Automated Resource-Constrained Science Planning for the MiRaTA Mission

Andrew “Kit” Kennedy,
Anne Marinan, Kerri L. Cahoy, James Byrne, Timothy Cordeiro, Zachary Decker,
Weston Marlow, Stephen Shea (MIT STAR Lab / Space Systems Lab)

William J. Blackwell, Michael DiLiberto, R. Vincent Leslie,
Idahosa Osaretin, Erik Thompson (MIT Lincoln Laboratory)

Rebecca Bishop (The Aerospace Corporation)

SmallSat 2015 : VI-Ground/Communications
August 11, 2015
Outline

• Purpose and Motivation

• Approach
  – Modeling MiRaTA Operations
  – The Resource-Aware SmallSat Planner (RASP)

• Constellation Simulation Results
  – Simulation Context
  – Varying Planning Window Length
  – Varying Resource Weighting Factors
  – Time Performance

• Conclusion
Outline

• **Purpose and Motivation**

• **Approach**
  – Modeling MiRaTA Operations
  – The Resource-Aware SmallSat Planner (RASP)

• **Constellation Simulation Results**
  – Simulation Context
  – Varying Planning Window Length
  – Varying Resource Weighting Factors
  – Time Performance

• **Conclusion**
The MiRaTA Mission

**MiRaTA : Microwave Radiometer Technology Acceleration**
- Funding sources: NASA ESTO – InVEST and NOAA
- 3U CubeSat
- Microwave radiometer and modified GPS receiver
- Launch November 2016

**Follow-on to MicroMAS CubeSat**
- Microsized Microwave Atmospheric Satellite
- Deployed from ISS March 2015

**Mission Goals:**
- Flight test new ultra-compact and low-power radiometer
- Flight test new GPS receiver and phased antenna array
- Demonstrate novel radiometer calibration through GPS Radio Occultation (RO) measurements

Marlow et al. [1]
Slide details: [2,3,4]
MiRaTA Calibration Maneuver

Nominal Sci Ops for Coupled Atmospheric GPSRO & Microwave Radiometry

~ 20 minute maneuver
0.5° / sec rate

Modified from Blackwell, Cahoy, Bishop, et al [2]
MiRaTA Science Planning

• **Calibration Maneuver goal:**
  – Execute 100 successful maneuvers…
  – In 90 day primary mission

• **Constraints:**
  – Limited ground contacts:
    • One primary ground station at NASA Wallops
    • 2 sets of 3 short passes per day
  – Maneuver windows don’t overlap passes
  – Limited onboard energy storage
  – Large science data production:
    greater than 60 Mbits per maneuver!

Develop and test an algorithm for automated, onboard science operations planning for MiRaTA
MiRaTA Science Planning

- **Calibration Maneuver goal:**
  - Execute 100 successful maneuvers…
  - In 60 day primary mission

- **Constraints:**
  - Limited ground contacts:
    - One primary ground station at NASA Wallops
    - 2 sets of 3 short passes per day
  - Maneuver windows don't overlap passes
  - Limited onboard energy storage
  - Large science data production:
    - greater than 60 Mbits per maneuver!

BIG FAT NOTE!

This algorithm is not currently an official part of the MiRaTA design

Develop and test an algorithm for automated, onboard science operations planning for MiRaTA

*Slide details: [2,3,4]*
Outline

• Purpose and Motivation

• Approach
  – Modeling MiRaTA Operations
  – The Resource-Aware SmallSat Planner (RASP)

• Constellation Simulation Results
  – Simulation Context
  – Varying Planning Window Length
  – Varying Resource Weighting Factors
  – Time Performance

• Conclusion
The 5 Ingredients

1. Orbital Geometry

2. MiRaTA Ops Model

3. Planning Algorithm

4. Activity Timeline

5. Replanning!
Modeling MiRaTA Operations
MiRaTA Operations Model

Perform Maneuvers and... Manage resources

Slew

Sun

Recharge

Maneuver

Downlink

Ground Station

Target Location

Time

SmallSat 2015 VI-I – AKK – a kennedy@mit.edu
Onboard Resource Constraints

- **Energy**
  - 20 W-Hr lithium polymer batteries
  - < 30 % depth of discharge desired
  - Maximum 24.8W solar panel input

