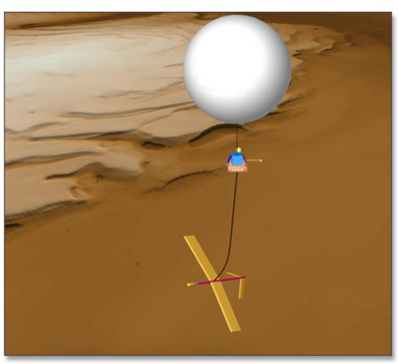
SLAM On High Speed Platforms

Salah Sukkarieh
Australian Centre for Field Robotics
The University of Sydney

Future Planetary Airborne Missions



First Response Units

Wombats

What's required?

- An ability to provide 6 DoF platform information (proprioception):
  - 3D position
  - 3D velocity,
  - attitude (roll and pitch) and
  - heading.
  - Without the need for knowledge of the dynamics or kinematics of the platform.
- An ability to provide 3D positional information of the features in the environment (exteroception):
  - Without prior knowledge of the feature's spatial or temporal properties, or any man made infrastructure to help with determination of localisation.
  - Build up knowledge of the feature characteristics and properties in real time.
  - (3D Attitude?)
  - (incorporation of known temporal properties of features?)
- Provide this information at high update rates
  - Monitoring highly dynamic systems
  - Control

What's required?

- Localisation Robustness and Integrity
  - Must be able to accept various forms of exteroception information (discrete such as an observation of a feature, as well as continuous such as nature's magnetic field or the magnetic field of the insides of a building, or relative electromagnetic range information).
  - Must be able to accommodate for loss of exteroception information
  - Must be able to constrain drift errors with proprioception information
- An understanding of how exteroceptive data affects proprioceptive data
  - Determine where to look next (data association, mapping, improving localisation).
  - Sharing of feature information between platforms

What's required?

- Must not rely on extroceptive sensors alone.
- Must be computationally expensive or slow.
- Cannot fall over when dealing with large environments (many features, large state space)
- Inexpensive proprio/extero-ceptive sensors
- Must not have significant data lag in localisation estimates
- Must not be based on heuristics or rules of thumb as it needs to be verified (and signed off).

What becomes apparent

- Requires inertial technology
- Requires an extroceptive sensor (at least one) which measures environmental properties
- Requires stochastic filtering technology which:
  - Fuses information from both
  - Centres around the inertial mechanisation
  - Builds up proprioceptive and exteroceptive knowledge simultaneously

What is inertial technology?

SLAM On High Speed Platforms 2006 SLAM Summer School

- Accelerometers

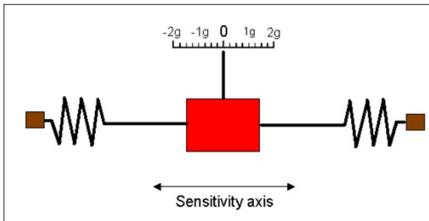
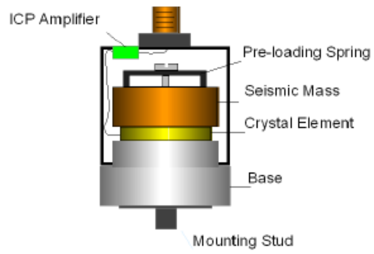
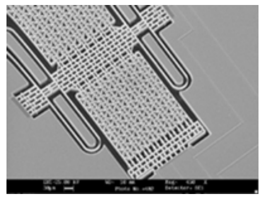


Fig. 1. Schematic structure of an accelerometer

www.rotoview.com/accelerometer.htm

www.dliengineering.com

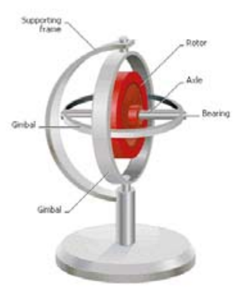


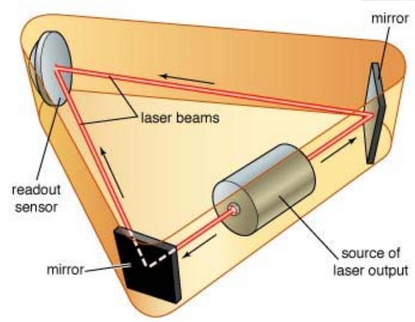


What is inertial technology?

SLAM On High Speed Platforms 2006 SLAM Summer School

- Gyroscopes



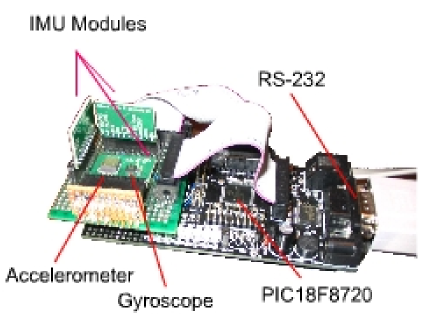


What is inertial technology?

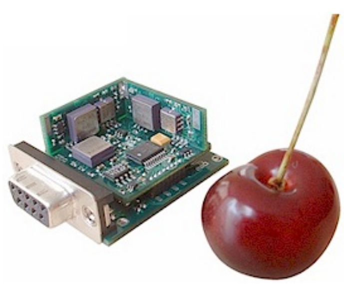
- Inertial Sensor Assembly

Honeywell Unit

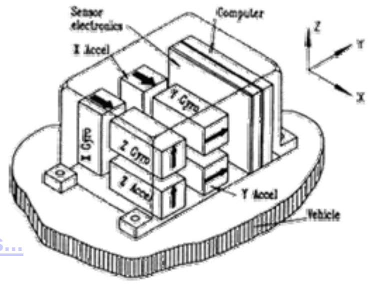
- **Inertial Measurement Units**
- www.engr.uvic.ca/~ksakaki/project/imu.php



- CloudCap Technologies

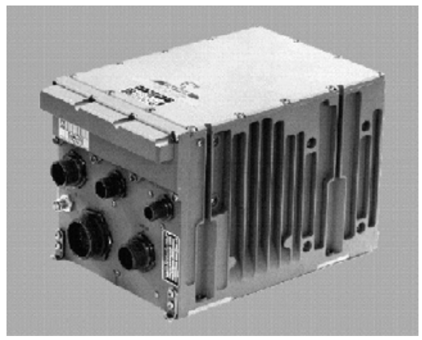


- Inertial Navigation System



Strapdown - INS

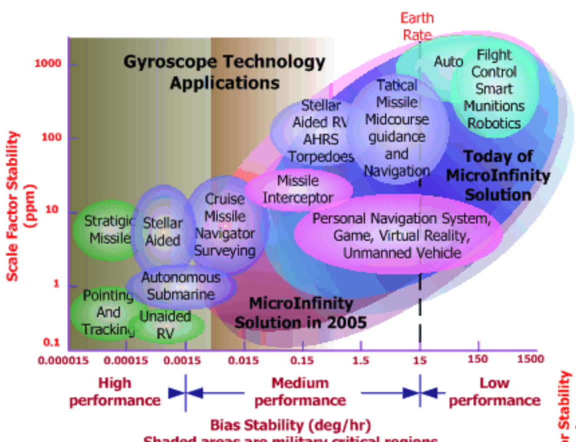
www.inertialengineeringinternational.com/inss...

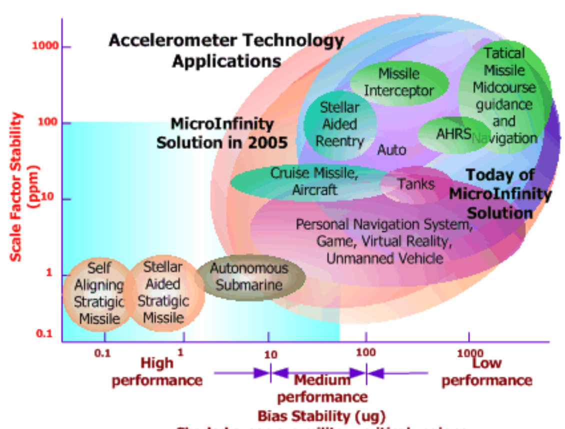


www.fas.org/spp/military/program/nav/egi.htm

SLAM On High Speed Platforms 2006 SLAM Summer School

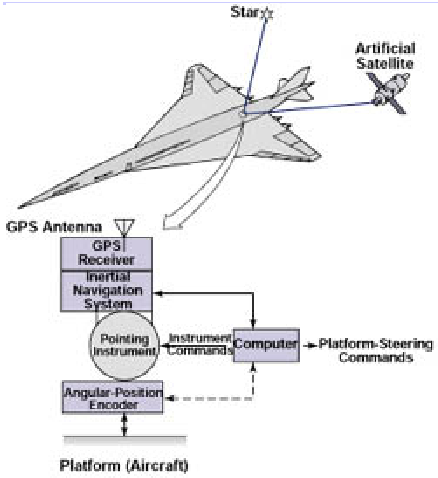
Performances

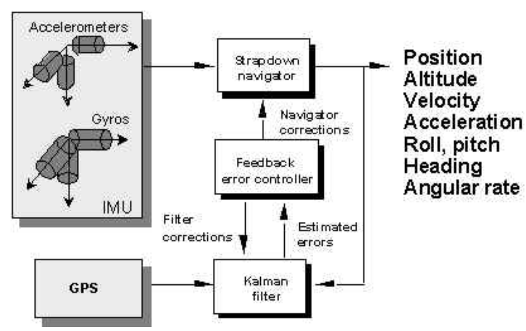




Improving Performances

- www.techbriefs.com/Briefs/Feb01/ARC14060.html

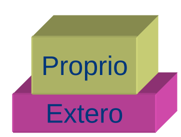


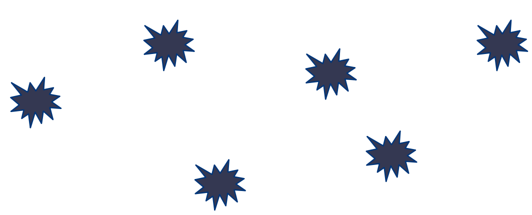


topo.epfl.ch/personnes/jsk/research/Sensors.htm

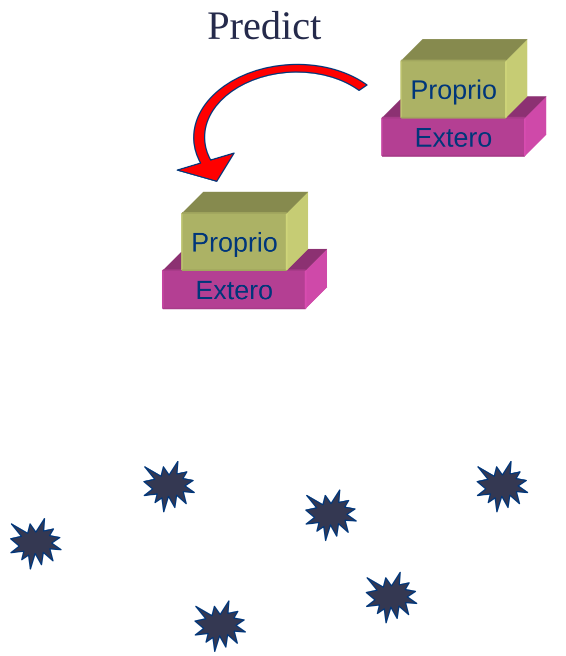
Inertial SLAM

- SLAM forms a major component of the algorithm – but it does not end there.
- Start at unknown location with no a priori map information.
- Predict motion through INS.
- Make relative observations to local features and build a map through these observations.
- Predict and re-observe features which are in the map and begin to correlate
- Correlation assists in constraining drift in inertial solution
- Update the vehicle and feature estimates at each observation



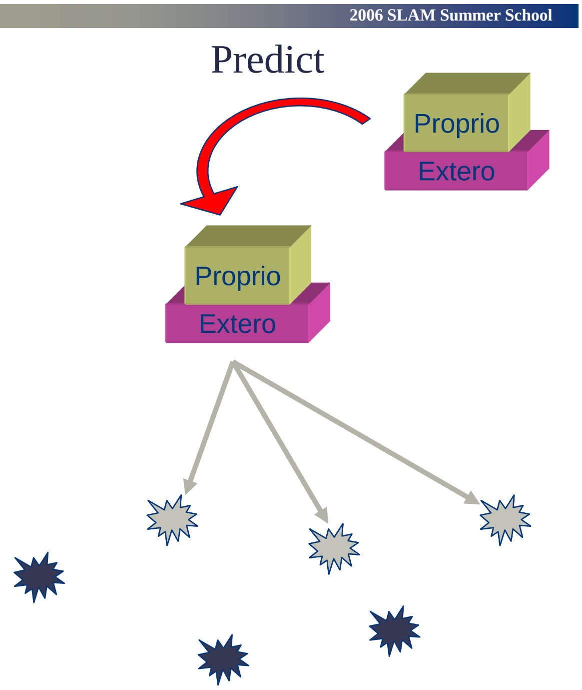


Inertial SLAM



- SLAM forms a major component of the algorithm – but it does not rest there.
- Start at unknown location with no a priori map information.
- Predict motion through INS.
- Make relative observations to local features and build a map through these observations.
- Predict and re-observe features which are in the map and begin to correlate
- Correlation assists in constraining drift in inertial solution
- Update the vehicle and feature estimates at each observation

