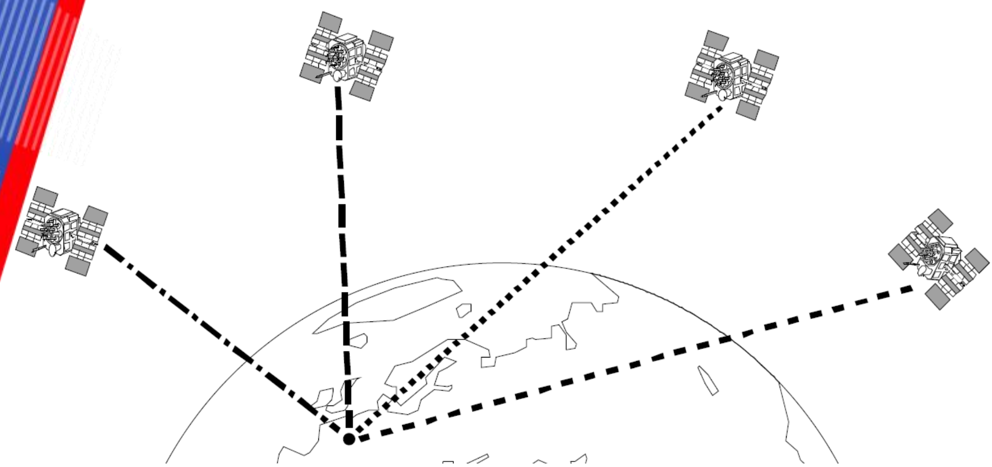
RRLAB
THE ROBOTICS RESEARCH LAB

Autonomous Mobile Robots (AMR)
6. Localization



Prof. Karsten Berns
Robotics Research Lab
Department of Computer Science
University of Kaiserslautern, Germany

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Contents

- Introduction
- Local Positioning (Odometry, INS, Optical Flow)
- Global Positioning (Landmarks, GNSS)

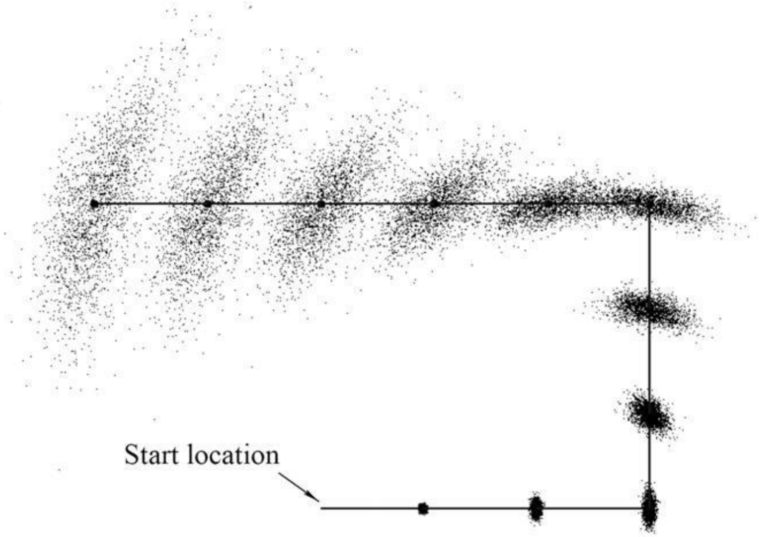
Localization

- Definition: Calculation of a mobile robot's position/ orientation relative to an external reference system
- Usually world coordinates serve as reference
- Basic requirement for several robot functions
  - Approach of target points, path following
  - Avoidance of obstacles, dead-ends
  - Autonomous environment mapping
- Key aspect in research for more than 30 years
- Often referred to as localization of robots

Localization

- Strong localization
  - Calculation of the robot pose in external reference frame
  - “Where am I?”
- Weak localization
  - Recognition of already visited positions
  - “Have I already been here before?”
- Global localization
  - Positioning without previous knowledge of own position
  - “Wake-up robot problem”
- Local localization
  - Positioning with previous knowledge
  - Example: known position before execution of a movement

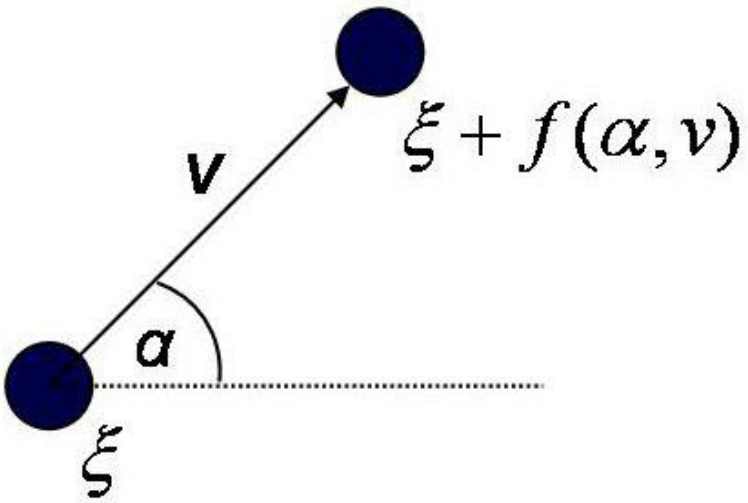
Local Positioning
Odometry, IMS, Optical Flow



Prof. Karsten Berns
Robotics Research Lab
Department of Computer Science
University of Kaiserslautern, Germany

Dead Reckoning

- Measurement of change in position relative to the starting position
  - Utilization of internal sensors sufficient
  - Example - positioning of car by speedometer and steering wheel rotation
- Only local localization possible
  - Change in orientation α and speed v measured internally
  - Knowledge of starting position ξ required



Odometry

- Odometry (gr.: hodós = “path”, métron = “measurement”)
- Calculation of the robot’s mileage by measuring of wheel rotation and wheel angle
- Dead reckoning of wheel driven AMR possible by usage of odometry
- Procedure
  - Measuring of wheel velocity in small time interval
  - Calculation of movement
  - Integration

Odometry

wheel speed — change in position — position

$$\longrightarrow \dot{\varphi}_r,\ \dot{\varphi}_l \longrightarrow \dot{\xi}_I = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \longrightarrow \xi_I = \int \dot{\xi}_I \, dt$$

measurement — forward kinematics — integration

$$\dot{x} = v \cos \varphi(t)$$
$$\dot{y} = v \sin \varphi(t)$$

Example for differential drive in $2\mathrm{D}$

Odometry

- Integration over time causes errors like
  - Wheel slip
  - Variations in suspension/wheel size
- Positioning by odometry only suitable for close-range

Odometry – Linear Approximation

Considering that at each time step Δt the next speed v_i and ω_i are measured, $x(t)$, $y(t)$ and $\varphi(t)$ can be calculated:

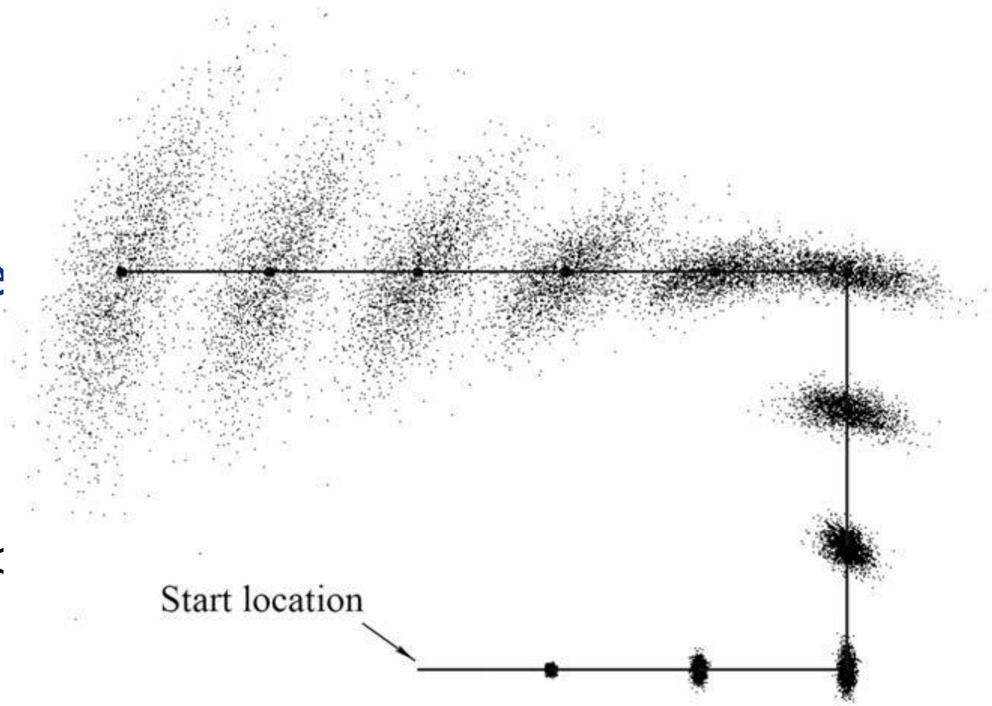
$$\varphi(t_i) \rightarrow \{\varphi_i\} \text{ with } \varphi_i = \varphi_{i-1} + (\omega_{i-1} + \omega_i) \cdot \frac{\Delta t}{2}$$

$$x(t_i) \rightarrow \{x_i\} \text{ with } x_i = x_{i-1} + (v_{i-1} \cdot \cos\varphi_{i-1} + v_i \cdot \cos\varphi_i) \cdot \frac{\Delta \mathrm{t}}{2}$$

$$y(t_i) \rightarrow \{y_i\} \text{ with } y_i = y_{i-1} + (v_{i-1} \cdot \sin\varphi_{i-1} + v_i \cdot \sin\varphi_i) \cdot \frac{\Delta t}{2}$$

Odometry – Pose Error

- Positioning error increases with mileage
- Orientation errors have bigger effects than distance errors
- Typical error level
  - distance deviation (after translatory movement): $< 0.03\ \%$
  - angle deviation (after rotation): $\approx 0.8\ \%$
- Complex routes cause significantly higher deviations



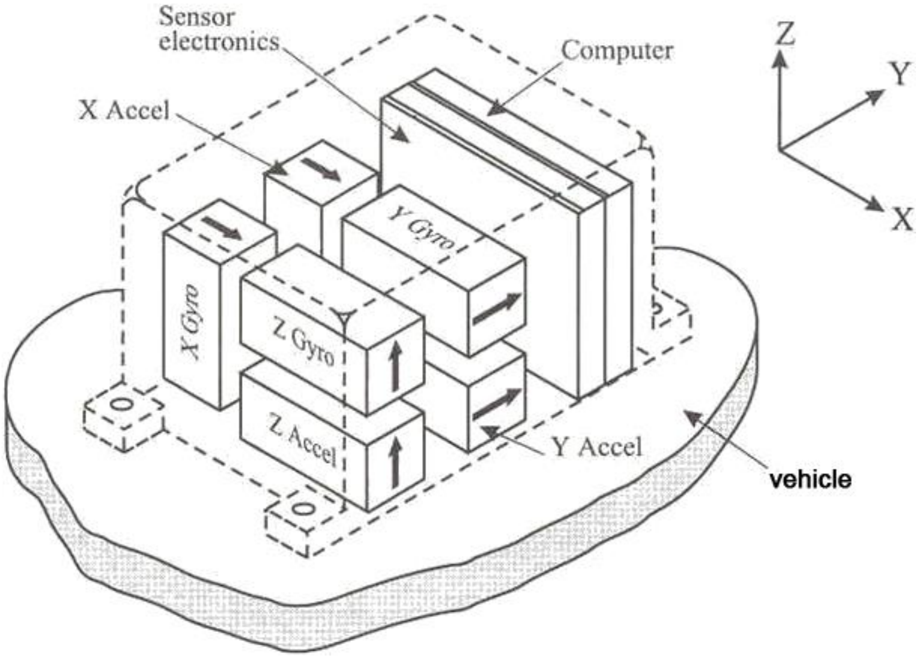
Distribution of the estimated robot position

Inertial Measurement System (IMS)

- Other dead reckoning method
- Measure accelerations $\ddot{x}$, $\ddot{y}$, $\ddot{z}$ and turning rates ω_x, ω_y, ω_z
- Sensor systems
  - Mechanical gyroscope
  - Fiber optical gyro
  - Micro-mechanical device
- No errors due to slip or unsteady wheel parameters
- But: double integration for positioning necessary → errors due to drift
- Integrated devices
  - Inertial Measurement Unit (IMU)
  - Inertial Measurement System (IMS)
  - Inertial Navigation System (INS)

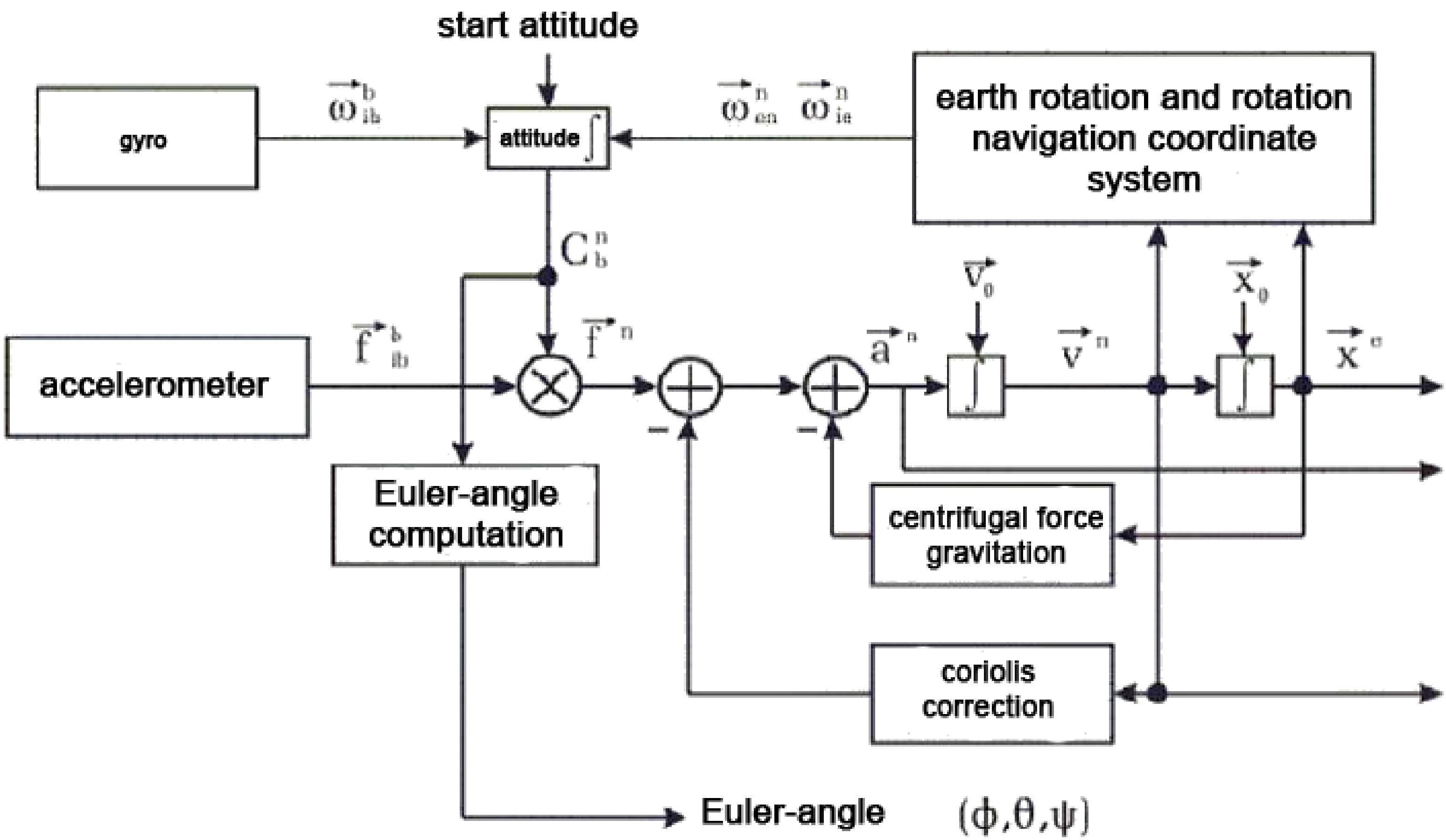
Inertial Measurement System (IMS)

- Regard inner information of a system: forces and angular velocities
- Change in direction of movement manifests in acceleration force and angular velocity
- Interpretation as change in pose



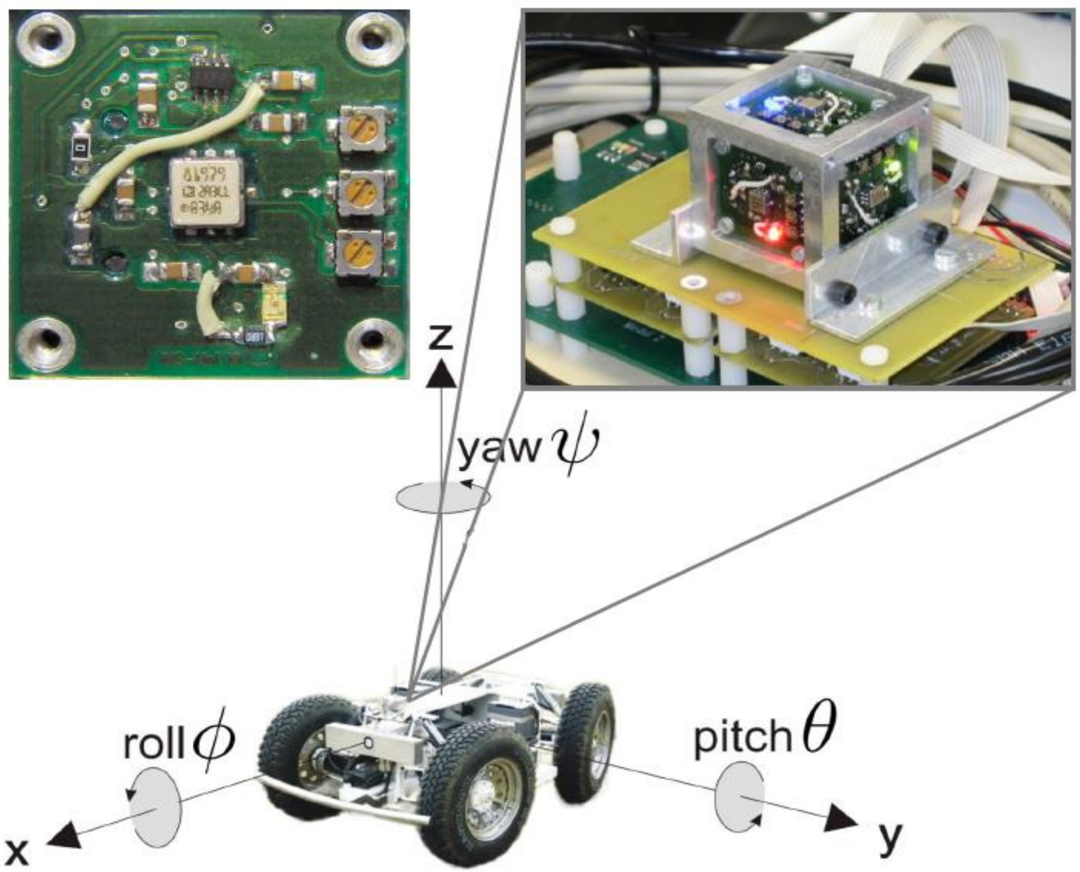
Components of an inertial measurement system

Strapdown Algorithm

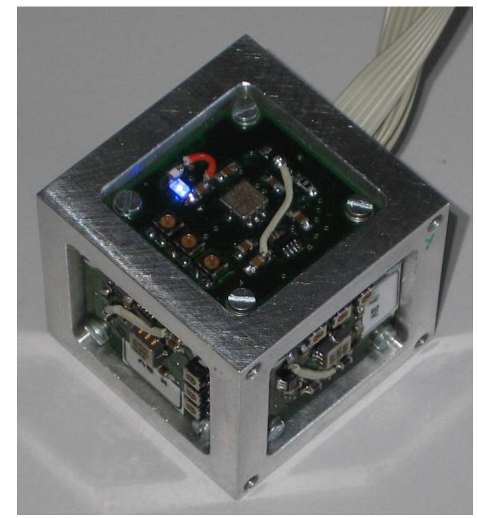


General algorithm for pose calculation: Strapdown algorithm

Application of IMS of RRLab



Application in robot RAVON



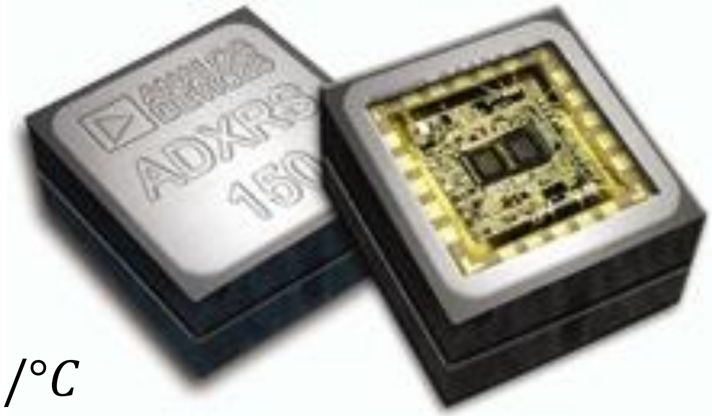
IMS of RRLab

Example – IMS of RRLab

- Based on the data gathered by the sensor the following information can be determined:
  - Acceleration: slip detection at translation
  - Velocity: navigation control
  - Angular velocity: slip detection in curves
  - Position: course correction
  - Attitude: safety

