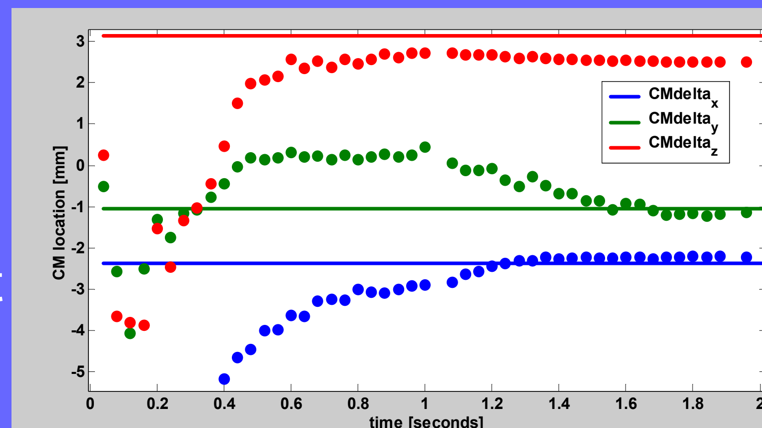
On-line, gyro-based mass-property identification for thruster-controlled spacecraft using recursive least squares

Edward Wilson, Chris Lages, Robert Mah

ed.wilson@intellization.com
chris.lages@intellization.com
rmah@mail.arc.nasa.gov

Research objective: For thruster-controlled spacecraft, **identify mass properties** (I, CM) using gyros, under normal vehicle control. Applicable for **adaptive control** or FDI. Develop and validate through application on realistic simulations and hardware.

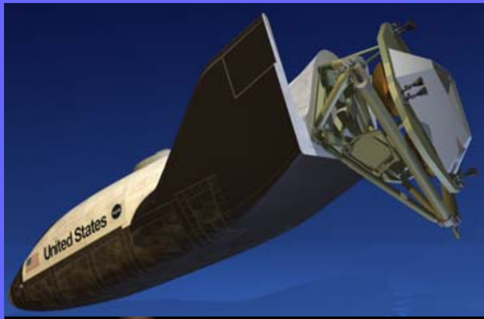


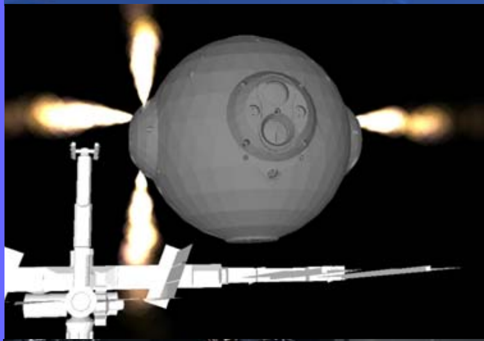
Outline:

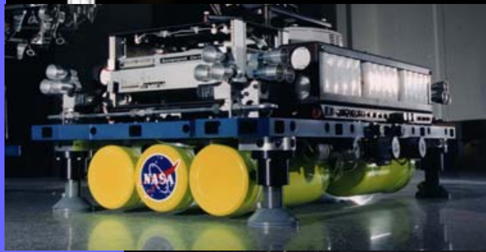
- Introduction:
  - Problem statement
  - LS ID
- Solution:
  - Approach
  - Algorithms
  - Results, Demo
- Conclusions

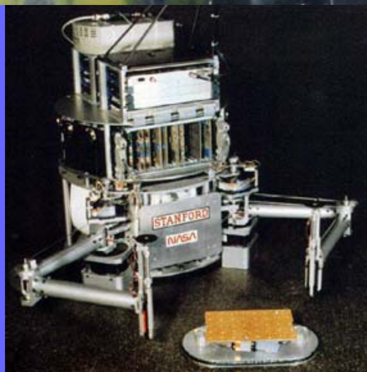
Introduction: Mass-property ID

- Problem statement:
  - For a thruster-controlled spacecraft, with relatively low rotation rates, realistic sensor noise models, realistic thrust variability, using gyros only
  - ID mass center and inertia matrix
  - For use with adaptive control, Fault detection and isolation (FDI)
- Related Research:
  - Tanygin and Williams (1997) – spinning, coasting, LS
  - Bergmann *et al* (1987) – Kalman filter
  - Wilson and Rock (1994) – RLS combined thruster/mass ID; used for on-line neural-network control reconfiguration following multiple thruster failures









