# Estimation of Time Delay in Gravitationally Lensed Photon Stream Pairs

Michał Staniaszek
Supervisor: Peter Tiňo

The University of Birmingham

March 21, 2013

# Outline

# A Basic Overview

- Two rays of photons are emitted from a star or quasar
- Gravitational field of a massive object bends the rays
- Bent light means that we see two images of the source
- Capture arrival times of photons from each image (events)
- Stream $s_A$ arrives at time $t$, stream $s_B$ at $t + \Delta$
- We want to find $\Delta$

# A Basic Overview

- Two rays of photons are emitted from a star or quasar
- Gravitational field of a massive object bends the rays
- Bent light means that we see two images of the source
- Capture arrival times of photons from each image (events)
- Stream $s_A$ arrives at time $t$, stream $s_B$ at $t + \Delta$
- We want to find $\Delta$

# A Basic Overview

- Two rays of photons are emitted from a star or quasar
- Gravitational field of a massive object bends the rays
- Bent light means that we see two images of the source
- Capture arrival times of photons from each image (events)
- Stream $s_A$ arrives at time $t$, stream $s_B$ at $t + \Delta$
- We want to find $\Delta$

# A Basic Overview

- Two rays of photons are emitted from a star or quasar
- Gravitational field of a massive object bends the rays
- Bent light means that we see two images of the source
- Capture arrival times of photons from each image (events)
- Stream $s_A$ arrives at time $t$, stream $s_B$ at $t + \Delta$
- We want to find $\Delta$

# A Basic Overview

- Two rays of photons are emitted from a star or quasar
- Gravitational field of a massive object bends the rays
- Bent light means that we see two images of the source
- Capture arrival times of photons from each image (events)
- Stream $s_A$ arrives at time $t$, stream $s_B$ at $t + \Delta$
- We want to find $\Delta$

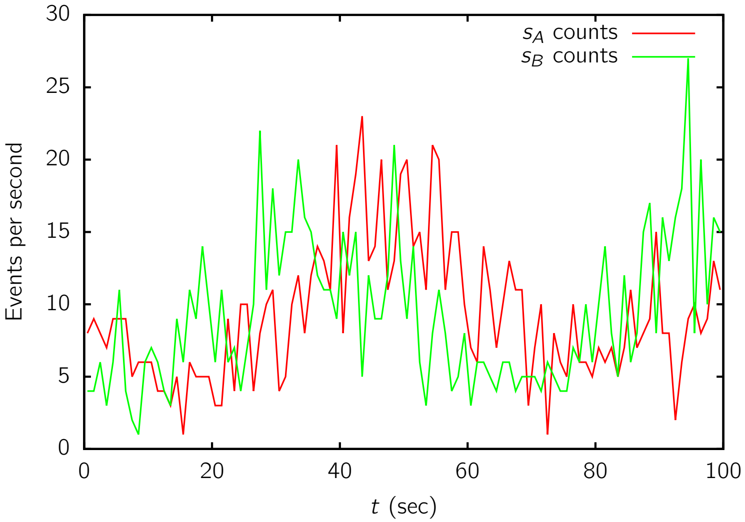
# A Basic Overview

- Two rays of photons are emitted from a star or quasar
- Gravitational field of a massive object bends the rays
- Bent light means that we see two images of the source
- Capture arrival times of photons from each image (events)
- Stream $s_A$ arrives at time $t$, stream $s_B$ at $t + \Delta$
- We want to find $\Delta$

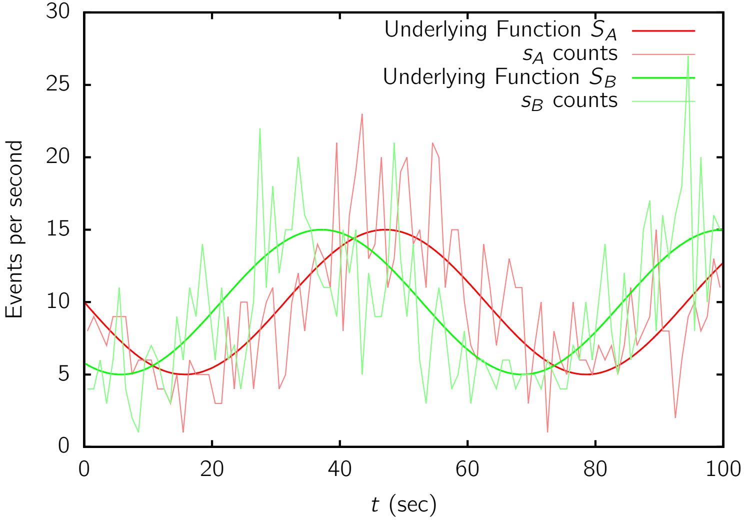
# A Basic Overview

- Two rays of photons are emitted from a star or quasar
- Gravitational field of a massive object bends the rays
- Bent light means that we see two images of the source
- Capture arrival times of photons from each image (events)
- Stream $s_A$ arrives at time $t$, stream $s_B$ at $t + \Delta$
- We want to find $\Delta$

