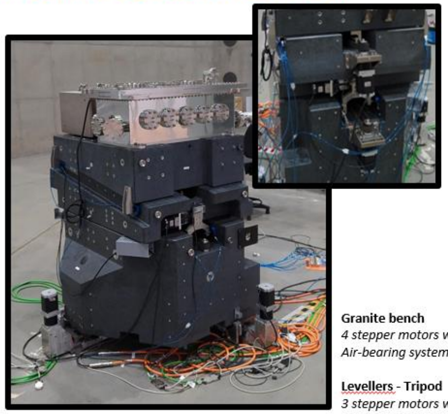
THE MIRROR SYSTEM BENCHES KINEMATICS DEVELOPMENT FOR SIRIUS LNLS

TUPV001

G. N. Kontogiorgos, L. M. Santos, L. F. Segalla
M. A. L. Moraes, A. Y. Horita
Brazilian Synchrotron Light Laboratory (LNLS), Campinas, Brazil

GRANITE BENCH OVERVIEW



Installation/1st allignment
Rough positioning onto the hutches

Commissioning
Fine positioning during beamline usage

Operation
Stripe change, focus adjustment, etc.

Power Brick controller

Granite bench
4 stepper motors with abs. feedback
Air-bearing system for movement

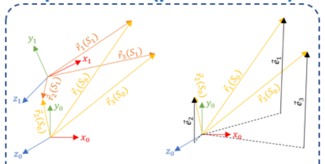
Levellers - Tripod
3 stepper motors with inc. feedback

Technologies involved

SymPy, DELTA TAU, EPICS, jupyter

FORWARD KINEMATICS MODELLING

Tripod model (parallel robot)



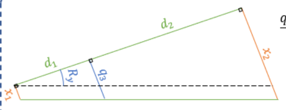
$$\vec{r}_i(S_0) = T_{levelers}(R_x, R_z, U_y^{lev})\vec{r}_i(S_1)$$

$$e_i = \hat{y}_0 \cdot \vec{r}_i(S_0) = f_i(R_x, R_z, U_y^{lev})$$

Second order approximation (Taylor expansion) and solve for R_x, R_z, U_y:

"Virtual" actuators
$$\begin{cases} R_x \cong g_1(\vec{e}_{lev}) \\ R_z \cong g_2(\vec{e}_{lev}) \\ U_y^{lev} \cong g_3(\vec{e}_{lev}) \end{cases}$$

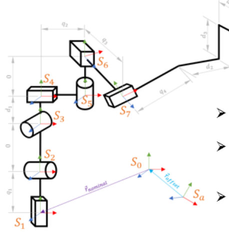
Composed actuators



$$\frac{q_3 - x_1}{d_1} = \frac{x_2 - x_1}{d_1 + d_2} \Leftrightarrow q_3 = d_1 \frac{x_2 - x_1}{d_1 + d_2} + x_1$$

$$\tan R_y = \frac{x_1 - x_2}{d_1 + d_2} \Leftrightarrow R_y = \arctan \frac{x_1 - x_2}{d_1 + d_2}$$

Full bench kinematics (serial robot)



$$\vec{r}(S_a) = T_{bench}(R_x, \quad , R_z, U_x, U_y, U_y^{lev}, U_z)\vec{r}(S_8)$$

$$\begin{cases} R_j = h_j^{rot}(\vec{e}) \\ U_j = h_j^{trans}(\vec{e}) \end{cases}$$

- *$\vec{r}(S_8)$ is the origin of S_8 referential*
- *$\vec{r}_{nominal}$ are the fixed distances (which are included in T_{bench})*
- $\vec{r}_{offset}$ are the user offsets.