- **Data management**
  - 2.6 Mbps downlink rate
  - NASA Wallops Ground station
  - Data storage limited to 100 MiB

- **Attitude Control**
  - Active 3-axis control
  - Attitude slews necessary between most activities

*MiRaTA Top Face [3]*

*GPS RO Antenna Array  Solar Panel (x4)*

*Monopole Antenna (x2)*

*Slide details: [2,3,4]*
Operational State Machine

- Maneuver
- Slew
- Idle
- Recharge
- Downlink

- Science
- Transition Activities
- Resource Management Activities
Operational State Machine

Managing Resource Storage:

- Energy Storage (ES)
- Data Storage (DS)

- Recharge
- Downlink

Science
Transition Activities
Resource Management Activities

Managing Resource Storage:

- Energy Storage (ES)
- Data Storage (DS)

- Recharge
- Downlink
Operational State Machine

Maneuver
Min Duration: 10 min.
Power: -15.1 W
Data rate: +73 kbps

Science
Transition Activities
Resource Management Activities

Recharge
Downlink
The Resource-Aware SmallSat Planner (RASP)
Activity Sequencing

• Can derive an activity sequence from:
  – Orbital geometry and activity windows
  – Operational state machine

• But…
  – how do we choose the timing of activities?
  – how do we meet resource constraints?

• What’s in the black box?....
The Black Box: RASP Algorithm

- Automated process of finding consistent activity timeline
- Algorithm:
  1. Construct **initial activity sequence** of maneuver activities
  2. Try to **assign timings** to activities in sequence
     - 3. If success, **stop**
     - 4. If fail, attempt to **add management activity** to sequence
  5. Repeat steps 2 to 4 until timeout
  6. If timeout, **modify initial sequence** by subtracting one maneuver
  7. Repeat 2 to 6 until solved. If no solution, **relax resource bounds**

---

**Initial Activity Sequence**

<table>
<thead>
<tr>
<th>i-1</th>
<th>Orbit i</th>
<th>Orbit i+1</th>
<th>i+2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Man.</td>
<td>Man.</td>
<td>Man</td>
</tr>
</tbody>
</table>
Depth First Search

Slews and Idles added

Orbit i

Idle Slew Man. Idle Slew Man. Idle Slew

After a while…

Slews and Idles added

Rech

Downlink

Rech

Downlink

Rech

Downlink

Rech

Downlink

Rech

…

…

…

…

After a while…

Step 2: Assign Timings

Mixed Integer Linear Program, Score Function:

\[
\max \left[ \sum_{\text{Man}} (t_a^E - t_a^S) - w_d \sum_{\text{Dlnk}} \dot{d}_i (t_a^E - t_a^S) + w_e \sum_{i=1}^{N} \sum_{j=1}^{i} \dot{e}_i (t_a^E - t_a^S) \right]
\]
Step 2: Assign Timings

Mixed Integer Linear Program, Score Function:

$$\max \left[ \sum_{Man} (t^E_a - t^S_a) - w_d \sum_{Dlnk} \dot{d}_i (t^E_a - t^S_a) + w_e \sum_{i=1}^{N} \sum_{j=1}^{i} \dot{e}_i (t^E_a - t^S_a) \right]$$

Maximize total time spent in maneuver activities
Step 2: Assign Timings

Mixed Integer Linear Program, Score Function:

\[
\max \left[ \sum_{\text{Man}} \left( t^E_a - t^S_a \right) - w_d \sum_{\text{Dlnk}} \dot{d}_i \left( t^E_a - t^S_a \right) + w_e \sum_{i=1}^{N} \sum_{j=1}^{i} \dot{e}_i \left( t^E_a - t^S_a \right) \right]
\]

