LSST Active Optics Research

David Thomas
Stanford University
LSST Mirrors Bend and Warp
Mirror Surface Impacts PSF

M2 Primary Bending Mode

500nm

-500nm
Mirror Surface Impacts PSF

\[ PSF = \text{Atm} \otimes \text{Sys} \otimes M2b1 \]
Mirror Surface Impacts PSF

\[ PSF = Atm \otimes Sys \otimes M2b1 \]
Mirror Surface Impacts PSF

\[ PSF = Atm \otimes Sys \otimes M2b1 \]
Mirror Surface Impacts PSF

$PSF = \text{Atm} \otimes \text{Sys} \otimes M2b1$
Many Degrees of Freedom

50 Degrees of Freedom:
- 5 Camera Hexapod
- 5 M2 Hexapod
- 20 M1M3 Bending Modes
- 20 M2 Bending Modes
Both Open and Closed Control Loops

Open Loop Corrects:
- Fabrication
- Gravity
- Bulk Temperature

Closed Loop Corrects:
- Gravity Residual
- Thermal Residual and Gradients
- Hysteresis
Sense Aberrations with Wavefront Sensors
Wavefront Sensors Produce Donuts
Three Solutions

Curvature Sensing

Forward Modeling

Machine Learning
Process Donuts in Multiple Steps

Wavefront Sensor Image

Intra  Extra
Process Donuts in Multiple Steps

Wavefront Sensor Image

Donut Pair

Extract: Find best intra-focal and extra-focal donuts and crop them.
Process Donuts in Multiple Steps

Extract: Find best intra-focal and extra-focal donuts and crop them.

De-blend: Remove flux from neighboring sources.
Process Donuts in Multiple Steps

**Extract**: Find best intra-focal and extra-focal donuts and crop them.

**De-blend**: Remove flux from neighboring sources.

**Correct**: Use precomputed interpolation to correct for off-axis distortion and field dependent effects; flip extra-focal donut.
Find best intra-focal and extra-focal donuts and crop them.

**De-blend:** Remove flux from neighboring sources.

**Correct:** Use precomputed interpolation to correct for off-axis distortion and field dependent effects; flip extra-focal donut.

**Normalize:** Divide by total flux so that donuts from different sources are on the same footing.
Solve for Wavefront

Transport of Intensity Equation

\[
\frac{\partial I}{\partial z} = - \nabla \cdot (I \nabla W)
\]
Solve for Wavefront

Plug in donut approximations

Transport of Intensity Equation

\[ \frac{\partial I}{\partial z} = -\nabla \cdot (I \nabla W) \]
Solve for Wavefront

Plug in donut approximations

Transport of Intensity Equation

Get out wavefront in annular Zernike basis

\[
\frac{\partial I}{\partial z} = -\nabla \cdot (I \nabla W)
\]

\[
W = \alpha_1 + \cdots + \alpha_{19}
\]
Solve for Optimal Next State

- We take the pseudo-inverse to solve for the current optical state $x_k$.

$$W = A x_k$$

- $W$: Sensitivity Matrix
- $A$: Optical State
- $x_k$: Current Optical State
Solve for Optimal Next State

- We take the pseudo-inverse to solve for the current optical state $x_k$.
- Solve for the next state $x_{k+1}$ by minimizing the cost function that accounts for *image quality* and *size* of the update.

\[
W = Ax_k
\]

\[
J(x_{k+1}) = x_{k+1}Qx_{k+1} + (x_{k+1} - x_k)H(x_{k+1} - x_k)
\]

- Image Quality
- Update Penalty
We Can Simulate Most of Baseline
Three Solutions

Curvature Sensing

Forward Modeling

Machine Learning
Donuts with Fraunhofer Diffraction Integral

\[ I_{\text{model}}(W) \propto \left| \mathcal{F}\left\{ Pe^{2\pi i W/\lambda} \right\} \right|^2 \]

Fourier Transform

Pupil Mask
Donuts with Fraunhofer Diffraction Integral

\[ I_{\text{model}}(W) \propto \left| \mathcal{F}\left\{ P e^{2\pi i W/\lambda} \right\} \right|^2 \]
Solve for Wavefront with Forward Modeling

\[
\min_W \left\| I_{model}(W) - I_{data} \right\|
\]
Forward Model Everything

$$\min_{x_k} \left\| I_{model}(x_k, \text{catalog}) - I_{data} \right\|$$
Forward Model Everything

$$\min_{x_k} \| I_{\text{model}}(x_k, \text{catalog}) - I_{\text{data}} \|$$
Forward Model Everything

\[
\min_{x_k} \left\| I_{model}(x_k, \text{catalog}) - I_{data} \right\|
\]
Three Solutions

Curvature Sensing

Forward Modeling

Machine Learning
Predict Mirror Surface with Deep Learning

Intra-focal Donuts → M2 Residual
Predict Mirror Surface with Deep Learning

Intra-focal Donuts

Neural Network

M2 Residual
Train Deep Neural Network

\[
\text{Loss} = \| \text{Output} - \text{Label} \|^2
\]

Training

- \text{train}
- \text{test}

Epochs

\[
10^{-2}
\]
Train Deep Neural Network

Training

\[ \text{Loss} = \|\text{Output} - \text{Label}\|^2 \]

epochs

Input  Output  Label
Framework for LSST AOS Research

ActiveOpticsSimulator

A framework for simulating and researching the LSST Active Optics system. We hope to document and transfer much of the functionality in bxin/IM (latest fork: davidthomas5412/IM) and bxin/cwfs (latest fork: davidthomas5412/cwfs) into this new framework, as well as make it easy to explore alternative approaches. We will also be introducing the following software practices:

1. Testing: unit tests, continuous integration, track code coverage.
2. Documentation: numpy docstrings, no magic interpolations - all hardcoded matrices etc. must be derived or explained in corresponding notebooks.
3. Modular design: we strive to make it easy and natural for users to try new alternatives and swap out parts of the pipeline.
5. Open source: public github repo, issue tracking, pull requests welcome.
Active Optics Research Is A Team Effort

**LSST:**
- Bo Xin
- Sandrine Thomas
- Chuck Claver
- Te-Wei Tsai

**SLAC/Stanford:**
- Aaron Roodman
- Steven Kahn
- Emily Li
- David Thomas

**LLNL:**
- Josh Meyers

**Harvard:**
- Jun Yin
- Chris Stubbs
Questions?

PSF Problem

Three Solutions
- Curvature Sensing
- Forward Modeling
- Machine Learning

One Framework