Lawrence Berkeley National Laboratory

Title
DESCQA: An Automated Validation Framework for Synthetic Sky Catalogs

Permalink
https://escholarship.org/uc/item/3dz0170b

Journal
The Astrophysical Journal Supplement Series, 234(2)

ISSN
0067-0049

Authors
Mao, Yao-Yuan
Kovacs, Eve
Heitmann, Katrin
et al.

Publication Date
2018-02-01

DOI
10.3847/1538-4365/aaa6c3

Peer reviewed

DESCQA: An Automated Validation Framework for Synthetic Sky Catalogs

Yao-Yuan Mao,[1] Eve Kovacs,[2] Katrin Heitmann,[2] Thomas D. Uram,[2] Andrew J. Benson,[3] Duncan Campbell,[4] Sofía A. Cora,[5,6] Joseph DeRose,[7] Tiziana Di Matteo,[4] Salman Habib,[2] Andrew P. Hearin,[2] J. Bryce Kalmbach,[8] K. Simon Krughoff,[9] François Lanusse,[4] Zarija Lukić,[10] Rachel Mandelbaum,[4] Jeffrey A. Newman,[1] Nelson Padilla,[11] Enrique Paillas,[11] Adrian Pope,[2] Paul M. Ricker,[12] Andrés N. Ruiz,[13] Ananth Tenneti,[4] Cristian A. Vega-Martínez,[5] Risa H. Wechsler,[7,14] Rongpu Zhou,[1] and Ying Zu[15,16]

(The LSST Dark Energy Science Collaboration)

[1]Department of Physics and Astronomy and the Pittsburgh Particle Physics, Astrophysics and Cosmology Center (PITT PACC), University of Pittsburgh, Pittsburgh, PA 15260, USA

[2]Argonne National Laboratory, Lemont, IL 60439, USA

[3]Carnegie Observatories, Pasadena, CA 91101, USA

[4]McWilliams Center for Cosmology and Department of Physics, Carnegie Mellon University, Pittsburgh, PA 15213, USA

[5]Instituto de Astrofísica de La Plata (CCT La Plata, CONICET, UNLP), Paseo del Bosque s/n, B1900FWA, La Plata, Argentina

[6]Facultad de Ciencias Astronómicas y Geofísicas, Universidad Nacional de La Plata, Observatorio Astronómico, Paseo del Bosque, B1900FWA La Plata, Argentina

[7]Kavli Institute for Particle Astrophysics and Cosmology & Department of Physics, Stanford University, Stanford, CA 94305, USA

[8]Department of Physics, University of Washington, Seattle, WA 98105, USA

[9]Large Synoptic Survey Telescope, 950 N Cherry Avenue, Tucson, AZ 85719, USA

[10]Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

[11]Instituto de Astrofísica, Pontificia Universidad Católica de Chile, Av. Vicuña Mackenna 4860, Santiago, Chile

[12]Department of Astronomy, University of Illinois, Urbana, IL 61801, USA

[13]Instituto de Astronomía Teórica y Experimental (CONICET-UNC) and Observatorio Astronómico (UNC), Laprida 854, X5000BGR, Córdoba, Argentina

[14]SLAC National Accelerator Laboratory, Menlo Park, CA, 94025, USA

[15]Department of Astronomy, Shanghai Jiao Tong University, 955 Jianchuan Road, Shanghai 200240, People's Republic of China

[16]Center for Cosmology and AstroParticle Physics (CCAPP), Ohio State University, Columbus, OH 43210, USA

ABSTRACT

The use of high-quality simulated sky catalogs is essential for the success of cosmological surveys. The catalogs have diverse applications, such as investigating signatures of fundamental physics in cosmological observables, understanding the effect of systematic uncertainties on measured signals and testing mitigation strategies for reducing these uncertainties, aiding analysis pipeline development and testing, and survey strategy optimization. The list of applications is growing with improvements in the quality of the catalogs and the details that they can provide. Given the importance of simulated catalogs, it is critical to provide rigorous validation protocols that enable both catalog providers and users to assess the quality of the catalogs in a straightforward and comprehensive way. For this purpose, we have developed the DESCQA framework for the Large Synoptic Survey Telescope Dark Energy Science Collaboration as well as for the broader community. The goal of DESCQA is to enable the inspection, validation, and comparison of an inhomogeneous set of synthetic catalogs via the provision of a common interface within an automated framework. In this paper, we present the design concept and first implementation of DESCQA. In order to establish and demonstrate its full functionality we use a set of interim catalogs and validation tests. We highlight several important aspects, both technical and scientific, that require thoughtful consideration when designing a validation framework, including validation metrics and how these metrics impose requirements on the synthetic sky catalogs.

Keywords: methods: numerical – large-scale structure of the universe

1. INTRODUCTION

The Large Synoptic Survey Telescope (LSST) will conduct the most comprehensive optical imaging survey of the sky to date, yielding a wealth of data for astronomical and cosmological studies. LSST data will offer many exciting scientific opportunities, including the creation of very detailed maps

of the distribution of galaxies, studies of transient objects in new regimes, investigations of the inner and outer solar system, observations of stellar populations in the Milky Way and nearby galaxies, and studies of the structure of the Milky Way disk and halo and other objects in the Local Volume. A broad description of the LSST scientific goals is provided in the LSST Science Book (LSST Science Collaboration et al. 2009).

Several science collaborations have been formed in order to prepare for the arrival of the rich and complex LSST data set. One of these collaborations, the LSST Dark Energy Science Collaboration (LSST DESC), focuses on the major dark energy investigations that can be carried out with LSST, including weak- and strong-lensing measurements, baryon acoustic oscillations, large-scale structure (LSS) measurements, supernova distances, and galaxy cluster abundance. An overview of LSST DESC goals is provided in the White Paper authored by the Collaboration (2012); a detailed Science Road Map can be found at the Collaboration Web site[1].

Science opportunities relevant to LSST DESC will pose many new analysis challenges on different fronts, including controlling systematic errors, extracting subtle signals from large data sets, combining different dark energy probes, cross-correlations with other observations, etc. The best tools for these tasks must extract the science of interest while simultaneously enabling control of systematic contaminants, whether observational or modeling induced. Before the data set arrives, robust synthetic sky catalogs that are validated against a range of observational data are essential to the development of the required analysis tools.

The comprehensive and systematic validation of synthetic sky catalogs presents major challenges. Observational data sets used for validation must be carefully curated and frequently updated with the best available measurements. Tests comparing observations with synthetic data sets have to be designed to address the wide range of tasks for which the catalogs will be used, e.g., tests of photometric pipelines, extraction of cosmological parameters, mass estimates for clusters, etc. The list is essentially as long as the set of analysis tasks to be covered by the survey. For each of these tasks, a set of requirements, such as accurate clustering statistics, best possible match to observed colors, detailed galaxy properties, or results over a range of different redshift epochs, needs to be defined and implemented as part of the validation tests. The synthetic catalogs will be revised, enhanced, and improved over time; a controlled and easy-to-use mechanism to expose new synthetic catalogs to a full battery of observational tests is essential to validating the catalogs properly. In addition, for the users of the catalogs, it is very desirable to have a convenient method to check the catalog quality for their specific needs. In order to provide an environment that can address all of these challenges in a streamlined way, we present DESCQA, a validation framework that has been con-

figured to compare and validate multiple different synthetic sky catalogs in an automated fashion.

There are a number of requirements that must be met in designing a comprehensive and flexible framework intended to assess the performance of multiple synthetic sky catalogs. The synthetic catalogs that the system has to handle will have a range of very different characteristics and features depending on how the catalogs are constructed and their ultimate purpose. There are many different ways to build synthetic sky catalogs. Because of the large survey volumes required for cosmological studies, most current methods are based on gravity-only simulations rather than on the significantly more expensive hydrodynamic methods. A range of approaches is used to assign galaxies to dark matter halos in post-processing. These include halo occupation distribution modeling (HOD; Kauffmann et al. 1997; Jing et al. 1998; Peacock & Smith 2000; Seljak 2000; Berlind & Weinberg 2002; Bullock et al. 2002; Zheng et al. 2005; Mandelbaum et al. 2006; Zu & Mandelbaum 2015), subhalo abundance matching (SHAM; Vale & Ostriker 2004; Conroy et al. 2006; Behroozi et al. 2010; Moster et al. 2010; Wetzel & White 2010; Reddick et al. 2013), and semi-analytic modeling (SAM; White & Frenk 1991; Kauffmann et al. 1993; Cole et al. 1994; Somerville & Primack 1999; Benson et al. 2000, 2003; Baugh 2006; Benson 2010). The choice of method used depends on the base simulation (e.g., resolution, available information with regard to time evolution) and the observational data sets that the catalog is expected to match. All known methods are, at best, only partially predictive, and each individual choice has its own advantages and disadvantages, such as resolution requirements, predictive power, ease of implementation, time to run, etc. In addition, different science cases impose different requirements on the catalogs. Roughly speaking, resolving smaller physical scales increases modeling difficulty, while including broader classes of galaxies adds to the complexity. The galaxy properties required will also influence the choice of method employed.

Given the current uncertainties in galaxy modeling, it is not possible to address the full range of science issues with only one catalog construction method (or a single base simulation). Instead, catalog providers choose the methods that are best suited to address specific questions (or classes of such questions) of interest. The heterogeneity among the catalogs manifests itself in both the implementation details, such as file formats, units, and quantity labels, and scientific details, such as the choice of halo-finder algorithms, mass definitions, and filter-band definitions. This heterogeneity presents a significant barrier for users who wish to use several of these catalogs. The framework therefore needs to be capable of ingesting a wide range of synthetic sky catalogs that have very different intrinsic characteristics.

The framework's success hinges on a set of well-thought-out validation tests that can act seamlessly on the catalogs. The tests, as well as the criteria for how well a catalog performs in a given test, will be provided by domain experts (e.g., members of analysis working groups). They will have the best understanding of the requirements and often also of

---

[1] lsstdesc.org

the validation data. The difficulty of adding a new test to the framework therefore has to be minimal so that most contributions can be made without significant assistance from the framework developers. In addition, the execution of the test, once it is implemented, needs to be carried out automatically on all catalogs that provide the necessary information for the test. Tests and catalogs will be improved over time and new ones will be added. The framework needs to provide straightforward methods to accommodate these updates and additions.

Finally, since the validation tests will be run on a wide range of synthetic sky catalogs, it is very desirable to have a convenient method to check which catalogs meet the specific needs for certain tasks. The results must be presented in a way that gives the catalog users an easy-to-use interface to peruse the different tests and catalogs. At the same time, the interface should provide a useful and easily interpretable overview. For this reason, sets of summary statistics and simple assessment criteria of the catalog quality need to be provided.

To summarize, a validation framework used by a survey collaboration such as LSST DESC should be able to (1) process a wide range of heterogeneous catalogs, (2) automate the validation tests, (3) provide straightforward methods to update and add catalogs and tests, and (4) provide easy access to catalog feature summaries and quality assessments for catalog users.

DESCQA addresses the above requirements and provides interfaces to access synthetic catalogs and to run a set of pre-specified validation tests. The results of the tests are graphically displayed and evaluated via well-defined error metrics. DESCQA is accessed via a Web-based portal, which is currently set up at the National Energy Research Scientific Computing Center (NERSC). The portal itself can, in principle, be set up anywhere, but collocating the portal with the storage and analysis resources is convenient.

In order to demonstrate the full functionality of the framework and to have a sufficiently complex environment for testing and development, it is vital to use realistic, scientifically interesting synthetic sky catalogs and validation tests. LSST DESC is still in the process of defining the requirements for each of its analyses and producing new synthetic sky catalogs (including comprehensive full-coverage light-cone catalogs). Therefore, we have chosen to implement a set of interim validation tests and requirements to assist us in the development of the current version of DESCQA; we present these tests as our case studies of the DESCQA framework. Also, since one major requirement of this framework is to process a wide range of heterogeneous catalogs, we also select a set of interim catalogs that cover the major synthetic methods to be used in our case studies.

