Calibration of ADJULES

Tim Jupp

University of Exeter

June 10, 2010

Outline

Outline

Introduction

With thanks to:

- Thomas Kaminski (FastOpt.com)
- Simon Blessing (FastOpt.com)
- Matt Pryor (Met Office, Wallingford)
- Eleanor Blyth (CEH, Wallingford)

Overview of adjoint model

Used commercial software (FastOpt) to differentiate JULES v2.0 line–by-line

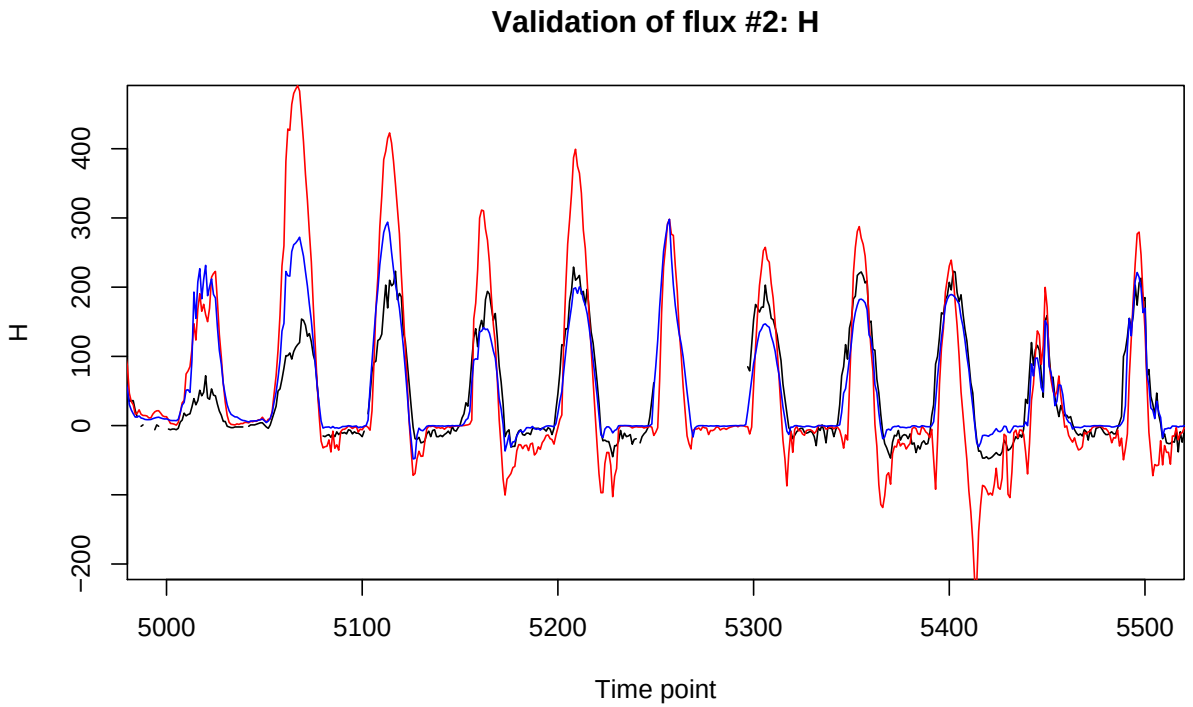
- ADJULES is subroutine callable from within `R`
- for a parameter vector $\mathbf{x}$:
- ADJULES finds cost function $f(\mathbf{x})$
- ADJULES finds derivative of cost function : $\nabla f(\mathbf{x})$

Use of ADJULES

- use R-routines for efficient search of parameter space
- start from 'initial' parameter vector $\mathbf{x}_0$
- find 'best' parameter vector $\mathbf{x}_1$
- most user decisions made at runtime
- works and ready for testing by JULES community
- licensing and future development?

Outline

Calibration



black – observed

red – modelled with 'initial' parameter vector $\mathbf{x_0}$

blue – modelled with 'final' parameter vector $\mathbf{x_1}$

Cost function

ADJULES currently hardwired to calibrate against 3 fluxes:

$$\mathbf{o}_t = \begin{pmatrix} NEE_t \\ H_t \\ LE_t \end{pmatrix}; \quad \mathbf{m}_t = \begin{pmatrix} NEE_t \\ H_t \\ LE_t \end{pmatrix}$$

Cost function is weighted least–squares

$$f(\mathbf{x}) = \sum_{t=1}^{n} (\mathbf{m}_t - \mathbf{o}_t)^T w^T w (\mathbf{m}_t - \mathbf{o}_t)$$

where 3–by–3 weight matrix w can be chosen at runtime.