Least-squares identification (LS ID)

- Cast governing equations into form $\boldsymbol{Ax} = \boldsymbol{b} + \varepsilon$
- Noise appears in ε
- Parameters to ID appear (linearly) in $\boldsymbol{x}$
- Closed form solution minimizes sum squared error: $\hat{x} = \left(A^T W A\right)^{-1} A^T W b$
- Batch or equivalent recursive solutions (RLS)

- *Challenge is in manipulating governing equations into correct form,* $\boldsymbol{Ax} = \boldsymbol{b} + \varepsilon$

Problem characteristics / Approach

- Full dynamics involve:
  - Thruster strength and alignment
  - Inertia matrix
  - CM location, Mass
- Variability:
  - Pulse-to-pulse thruster variation
  - Sensor noise
  - Disturbance forces and torques
- Parameters appear in governing equations of motion (EOM) coupled, nonlinear
- Approach: divide into separate approximate linear solutions
- Separate RLS IDs for inertia, CM, thruster strength

Mass-center ID algorithm

Equations of motion:

$$\dot{\omega} = I^{-1}((L \times D)B(F_{nom} + F_{bias} + F_{random,k})T_k + \tau_{disturb} - \omega \times (I\omega))$$

Manipulated EOM:

$$C \equiv C_{nom} + \Delta; L = L_{nom} - \Delta[1 \quad 1 \quad \ldots \quad 1]$$

$$I^{-1}\begin{bmatrix} 0 & -c_3 & c_2 \\ c_3 & 0 & -c_1 \\ -c_2 & c_1 & 0 \end{bmatrix}_k \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix} = \dot{\omega} + I^{-1}(\omega \times (I\omega)) - I^{-1}(L_{nom} \times D)F_{nom}T_k; c_k \equiv DF_{nom}T_k$$

LS (or RLS) formulation: $A_k x = b_k$

$$A_k = I^{-1}\begin{bmatrix} 0 & -c_3 & c_2 \\ c_3 & 0 & -c_1 \\ -c_2 & c_1 & 0 \end{bmatrix}_k ; x = \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{bmatrix}; b_k = \dot{\omega} + I^{-1}(\omega \times (I\omega)) - I^{-1}(L_{nom} \times D)F_{nom}T_k$$

(inverse) Inertia ID algorithm

Equations of motion:

$$\dot{\omega} = I^{-1}((L \times D)B(F_{nom} + F_{bias} + F_{random,k})T_k + \tau_{disturb} - \omega \times (I\omega))$$

Manipulated EOM:

$$\begin{bmatrix} a_1 & & & a_2 & a_3 & \\ & a_2 & & a_1 & & a_3 \\ & & a_3 & & a_1 & a_2 \end{bmatrix} \begin{bmatrix} I_{11}^{-1} \\ I_{22}^{-1} \\ I_{33}^{-1} \\ I_{12}^{-1} \\ I_{13}^{-1} \\ I_{23}^{-1} \end{bmatrix} = \dot{\omega}; a_k \equiv (L \times D)F_{nom}T_k - \omega \times (I\omega)$$

LS (or RLS) formulation:

$$A_k x = b_k \qquad A_k = \begin{bmatrix} a_1 & & & a_2 & a_3 & \\ & a_2 & & a_1 & & a_3 \\ & & a_3 & & a_1 & a_2 \end{bmatrix}; x = \begin{bmatrix} I_{11}^{-1} \\ I_{22}^{-1} \\ I_{33}^{-1} \\ I_{12}^{-1} \\ I_{13}^{-1} \\ I_{23}^{-1} \end{bmatrix}; b_k = \dot{\omega}$$

RLS, batch LS solution

- RLS implementation: use $\boldsymbol{A_k}$, $\boldsymbol{b_k}$ at each update, either exponentially weighted or unweighted. Use standard RLS equations.
- Batch LS: concatenate $\boldsymbol{A_k}$ matrices and $\boldsymbol{b_k}$ vectors, using any desired weighting

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_k \end{bmatrix}; b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix}$$

