Injection efficiency simulations

Introduction

This document summarizes how the simulations are executed to compute the Injection Efficiency (IE). This document is not a duplicate of the IE budget and model included in the Sensitivity budget allocation document. This document focuses on the actual Python codes that are executed to simulate the IE.

The codes

There is a main Python code that contains the Python class and methods used to compute the PSF for MSE under various conditions and there are several scripts that uses that class and those methods to execute steps in the computation of the IE. The main code is mse\_injeff\_simu.py while the scripts are run\_psf\_iq.py, run\_psf\_optim.py, and run\_psf\_ie.py. The codes have not been optimized for performance and have never been checked by anyone else than me.

The MsePSF class and methods

The MsePSF class and its associated methods are defined in mse\_injeff\_simu.py. The MsePSF class is intially defined by a source type (point or extended source), zenith distance (0, 30, 50, or 60º), a field position (one of twelve from the Zemax model), a natural seeing value[[1]](#footnote-1), a spectrograph (LR, MR, or HR), and a fiber diameter (0.9 and 1.0˝ for LR and MR, 0.7 and 0.8˝ for HR).

The main methods are open\_zemax, apply\_iq, find\_optim, ie\_budget, and compute\_ie. Other methods are used to compute specific items: simu\_offset\_adr, simu\_offset\_plate, simu\_offset\_inro, simu\_defoc\_poss, and simu\_defoc\_inrotilt.

### open\_zemax

This method simply opens the Zemax files for a given field position and zenith distance, for all five focus positions ran in Zemax. It stores the five spectral cubes (each has 11 frames) and their headers in the class definition.

### apply\_iq

This method convolves each of the fifty-five images (five focus, eleven wavelengths) with a Moffat whose width is provided as an input when calling the MsePSF class. It saves the convolved images into a new file on the harddrive.

### find\_optim

This method finds the optimal position of the fiber given the weights given to each wavelength and for the spectrograph and fiber size given when initiating the class. It saves the optimal position (X and Y) for each spectral cube, but only for the one in focus. It also saves two plots showing the coarse and fine search for the optimal position.

### ie\_budget

This method computes all the random values used in the IE budget to define the X, Y, and Z offsets. We currently use 50 simulations, each with 25 samples for the contributors that vary during an observation. It stores the offsets in the class and saves two plots showing the distribution of offsets.

### compute\_ie

This method measures the flux entering a fiber of the given diameter at the location given by the optimal position and the offsets computed previously. The IE is computed by dividing this flux by that integrated over the image stored in the open\_zemax function (I.e. before the convolution by the Moffat function). The method does that for all wavelengths and focus positions, using the photutils package, defining circular apertures for all simulations and samples. The method then interpolates at the simulated focus (Z offset) from the five simulated data points, averages all samples, and averages all simulations. A plot is saved showing all samples and all simulations, as well as the optimal IE and the simulated IE. The simulated IE is then stored.

simu\_offset\_adr, simu\_offset\_plate, simu\_offset\_inro, simu\_defoc\_poss, and simu\_defoc\_inrotilt

Those methods compute various offsets that are not simply picked from uniform or normal distributions.

How to run the simulations

Several scripts exist to run the code one step at a time: run\_psf\_iq.py, run\_psf\_optim.py, and run\_psf\_ie.py.

run\_psf\_iq

This script opens the Zemax files and execute the apply\_iq method so that files are created and saved for each configuration (field position, zenith distance, fiber diameter, natural seeing). It currently loops over the twelve field positions, four zenith distances, and thirteen natural seeing values ranging from 0.3˝ to 2.0˝).

run\_psf\_optim

This script finds and saves the optimal position of the fiber for all configuration (field position, zenith distance, fiber diameter, natural seeing, spectrograph, fiber diameter). It currently loops over the twelve field positions, four zenith distances, thirteen natural seeing values ranging from 0.3˝ to 2.0˝), and four spectrographs setting (LR with 1.0˝ fibers, LR with 0.9˝ fibers, HR with 0.8˝ fibers, HR with 0.7˝ fibers). It saves all optimal IE curves at the end in both a numpy array file and picke file.

run\_psf\_ie

This script simulates the IE budget (X, Y, Z offsets) and computes the IE for all configuration (field position, zenith distance, fiber diameter, natural seeing, spectrograph, fiber diameter). It saves all simulated IE curves at the end in both a Numpy array file and Pickle file.