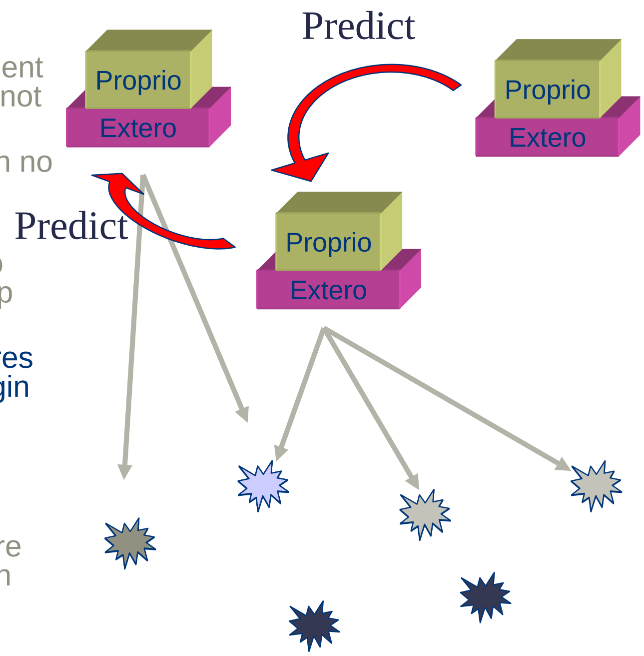
Inertial SLAM



- SLAM forms a major component of the algorithm – but it does not rest there.
- Start at unknown location with no a priori map information.
- Predict motion through INS.
- Make relative observations to local features and build a map through these observations.
- Predict and re-observe features which are in the map and begin to correlate
- Correlation assists in constraining drift in inertial solution
- Update the vehicle and feature estimates at each observation

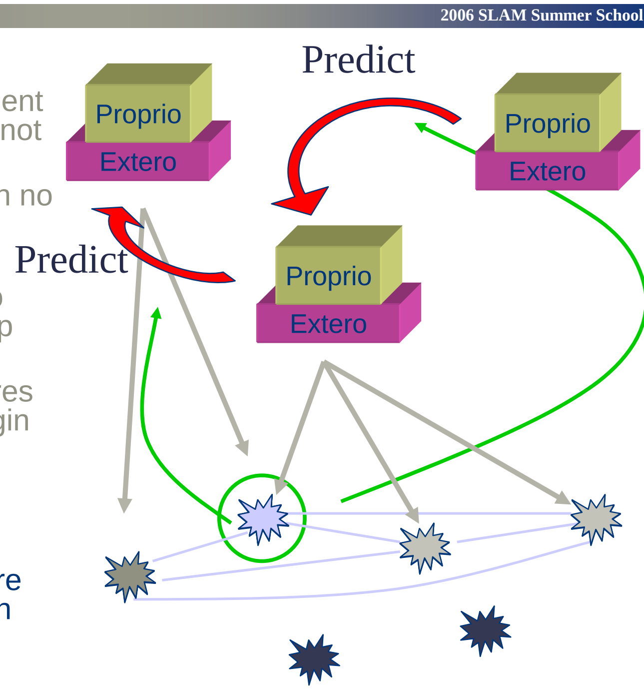
Inertial SLAM

- SLAM forms a major component of the algorithm – but it does not rest there.
- Start at unknown location with no a priori map information.
- Predict motion through INS.
- Make relative observations to local features and build a map through these observations.
- Predict and re-observe features which are in the map and begin to correlate
- Correlation assists in constraining drift in inertial solution
- Update the vehicle and feature estimates at each observation



Inertial SLAM

- SLAM forms a major component of the algorithm – but it does not rest there.
- Start at unknown location with no a priori map information.
- Predict motion through INS.
- Make relative observations to local features and build a map through these observations.
- Predict and re-observe features which are in the map and begin to correlate
- Correlation assists in constraining drift in inertial solution
- Update the vehicle and feature estimates at each observation



The Inertial SLAM Model

- The Problem

$$\mathbf{P}(\mathbf{x}_k, \mathbf{m}|\mathbf{Z}^k, \mathbf{U}^k, \mathbf{x}_0)$$

- The Models

$$\mathbf{P}(\mathbf{Z}^k|\mathbf{X}^k, \mathbf{m}) = \prod_{i=1}^{k} \mathbf{P}(\mathbf{z}_i|\mathbf{x}_i, \mathbf{m}) \qquad \text{and} \qquad \mathbf{P}(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_k)$$

- The Solution

$$\begin{aligned}
&\mathbf{P}(\mathbf{x}_k, \mathbf{m}|\mathbf{Z}^k, \mathbf{U}^k, \mathbf{x}_0) \\
&\qquad = K \cdot \mathbf{P}(\mathbf{z}_k|\mathbf{x}_k, \mathbf{m}) \int \mathbf{P}(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_k) \\
&\qquad\quad \times \mathbf{P}(\mathbf{x}_{k-1}, \mathbf{m}|\mathbf{Z}^{k-1}, \mathbf{U}^{k-1}, \mathbf{x}_0) d(\mathbf{x}_{k-1})
\end{aligned}$$

- State and Map Vectors

$$\mathbf{x}_v(k) = [\ \mathbf{p}^n(k) \quad \mathbf{v}^n(k) \quad \boldsymbol{\psi}^n(k)\ ]^T$$

$$\mathbf{x}_m(k) = [\ \mathbf{x}_{m1}^{nT}(k) \quad \mathbf{x}_{m2}^{nT}(k) \quad \ldots \quad \mathbf{x}_{mN}^{nT}(k)\ ]^T$$

- Augmented State and Covariance Matrix

$$\mathbf{x}(k \mid k) = \begin{bmatrix} \mathbf{x}_v(k \mid k) \\ \mathbf{x}_m(k \mid k) \end{bmatrix}$$

$$\mathbf{P}(k \mid k) = \begin{bmatrix} \mathbf{P}_{vv}(k \mid k) & \mathbf{P}_{vm}(k \mid k) \\ \mathbf{P}_{mv}(k \mid k) & \mathbf{P}_{mm}(k \mid k) \end{bmatrix}$$

Navigation Frame Mechanisation

- Vehicle Model

$$\begin{bmatrix} \mathbf{p}^n(k) \\ \mathbf{v}^n(k) \\ \boldsymbol{\psi}^n(k) \end{bmatrix} = \begin{bmatrix} \mathbf{p}^n(k-1) + \mathbf{v}^n(k-1)\triangle t \\ \mathbf{v}^n(k-1) + \{\mathbf{C}_b^n(k-1)[\mathbf{f}^b(k) + \boldsymbol{\delta}\mathbf{f}^b(k)] + \mathbf{g}^n\}\triangle t \\ \boldsymbol{\psi}^n(k-1) + \mathbf{E}_b^n(k-1)[\boldsymbol{\omega}^b(k) + \boldsymbol{\delta\omega}^b(k)]\triangle t \end{bmatrix} + \begin{bmatrix} \mathbf{w}_{p^n}(k) \\ \mathbf{w}_{v^n}(k) \\ \mathbf{w}_{\psi^n}(k) \end{bmatrix}$$

- Feature Model

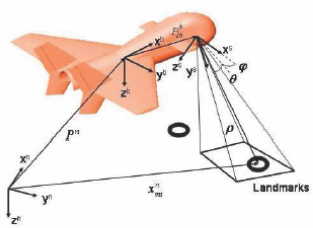
$$\mathbf{x}_{mi}^n(k) = \mathbf{x}_{mi}^n(k-1)$$

- Feature Location

$$\begin{aligned}\mathbf{x}^n_{mi}(k) &= \mathbf{g}_1(\mathbf{p}^n(k), \boldsymbol{\psi}^n(k), \mathbf{z}^s(k)) \\ &= \mathbf{p}^n(k) + \mathbf{C}^n_b(k)\mathbf{p}^b_{bs} + \mathbf{C}^n_b(k)\mathbf{C}^b_s(k)\mathbf{p}^s_{sm}(\mathbf{z}^s(k))\end{aligned}$$

- Observation

$$\mathbf{p}^s_{sm} \quad = \quad \mathbf{g}_2(\mathbf{z}^s(k)) = \begin{bmatrix} \rho\cos(\varphi)\cos(\vartheta) \\ \rho\sin(\varphi)\cos(\vartheta) \\ \rho\sin(\vartheta) \end{bmatrix}$$



SLAM On High Speed Platforms 2006 SLAM Summer School

- Innovation Gate

$$\boldsymbol{\nu}(k)^T\mathbf{S}(k)^{-1}\boldsymbol{\nu}(k) \leq \lambda_n$$

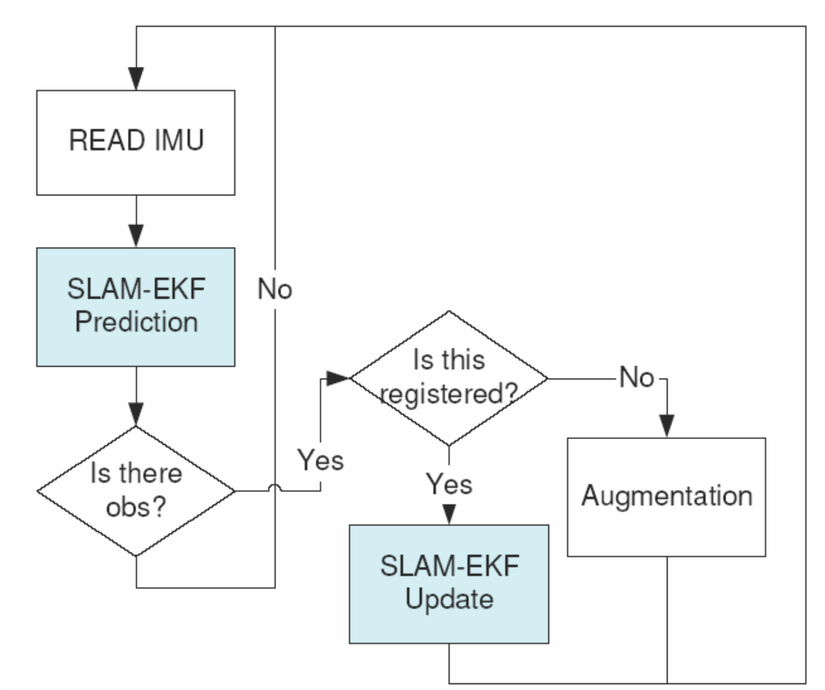
- State Augmentation

$$\begin{aligned}\mathbf{x}_{aug}(k) &= \mathcal{G}(\mathbf{x}_v(k), \mathbf{x}_m(k), \mathbf{z}(k)) \\ &= \begin{bmatrix} \mathbf{x}_v(k) \\ \mathbf{x}_m(k) \\ \mathbf{g}_1(\mathbf{x}_v(k), \mathbf{z}(k)) \end{bmatrix}\end{aligned}$$

- Covariance Augmentation

$$\mathbf{P}_{aug}(k) = \nabla\mathcal{G}\begin{bmatrix} \mathbf{P}_{vv}(k) & \mathbf{P}_{vm}(k) & 0 \\ \mathbf{P}_{mv}(k) & \mathbf{P}_{mm}(k) & 0 \\ 0 & 0 & \mathbf{R}(k) \end{bmatrix}\nabla\mathcal{G}^T$$

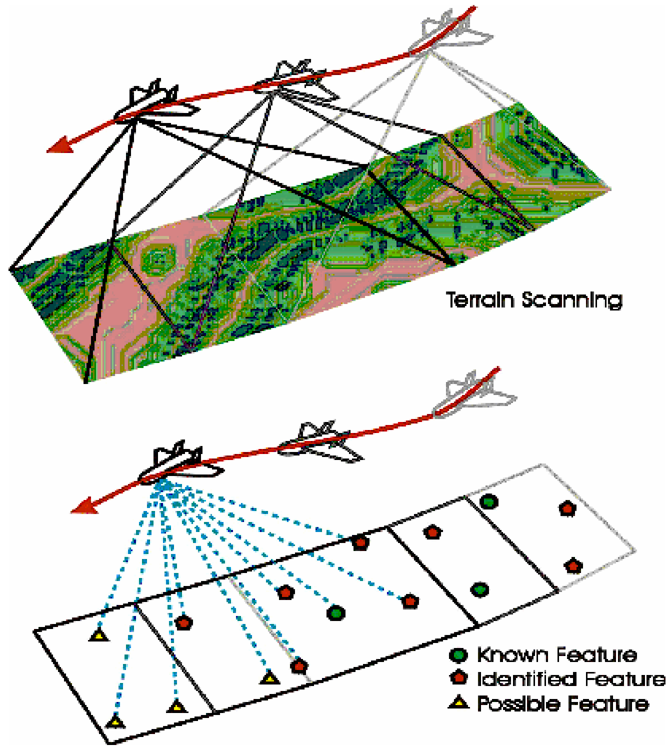
Inertial SLAM Structure



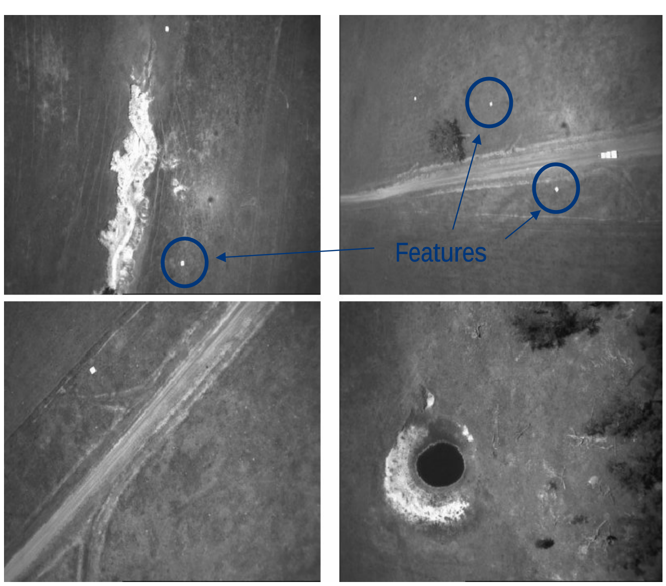
SLAM On High Speed Platforms
2006 SLAM Summer School