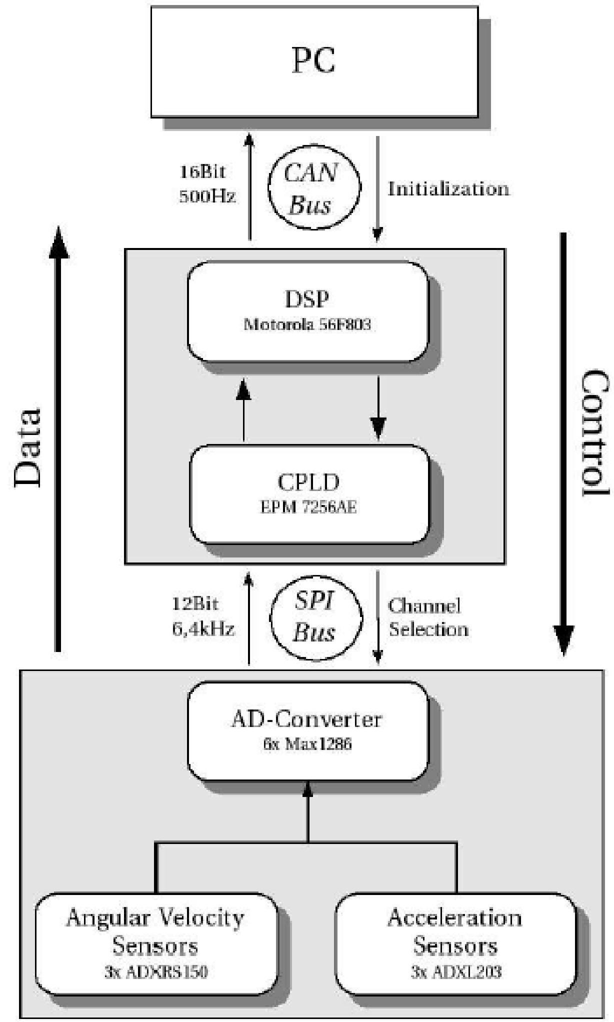
Used Sensors for IMS of RRLab

- Angular rate sensor ADXRS150
  - Measurement area: $\pm 150\ °/\mathrm{s}$ or $\pm 2.62\ \mathrm{rad/s}$
  - Non-linearity: $0.1\ \%$
  - Acceleration influence: $0.023\ °/\mathrm{s/g}$
  - Temperature influence: $15\ \%$ or $21.7\ °/s$ ($-40\ °C$ to $+85\ °C$), $0.17\ °/s/°C$
- Acceleration sensor ADXL203
  - Measurement area: $1.7\ \mathrm{g}$ or $16.67\ \mathrm{m/s}$
  - Non-linearity: $0.5\ \%$
  - Temperature influence: $0.3\ \%$ (full scale)
  - Cross axis influence: $2\ \%$
  - Alignment error on chip: $\pm 1.0\ °$



Rate sensor ADXRS150

System Architecture



Notations

- $\vec{\omega}$ measured angular velocities
- $\vec{\sigma}$ angles from integration of angular velocity
- ϕ roll-angle around x-axis
- θ pitch-angle around y-axis
- ψ yaw-angle around z-axis
- $\vec{a}$ accelerations
- $\vec{g}$ gravity force
- $\vec{v}$ velocities
- $\vec{s}$ position
- $\vec{u}$ rotation axis for attitude correction heuristic
- α rotation angle for attitude correction heuristic

Notations

- t time interval
- W matrix with unit vectors of global frame
- B matrix with unit vectors of local frame
- T rotation tensor, translates from local to global frame
- ϵ tolerance interval
- lower index b vector/matrix referred to local frame
- lower index w vector/matrix referred to global frame
- double lower index $\vec{\alpha}, \vec{\beta}$ tensor rotates around axis $\vec{\alpha}$ by angle β
- upper index k vector/matrix/tensor in kth time step

IMS – Pose Calculation

Matrix W_W containing unit vectors of world frame and matrix B_W with unit vectors of body frame

$$W_W = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$B_W = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}$$

B_W is called **direction cosine matrix** since the element in b_{ij} represents cosine of angle between i-axis of reference frame and j-axis of body frame

IMS – Pose Calculation

- Angle change per time step $\vec{\sigma}_b^k$ is deduced by trapezoid integration of the current and last angular velocity measurement $\vec{\omega}_b^k$, $\vec{\omega}_b^{k-1}$ and accordingly transformed into the world frame

$$\vec{\sigma}_b^k = \frac{t^k\left(\vec{\omega}_b^k + \vec{\omega}_b^{k-1}\right)}{2}$$

$$\vec{\sigma}_w^k = B_w^k \vec{\sigma}_b^k$$

- Attitude information is updated by means of multiplication with a rotation tensor
- Rotation axis represented in this tensor has the actual angular rate as components
- Rotation angle is derived as the length of the actual angular velocity vector

$$B_w^{k+1} = T_{\vec{\sigma}^k, |\vec{\sigma}^k|} \cdot B_w^k$$

IMS – Pose Calculation

- High update rate allows for application of infinitesimal transformation

$$R(\alpha) = I + \alpha J$$

  - I = identity matrix
  - α = infinitesimal rotation angle
  - J = infinitesimal matrix

$$B_w^{k+1} = T_{\vec{\sigma}^k, |\vec{\sigma}^k|} \cdot B_w^k$$

$$B_w^{k+1} = \left(I + \left|\vec{\sigma}^k\right| J\right) \cdot B_w^k$$

$$B_w^{k+1} = \begin{pmatrix} 1 & -\omega_z^k & \omega_y^k \\ \omega_z^k & 1 & -\omega_x^k \\ -\omega_y^k & \omega_x^k & 1 \end{pmatrix} \cdot B_w^k$$

IMS - Representation as Euler angles

- Matrix represent three sequential rotations by the angles ϕ, θ and ψ around the world axis x, y, and z, or the body axis in opposite order (c: cos, s: sin):

$$B_w = \begin{pmatrix} c\psi\, c\theta & -s\psi\, c\phi + c\psi\, s\theta\, s\phi & s\psi\, s\phi + c\psi\, s\theta\, c\phi \\ s\psi\, c\theta & c\psi\, c\phi + s\psi\, s\theta\, s\phi & -c\psi\, s\phi + s\psi\, s\theta\, c\phi \\ -s\theta & c\theta\, s\phi & c\theta\, c\phi \end{pmatrix}$$

- Determination of angles by coefficient comparison from B_w

$$\phi = \arctan \frac{b_{32}}{b_{33}}$$

$$\theta = \arcsin -b_{31}$$

$$\psi = \arctan \frac{b_{21}}{b_{11}}$$

IMS – Integration of Acceleration

- By multiplication of the acceleration vector $\vec{a}_B$ with the matrix B_w gives the acceleration in world coordinates $\vec{a}_W$

- Trapezoid integration of $\vec{a}_W$ at time step k will result in the linear velocity $\vec{v}_w^k$
- Trapezoid integration of $\vec{v}_w^k$ at time step k will result in the position vector $\vec{s}_w^k$

$$\vec{g}_w = \begin{pmatrix} 0 \\ 0 \\ 9.80665\,\mathrm{m/s^2} \end{pmatrix}$$

$$\vec{a}_w = B_w \vec{a}_B - \vec{g}_w$$

$$\vec{v}_w^k = \vec{v}_w^{k-1} + t^k \cdot \frac{\vec{a}_w^k + \vec{a}_w^{k-1}}{2}$$

$$\vec{s}_w^k = \vec{s}_w^{k-1} + t^k \cdot \frac{\vec{v}_w^k + \vec{v}_w^{k-1}}{2}$$

IMS - Attitude Heuristics

- Minimize drift of position and attitude by adjusting the attitude with respect to the gravity force
- Determination of attitude from the angular velocities as presented before (when vehicle does not move)
- If the amount of the acceleration determined equals the standard gravity force attitude is o.k.

$$|\vec{a}_w| \leq \epsilon$$

- If it differs a correction of the attitude representation is done

IMS - Attitude Heuristics

- Correction angle is calculated by the scalar product between the two vectors

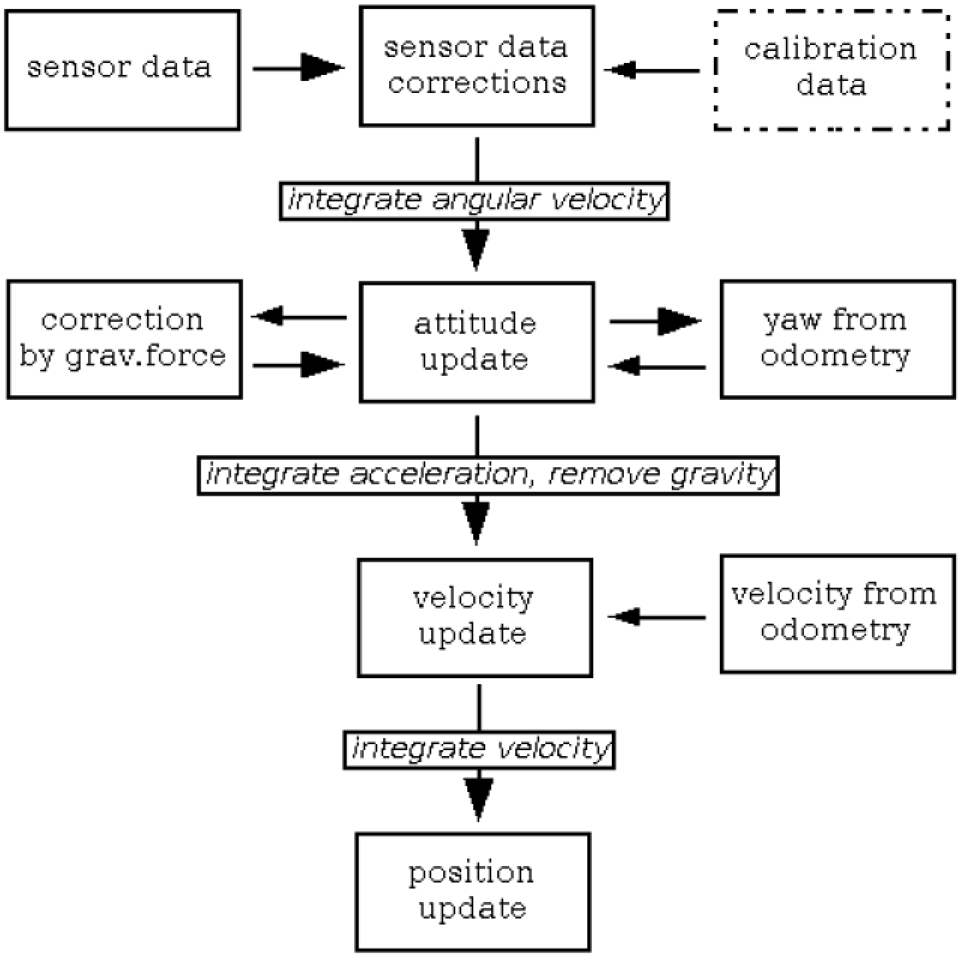
$$\alpha = \arccos\left(\frac{(B_w \vec{a}_b)\vec{g}}{|B_w \vec{a}_b||\vec{g}|}\right)$$

- As a result of the lack of one dimension, only the roll and pitch values can be corrected by this method

$$\vec{u} = (B_w \vec{a}_b) \times \vec{g}$$

$$B_w = T_{\vec{u},\alpha} \cdot B_w$$

IMS Summary of Algorithm



Modified strapdown algorithm

IMS - Error Sources

- Assembly errors
  - Sensor alignment on board
  - Rectangularity of board axis
- Electrical errors
  - Voltage fluctuations
  - System noise
  - Digital jitter
- Sensor errors
  - Characteristic curve errors
  - Package alignment on chip
  - Temperature dependence
  - Cross axis influence
  - Acceleration influence (angular rate sensors)

Localization based on Optical Flow

- Current localization methods do not offer enough precision
- Idea - in nature visual information is used for localization
- Visual odometry - motion tracking based on comparison of two consecutive images
- Advantage - errors not correlated with localization approaches

Optical Flow Algorithm

- Requirement - stereo camera system on robot
- Extract features from a intensity image (Harris Corner)
- Capture next image
- Extract features + compute optical flow of features
- Determine motion based on optical flow

Image processing

Step 1: Feature extraction

Apply Harris Corner Detection to identify stable feature in one camera image

Feature selection performed

Image processing

Step 2: Stereo Matching and Triangulation

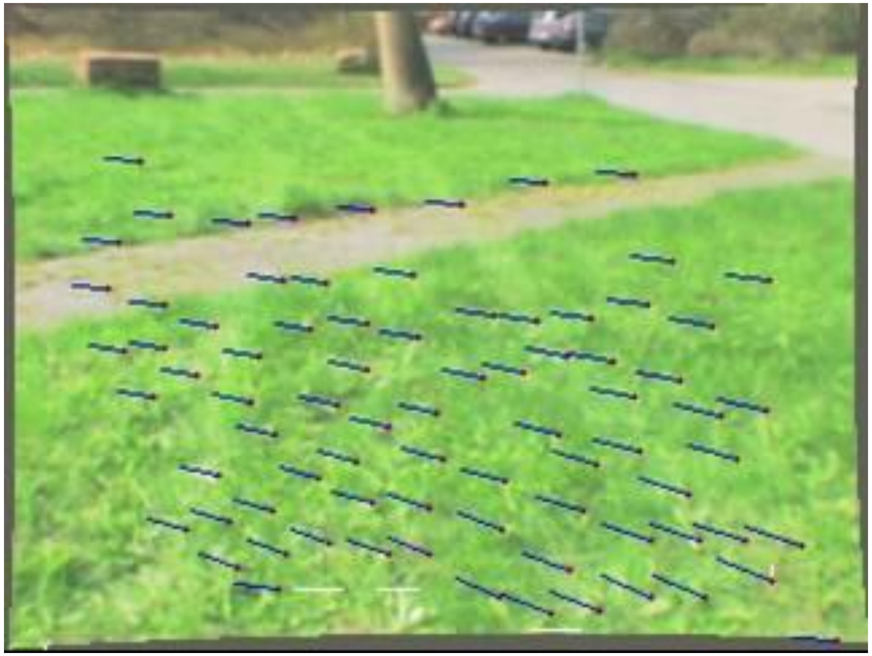
- Drop features that do not match in both images
- Determine 3D coordinates of feature points (stereo image processing)

Image processing

Image processing

Step 3: Feature Tracking

- After robot motion - extract features in new image
- Discard wrong vectors employing standard deviation
- Problems - large moving objects close to camera
- Compute 3D motion of features



Feature tracking

Determination of Motion

- Given: $3\mathrm{D}$ feature locations before (L_i^b) and after (L_i^a) the robot moved

- Determine translation vector T and rotation matrix R of the robot

$$L_i^b = RL_i^a + T + e_i$$

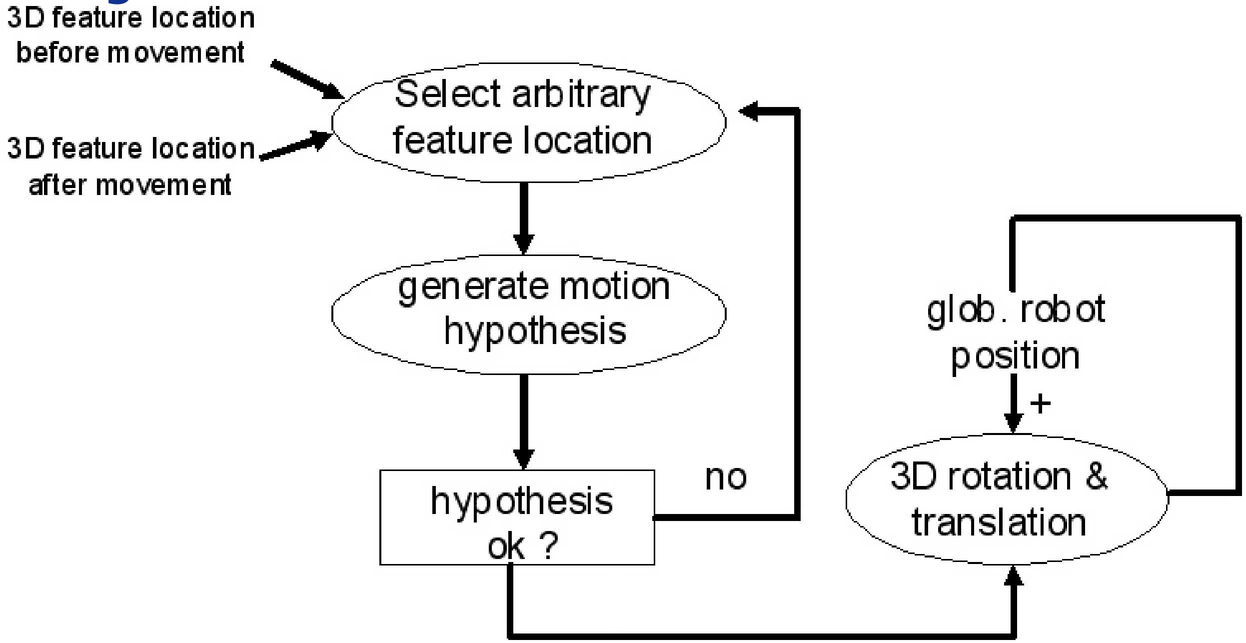
- Minimize error e_i

$$e^2 = \frac{1}{N}\sum_{i=1}^{n}\left\|L_i^b - (RL_i^a + T)\right\|$$

- Least square optimization problem (will be discussed later)

Second Minimization Solution - Ransac-Algorithm

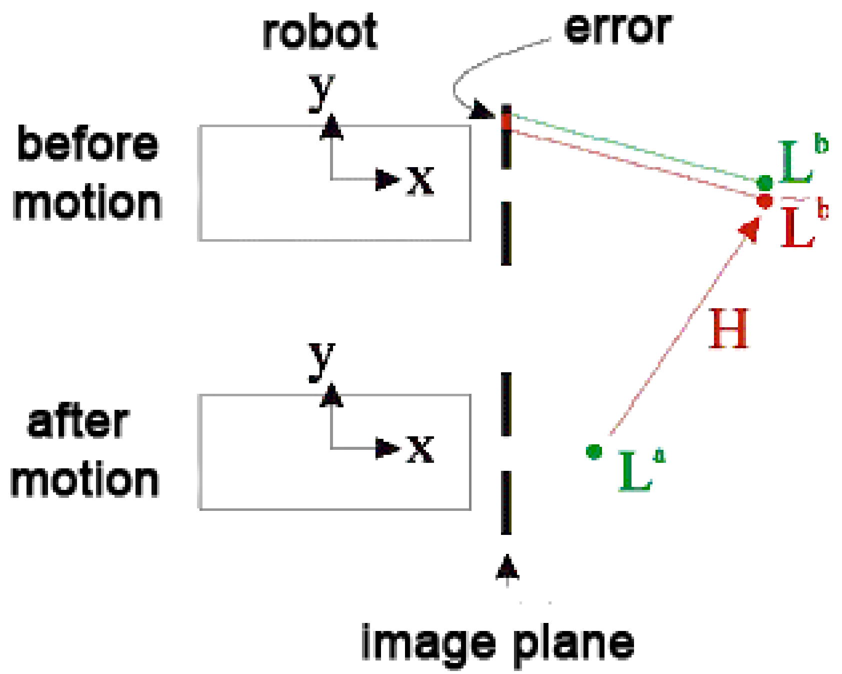
- Least square algorithm only works properly with correct optical flow vectors
- Disturbance by dynamical objects
- Ransac algorithm often delivers better solutions