The interim catalogs and tests presented in this work fulfill functions beyond the provision of demonstration examples. Although these catalogs and tests are not the final versions that will be used for LSST DESC science, their realistically complicated features provide a unique opportunity to delve into the conceptual challenges of building a validation framework. These challenges originate from the different choices made by the creators of the catalogs and tests, such as the definitions of physical quantities. These intrinsic differences cannot be easily homogenized by the framework; however, the framework can highlight them for the scientists who use the framework. Working with this set of interim catalogs and tests, we have identified several such conceptual challenges. Furthermore, our implementation of the validation framework also provides a concrete platform for publicly and quantitatively *defining* the requirements for a particular scientific analysis.

The paper is organized as follows. We first describe the design and implementation of the DESCQA framework in Section 2. We explain in detail our method for adding synthetic catalogs to the framework and show how the method enables the automated testing and validation of these catalogs. We also discuss how the Web interface helps the user to navigate the catalogs and validation tests. Then, in Section 3, we present our case studies of five interim validation tests to demonstrate the features of the framework. The description of the different methods employed to build a range of interim synthetic sky catalogs can be found in Appendix A. We conclude in Section 4 with a summary and discussion of DESCQA and future development paths.
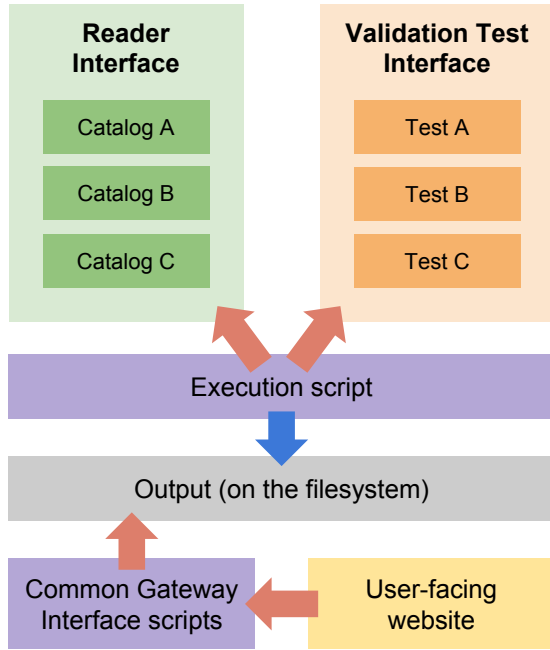
2. DESCQA FRAMEWORK

In this section, we describe the DESCQA framework, a unified environment for comparing different synthetic catalogs and data sets using a number of validation tests.

The DESCQA framework is based on DESQA, which was originally developed for validating catalogs for the Dark Energy Survey. DESQA, in turn, originated from the FlashTest framework, which was developed for standard regression testing (software testing) of the Flash code. Since regression testing is a considerably simpler task than the validation of catalogs, we had to make multiple changes to the framework to accommodate the design goals discussed in Section 1. The basic structure common to all of the variants of the framework is a set of scripts that execute the tests and a Web interface that displays the results. For DESCQA, although much of the original framework has been revised or replaced, the use of the Python programming language is retained (but revised to be compatible with Python 3), along with some portion of the original Web interface, and several of the original concepts used in FlashTest.

Figure 1 presents the organization of the framework, which possesses four main components: (1) the reader interface, (2) the validation test interface, (3) the automated execution process, and (4) the Web interface. Together, they enable an expandable, automatable, and trackable validation process. The code of the DESCQA framework is publicly available in a GitHub repository[2]. A frozen version of the code can be found in LSST Dark Energy Science Collaboration (2017).

---

[2] github.com/LSSTDESC/descqa

Figure 1. Illustration of the DESCQA framework. The two purple boxes are the main drivers of this framework. The execution script, which is designed to be triggered periodically, accesses (red arrows) the available catalogs and tests through the reader interface and the validation test interface respectively, and then writes out (blue arrow) the results (including scores and figures) on the filesystem. The Common Gateway Interface scripts, which are triggered when a user accesses the Web interface, read the test results that are available on the filesystem, and present them as browsable Web pages.

2.1. *Design Guidelines*

In designing the framework for DESCQA, our priorities were to provide a unified environment for a set of validation tests that examine and compare different synthetic catalogs, while ensuring that new catalogs and validation tests can be added or updated with minimal effort required from the framework developers, the test developers, and the catalog providers. At the same time, we also want to ensure that the validation results generated by the framework and delivered to the user are easy to understand and compare.

The above set of considerations all aim at minimizing the overhead in meeting the requirements imposed by the framework. To help achieve this goal, we separate the framework into three major components, which are as independent as possible. These components are detailed in the following sections: the reader interface, the validation test interface, and the Web interface. In fact, the only requirement for the catalog providers and test developers is to conform to these common application programming interfaces (APIs) for accessing the catalogs (with the reader interface) and for executing the validation tests (with the validation test interface). There is *no* formal requirement regarding the underlying implementation. The flexibility of Python enables providers and users to implement their readers and validation tests using

methods ranging from reading existing files to running an external executable, as long as they provide a Python interface that is consistent with the API specification.

In practice, most catalog providers (test developers) need similar high-level functionality in their catalog readers (validation tests). From the point of view of code development, it is desirable to reduce code duplication in order to maintain consistency and reduce human errors. We have designed base classes for both catalog readers and validation tests. Catalog providers and test developers write subclasses that inherit from the base classes, but overwrite the core functions with their own, if needed.

All of the above features reflect our design philosophy of providing an efficient, flexible, automated framework that is capable of including a diverse set of synthetic catalogs and validation tests with the fewest possible requirements imposed on the contributors and code developers.

2.2. *Reader Interface*

Given the heterogeneity among different synthetic catalogs, it is impractical to access all the different catalogs directly in their native formats within the framework. The standardized reader interface solves this problem by associating with each catalog a corresponding *reader* that loads the catalog in its native format, provides metadata and a list of galaxy properties available in that catalog, and processes any necessary unit or definition conversions. The catalog provider implements the corresponding *reader* that conforms to the specification of the standardized reader interface. In this fashion, all of the catalogs can be stored in their native formats, and no change to the catalog's generation pipeline is required. Similarly, the reader can also be used to fix minor errors in the native catalogs (e.g., incorrect definitions or units) during the conversion between native quantities and user-facing quantities, thereby reducing the number of catalog files while still propagating updates to the users.

We implemented a base class that contains some basic catalog process methods. To implement a new reader for a specific catalog, one would first subclass this base class, and then supply or overwrite the methods (e.g., data loading routine) to accommodate the catalog under consideration. Since we expect that different versions of a specific kind of catalog would be accessed using the same code, we allow the same reader to be used with different configuration parameters that specify, for example, the paths of the catalog files and versions. These configuration parameters are passed as arguments when the subclass is initialized. The reader can also check the catalog's version against an online catalog repository and warn the user if the catalog in use is out of date.

Although some catalog variations such as units and quantity labels can be homogenized by the reader interface, others such as mass definitions and cosmology cannot. We do *not* ask the catalog providers to conform to a specific list of standards, but instead ask that they specify their choices as metadata available in the readers. Similarly, we do *not* require the catalog providers to include all quantities that are needed for all validation tests. During execution, a catalog that does

not have some requested quantities for a specific test will be skipped and noted.

In this particular study, the reader interface is used to enforce consistent units across different catalogs. We use comoving Mpc (not scaled to $h = 1$) for all distance units, physical km s^{-1} for all velocity units, and M$_\odot$ (not scaled to $h = 1$) for all masses (including stellar mass and halo mass).

Note that the reader interface itself can actually do more than serving data to the DESCQA framework. It can, in fact, be used as a standalone catalog data server or as a converter to convert catalogs from their native format into a database with common schema. We package the reader interface as a standalone Python module,[3] which allows people to access the homogenized synthetic galaxy catalogs conveniently outside the validation framework. Under this new structure, the DESCQA framework itself becomes a user of the reader interface.

2.3. *Validation Test Interface*

An important part of our framework is the quality assurance (QA) implementation, which allows test developers to design validation tests and provides a convenient interface for users to assess the quality of the synthetic catalogs. In DE-SCQA, a *validation test* is carried out to establish whether a synthetic catalog meets some particular requirements that have been set by the test developer. The validation test consists of two parts. First, the catalog is checked to see if it provides the quantities required for the specific test. Next, the catalog is tested to see if it can reproduce relevant observational data over a specified range at the required accuracy.

We have designed a standardized validation test interface which is similar in concept to the reader interface. We implemented a base class with abstract methods for the validation test interface. Each individual test is a subclass of this base class and contains the non-abstract methods to conduct the test. The test interface also separates the configuration from the code that carries out the actual computation to allow convenient changes to the specific settings for each test.

Each test uses specified catalog quantities (already wrapped by the reader interface) as input, carries out necessary calculations, creates figures and reduced data, and finally returns a summary statistic. As mentioned in Section 2.1, if a catalog does not provide all of the required quantities for a particular test, the test will automatically skip the catalog and proceed with the remaining catalogs.

Each test must provide a summary statistic (score) for each catalog on which it runs and also provide a score threshold to determine if a catalog "passes" the test. The score and the passing threshold are both up to the test developer to set in the most useful way for that particular test. The notion of "passing" and "failing" a test is intended to give the user a quick method to inspect the summarized results using the Web interface, as we detail below. The notion is not to judge the quality of a catalog, as each catalog has its own features.

Furthermore, many users are only interested in a subset of validation tests, so a catalog does not need to pass every test in order to be scientifically useful.

In addition to the score, the framework allows the validation tests to generate figures, tables, and other products. These supplementary products are saved on the filesystem so that they can be accessed by the web-interface component of the framework as described below. Many users also find it helpful to have a summary figure that displays the relevant statistics for all available catalogs. Although the validation test interface does not formally require all tests to generate plots, we do provide a convenient plotting module to produce basic summary figures which test developers can utilize. Alternatively, developers may supply their own plotting modules. We will show examples of these summary figures that are generated by our common plotting module in Section 3 for each of our currently implemented tests.

We should note that currently all validation tests implemented in this framework are for demonstration purposes. Although this set of tests represents the major tests that are relevant to LSST DESC science, the choice of summary statistics and the passing criteria presented here are preliminary and introduced only as interim values. In future, test developers and catalog users will set more realistic criteria by which to evaluate the catalogs according to the LSST DESC science goals.

2.4. *Automated Execution and Web Interface*

Since our design separates the configurations and the actual reader and test code (which are implemented as classes), a master execution script is required to run the tests. For both catalogs and tests, the master execution script reads in the configuration files (specified in the YAML format), identifies the corresponding reader or test classes to load, and passes the configurations to the class and executes the class methods. The master execution script has access to all available catalogs and tests, and can execute all desired combinations. By default, the master execution script is set to run periodically, thereby automating the full validation process.

The master execution script is also equipped to handle failures, relying on Python's context manager. The execution script captures all exceptions and traceback information, together with any content printed to standard output or standard error during runtime, and stores them in a log file. This is done for each of the combinations of all tests and catalogs, and when one of them fails, the master execution script writes out the log file, and continues to the next combination without being interrupted.

This design makes our framework easily expandable. Including new catalogs or validation tests requires no changes to existing code. Once the new reader or test class and its corresponding configuration file are placed in the pre-specified location, the master script will automatically include the new catalog or test in future runs. In this fashion, catalog readers and validation tests can remain agnostic about what catalogs and tests are available, and hence do not require updates when new catalogs and tests are added.

---

[3] github.com/LSSTDESC/gcr-catalogs

For the seven tests and eight catalogs presented in Section 3, since the catalogs are fairly small (made out of the $100\,\mathrm{Mpc}\,h^{-1}$ box), it takes only about 10 minutes on a single CPU core to run all of the combinations. The eight catalogs take less than 25 GB of disk space (excluding the underlying DMO simulation). If we were to run a test whose computational cost scales linearly with the number of galaxies on a mock catalog that corresponds to about 10,000 square degrees, it would take tens of minutes on a single CPU core, and this catalog would take about 1TB of disk space (depending on how many galaxy properties are stored). These numbers are still manageable by modern standards; however, in the future when facing very large catalogs and more computationally involved tests (such as two-point statistics), we will need distributed computation. Although we have not yet explored this direction, we believe our framework is flexible enough to accommodate distributed computation. In particular, our framework can be configured to run different tests on different catalogs using separate cores, without the need for cross-node communication (i.e., embarrassingly parallelizable).