Flexibility in choice of w

For example

- $w \propto \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ – consider *NEE* only
- $w \propto \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ – consider *H* only
- often choose $w^T w \propto$ 'observed covariance' when $\mathbf{x} = \mathbf{x}_0$

Full parameter space: **x**

ADJULES currently has 50 parameters in the vector **x**:

q10_leaf	nl0_1	nl0_2	nl0_3	nl0_4
nl0_5	alpha_1	alpha_2	alpha_3	alpha_4
alpha_5	f0_1	f0_2	f0_3	f0_4
f0_5	tlow_1	tlow_2	tlow_3	tlow_4
tlow_5	tupp_1	tupp_2	tupp_3	tupp_4
tupp_5	lai_1	lai_2	lai_3	lai_4
lai_5	canht_ft_1	canht_ft_2	canht_ft_3	canht_ft_4
canht_ft_5	satcon_1	satcon_2	satcon_3	satcon_4
b_1	b_2	b_3	b_4	cs
rootd_ft_1	rootd_ft_2	rootd_ft_3	rootd_ft_4	rootd_ft_5

Table: Parameters currently hardwired into ADJULES.

Procedure

In practice, choose a subset $\mathbf{z}$ of full parameter vector $\mathbf{x}$

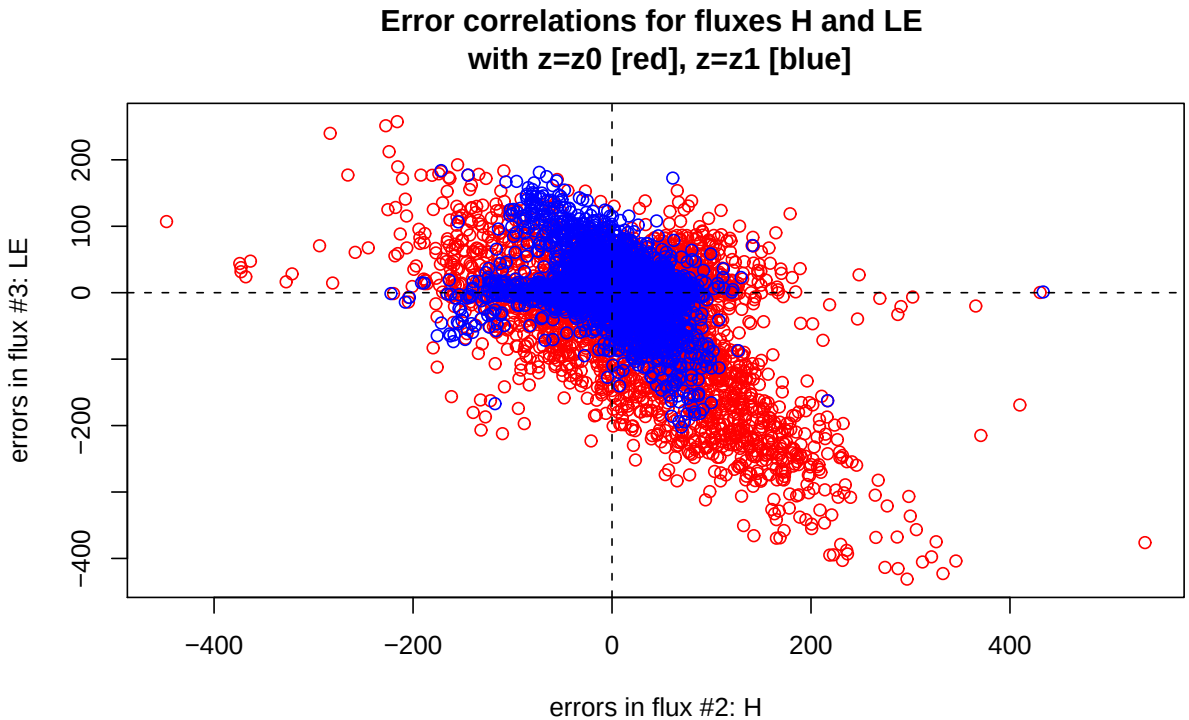
- choose suitable weight matrix w
- rescale w so that $f(\mathbf{z_0}) = 1$
- ADJULES can evaluate $f(\mathbf{z})$ and $\nabla f(\mathbf{z})$ for any $\mathbf{z}$
- Optimisation routines find local minimum: $\mathbf{z_1}$
- $f(\mathbf{z}_1) < f(\mathbf{z}_0) = 1$