Ransac algorithm

Motion Hypothesis

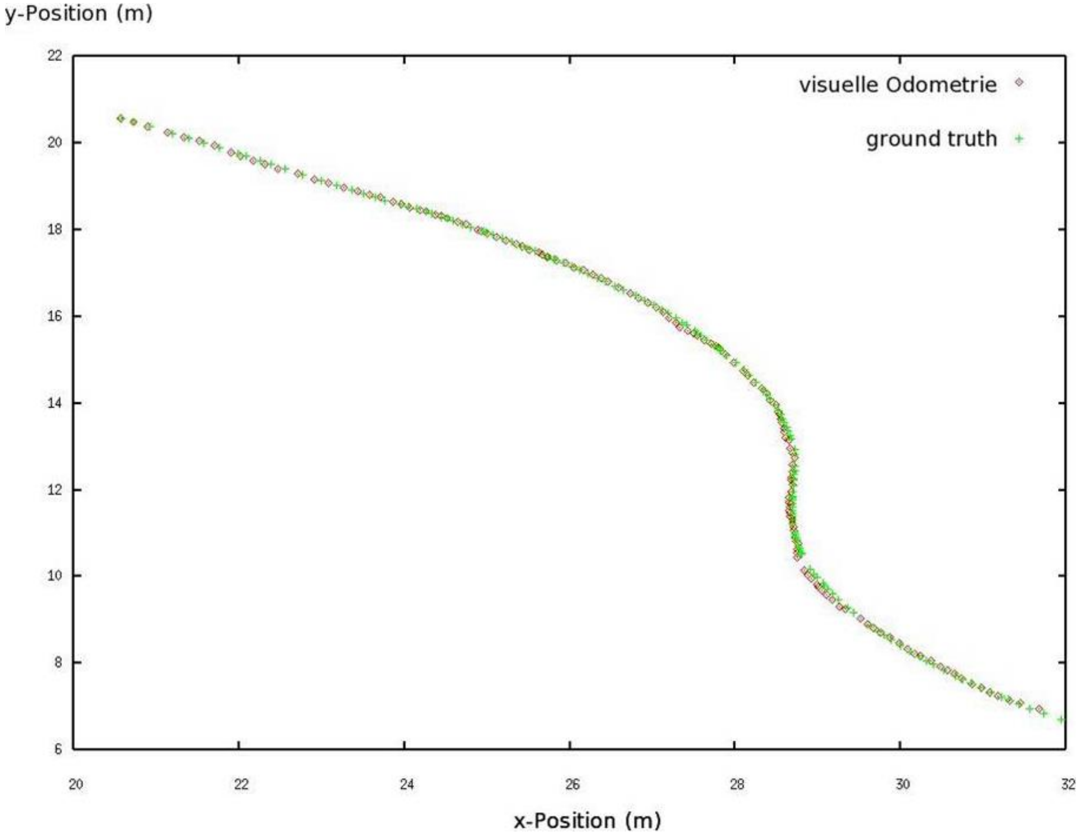
- Consider error for all feature points
- Find hypothesis that keeps all errors below threshold
- If no hypothesis satisfies criteria select best candidate



Motion hypothesis by minimizing error

Experiment with Ravon in Simulation

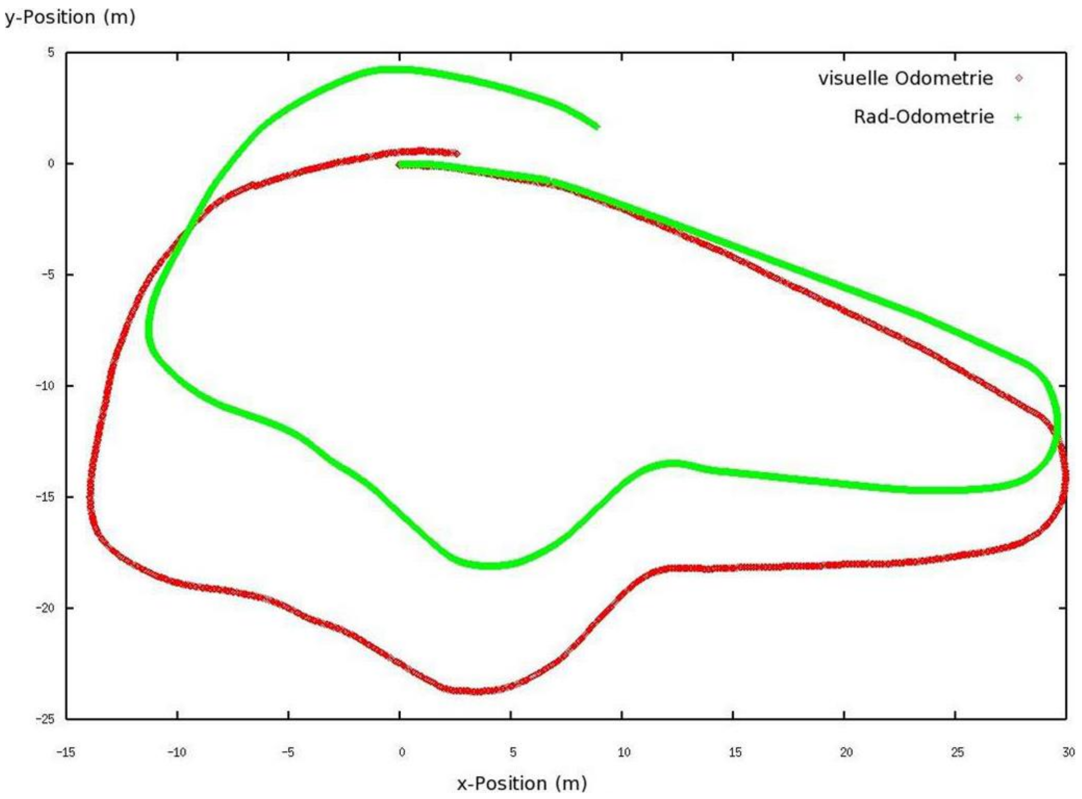
- Track length: 21 m
- Endpoint deviation
  1.75 % (0.38 m)
- Mean deviation
  0.15 m



Simulation results

Experiment in Real World (Meadow behind Building 48)

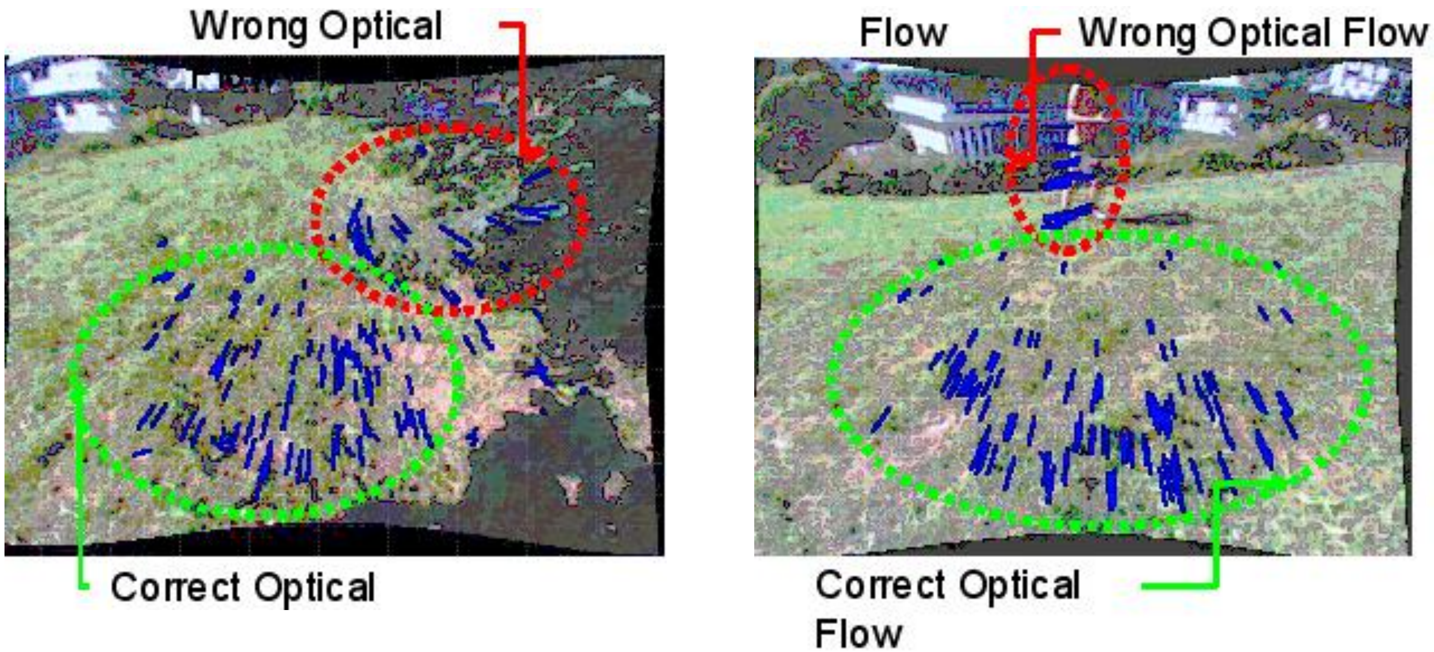
- Track length: $112\ \mathrm{m}$
- Deviation based on odometry $8.08\ \%$ $(9.05\ \mathrm{m})$
- Endpoint deviation $2.29\ \%$ $(2.57\ \mathrm{m})$
- Visible angle $33\ °$



Recorded data

Dynamic Environment

- Optical flow based on robot motion or object movement?
- Error sources - people, cars, animals, moving plants
- Errors increase without boundaries → no local stability
- Idea - use odometry data as filter



Visualization of optical flow

Prediction for Optimizing Optical Flow

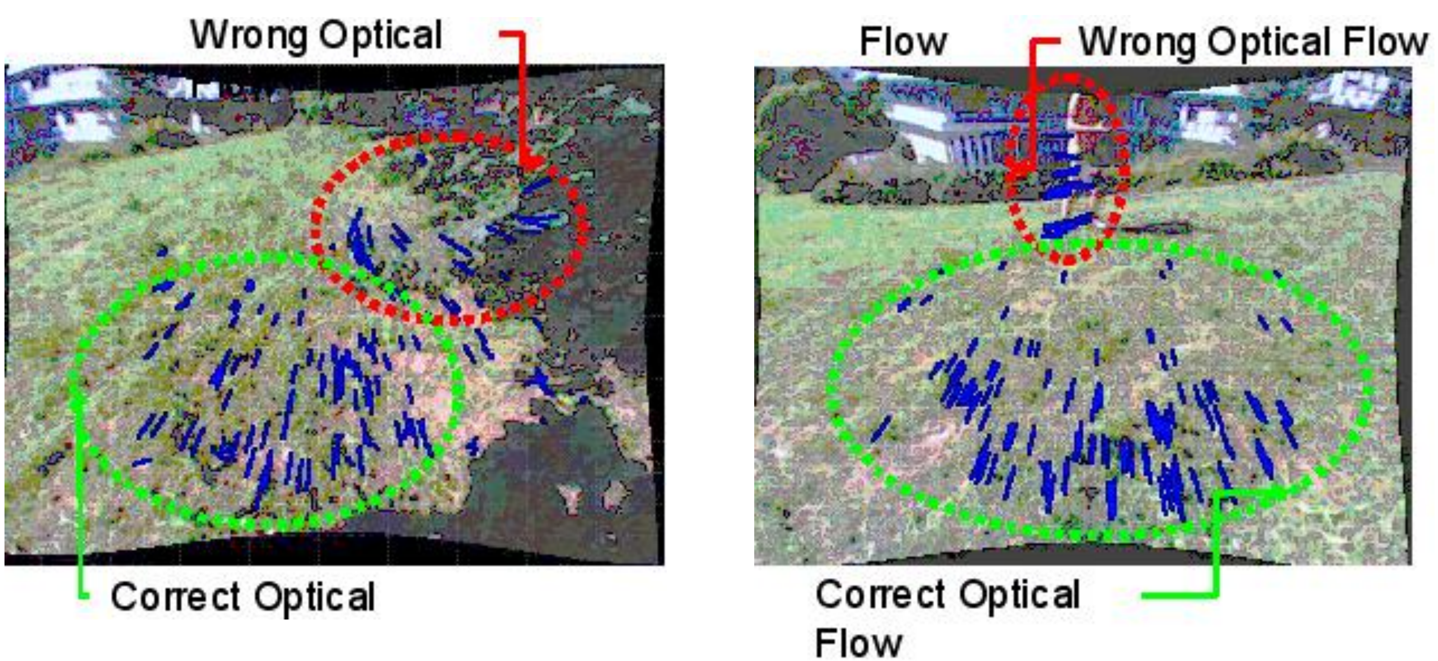
- Assume $S_{t_{i-1}}$ is a set detected features in step $i-1$
- Assume further $\Delta^{I}_{pose_i}$ is the change of pose p in between step $i-1$ and i
- Set of predicted features can be computed as:

$$\forall p \in S_{t_{i-1}}: p_{predict} = \Delta^{I}_{pose_i} + p$$

- Compute difference of captured and predicted point in step i
- Filter results using threshold

Results

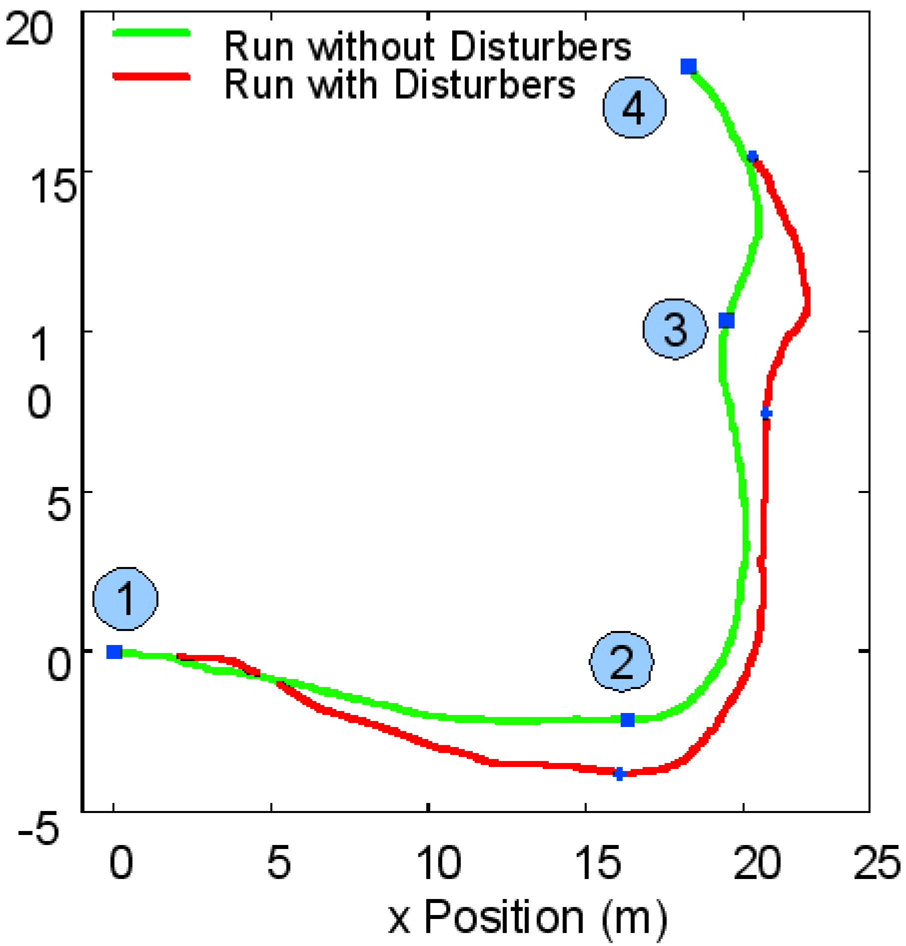
- Deviation in optical flow is determined reliably (red vectors)
- Sufficient number of vectors remain to determine position



Prediction of optical flow

Final Results in a real Environment

- Same track with and without interferences
- Checkpoints marked on ground
- Length approx. $40\ \mathrm{m}$
- Mean deviation $\approx 10\ \%$



Data recorder during test run

Scan Correlation

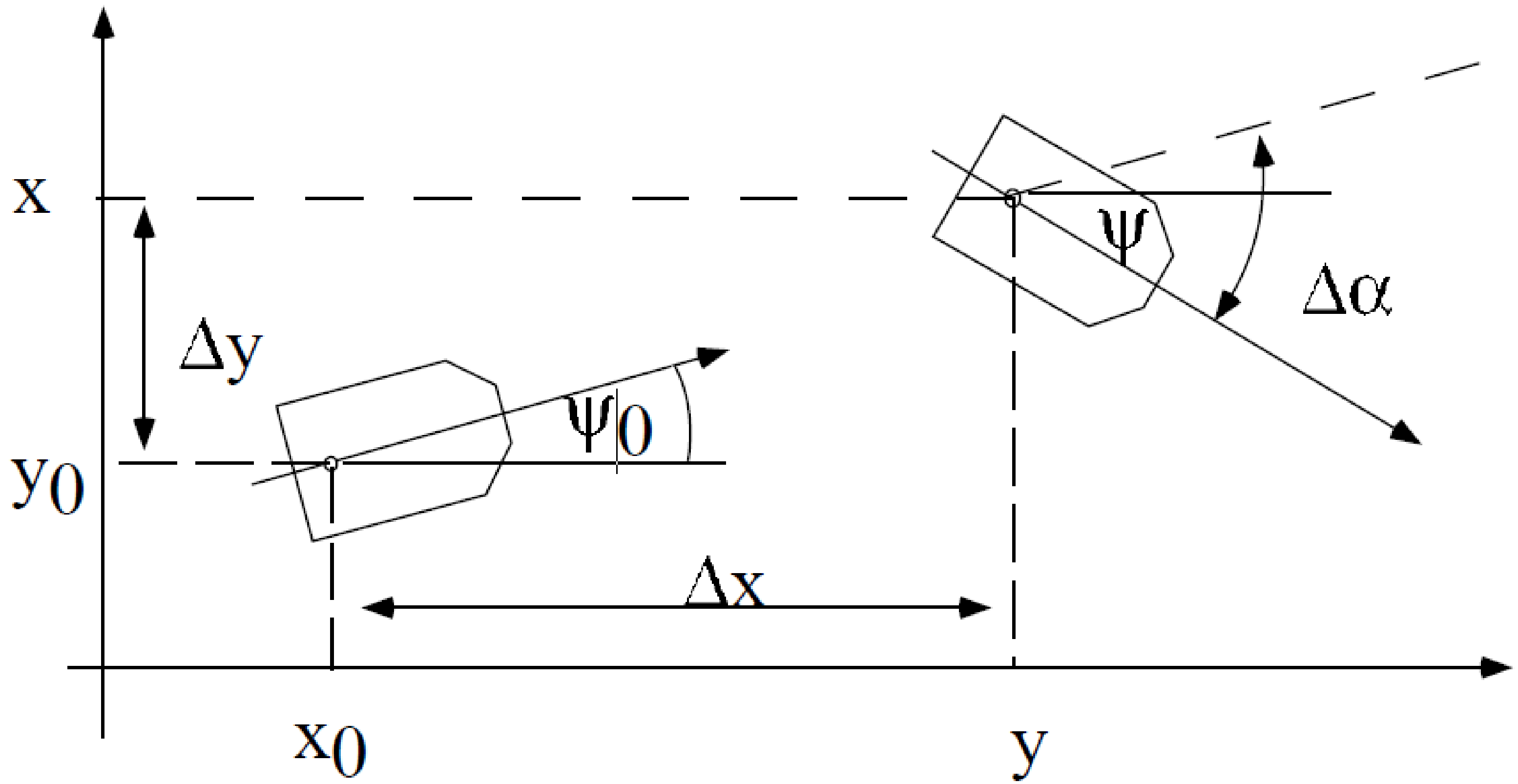
- Another form of “optical flow” can be applied in structured indoor environments

- Here, walls and furniture show geometric features (e. g. angles and distance) that can be extracted using laser scanners

Scan Correlation

- Histogram-based method for covering two scans
- Searches for local maxima inside histograms via cross correlation to get translational and rotational displacements
  - Calculate angle histograms for both scans
  - Rotate actual scan with rotation $\Delta\alpha$ calculated via phase shifting both histograms
  - Turn both scans via ψ so that their main directions are parallel to x-axis
  - Calculate histogram of allocation of x-values
  - Shift actual scan in x direction with calculated value Δx
  - Calculate histogram of allocation of y-values
  - Shift actual scan in y-direction with calculated value Δy
  - Rotate scans back with angle $-\psi$