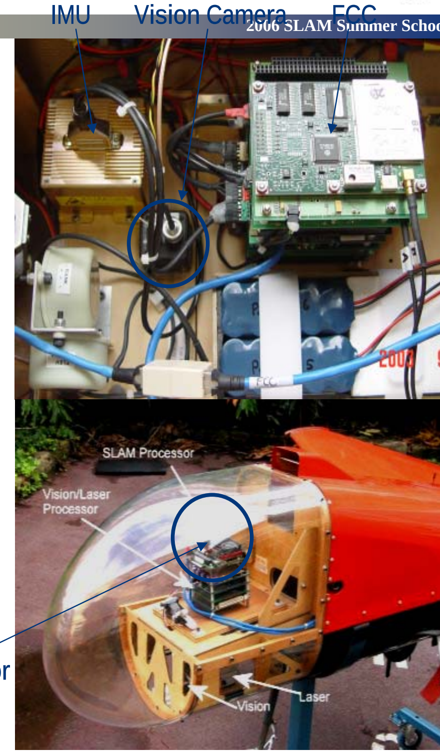
Airborne Inertial SLAM

Airborne Inertial SLAM

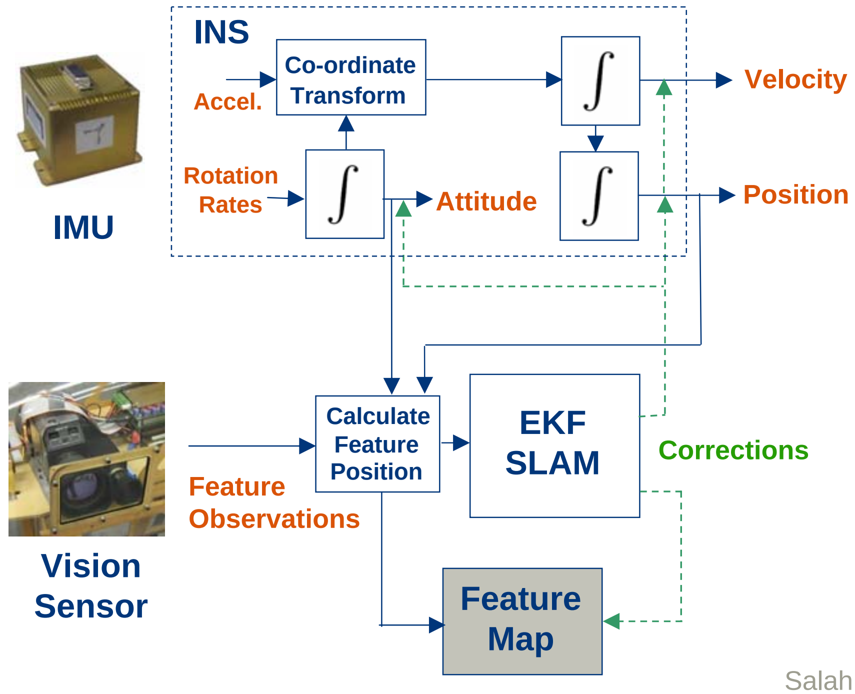


Physical System

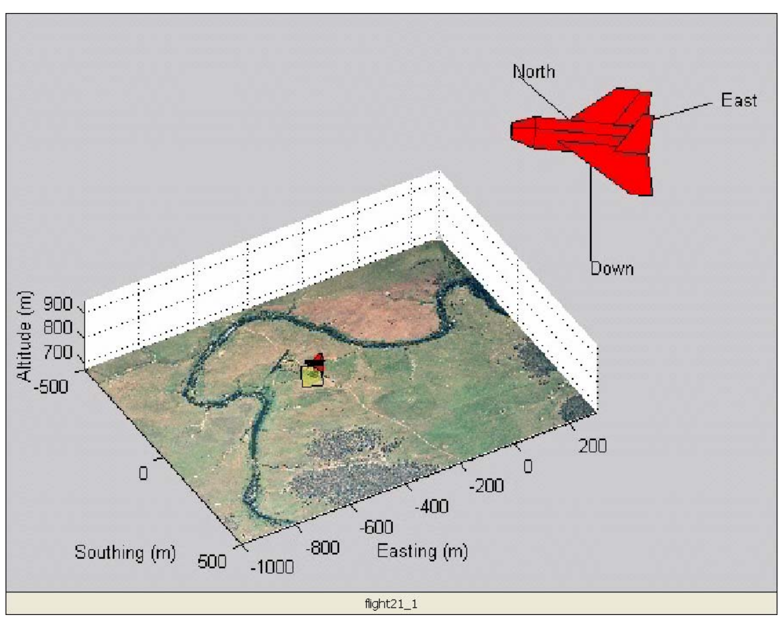




Inertial-Vision SLAM

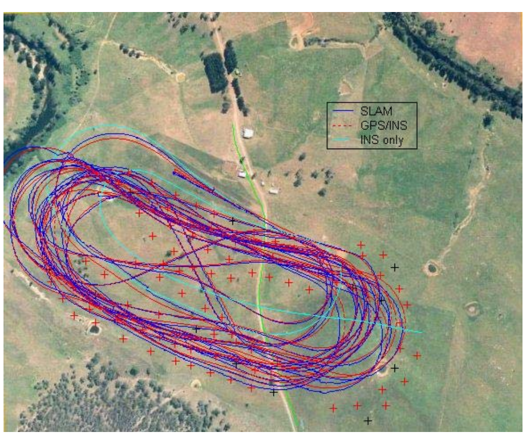


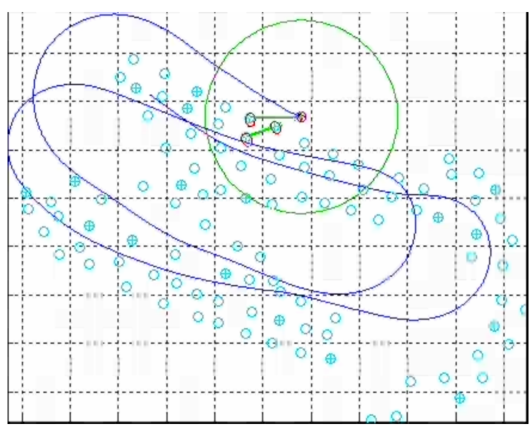
HWIL Testing



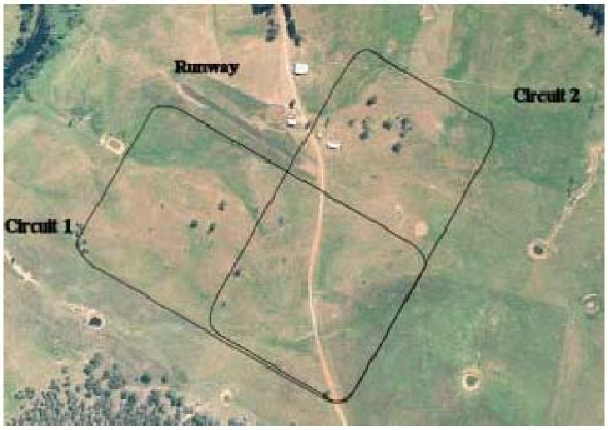
The Flight

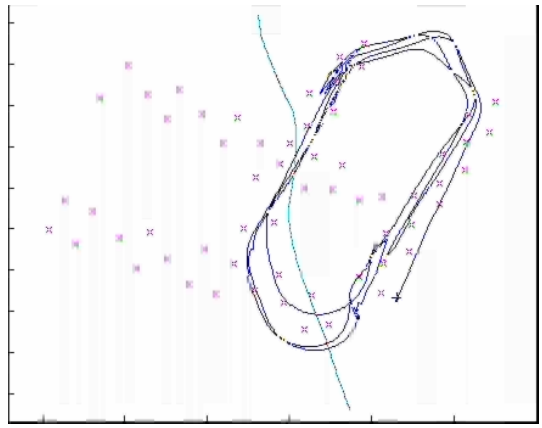
Real Time Airborne SLAM





Real Time Airborne SLAM



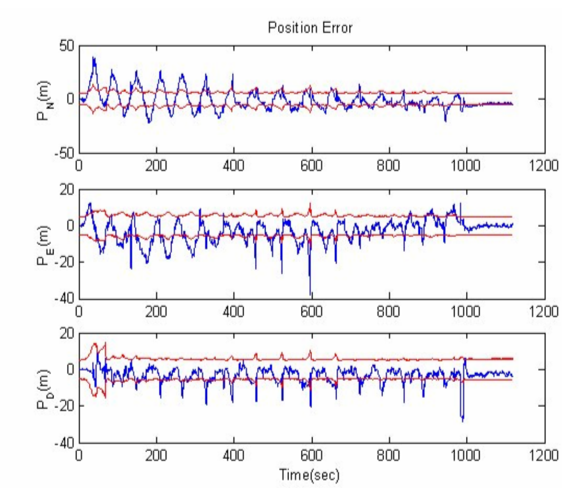


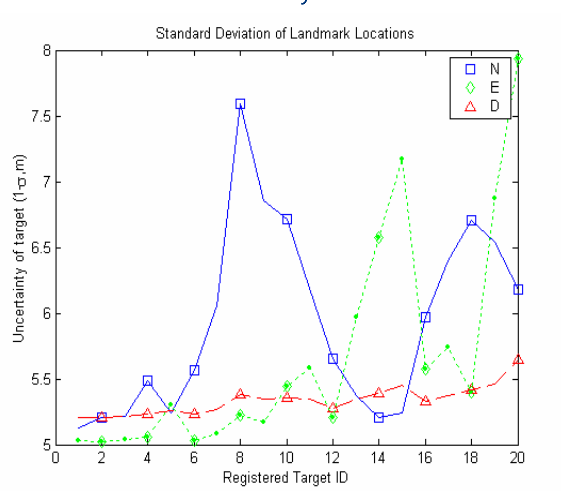
Real Time Airborne SLAM

SLAM On High Speed Platforms 2006 SLAM Summer School

Final 2D Map with 20 registered landmarks

Final Map shows approximately 7.5m uncertainty

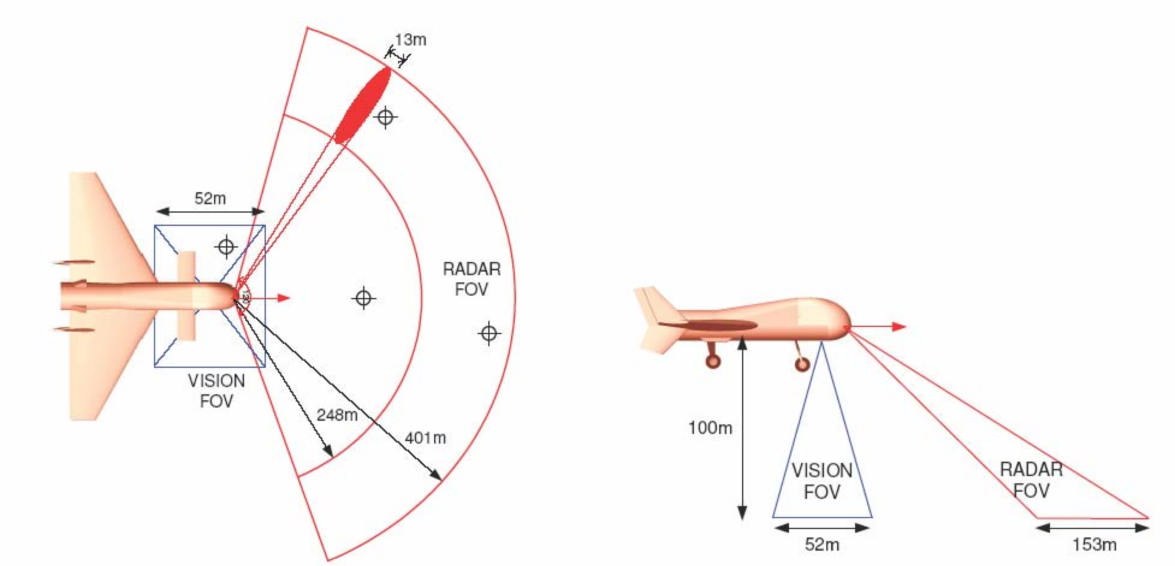




Error Modeling – Understanding the Performance of Inertial SLAM

$$\begin{aligned}\boldsymbol{\delta}\mathbf{p}^n_m &= \boldsymbol{\delta}\mathbf{p}^n + \mathbf{C}^n_b\boldsymbol{\delta}\mathbf{p}^b_{bs} \\ &+\mathbf{C}^n_b[\mathbf{p}^b_{bs}\times]\boldsymbol{\delta\psi}^n + \mathbf{C}^n_b\mathbf{C}^b_s\boldsymbol{\delta}\mathbf{p}^s_{sm} + \mathbf{C}^n_b\mathbf{C}^b_s[\hat{\mathbf{p}}^s_{sm}\times](\boldsymbol{\delta\psi}^n + \boldsymbol{\delta\theta}^b_s)\end{aligned}$$

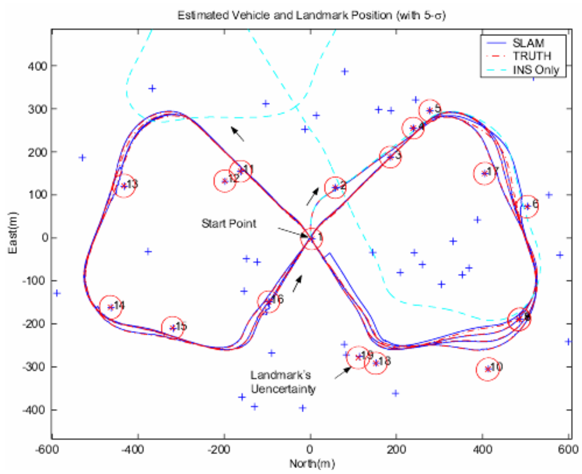
- $\boldsymbol{\delta}\mathbf{p}^n$ is the position error of the INS in the navigation frame.
- $\hat{\mathbf{C}}^n_b\boldsymbol{\delta}\mathbf{p}^b_{sb}$ is the position error in the navigation frame due to uncertainty in the lever-arm vector between sensor instalments position and the vehicle centre. This error can be reduced by precisely measuring the lever-arm vector.
- $\mathbf{C}^n_b[\mathbf{p}^b_{bs}\times]\boldsymbol{\delta\psi}^n$ is the position error in the navigation frame due to the INS attitude error, scaled by the sensor lever-arm vector.
- $\mathbf{C}^n_b\mathbf{C}^b_s\boldsymbol{\delta}\mathbf{p}^s_{ms}$ is the error in the relative position between the vehicle and the landmark in the navigation frame, which is computed from the range/bearing measurement.
- $\mathbf{C}^n_b\mathbf{C}^b_s[\hat{\mathbf{p}}^s_{ms}\times](\boldsymbol{\delta\psi}^n + \boldsymbol{\delta\theta}^b_s)$ is the landmark positioning error due to the INS attitude error and the sensor misalignment, scaled by the measured relative distance.