Run times

This are run times using PyCharm on my MacMini (3GHz i7, 16 GB):

* run\_psf\_iq (about 400,000s or 111 hours, running in parallel with other processes, for 12 field positions, 4 zenith distances, and 13 IQ values between 0.3 and 2.0˝).
* run\_psf\_optim (about 100,000s or 28 hours, running in parallel with other processes, for 12 field positions, 4 zenith distances, 4 spectro/fiber diameters, and 13 IQ values between 0.3 and 2.0˝).
* run\_psf\_ie (about 300,000s or 83 hours, running in parallel with other processes, for 12 field positions, 4 zenith distances, 4 spectro/fiber diameters, and 13 IQ values between 0.3 and 2.0˝).

# Results

All curves

At the end of the simulations (ran in September 2018), we have 2 files with all injection efficiency (IE) curves: one for the IE curves at the optimal position and one for the IE at the simulated position. The average IE curve and the standard deviation for each configuration are saved. The files are saved as Numpy and Pickle files, which should be straightforward to read in Python.

The format of the IE curves is that of a dictionary of dictionaries of dictionaries of dictionaries of Numpy arrays. The first keys identify the IQ, the second keys identify the Zenith distance, the third keys identify the field position, and the fourth keys identify the spectrograph and fiber size. For instance, once the file is loaded in a Python session, one might look at one IE curve by doing:

> ie['iq0.6']['ZD30']['X\_+0.650']['LR1.0']

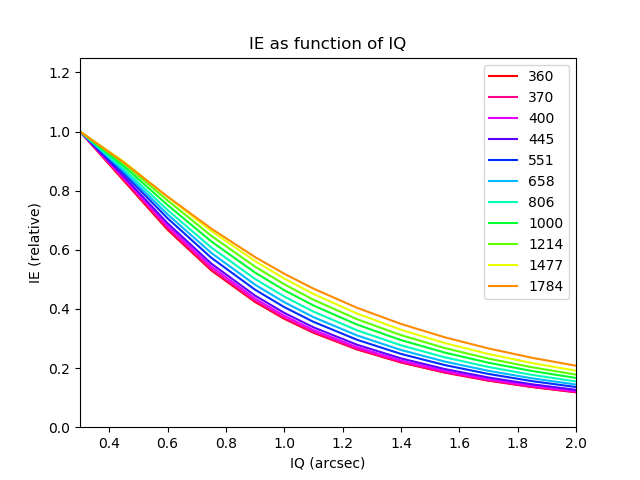
This then singles out the 11-element Nupy array which is the IE curve in that configuration.

The 11 elements correspond to the 11 wavelengths used in Zemax and shown below. The configurations are defined by four parameters taken from the following.

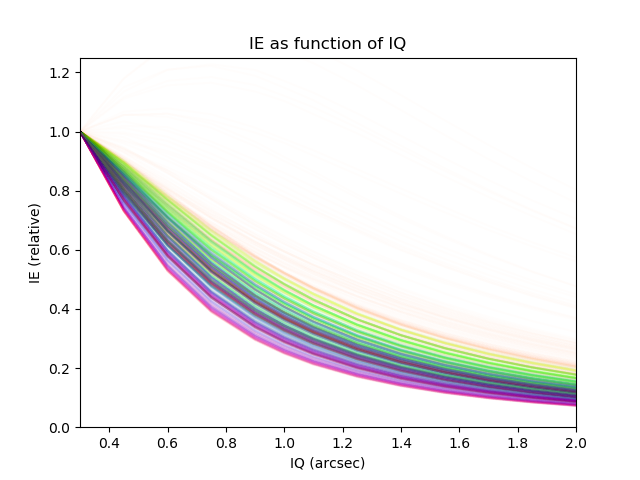
* Wavelengths: 360, 370, 400, 445, 551, 658, 806, 1000, 1214, 1477, and 1784 in nm
* Parameters for IQs: 0.3, 0.45, 0.6, 0.75, 0.9, 1.0, 1.1, 1.25, 1.4, 1.55, 1.7, 1.85, and 2.0 arcsec
* Fields positions: X\_+0.530, X\_+0.650, X\_+0.750, Y\_+0.000, Y\_+0.375, Y\_+0.530, Y\_+0.650, Y\_+0.750, Y\_-0.375, Y\_-0.530, Y\_-0.650, Y\_-0.750
* Zenith distances: 0, 30, 50, and 60 degrees
* Spectrograph and fiber diameters: LR 1.0˝, LR 0.9˝, MR 1.0˝, HR 0.8˝, HR 0.7˝.