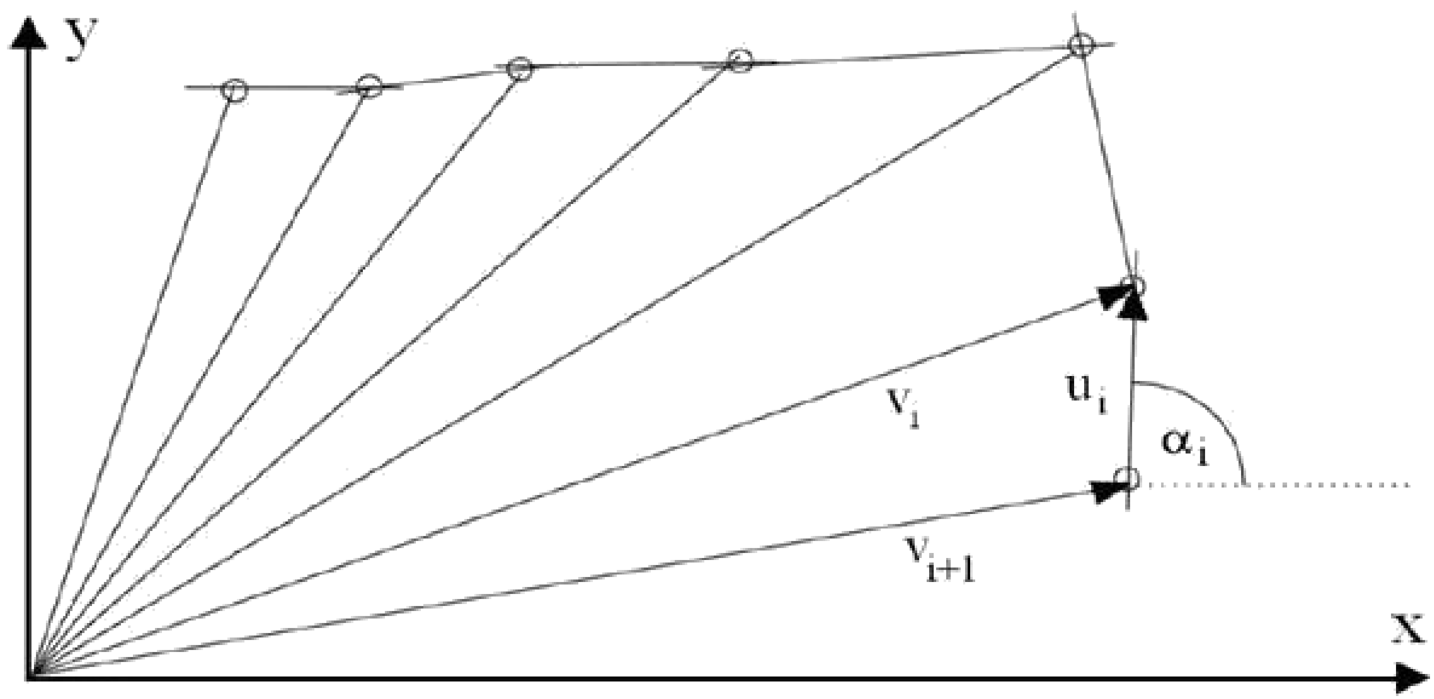
Pose Shift



Two scans are taken from different poses shifted by $(\Delta x, \Delta y, \Delta\alpha)$

Edge Extraction

- Calculate difference of two position vectors $u_i = v_{i+1} - v_i$
- Compute angle α_i between vector u_i and x-axis
- Frequency distribution of all angles is generated



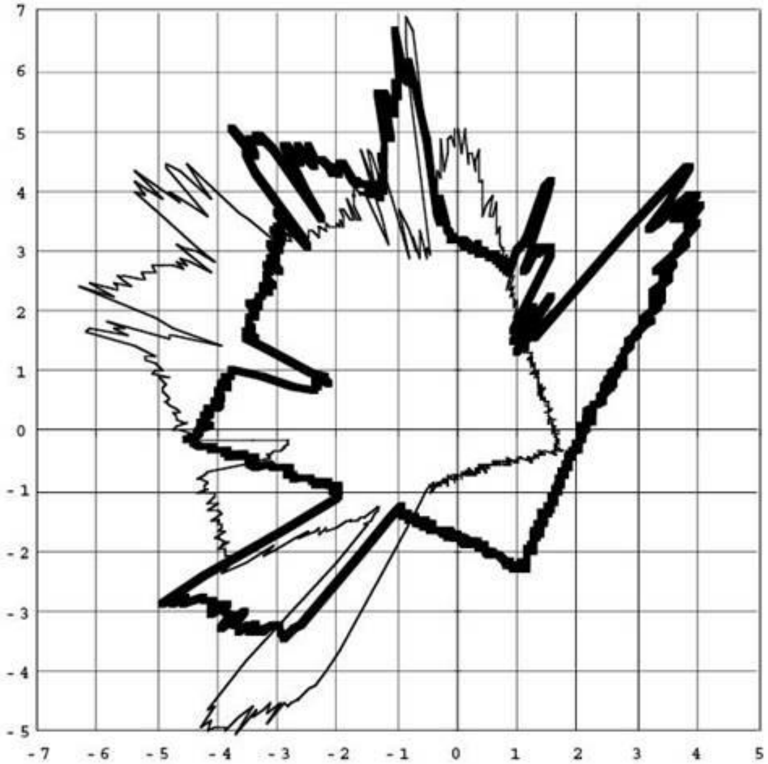
Edge Extraction

- Recording of ranging information via laser $\rightarrow \{r_i, \varphi_i\}$
- Robot-centered coordinates: $x_i^* = r_i \cdot \cos\varphi_i,\ y_i^* = r_i \cdot \sin\varphi_i$
- Determine angle α_i between (x_i, y_i) and $(x_{i+k}, y_{i+k}),\ k = 3 \ldots 5$ to compensate statistical errors

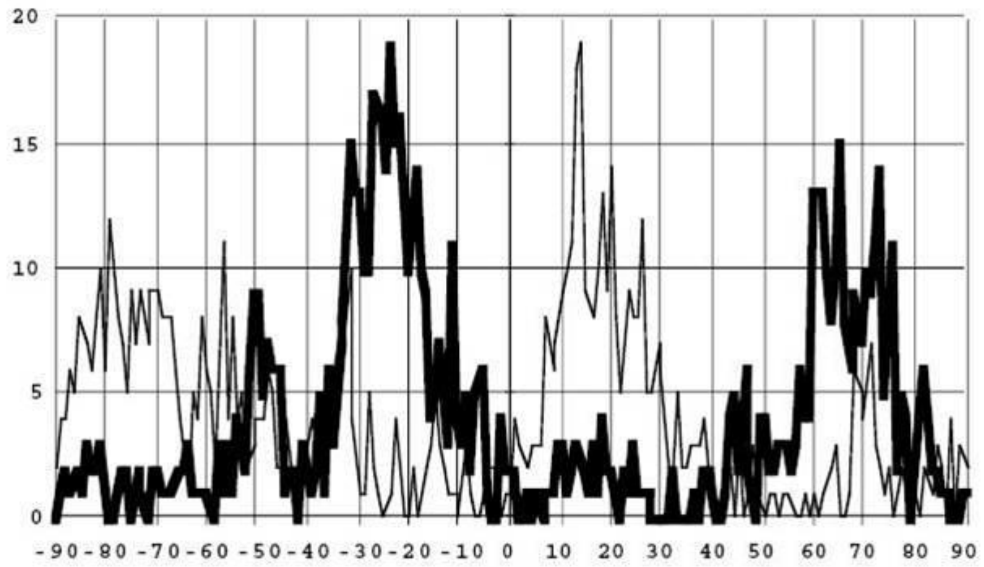
$$\tan\alpha_i = \frac{y_i^* - y_{i+k}^*}{x_i^* - x_{i+k}^*} = \frac{r_i \sin\varphi_i - r_{i+k}\sin\varphi_{i+k}}{r_i \cos\varphi_i - r_{i+k}\cos\varphi_{i+k}}$$

- To just determine the general direction: $\alpha \bmod 180\,°$

Angle Histogram (Orientation)



Exemplary scans to be matched



Frequency distribution of all angles

Angle Histogram

- Angle histogram $\{N_k\}$ with $0 \le k \le K$ divides the full circle $(360\,°)$ into K bins and counts the scanned points per bin

- Median of the histogram is

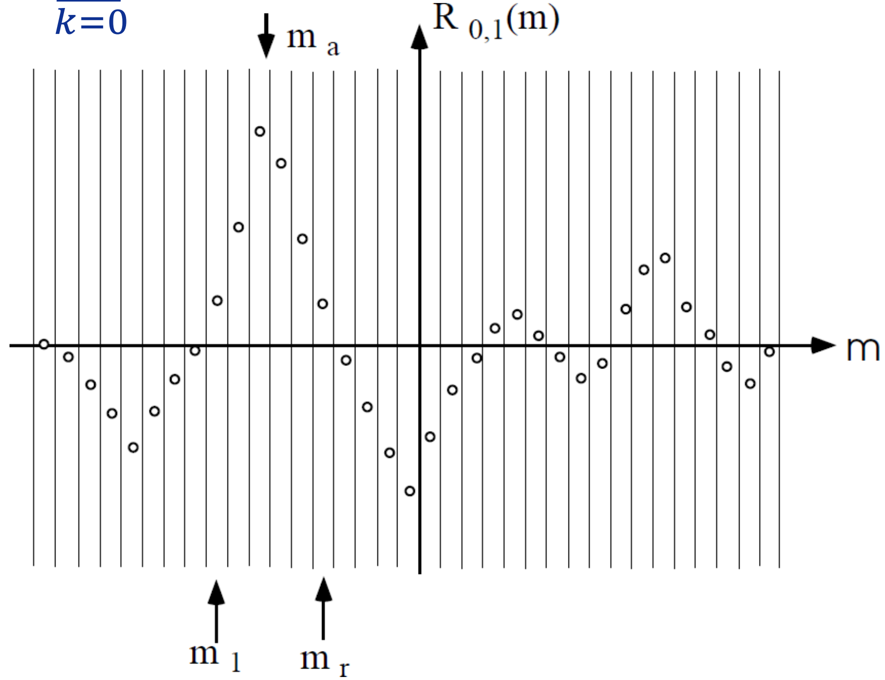
$$\widetilde{N} = \frac{1}{K} \cdot \sum_{k=0}^{K-1} N_k$$

Discrete Correlation of Angle Histograms

For two histograms $\{N_k\}_0$ and $\{N_k\}_1$

$$R_{0,1}(m) = \sum_{k=0}^{K-1} \left(N_{k,0} - \widetilde{N}_0\right) \cdot \left(N_{k-m,1} - \widetilde{N}_1\right)$$

is a function of the discrete shifts m



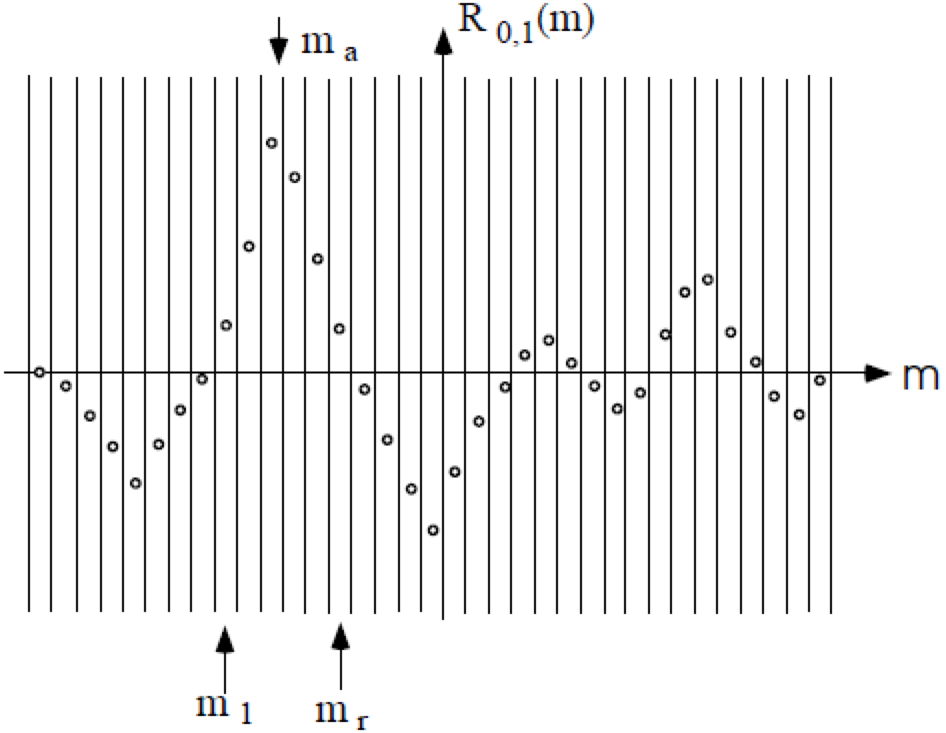
Correlated angle histograms

Interpretation of the Correlation

If correlation has a pronounced maximum m_a between zeros, the most probable angle of rotation is

$$\Delta\alpha = \gamma \cdot \frac{\sum_{m=m_l}^{m_r} R_{0,1}(m) \cdot m}{\sum_{m=m_l}^{m_r} R_{0,1}(m)}$$

with γ being the angular size of one bin



Correlated angle histograms

Position Histogram Correlation

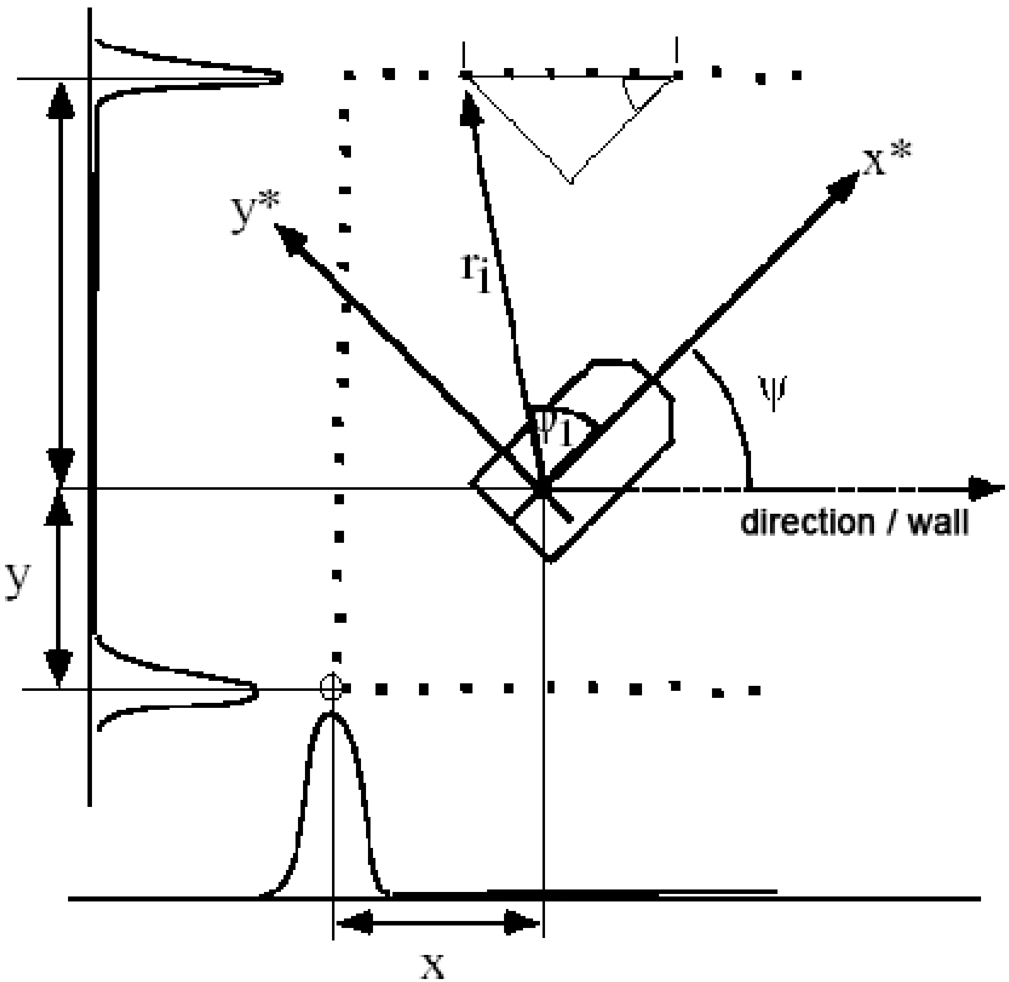
- Means of the position histogram (in x,y)

$$\tilde{H}_{0x} = \frac{1}{K}\sum_{n=0}^{k-1} H_{0x,n} \qquad \tilde{H}_{1x} = \frac{1}{K}\sum_{n=0}^{k-1} H_{1x,n} \qquad \tilde{H}_{0y} = \frac{1}{K}\sum_{n=0}^{k-1} H_{0y,n} \qquad \tilde{H}_{1y} = \frac{1}{K}\sum_{n=0}^{k-1} H_{1y,n}$$

- Correlate H_{0x} and H_{1x}: $Qx_{0,1}(m) = \sum_{n=0}^{k-1}\left(H_{0x,n} - \tilde{H}_{0x}\right)\cdot\left(H_{1x,n-m} - \tilde{H}_{1x}\right)$
- Correlate H_{0y} and H_{1y}: $Qy_{0,1}(m) = \sum_{n=0}^{k-1}\left(H_{0y,n} - \tilde{H}_{0y}\right)\cdot\left(H_{1y,n-m} - \tilde{H}_{1y}\right)$
- Calculate $Qx_{0,1}(m)$ and $Qy_{0,1}(m)$ for $-(N-P) \leq m < (N-P)$ to find maxima m_x and m_y
- Determine translation in x- and y-direction

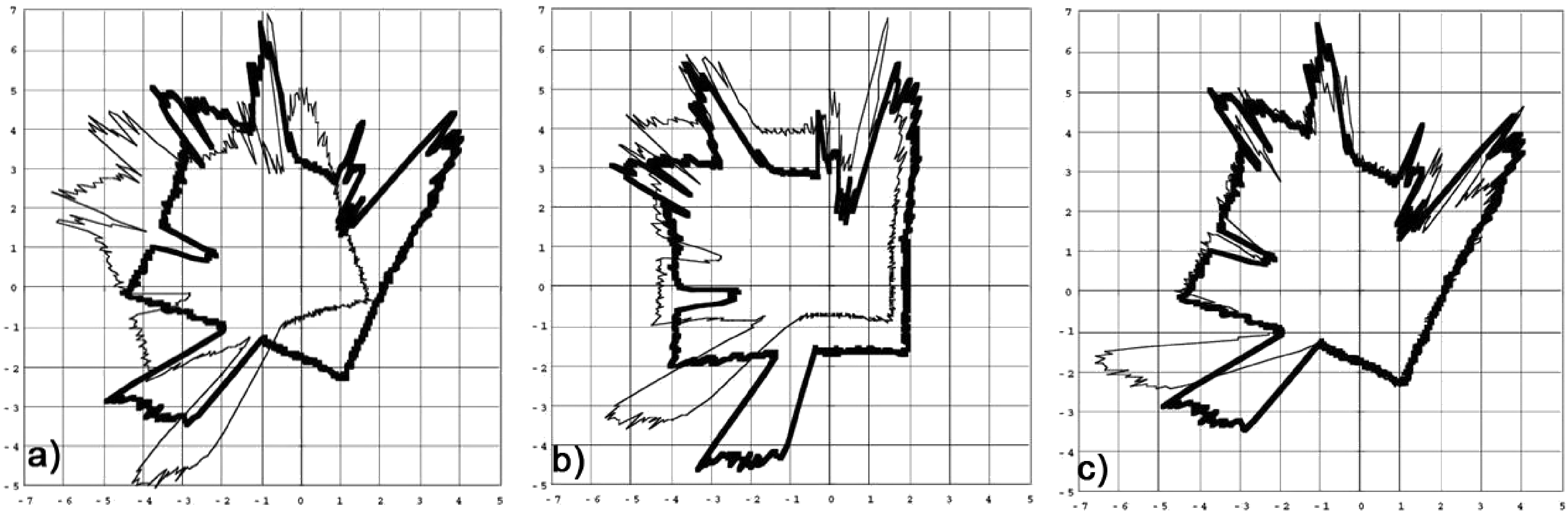
$$\Delta x = m_x \cdot \delta x,\ \Delta y = m_y \cdot \delta y$$

Position Histogram Correlation

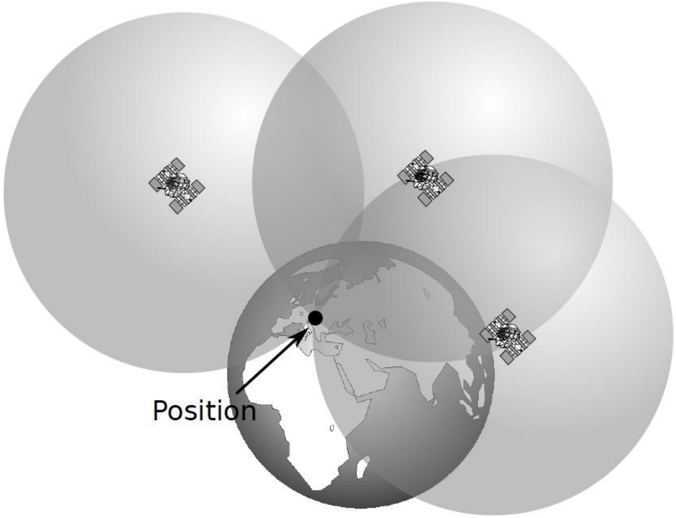


Summary of Steps during Calculation

- Situation of two displaced scans
- Both scans after rotation with calculated angle $\Delta\alpha$ and turned parallel to x-axis via angle ψ
- Scans after translation $(\Delta x, \Delta y)^T$ and turned back with $-\psi$



Global Positioning
Landmarks, GNSS



Prof. Karsten Berns
Robotics Research Lab
Department of Computer Science
University of Kaiserslautern, Germany

Global Localization

- Task: AMR "wakes up" and looks around. Where is it at?
- Requirement: the wakeup area is known to the AMR
- Solutions
  - Scan the current scene and compare with reference scans
  - The current scan is not identical to the reference scan but it resembles some more than others (location most likely there)
  - Drive several predefined paths and capture scans → accumulation of probabilities → location

Landmarks

- A Landmark causes a significant variation in the data collected by the sensors of a mobile robot system
  - Association with a certain position
  - Recognition can be performed fast and reliable
- Correction of errors in reckoning methods through consideration of external sources of information
  - No imprecision due to adding up of internal measurements

