High Stability Deformable Mirror for Open-Loop Applications

AO4ELT5, 27 June 2017

Urban Bitenc, Joseph Gallagher, Mickael Micallef, Sébastien Camet, Julien Charton, Tim Morris, Richard Myers

High-stability DM for open loop
Outline

• DM requirements for open loop applications
• Technology of magnetic DMs
• How to ensure stability:
  - software method 1
  - software method 2
  - improved material: silicon springs
• Results of all three methods
• Conclusions
Open-loop DM control

- Key DM requirements:
  - linearity, no hysteresis, repeatability
  - DM keeps shape over hours

Interferometer image of a flattened DM.

CLOSED LOOP

OPEN LOOP, e.g Multi-object AO

High-stability DM for open loop

Figures stolen from Chao Li et al. 2009, and modified to fit this talk.
Technology of magnetic DMs

- Spring material: polymer OR silicon

- Main features of the DM97-15
  - Pupil diameter: 13.5 mm
  - Tip/tilt stroke: 60 µm (wft)
  - 3x3 stroke: 25 µm (wft)
  - Settling time: 800 µs
  - First resonance frequency: 800 Hz
  - Hysteresis error: <2%
  - Non-linearity error: <3%
Polymer springs

- Polymer material exhibits creep (time-delayed deformation under force)
- Physical model for polymer springs (Burger model)

\[ F_1 = k \cdot x \]
\[ F_2 = \beta \cdot \frac{dx}{dt} \]
What does creep look like

- DM commands unchanged, but DM shape changes

initial DM shape       DM flattened       DM 1 hour after flattening       DM many hours later

DM commands A  |  DM commands B

- Very repeatable!
Software compensation

IDEA:

\[
B(t) = B(0) + x(t) \cdot [A - B(0)]
\]

CREEP COMPENSATION:

From RMS of DM shape

High-stability DM for open loop
Software compensation

NO COMPENSATION

WITH COMPENSATION

RMS change in 3 hours: **80 - 90 nm**

RMS change in 3 hours: **2.5 - 6 nm**

Optics Express Vol. 22, Iss. 10, pp. 12438–12451 (2014)
Software compensation - for a general use of the DM

• Change DM shape several times:

  \[ B_0 \rightarrow \Delta B_{1,0} \rightarrow B_1 \rightarrow \Delta B_{2,1} \rightarrow B_2 \rightarrow \Delta B_{3,2} \rightarrow B_3 \]

• The polymer will “remember” all these shapes

• Correct for all shapes from the past few hours:

  \[ B_N(t) = B_N(t_{N-1}) - \sum_{i=0}^{N-1} \Delta x_{i,N}(t) \cdot [B_N - B_i] \]
Performance

- Compensating after changing shape 7 times; shape differences: 330 - 1060 nm RMS

![Diagram showing DM stability with and without compensation](image)

**NO COMPENSATION:**
90 - 110 nm RMS

**WITH COMPENSATION:**
6 - 10 nm RMS

Optics Express Vol. 25, Issue 4, pp. 4368-4381 (2017)
High stability: software solution

- Creep compensation per actuator
- The creep parameters can be estimated for any DM
- A feed-forward compensation can cancel the drift

- But a simple per-actuator model is not enough
  - Mechanical coupling must be taken into account
High stability: software solution

- Mechanical coupling can be:
  - Estimated using measurements or FEM simulations
  - Represented by a stiffness matrix

- Pre-compensation implementation is simple:
  - Low-pass filtering of the command vector
  - Every second:
    - One matrix-vector multiplication
    - A few exp()
High stability: performances

- Sequence of Zernike modes (open loop)

- The software methods have comparable performance: 2% remaining instability
Spring material for high stability

- Springs in regular DM are made of polymers
- For open-loop applications **silicon** is preferred
  - No plastic domain: extremely linear
  - Extremely stable (used for springs in high-end watches)
High stability: performances

- Sequence of Zernike modes (open loop)
  - Up to $1\mu m$ peak-to-valley (270 nm RMS), over 1H30

High-stability DM for open loop
Silicon spring prototype

SHAPES TESTED (B-A): 360 - 770 nm RMS

STABILITY OF B: 1% - 2%

run7_2017Mar03
Silicon spring prototype: extreme amplitudes

RESULT: instability 1% - 2% of B-A

High-stability DM for open loop
Conclusion

• Two options for high stability DM:
  – Polymer spring + software compensation
  – Silicon springs (more expensive)

