star	Pl.	WA [arcsec]	Cp [μ s]	Cb [μ s]	Csp [μ s]	Csr [μ s]	Cexozodi [μ s]	Tint [s]
55 Cnc	d	0.439	0.023	0.0261	2.3e-5	0.00023	8.49e-5	1.23e+3
HD 160691	c	0.343	0.0323	0.0266	4.86e-5	0.000486	0.000294	635
HD 217107	c	0.27	0.0157	0.0262	1.92e-5	0.000192	0.000187	2.65e+3
HD 134987	c	0.261	0.00468	0.0261	1.47e-5	0.000147	0.000153	2.98e+4
47 UMa	c	0.256	0.0334	0.0269	5.33e-5	0.000533	0.000579	602
HD 190360	b	0.252	0.0292	0.0264	2.85e-5	0.000285	0.000318	773
HD 114613	b	0.252	0.0181	0.0271	6.4e-5	0.00064	0.000717	2.08e+3
HD 154345	b	0.232	0.00743	0.026	1.1e-5	0.00011	0.000146	1.18e+4
HD 87883	b	0.199	0.0074	0.0259	5.33e-6	5.33e-5	9.53e-5	1.18e+4
ups And	d	0.187	0.639	0.0297	0.000128	0.00128	0.00261	1.82
HD 39091	b	0.186	0.155	0.0267	3.06e-5	0.000306	0.000633	27.6
HD 62509	b	0.159	15.5	0.1	0.00195	0.0195	0.0552	0.0104
14 Her	b	0.153	0.0563	0.0263	1.27e-5	0.000127	0.000386	207
47 UMa	b	0.149	0.275	0.028	5.33e-5	0.000533	0.0017	9.26
gam Cep	b	0.149	1.28	0.038	0.00029	0.0029	0.00933	0.584
HD 192310	c	0.134	0.0448	0.0272	2.85e-5	0.000285	0.00113	339
HD 10647	b	0.116	0.1	0.0279	3.45e-5	0.000345	0.00182	69.7
HD 117207	b	0.115	0.00904	0.0262	6.96e-6	6.96e-5	0.000375	8.02e+3
HD 181433	d	0.115	0.00219	0.0259	2.43e-6	2.43e-5	0.000132	1.35e+5
HD 13931	b	0.113	0.00359	0.0261	5.1e-6	5.1e-5	0.000283	5.06e+4
HD 70642	b	0.112	0.0138	0.0263	7.49e-6	7.49e-5	0.000422	3.43e+3
GJ 676 A	b	0.112	0.00901	0.0258	8.21e-7	8.21e-6	4.69e-5	7.95e+3
GJ 649	b	0.11	0.00336	0.0258	7.42e-7	7.42e-6	4.36e-5	5.72e+4
HD 24040	b	0.106	0.00718	0.0262	5.61e-6	5.61e-5	0.000357	1.27e+4
HD 222155	b	0.101	0.00576	0.0264	7.96e-6	7.96e-5	0.000554	1.99e+4

Table 6. Photometric Parameters for Significant Exoplanets: Detection at 750nm.

Star	Pl.	WA [arcsec]	Cp [μ s]	Cb [μ s]	Csp [μ s]	Csr [μ s]	Cexozodi [μ s]	Tint [s]
55 Cnc	d	0.439	0.0314	0.041	5.84e-5	0.000584	0.000118	1.04e+3
HD 160691	c	0.343	0.0392	0.0418	0.00011	0.0011	0.00041	678
HD 217107	c	0.27	0.0196	0.041	4.44e-5	0.000444	0.000261	2.67e+3
HD 134987	c	0.261	0.00568	0.0408	3.31e-5	0.000331	0.000213	3.17e+4
47 UMa	c	0.256	0.0381	0.0422	0.000113	0.00113	0.000808	726
HD 190360	b	0.252	0.0359	0.0414	6.51e-5	0.000651	0.000443	801
HD 114613	b	0.252	0.0219	0.0427	0.000144	0.00144	0.001	2.22e+3
HD 154345	b	0.232	0.0092	0.0407	2.54e-5	0.000254	0.000203	1.2e+4
HD 87883	b	0.199	0.011	0.0405	1.47e-5	0.000147	0.000133	8.43e+3
ups And	d	0.187	0.694	0.0465	0.000258	0.00258	0.00364	2.41
HD 39091	b	0.186	0.176	0.0418	6.46e-5	0.000646	0.000882	33.7
HD 62509	b	0.159	23.1	0.171	0.00541	0.0541	0.0769	0.00801
14 Her	b	0.153	0.0773	0.0411	3.23e-5	0.000323	0.000539	172
47 UMa	b	0.149	0.314	0.0438	0.000113	0.00113	0.00237	11.1
gam Cep	b	0.149	1.94	0.0615	0.000821	0.00821	0.013	0.408
HD 192310	c	0.134	0.0614	0.0426	7.26e-5	0.000726	0.00158	282
HD 10647	b	0.116	0.11	0.0435	7.03e-5	0.000703	0.00254	90.5
HD 117207	b	0.115	0.0111	0.0409	1.59e-5	0.000159	0.000523	8.27e+3
HD 181433	d	0.115	0.00336	0.0405	6.94e-6	6.94e-5	0.000183	8.95e+4
HD 13931	b	0.113	0.00419	0.0408	1.11e-5	0.000111	0.000395	5.82e+4
HD 70642	b	0.112	0.0169	0.041	1.7e-5	0.00017	0.000589	3.59e+3
GJ 676 A	b	0.112	0.0182	0.0404	3.08e-6	3.08e-5	6.54e-5	3.05e+3
GJ 649	b	0.11	0.00707	0.0404	2.9e-6	2.9e-5	6.08e-5	2.02e+4
HD 24040	b	0.106	0.00857	0.0409	1.25e-5	0.000125	0.000497	1.39e+4
HD 222155	b	0.101	0.00673	0.0412	1.73e-5	0.000173	0.000772	2.28e+4