Classification of Landmarks

- Natural landmarks: Objects already present in the environment with appropriate properties
- Artificial landmarks: Objects added to the environment for positioning purposes
- Active landmarks: Active emission of a signal (active energy supply is needed) which can easily detected by a sensor system
- Passive landmarks

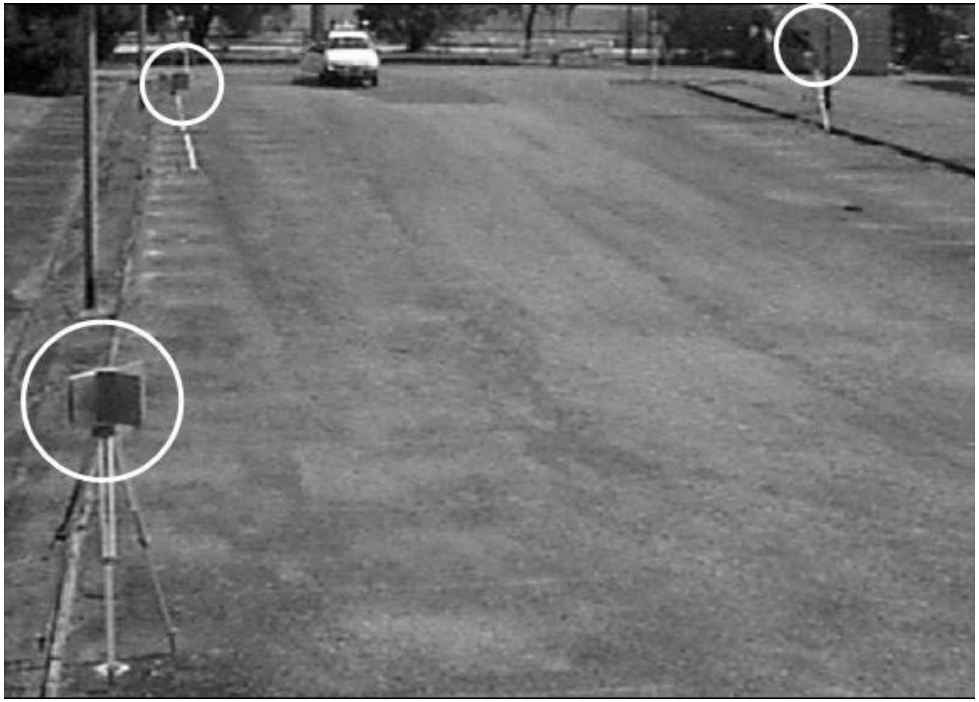
Example

- Lighthouse: artificial, active
- Detection by recognition bright spot patterns in the camera snapshots

Lighthouse

Example

- Radar reflector: artificial, passive
- Detection by filtering of radar echoes according to magnitude and stability aspects

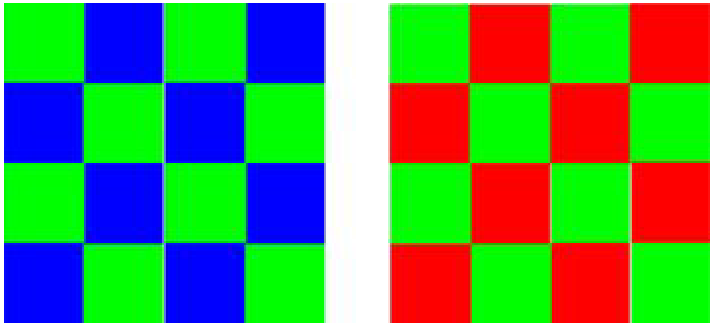


Radar reflectors

Landmark Recognition

- Usually specific methods have to be applied to perform landmark recognition
- Robust recognition methods in an complex environment are the main problem of localization with landmarks
  - Limitation on specific environment / landmark types
  - Application of complex image- and signal processing methods
- Assumptions for the following:
  - Distance to landmarks
  - Relative angle between landmarks
  - Explicit identification of landmarks

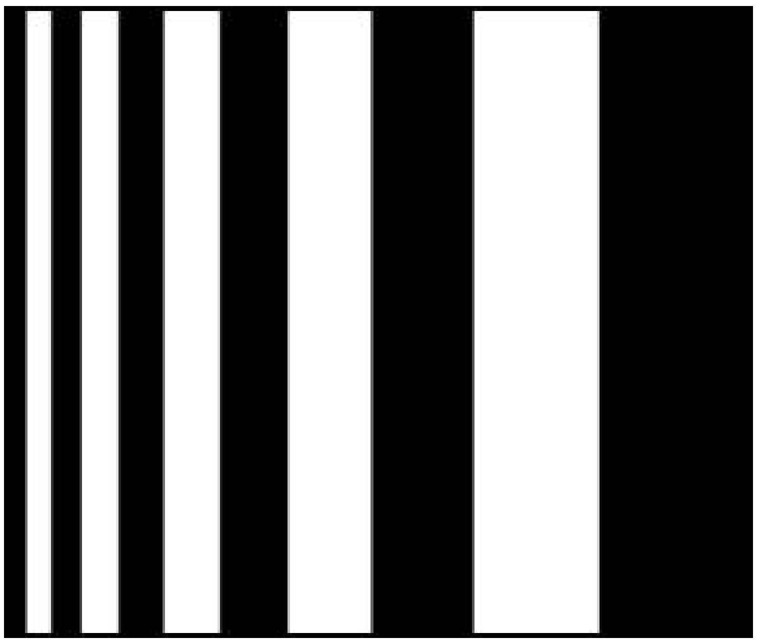
Camera-based Artificial Landmark Recognition

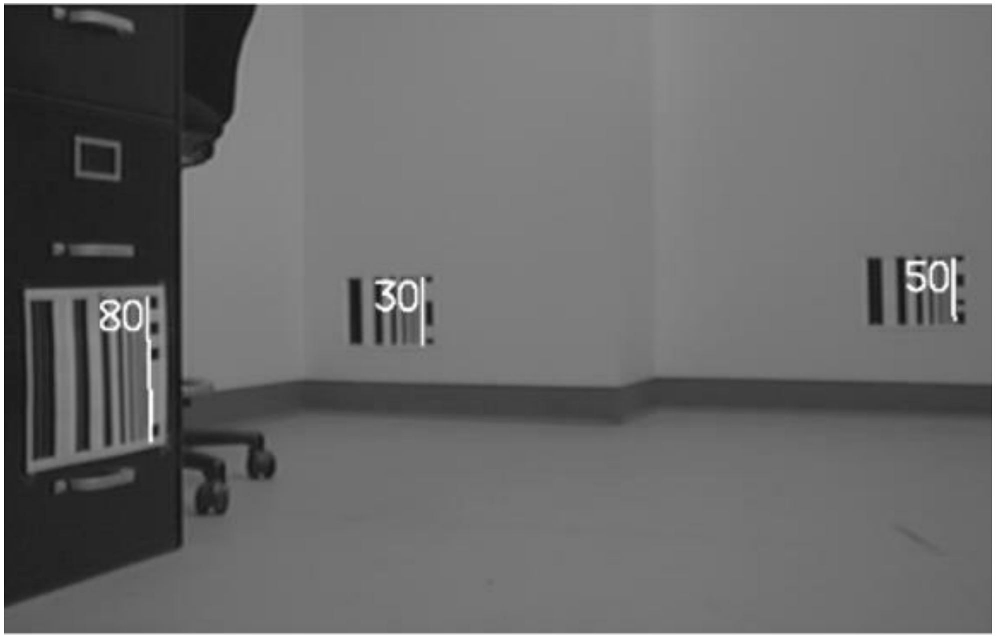


Unique color patterns

Landmark Recognition
Based on Camera Snapshots

Camera-based Artificial Landmark Recognition





Unique color patterns

Positioning by Landmarks

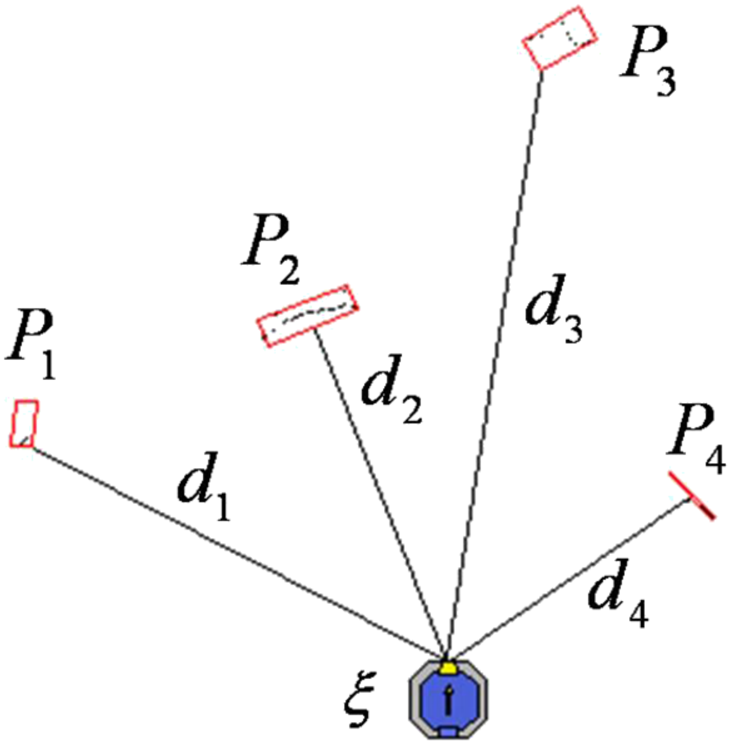
- In the following: determination of the robot’s position based on different captured data sets of landmarks
- Assumed: 2D-Map including all landmark positions

$$P_1, \dots, P_i,\ P_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix}$$

- Further designators
  - ξ: Position of the AMR with $\xi = (p_x \quad p_y)^T$
  - θ: Orientation of the AMR
  - d_i: Distance between AMR and landmark P_i
  - $\alpha_{i,j}$: “visible” angle between landmark P_i and P_j
  - $v_{i,j}$: Distance between landmark P_i and P_j

Landmarks with Known Distances

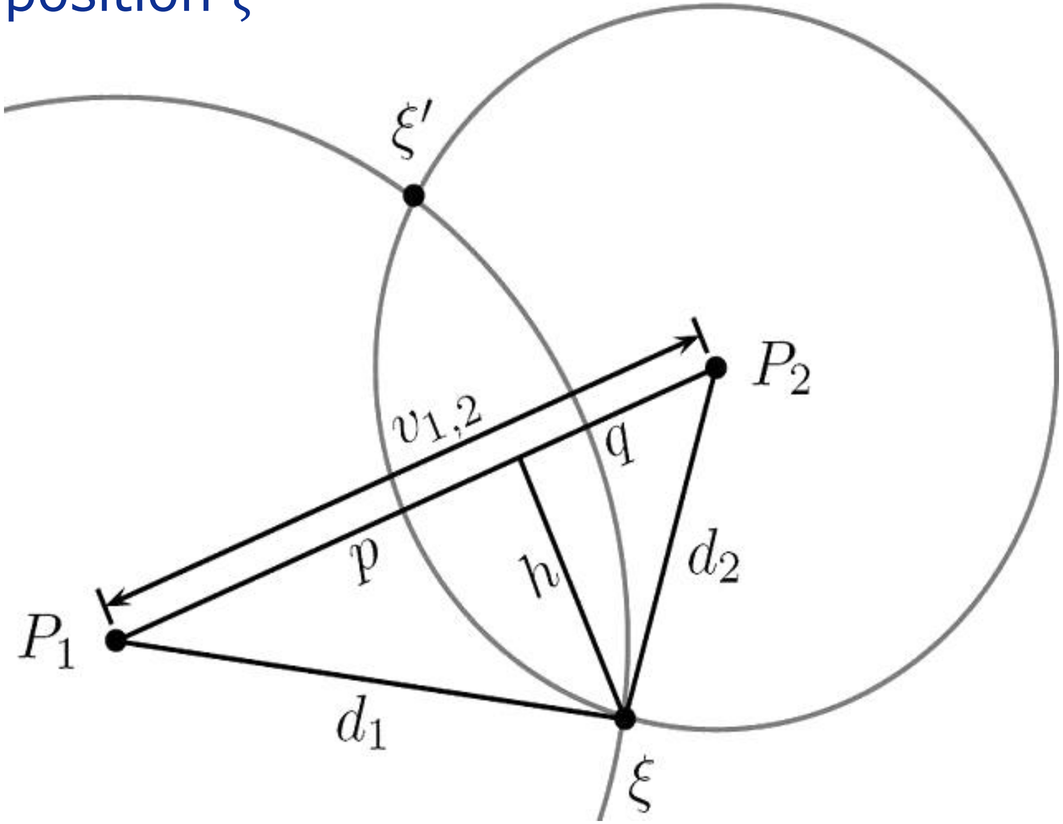
- Determination of distance to landmarks using e. g. laser scanners
- Determination of robot position ξ based on known d_i



Robot detecting multiple landmarks

Two Landmarks with Known Distances

- Given landmarks P_1, P_2 and their distances d_1, d_2 to the robot
- Wanted: position ξ



Two landmarks with known distance

Two Landmarks with known Distances

According to trigonometry:

$$p + q = v_{1,2}$$

$$p^2 + h^2 = d_1^2, q^2 + h^2 = d_2^2$$

$$\rightarrow h = \pm\sqrt{d_1^2 - p^2}$$

$$\rightarrow p = \frac{d_1^2 - d_2^2 + v_{1,2}^2}{2 \cdot v_{1,2}}$$

Two Landmarks with known Distances

- Given p and h $\xi = (p_x \quad p_y)^T$ can be calculated

$$p_x = x_1 + \frac{x_2 - x_1}{v_{1,2}} \cdot p + \frac{y_2 - y_1}{v_{1,2}} \cdot h$$

$$p_y = y_1 + \frac{x_2 - x_1}{v_{1,2}} \cdot h + \frac{y_2 - y_1}{v_{1,2}} \cdot p$$

- θ can not be calculated
- Results are two possible positions
- ξ' can be deduced from previous knowledge

Multiple Landmarks with known Distances

- Known distances to more than two landmarks allows simplified, linear calculation of ξ
- i known distances result in i conditions

$$
\begin{gathered}
d_0^2 = (x_0 - p_x)^2 + \left(y_0 - p_y\right)^2 \\
d_1^2 = (x_1 - p_x)^2 + \left(y_1 - p_y\right)^2 \\
\vdots \\
d_i^2 = (x_i - p_x)^2 + \left(y_i - p_y\right)^2
\end{gathered}
$$

- Subtraction results in $i - 1$ equations

$$
\begin{gathered}
d_1^2 - d_2^2 = x_1^2 - x_2^2 + 2p_x(x_2 - x_1) + y_1^2 - y_2^2 + 2p_y(y_2 - y_1) \\
\vdots \\
d_1^2 - d_i^2 = x_1^2 - x_i^2 + 2p_x(x_i - x_1) + y_1^2 - y_i^2 + 2p_y(y_i - y_1)
\end{gathered}
$$

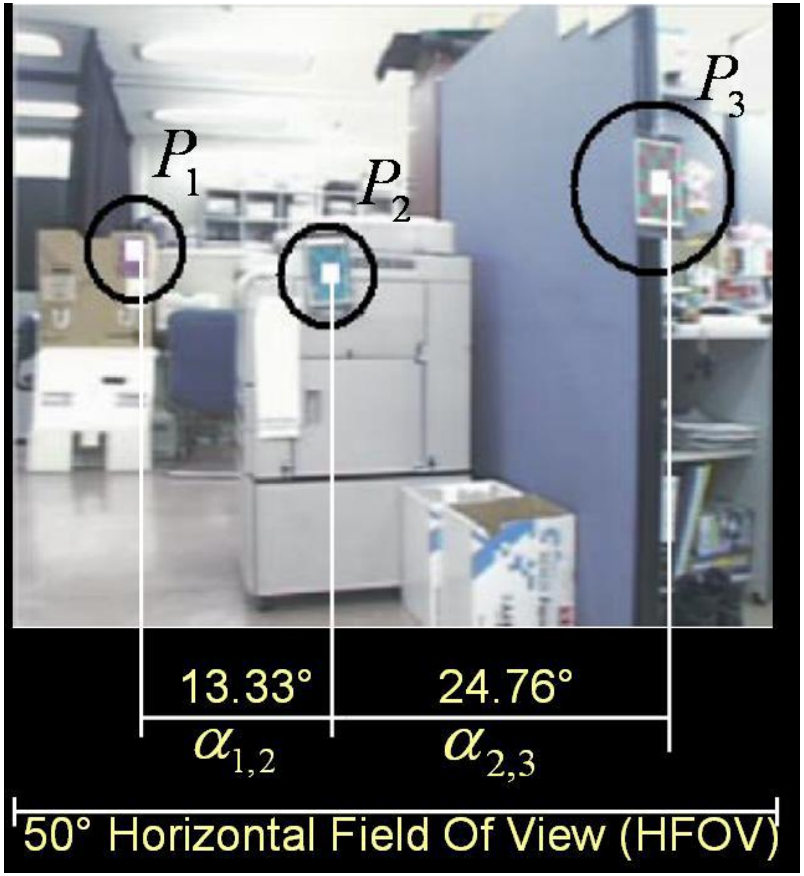
Multiple Landmarks with known Distances

- Equations can be transformed to $a_j p_x + b_j p_y = c_j$
  → System is linear and can be written as: $A\vec{x} = \vec{b}$ with

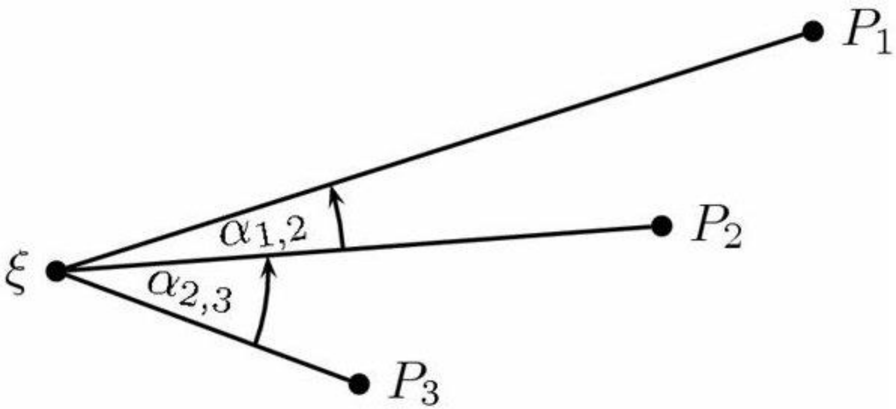
$$A = \begin{pmatrix} a_1 & b_1 \\ \vdots & \vdots \\ a_{i-1} & b_{i-1} \end{pmatrix}, \vec{x} = \begin{pmatrix} p_x \\ p_y \end{pmatrix}, \vec{b} = \begin{pmatrix} c_1 \\ \vdots \\ c_{i-1} \end{pmatrix}$$

- Solution is $\vec{x} = A^{-1} b^{\rightarrow}$, if $\mathrm{rank}(A) = 2$
- Due to noisy sensor signal often $\mathrm{rank}(A) > 2$
- Determination of $\vec{x}'$ using least square fitting:
  $\min_{\vec{x}'} \left(\vec{b} - A\vec{x}'\right)^T \left(\vec{b} - A\vec{x}'\right)$ via singular-value decomposition of A (→ linear algebra)

Landmarks with known Angles



Recognition of relative angles $\alpha_{i,j}$ between landmarks



Schematic

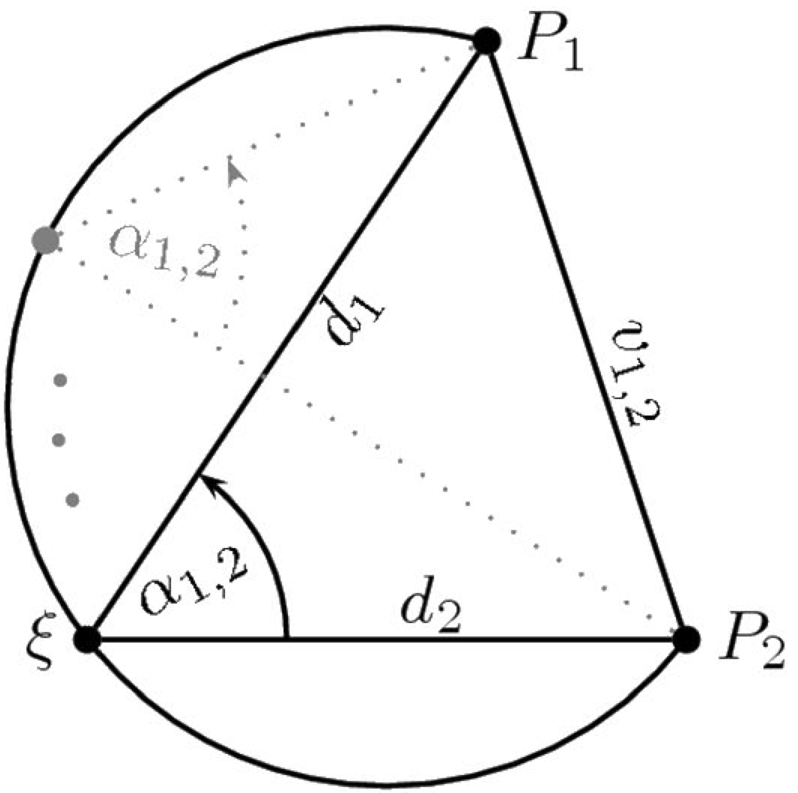
Two Landmarks with known Angle

- A known angle between two landmarks narrows robot position down to a circular arc