Trends

We sample the typical IQ observed at CFHT (0.3 to 2.0˝) and expect to interpolate the IE at any IQ in that range. To verify this is actually possible and that the IE varies smoothly with IQ, we plot the IE as a function of IQ for all configurations. One example is shown below at a Zenith distance of 30º, for a field position of X+0.650º, and for the LR spectrograph fed with 1.0˝ fiber. In this plot, the relative IE (IE at a given IQ divided by the IE at an IQ of 0.3˝) is shown. All the wavelengths indeed do not produce the same IE because of IQ variations with wavelength and normalizing the IE this way to make it easier to see the trend.



The trend is slightly wavelength dependent, as well as field position and Zenith distance dependent (seen when comparing plots in different configurations). When plotting all IE as a function of IQ for all field position, Zenith distance, and wavelength, we obtain the figure below, with the same color code as above.



The IE drops faster with increasing IQ at short wavelength than long wavelength. This surely is because the IQ depends on wavelengths as 𝜆−0.2. We also note that the longer wavelength quite often presents a peak at an IQ that is not the best. It seems like the IE near 1800 nm is sometimes the largest when the IQ is somewhere around 0.6 and 0.8˝ rather than at an IQ of 0.3˝. Even when this is not the case, it frequently occurs that the decrease of IE at that wavelength is much slower with increasing IQ than at other wavelengths.

This effect is the most visible at a Zenith distance of 60º, near one edge of the field, and for smaller fibers. This can be explained by the lateral chromatic aberration of the optical design. At 60º from Zenith, the system is more achromatic and the extreme infrared wavelength is significantly away from most of the other wavelength. The optimal position of the fiber thus is so “far away” from the position of the 1800 nm PSF that a small fiber at that optimal position does not cover the peak of the 1800 nm PSF. Only when the fiber or IQ becomes larger does the IE at 1800 nm increases because a larger fraction of the PSF is falling within the fiber’s radius. This remains true up to the point where the PSF becomes too large relative to the fiber.

The following figures illustrate the chromatic aberration of the optical design at various Zenith distances and how it increases on one side of the field (the positive Y) significantly more than the other. Each panel is about 150 microns on the side so in the most extreme cases (Y+0.75 at ZD 60º) the peaks of the PSF cover more than about 100 microns which is the largest fiber size we use in those simulations.

|  |  |
| --- | --- |
| ZD 0º | ZD 30º |
| ZD 50º | **ZD 60º** |

Budget

In the sensitivity budget, we use a configuration that is defined in the science requirements: airmass of 1.2 and median seeing. We assume here that the wording of the requirement will be changed from seeing of 0.6˝ to median seeing. The airmass of 1.2 does not correspond to any Zenith distance in the Zemax model (limited to 0, 30, 50, and 60º) though it is close enough to that at 30º (AM1.15) that we consider this discrepancy as negligible.

In addition, the science requirements on sensitivity seem to imply they should be met at any position in the field of view. In the sensitivity budget, we assume however that the requirement shall only be met over 95% of the field of view, which corresponds to about 90% radius (91.4% seems a more accurate value) or an angular distance of about 0.675º (0.695º seems a more accurate value) from the optical axis.

Hereafter we give the values for the IE along the X and Y axis at 0.675º and 0.695º from the optical axis, for the LR spectrograph and 1.0˝ fiber. Because the optical design is not symmetric along the Y axis, we give values on both sides of the optical axis. The first table gives the IE values for an IQ of 0.45˝ and the second for 0.6˝. In each case the worst value is highlighted, using the weights defined in the Zemax model.