Outline

Example: Morgan–Monroe data

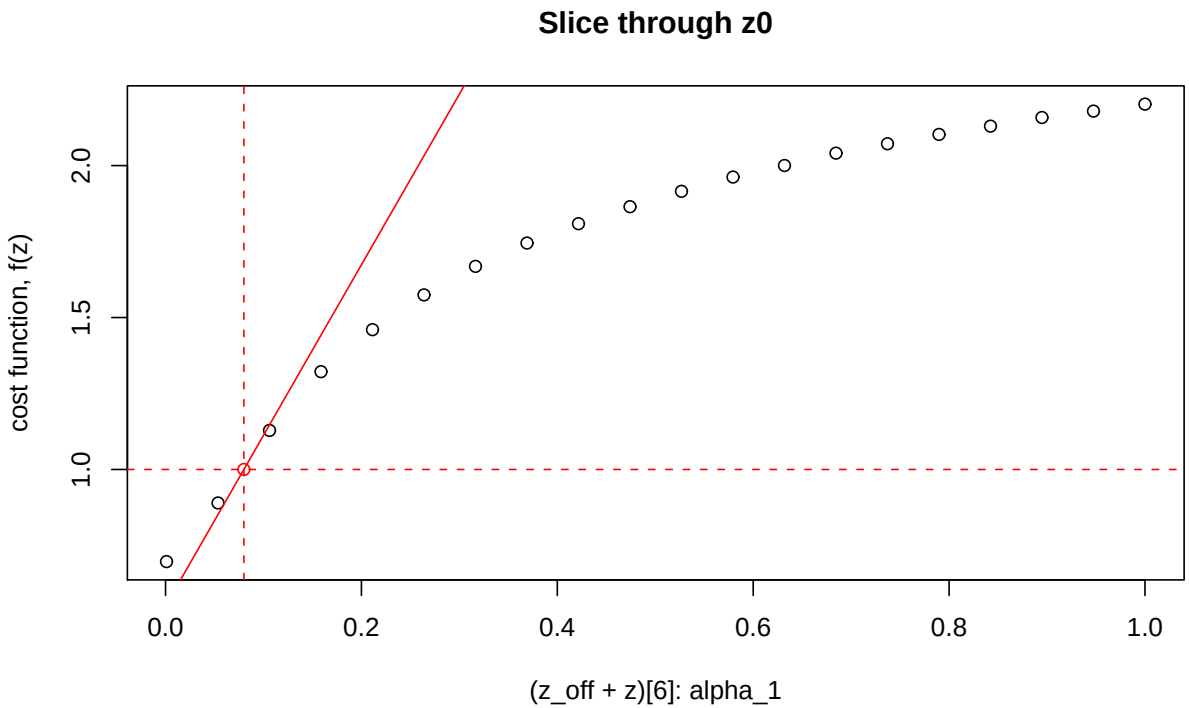
- one year of hourly data $\sim$ 9000 data points
- calibrate against H and LE
- so $w = \begin{pmatrix} 0 & 0 & 0 \\ 0 & a & b \\ 0 & 0 & c \end{pmatrix}$
- chose to optimise over 29 parameters in z
- optimisation takes $\sim$ 30 $\mathrm{minutes}$ on high end unix machine