- Possible values ξ fulfill law of cosine

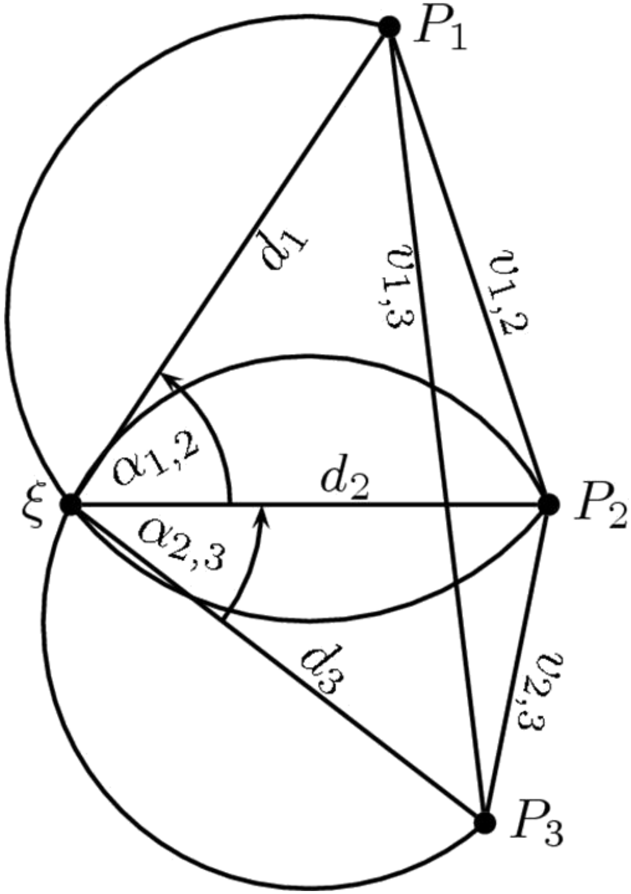
$$v_{1,2}^2 = d_1^2 + d_2^2 - 2|d_1||d_2| \cos \alpha_{1,2}$$

- Problems
  - ξ is ambiguous
  - Unknown distances d_i



Two Landmarks with Known Angle

Three Landmarks with known Angles



Three landmarks P_1 through P_3 with two known angles

Three Landmarks with known Angles

- Addition of third landmark results in three conditions

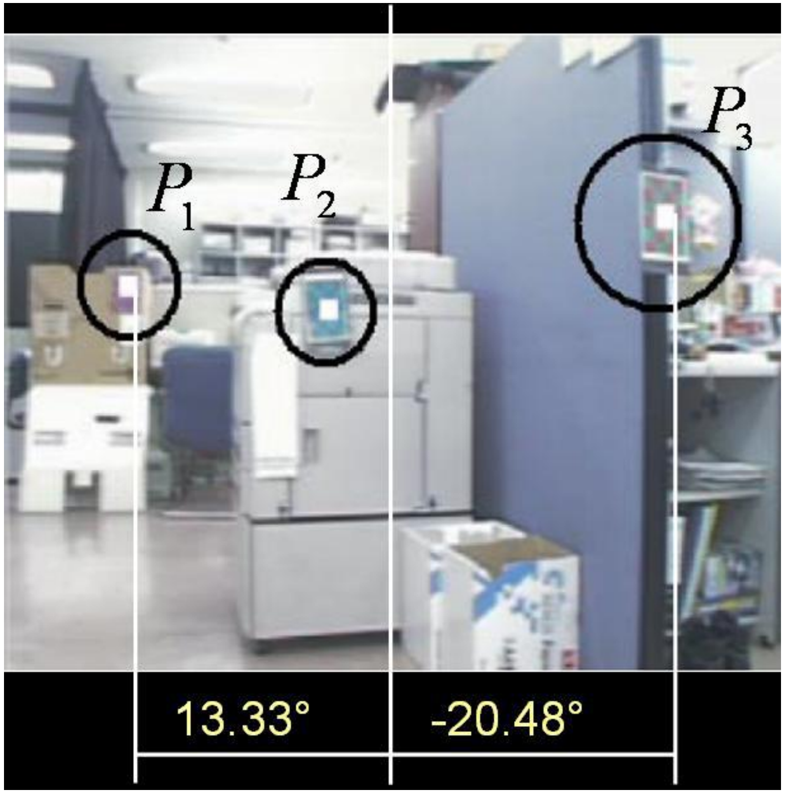
$$v_{1,2}^2 = d_1^2 + d_2^2 - 2|d_1||d_2|\cos\alpha_{1,2}$$

$$v_{2,3}^2 = d_2^2 + d_3^2 - 2|d_2||d_3|\cos\alpha_{2,3}$$

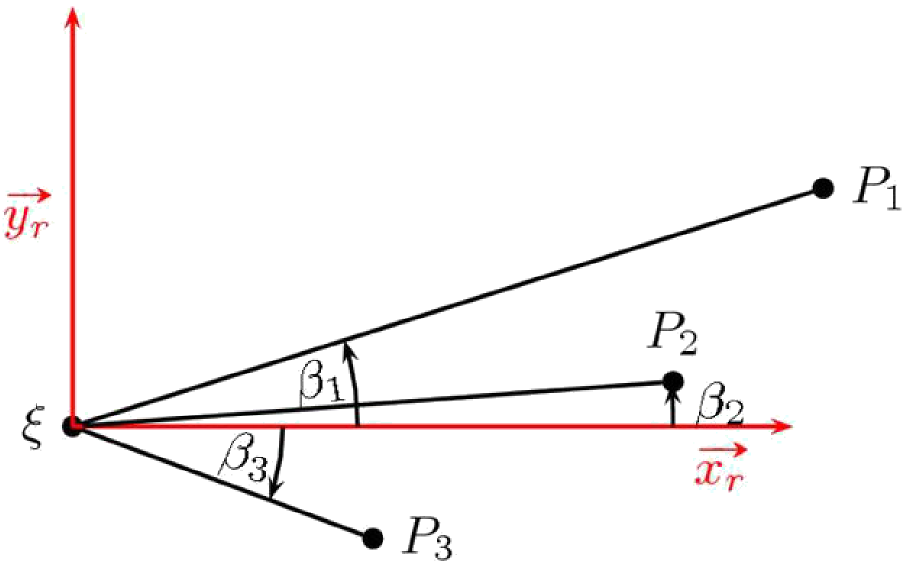
$$v_{1,3}^2 = d_1^2 + d_3^2 - 2|d_1||d_3|\cos(\alpha_{1,2} + \alpha_{2,3})$$

- Non-linear equation system
  - Iterative approximation of d_i via e. g. Newton's method or Levenberg-Marquardt
- Reduction to problem with known distances

Determination of Orientation



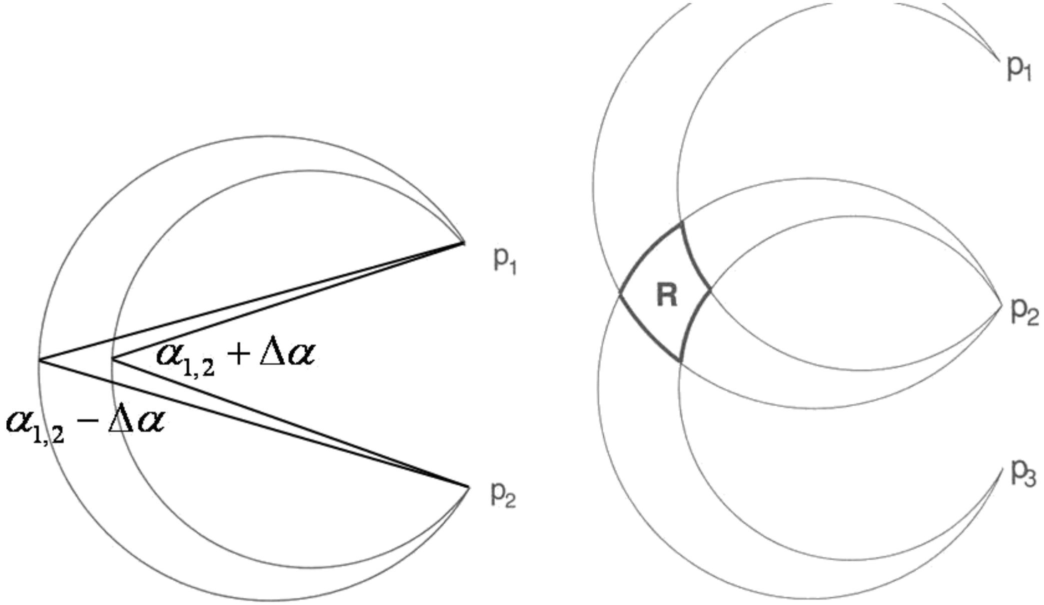
Measurement of angles β_i between x-axis of the robot's coordinates-systems and the landmarks



Schematic

Determination of Orientation

- Error in angle measurement $\Delta\alpha$ results in "crescent-shaped" area, the robot is located in
- Quantitative judgment of positional error due to imprecise sensor data can be performed using covariance analysis



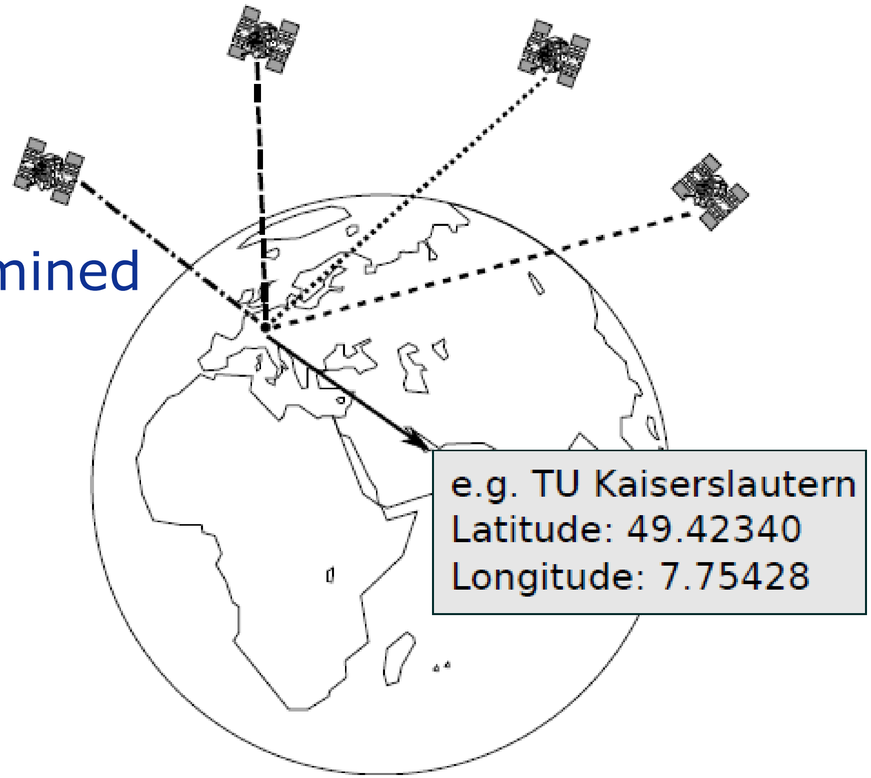
Determination of orientation

Natural Indoor Landmarks

- Corners as points of reference
  - Door edges
  - Wall edges
  - Edges of fixed furniture
- Corners orientation is used to determine robot orientation
  - Transition: wall — floor
  - Neon tubes on the ceiling
  - Ceiling structures
  - Walls
- Static objects as landmarks

Global Navigation Satellite System (GNSS)

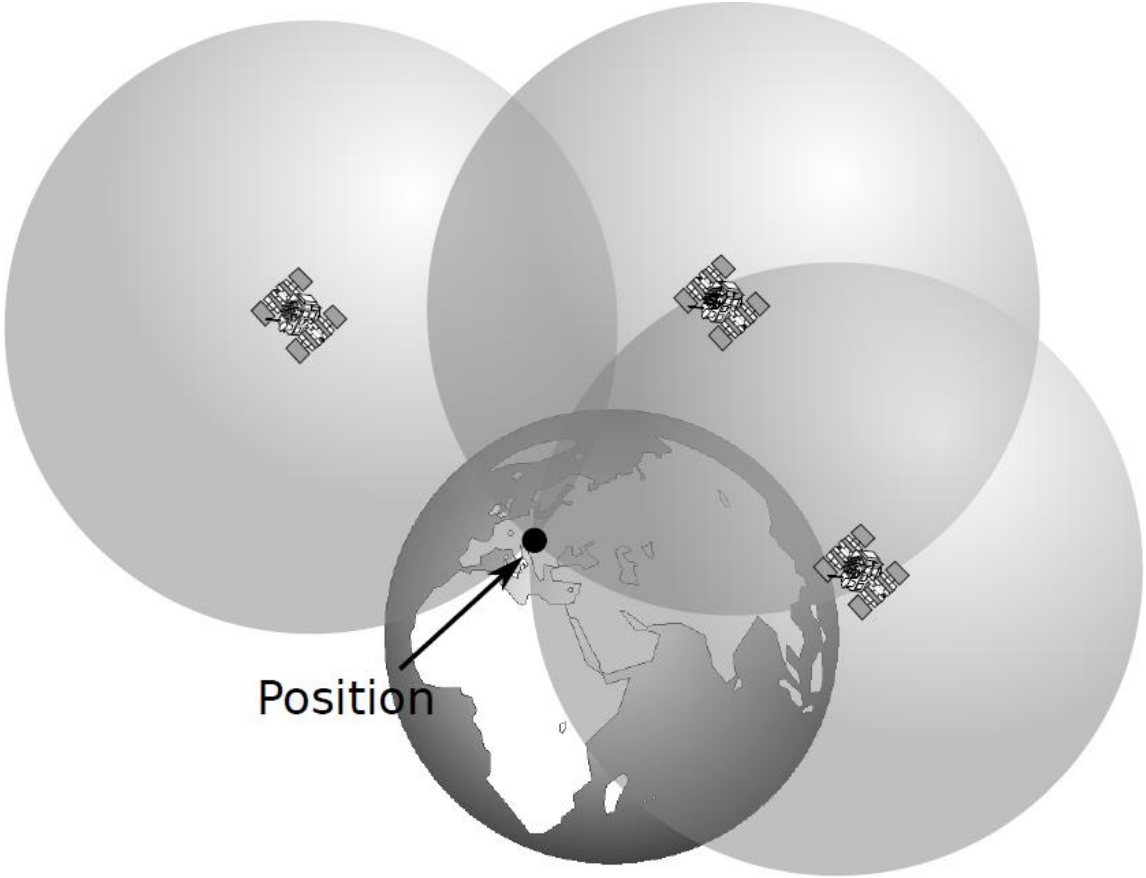
- Satellite-based positioning system
  - Latitude $\varphi \in [-90°, 90°]$
  - Longitude $\lambda \in [-180°, 180°]$
  - Altitude
  - Time
- Other values can also be determined
  - Speed
  - Acceleration
  - Course
  - Local time
  - Range measurements
- Worldwide coverage
- Precision: civil: $5 - 10\ \mathrm{m}$, military: $2 - 9\ \mathrm{m}$



Functional Principle

- Satellites send messages containing transmission time and their exact locations
- GPS receiver calculates location through signal propagation time Δt_i and satellite location (x_i, y_i, z_i)
- To determine the position (x, y, z)
  - At least 3 satellites are required for a time synchronized receiver
    - Geometric interpretation: intersection of 3 spheres
  - At least 4 satellites are needed, if the receiver is not synchronized with the satellite system
    - Normal case
    - Time error Δt_0 between receiver's clock and the GPS-time
    - Reason: atomic clocks are too large, too expensive and maintenance-intensive

Geometric interpretation



The position is determined at the point where all three spheres intersect

Asynchronous Position Estimation

- The position (x, y, z) and time-shift Δt_0 can be calculated by solving the following system of equations

$$(\Delta t_1 + \Delta t_0) \cdot c = \sqrt{(x_1 - x)^2 + (y_1 - y)^2 + (z_1 - z)^2}$$

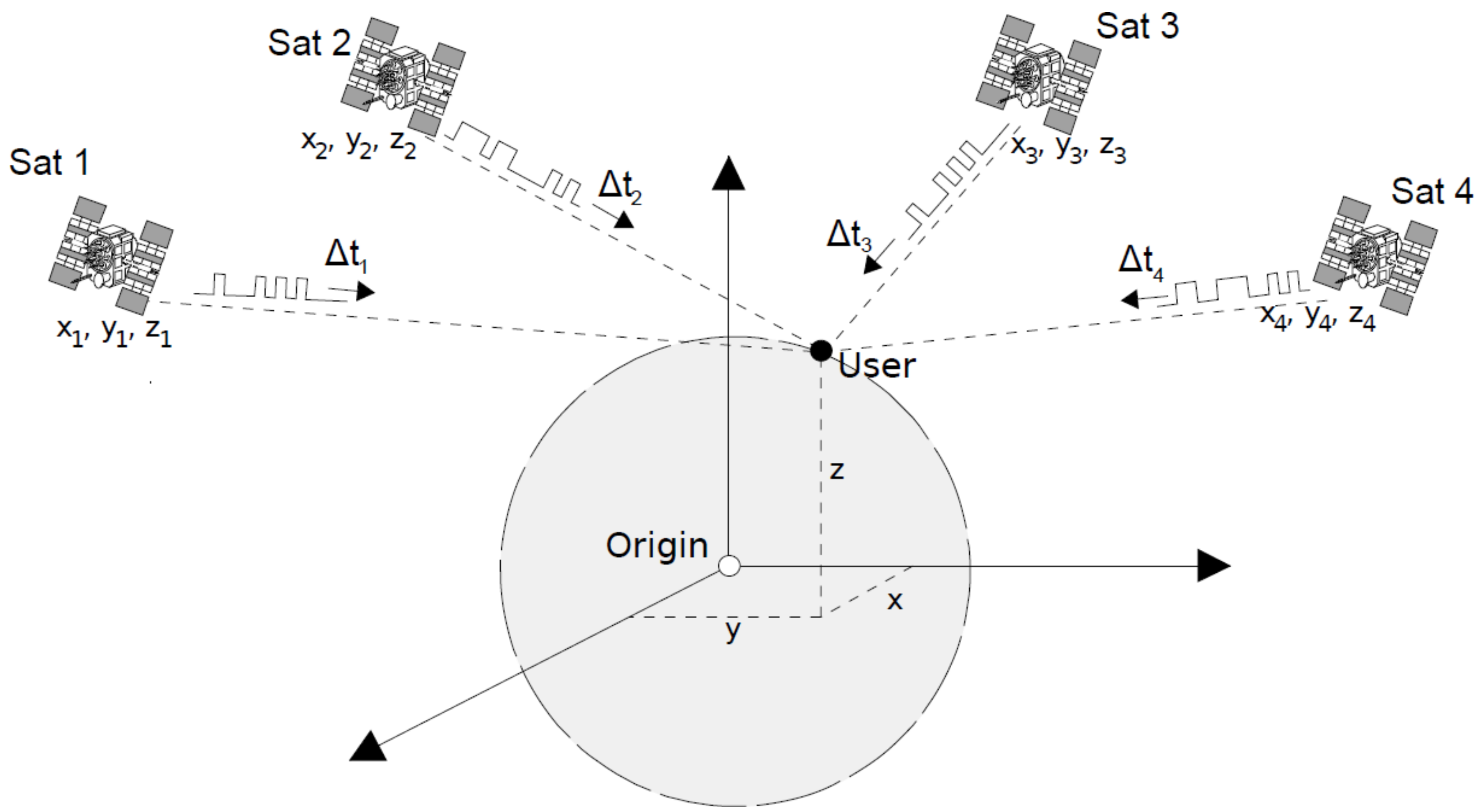
$$(\Delta t_2 + \Delta t_0) \cdot c = \sqrt{(x_2 - x)^2 + (y_2 - y)^2 + (z_2 - z)^2}$$

$$(\Delta t_3 + \Delta t_0) \cdot c = \sqrt{(x_3 - x)^2 + (y_3 - y)^2 + (z_3 - z)^2}$$

$$(\Delta t_4 + \Delta t_0) \cdot c = \sqrt{(x_4 - x)^2 + (y_4 - y)^2 + (z_4 - z)^2}$$

- c is the speed of light
- $(\Delta t_i + \Delta t_0)$ is the measured, error-prone transmit time