All of the results (including the plots and summary statistics generated by the validation tests) can be archived periodically. Although the user can certainly inspect individual output files to access validation results, this process will rapidly become tedious with the increasing numbers of catalogs and tests. To avoid this difficulty, we have built a Web interface[4] at NERSC to assist users in quickly inspecting validation results. When users visit the Web interface, the Common Gateway Interface scripts will read in the output files available on the filesystem, and present users with a visual summary of the results. The Web interface also allows the user to browse through different runs. When each run is executed, a copy of the code used is recorded so that the results can be easily tracked.

Figure 2 shows an example of the summary page of the Web interface, which is presented in the form of a validation matrix. This matrix provides a quick summary of all validation tests (rows) and all available catalogs (columns). Each colored cell shows the corresponding test result. Each of the current set of validation tests provides a score that is between 0 and 1. A higher score indicates a larger discrepancy between the catalog under consideration and the validation data set. When the score is higher than a certain predefined value, it is noted as "failed." As already noted, the specific values and passing criteria are for demonstration purposes and do not reflect the actual LSST DESC requirements. In this matrix view, users can further click on the header or the cell to see the associated plots and output files. This interface helps the users to quickly find the catalogs that satisfy their desired requirements.

The DESCQA framework utilizes the filesystem to serve the Web interface and avoids direct communication between the test scripts and the Web interface. Hence, the framework can be easily adapted and applied more broadly to many other comparison tasks, such as a comparison of different code implementations.

2.5. *Documentation and Maintenance*

The DESCQA framework faces many different types of users, and hence requires many different forms of documentation. The framework is highly modularized, enabling different types of users to contribute without the overhead of understanding the full framework. Here we describe the different requirements on and our implementation of the documentation as well as related issues regarding maintenance.

1. For Web interface users who want to browse the results and plots of validation tests: the Web interface is self-explanatory and requires little documentation. On the front page, we provide basic instructions of how to navigate the interface, along with links to our papers, code repositories, and internal documentation pages, where users can easily find further information.

2. For catalog users who want to access the synthetic catalogs through the reader interface: we provide both API documentation[5] to the reader interface and also an example code[6] (many of which use easily browsable Jupyter notebooks) to help users understand quickly how to access the synthetic catalogs using the reader interface.

3. For users who want to implement a validation test to be included in the DESCQA framework: as discussed above, since the validation tests themselves are very much independent of the rest of the framework, the knowledge required for a test writer to be able to contribute is much reduced. In particular, test developers need only to implement a subclass that inherits the base class of validation tests and to follow the instructions[7] on how to implement a few specific member methods. We also provide step-by-step instructions on how to manually trigger the validation framework to test newly implemented validation tests[8].

4. For catalog providers who want to contribute their catalogs: similar to the case for new test developers, new catalog providers need only to implement a subclass that inherits the reader base class and to follow the instructions[5] on how to implement a few specific member methods. The catalog providers can test their newly implemented readers either with the full DESCQA framework or through the importable reader module.

	CAM_LiWhite	CAM_MBII	Galacticus	MBII	SAG	SHAM_LiWhite	SHAM_MBII	iHOD
Color_SDSS	FAILED 0.0158	FAILED 0.0172	FAILED 0.0717	FAILED 0.0494	FAILED 0.0703	FAILED 0.0461	FAILED 0.0445	FAILED 0.0373
HMF_Tinker	PASSED 0.85	PASSED 0.85	PASSED 0.523	FAILED 1	PASSED 0.567	PASSED 0.527	PASSED 0.527	PASSED 0.537
SMF_LiWhite	PASSED 0.00331	FAILED 1	FAILED 1	FAILED 1	FAILED 1	PASSED 0.031	FAILED 1	FAILED 1
SMF_MBII	FAILED 1	PASSED 0.542	FAILED 1	PASSED 2.77e-106	FAILED 1	FAILED 1	PASSED 0.0903	FAILED 1
SMHM_MBII	FAILED 1	FAILED 1	FAILED 1	PASSED 2.61e-16	FAILED 1	FAILED 1	FAILED 1	FAILED 1
WpRp_MBII	FAILED 0.986	PASSED 0.768	PASSED 0.717	PASSED 6.49e-133	PASSED 0.762	FAILED 0.984	PASSED 0.72	PASSED 0.252
WpRp_SDSS	FAILED 1	FAILED 0.986	FAILED 0.999	PASSED 0.256	FAILED 0.987	FAILED 1	PASSED 0.911	FAILED 0.972

Figure 2. Screenshot of the summary page of the DESCQA Web interface[9] demonstrating one aspect of the framework. For the current set of catalogs and validation tests, this matrix provides a quick overall view of how each catalog (column) performs on each validation test (row). Each validation test also provides a score for each catalog it runs on and a passing threshold that is designed to indicate whether the catalog being tested meets the specific requirement of the test under consideration. For the current set of validation tests, the scores are always between 0 and 1, and a larger score value indicates a larger discrepancy between the catalog in consideration and the validation data set; note that the specific values and passing criteria are for demonstration purposes and do not reflect the actual LSST DESC requirements. This matrix enables users to quickly identify the synthetic catalogs that satisfy their needs. A detailed discussion of these validation tests is provided in Section 3.

5. For users who maintain the Web interface and execution scripts, which includes DESCQA framework maintainers but not most regular users: we have made these components of our framework highly modularized and self-documenting such that future maintainers can navigate the code easily. We are working on other visual aids such as flowcharts to help future maintainers understand the code structure better.

While we continue to improve these various aspects of documentation, recent feedback from users both within the DESC Collaboration and elsewhere suggest that we already have adequate documentation for the different types of users to utilize or to contribute to this framework.

In addition, we work closely with the computing infrastructure working group of the Collaboration to ensure that the code base of this framework is kept up to date with the development environment (e.g., Python and Python packages), so as to reduce potential dependencies on deprecated packages and to benefit from better performance and new features. The Collaboration intends to continue to use, support, and develop the DESCQA framework, and will help to ensure that a period of overlap and effective communication is enabled between the current and future maintainers.

3. CASE STUDIES

To demonstrate the design and features of the DESCQA framework, we present five validation tests as case studies. Table 1 provides a summary of these five tests and the corresponding validation data sets and criteria. These criteria can be defined to satisfy specific science goals; however, as LSST DESC is still finalizing science requirements, the criteria used here are for demonstration purposes.

As mentioned earlier, one important requirement of this framework is that it needs to be suitable for a wide range of heterogeneous synthetic catalogs. Hence, we select eight realistic synthetic catalogs that encompass the major classes of methods that are generally used to create synthetic galaxies

(HOD, SHAM, SAM, and hydrodynamical simulations) to use in our case studies.

Table 2 summarizes the eight catalogs used here. In particular, the hydrodynamical galaxy catalog is extracted from the MassiveBlack-II (MBII) simulation, further described in Section A.1. All other catalogs are built upon the dark structures of the same gravity-only simulation, MBII DMO, which is a companion run of MBII using the same initial conditions, resolution, and box length as its hydrodynamical counterpart. The description of MBII DMO can also be found in Section A.1. The details of how each catalog is implemented can be found in the rest of the subsections of Appendix A.

In order to keep the creation of the catalogs as simple as possible, we choose to compare them only at a single snapshot (fixed redshift). This approach is sufficient to establish the functionality of the DESCQA framework. In the near future, we will expand the framework to include light-cone catalogs.

For each test, we present a summary comparison plot to compare the results from different catalogs and the validation data set, and discuss how these case studies have in turn influenced the design of the framework. We summarize these findings in Section 3.6. All plots presented here are directly taken from the output of the DESCQA framework, without any further editing. We encourage the reader to further investigate all results directly using the DESCQA Web interface,[9] which includes some tests that we did not display here and also includes plots that compare the results for each catalog with validation data separately.

3.1. *Stellar Mass Function*

Table 1. Summary of Validation Tests and the Corresponding Interim Validation Data Sets and Passing Criteria Presented in this Study.

| Test	Section	Validation Data Sets	Passing Criteria
Stellar mass function (SMF)	3.1	Li & White (2009), MBII (not shown)	$\text{CDF}(\chi^2, \text{dof}) < 0.95$
Halo mass function (HMF)	3.2	Tinker et al. (2008)	$\text{CDF}(\chi^2, \text{dof}) < 0.95$
Stellar mass–halo mass (SMHM) relation	3.3	MBII	$\text{CDF}(\chi^2, \text{dof}) < 0.95$
Projected two-point correlation function	3.4	SDSS (as in Reddick et al. 2013), MBII (not shown)	$\text{CDF}(\chi^2, \text{dof}) < 0.95$
Galaxy color distribution	3.5	SDSS	$\omega < 0.05$ for all 4 colors (shifted)

Table 2. Summary of the Synthetic Catalogs Used in the Case Studies. See Appendix A for Details.

Catalog	Abbreviation	Appendix	Model Type	Variants
MassiveBlack-II Hydrodynamic	MBII	A.2	hydrodynamic simulation	–
Improved Halo Occupation Distribution	iHOD	A.3	halo occupation distribution	–
SHAM-ADDSEDS	SHAM	A.4	abundance matching	LiWhite, MBII
Conditional Abundance Matching	CAM	A.5	abundance matching	LiWhite, MBII
Semi-Analytic Galaxies	SAG	A.6	semi-analytic model	–
Galacticus	Galacticus	A.7	semi-analytic model	–

For each of the synthetic catalogs, we calculate the stellar mass density as a function of the total stellar mass for each galaxy. The densities are derived from the number counts of galaxies in each stellar mass bin, divided by the simulation volume. These densities are compared with the stellar mass functions from the MBII hydrodynamic simulation and from Li & White (2009). Figure 3 shows a comparison between these two stellar mass functions.

Stellar masses are most commonly defined as the mass locked up in long-lived stars and stellar remnants. However, synthetic catalogs and the validation data sets may all have defined stellar mass differently, and the discrepancy cannot be homogenized by the reader interface. For the SAM models, the total stellar mass is the sum of the disk and spheroid components. For the SHAM-based models, the stellar masses correspond to the galaxy catalogs to which they were matched. On the other hand, the stellar masses used to construct the Li & White (2009) stellar mass function are taken from the New York University Value-Added Galaxy Catalog (NYU-VAGC; Blanton et al. 2005), and were derived from the KCORRECT code (Blanton & Roweis 2007). As such, they do not include the mass in stellar remnants (white dwarfs, neutron stars, etc.), which is more commonly included in the definition of stellar mass. For a Chabrier (2001) initial mass function and using stellar data from Marigo (2001) and Portinari et al. (1998), we find that the fraction of the original population mass in stellar remnants for a single-age population of age 10 Gyr is 14.6%, with 39.7% of the original population's mass remaining in stars after this time. Therefore, we shift the Li & White (2009) mass function masses by +0.136 (i.e., $\log[39.7\%/(39.7\% + 14.6\%)]$) in order to include the mass in stellar remnants.
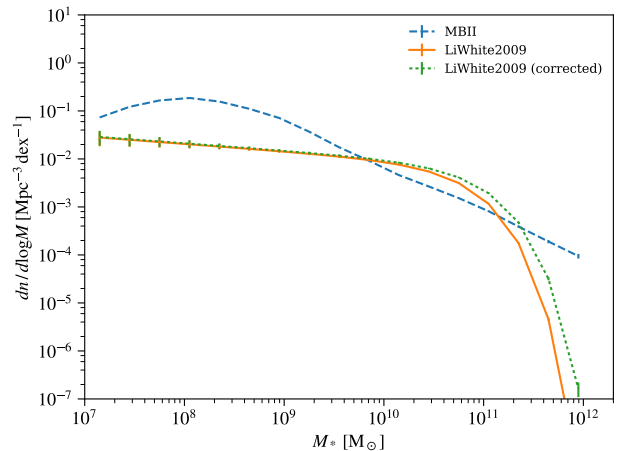
Figure 3. Stellar mass function from the MBII simulation (blue dashed line) and from Li & White (2009, orange solid line), both at $z = 0.06$. The green dotted line shows the Li & White (2009) stellar mass function but corrected for the mass in stellar remnants, as described in Section 3.1, to ensure that its stellar mass definition is consistent with that of MBII. To build the abundance-matching-based catalogs, we use both the MBII stellar mass function and the stellar-remnant–corrected measurements from Li & White (2009) as input.