# What We Get


30
25
20
15
10
5
0
0
20
40
60
80
100
Events per second
t (sec)
$s_A$ counts
$s_B$ counts

# What We Want


Underlying Function $S_A$
$s_A$ counts
Underlying Function $S_B$
$s_B$ counts
Events per second
$t$ (sec)
0
5
10
15
20
25
30
0
20
40
60
80
100

# What Can We Do With $\Delta$?

- Estimate $H_o$
- Improve accuracy of stellar distance measurements
- Probe the nature of dark matter
- Detect extrasolar planets
- Measure mass distribution
- Many more proposed applications

# What Can We Do With $\Delta$?

- Estimate $H_o$
- Improve accuracy of stellar distance measurements
- Probe the nature of dark matter
- Detect extrasolar planets
- Measure mass distribution
- Many more proposed applications

# What Can We Do With $\Delta$?

- Estimate $H_o$
- Improve accuracy of stellar distance measurements
- Probe the nature of dark matter
- Detect extrasolar planets
- Measure mass distribution
- Many more proposed applications

# What Can We Do With Δ?

- Estimate $H_o$
- Improve accuracy of stellar distance measurements
- Probe the nature of dark matter
- Detect extrasolar planets
- Measure mass distribution
- Many more proposed applications

# What Can We Do With $\Delta$?

- Estimate $H_o$
- Improve accuracy of stellar distance measurements
- Probe the nature of dark matter
- Detect extrasolar planets
- Measure mass distribution
- Many more proposed applications

# What Can We Do With $\Delta$?

- Estimate $H_o$
- Improve accuracy of stellar distance measurements
- Probe the nature of dark matter
- Detect extrasolar planets
- Measure mass distribution
- Many more proposed applications

# What Can We Do With $\Delta$?

- Estimate $H_o$
- Improve accuracy of stellar distance measurements
- Probe the nature of dark matter
- Detect extrasolar planets
- Measure mass distribution
- Many more proposed applications

# Previous Approaches

For estimation of the time delay from quasar Q0957+561, the following methods have been used:

- Structure-function based methods (PRH)
- Dispersion Spectra
- Cross correlation

# How We Approach the Problem

Basic function estimation:

- Linear
- Piecewise linear

As a final goal:

- Model-based estimators with maximum likelihood

# Implementation

The project will require implementation of:

- Photon event simulator (arrival time only)
- Linear and piecewise linear estimator
- Model based estimator
- A method for calculation of $\Delta$

# Implementation

The project will require implementation of:

- Photon event simulator (arrival time only)
- Linear and piecewise linear estimator
- Model based estimator
- A method for calculation of $\Delta$

# Implementation

The project will require implementation of:

- Photon event simulator (arrival time only)
- Linear and piecewise linear estimator
- Model based estimator
- A method for calculation of $\Delta$

# Implementation

The project will require implementation of:

- Photon event simulator (arrival time only)
- Linear and piecewise linear estimator
- Model based estimator
- A method for calculation of $\Delta$

# Implementation

The project will require implementation of:

- Photon event simulator (arrival time only)
- Linear and piecewise linear estimator
- Model based estimator
- A method for calculation of $\Delta$

# What Have We Done So Far?

- Photon arrival time simulation - complete
- Linear estimator - complete
- Piecewise estimator - in progress

# What Have We Done So Far?

- Photon arrival time simulation - complete
- Linear estimator - complete
- Piecewise estimator - in progress

# What Have We Done So Far?

- Photon arrival time simulation - complete
- Linear estimator - complete
- Piecewise estimator - in progress

# What Have We Done So Far?

- Photon arrival time simulation - complete
- Linear estimator - complete
- Piecewise estimator - in progress

# Photon Simulation

We use a non-homogeneous poisson process (NHPP) to simulate arrival times.

- Rate parameter $\lambda$ is the expected number of arrivals per unit time
- Waiting time until the next event has an exponential distribution
- Time to next event in homogeneous process $t = -\frac{1}{\lambda}\ln(U)$

# Poisson Generation Algorithm

**Algorithm 1** Simulating T Time Units of a NHPP by Thinning

**Require:** $\lambda \geq \lambda(t), 0 \leq t \leq T$

$E = \emptyset,\ t = 0,\ T =$ interval length
**while** $t < T$ **do**
    Generate $U_1 \sim U(0,1)$
    $t = t - \frac{1}{\lambda} \ln(U_1)$
    Generate $U_2 \sim U(0,1)$, independent of $U_1$
    **if** $U_2 \leq \frac{\lambda(t)}{\lambda}$ **then**
        $E \leftarrow t$
    **end if**
**end while**
**return** $E$