We highlight here that the IQ represents the FWHM of a Moffat with which the optical design of MSE (ideal ADC/WFC + M1) is convolved. This IQ value is dominated by the natural seeing at the site but, as indicated in the IQ budget, several other contributors are significant. In its current state, the IQ budget corresponds to 0.57˝ FWHM at 500 nm with a contribution from the ideal WFC/ADC+M1 of 0.115˝ FWHM. Considering this contribution is added in quadrature, computing the IQ without it corresponds to 0.56˝ FWHM at 500nm.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Wav.** | **360** | **370** | **400** | **445** | **551** | **658** | **806** | **1000** | **1214** | **1477** | **1784** |
| **IQ0.45˝ X+0.695** | 0.65 | 0.66 | 0.68 | 0.69 | 0.69 | 0.70 | 0.70 | 0.71 | 0.71 | 0.70 | 0.66 |
| **IQ0.45˝** **Y-0.695** | 0.65 | 0.67 | 0.68 | 0.68 | 0.70 | 0.71 | 0.72 | 0.72 | 0.72 | 0.71 | 0.68 |
| **IQ0.45˝** **Y+0.695** | *0.62* | *0.65* | *0.68* | *0.68* | *0.66* | *0.66* | *0.67* | *0.70* | *0.71* | *0.67* | *0.51* |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Wav.** | **360** | **370** | **400** | **445** | **551** | **658** | **806** | **1000** | **1214** | **1477** | **1784** |
| **IQ0.60˝ X+0.695** | 0.52 | 0.53 | 0.55 | 0.56 | 0.57 | 0.58 | 0.59 | 0.60 | 0.61 | 0.61 | 0.58 |
| **IQ0.60˝** **Y-0.695** | 0.53 | 0.53 | 0.55 | 0.55 | 0.57 | 0.59 | 0.60 | 0.61 | 0.61 | 0.60 | 0.59 |
| **IQ0.60˝** **Y+0.695** | *0.50* | *0.52* | *0.55* | *0.55* | *0.55* | *0.55* | *0.57* | *0.60* | *0.61* | *0.58* | *0.45* |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Wav.** | **360** | **370** | **400** | **445** | **551** | **658** | **806** | **1000** | **1214** | **1477** | **1784** |
| **IQ0.45˝ X+0.675** | 0.66 | 0.67 | 0.68 | 0.69 | 0.70 | 0.70 | 0.70 | 0.71 | 0.72 | 0.71 | 0.67 |
| **IQ0.45˝** **Y-0.675** | 0.67 | 0.68 | 0.69 | 0.69 | 0.70 | 0.71 | 0.72 | 0.72 | 0.72 | 0.71 | 0.69 |
| **IQ0.45˝** **Y+0.675** | *0.64* | *0.66* | *0.69* | *0.68* | *0.67* | *0.67* | *0.68* | *0.70* | *0.71* | *0.67* | *0.52* |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Wav.** | **360** | **370** | **400** | **445** | **551** | **658** | **806** | **1000** | **1214** | **1477** | **1784** |
| **IQ0.60˝ X+0.675** | 0.53 | 0.54 | 0.55 | 0.56 | 0.57 | 0.58 | 0.59 | 0.60 | 0.61 | 0.61 | 0.58 |
| **IQ0.60˝** **Y-0.675** | 0.53 | 0.54 | 0.55 | 0.55 | 0.57 | 0.59 | 0.60 | 0.61 | 0.61 | 0.61 | 0.59 |
| **IQ0.60˝** **Y+0.675** | *0.51* | *0.53* | *0.56* | *0.56* | *0.55* | *0.56* | *0.58* | *0.60* | *0.61* | *0.58* | *0.46* |

1. By construction, this value corresponds to the image quality delivered by MSE at the focal surface, except for the contribution of the optical design (9/20/2019). This definition could benefit from a modification where the input value would only correspond to the natural seeing and the code would add the other contributors to the image quality. [↑](#footnote-ref-1)