Estimates of the stellar masses of galaxies also suffer from other sources of systematic error. For example, Mobasher et al. (2015) show that uncertainties arise from the template-fitting procedures used to estimate stellar masses from multi-band photometry. Although they considered other surveys, they demonstrated that systematics at the 0.1 dex level can

arise from these error sources. In the specific case of the stellar mass function from the Sloan Digital Sky Survey (SDSS), Bernardi et al. (2013) reanalyzed the SDSS photometry and concluded that there are significant systematic biases in the inferred stellar masses arising from the choice of surface brightness profile fit to the data. In their latest work, Bernardi et al. (2017) estimate that the photometric systematic errors are at the level of 0.1 dex. Here we have chosen the Li & White (2009) measurement as an example, which is significantly different from the measurement in MBII.

As mentioned in Section 2, each test must provide a summary score. For this test, the summary score is the probability for a χ^2 distribution, given the number of bins, to have a value less than the one calculated from the comparison of the catalog result to the validation data:

$$\chi^2 = \sum_{i,j} \left(\phi_i - \hat{\phi}_i \right) \left[\left(C + \hat{C} \right)^{-1} \right]_{i,j} \left(\phi_j - \hat{\phi}_j \right), \qquad (1)$$

where ϕ_i and $\hat{\phi}_i$ are the differential stellar mass number density for mass bin i calculated from the catalogs and from the validation data, respectively, C is the covariance matrix calculated from the catalog using the jackknife resampling method, and $\hat{C}$ is the covariance matrix calculated from the validation data, in which case we include only the diagonal terms. To evaluate C, we implement the jackknife resampling method by dividing the simulated box into $5^3 = 125$ smaller cubic boxes.

The criterion to pass this test is set to be a score less than 0.95 (equivalent to having a right-tail p-value larger than 0.05). When designing this test, we also notice that in most cases, the passing criterion may not apply to the full range of stellar masses. For example, the low-mass end is bound to be affected by resolution, and, depending on the user's application, this may or may not be an issue. Hence, we design the test to have a configurable validation range. We demonstrate this feature here (as the white band in Figure 4) and require the synthetic catalogs to reproduce stellar masses above $10^9 \, \mathrm{M}_\odot$. We also exclude the most massive bin when calculating the score as the last bin is dominated by cosmic variance.

Figure 4 shows the results for the stellar mass function compared to the observational measurements from Li & White (2009). The reader is encouraged to inspect the results in more detail with the help of the DESCQA Web interface. The validation data is shown in black solid circles with error bars, and the synthetic catalogs are represented by the colored lines with shaded error bands.

By construction, CAM_LiWhite and SHAM_LiWhite are almost identical to the Li & White (2009) validation data set as they are based on the abundance-matching technique, which guarantees an exact match to the input stellar mass function. For the same reason, CAM_MBII and SHAM_MBII are very close to the MBII stellar mass function. iHOD was originally tuned to fit Li & White (2009) as well, leading to good agreement in this test. More interesting are the results from the two SAM approaches. Both
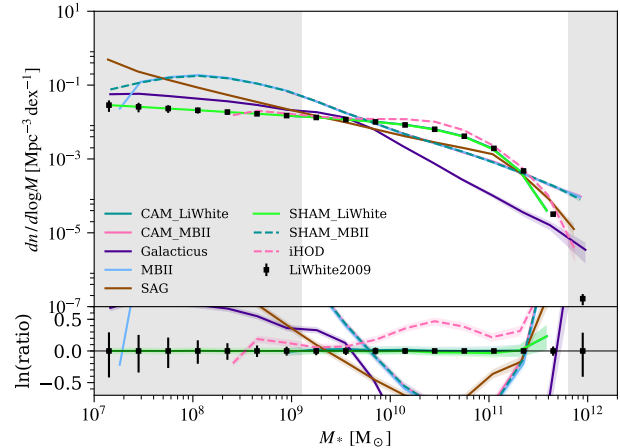

Figure 4. Stellar mass functions for all synthetic galaxy catalogs (shown as the colored lines with shaded regions) compared to the validation data set (shown as black points with error bars). In this case, the validation data set is the stellar-remnant-corrected Li & White (2009) stellar mass function. The labels in the legend indicate the method used to generate the catalog (e.g., SHAM_LiWhite is the SHAM-ADDSEDS method described in Section A.4 tuned to the Li & White (2009) stellar mass function). The shaded colored region shows the square root of the diagonal terms of the covariance matrix C of the stellar mass function, obtained from the jackknife resampling method, for the synthetic catalog of the corresponding color. For a close-in comparison, the lower panel shows the natural logarithm of the ratio to the validation data set (i.e., $\ln y/y_{\mathrm{ref}}$). To see the result for each catalog more clearly, please visit the Web interface[9]. Points within the vertical gray bands are not used to calculate the scores that appear in Figure 2.

of them overpredict the stellar mass function at low masses compared to the Li & White (2009) measurement, SAG somewhat more than Galacticus, similar to MBII. The shape of the stellar mass function for the SAMs does show a hint of a knee, similar to what is seen in the validation data set and unlike the MBII catalog, but the shape of the measurement is still not captured very well. This test demonstrates that, if a catalog user wants to impose a stringent requirement such as the one we use here, currently only SHAM-based models would pass the test due to their construction method. Hence, careful consideration is advised when designing the requirement for SMF tests.

3.2. Halo Mass Function (HMF)

The mass distribution of halos is one of the essential components of precision cosmology and occupies a central place in the paradigm of structure formation. There are two common ways to define halos in a simulation. One of these, the spherical overdensity definition, is based on identifying overdense regions above a certain threshold. The threshold can be set with respect to the (time-varying) critical density $\rho_c = 3H^2/8\pi G$ or the background density $\rho_b = \Omega_m \rho_c$. The mass M of a halo identified this way is defined as the mass
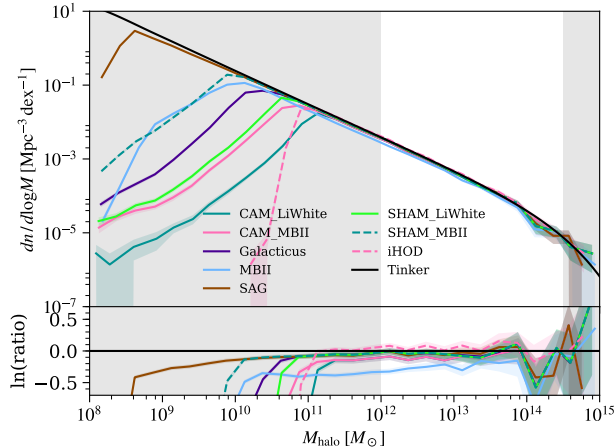
Figure 5. Same as Figure 4 but for the halo mass function. Here, the validation data set is an analytic fit from Tinker et al. (2008) and is shown as the black line.

enclosed in a sphere of radius r_Δ whose mean density is $\Delta\rho_c$, with common values ranging from 100 to 500. The other method, the friends-of-friends (FOF) algorithm, is based on finding neighbors of particles and neighbors of neighbors as defined by a given separation distance (Einasto et al. 1984; Davis et al. 1985). The FOF algorithm is essentially an iso-density estimation method (mass enclosed within a given iso-density contour). FOF halos can have arbitrary shapes, since no prior symmetry assumptions have been made; the halo mass is simply the sum of the masses of particles that are halo members.

Here, we calculate the HMF from each catalog for the distinct halos and provide a comparison to the well-established analytic fit by Tinker et al. (2008) for spherical overdensity-defined halos (M_{100c}), which is accurate at the 5–10% level at $z = 0$ for a ΛCDM cosmology. We have also implemented (not shown) the Sheth & Tormen (1999) and Bhattacharya et al. (2011) fits for the FOF halos, in addition to many other analytic mass function fits; for details, see Lukić et al. (2007). The original code was written in Fortran, and we provide a simple Python interface to include the code in DESCQA; we have made the code publicly available[10]. This test uses the same summary statistic as the SMF test as described in Section 3.1, except that the covariance of the validation data $\hat{C}$ is set to Poisson errors for the diagonal terms and zeros for the off-diagonal terms for this particular test, as the validation data is an analytic fit.

To compare the mass function for only distinct (host) halos from each catalog, this test requires two important pieces of information: (1) the halo mass associated with each galaxy and (2) whether or not the galaxy is a central galaxy. We then select only halos that are associated with central galaxies. Hence, although all the interim catalogs presented here

[10] github.com/zarija/HaloMassFunction

use the same base simulations, catalogs that assign central galaxies differently may have different HMFs. We note that the HMF we evaluate here is defined slightly differently from the usual HMF in the sense that we require the halos to host central galaxies. Distinct halos that do not contain any central galaxy are not included in these catalogs. As a result, different catalogs include very different abundances of low-mass halos, depending on their halo occupation function at the low-mass end. This effect can be clearly seen in Figure 5: above $\sim 10^{11}\,M_\odot$, all mass functions agree extremely well and follow the Tinker fit at the expected level of accuracy. Below this mass, the catalogs start to disagree. These issues should be taken into account when designing the requirement for the HMF test.

Other possible discrepancies that cannot be homogenized by the reader interface include cosmology, halo mass definitions, and halo finders. Different research groups often use different halo mass definitions and halo finders, and hence flexibility in this regard is important. Our test routine provides different fitting functions that can be chosen to match the underlying cosmology and halo mass definition used to create the synthetic catalog. Differences in halo finders, on the other hand, are more difficult to deal with. For example, the MBII hydrodynamic simulation-based catalog was generated using an FOF halo finder with linking length of $b = 0.2$, while all other synthetic catalogs that we used in this study are based on the same N-body simulation (MBII DMO) analyzed with the Rockstar halo finder and the same spherical overdensity definition. In Figure 5 we see that the MBII HMF is overall lower relative to results from the populated catalogs, in particular for the low-mass halos. This result is not only due to different halo mass definitions, but also due to the presence of baryons as indicated by the findings by Tenneti et al. (2015), where the MBII HMF was compared to the DMO mass function (for the same linking length $b = 0.2$) and a difference at the 15% level was found for halos with masses of $\sim 10^{11}\,M_\odot$. Similar results were found in the OverWhelmingly Large Simulations (OWLS; Velliscig et al. 2014), and in the Magneticum simulations (Bocquet et al. 2016). This shows that baryonic effects and halo definitions are potentially important for designing the requirements for an HMF test.

3.3. *Stellar Mass–Halo Mass (SMHM) Relation*

The SMHM relation, defined as the stellar mass of central galaxies within a distinct halo of total spherical overdensity mass, is a now well-established estimator of the efficiency of gas cooling and star formation over a wide range of halo masses. Both observational and theoretical works have shown that the efficiency of the stellar mass assembly (M_*/M_{halo}) peaks at the scale roughly corresponding to the knee of the stellar mass function and declines at smaller and larger masses. This is generally interpreted in simple terms as the efficiency of the stellar mass assembly being suppressed at the low-mass end by supernova feedback and at the high-mass end by active galactic nucleus (AGN) feedback (Silk & Mamon 2012).
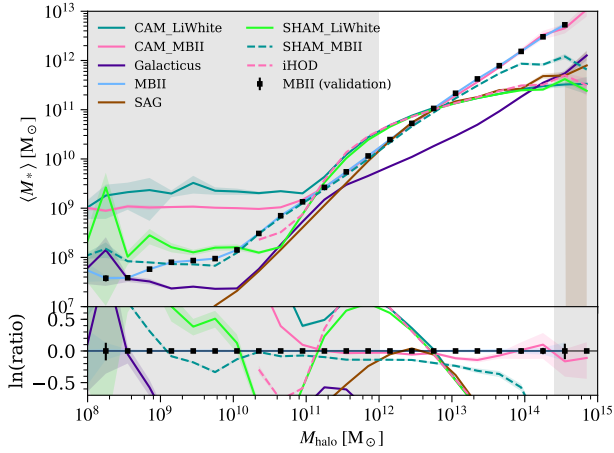
Figure 6. Same as Figure 4 but for the stellar mass–halo mass relation. Here, the validation data set (black points) is drawn from the MBII hydrodynamical simulation, and hence the MBII catalog shows a perfect match by construction.