Maximize total time spent in maneuver activities

Minimize increase in data storage over sequence
Step 2: Assign Timings

Mixed Integer Linear Program, Score Function:

\[
\max \left[ \sum_{Man} (t^E_a - t^S_a) - w_d \sum_{Dlnk} d_i (t^E_a - t^S_a) + w_e \sum_{i=1}^{N} \sum_{j=1}^{i} \dot{e}_i (t^E_a - t^S_a) \right]
\]

- Maximize total time spent in maneuver activities
- Minimize increase in data storage over sequence
- Maximize energy storage margin over sequence
Step 2: Assign Timings

Mixed Integer Linear Program:  
\[
\max \left[ \sum_{Man} (t^E_a - t^S_a) - w_d \sum_{Dlnk} d_i (t^E_a - t^S_a) + w_e \sum_{i=1}^{N} \sum_{j=1}^{i} e_i (t^E_a - t^S_a) \right]
\]

Subject to constraints:
Step 2: Assign Timings

Mixed Integer Linear Program:

$$\max \left[ \sum_{Man} (t^E_a - t^S_a) - w^d \sum_{Dlnk} d_i (t^E_a - t^S_a) + w^e \sum_{i=1}^{N} \sum_{j=1}^{i} \dot{e}_i (t^E_a - t^S_a) \right]$$

Subject to constraints:

Activity Timing

$$t^S_{a,1} = 0$$

$$\forall j = i + 1,$$

$$t^S_{a,j} = t^E_{a,i}$$

$$i \geq 1, j \leq K$$

$$t^E_{a,K} = t_{\text{horizon}}$$

$$t^S_{a,i} \leq t^E_{a,i} \quad \forall i \geq 1, i \leq K$$

$$t = 0$$

$$t = t_{\text{horizon}}$$
Step 2: Assign Timings

Mixed Integer Linear Program:  
\[
\max \left[ \sum_{Man} (t^E_a - t^S_a) - \sum_{Dlink} d_i (t^E_a - t^S_a) + \sum_{i=1}^{N} \sum_{j=1}^{i} \dot{e}_i (t^E_a - t^S_a) \right]
\]

Subject to constraints:

Activity Timing
\[
t^S_{a,1} = 0
\]
\[
t^S_{a,j} = t^E_{a,i} \quad \forall j = i + 1,
\]
\[
t^E_{a,K} = t_{\text{horizon}}
\]
\[
t^S_{a,i} \leq t^E_{a,i} \quad \forall i \geq 1, i \leq K
\]

Man. Timing*
\[
t^S_{o,i} \geq t^S_{o,i} \quad \forall i \geq 1, i \leq K
\]
\[
t^E_{o,i} \geq t^E_{o,i}
\]
\[
t^E_{o,i} - t^S_{o,i} \geq LB_{o,i}
\]

*Similar for Dlink, Recharge, Slew

\[
\text{Obs Window i}
\]
\[
\text{Obs i}
\]
\[
LB_{o,i}
\]
Step 2: Assign Timings

Mixed Integer Linear Program: 
\[
\max \left[ \sum_{\text{Man}} (t^E - t^S) - w_d \sum_{\text{Dlink}} d_i (t^E - t^S) + w_e \sum_{i=1}^{N} \sum_{j=1}^{i} \hat{e}_i (t^E - t^S) \right]
\]

Subject to constraints:

Activity Timing
\[
t^S_{a,1} = 0 \\
\forall j = i + 1, i \geq 1, j \leq K \\
t^S_{a,j} = t^E_{a,i} \\
t^E_{a,K} = t_{\text{horizon}} \\
t^S_{a,i} \leq t^E_{a,i} \quad \forall i \geq 1, i \leq K
\]

Resource Upper Bounds
**
\[
ES_{\text{init}} + \hat{e}_1 (t^E_{a,1} - t^S_{a,1}) \leq UB_{ES} + M \left( 1 - z_{ES,1} \right)
\]
\[
ES_{\text{init}} + \sum_{i=1}^{2} \hat{e}_i (t^E_{a,i} - t^S_{a,i}) \leq UB_{ES} + M \left( 1 - z_{ES,2} \right)
\]
\[
\vdots
\]
\[
ES_{\text{init}} + \sum_{i=1}^{K} \hat{e}_i (t^E_{a,i} - t^S_{a,i}) \leq UB_{ES} + M \left( 1 - z_{ES,N} \right)
\]