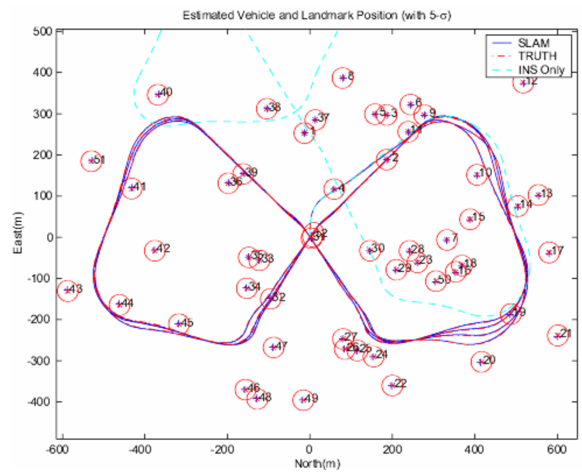


(a) (b)

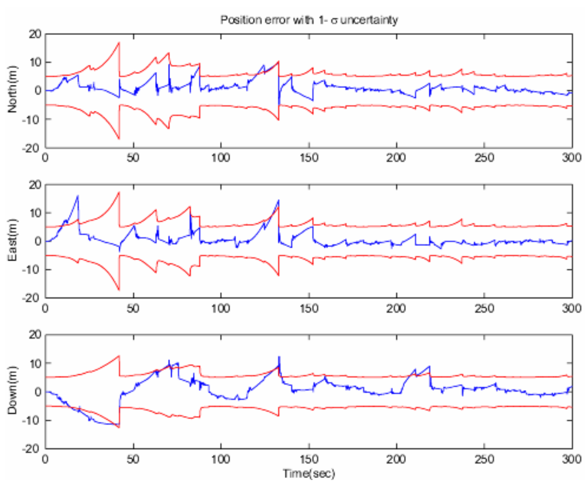
Sensor	Type	Unit	Specification
IMU	Sampling rate	(Hz)	400
	Accel noise	$(m/s^2/\sqrt{Hz})$	0.1
	Gyro noise	$(^\circ/s/\sqrt{Hz})$	0.1
Vision	Frame rate	(Hz)	25
	FOV angle	$(^\circ)$	± 15
	Bearing noise strength	$(^\circ)$	0.1604
	Elevation noise strength	$(^\circ)$	0.1206
Radar	Scanning rate	(Hz)	2
	Front range coverage	(m)	50-300
	Azimuth FOV	$(^\circ)$	120
	Range noise strength	(m)	0.2
	Bearing noise strength	$(^\circ)$	0.2
	Elevation noise strength	$(^\circ)$	2.0
Alignment Error	Horizontal axis	$(^\circ)$	0.5
	Vertical axis	$(^\circ)$	0.0

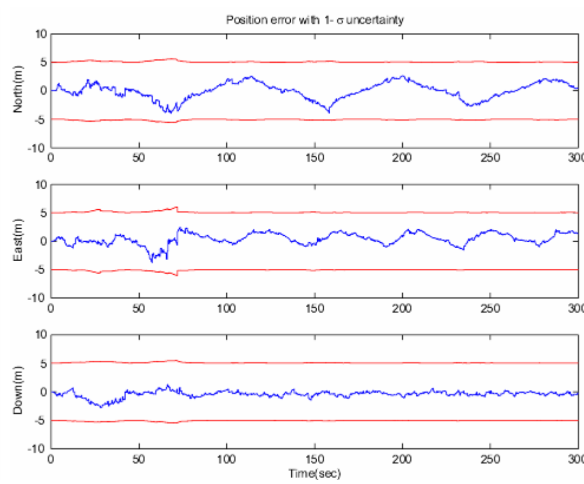
Error Analysis



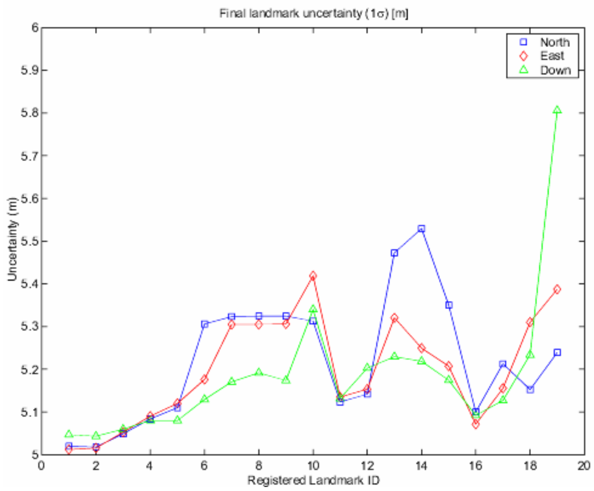


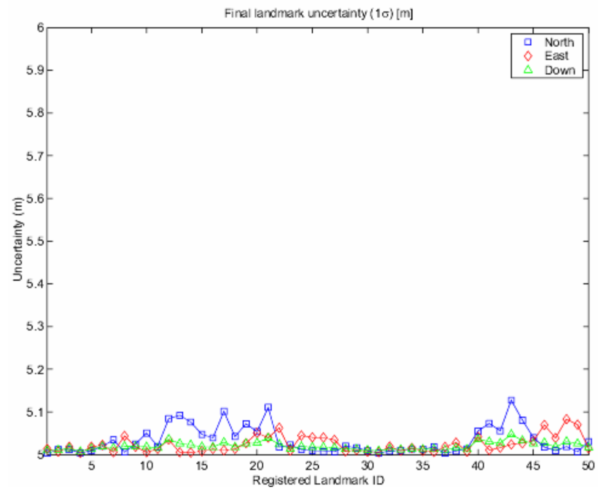
Error Analysis





Error Analysis





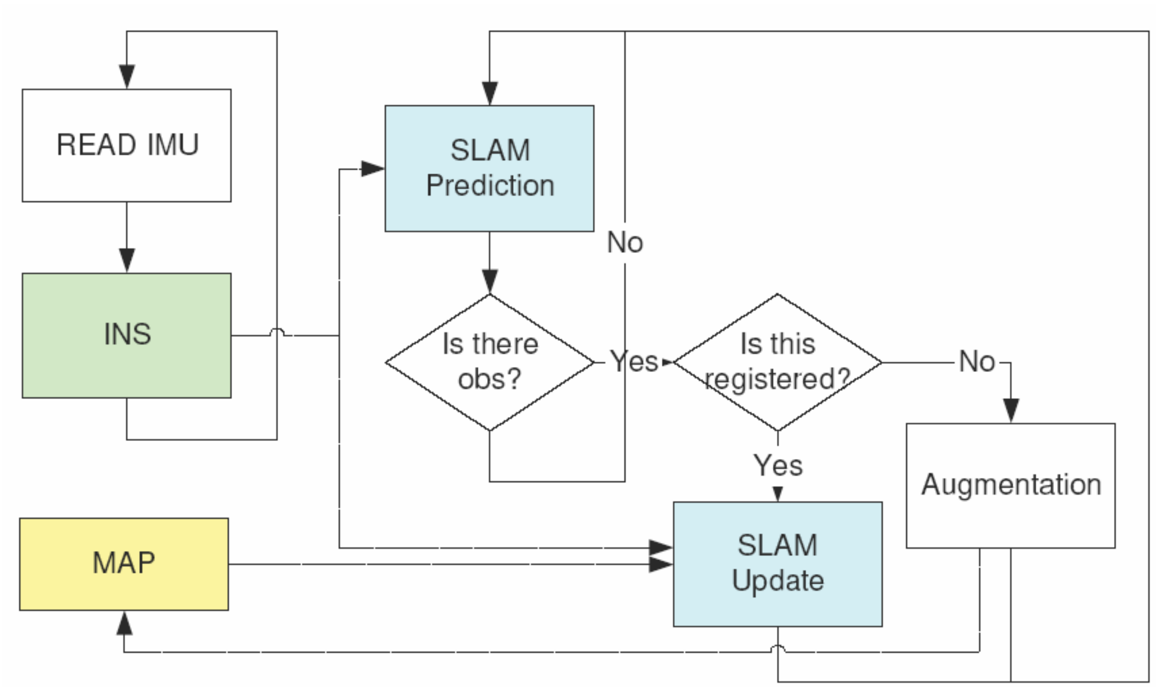
Error Modeling – Recasting Inertial SLAM

$$
\begin{bmatrix} \boldsymbol{\delta}\mathbf{p}^n(k) \\ \boldsymbol{\delta}\mathbf{v}^n(k) \\ \boldsymbol{\delta}\boldsymbol{\psi}^n(k) \\ \boldsymbol{\delta}\mathbf{x}^n_m(k) \end{bmatrix} =
$$

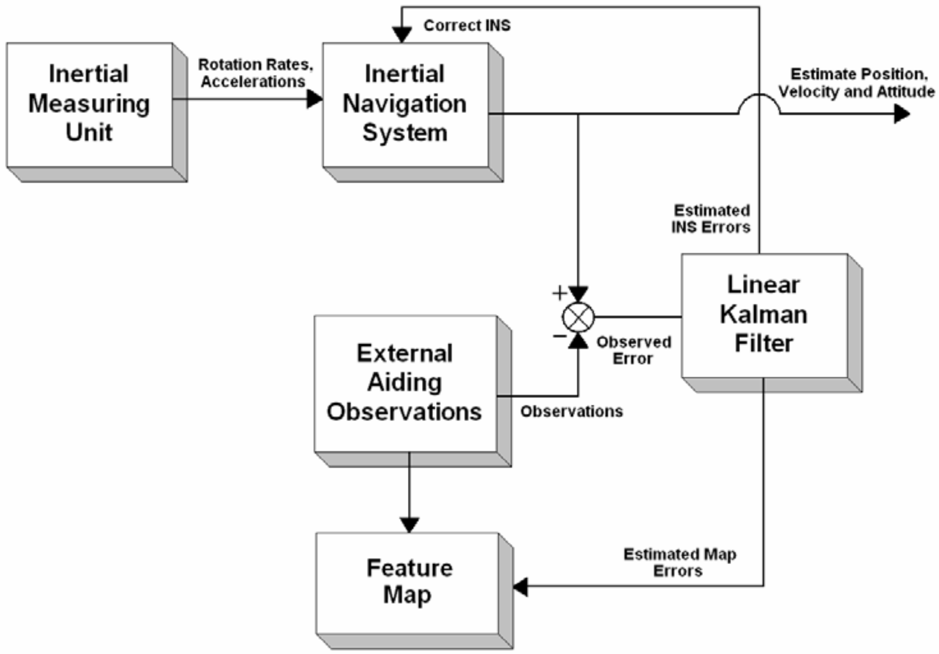
$$
\begin{bmatrix} \mathbf{I} & \triangle t\mathbf{I} - \frac{\triangle t^2}{2}[\boldsymbol{\omega}^n_{in}\times] & \frac{\triangle t^2}{2}[\mathbf{f}^n\times] & \mathbf{0} \\ \mathbf{0} & \mathbf{I} - \triangle t[\boldsymbol{\omega}^n_{in}\times] + \frac{\triangle t^2}{2}[\boldsymbol{\omega}^n_{in}\times]^2 & \triangle t[\mathbf{f}^n\times] + \frac{\triangle t^2}{2}[\boldsymbol{\omega}^n_{in}\times][\mathbf{f}^n\times] & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{m\times m} \end{bmatrix}
$$

$$
\times \begin{bmatrix} \boldsymbol{\delta}\mathbf{p}^n(k-1) \\ \boldsymbol{\delta}\mathbf{v}^n(k-1) \\ \boldsymbol{\delta}\boldsymbol{\psi}^n(k-1) \\ \boldsymbol{\delta}\mathbf{x}^n_m(k-1) \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{C}^n_b(k) & \mathbf{0} \\ \mathbf{0} & -\mathbf{C}^n_b(k) \\ \mathbf{0}_{m\times 3} & \mathbf{0}_{m\times 3} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \sqrt{\triangle t}\boldsymbol{\delta}\mathbf{f}^b(k) \\ \sqrt{\triangle t}\boldsymbol{\delta\omega}^b_{ib}(k) \\ \mathbf{0}_{m\times 3} \end{bmatrix}
$$