# Linear Estimation

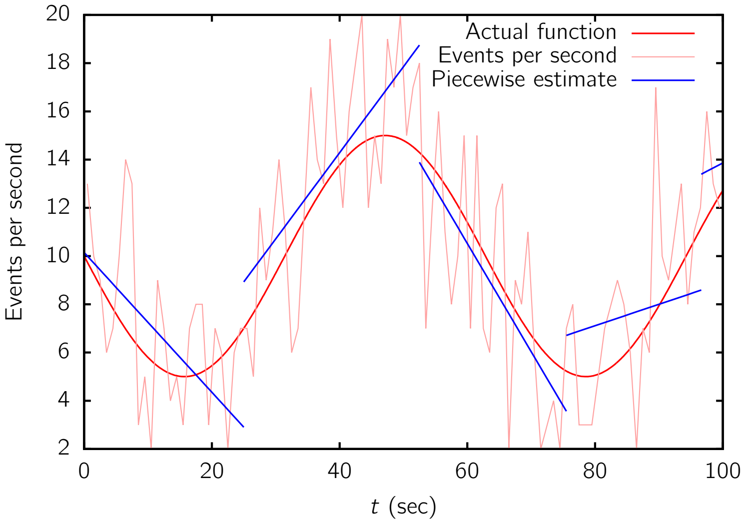
As a first step, attempt to estimate linear functions of the form $y = a + bx$ using Iterative Weighted Least Squares (IWLS)

- Extension of Optimum Least Squares (OLS) method.
- Find
$$\min_{\alpha,\beta} \sum_{k=1}^{n} w_k(Y_k - [\alpha + \beta x])^2$$
- $\alpha$ and $\beta$ are estimators for $a$ and $b$, $w_k$ is the weight assigned to each value $Y_k$, which is the event count for the $k$th bin. $x$ is the midpoint of the sub-interval.
- Update weights at each iteration by using estimated values of $\lambda$ in each sub-interval.

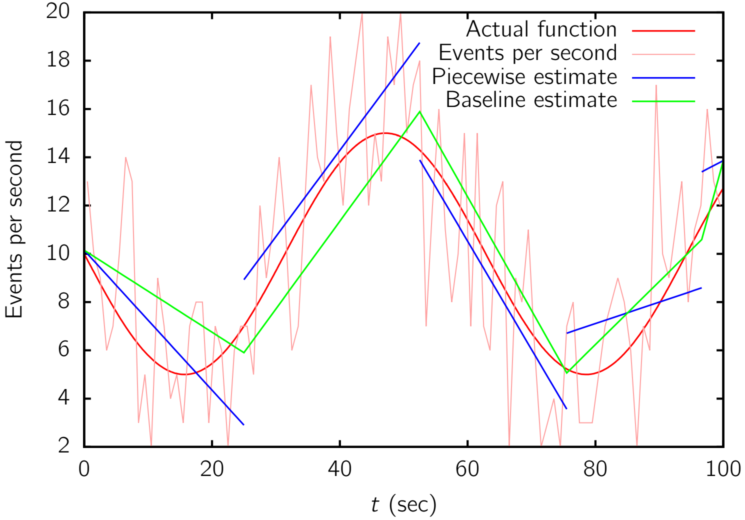
# Other Simple Methods

1. Piecewise estimation
   - Split the interval into sub-intervals and find an estimate for each one separately.
2. Baseline estimation
   - Extension of the piecewise method. Find midpoint between two function estimates at breakpoints.

# Piecewise Estimate Example



# Baseline Estimate vs Piecewise Estimate



# Model Based Estimation

- Attempt to impose an explicit model on how $\lambda$ changes in time.
- Two streams of photons, $S_A$ and $S_B$ have models $\lambda_A(t)$ and $\lambda_B(t)$ respectively.
- If the two streams come from the same source, then $\lambda_B(t) = \lambda_A(t - \Delta)$, as both streams will have the same model.
- To calculate the time delay, we vary $\Delta$ and attempt to maximise the value of

$$Pr(S_A, S_B | \lambda_A, \Delta)$$

# Code Structure

The code is intended to be as modular as possible. Libraries are available to all modules and provide basic utilities that do not belong elsewhere. Current code is split as follows:

- Libraries (math, file, general)
- Generators
- Estimators
- Plotting scripts

# Projected Schedule

- Over Christmas:
    - Complete piecewise estimators
    - Code experiment harness, write unit tests, polish current code base
    - Reading about ML, Latent Variable Models
    - Implementing parts of ML estimator
- End of January: Prototype ML estimator complete
- Mid February: ML estimator complete
- End of February: Estimation of $\Delta$
- Mid March: Experimentation complete

# Summary

- We want to find the value of $\Delta$, the time delay between photon stream arrival times
- Currently have linear and piecewise linear estimators
- Final goal is to use a model-based estimator