Man. Timing* 
\[
t^S_{o,i} \geq t^S_{o,i} \\
\forall i \geq 1, i \leq K \\
t^E_{o,i} \geq t^E_{o,i} \\
t^E_{o,i} - t^S_{o,i} \geq LB_{o,i}
\]

*Similar for Dlink, Recharge, Slew

** z values are 0 or 1, M is large, similar for Data Storage
Step 2: Assign Timings

**Mixed Integer Linear Program:** 
\[
\text{max} \left[ \sum_{\text{Man}} (t^E - t^S) - w_d \sum_{\text{Dlink}} d_i (t^E - t^S) + w_e \sum_{i=1}^{N} \sum_{j=1}^{i} \dot{e}_i (t^E - t^S) \right]
\]

Subject to constraints:

- Activity Timing
  - \( t^S_{a,1} = 0 \)
  - \( t^S_{a,j} = t^E_{a,i} \) \( \forall j = i + 1, \)
  - \( t^E_{a,K} = t_{\text{horizon}} \)
  - \( t^S_{a,i} \leq t^E_{a,i} \) \( \forall i \geq 1, i \leq K \)

- Man. Timing
  - \( t^S_{o,i} \geq t^S_{W,o,i} \) \( \forall i \geq 1, i \leq K \)
  - \( t^E_{o,i} \geq t^E_{W,o,i} \) \( \forall i \geq 1, i \leq K \)
  - \( t^E_{o,i} - t^S_{o,i} \geq \text{LB}_{o,i} \)

**Resource Upper Bounds**
- \( \text{ES}_{\text{init}} + \dot{e}_1 (t^E_{a,1} - t^S_{a,1}) \leq \text{UB}_{ES} + M(1 - z^\text{UB}_{ES,1}) \)
- \( \text{ES}_{\text{init}} + \sum_{i=1}^{2} \dot{e}_i (t^E_{a,i} - t^S_{a,i}) \leq \text{UB}_{ES} + M(1 - z^\text{UB}_{ES,2}) \)
- ... 
- \( \text{ES}_{\text{init}} + \sum_{i=1}^{K} \dot{e}_i (t^E_{a,i} - t^S_{a,i}) \leq \text{UB}_{ES} + M(1 - z^\text{UB}_{ES,N}) \)

**Resource Lower Bounds**
- \( \text{ES}_{\text{init}} + \dot{e}_1 (t^E_{a,1} - t^S_{a,1}) \geq \text{LB}_{ES} - M(1 - z^\text{LB}_{ES,1}) \)
- \( \text{ES}_{\text{init}} + \sum_{i=1}^{2} \dot{e}_i (t^E_{a,i} - t^S_{a,i}) \geq \text{LB}_{ES} + M(1 - z^\text{LB}_{ES,2}) \)
- ... 
- \( \text{ES}_{\text{init}} + \sum_{i=1}^{K} \dot{e}_i (t^E_{a,i} - t^S_{a,i}) \geq \text{LB}_{ES} + M(1 - z^\text{LB}_{ES,N}) \)

**z values are 0 or 1, M is large, similar for Data Storage**
Outline

• Purpose and Motivation

• Approach
  – Modeling MiRaTA Operations
  – The Resource-Aware SmallSat Planner (RASP)

• Constellation Simulation Results
  – Simulation Context
  – Varying Planning Window Length
  – Varying Resource Weighting Factors
  – Time Performance

• Conclusion
Simulation Context

• Ran RASP in 24 hour simulation
  – RASP implemented in MATLAB, simulation wrapper in Python
  – Replanning time = 20 minutes
  – Varying planning window length ($t_h$), resource weighting factors ($w_e, w_d$)