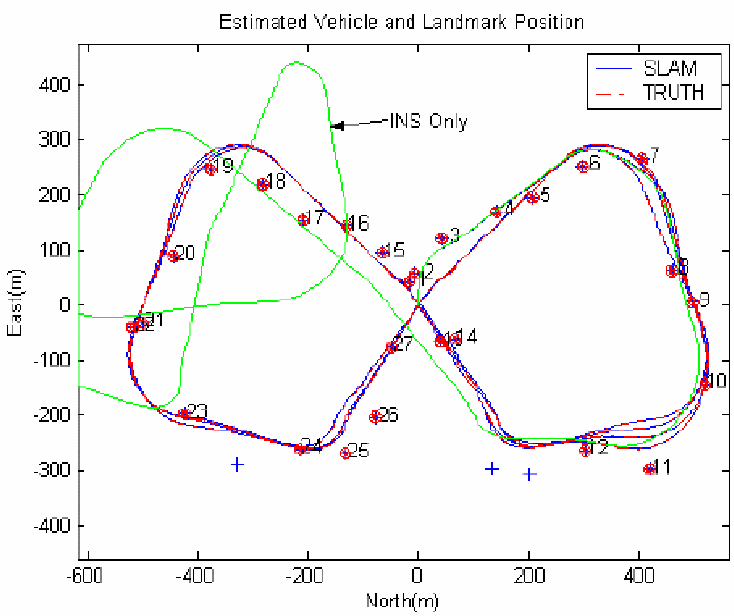
Schematic



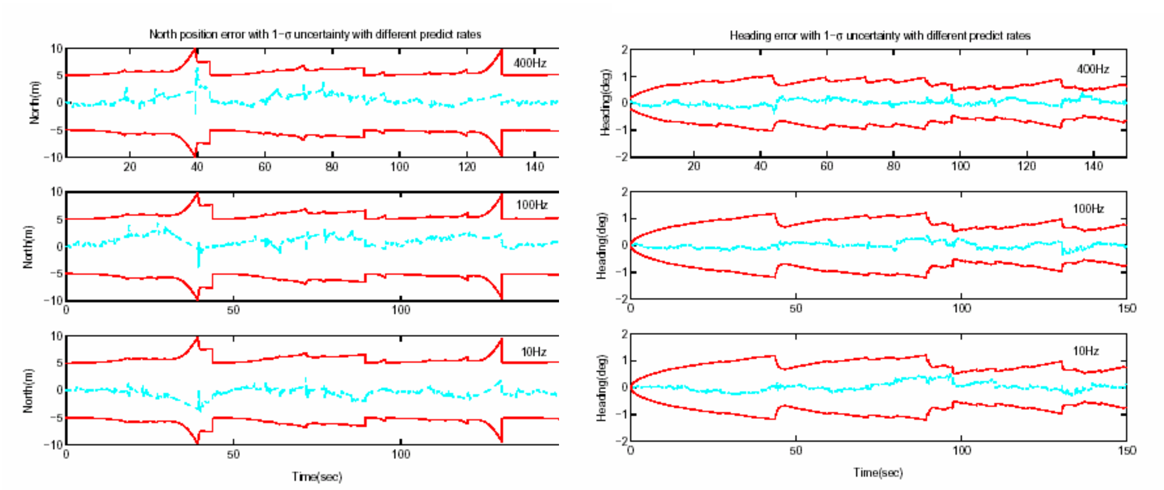
Another Look



Results



Indirect SLAM



Bearing Only Inertial SLAM

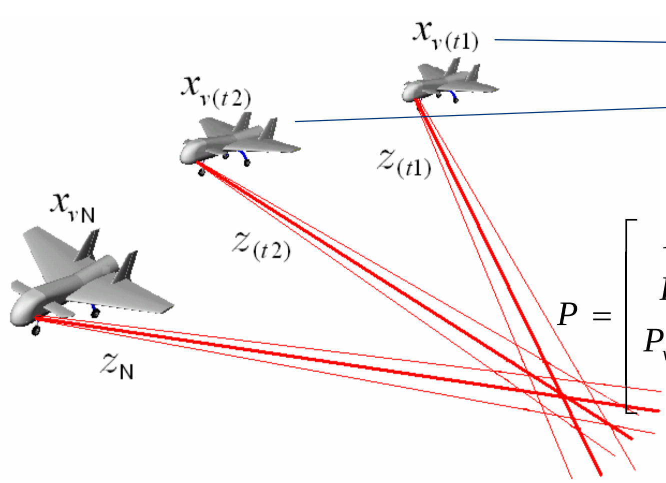
Bearing-Only SLAM

$$\begin{bmatrix} \varphi_i \\ \vartheta_i \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{y^s}{x^s}\right) \\ \tan^{-1}\left(\frac{z^s}{\sqrt{(x^s)^2+(y^s)^2}}\right) \end{bmatrix}$$

Tackling the Bearing-Only Feature Initialisation Problem

- At each observation, augment state vector with vehicle pose (position and attitude, 6 states)
- Store camera observation data

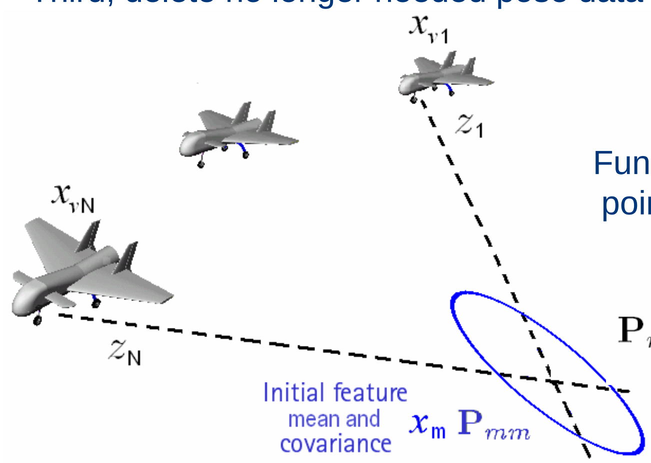
$$x = \begin{bmatrix} x_v \\ x_m \\ x_{v(t1)} \\ x_{v(t2)} \\ \vdots \end{bmatrix}$$



$$P = \begin{bmatrix} P_{vv} & P_{vm} & P_{vv(t1)} & \cdots \\ P_{mv} & P_{mm} & P_{mv(t1)} & \cdots \\ P_{v(t1)v} & P_{v(t1)m} & P_{v(t1)v(t1)} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Tackling the Bearing-Only Feature Initialisation Problem

- When a sufficient base-line is available:
- First, initialise feature position using constrained initialisation
- Second, fuse in remaining observations in batch Kalman filter update step
- Third, delete no longer needed pose data from state vector



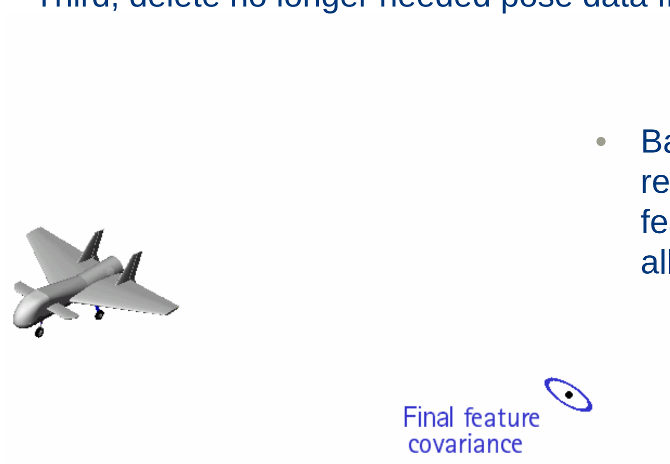
$$x_m = \mathbf{G}(x_{vN}, x_{v1}, z_N, z_1)$$

Function G() – Finds closest point of 2 lines in 3D space

$$\mathbf{P}_{mm} = \nabla\mathbf{G}\begin{bmatrix}\mathbf{P}_{vv} & 0\\ 0 & \mathbf{R}\end{bmatrix}\nabla\mathbf{G}^T$$

Tackling the Bearing-Only Feature Initialisation Problem

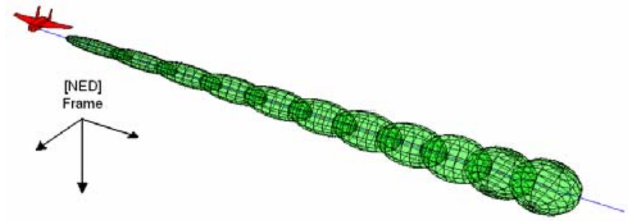
- When a sufficient base-line is available:
- First, initialise feature position using constrained initialisation
- Second, fuse in remaining observations in batch Kalman filter update step
- Third, delete no longer needed pose data from state vector



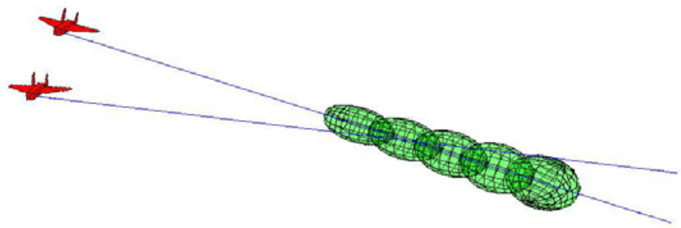
- Batch Kalman filter update recovers information to the feature and vehicle states from all the observations

Data Association Process

- Feature seen for first time: Initialise multiple Gaussian 3D position hypotheses along line of sight based on range cut-off



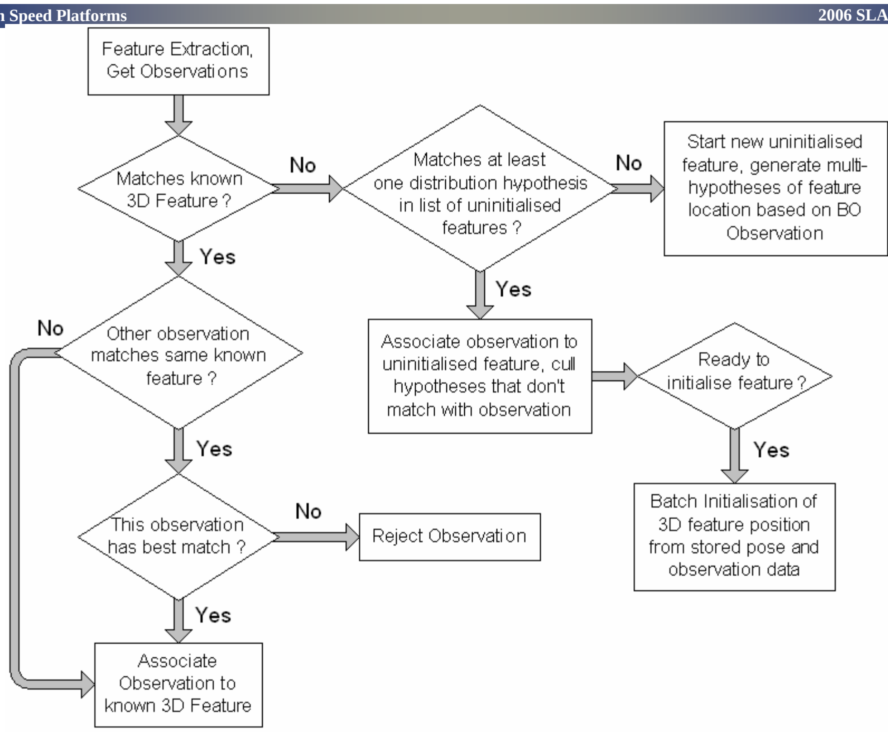
- In subsequent observations we can cull hypotheses that don't match



- When only a single hypotheses is left, can use this as an indicator that the feature is ready to be initialised

Data Association Process

SLAM On High Speed Platforms
2006 SLAM Summer School



Change in Sensor Orientation

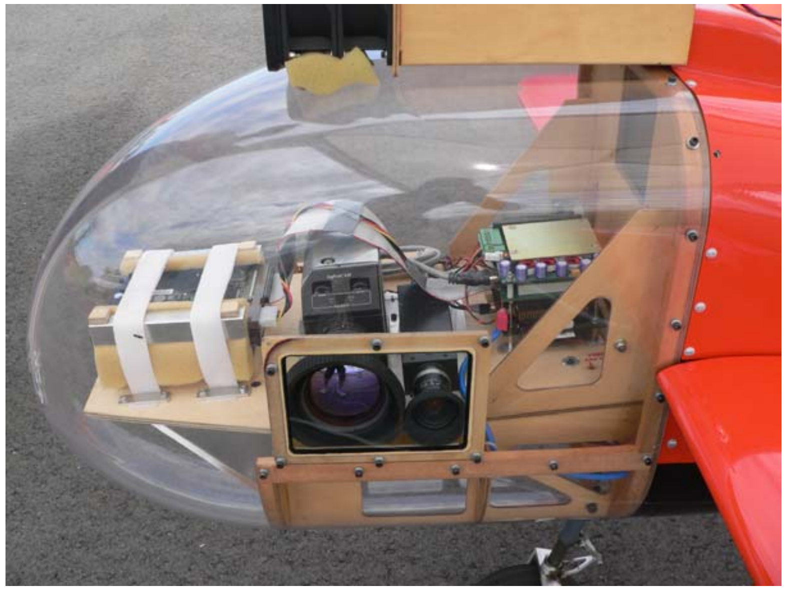
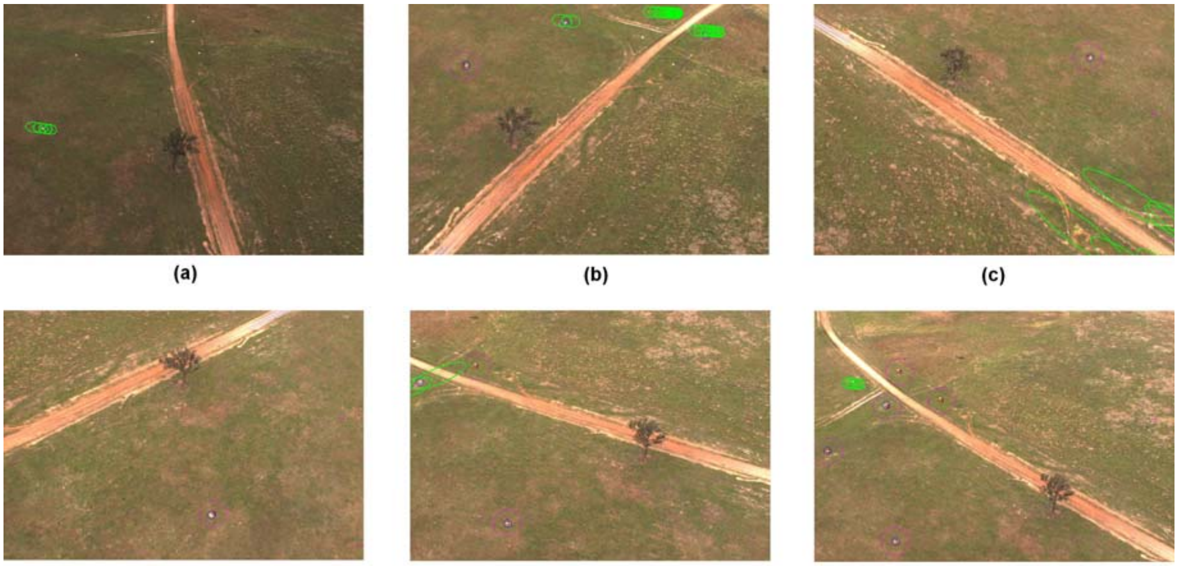