Asynchronous Position Estimation



Position determination with four satellites

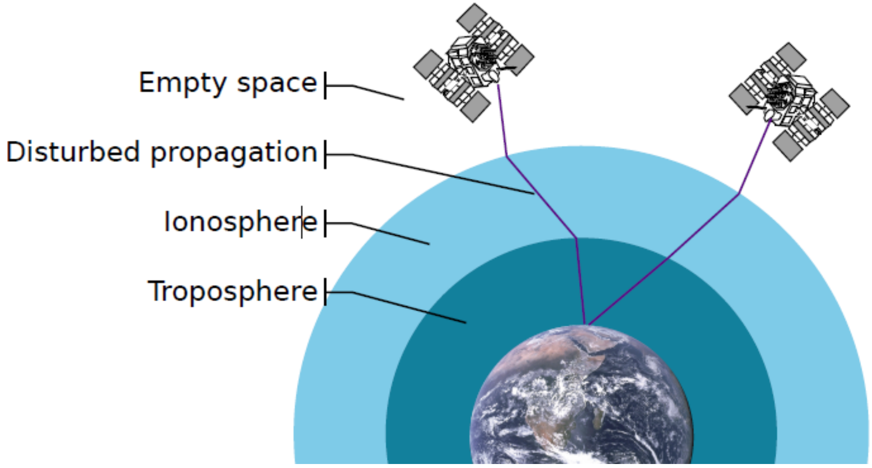
Sources of Measurement Errors

- Ionospheric error ($3.0\ \mathrm{m}$), through reduced speed of signal propagation (refraction), improvements through dual frequency receiver or correction model

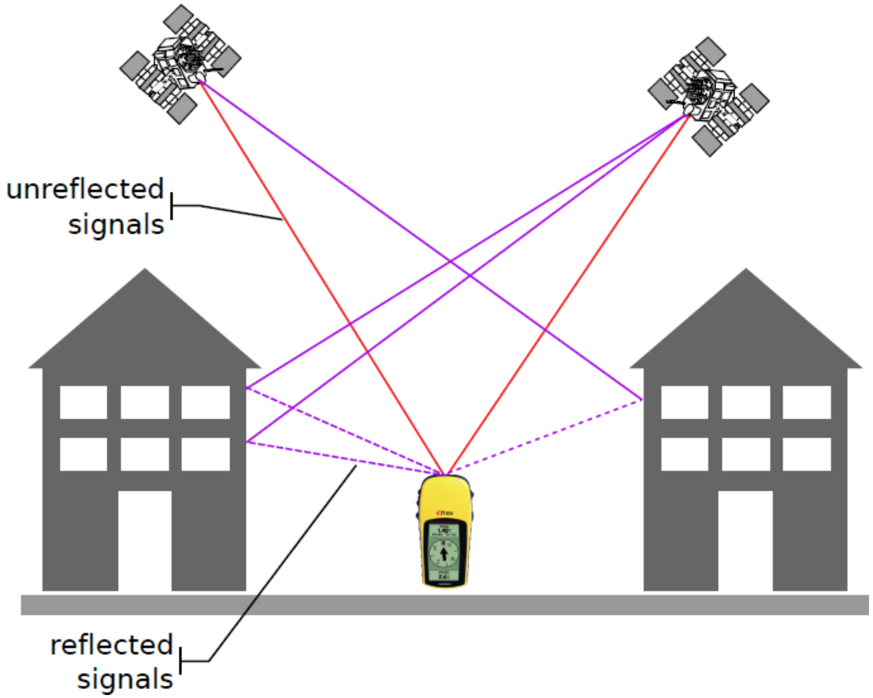
- Tropospheric error ($0.7\ \mathrm{m}$), see ionospheric error

- Multipath error ($1.0\ \mathrm{m}$), caused by terrestrial reflections, improvements through dual frequency usage, directional antennas or special signal processing algorithms

Sources of Measurement Errors



Refraction error



Multipath error

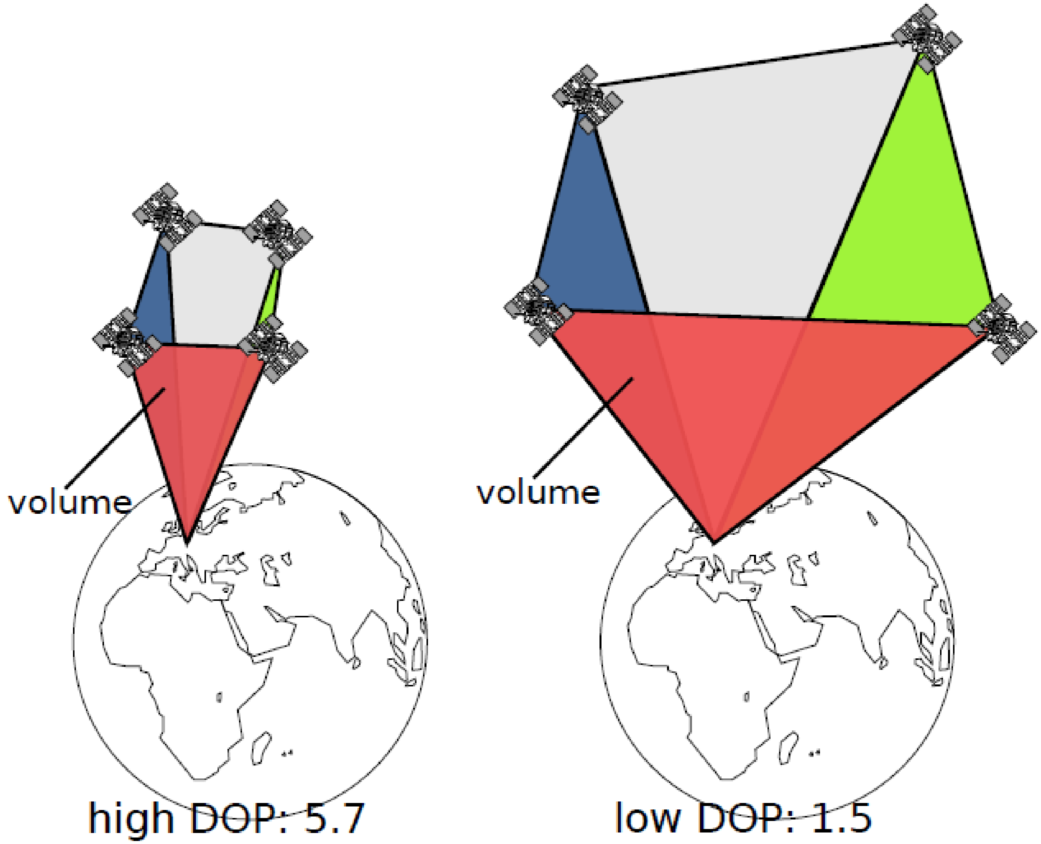
Sources of Measurement Errors

- Satellite clock error ($1.5\,\mathrm{m}$), improvements through frequent synchronization and clock correction
- Orbit error ($1.5\,\mathrm{m}$), through deviations in the satellite orbit, compensation through frequent adjustment of the satellite orbit parameters (almanac and ephemeris) transmitted with the satellite messages
- Receiver error ($0.5\,\mathrm{m}$), caused by clock limitations of the receiver
- Geometric effects (amplifies all other errors), through satellite constellation, known as “Dilution of Precision” (DOP)

Dilution of Precision

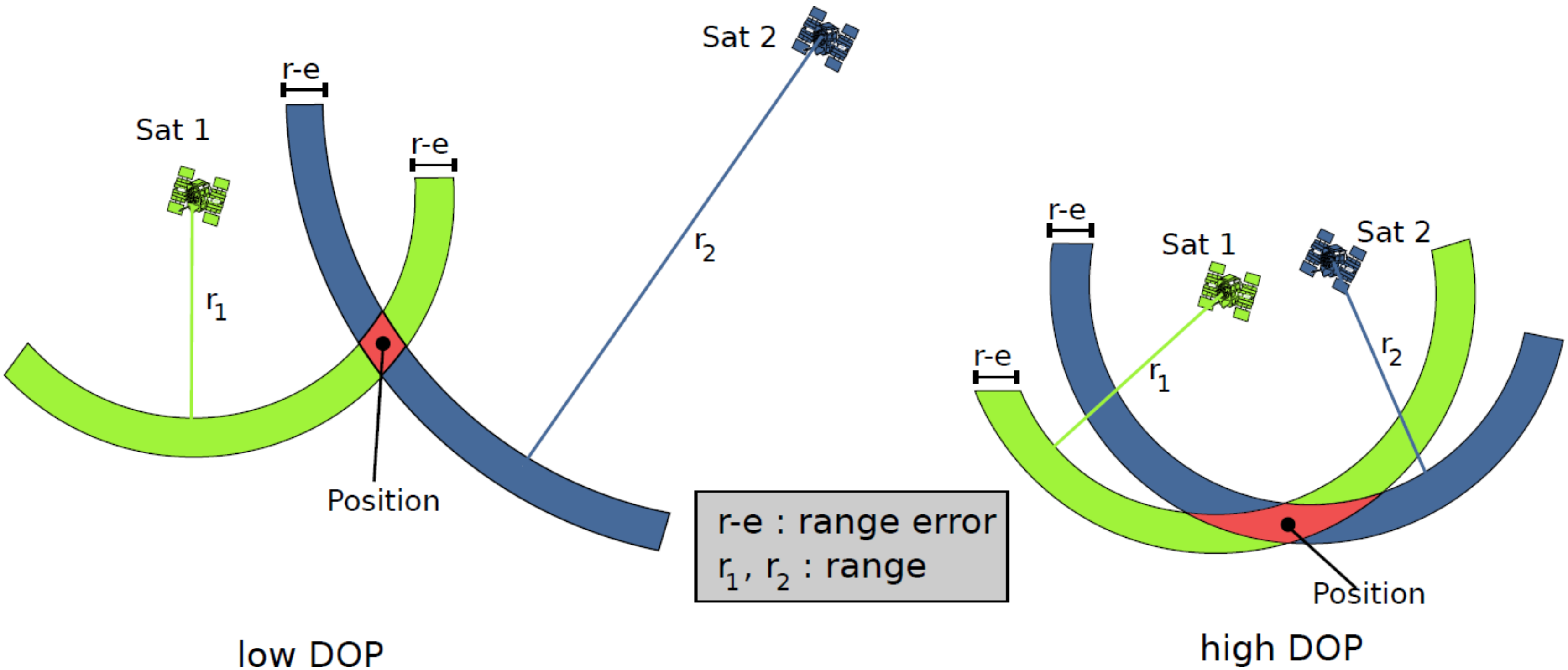
- Precision of position determination depends on the satellite geometry/the position of the visible satellites, expressed in different DOP-values like …
  - Horizontal DOP (2D-coordinates)
  - Vertical DOP (altitude)
- DOP-value can be interpreted as the reciprocal volume of a tetrahedron built by the four satellites and the receiver
- Biggest inaccuracies if all satellites are located close by each other or are in collinear alignment

Dilution of Precision



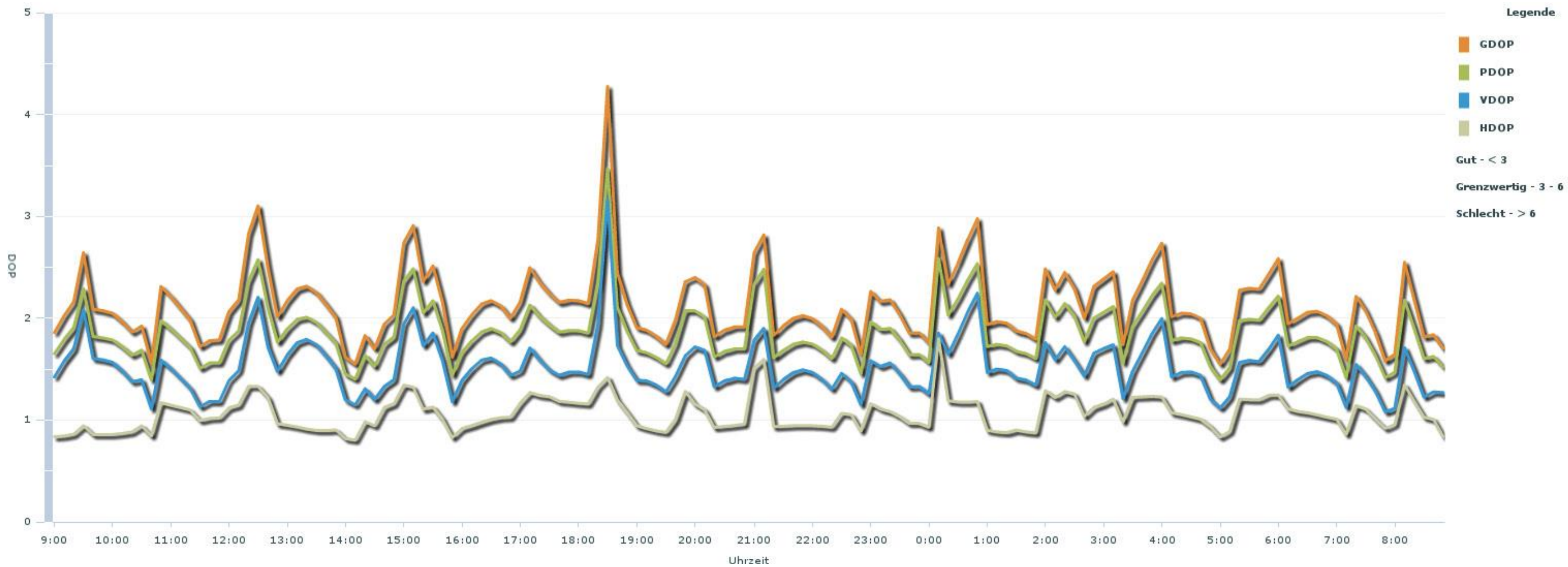
The larger the enclosed volume, the smaller the DOP value

Dilution of Precision



The flatter the angle with which the circles with ranges r_1 and r_2 intersect, the higher the DOP value

Dilution of Precision

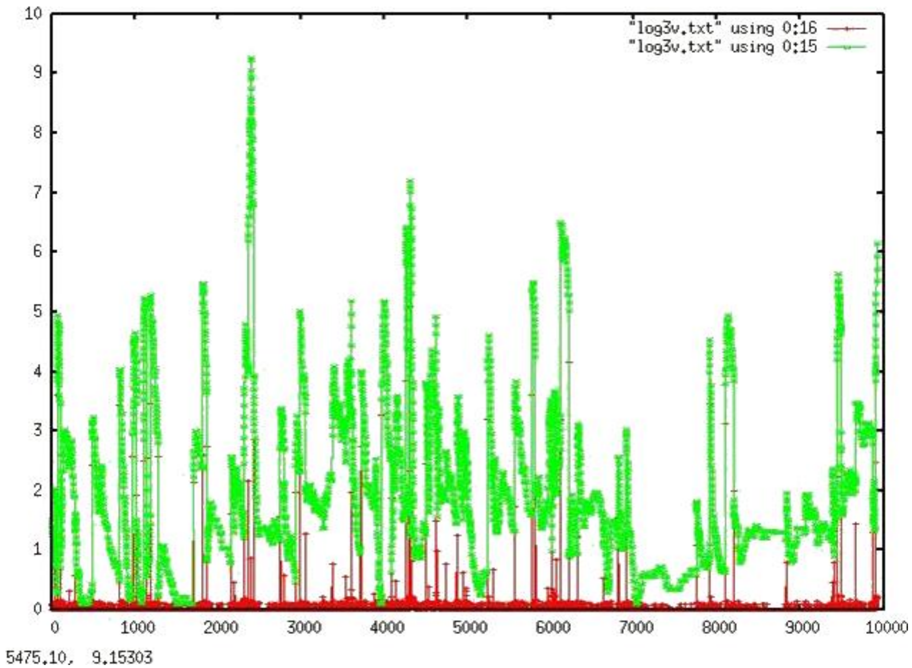


DOP prediction for TU Kaiserslautern (GPS)

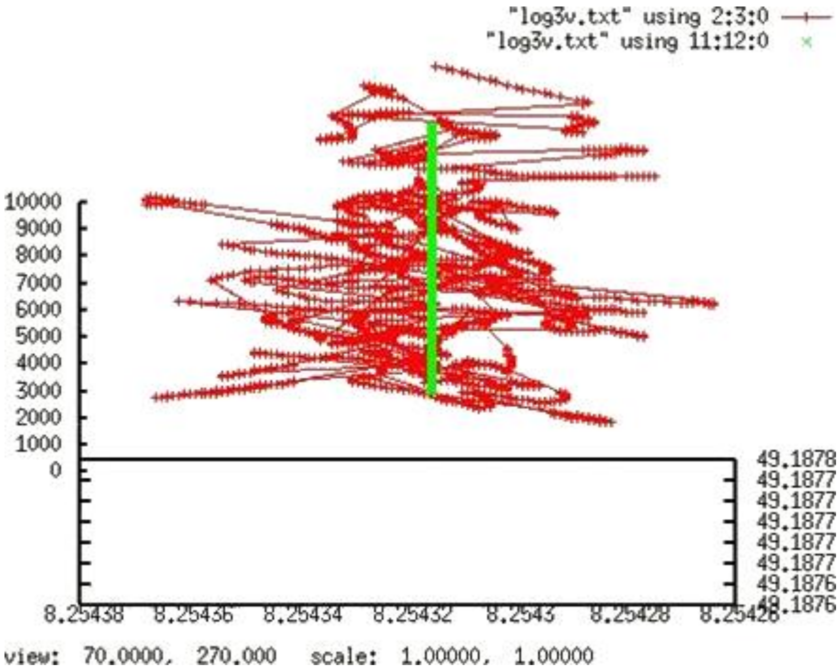
$\varphi = 49.422902, \;\; \lambda = 7.753574$

Date: Thu May 12 2011

Local Experiment: 6 Hours of Measurement

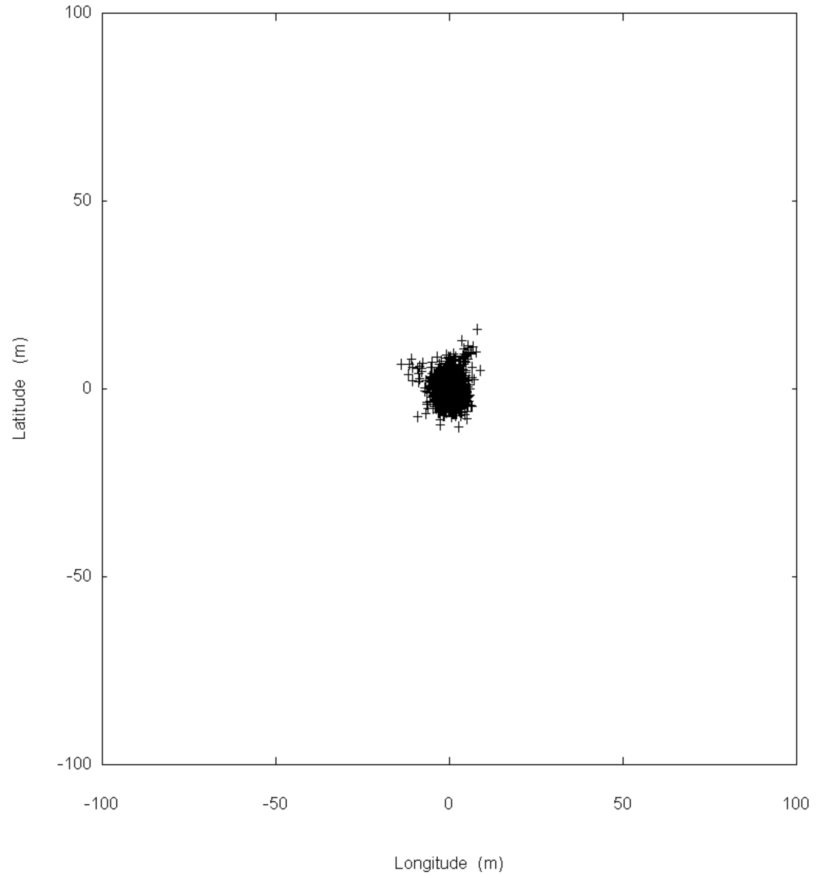


Deviation from actual position in meters



Deviation over time

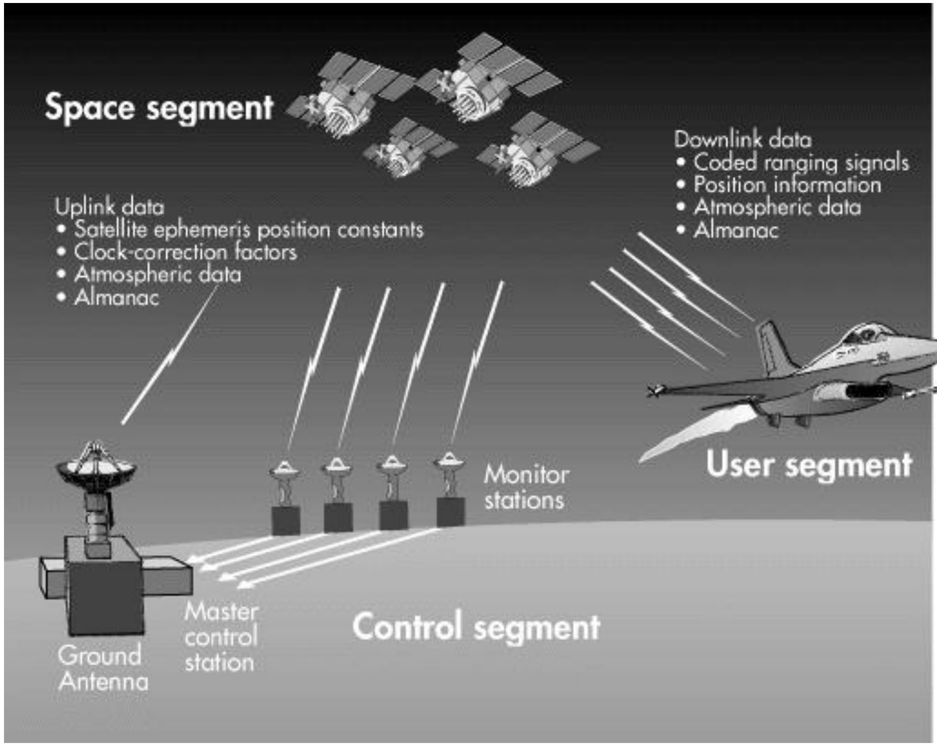
Reference: Errors during 24 Hours of Measurement