INVERSE KINEMATICS AND DEPLOYMENT WORKFLOW

Applied Newton method

Define an error function $\vec{G}$ where $\vec{u} = (R_x, R_y, R_z, U_x, U_y, U_y^{lev}, U_z)$:

$$\vec{G}(\vec{e}, \vec{u}) \triangleq \vec{h}(\vec{e}) - \vec{u}$$

Calculate the Jacobian with respect to the non-linear functions

$$\mathcal{J} = \begin{pmatrix} \frac{\partial h_1}{\partial e_1} & \cdots & \frac{\partial h_1}{\partial e_7} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_7}{\partial e_1} & \cdots & \frac{\partial h_7}{\partial e_7} \end{pmatrix}$$

And finally perform the iterations

$$\vec{e}_{k+1} = \vec{e}_k + \mathcal{J}^{-1}(\vec{e}_k)\vec{G}(\vec{e}_k, \vec{u})$$

until the norm off error function is greater than one count (Just inverse case)

$$\|\vec{G}(\vec{e}, \vec{u})\| \geq 1$$

JUPYTER ENVIRONMENT

Forward equations
From geometric model

Newton method equations
Obtain error functions and Jacobian

Virtual validation
Validade inverse against forward with simulated data

Code generation
Generate C code for Power Brick based on previous generated equations

PMAC IDE

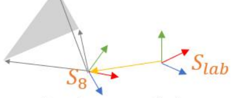
Configuration and Deployment
Power Brick files generation, compiling and downloading

Ready for operation on Power Brick
*System is ready for operation on alignment, commissioning and operation**

**For simplicity EPICS configuration is not here. However, it plays a hole importance on integrated operation.*

KINEMATICS VALIDATION

The validation was done by modelling each gauge surface as planes
- *They were obtained by 3 points each from metrology data*
- *Those points are fixed into the granite S_8 reference frame (The mirror control point)*



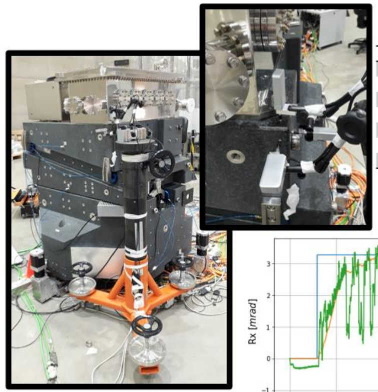
A transformation was performed to represent the planes on laboratory reference frame:

$$a_i(\vec{u})x_i + b_i(\vec{u})y_i + c_i(\vec{u})z_i + d_i(\vec{u}) = 0$$

Where $\vec{u} = (R_x, R_y, R_z, U_x, U_y, U_z)$ and i is the plane index. The gauges positions were measured by a laser tracker instrument, which gives two of three coordinates on each plane
- *The last coordinate is variable and determine the plane position on laboratory reference frame. For plane one as example:*

$$x_1 = -\frac{b_1(\vec{u})y_1^{meas} + c_1(\vec{u})z_i^{meas} + d_1(\vec{u})}{a_1(\vec{u})}$$

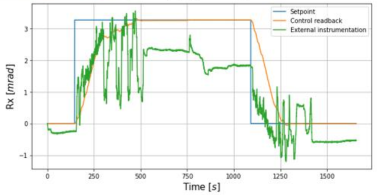
The Newton method was used to transform the gauges positions into user $\vec{u}$ coordinates on S_8 frame from the measured data during experimental movements



Post processed data from tests

Parameter	Rx	Rz
Accuracy [mrad]	3.44e-1	6.16e-1
Rx drift [mrad]	3.44e-1	2.04e1
Ry drift [mrad]	3.57e-2	5.71e-3
Rz drift [mrad]	2.87e-1	1.79e1
Tx drift* [mm]	2.94e3	1.01e1
Ty drift* [mm]	1.95e1	1.70e2
Tz drift* [mm]	8.64e1	1.52e2

** High values of drift on those axes are justified by absence of compensation by all other motors during R_x and R_x movements.* Just tripod layer was playing into this presented test.

