

MTAOS add aberrations to M1M3+M2+hexapod

This notebook is used for the level 3 integration tests from test plan LVV-P81 (<https://jira.lsstcorp.org/secure/Tests.jspa#/testPlan/LVV-P81>) as part of test cycle LVV-C176 (<https://jira.lsstcorp.org/secure/Tests.jspa#/testCycle/LVV-C176>). The following tests are currently run as part of this notebook:

- LVV-T2190 (<https://jira.lsstcorp.org/secure/Tests.jspa#/testCase/LVV-T2190>)

Execution steps are separated by horizontal lines. Upon completion, save the notebook and its output as a pdf file to be attached to the test execution in JIRA.

Last updated by E. Dennihy 20211020

Load all the needed libraries. Get the remotes ready Code in the notebook including section: "Check the summary state of each CSC".

```
In [1]: %load_ext autoreload
        %autoreload 2
```

```
In [2]: import rubin_jupyter_utils.lab.notebook as nb
        nb.utils.get_node()
```

```
/tmp/ipykernel_21244/1665379685.py:2: DeprecationWarning: Call to deprecated function (or staticmethod) get_node. (Please use lsst.rsp.get_node())
      nb.utils.get_node()
```

```
Out[2]: 'yagan03'
```

```
In [3]: import os
        import sys
        import asyncio
        import logging

        import pandas as pd
        import numpy as np

        from matplotlib import pyplot as plt

        from lsst.ts import salobj
        from lsst.ts.observatory.control.maintel import MTCS, ComCam
        from lsst.ts.observatory.control import RotType
```

```
| lsst.ts.utils.tai INFO: Update leap second table
| lsst.ts.utils.tai INFO: current_tai uses the system TAI clock
```

```
In [4]: logging.basicConfig(format="%(name)s: %(message)s", level=logging.DEBUG)
```

```
In [5]: log = logging.getLogger("setup")
log.level = logging.DEBUG
```

```
In [6]: domain = salobj.Domain()
```

```
In [7]: mtcs = MTCS(domain=domain, log=log)
mtcs.set_rem_loglevel(40)
```

```
|setup.MTCS DEBUG: mtmount: Adding all resources.
|setup.MTCS DEBUG: mptg: Adding all resources.
|setup.MTCS DEBUG: mtaos: Adding all resources.
|setup.MTCS DEBUG: mtm1m3: Adding all resources.
|setup.MTCS DEBUG: mtm2: Adding all resources.
|setup.MTCS DEBUG: mthexapod_1: Adding all resources.
|setup.MTCS DEBUG: mthexapod_2: Adding all resources.
|setup.MTCS DEBUG: mtrotator: Adding all resources.
|setup.MTCS DEBUG: mtdome: Adding all resources.
|setup.MTCS DEBUG: mtdometrajectory: Adding all resources.
```

```
In [8]: await mtcs.start_task
```

```
|MTHexapod INFO: Read historical data in 0.06 sec
|MTHexapod INFO: Read historical data in 0.06 sec
|MTHexapod.electrical WARNING: electrical DDS read queue is filling: 11 of 100 elements
```

```
Out[8]: [None, None, None, None, None, None, None, None, None, None]
```

```
|MTHexapod.application WARNING: application DDS read queue is filling: 11 of 100 elements
|MTHexapod.actuators WARNING: actuators DDS read queue is filling: 12 of 100 elements
```

Ready M1M3: Raise mirror, turn on FB, clear forces

Need to have M1M3 LUT use its inclinometer.

Ready M2: Turn on FB, clear forces

Need to have M2 LUT use its inclinometer

Get camera hexapod ready: check config; make sure LUT is on, and has valid inputs; make sure hex is at LUT position

Get M2 hexapod ready: check config; make sure LUT is on, and has valid inputs; make sure hex is at LUT position

Slew to the next target. Choose a target such that the rotator stays within a couple of degrees of its initial position. This is because the CCW is not running (MTmount in simulation mode).

```
In [9]: target = await mtcs.find_target(el=60, az=120, mag_limit=8)
        print(target)
```