Image Results

Results – Single Orbit



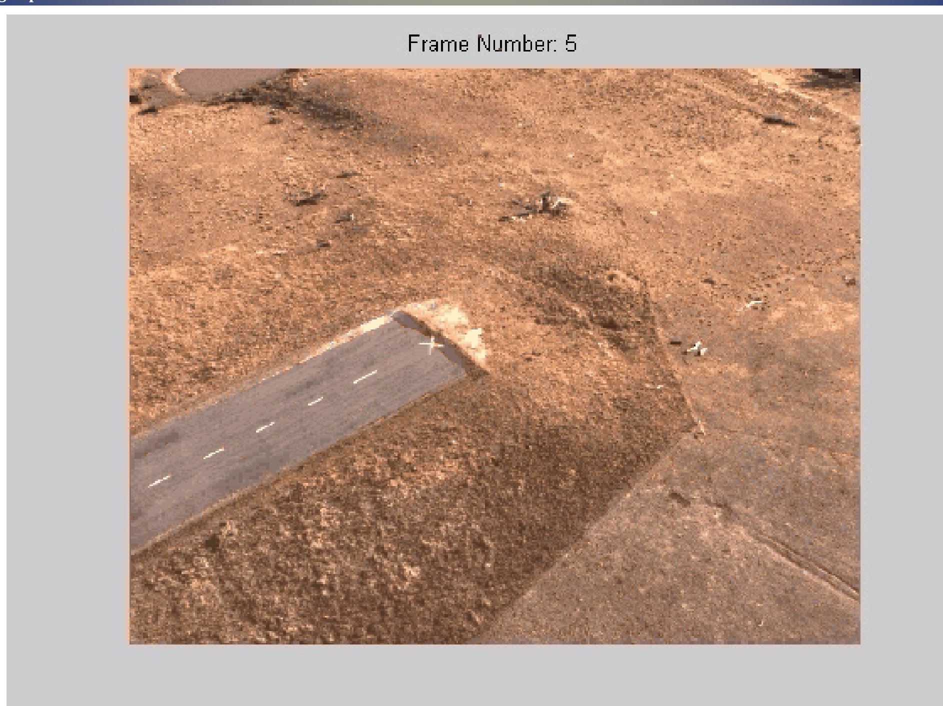
(a) (b) (c)

(d) (e) (f)

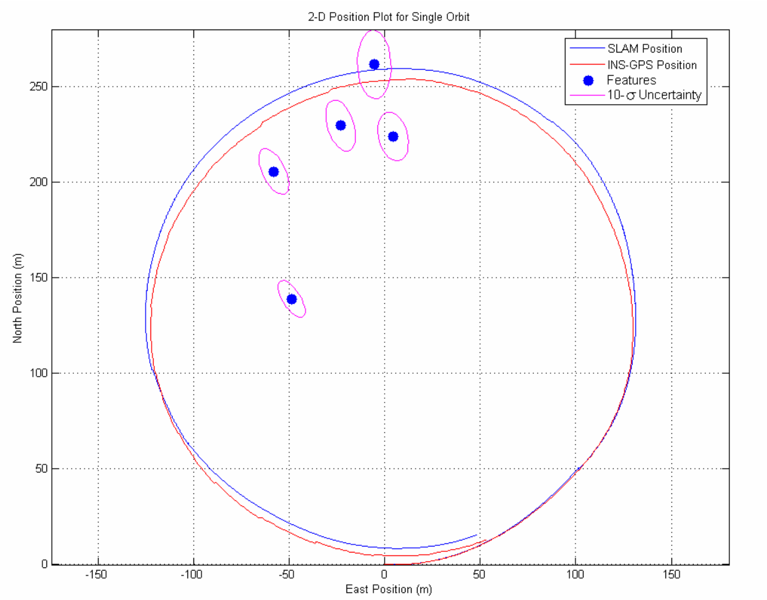
Observations Initialised Feature Multi-hypothesis Uninitialised Feature

Movie

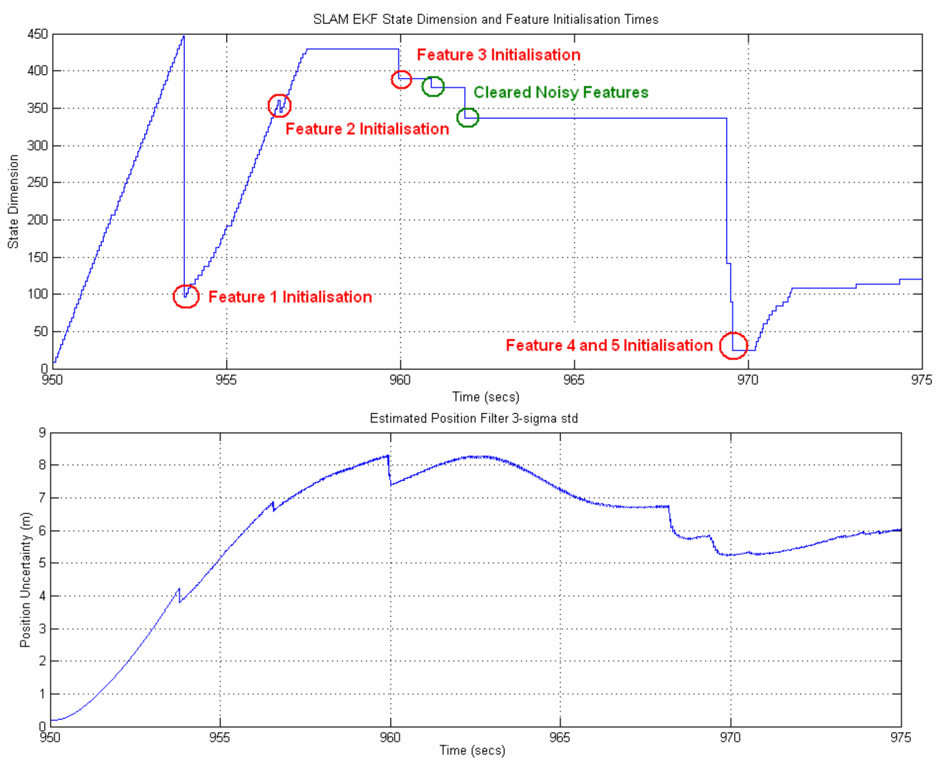
SLAM On High Speed Platforms 2006 SLAM Summer School