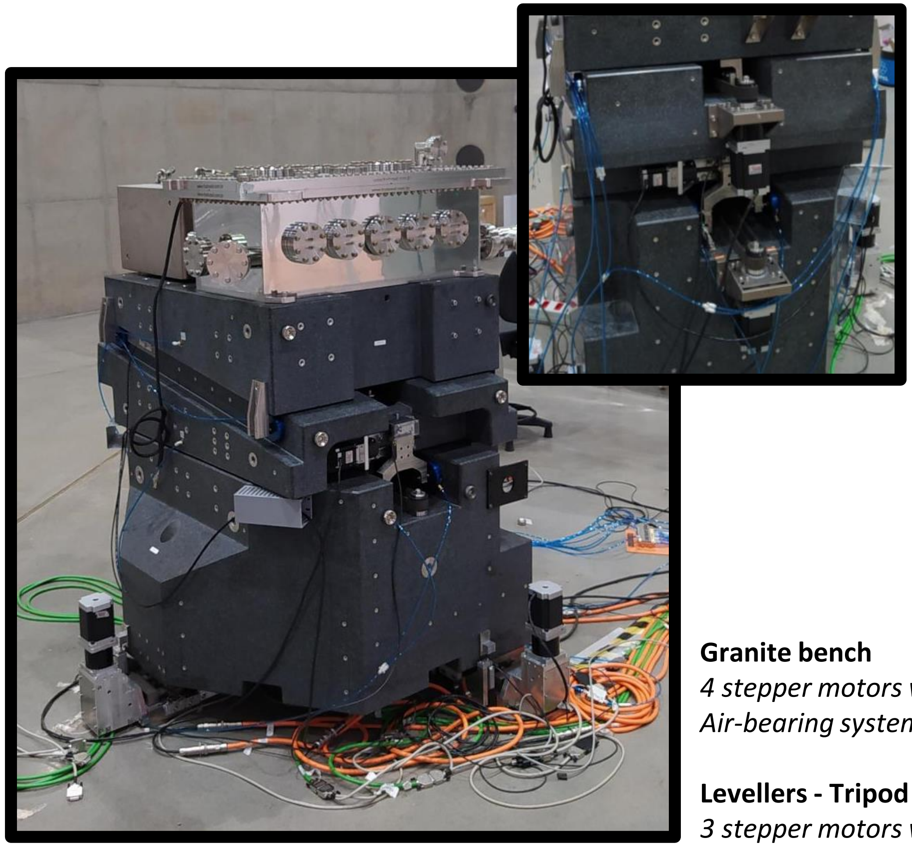


Brazilian Synchrotron Light Laboratory
CNPEM Brazilian Center for Research in Energy and Materials
MINISTRY OF SCIENCE, TECHNOLOGY AND INNOVATIONS
PÁTRIA AMADA BRASIL BRAZILIAN GOVERNMENT

THE MIRROR SYSTEM BENCHES KINEMATICS DEVELOPMENT FOR SIRIUS LNLS

G. N. Kontogiorgos, L. M. Santos, L. F. Segalla
M. A. L. Moraes, A. Y. Horita
Brazilian Synchrotron Light Laboratory (LNLS), Campinas, Brazil

GRANITE BENCH OVERVIEW



Installation/1st allignment
Rough positioning onto the hutches

Commissioning
Fine positioning during beamline usage

Operation
Stripe change, focus adjustment, etc.

Power Brick controller

High stiffness for disturbance coupling

Granite bench
4 stepper motors with abs. feedback
Air-bearing system for movement

Levellers - Tripod
3 stepper motors with inc. feedback

Technologies involved

SymPy
DELTA TAU
EPICS
jupyter

TUPV001

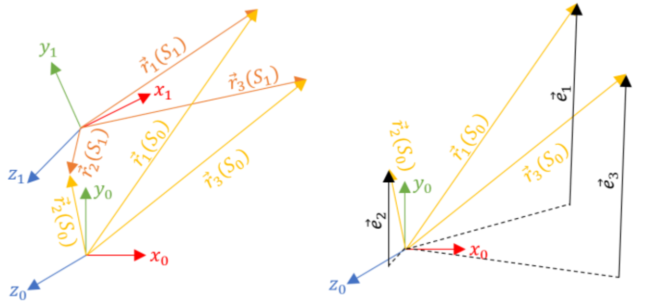
THE MIRROR SYSTEM BENCHES KINEMATICS DEVELOPMENT FOR SIRIUS LNLS

G. N. Kontogiorgos, L. M. Santos, L. F. Segalla
M. A. L. Moraes, A. Y. Horita
Brazilian Synchrotron Light Laboratory (LNLS), Campinas, Brazil