- Solve using standard batch LS solution

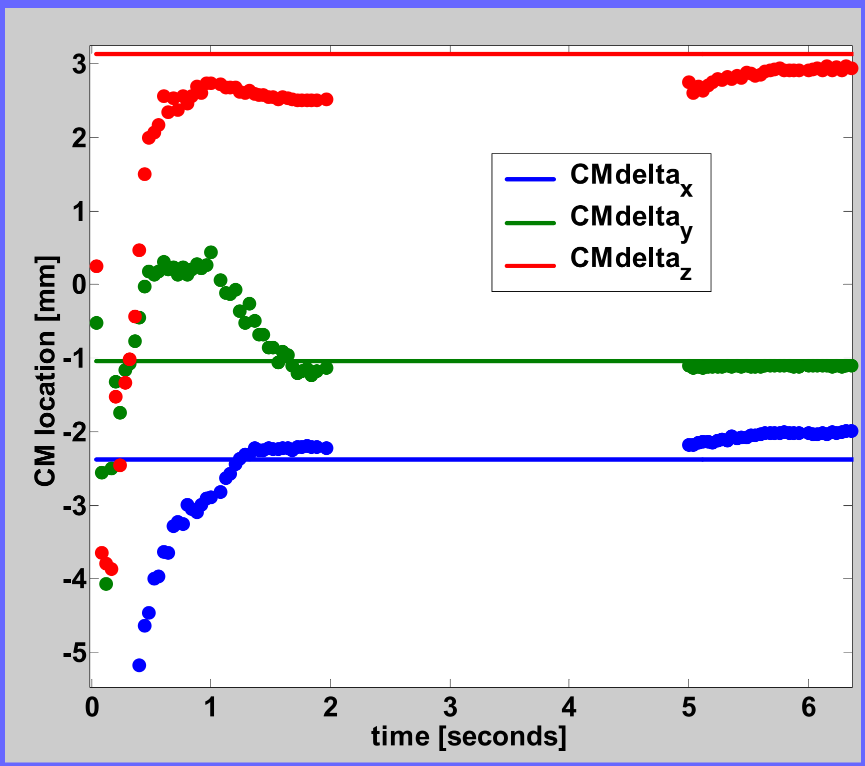
$$\hat{x} = \left(A^T W A\right)^{-1} A^T W b \qquad \hat{x} = \left(A^T A\right)^{-1} A^T b$$