Example: Morgan–Monroe data

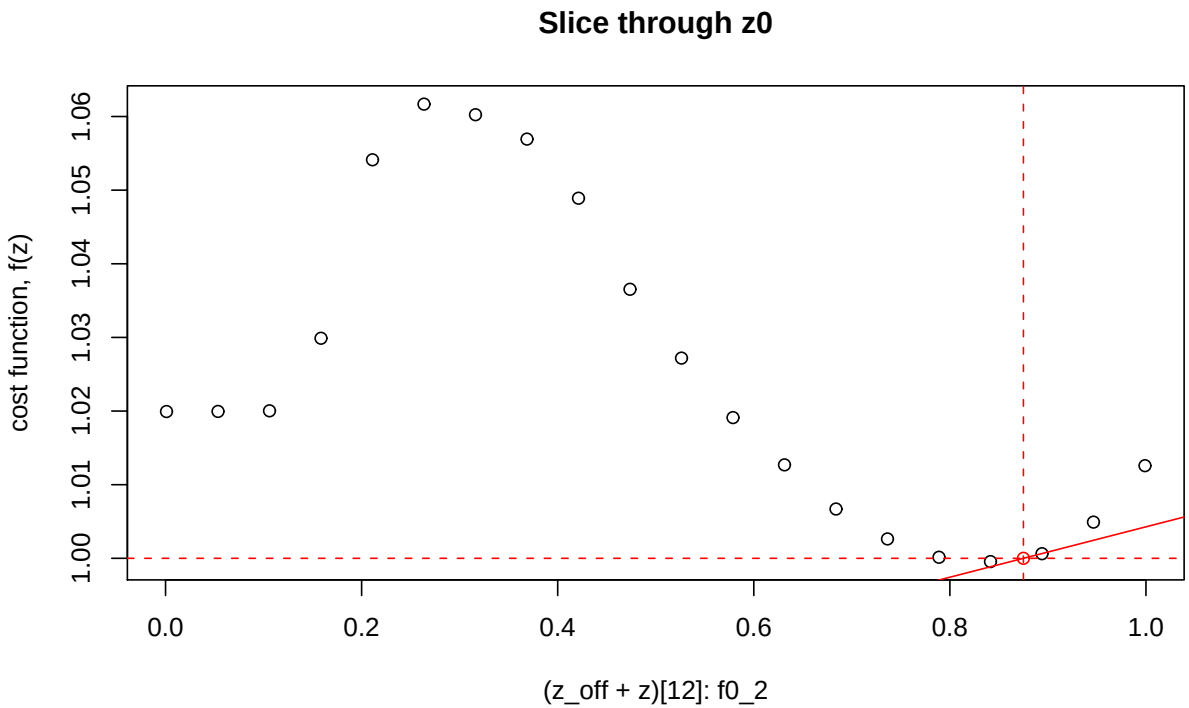


red – modelled with 'initial' parameter vector $\mathbf{x_0}$. Used to define w.
blue – modelled with 'final' parameter vector $\mathbf{x_1}$