• Steps
  1. Orbit data obtained using AGI’s STK program
  2. Orbit data used to determine activity windows
  3. RASP creates first plan from initial state
  4. Propagate state forward and replan at fixed interval

• Performance investigations
  – Maneuver and Resource Performance with varying $t_h$
  – Maneuver and Resource Performance with varying $w_e, w_d$
  – Execution time performance with varying $t_h$
Varying $t_h$: Maneuver Execution

Fewer maneuvers are executed with a longer planning window. Seems counterintuitive!
$t_h = 30$ minutes

Multiple ES lower bound violations
$t_h = 120$ minutes

No ES bound violations

Suggests small planning window lengths are too “short-sighted”
Zero ES Weighting

$\tau_h = 120$ minutes

$w_e = 0$

- Many, large ES lower bound violations
- Also, few maneuvers executed

RASP performs poorly when ignoring ES
Zero DS Weighting

t_h = 120 minutes

w_d = 0

- DS saturates – no more room for data
- Also, few maneuvers executed

RASP performs poorly when ignoring DS
Varying $t_h$: Time Performance

- 72 plans generated by RASP over every simulation run
- Accumulated RASP run time and averaged over 72 plans
- Run on 2 GHz Quad-Core CPU with 8 GB of RAM

<table>
<thead>
<tr>
<th>Mode</th>
<th>Planning Window Length (mins)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>30</td>
</tr>
<tr>
<td>Average Time (seconds)</td>
<td>0.66</td>
</tr>
</tbody>
</table>

All times under a minute!
With a capable processor, RASP runs relatively quickly

*from MATLAB
Outline

• Purpose and Motivation

• Approach
  – Modeling MiRaTA Operations
  – The Resource-Aware SmallSat Planner (RASP)

• Constellation Simulation Results
  – Simulation Context
  – Varying Planning Window Length
  – Varying Resource Weighting Factors
  – Time Performance

• Conclusion
Simulation Results

- **Initial demonstration of an automated science planning algorithm for MiRaTA:**
  - Designed to run onboard spacecraft
  - Interleaves resource management activities with science activities
  - Relatively little interaction with ground: initial input of activity windows

- **Findings from simulation cases:**
  1. **RASP planning window can’t be too short**
     - Robust performance at $t_h = 120$ mins, but ES bounds violations with $t_h = 30$
     - However, performance was anomalous (worse) at $t_h = 240$ minutes
  2. **RASP performance degrades significantly when ES and DS margin is ignored**
     - With $w_e = 0$, multiple ES violations, ave. ES margin decreases to 42%
     - With $w_d = 0$, DS saturates, ave. DS margin decreases to 34%
  3. **RASP executes relatively quickly**
     - At $t_h = 120$ minutes, average execution time ~9 seconds
     - Promising for ARM-based SmallSat computer (~1 GHz, 1 GB RAM, 2.5W)
Limitations and Future Work

- **Limitations:**
  - Only a single activity can be scheduled at a time
  - No treatment of data latency in score function
  - Energy production and consumption are not modeled separately
    - all energy production assumed in recharge mode

- **Future work:**
  - Improve fidelity of MiRaTA operational model
  - Adapt planner to real flight software environment
    - Port planner to C, test on representative computer
    - Figure out strategy for mixing planning with plan execution
  - Develop better model of downlink communications link
Acknowledgements

• This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. 1122374. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

• This work is also supported by NASA Earth Science Technology Office grant number NNX14AC75G and NASA Space Technology Research Grant NNX12AM30H.
Contact

- Andrew Kitrell Kennedy, a kennedy@mit.edu
References


Backup
Maneuver Timing

• **Design Orbit**
  – Sun-synchronous, 97.2° inclination
  – 450 x 810 km altitude, LTAN: 13:25

• **Maneuver Windows**
  – Fixed by geometry of MiRaTA and GPS orbits
  – Found from overlap of:
    • Radiometer field of view
      (174-176° az., -20-0° el.)
    • GPS trajectory
  – Relaxed elevation restriction for this work

*Maneuver windows (blue) from a 24 hour period [3,4]*