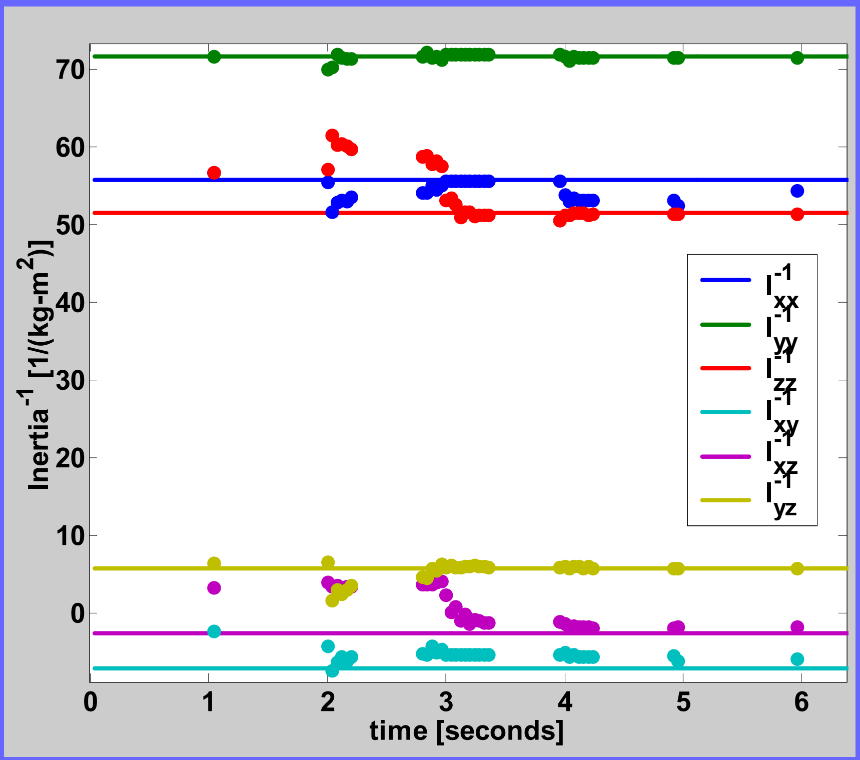
Deviations from correct LS form

- $\hat{x} = \left(A^T A\right)^{-1} A^T b$ is optimal if $\boldsymbol{Ax = b + \varepsilon}$ form can be achieved.
- Not strictly possible due to form of EOM. Depending on rates, disturbances, sensor accuracy, thruster variability, control policy, etc., different formulations may be better.
- Deviations from correct LS form:
  - Noisy measurements appear in the $\omega \times (I\omega)$ term in the $\boldsymbol{A}$ matrix. Negligible for slow rotational speeds in many spacecraft applications.
  - Other terms in $\boldsymbol{A}$ and $\boldsymbol{b}$ are not known perfectly: $\boldsymbol{L}$, $\boldsymbol{D}$, $\boldsymbol{B}$, $\boldsymbol{F_{bias}}$, *etc*. are all estimated or nominal values.
  - Random variables $\boldsymbol{F_{random,k}}$ and $\tau_{disturb}$ (set to zero) do not appear directly in the ε term as they should.
  - CM ID uses nominal or estimated values for $\boldsymbol{I}$ and $\boldsymbol{I^{-1}}$. Inertia ID uses nominal or estimated values for CM.

Results

- Accuracy depends on sensor noise, thruster variability, variability in non-ID'ed parameters.
- Applied to 3 vehicles (X-38, Mini-AERCam, S4) in simulation, being applied in hardware on S4 (same code).

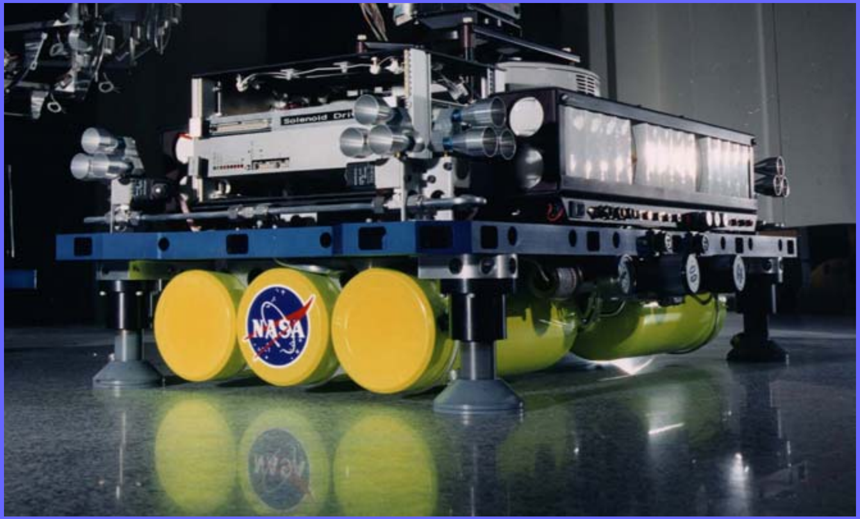




- More accurate than ground analysis/meas. (Mini-AERCam)
- MATLAB demo

Extensions, continuing work

- Use of translational accelerometers
- Integration of on-line mass-property ID with FDI
- Implementation on air-bearing vehicle
  - Same MATLAB code runs on X-38 sim, Mini-AERCam sim, S4 sim, S4 hardware
- Standing by for X-38, Mini-AERCam programs



Smart Systems Spacecraft Simulator (S4)

Conclusions

- Algorithms presented provide mass-property ID for thruster-controlled spacecraft
- Non-invasive – uses existing gyros, no special motions required
- Generic algorithm - applied to 3 vehicles in simulation, 1 in laboratory hardware
- Useful for adaptive control, FDI, especially applicable to vehicles with changing payload, fuel mass, configuration

- Paper and presentation are available at http://intellization.com/files/

Acknowledgements

- Funded by NASA Headquarters, HQ AA, PWC 349-00: William Readdy and Gary Martin
- Problem definition from NASA JSC: Rodolfo Gonzalez, Dr. Steven Fredrickson, Tim Straube, Dave Hammen
- NASA Ames SSRL members help on 3D visualization: Richard Papasin, Alan Gasperini, Rommel Del Mundo