Similar to the HMF test, the SMHM relation is not directly comparable to observational data. Not all validation tests, however, need to be tests that compare synthetic catalogs with actual observations, as some tests are designed to aid catalog users in understanding the features and characteristics of each catalog, or to provide comparisons with other results in the literature, or from other models. For example, when a user validates the SMHM relation, they might actually be verifying if the catalogs match a specific SMHM relation that is derived from a hydrodynamical simulation or inferred from a theoretical model. Moreover, the user can also validate only a certain regime of the SMHM relation, for example, to verify if the effect of AGN feedback is present in the catalog under consideration.

In the current DESCQA test implementation, we use the results from the MBII hydrodynamical simulation as an interim validation data set. As mentioned in Section A.1 for MBII, the halos were identified with an FOF halo finder with linking length $b = 0.2$. The halo definition is therefore different for the validation data set than for the synthetic catalogs, where the mass definition is the virial mass ($M_{\rm vir}$). In addition, due to baryonic effects, the halo masses from the MBII hydrodynamic simulation are lower than those from the DMO simulation (see Tenneti et al. 2015 and Section 3.2). The SMHM relation uses only distinct halos (host halos) and excludes subhalos. As such, most of the caveats that apply in the HMF test would also apply here. This test also uses the same summary statistic as the stellar mass function test as described in Section 3.1, and the same validation range as the HMF test (Section 3.2).

Figure 6 shows the results from MBII and the catalog results with error bands. The MBII result is trivially perfect by construction because it is a self-comparison. SHAM_MBII and CAM_MBII perform quite well over most of the mass range as these models were tuned to the MBII stellar mass

function. However, at the high-mass end, the SMHM relation from SHAM_MBII flattens out due to the constant scatter used in the abundance-matching technique, while MBII's SMHM relation exhibits a much smaller scatter in stellar mass at high halo mass. SHAM_LiWhite and CAM_LiWhite perform reasonably well over the intermediate-mass range where the MBII and Li & White stellar mass function are closest. The overprediction of MBII at the low- and high-ass end compared to Li & White is reflected in the discrepancy seen in SHAM_LiWhite and CAM_LiWhite. Galacticus is overall lower than MBII but has the correct rise at the high-mass end. The SAG catalog underpredicts the SMHM relation compared to the MBII test for the low-mass halos. iHOD overall fits reasonably well though the results are worse at extreme mass values. Note that, except for MBII (which is a self-comparison), none of the catalogs passes the current validation criterion. This indicates that more thoughtful criteria and more realistic validation data sets (e.g., ones derived from empirical models) should be adopted if catalog users view the SMHM relation as an essential measurement that catalogs must reproduce.
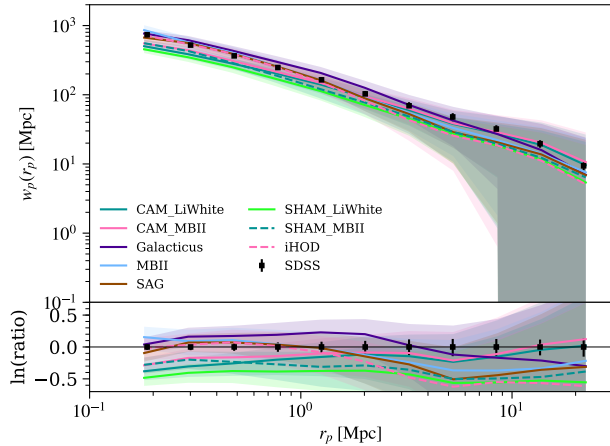
3.4. *Projected Two-point Correlation Function*

The projected galaxy two-point auto-correlation function, $w_p(r_p)$, is one of the most-used clustering statistics for testing both cosmology and galaxy formation models. Here we describe our test to compare $w_p(r_p)$ among different synthetic catalogs and against observational and simulated data. Since our interim synthetic catalogs are given at a single epoch, we calculate $w_p(r_p)$ using the thin-plane approximation. We use the catalogs at one epoch and then add redshift space distortions along one spatial axis (z-axis). We then calculate the projected pair counts, with a projection depth of $\pm 40\,h^{-1}$Mpc. We assume periodic boundary conditions for all three spatial axes.

This test also uses the same summary statistic as the stellar mass function test described in Section 3.1, though the evaluation of the covariance for the catalog, C, is slightly different, and we include the full covariance of the validation data, $\hat{C}$. We estimate the sample variance of $w_p(r_p)$ using the jackknife technique. We divide the projected 2D plane ($x - y$ plane) into 10×10 smaller regions, with each region having an area of $(10\,h^{-1}$Mpc$)^2$. We then re-evaluate the $w_p(r_p)$ when removing one region at a time. The code that calculates $w_p(r_p)$ and its jackknife variance is publicly available[11].

In this paper, we compare the $w_p(r_p)$ from each catalog for all galaxies that have a stellar mass larger than $10^{9.8}\,h^{-2}$M$_\odot$. We use two interim validation data sets for this test: the $w_p(r_p)$ calculated from the MBII hydrodynamical simulation and the $w_p(r_p)$ from SDSS as presented in Reddick et al. (2013). This measurement was made on the volume-limited samples from the NYU-VAGC catalog (Blanton et al. 2005), based on Data Release 7 from the SDSS (Padmanabhan et al. 2008; Abazajian et al. 2009).

---

[11] Module 'CorrelationFunction' in bitbucket.org/yymao/helpers

Figure 7. Same as Figure 4 but for the projected two-point correlation function for all galaxies down to a stellar mass of $1.28 \times 10^{10} \, \mathrm{M}_\odot$ ($= 10^{9.8} \, h^{-2} \mathrm{M}_\odot$) . Here the validation data set (black points) is from SDSS as presented in Reddick et al. (2013). Visit the Web interface to see the individual error bars more clearly.

In Figure 7 we show the results of comparisons with the SDSS data being used for validation; the reader is encouraged to inspect the comparisons with MBII as the validation data directly on our Web interface. Overall, the agreement between the catalogs and SDSS data is rather good, and most catalogs pass our preliminary validation metric. Most synthetic catalogs underpredict the small-scale clustering when compared with SDSS data, though they are still within the 2σ errors. Due to the small volume of our simulation box, the jackknife sample variance dominates the error budget. The data are likely to better distinguish between the same models if they are applied to a larger cosmological volume. Hence, one should consider including the volume of the catalogs as part of the evaluation for a $w_p(r_p)$ test.

3.5. *Galaxy Color Distribution*

We also include a test of how well the synthetic galaxy color distributions compare to the observed colors of galaxies. In principle, this test should be done with light-cone catalogs that cover the same redshift range as the observed data set, with the same set of cuts on observed properties. However, for proof of concept, we present it using our current catalogs at a single epoch, $z = 0.0625$ (except for the SAG catalog, which is at $z = 0$, and the Galacticus catalog, which is at $z = 0.05$).

We determine the color distributions in these catalogs for comparison with our validation data set — measurements of the *ugriz* colors of a volume-limited sample of $0.06 < z < 0.09$ galaxies from SDSS DR13 (Albareti et al. 2017). In the future, when light-cone synthetic catalogs are included in the framework, we will incorporate a broader range of SDSS galaxies, as well as objects with deeper imaging, e.g., from CFHTLS (Hudelot et al. 2012) or DES (Collaboration 2005), and spectroscopy, e.g., from GAMA (Driver et al. 2011),

DEEP2 (Newman et al. 2013), and DESI (DESI Collaboration et al. 2016).

In order to compare SDSS colors with synthetic galaxy colors, we use the SDSS apparent magnitudes to construct K-corrected absolute magnitudes. First, we select SDSS galaxies in the redshift range $0.06 < z < 0.09$, where the variation of the distribution of K-corrected colors with redshift is small, and correct for Galactic extinction (we implicitly assume that the color evolution for $0 < z < 0.09$ is negligible when comparing to the single-epoch catalogs). We then use the KCORRECT code of Blanton & Roweis (2007) to find the rest-frame spectral energy distributions (SEDs) and obtain K-corrected absolute magnitudes for each SDSS galaxy (e.g., M_i for i-band absolute magnitude). Since different catalogs include galaxy colors that are K-corrected to different redshifts and this difference cannot be eliminated by the reader interface, this test K-corrects the SDSS data to the same redshift that each of the catalogs uses for its passbands.

To minimize the effects of incompleteness in the validation data set, we construct a volume-limited sample by applying a cut in r-band absolute magnitude $M_r < M_{r,\mathrm{max}}$, where $M_{r,\mathrm{max}}$ is chosen to be the value of the 85th percentile of the SDSS M_r distribution in a narrow redshift bin at $0.089 < z < 0.09$. The same $M_r < M_{r,\mathrm{max}}$ cut is also applied to the synthetic catalogs. We then compare the K-corrected colors of the volume-limited samples from SDSS and from the synthetic galaxy sample.

To obtain a quantitative estimate of the level of difference between the SDSS and synthetic color distributions, we calculate the two-sample Cramér-von Mises (CvM) statistic (Anderson 1962). The CvM test is a nonparametric test for whether multiple data sets are drawn from the same probability distribution, similar to the Kolmogorov–Smirnov (K–S) test. However, instead of only looking at the maximal difference in the cumulative distribution function (CDF), as is done in the K–S test, the CvM test statistic ω (defined below), calculates the average L^2 distance across the entire CDF. As a result, it is more sensitive to differences in the tails of the distribution than the K–S statistic, and constrains the closeness of two CDFs at every point along them.

The CvM statistic is calculated from the formula

$$\omega^2 = \int_{-\infty}^{+\infty} \left(F_1(x) - F_2(x)\right)^2 \mathrm{d}H(x) \qquad (2)$$

where $F_1(x)$ and $F_2(x)$ are the CDFs of each sample estimated from the data, $H(x) = (n_1 F_1(x) + n_2 F_2(x))/N$, and n_1 and n_2 are the numbers of objects in the two samples, with $N = n_1 + n_2$. In our case, $F_1(x)$ and $F_2(x)$ are the CDFs for one SDSS color (e.g., $g - r$) and the equivalent synthetic color; ω provides a measure of the RMS difference between these two CDFs. As an example, Figure 8 shows the CDFs of the color distribution from SDSS and one of the synthetic catalogs. To remove potential zero-point offsets between SDSS and the synthetic catalogs (whether due to photometric zero-point uncertainties or issues with K-corrections), we apply a constant shift to the synthetic galaxy colors so that their median matches the median SDSS color; we calculate ω for both unshifted and
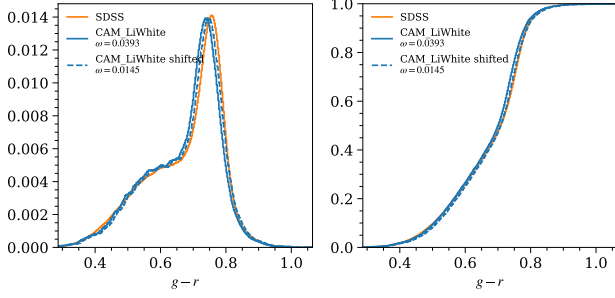
Figure 8. Comparison of the color distributions of SDSS galaxies and of the synthetic colors from the CAM_LiWhite catalog. The left panel shows the probability density functions and the right panel shows the cumulative distribution functions for the $g-r$ colors in each sample. The orange and blue solid curves are the distribution of SDSS colors and synthetic colors, respectively. The dashed blue curve is the distribution of the synthetic color after applying a constant shift to match its median color value to that of SDSS. This should remove the effect of zero-point offsets if they are the main contributors to differences in the color distribution. The definition of ω can be found in Equation 2.

shifted colors. The current criterion for a synthetic catalog to pass the color test is that the ω calculated from the shifted colors must be smaller than 0.05 for all four SDSS colors ($u-g$, $g-r$, $r-i$, and $i-z$), i.e., the RMS difference between the color CDFs must each be smaller than this threshold. This criterion is very stringent and may not reflect the final actual requirements, but we do expect that LSST will have stringent requirements on the distribution and evolution of galaxy colors to mitigate systematic errors in photometric redshifts.