The positional scatter of 24 hours of data (0000 to 2359 UTC) taken at one of the Continuously Operating Reference Stations (CORS) operated by the NCAD Corp. at Erlanger, Kentucky

System components

- Space segment
- Control segment
  - Master control and several worldwide monitoring stations
  - Uplink to satellites for clock, orbit and almanac updates
- User segment



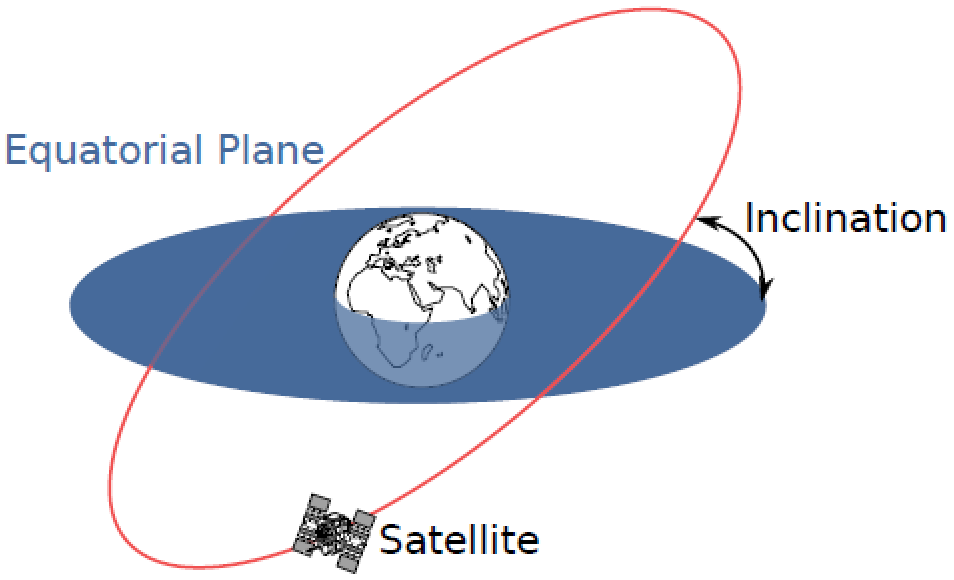
GNSS segments

Available Systems

- Global Positioning System (GPS), USA, full-functional
- GLONASS, Russia, rebuilding/functional
- Galileo, EU, build-up phase
- Compass, China, build-up phase
- Several regional systems
  - Beidou1 (China)
  - IRNSS (India)
  - DORIS (France)
  - QZSS (Japan)

Global Positioning System (GPS)

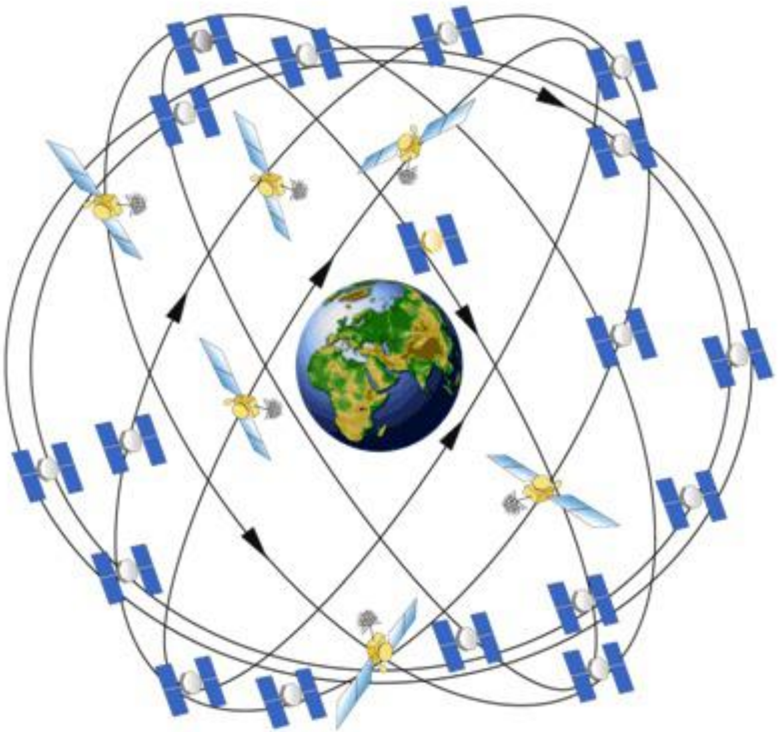
- At least 24 active satellites, 31 currently in orbit and healthy (05/2011)
- 3 additional active planed for 2011 to improve positioning
- Orbit height $\sim 20200\ \mathrm{km}$
- Distributed on 6 nearly circular orbital planes with $55\,°$ inclination with reference to the equatorial plane
- Orbital period: $11\ \mathrm{h}\ 58\ \mathrm{min}$, $12\ \mathrm{h}$ sidereal time



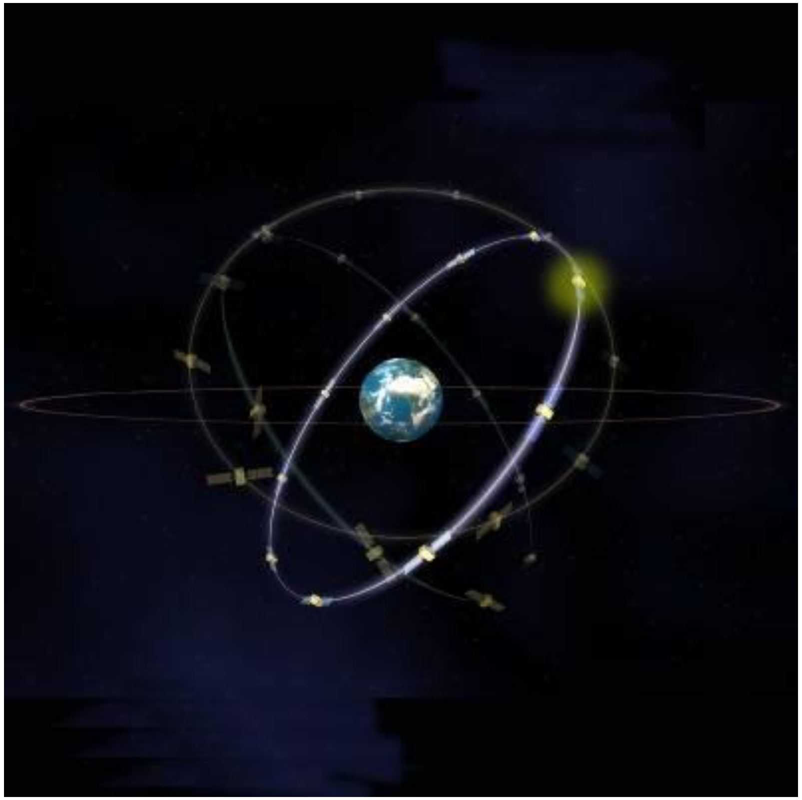
Global Positioning System (GPS)

- Two frequencies
  - $L1$: $1575.42\ \mathrm{MHz}$
  - $L2$: $1227.6\ \mathrm{MHz}$
  - Additional frequencies planed
- $L1$ carries C/A-Code (clear access/coarse acquisition) for civilian usage
- $L2$ carries P-code (protected/precise) for military usage

Constellation GPS/GALILEO



GPS constellation



GALILEO constellation

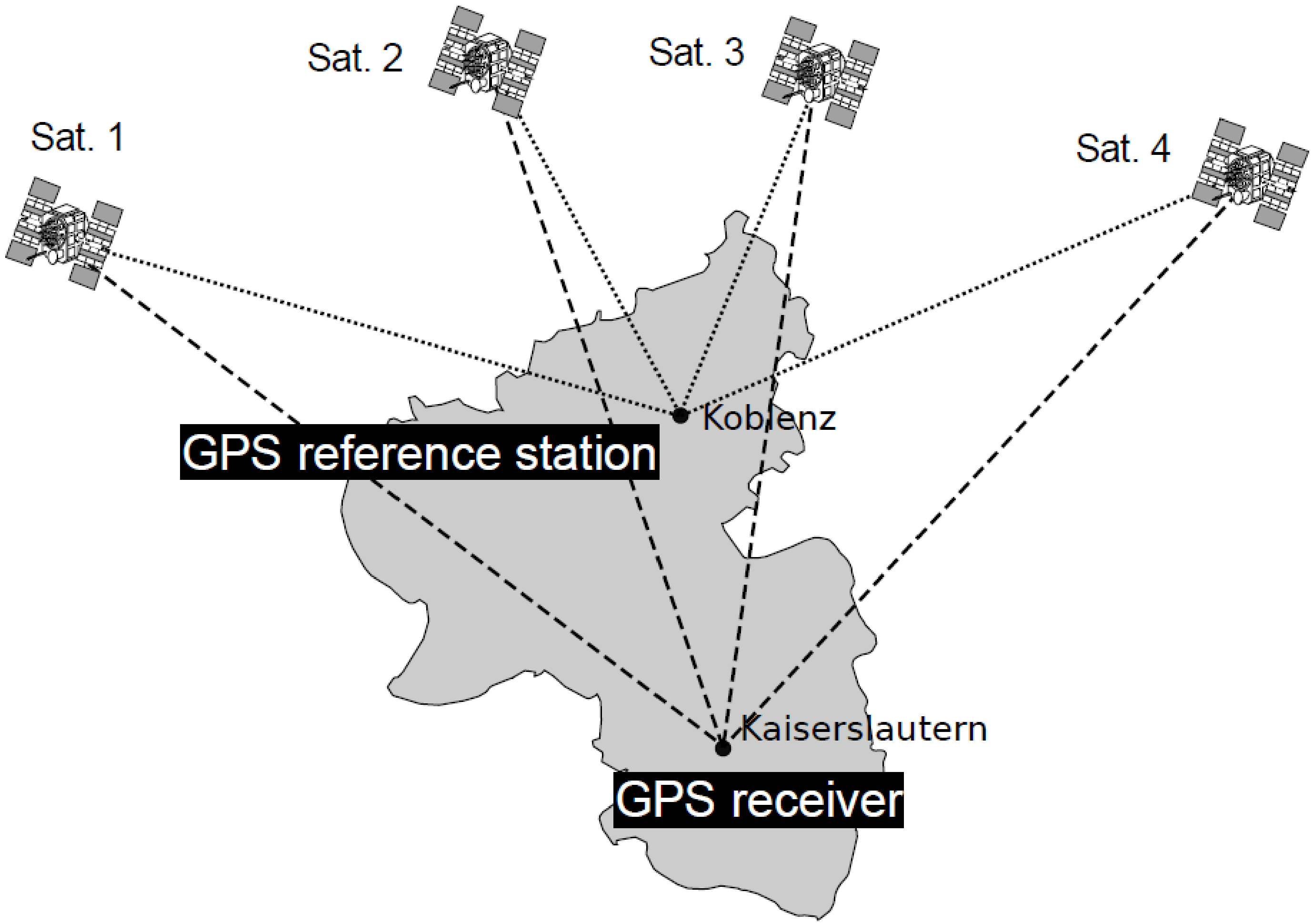
GALILEO

- 30 satellites ($27 + 3$ backup)
- 3 orbits, $56\,°$ inclination, $23222\,\mathrm{km}$ above the Earth, $14\,\mathrm{h}$ to orbit the Earth
- Improvements
  - Under civilian control (EU and many other countries, independent from USA!?)
  - Greater inclination → better coverage at high latitudes, esp. Europe (which is not well covered by GPS)
  - Receivers will be GPS compatible → more accurate positioning, more visible satellites
  - 2 frequencies: reduction of ionospheric and multipath error
  - Improvements in the used signal coding → reduces noise at the receiver
  - Horizontal precision: Open Service: $\sim 4\,\mathrm{m}$,
Commercial Service: up to $10\,\mathrm{cm}$

Differential GPS (DGPS)

- Reduces error sources like local atmospheric disturbances, satellite orbit and clock deviations
- Function principle (Real Time Kinematics)
  - Reference stations with exactly known coordinates determine their GPS positions using the visible satellites
  - The position deviation between the known and the measured position is calculated as well as the deviations of the expected propagation times between the station and the satellites
  - The deviations are transmitted to compatible and licensed receivers via radio, cellular networks or using geostationary satellites (e.g. OmniSTAR, StarFire)
- The correction data can be used up to 200km around the reference stations

Differential GPS (DGPS)

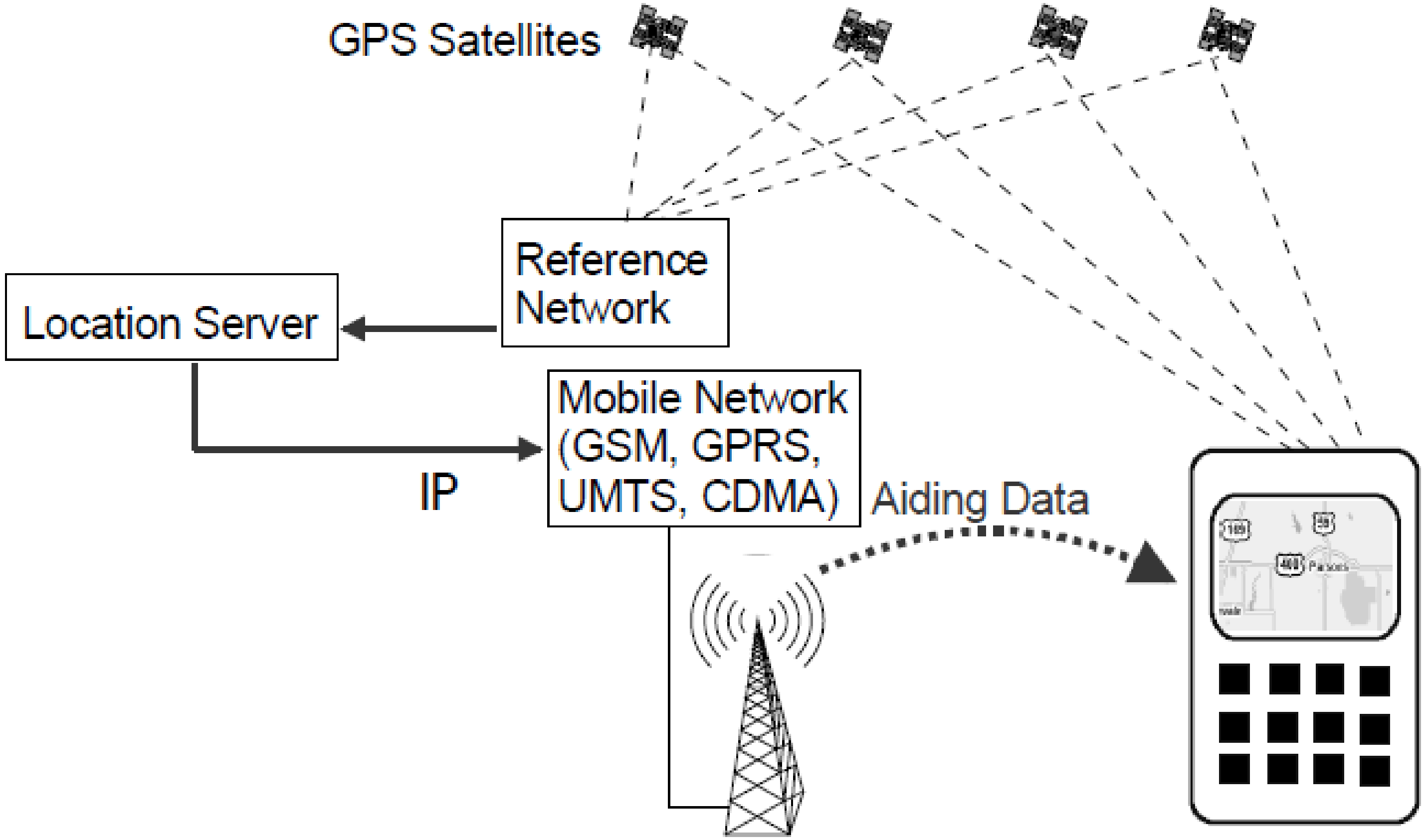


DGPS Functional principle

Assisted GPS (A-GPS)

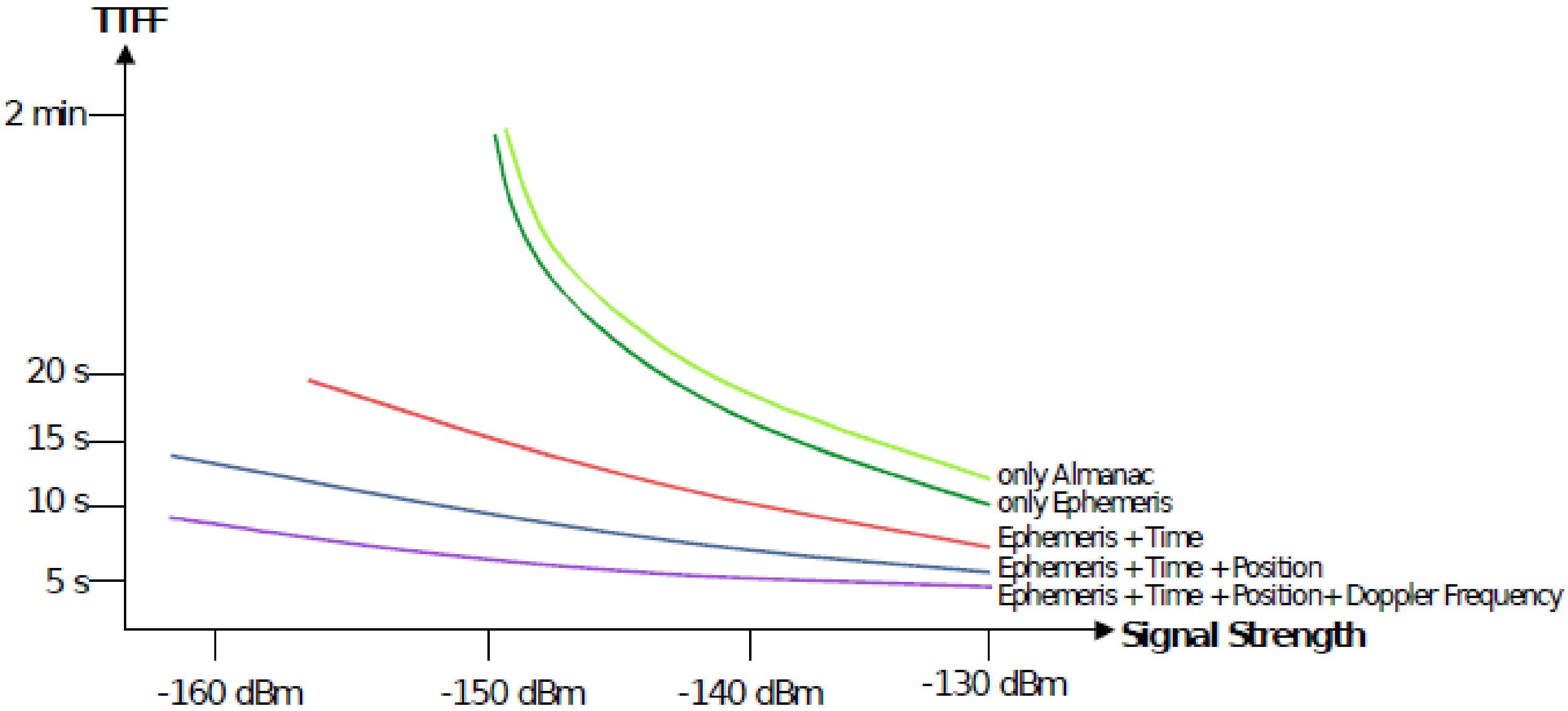
- Aiding data (almanac, ephemeris, time) is made available through other communication channels like GSM, GPRS, UMTS
- Reduces start-up time (Time to First Fix)/time to obtain orbital data (else orbital data has to be received from the satellites which can require minutes due to low bandwidth)
- Enables positioning under poor signal conditions

Assisted GPS (A-GPS)



A-GPS functional principle

Assisted GPS (A-GPS)



Time to First Fix with different aiding data

Coming Next

Kalman Filter