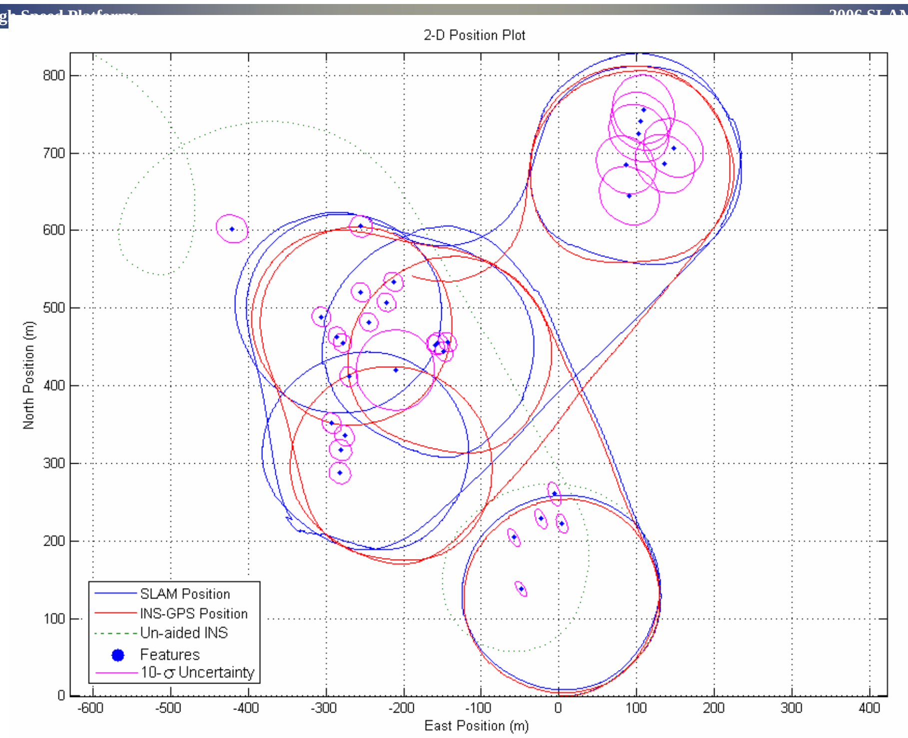
Results – Single Orbit



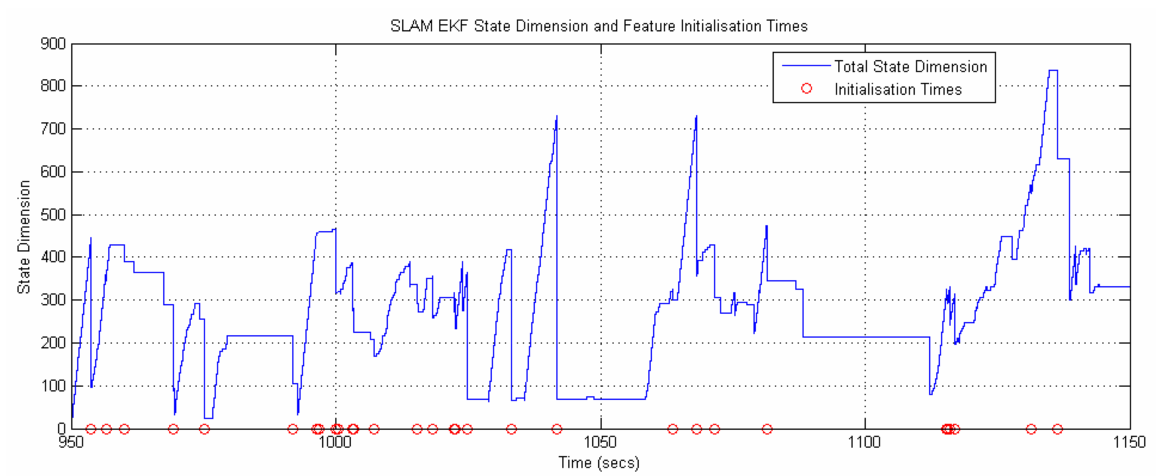
Results – Single Orbit



Results – Larger Area

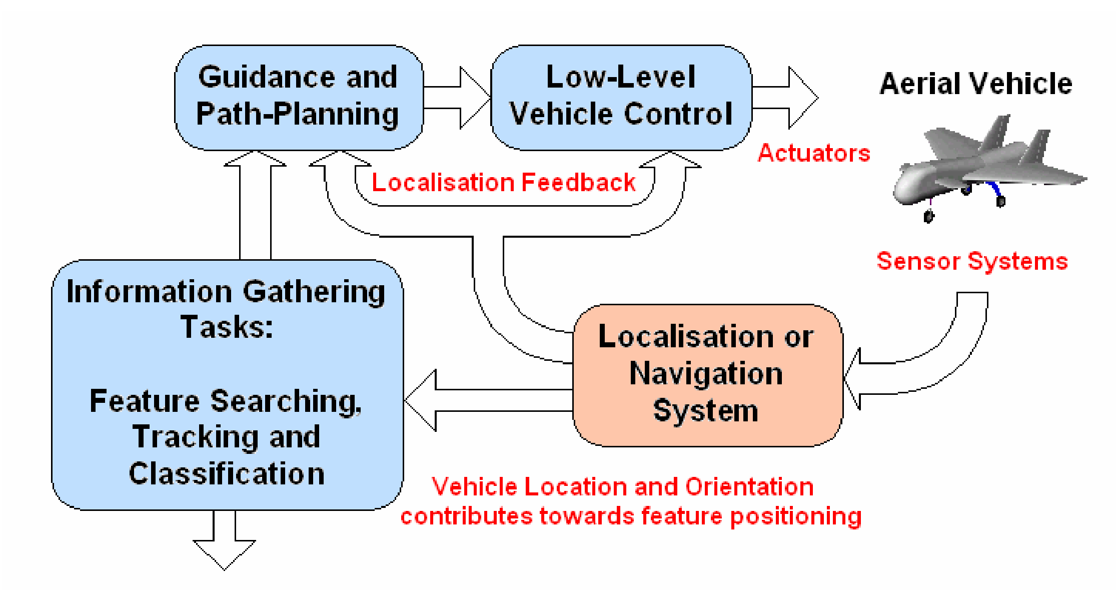


Results – Larger Area

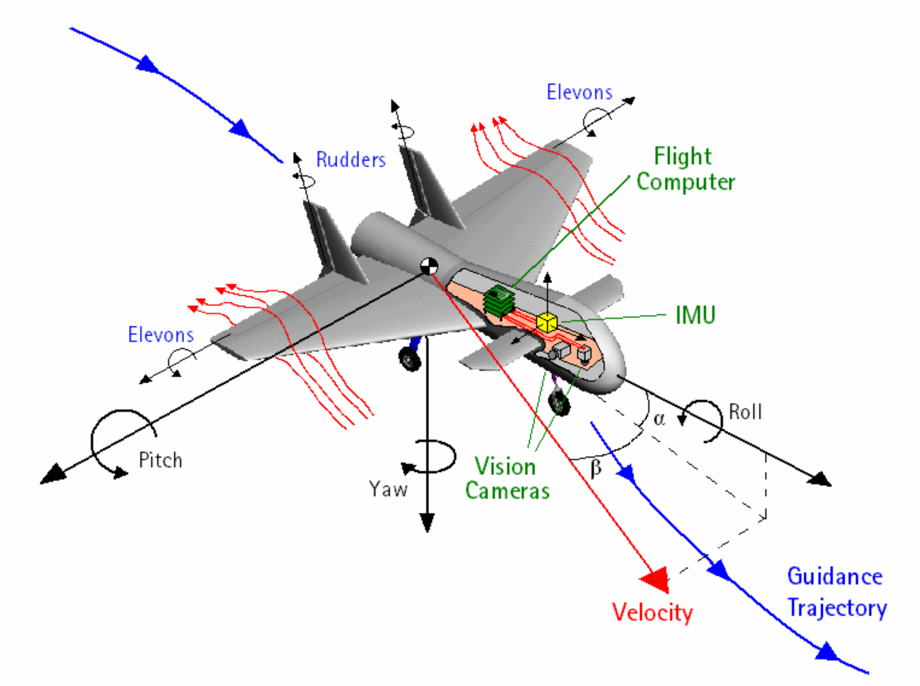


Active SLAM

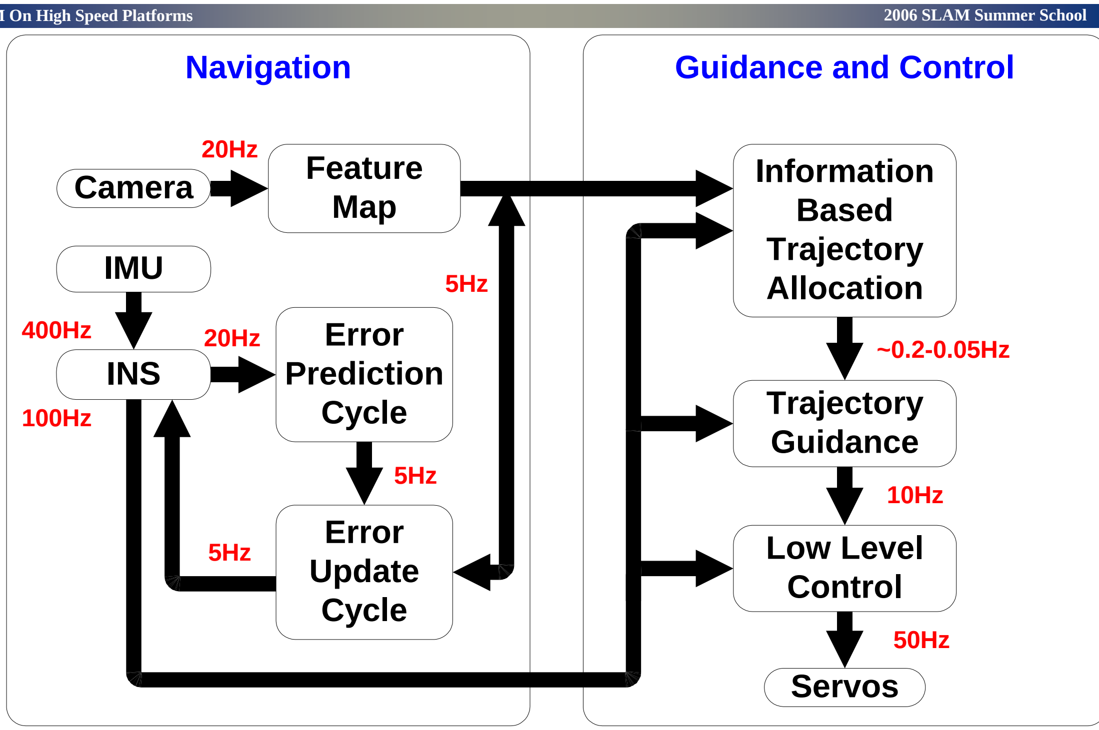
Like any other Autonomous Vehicle…



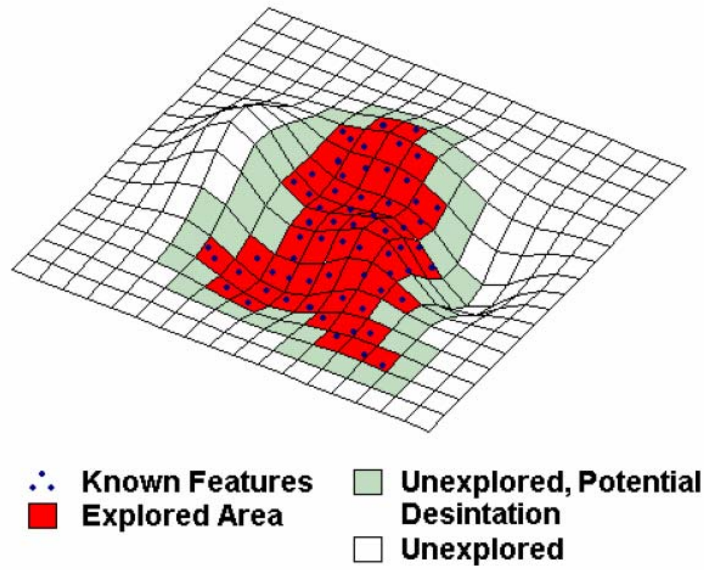
Complex System



A closer look at GNC

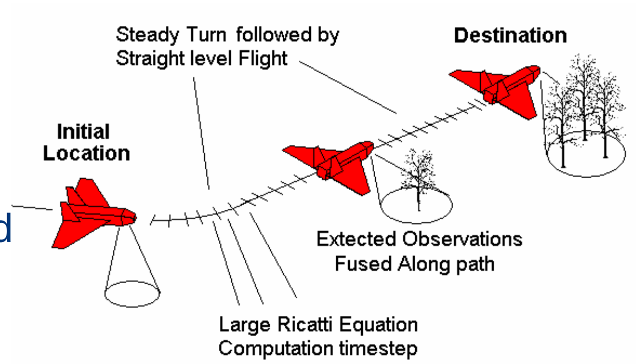


Information Gain



Predicting Forward Information Content

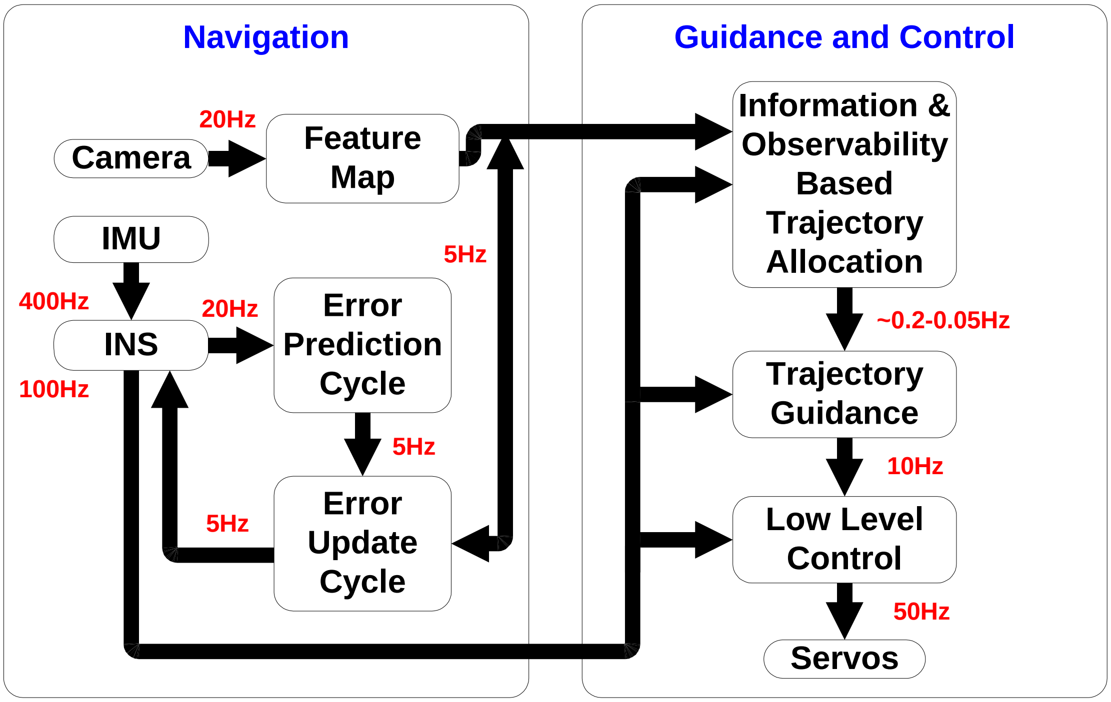
- Flight motion predicted forward as a steady bank to turn towards the destination followed by straight and level flight



- Covariance Propagated Forward via the Ricatti Equation with approximate vehicle flight motion
- Use large computation step (~1-2 seconds)

$$
\begin{aligned}
\mathbf{P}(k+1) &= \mathbf{F}(k+1)[\mathbf{P}(k) - \mathbf{P}(k)\mathbf{H}^T(k)(\mathbf{H}(k)\mathbf{P}\mathbf{H}^T(k)+\mathbf{R}(k))^{-1}\mathbf{H}(k)\mathbf{P}]\mathbf{F}^T(k+1)\ldots \\
&\quad +\mathbf{G}(k+1)\mathbf{Q}(k+1)\mathbf{G}^T(k+1)
\end{aligned}
$$

Extending GNC - Observability



Observability of Inertial SLAM

- Indirect Equations are represented as a time-variant linear system

$$S \rightarrow \begin{array}{lll} F(t) = F_1 & H(t) = H_1 & t_0 < t < t_1 \\ F(t) = F_2 & H(t) = H_2 & t_1 < t < t_2 \\ & \vdots & \\ F(t) = F_k & H(t) = H_k & t_{k-1} < t < t_k \end{array}$$

- A Stripped Observability Matrix Analysis is used to study the observability of the system, over k time segments

$$Q_1 = \begin{bmatrix} H_k \\ H_k F \\ H_k F_k \\ \vdots \\ H_k F_k^{\,n-1} \end{bmatrix} Q_1 = \begin{bmatrix} -I_{3x3} & 0 & \times \hat{r}^n{}_{mvk} & I_{3x3} \\ 0 & -I_{3x3} & 0 & 0 \\ 0 & 0 & \times f^n{}_k & 0 \end{bmatrix} \qquad Q_{SOM} = \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_k \end{bmatrix}$$

Understanding the Observability

- Observability matrix for two time segments
- If f^n or $r^n{}_{mv}$ change direction then second segment adds an extra linearly independent row

$$Q = \begin{bmatrix} -I_{3x3} & 0 & \times \hat{r}^n{}_{mv1} & I_{3x3} \\ 0 & -I_{3x3} & 0 & 0 \\ 0 & 0 & \times f^n{}_1 & 0 \\ -I_{3x3} & 0 & \times \hat{r}^n{}_{mv2} & I_{3x3} \\ 0 & -I_{3x3} & 0 & 0 \\ 0 & 0 & \times f^n{}_2 & 0 \end{bmatrix}$$

- unobservable modes are:

$$x_{unobs} = \delta p^n + \delta m_1{}^n + \delta m_2{}^n + \ldots$$

- Velocity and attitude error states are completely observable

Understanding the Observability

$$\begin{bmatrix} \delta p^n \\ \delta v^n \\ \delta \Psi^n \\ \delta m_1^n \\ \delta m_2^n \\ \vdots \\ \delta m_N^n \end{bmatrix} \rightarrow \begin{bmatrix} \delta m_1^n - \delta p^n \\ \delta m_2^n - \delta p^n \\ \vdots \\ \delta m_N^n - \delta p^n \\ \delta v^n \\ \delta \Psi^n \end{bmatrix}$$

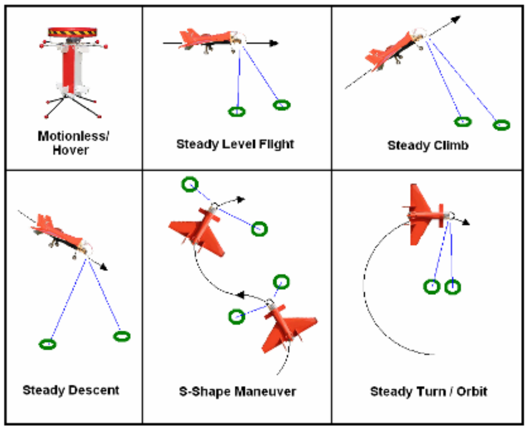
- Can augment state vector to achieve completely observable SLAM (relative representation)
- One Instantaneously Unobservable mode in new state vector, related to attitude along direction of specific force

$$x_{unobs} = (f^n)\delta\Psi^n + (f^n \times \hat{r}^n{}_{m1v})[\delta m^n{}_1 - \delta p^n] + \ldots$$
$$(f^n \times \hat{r}^n{}_{m2v})[\delta m^n{}_2 - \delta p^n] + \ldots$$

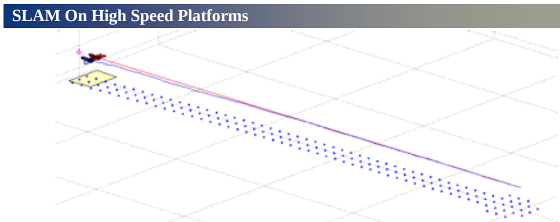
- System becomes completely observable when instantaneously unobservable mode changes direction between time segments (i.e. with the right manoeuvres)

Observability and Manoeuvres

Maneuver	Δf^n	$\Delta \hat{r}^n_{mv}$	Expected Estimate Accuracy
Motionless/Hover	none	none	continuing loss in heading accuracy
Climb or Descent	none	steady change in vector direction	continuing loss in heading accuracy
Steady Level Flight	none	steady change in vector direction	continuing loss in heading accuracy
Steady Turn/Orbit	vector traces a cone shape	range to feature constant for features at center of turn, direction changing	high accuracy on all attitude states
S-Shape Maneuver	vector oscillates back and forward	vector to features traces S-shape	high accuracy on all attitude states