A set of summary plots for all catalogs with available colors is shown in Figure 9. The SHAM catalogs agree well with SDSS in $i-z$ but are redder in $u-g$, $g-r$ and $r-i$. The CAM catalog agrees well with SDSS in $g-r$ and $r-i$, but not as well in $u-g$ and $i-z$, although the difference in $i-z$ might be due to a zero-point offset. Different releases of SDSS, and moderately different redshift ranges, were used in the production of the abundance-matching-based catalogs and the validation catalog, and these two factors likely contribute to the differences. The two stellar mass functions (from MBII and from Li & White 2009) yield only a very small difference in the color distributions. The two catalogs from semianalytic models do not produce very realistic color distributions, with SAG color distributions exhibiting a stronger bimodality than SDSS and Galacticus having long blue tails. We see that some of the features in the color distributions are not captured by the summary statistics. Although the framework requires each test to report a summary statistic for the ease of quick overall comparison, for tests like the color distribution test, the figures are essential components as they provide more information for the catalog users.

3.6. *How These Case Studies Influence Our Design*
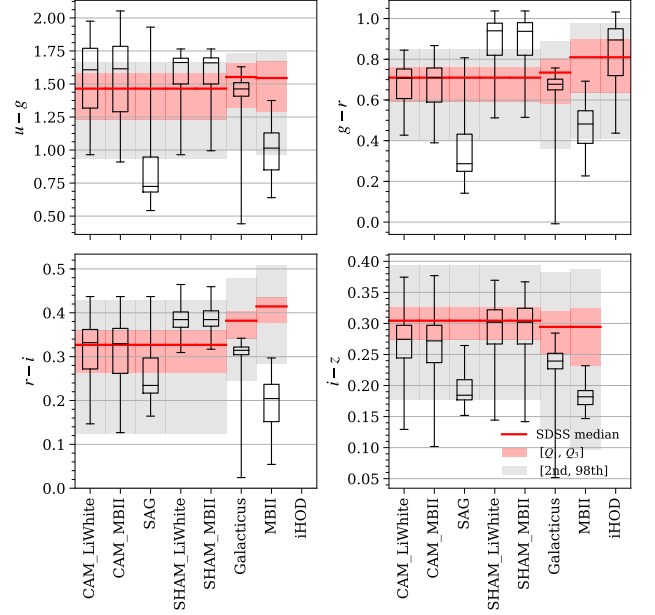
Figure 9. Summary plot illustrating the differences in the color distributions for those synthetic catalogs that provide color information. The results are shown in the form of box plots for the catalogs and bands for the validation data set (SDSS). For the validation data set, the red line marks the median, the red band shows the interquartile range, and the gray band shows the range between the 2.5th and 97.5th percentiles. The synthetic catalogs are represented by the box plots, with the boxes showing the median and the interquartile range and the ends of the extended lines showing the 2.5th and 97.5th percentiles. The SDSS color distributions vary because for each synthetic catalog, the SDSS colors are K-corrected to the redshift corresponding to a given catalog's snapshot. The iHOD catalog only provides the $g-r$ color.

As mentioned above, our case studies have helped us identify several features that are particularly useful for a validation framework like DESCQA. Here we summarize these features.

1. *Uniform interfaces for both reading catalogs and executing tests.* To carry out the validation tests presented in our case studies, we can easily see the necessity of the two main components of the DESCQA framework: the reader and test interfaces. By providing a uniform interface, it enables the users to access different catalogs in a uniform way and also standardizes the necessary elements of a validation test.

2. *Allowing absent quantities.* Since each validation test only accesses a subset of quantities, the framework should not impose a global set of required quantities for all catalogs. Each catalog will be validated by the tests for which it contains the quantities needed. Thus, the reader interface should provide a method for checking available quantities.

3. *Documenting the intrinsic differences in quantity definitions.* Some quantities may be defined differently (e.g., halo mass, magnitudes) in different catalogs. In the cases where these differences cannot be homogenized by the reader interface, the difference should be recorded in the metadata, and be exposed to the tests through the reader interface. It is up to each test developer to decide how to deal with these intrinsic differences.

4. *Adaptive tests.* In the cases where intrinsic differences that cannot be homogenized exist among the catalogs, sometimes it is computationally more efficient for the test to change its configuration on the fly to adapt to each catalog. For example, in our case studies, the HMF validation data set is computed analytically and hence can be tuned to different redshifts to match the catalogs. Similarly, the color distribution test also applies K-corrections to the validation data set to match the specification in the catalogs.

5. *Configurable tests.* For a given test, it may be desirable to have some variants that use different validation data sets, passing criteria, or validation ranges. Hence, the validation test interface should provide a convenient method to allow quick changes of these settings. In practice, these settings can be specified in a configuration file that is read in by the test interface.

6. *Providing both numerical and visual results.* A potential concern with a highly automated validation system is that some important issue is buried under a simple pass-or-fail result. Hence, we encourage the validation test developers to create plots in their test implementation, and we have designed an automated framework to manage such plots and display them on the Web interface. The framework also collects a numerical score from the tests to present in the summary view (Figure 2) so that users can spot any potential issues more quickly.

4. CONCLUSION AND OUTLOOK

In this paper, we have presented DESCQA, a framework that enables the automated validation of synthetic sky catalogs. The major aims of the framework are (1) to provide simulators an easy interface to test the quality of their catalogs and (2) to provide the LSST DESC and larger community a platform that enables them to easily choose synthetic catalogs that best fulfill their needs to test their analysis pipelines. The necessity of this framework arises from the fact that, with the wide variety of cosmological investigations possible with LSST—weak to strong lensing, cluster cosmology, LSS measurements, and supernova distances—no one single synthetic catalog will be optimal for every task. For example, obtaining large volumes for LSS investigations will clearly only be possible with limited mass resolution, while on the other hand, photo-z tests do not necessarily require large volumes, but rather excellent modeling of the color distribution. Hence, a framework that can test a wide range of synthetic sky catalogs against a large set of different target requirements is essential in systematically preparing for LSST science.

The goal of the DESCQA framework is to minimize the burden on both the catalog creators and users when they deal with an inhomogeneous set of catalogs and tests. We have designed common APIs for both accessing and testing the synthetic sky catalogs with the Python Programming Language, and have also built a Web interface for ease of comparison. Nevertheless, the fundamental challenge here is to design a framework that can actually respond to the scientific needs of catalog validation. To meet this challenge during the development of the framework, we have selected a set of realistic synthetic sky catalogs and validation tests to test and improve our framework.

Although these interim synthetic sky catalogs and validation tests are not necessarily the final product or requirements that LSST DESC will eventually adopt, they have provided useful insights into questions such as how to homogenize a diverse set of synthetic sky catalogs and how to design meaningful validation tests, as we have summarized in this paper. Thanks to the use of these realistic trial catalogs and tests during our development process, we have already identified several needed improvements for upcoming LSST DESC Data Challenges. For example, although the code itself is maintained in a GitHub repository[2] and all the outputs are stored on the NERSC filesystem, a more rigorous catalog and test version control system for the framework is still needed. We also need to improve the ability of the framework to process even larger sky catalogs efficiently and to enable a convenient way to download catalogs of interest (currently only available to LSST DESC members in our NERSC LSST project space).

Another major step is to include light-cone catalogs, which is essential for realistic comparison with photometric data. In addition, the set of validation tests will also be considerably extended to cover a large range of possible LSST DESC projects. During our development process, it became clear that more consideration is needed when designing the catalog requirements. In particular, the validation tests need to carefully handle the intrinsic differences between catalogs that cannot be homogenized by the framework; we have highlighted many issues of this kind here, as summarized in Section 3.6. With this study and the implementation of the DESCQA framework, we have made an important step toward the full utilization of the wide variety of synthetic sky catalogs.

APPENDIX

A. SIMULATIONS AND SYNTHETIC SKY CATALOGS

Here we describe the set of synthetic catalogs used in our case studies (Section 3). We first provide a description of the MassiveBlack-II simulations (Section A.1), for both the hydrodynamical and gravity-only runs. In the following subsections, we discuss the six different methods (one hydrodynamical simulation, one HOD-based model, two SHAM-based models, and two SAMs) to generate the eight synthetic catalogs used in Section 3. All of the synthetic methods used to generate the catalogs are applied to the same dark matter structures (i.e., halos and merger trees) of the MBII DMO simulation. A brief summary of these catalogs is listed in Table 2.

A.1. *The MassiveBlack-II (MBII) Simulations*

MBII is a state-of-the-art, high-resolution cosmological hydrodynamic simulation (Khandai et al. 2015) of structure formation with subgrid model physics described in detail below. A companion simulation, MBII DMO, uses the same volume, resolution, cosmological parameters, and initial conditions but only takes gravitational forces into account (Tenneti et al. 2015). Both of these simulations have been performed in a cubic periodic box of size 142.45 Mpc on a side using the cosmological TreePM Smooth Particle Hydrodynamics (SPH) code P-GADGET, which is a hybrid version of the parallel code, GADGET2 (Springel et al. 2005), that has been upgraded to run on petascale supercomputers. The total number of dark matter particles in both simulations is 1792^3 with an equal (initial) number of gas particles in the hydrodynamical simulation.

Table 3. Simulation parameters: Box size (L_{box}), force softening length (ϵ), number of particles (N_{part}), mass of dark matter particle (m_{DM}), and mass of gas particle (m_{gas}).

Parameters	Hydrodynamical	Dark Matter-Only
L_{box} (Mpc)	142.45	142.45
ϵ (kpc)	2.64	2.64
N_{part}	2×1792^3	1792^3
m_{DM} ($M_\odot$)	1.6×10^7	1.9×10^7
m_{gas} ($M_\odot$)	3.1×10^6	–

The cosmological parameters are chosen for consistency with WMAP7 (Komatsu et al. 2011), with amplitude of matter fluctuations set by $\sigma_8 = 0.816$, the scalar spectral index $n_s = 0.96$, matter density parameter $\Omega_m = 0.275$, cosmological constant density parameter $\Omega_\Lambda = 0.725$, baryon density parameter $\Omega_b = 0.046$ (in MBII), and Hubble parameter $h = 0.702$. Table 3 lists the box size (L_{box}), force softening length (ϵ), total number of particles including dark matter and gas (N_{part}), mass of dark matter particles (m_{DM}), and mass of gas particles (m_{gas}) for the two simulations. The major results from the hydrodynamical simulation, MBII, are available in Khandai et al. (2015). In addition to gravity and SPH, MBII also includes the physics of a multiphase interstellar medium model with star formation (Springel & Hernquist 2003), and black hole accretion and feedback (Springel et al. 2005; Di Matteo et al. 2012). Radiative cooling and heating processes are included (as in Katz et al. 1996), as is photoheating due to an imposed homogeneous ionizing ultraviolet background.

For the analysis of the MBII DMO (gravity-only) simulation, we use ROCKSTAR[12], a six-dimension phase-space halo finder, to identify halos and subhalos (Behroozi et al. 2013a), and use CONSISTENT TREES[13] to build the halo merger trees (Behroozi et al. 2013b). The halo catalogs and merger trees are available on the NERSC filesystem and will be made publicly available.

The halos are defined with spherical overdensity at virialization (Bryan & Norman 1998). At $z = 0$, this virial overdensity (Δ_{vir}) is approximately 97.7 for this cosmology. Subhalos are defined as halos whose centers are within the virial radius of any other larger halo. When building the merger trees, we skip some very close-timed snapshots and use in total 177 snapshots from the simulation, with the earliest snapshot at $z = 20$. In the halo catalogs that we provide for the synthetic catalog creators, each halo or subhalo in the catalog must have at least 20 particles associated with it. The catalog creators may apply more stringent halo mass cuts if required.