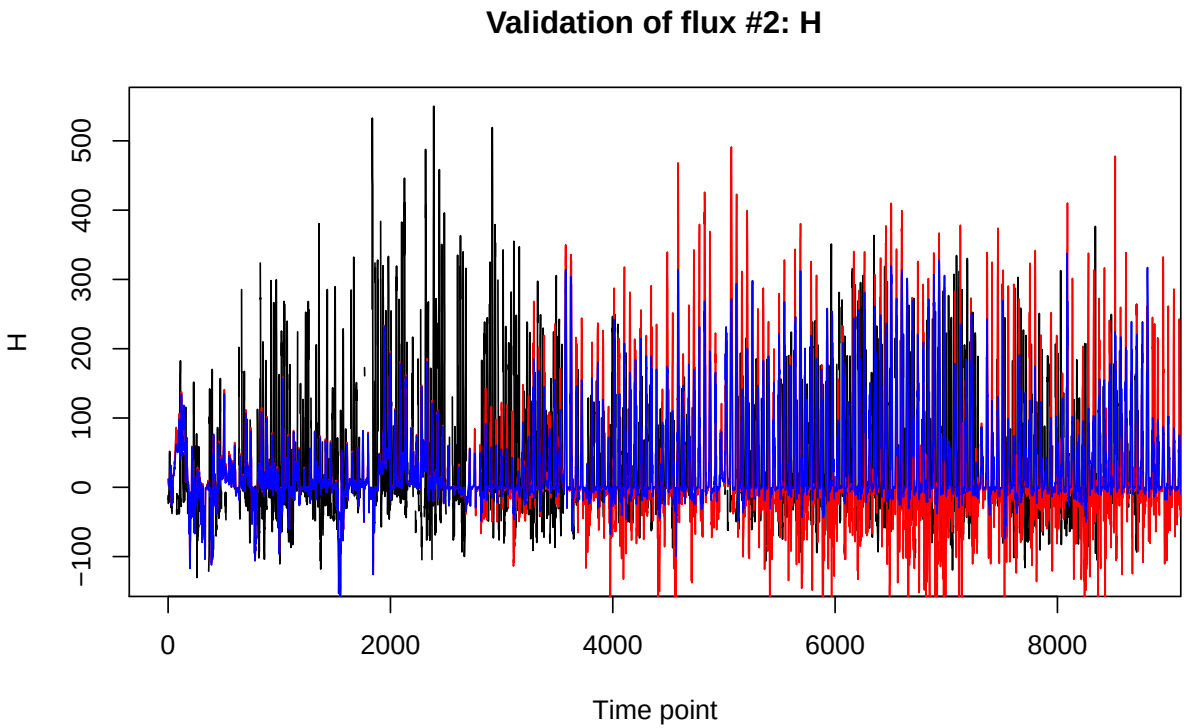
Example: Morgan–Monroe data



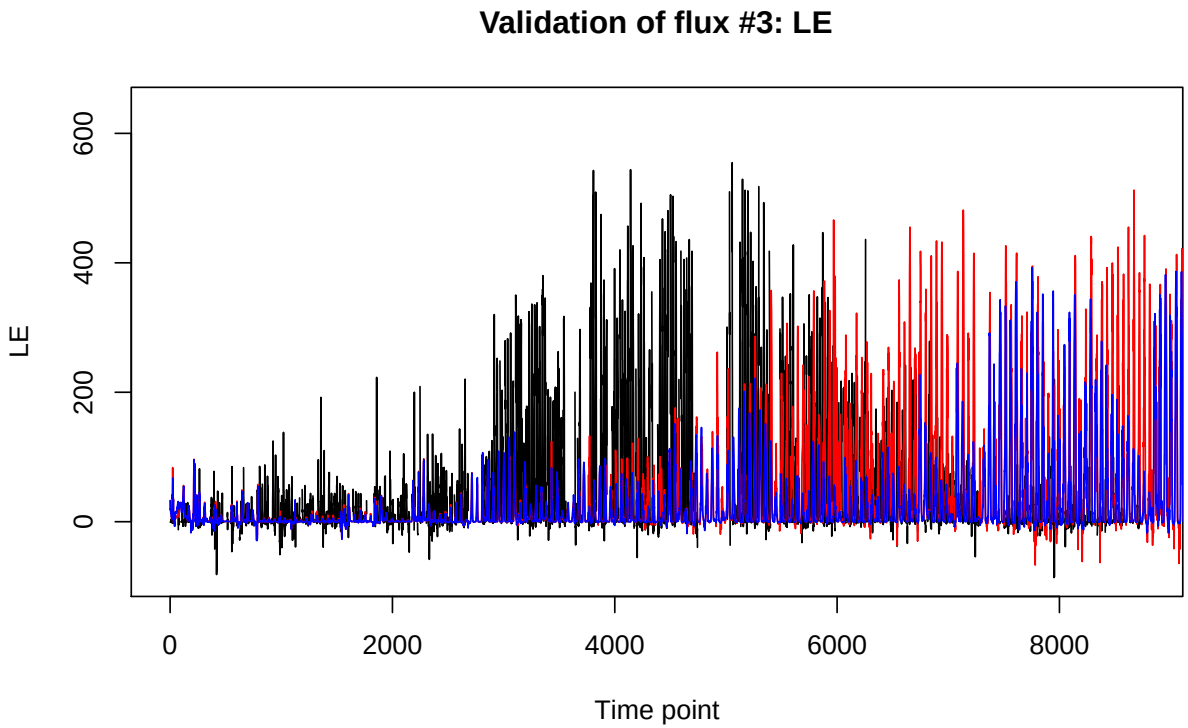
Example: Morgan–Monroe data



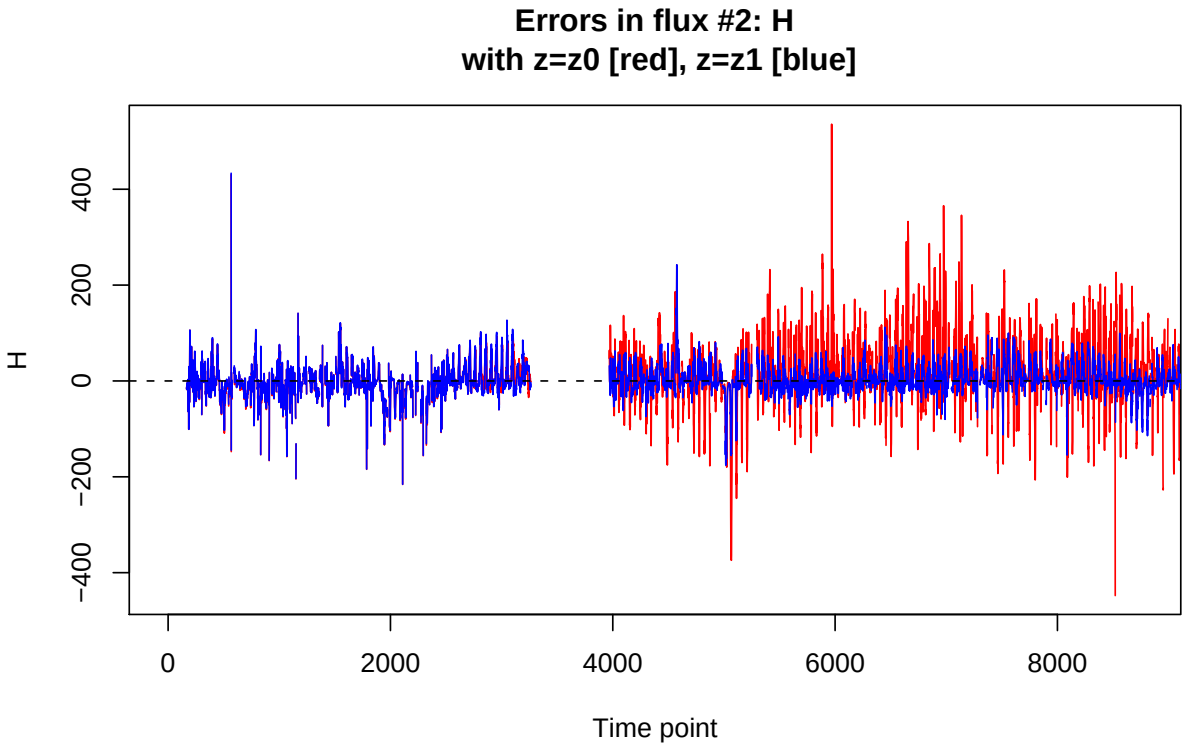
Example: Morgan–Monroe data



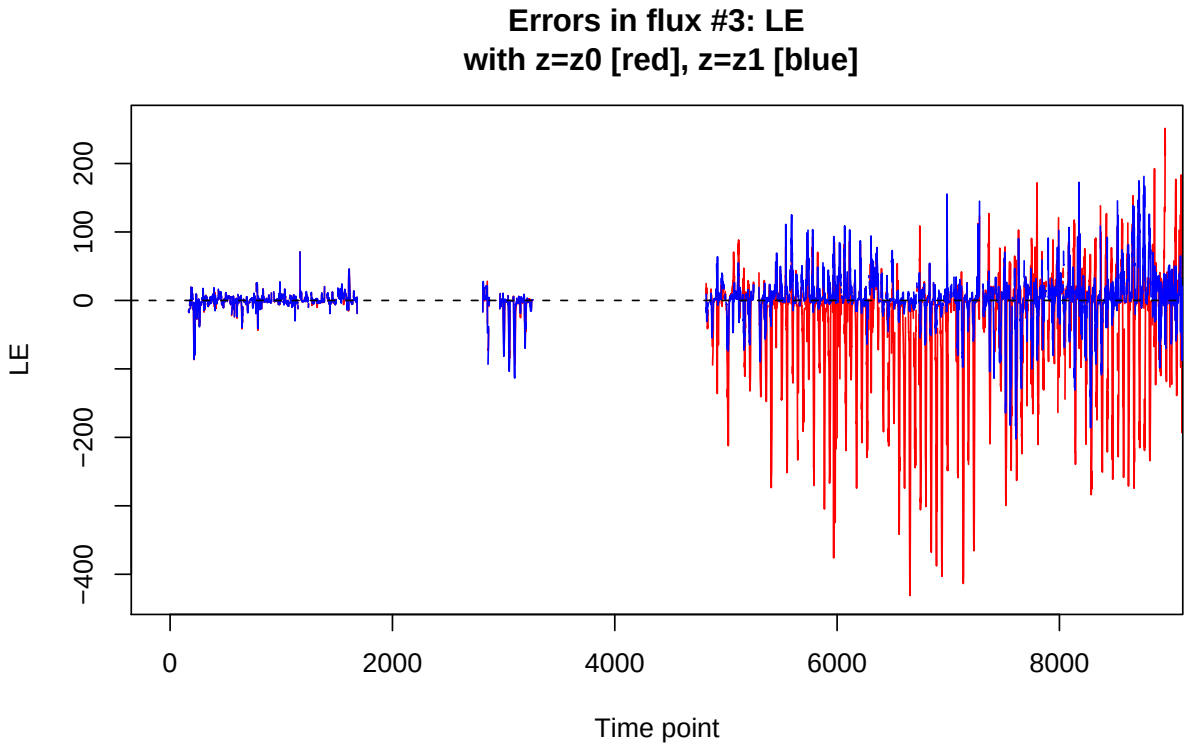