HD 54207

```
In [25]: await mtcs.slew_object(target, rot_type=RotType.PhysicalSky, rot=1.9)
```

```
setup.MTCS INFO: Slewing to HD 54207: 07 06 05.6698 -40 57 01.564
setup.MTCS DEBUG: Setting rotator physical position to 1.9 deg. Rotator will track sky.
setup.MTCS WARNING: Camera cable wrap following disabled in MTMount.
setup.MTCS DEBUG: Wait 5.0s for rotator to settle down.
setup.MTCS DEBUG: Workaround for rotator trajectory problem. Moving rotator to its current position: 1.88
setup.MTCS DEBUG: Wait for MTRotator in position event.
setup.MTCS DEBUG: MTRotator in position: False.
setup.MTCS INFO: MTRotator in position: True.
setup.MTCS DEBUG: MTRotator in position True. Waiting settle time 5.0s
setup.MTCS DEBUG: Sending slew command.
setup.MTCS DEBUG: Scheduling check coroutines
setup.MTCS DEBUG: process as completed...
setup.MTCS DEBUG: Monitor position started.
setup.MTCS DEBUG: Waiting for Target event from mtmount.
setup.MTCS DEBUG: mtmount: <State.ENABLED: 2>
setup.MTCS DEBUG: mtptg: <State.ENABLED: 2>
setup.MTCS DEBUG: mtaos: <State.ENABLED: 2>
setup.MTCS DEBUG: mtm1m3: <State.ENABLED: 2>
setup.MTCS DEBUG: mtm2: <State.ENABLED: 2>
setup.MTCS DEBUG: mthexapod_1: <State.ENABLED: 2>
setup.MTCS DEBUG: mthexapod_2: <State.ENABLED: 2>
setup.MTCS DEBUG: mtrotator: <State.ENABLED: 2>
setup.MTCS DEBUG: mtdome: <State.ENABLED: 2>
setup.MTCS DEBUG: mtdometrajectory: <State.ENABLED: 2>
setup.MTCS DEBUG: Wait for mtmount in position events.
setup.MTCS DEBUG: Wait for dome in position event.
setup.MTCS DEBUG: Wait for MTRotator in position event.
setup.MTCS DEBUG: MTRotator in position: True.
setup.MTCS DEBUG: MTRotator already in position. Handling potential race condition.
setup.MTCS DEBUG: Wait for MTMount elevation in position event.
setup.MTCS DEBUG: MTMount elevation in position: True.
setup.MTCS DEBUG: MTMount elevation already in position. Handling potential race condition.
setup.MTCS DEBUG: Wait for MTMount azimuth in position event.
setup.MTCS DEBUG: MTMount azimuth in position: True.
```

```

setup.MTCS DEBUG: MTMount azimuth already in position. Handling potential race condition.
setup.MTCS INFO: MTMount azimuth in position: False.
setup.MTCS INFO: MTMount elevation in position: False.
setup.MTCS DEBUG: Mount target: private_revCode: bdcba00ba, private_sndStamp: 1648588384.0342007, private_rcvStamp: 1648588384.0342414, private_seqNum: 5439, private_identity: MTMount, private_origin: 5948, elevation: 61.99201618036403, elevationVelocity: 0.0030808430407704155, azimuth: 121.37228504958487, azimuthVelocity: 0.0014273329514333927, taiTime: 1648588384.0927546, trackId: 4, tracksys: SIDERREAL, radesys: ICRS, priority: 0
setup.MTCS INFO: MTRotator in position: False.
setup.MTCS DEBUG: [Tel]: Az = +000.041[+121.3]; El = +089.979[ -28.0] [Rot]: +001.885[ -0.0] [Dome] Az = +000.000; El = +000.000
setup.MTCS DEBUG: Dome azimuth in position.
setup.MTCS DEBUG: Dome elevation in position.
setup.MTCS INFO: MTRotator in position: True.
setup.MTCS DEBUG: MTRotator in position True. Waiting settle time 3.0s
setup.MTCS DEBUG: [Tel]: Az = +047.000[ +74.4]; El = +066.557[ -4.5] [Rot]: +002.038[ +0.0] [Dome] Az = +000.000; El = +000.000
setup.MTCS INFO: MTMount elevation in position: True.
setup.MTCS DEBUG: MTMount elevation in position True. Waiting settle time 3.0s
setup.MTCS DEBUG: [Tel]: Az = +092.203[ +29.2]; El = +062.031[ +0.0] [Rot]: +002.014[ +0.0] [Dome] Az = +000.000; El = +000.000
setup.MTCS INFO: MTMount azimuth in position: True.
setup.MTCS DEBUG: MTMount azimuth in position True. Waiting settle time 3.0s
setup.MTCS DEBUG: [Tel]: Az = +121.399[ -0.0]; El = +062.050[ -0.0] [Rot]: +001.990[ +0.0] [Dome] Az = +000.000; El = +000.000

```

add 1um of z7 to the system via OFC

Compare the corrections sent vs forces and position changes applied. This is currently done in a separate notebook.

```
In [11]: wavefront_errors = np.zeros(19)
```

```
In [12]: wavefront_errors[3] += 1.0 # add 1 um to z7
```

```
In [13]: await mtcs.rem.mtaos.cmd_addAberration.set_start(wf=wavefront_errors, timeout=1)
```

```
Out[13]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f5c5d6c6100>
```

This command primes the corrections, the issueCorrection command is needed to actually command them to be sent

```
In [14]: await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)
```

```
Out[14]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f5cd871f250>
```

Make plots using telemetry from each component to verify the changes in the DOFs. This step does not currently involve running any commands in this notebook. This step must be verified using a separate notebook.

reset the corrections using the resetCorrection command

Compare the corrections sent vs forces and position changes applied (these are all expected to be zero). This is currently done in a separate notebook or on Chronograf.

```
In [15]: await mtcs.rem.mtaos.cmd_resetCorrection.start()
```

```
Out[15]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f5c5d6e7490>
```

```
In [16]: await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)
```

```
Out[16]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f5c884a1310>
```

add 2um of z7 to the system via OFC

Compare the corrections sent vs forces and position changes applied. This is currently done in a separate notebook or on Chronograf.

```
In [17]: wavefront_errors[3] = 2.0 # add 2.0 um of z7
```

```
In [18]: await mtcs.rem.mtaos.cmd_addAberration.set_start(wf=wavefront_errors, timeout=1
```

```
Out[18]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f5c87f8ba60>
```

```
In [19]: await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)
```

```
Out[19]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f5cd81a68e0>
```

Stop Tracking

```
In [26]: await mtcs.stop_tracking()
```

```
| setup.MTCS DEBUG: Stop tracking.
```

Check that the corrections in step 10 are twice of those in step 7. This step does not currently involve running any commands in this notebook. This step must be verified using a separate notebook.