• Both: excellent performance for open loop - stability within 1%-2%

• Further ideas:
  – implement software method in drive electronics
  – combine both methods (for extremely high amplitudes)
Thank you

ALPAO

Durham University

Science & Technology Facilities Council
Impact Acceleration Account and ST/L00075X/1
High-stability DM for open loop
Silicon spring prototype

SHAPES TESTED (B-A):
360 - 770 nm RMS

STABILITY OF B:
6 - 10 nm RMS

High-stability DM for open loop
Shapes to test general creep compensation

<table>
<thead>
<tr>
<th>Shape</th>
<th>difference [nm RMS]</th>
</tr>
</thead>
<tbody>
<tr>
<td>B₀ - A</td>
<td>636</td>
</tr>
<tr>
<td>B₁ - B₀, B₁ - A</td>
<td>554, 509</td>
</tr>
<tr>
<td>B₂ - B₁, B₂ - B₀, B₂ - A</td>
<td>504, 445, 386</td>
</tr>
<tr>
<td>B₃ - B₂, B₃ - B₁, B₃ - B₀, B₃ - A</td>
<td>573, 332, 657, 563</td>
</tr>
<tr>
<td>B₄ - B₃, B₄ - B₂, B₄ - B₁, B₄ - B₀, B₄ - A</td>
<td>1058, 658, 1022, 691, 669</td>
</tr>
<tr>
<td>B₀-B₄, B₀-B₃, B₀-B₂, B₀-B₁, B₀-B₀, B₀-A</td>
<td>696, 650, 444, 553, 6, 633</td>
</tr>
<tr>
<td>C-B₀, C-B₄, C-B₃, C-B₂, C-B₁, C-B₀, C-A</td>
<td>505, 703, 656, 642, 567, 503, 478</td>
</tr>
</tbody>
</table>

- The shapes differ by 332 - 1058 nm RMS.
- Note that B₀ is used twice, at the beginning and at the end. This is to test the repeatability.
Software compensation

- Compensating the 6 intermediate shapes:

**NO COMPENSATION:**
90 - 200 nm RMS

**WITH COMPENSATION:**
7 - 13 nm RMS

High-stability DM for open loop
Calibration for software compensation

(1) Measure the correction factors:

\[ x_{i+1}(t) = x_i(t) + \frac{\text{RMS}[B_i(t) - A_i]}{\text{RMS}[B_i(0) - A_i]} - 1 \]

Correction factors
Several iterations, 12 hours each
### Software compensation - terms in the equation

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_N(t)$</td>
<td>actuator commands that will at time $t$ generate DM shape $B_N$</td>
</tr>
<tr>
<td>$B_N(t_{N-1})$</td>
<td>$= B_{N-1}(t_{N-1}) + \Delta B_{N,N-1}$, the initial value of the actuator commands that at time $t_{N-1}$ give shape $B_N$</td>
</tr>
<tr>
<td>$B_N - B_i$</td>
<td>$\sum_{j=i}^{N-1} \Delta B_{j+1,j}$, the commands that compensate for creep</td>
</tr>
<tr>
<td>$\Delta B_{j+1,j}$</td>
<td>$B_{j+1} - B_j$, user input actuator commands</td>
</tr>
<tr>
<td>$\Delta x_{i,N}(t)$</td>
<td>$x(t - t_i, t_{Bi}) - x(t_{N-1} - t_i, t_{Bi})$</td>
</tr>
<tr>
<td>$x(t - t_i, t_{Bi})$</td>
<td>Correction factor determined with a calibration. It represents the amount of creep at time $t - t_i$ if the DM had been at shape $B_i$ for $t_{Bi}$.</td>
</tr>
<tr>
<td>$t_i$</td>
<td>the point in time when the DM surface was changed from $B_i$ to $B_{i+1}$</td>
</tr>
<tr>
<td>$t_{Bi} = t_i - t_{i-1}$</td>
<td>the length of time the DM surface had shape $B_i$.</td>
</tr>
</tbody>
</table>

\[
B_N(t) = B_N(t_{N-1}) - \sum_{i=0}^{N-1} \Delta x_{i,N}(t) \cdot [B_N - B_i]
\]
Temperature

• Temperature sensor inside the DM

DM commands (closed loop) to hold actuator position:

![Graph C](image1.png)

![Graph D](image2.png)