ICALEPCS

FORWARD KINEMATICS MODELLING

Tripod model (parallel robot)



$$\vec{r}_i(S_0) = T_{levelers}(R_x, R_z, U_y^{lev})\vec{r}_i(S_1)$$

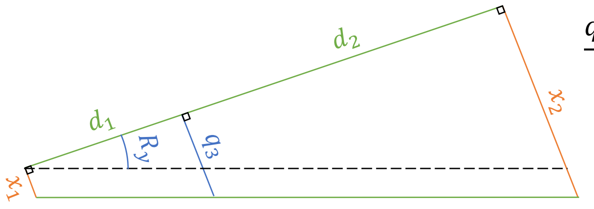
$$e_i = \hat{y}_0 \cdot \vec{r}_i(S_0) = f_i(R_x, R_z, U_y^{lev})$$

Second order approximation (Taylor expansion) and solve for R_x, R_z, U_y:

"Virtual" actuators

$$\begin{cases} R_x \cong g_1(\vec{e}_{lev}) \\ R_z \cong g_2(\vec{e}_{lev}) \\ U_y^{lev} \cong g_3(\vec{e}_{lev}) \end{cases}$$

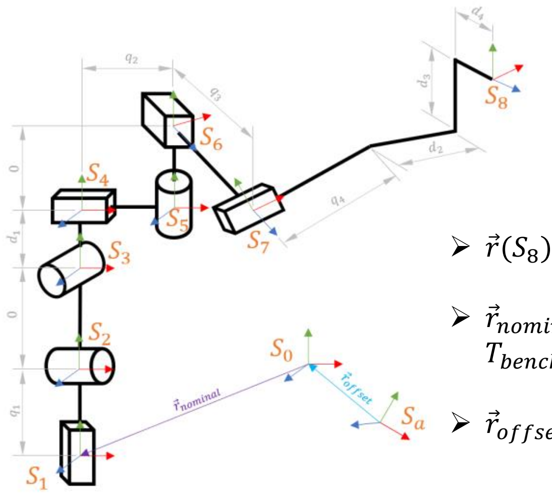
Composed actuators



$$\frac{q_3 - x_1}{d_1} = \frac{x_2 - x_1}{d_1 + d_2} \Leftrightarrow q_3 = d_1 \frac{x_2 - x_1}{d_1 + d_2} + x_1$$

$$\tan R_y = \frac{x_1 - x_2}{d_1 + d_2} \Leftrightarrow R_y = \arctan \frac{x_1 - x_2}{d_1 + d_2}$$

Full bench kinematics (serial robot)



$$\vec{r}(S_a) = T_{bench}(R_x, \quad , R_z, U_x, U_y, U_y^{lev}, U_z)\vec{r}(S_8)$$

$$\begin{cases} R_j = h_j^{rot}(\vec{e}) \\ U_j = h_j^{trans}(\vec{e}) \end{cases}$$

- *$\vec{r}(S_8)$ is the origin of S_8 referential*
- *$\vec{r}_{nominal}$ are the fixed distances (which are included in T_{bench})*
- *$\vec{r}_{offset}$* are the user offsets.

TUPV001

THE MIRROR SYSTEM BENCHES KINEMATICS DEVELOPMENT FOR SIRIUS LNLS

G. N. Kontogiorgos, L. M. Santos, L. F. Segalla
M. A. L. Moraes, A. Y. Horita
Brazilian Synchrotron Light Laboratory (LNLS), Campinas, Brazil

ICALEPCS

INVERSE KINEMATICS AND DEPLOYMENT WORKFLOW

Applied Newton method

Define an error function $\vec{G}$ *where* $\vec{u} = (R_x, R_y, R_z, U_x, U_y, U_y^{lev}, U_z)$:

$$\vec{G}(\vec{e},\vec{u}) \triangleq \vec{h}(\vec{e}) - \vec{u}$$

Calculate the Jacobian with respect to the non-linear functions

$$\mathcal{J} = \begin{pmatrix} \frac{\partial h_1}{\partial e_1} & \cdots & \frac{\partial h_1}{\partial e_7} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_7}{\partial e_1} & \cdots & \frac{\partial h_7}{\partial e_7} \end{pmatrix}$$