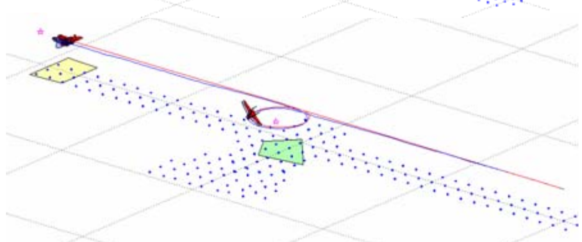


Manoeuvres



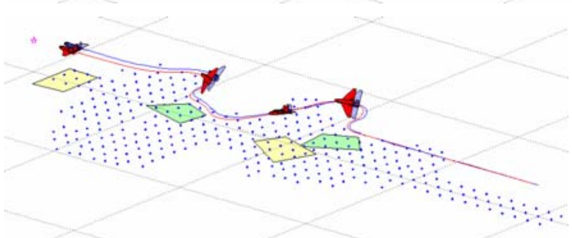
Straight Level Flight

Low Heading Observability



Straight Level Flight, followed by a full Orbit, followed by level flight

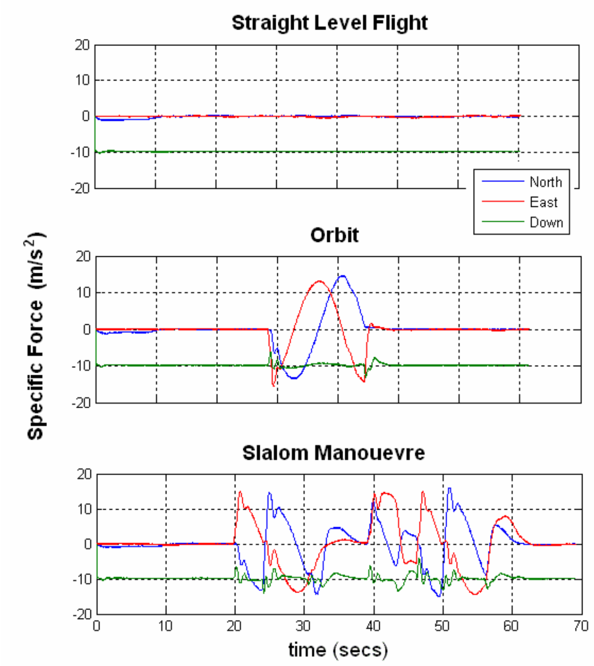
Improved Heading Observability



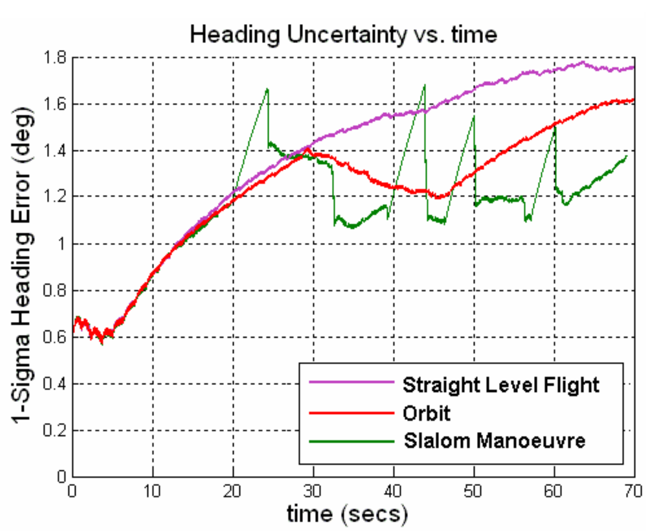
Straight Level Flight, followed by a Slalom Manoeuvre, followed by level flight

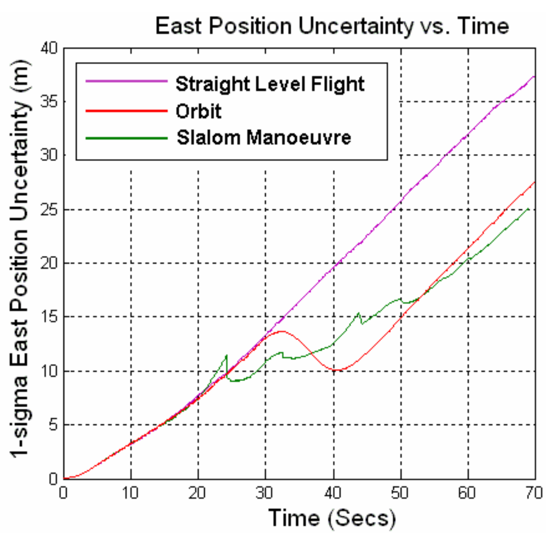
Improved Heading Observability

Manoeuvres



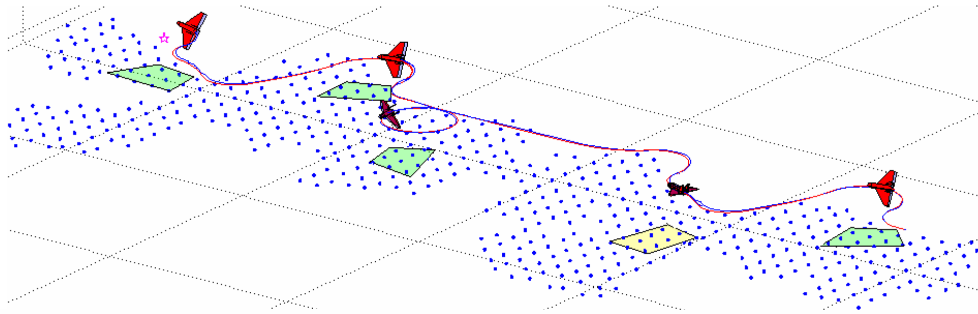
Results

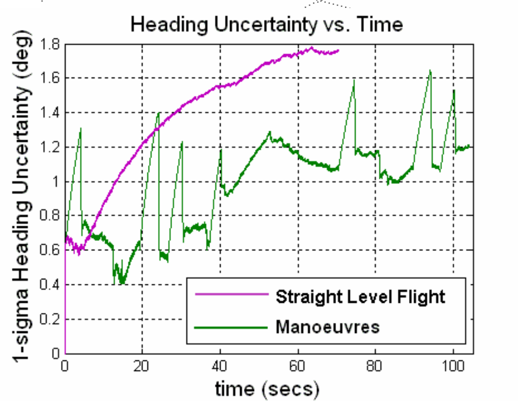


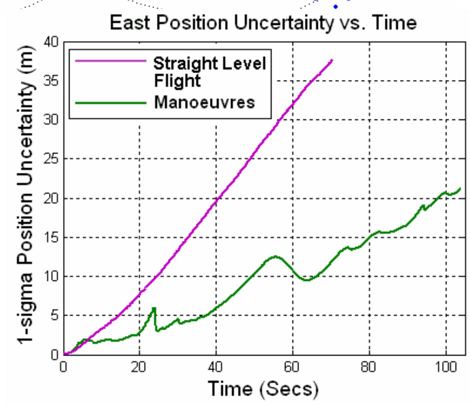


- Heading Uncertainty reduced by manoeuvres, in turn reduces position uncertainty

Putting the Manoeuvres Together







HWIL Results

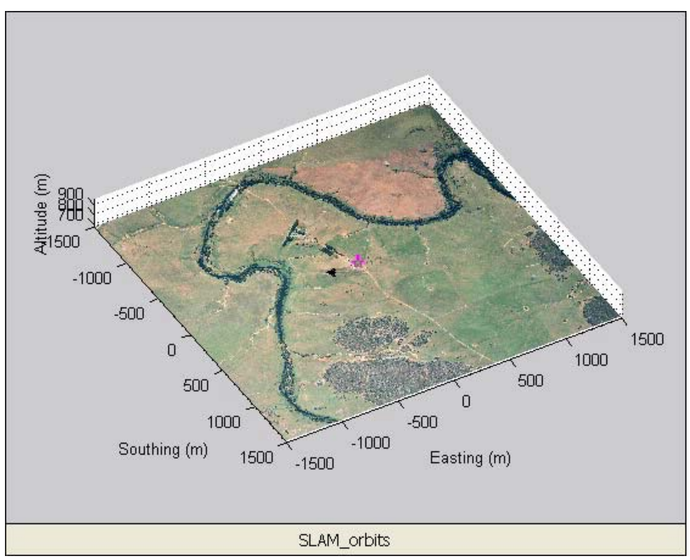
SLAM

INS/GPS

Feature to observe

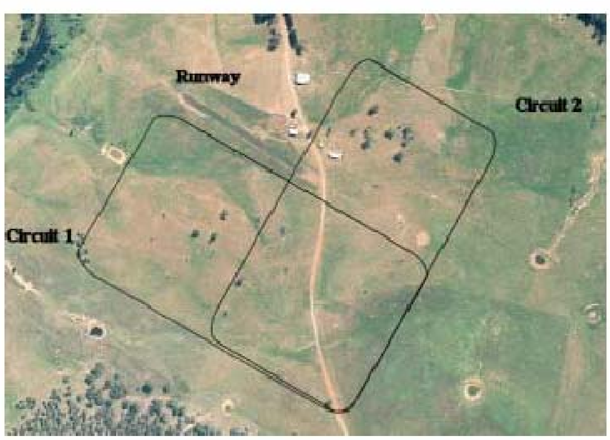
Waypoint

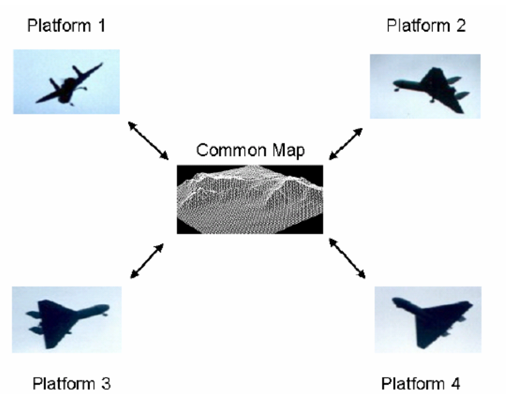
White squares are the landmarks



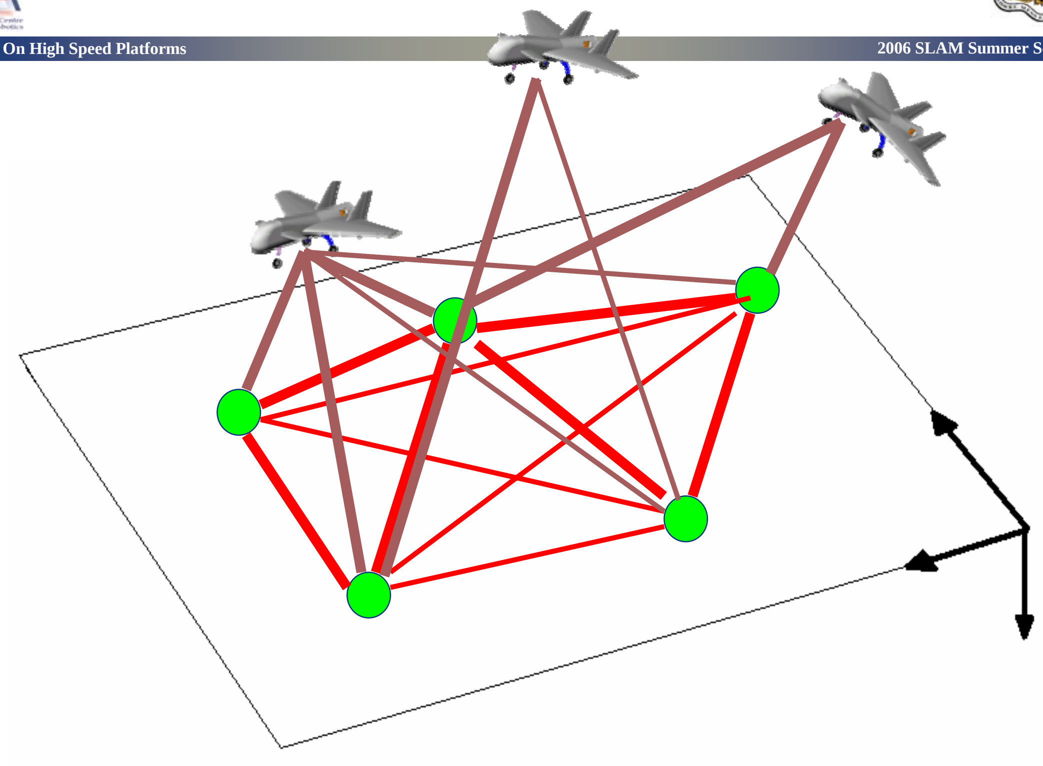
Multi-Vehicle SLAM

Multi-Vehicle SLAM





DDF SLAM

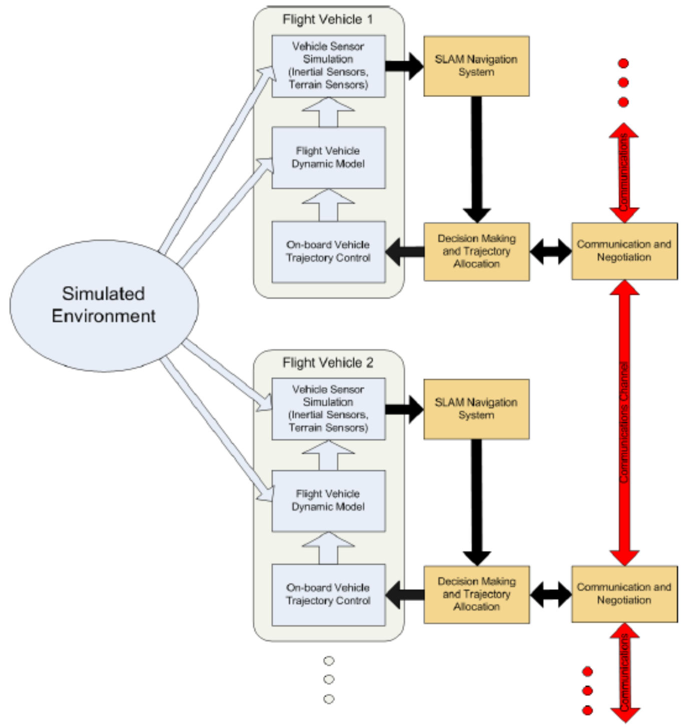


2 Important Results from the Theory on DDF SLAM

$$Y(t) = \begin{bmatrix} Y_{PP}(t) & Y_{PM}(t) \\ Y_{PM}^T(t) & Y_{MM}(t) \end{bmatrix}$$

- The platform to platform cross information is zero.
- The global map information is the sum of the map information from each platform.

Multi-Vehicle Active SLAM



Multi-Vehicle Active SLAM