Wrap up. Put each component to the following states: mtaos --> standby m1m3 --> lower mirror --> standby m2 --> standby camera hex --> standby m2 hex --> standby

```
In [27]: await mtcs.set_state(salobj.State.STANDBY, components=["mtaos"])

|setup.MTCS DEBUG: [mtaos]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
|setup.MTCS INFO: All components in <State.STANDBY: 5>.
```

```
In [29]: await mtcs.lower_m1m3()
```

```
-----
RuntimeError                                Traceback (most recent call last)
Input In [29], in <cell line: 1>()
----> 1 await mtcs.lower_m1m3()

File ~/auto-op-env-packages/ts_observatory_control/python/lssst/ts/observatory/
control/maintel/mtcs.py:672, in MTCS.lower_m1m3(self)
    670 async def lower_m1m3(self):
    671     """Lower M1M3."""
--> 672     await self._execute_m1m3_detailed_state_change(
    673         execute_command=self._handle_lower_m1m3,
    674         initial_detailed_states={
    675             MTM1M3.DetailedState.ACTIVE,
    676             MTM1M3.DetailedState.ACTIVEENGINEERING,
    677         },
    678         final_detailed_states={
    679             MTM1M3.DetailedState.PARKED,
    680             MTM1M3.DetailedState.PARKEDENGINEERING,
    681         },
    682     )

File ~/auto-op-env-packages/ts_observatory_control/python/lssst/ts/observatory/
control/maintel/mtcs.py:726, in MTCS._execute_m1m3_detailed_state_change(self,
execute_command, initial_detailed_states, final_detailed_states)
    722     self.log.info(
    723         f"M1M3 current detailed state {final_detailed_states!r}. Nothi
ng to do."
    724     )
    725 else:
--> 726     raise RuntimeError(
    727         f"M1M3 detailed state is {MTM1M3.DetailedState(m1m3_detailed_s
tate.detailedState)!r}, "
    728         f"expected {initial_detailed_states!r}. "
    729         "Cannot execute command."
    730     )

RuntimeError: M1M3 detailed state is <DetailedState.LOWERING: 8>, expected {<D
etailedState.ACTIVEENGINEERING: 11>, <DetailedState.ACTIVE: 7>}. Cannot execut
e command.
|setup.MTCS DEBUG: mtm1m3: <State.ENABLED: 2>
|setup.MTCS DEBUG: M1M3 detailed state 5
```

```
In [37]: await mtcs.set_state(salobj.State.OFFLINE, components=["mtm1m3"])

|setup.MTCS DEBUG: [mtm1m3]::[<State.STANDBY: 5>, <State.OFFLINE: 4>]
|setup.MTCS INFO: All components in <State.OFFLINE: 4>.
```

```
In [33]: await mtcs.set_state(salobj.State.STANDBY, components=["mtm2"])

| setup.MTCS DEBUG: [mtm2]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <State.
```

```
STANDBY: 5>]
```

```
| setup.MTCS INFO: All components in <State.STANDBY: 5>.
```

```
In [34]: await mtcs.set_state(salobj.State.STANDBY, components=["mthexapod_1"])

| setup.MTCS DEBUG: [mthexapod_1]::[<State.ENABLED: 2>, <State.DISABLED: 1>,
| <State.STANDBY: 5>]
```

```
| setup.MTCS INFO: All components in <State.STANDBY: 5>.
```

```
In [35]: await mtcs.set_state(salobj.State.STANDBY, components=["mthexapod_2"])

| setup.MTCS DEBUG: [mthexapod_2]::[<State.ENABLED: 2>, <State.DISABLED: 1>,
| <State.STANDBY: 5>]
```

```
| setup.MTCS INFO: All components in <State.STANDBY: 5>.
```

```
In [36]: await mtcs.standby()

| setup.MTCS DEBUG: [mtmount]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <Sta
| te.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mtptg]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <Stat
| e.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mtaos]::[<State.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mtm1m3]::[<State.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mtm2]::[<State.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mthexapod_1]::[<State.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mthexapod_2]::[<State.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mtrotator]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <S
| tate.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mtdome]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <Stat
| e.STANDBY: 5>]
```

```
| setup.MTCS DEBUG: [mtdometrajectory]::[<State.ENABLED: 2>, <State.DISABLED:
| 1>, <State.STANDBY: 5>]
```

```
| setup.MTCS INFO: All components in <State.STANDBY: 5>.
```

```
In [ ]:
```