And finally perform the iterations

$$\vec{e}_{k+1} = \vec{e}_k + \mathcal{J}^{-1}(\vec{e}_k)\vec{G}(\vec{e}_k,\vec{u})$$

until the norm off error function is greater than one count (Just inverse case)

$$\|\vec{G}(\vec{e},\vec{u})\| \geq 1$$

JUPYTER ENVIRONMENT

Forward equations
From geometric model

Newton method equations
Obtain error functions and Jacobian

Virtual validation
Validade inverse against forward with simulated data

Code generation
Generate C code for Power Brick based on previous generated equations

PMAC IDE

Configuration and Deployment
Power Brick files generation, compiling and downloading

Ready for operation on Power Brick
*System is ready for operation on alignment, commissioning and operation**

**For simplicity EPICS configuration is not here. However, it plays a hole importance on integrated operation.*

TUPV001

THE MIRROR SYSTEM BENCHES KINEMATICS DEVELOPMENT FOR SIRIUS LNLS

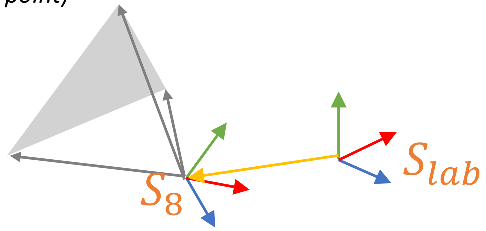
G. N. Kontogiorgos, L. M. Santos, L. F. Segalla
M. A. L. Moraes, A. Y. Horita
Brazilian Synchrotron Light Laboratory (LNLS), Campinas, Brazil

ICALEPCS

KINEMATICS VALIDATION

The validation was done by modelling each gauge surface as planes

- *They were obtained by 3 points each from metrology data*
- *Those points are fixed into the granite S_8 reference frame (The mirror control point)*



A transformation was performed to represent the planes on laboratory reference frame:

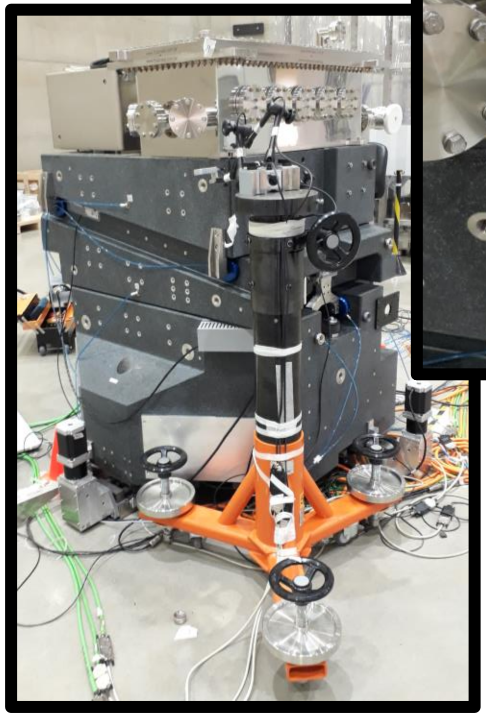
$$a_i(\vec{u})x_i + b_i(\vec{u})y_i + c_i(\vec{u})z_i + d_i(\vec{u}) = 0$$

Where $\vec{u} = (R_x, R_y, R_z, U_x, U_y, U_z)$ *and* i *is the plane index. The gauges positions were measured by a laser tracker instrument, which gives two of three coordinates on each plane*

- *The last coordinate is variable and determine the plane position on laboratory reference frame. For plane one as example:*

$$x_1 = -\frac{b_1(\vec{u})y_1^{meas} + c_1(\vec{u})z_i^{meas} + d_1(\vec{u})}{a_1(\vec{u})}$$

The Newton method was used to transform the gauges positions into user $\vec{u}$ *coordinates on* S_8 *frame from the measured data during experimental movements*



Post processed data from tests

Parameter	Rx	Rz
Accuracy [mrad]	3.44e-1	6.16e-1
Rx drift [mrad]	3.44e-1	2.04e1
Ry drift [mrad]	3.57e-2	5.71e-3
Rz drift [mrad]	2.87e-1	1.79e1
Tx drift* [mm]	2.94e3	1.01e1
Ty drift* [mm]	1.95e1	1.70e2
Tz drift* [mm]	8.64e1	1.52e2

** High values of drift on those axes are justified by absence of compensation by all other motors during* R_x and R_x movements. Just tripod layer was playing into this presented test.