Table 7. Photometric Parameters for Significant Exoplanets: Spectral Characterization at 750nm.

Star	Pl.	WA [arcsec]	Cp [/s]	Cb [/s]	Csp [/s]	Csr [/s]	Cexozodi [/s]	Tint [s]
55 Cnc	d	0.439	0.00224	0.0212	4.17e-6	4.17e-5	8.46e-6	4.21e+5
HD 160691	c	0.343	0.0028	0.0218	7.83e-6	7.83e-5	2.93e-5	2.78e+5
HD 217107	c	0.27	0.0014	0.0204	3.17e-6	3.17e-5	1.86e-5	1.04e+6
HD 134987	c	0.261	0.000406	0.0194	2.36e-6	2.36e-5	1.52e-5	1.18e+7
47 UMa	c	0.256	0.00272	0.0218	8.07e-6	8.07e-5	5.77e-5	2.94e+5
HD 190360	b	0.252	0.00257	0.0216	4.65e-6	4.65e-5	3.17e-5	3.27e+5
HD 114613	b	0.252	0.00157	0.0207	1.03e-5	0.000103	7.14e-5	8.44e+5
HD 154345	b	0.232	0.000657	0.0196	1.81e-6	1.81e-5	1.45e-5	4.54e+6
HD 87883	b	0.199	0.000783	0.0197	1.05e-6	1.05e-5	9.49e-6	3.21e+6
ups And	d	0.187	0.0496	0.069	1.84e-5	0.000184	0.00026	2.8e+3
HD 39091	b	0.186	0.0126	0.0316	4.61e-6	4.61e-5	6.3e-5	2e+4
HD 62509	b	0.159	1.65	1.68	0.000387	0.00387	0.00549	61.6
14 Her	b	0.153	0.00552	0.0245	2.31e-6	2.31e-5	3.85e-5	8.03e+4
47 UMa	b	0.149	0.0224	0.0416	8.07e-6	8.07e-5	0.00017	8.28e+3
gam Cep	b	0.149	0.139	0.159	5.86e-5	0.000586	0.000929	827
HD 192310	c	0.134	0.00439	0.0235	5.19e-6	5.19e-5	0.000113	1.22e+5
HD 10647	b	0.116	0.00783	0.027	5.02e-6	5.02e-5	0.000181	4.4e+4
HD 117207	b	0.115	0.000795	0.0198	1.14e-6	1.14e-5	3.74e-5	3.13e+6
HD 181433	d	0.115	0.00024	0.0192	4.96e-7	4.96e-6	1.31e-5	3.32e+7
HD 13931	b	0.113	0.000299	0.0192	7.9e-7	7.9e-6	2.82e-5	2.15e+7
HD 70642	b	0.112	0.00121	0.0202	1.22e-6	1.22e-5	4.2e-5	1.38e+6
GJ 676 A	b	0.112	0.0013	0.0202	2.2e-7	2.2e-6	4.67e-6	1.2e+6
GJ 649	b	0.11	0.000505	0.0194	2.07e-7	2.07e-6	4.34e-6	7.62e+6
HD 24040	b	0.106	0.000612	0.0196	8.9e-7	8.9e-6	3.55e-5	5.22e+6
HD 222155	b	0.101	0.00048	0.0195	1.23e-6	1.23e-5	5.51e-5	8.44e+6