Example: Morgan–Monroe data



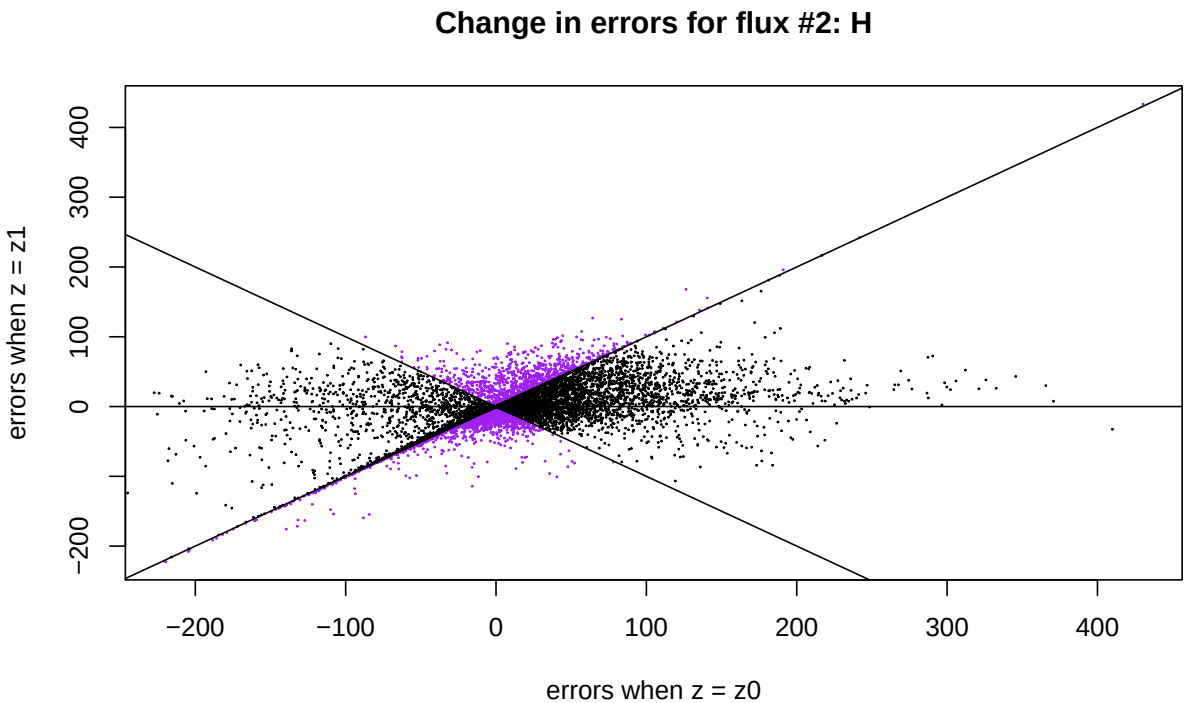
Example: Morgan–Monroe data



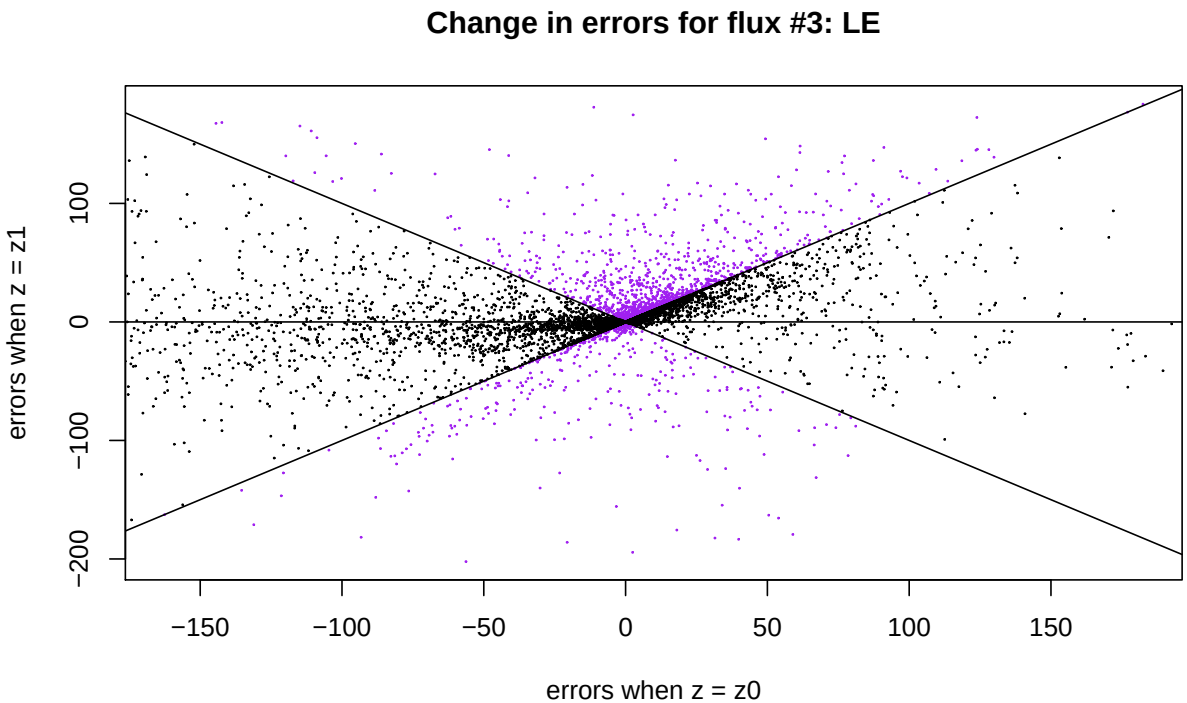
Example: Morgan–Monroe data



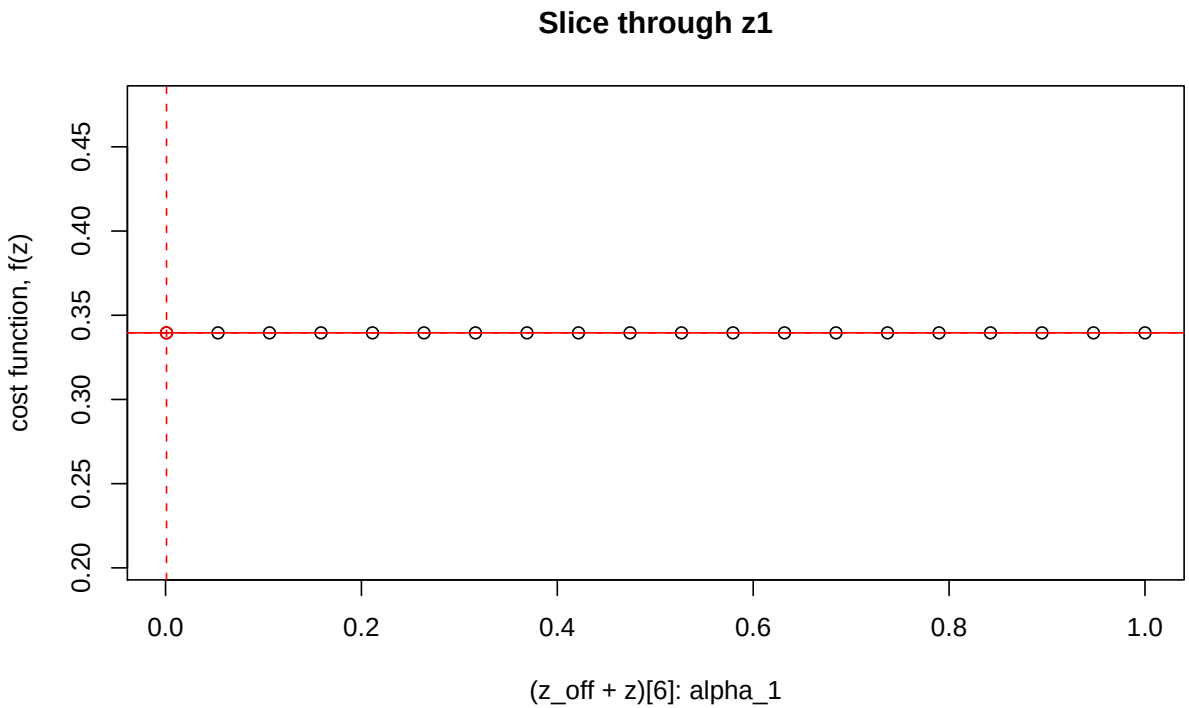
Example: Morgan–Monroe data