A.2. *MBII Galaxy Catalog*

The MBII hydrodynamical simulation was analyzed by applying an FOF procedure to dark matter particles, with a dimensionless linking length of $b = 0.2$. Gas, star, and black hole particles were then associated to their nearest dark matter particles.

---

[12] bitbucket.org/gfcstanford/rockstar (commit #ca79e51)

[13] bitbucket.org/pbehroozi/consistent-trees (commit #2ddc70a)

Subhalos were identified with the subhalo finder SUBFIND (Springel et al. 2001). The galaxy stellar mass is the total mass of all the star particles bound to the subhalo. The SED of star particles in MBII are generated using the PEGASE.2 stellar population synthesis code (Fioc & Rocca-Volmerange 1997, 1999), based on the ages, masses, and metallicities of the stars, with the assumption of a Salpeter initial mass function. Nebula (continuum and line) emissions are also added to each star particle SED, along with a correction for absorption in the intergalactic medium using the standard Madau et al. (1996) prescription. The SED of a galaxy is then obtained by summing the SEDs of all the star particles in the galaxy, from which SDSS-band luminosities are calculated, based on the respective filter. More details can be found in Khandai et al. (2015).

Here we use the SUBFIND halo catalogs for the MBII hydrodynamical simulation as those match the published version, but we use the ROCKSTAR–CONSISTENT TREES catalogs for the MBII DMO run as they provide more robust merger histories (Avila et al. 2014). Therefore, we caution the reader that when comparing results in Section 3, it should be kept in mind that some differences in the tests that rely on halo masses are to be expected because the MBII hydrodynamic and DMO runs use different halo finders and different mass definitions.

A.3. *Improved Halo Occupation Distribution (iHOD) Model*

The iHOD model (Zu & Mandelbaum 2015, 2016, 2017) aims to provide a *probabilistic* mapping between halos and galaxies, assuming that the enormous diversity in the individual galaxy assembly histories inside similar halos would reduce to a stochastic scatter about the *mean* galaxy-to-halo connection by virtue of the central limit theorem. Therefore, the key is to derive the conditional probability distribution of host halos at fixed galaxy properties, $P(\tilde{\mathbf{h}} \mid \tilde{\mathbf{g}})$, where $\tilde{\mathbf{g}}$ and $\tilde{\mathbf{h}}$ are the corresponding vectors that describe the most important sets of properties. For $\tilde{\mathbf{g}}$, those properties can be the stellar mass, central/satellite dichotomy, color, velocity, and alignment, and for $\tilde{\mathbf{h}}$ the dark matter mass, concentration, and tidal environment.

Building on the global HOD parameterization of Leauthaud et al. (2011), Zu & Mandelbaum (2015) developed the iHOD formalism to solve the mapping between galaxy stellar mass and halo mass, i.e., $P(M_h|M_*)$, using the spatial clustering and the galaxy–galaxy lensing of galaxies in SDSS. Compared to the traditional HOD methods, iHOD can include ~84% more galaxies while taking into account the stellar mass incompleteness of galaxy samples in a self-consistent fashion.

In order to link galaxy colors to the underlying dark matter halos and constrain galaxy quenching, Zu & Mandelbaum (2016) extended the iHOD model to describe galaxies of different $g-r$ colors, i.e., $\tilde{\mathbf{g}} \equiv \{M_*, g-r\}$, by considering two popular quenching scenarios: (1) a "halo" quenching model in which halo mass is the sole driver for turning off star formation in both central and satellite galaxies and (2) a "hybrid" quenching model in which the quenched fraction of galaxies depends on their stellar mass while the satellite quenching has an extra dependence on halo mass. Zu & Mandelbaum (2016) found that the halo quenching model provides significantly better fits to the clustering and galaxy–galaxy lensing of blue galaxies above stellar masses of $10^{11}\,\mathrm{M}_\odot$. The best-fitting iHOD quenching model of $P(M_h|M_*, g-r)$ also correctly predicts the average halo mass of the red and blue centrals, showing excellent agreement with the direct weak-lensing measurements of central galaxies (Mandelbaum et al. 2016). The iHOD modeling of galaxy colors provides strong evidence that the physical mechanism that quenches star formation in galaxies above stellar masses of $10^{10}\,\mathrm{M}_\odot$ is tied principally to the masses of their dark matter halos rather than the properties of their stellar components or halo age.

Zu & Mandelbaum (2017) further demonstrated that the iHOD model provides an excellent description of the environmental dependence and conformity of galaxy colors observed in SDSS. The current iHOD model, as constrained by the clustering and galaxy–galaxy lensing of red and blue galaxies in SDSS, also correctly reproduces the stellar mass functions within each color observed by SDSS. For the purpose of this paper, we populate the halo catalog using the best-fit parameters from Zu & Mandelbaum (2017).

A.4. *SHAM-ADDSEDS Model*

This synthetic catalog is a combination of the SHAM (see e.g., Kravtsov et al. 2004; Vale & Ostriker 2004, 2006; Conroy et al. 2006) and the ADDSEDS algorithm (R. H. Wechsler et al. 2018, in preparation; J. DeRose et al. 2018, in preparation) which is explained in more detail below. The SHAM technique is a generic scheme to connect one galaxy property (e.g., stellar mass or luminosity) with one (sub)halo property (e.g., virial mass) by assuming an approximately monotonic relation between these two properties. The two properties are matched at the same cumulative number density, and the resulting galaxy catalog, by explicit construction, preserves the input stellar mass (or luminosity) function.

Common choices of the matching (sub)halo properties include M_{peak} and V_{peak}, which are the mass and the maximal circular velocity, respectively, at their peak values along the accretion history of the (sub)halo. Scatter between the galaxy and (sub)halo properties can be introduced into the matching procedure. For a constant log-normal scatter in stellar mass or luminosity, one can follow the procedure in Behroozi et al. (2010): first deconvolve the scatter from the stellar mass (or luminosity) function, match the two properties, and finally add a random log-normal scatter in the catalog.

To generate the synthetic catalogs used in this work, we use a publicly available SHAM code,[14] which follows the procedure we outlined above, to match the (sub)halo V_{peak} function to the stellar mass functions from Li & White (2009) and MassiveBlack-II,

---

[14] bitbucket.org/yymao/abundancematching

respectively. In both cases, we adopt a constant log-normal scatter of 0.15 dex in stellar mass (see, e.g., Reddick et al. 2013; Gu et al. 2016; Lehmann et al. 2017). Once we obtain the stellar mass for each synthetic galaxy, we also assign an absolute r-band magnitude by simply matching the stellar mass function to the luminosity function of Bernardi et al. (2013).

We then further generate multiband magnitudes using the ADDSEDS algorithm. For each synthetic galaxy, we measure the projected distance to its fifth nearest neighbor. We then bin galaxies in absolute r-band magnitude and rank-order them in terms of this projected distance. We compile a training set consisting of the magnitude-limited spectroscopic SDSS DR6 VAGC cut to $z < 0.2$ and local density measurements from Cooper (2006). This training set is rank-ordered the same way as the simulation. Each simulated galaxy is assigned the SED from the galaxy in the training set with the closest density rank in the same absolute magnitude bin. The SED is represented as a sum of templates from Blanton & Roweis (2007), which can then be used to shift the SED to the correct reference frame and generate magnitudes in SDSS bandpasses. In our case, we assume that all of our galaxies are at redshift $z = 0$, and the magnitudes are K-corrected to $z = 0$.

A.5. *Conditional Abundance Matching (CAM) Model*

This synthetic galaxy catalog is created using the CAM technique (Hearin et al. 2014). We provide a brief description of the catalog and technique here, and point the reader to Campbell et al. (2017) for details. The CAM technique is similar to the SHAM technique (see Section A.4), but further assigns a secondary galaxy property (e.g., specific star formation rate) according to a secondary (sub)halo property (e.g., mass accretion history). In this work, the primary galaxy property is the stellar mass, which is assigned to the $V_{\rm peak}$ of (sub)halos using a simple SHAM technique. We include a fixed log-normal scatter $\sigma_{\rm SMHM}$ in stellar mass at fixed $V_{\rm peak}$. We deconvolve the scatter from the stellar mass function before matching.

As for the secondary properties, specific star formation rates (sSFRs) are assigned to galaxies such that there is a correlation between sSFR and $a_{1/2}$ at fixed stellar mass, where $a_{1/2}$ is the scale factor at which the (sub)halo reached half its peak mass. This step requires drawing from a $P(\mathrm{sSFR}|M_*)$ distribution, here based on the Main Galaxy Sample from the SDSS Data Release 7 (DR7; Padmanabhan et al. 2008; Abazajian et al. 2009), specifically a re-reduction of DR7 in the form of the NYU-VAGC LSS sample (Blanton et al. 2005). sSFRs are taken from the MPA-JHU DR7 catalog based on the method of Kauffmann et al. (2003). The strength of this correlation is encoded in a rank-order correlation statistic, $\rho_{\rm sSFR}$, which can generally take values in the range $[-1, 1]$, perfect anticorrelation to perfect correlation. The result is that our model has two explicit parameters that can be tuned: $\sigma_{\rm SMHM}$ and $\rho_{\rm sSFR}$. For this study, we use fiducial values of 0.15 and 1.0, respectively. Absolute magnitudes in five bands (*ugriz*), K-corrected to $z = 0$, are associated with each galaxy in the synthetic catalog by selecting a galaxy in the NYU-VAGC LSS catalog with similar stellar mass and sSFR, and carrying over each of the magnitudes. In this way, the conditional stellar mass, sSFR, and colors are preserved in the synthetic catalog.

A.6. *Semi-analytic Galaxies (SAG) Model*

The Semi-analytic Galaxies (SAG) approach is based on the model developed by Springel et al. (2001), which, as is usual with semi-analytic models, combines merger trees extracted from a gravity-only cosmological simulation with a set of coupled differential equations for the baryonic processes taking place within these merger trees as time evolves. The most up-to-date version of the SAG model has been further improved from the model described in Gargiulo et al. (2015); however, those improvements are not used in the current study in order to achieve good performance.

The model used here assumes that the hot gas in dark matter halos is isothermally distributed, with an initial mass calculated using the cosmic baryon fraction. This hot gas cools to form an exponential disk where stars form quiescently. Gas cooling takes place only in central galaxies (i.e., the galaxy residing in the main subhalo of a given dark matter halo); the hot gas atmosphere is stripped instantly when a galaxy becomes a satellite (strangulation scheme). Stars also form through starbursts, which can be triggered by mergers and disk instabilities contributing to bulge formation. In that case, the gas is transferred to a reservoir that is continuously consumed by star formation in a given timescale. This reservoir can be modified by successive mergers and instabilities (Gargiulo et al. 2015). Bursts are the main channel for supermassive black hole growth. Gas accretion onto these objects produces AGN feedback (Lagos et al. 2008). The stars formed in each star formation event produce a number of supernovae depending on the selected initial mass function. These supernovae reheat the cold gas transferring it back to the hot phase (supernovae feedback). Chemical elements produced by stellar winds and supernova explosions (both core-collapse and Type Ia supernovae) are tracked in different baryonic components, taking into account the lifetime of stars (Cora 2006). The current chemical implementation has been updated with new stellar yields (Gargiulo et al. 2015). Stellar luminosities and colors are modeled using the Bruzual & Charlot (2003) stellar synthesis models for the stellar populations generated in model galaxies (at each integration time step).

SAG depends on a number of parameters. These are tuned using the Particle Swarm Optimization technique (Ruiz et al. 2015). For this particular run, we consider a set of best-fitting parameters obtained from the application of SAG to one of the MultiDark gravity-only cosmological simulations (MDPL2; Klypin et al. 2016) with a cubic volume of $(1475.6\,\mathrm{Mpc})^3$ and *Planck* cosmological parameters (Planck Collaboration et al. 2014). The observational constraints used for this calibration are the stellar mass function and the black hole–bulge mass relation, both at $z = 0$. For the former, we adopt the compilation of data used by Henriques et al. (2015), while for the latter we combine the data sets from McConnell & Ma (2013) and Kormendy & Ho (2013).