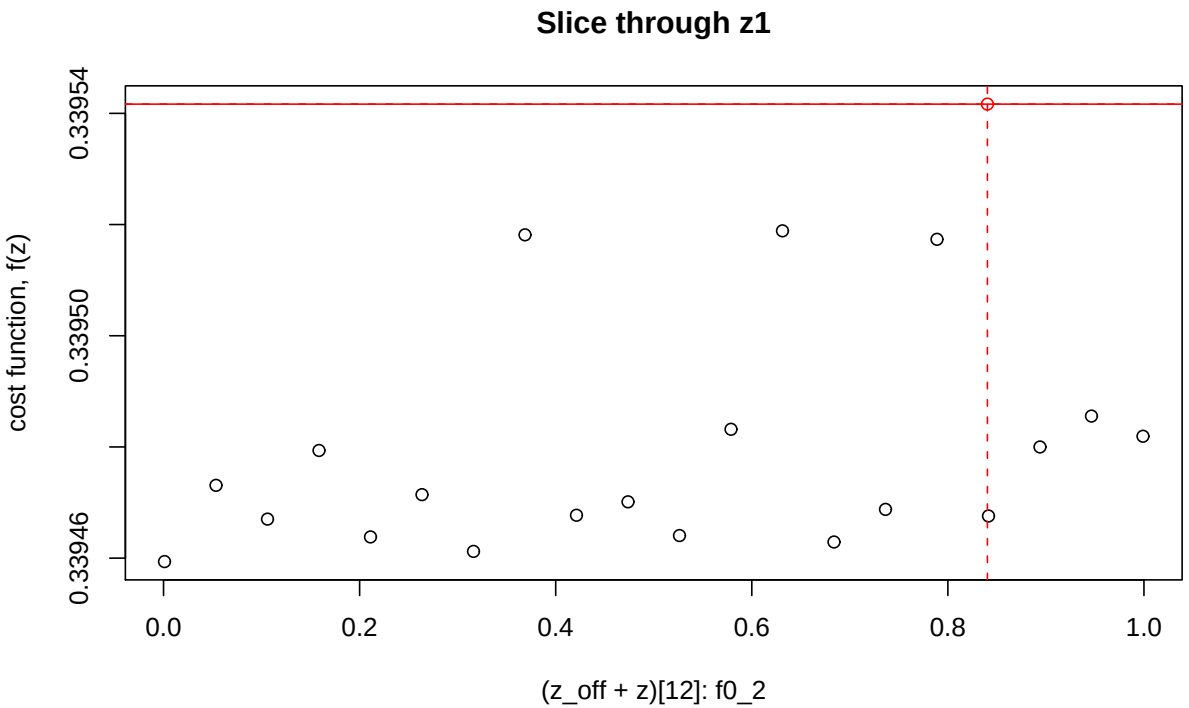
Example: Morgan–Monroe data



Example: Morgan–Monroe data



Example: Morgan–Monroe data



Outline

State–of–play

- find (locally) optimum parameters from within `R`
- successfully compiled under `ifort` and `g95`
- get fortran source from me
- get Windows–compatible executable from me
- get `R` from `r-project.org`

Issues for the future

- subject to LICENSING...
- what parameters should be added to $\mathbf{x}$?
- what 'fluxes' should be added?
- update version of JULES (currently 2.0)