A.7. *Galacticus*

Galacticus (Benson 2012) is another semi-analytic model of galaxy formation that we employ in this paper. Galacticus models the baryonic physics of galaxy growth within an evolving, merging hierarchy of dark matter halos. Baryonic processes (including gas cooling and inflow, star formation, stellar and AGN feedback, and galaxy merging) are described by a collection of differential equations that are integrated to a specified tolerance along each branch of the merger tree. Also included are instantaneous transformations, such as starbursts, that are associated with merger events. Galacticus is designed to be fully modular, allowing the physical components and processes in galaxies and halos to be interchanged easily. This permits the possibility of running everything from simplistic models based on empirical fitting functions for the rates of key processes through to fully physical models incorporating detailed treatments of chemical enrichment, galaxy and halo dynamics, black hole accretion disks, and feedback.

The output of Galacticus is a catalog of galaxies at all redshifts that includes both physical properties, such as stellar masses, sizes, and morphologies, and observational properties, such as luminosities in any specified bandpass filter. The luminosities are computed by convolving the star-formation history for each galaxy with spectra obtained from stellar population synthesis models. For this paper, we computed rest-frame luminosities in SDSS *ugriz* filters. Note that, for this paper, in order to ensure consistency of the input halo catalogs with the other synthetic methods, we disable a standard convergence-testing feature in Galacticus, which ensures that they reach sufficient temporal and mass resolution.

The parameters of the model are constrained through either particle swarm optimization or Markov Chain Monte Carlo techniques to match a wide variety of data on the galaxy population, including the stellar mass function from $z = 0$ to $z = 5$, the $z = 0$ H I mass function, the galaxy size–mass relation, and the two-point correlation function of galaxies. The resulting models can, in principle, accurately reproduce key observables of the galaxy population across a wide range of redshifts. However, the parameters also depend on simulation details such as the mass resolution, and so, in practice, the parameters need to be tuned for each simulation. For this paper, we used a default parameter set obtained by tuning on Press–Schechter trees, and so we do not expect to find good agreement between the Galacticus catalog and the validation data.

REFERENCES

Abazajian, K. N., Adelman-McCarthy, J. K., Agüeros, M. A., et al. 2009, ApJS, 182, 543

Albareti, F. D., Allende Prieto, C., Almeida, A., et al. 2017, ApJS, 233, 25

Anderson, T. W. 1962, Ann. Math. Statist., 33, 1148

Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33

Avila, S., Knebe, A., Pearce, F. R., et al. 2014, MNRAS, 441, 3488

Baugh, C. M. 2006, Reports on Progress in Physics, 69, 3101

Behroozi, P. S., Conroy, C., & Wechsler, R. H. 2010, ApJ, 717, 379

Behroozi, P. S., Wechsler, R. H., & Wu, H.-Y. 2013a, ApJ, 762, 109

Behroozi, P. S., Wechsler, R. H., Wu, H.-Y., et al. 2013b, ApJ, 763, 18

Benson, A. J. 2010, PhR, 495, 33

—. 2012, NewA, 17, 175

Benson, A. J., Bower, R. G., Frenk, C. S., et al. 2003, ApJ, 599, 38

Benson, A. J., Cole, S., Frenk, C. S., Baugh, C. M., & Lacey, C. G. 2000, MNRAS, 311, 793

Berlind, A. A., & Weinberg, D. H. 2002, ApJ, 575, 587

Bernardi, M., Meert, A., Sheth, R. K., et al. 2017, MNRAS, 467, 2217

—. 2013, MNRAS, 436, 697

Bhattacharya, S., Heitmann, K., White, M., et al. 2011, ApJ, 732, 122

Blanton, M. R., & Roweis, S. 2007, AJ, 133, 734

Blanton, M. R., Schlegel, D. J., Strauss, M. A., et al. 2005, AJ, 129, 2562

Bocquet, S., Saro, A., Dolag, K., & Mohr, J. J. 2016, MNRAS, 456, 2361

Bruzual, G., & Charlot, S. 2003, MNRAS, 344, 1000

Bryan, G. L., & Norman, M. L. 1998, ApJ, 495, 80

Bullock, J. S., Wechsler, R. H., & Somerville, R. S. 2002, MNRAS, 329, 246

Campbell, D., van den Bosch, F. C., Padmanabhan, N., et al. 2017, ArXiv e-prints, arXiv:1705.06347

Chabrier, G. 2001, ApJ, 554, 1274

Cole, S., Aragon-Salamanca, A., Frenk, C. S., Navarro, J. F., & Zepf, S. E. 1994, MNRAS, 271, 781

Collaboration, D. E. S. 2005, ArXiv Astrophysics e-prints, astro-ph/0510346

Collaboration, L. D. E. S. 2012, ArXiv e-prints, arXiv:1211.0310 [astro-ph.CO]

Conroy, C., Wechsler, R. H., & Kravtsov, A. V. 2006, ApJ, 647, 201

Cooper, M. C. 2006, in Bulletin of the American Astronomical Society, Vol. 38, American Astronomical Society Meeting Abstracts, 1159

Cora, S. A. 2006, MNRAS, 368, 1540

Davis, M., Efstathiou, G., Frenk, C. S., & White, S. D. M. 1985, ApJ, 292, 371

DESI Collaboration, Aghamousa, A., Aguilar, J., et al. 2016, ArXiv e-prints, arXiv:1611.00036 [astro-ph.IM]

Di Matteo, T., Khandai, N., DeGraf, C., et al. 2012, ApJL, 745, L29

Driver, S. P., Hill, D. T., Kelvin, L. S., et al. 2011, MNRAS, 413, 971

Einasto, J., Klypin, A. A., Saar, E., & Shandarin, S. F. 1984, MNRAS, 206, 529

Fioc, M., & Rocca-Volmerange, B. 1997, A&A, 326, 950

—. 1999, ArXiv Astrophysics e-prints, astro-ph/9912179

Gargiulo, I. D., Cora, S. A., Padilla, N. D., et al. 2015, MNRAS, 446, 3820

Gu, M., Conroy, C., & Behroozi, P. 2016, ApJ, 833, 2

Hearin, A. P., Watson, D. F., Becker, M. R., et al. 2014, MNRAS, 444, 729

Henriques, B. M. B., White, S. D. M., Thomas, P. A., et al. 2015, MNRAS, 451, 2663

Hudelot, P., Cuillandre, J.-C., Withington, K., et al. 2012, VizieR Online Data Catalog, 2317, http://irfu.cea.fr/Images/astImg/3226/T0007-ExecSummary.pdf

Hunter, J. D. 2007, Computing in Science Engineering, 9, 90

Jing, Y. P., Mo, H. J., & Börner, G. 1998, ApJ, 494, 1

Jones, E., Oliphant, T., Peterson, P., et al. 2001–, SciPy: Open source scientific tools for Python, [Online; scipy.org]

Katz, N., Weinberg, D. H., & Hernquist, L. 1996, ApJS, 105, 19

Kauffmann, G., Nusser, A., & Steinmetz, M. 1997, MNRAS, 286, 795

Kauffmann, G., White, S. D. M., & Guiderdoni, B. 1993, MNRAS, 264, 201

Kauffmann, G., Heckman, T. M., White, S. D. M., et al. 2003, MNRAS, 341, 33

Khandai, N., Di Matteo, T., Croft, R., et al. 2015, MNRAS, 450, 1349

Klypin, A., Yepes, G., Gottlöber, S., Prada, F., & Heß, S. 2016, MNRAS, 457, 4340

Komatsu, E., Smith, K. M., Dunkley, J., et al. 2011, ApJS, 192, 18

Kormendy, J., & Ho, L. C. 2013, ARA&A, 51, 511

Kravtsov, A. V., Berlind, A. A., Wechsler, R. H., et al. 2004, ApJ, 609, 35

Lagos, C. D. P., Cora, S. A., & Padilla, N. D. 2008, MNRAS, 388, 587

Leauthaud, A., Tinker, J., Behroozi, P. S., Busha, M. T., & Wechsler, R. H. 2011, ApJ, 738, 45

Lehmann, B. V., Mao, Y.-Y., Becker, M. R., Skillman, S. W., & Wechsler, R. H. 2017, ApJ, 834, 37

Li, C., & White, S. D. M. 2009, MNRAS, 398, 2177

LSST Dark Energy Science Collaboration. 2017, DESCQA, v2.0.0-0.4.5. doi:10.5281/zenodo.1120294

LSST Science Collaboration, Abell, P. A., Allison, J., et al. 2009, ArXiv e-prints, arXiv:0912.0201 [astro-ph.IM]

Lukić, Z., Heitmann, K., Habib, S., Bashinsky, S., & Ricker, P. M. 2007, ApJ, 671, 1160

Madau, P., Ferguson, H. C., Dickinson, M. E., et al. 1996, MNRAS, 283, 1388

Mandelbaum, R., Seljak, U., Kauffmann, G., Hirata, C. M., & Brinkmann, J. 2006, MNRAS, 368, 715

Mandelbaum, R., Wang, W., Zu, Y., et al. 2016, MNRAS, 457, 3200

Marigo, P. 2001, A&A, 370, 194

McConnell, N. J., & Ma, C.-P. 2013, ApJ, 764, 184

Mobasher, B., Dahlen, T., Ferguson, H. C., et al. 2015, ApJ, 808, 101

Moster, B. P., Somerville, R. S., Maulbetsch, C., et al. 2010, ApJ, 710, 903

Newman, J. A., Cooper, M. C., Davis, M., et al. 2013, ApJS, 208, 5

Padmanabhan, N., Schlegel, D. J., Finkbeiner, D. P., et al. 2008, ApJ, 674, 1217

Peacock, J. A., & Smith, R. E. 2000, MNRAS, 318, 1144

Planck Collaboration, Ade, P. A. R., Aghanim, N., et al. 2014, A&A, 571, A16

Portinari, L., Chiosi, C., & Bressan, A. 1998, A&A, 334, 505

Reddick, R. M., Wechsler, R. H., Tinker, J. L., & Behroozi, P. S. 2013, ApJ, 771, 30

Ruiz, A. N., Cora, S. A., Padilla, N. D., et al. 2015, ApJ, 801, 139

Seljak, U. 2000, MNRAS, 318, 203

Sheth, R. K., & Tormen, G. 1999, MNRAS, 308, 119

Silk, J., & Mamon, G. A. 2012, Research in Astronomy and Astrophysics, 12, 917

Somerville, R. S., & Primack, J. R. 1999, MNRAS, 310, 1087

Springel, V., Di Matteo, T., & Hernquist, L. 2005, MNRAS, 361, 776

Springel, V., & Hernquist, L. 2003, MNRAS, 339, 289

Springel, V., White, S. D. M., Tormen, G., & Kauffmann, G. 2001, MNRAS, 328, 726

Tenneti, A., Mandelbaum, R., Di Matteo, T., Kiessling, A., & Khandai, N. 2015, MNRAS, 453, 469

Tinker, J., Kravtsov, A. V., Klypin, A., et al. 2008, ApJ, 688, 709

Vale, A., & Ostriker, J. P. 2004, MNRAS, 353, 189

—. 2006, MNRAS, 371, 1173

van der Walt, S., Colbert, S. C., & Varoquaux, G. 2011, Computing in Science Engineering, 13, 22

Velliscig, M., van Daalen, M. P., Schaye, J., et al. 2014, MNRAS, 442, 2641

Wetzel, A. R., & White, M. 2010, MNRAS, 403, 1072

White, S. D. M., & Frenk, C. S. 1991, ApJ, 379, 52

Zheng, Z., Berlind, A. A., Weinberg, D. H., et al. 2005, ApJ, 633, 791

Zu, Y., & Mandelbaum, R. 2015, MNRAS, 454, 1161

—. 2016, MNRAS, 457, 4360

—. 2017, ArXiv e-prints, arXiv:1703.09219