Time of Arrival (TOR)
FPs on the centers of the spheres
Assumptions

- for simplicity 2D-scenario, extension to 3D-scenario straightforward
- K fixed points (FPs) at known positions \((x_k, y_k), k = 0, \ldots, K-1\), e.g., satellites in GPS
- without loss of generality: choose coordinates such that FP0 is at the origin, i.e., \((x_0, y_0) = (0, 0)\)
- measure ranges \(r_k, k = 0, \ldots, K-1\)
  (equivalent to delays, "Time of Arrival" (TOA))
- requires synchronized clocks at \(Tx\) and \(Rx\)
- task 1: determine the position \((x, y)\)
  of the mobile terminal (MT)
- here: point positioning (single set of measurements)
- later: tracking (sequence of measurements → trajectory)
Time of Arrival (TOA)

- Each measured range
  \[ r_n = \sqrt{(x-x_n)^2 + (y-y_n)^2} \]

defines a sphere of possible NT positions

\[ \Rightarrow \text{find the intersection point of the spheres; various algorithms to solve this nonlinear (in general overdetermined) system of equations} \]

- In 2D scenarios, 2 unknowns x and y
  - 2 FPs \( \Rightarrow \) 2 equations, but in general two intersection points; ambiguity might be resolved by some side information
  - 3 FPs \( \Rightarrow \) one unique intersection point (if no measurement errors)
Measurement Error

• Typically no common intersection point of all spheres
  ⇒ pseudo solution required

• Localization error (error propagation) depends on FP constellation
  ⇒ design of good constellations

good constellation

bad constellation
Analytical Method

required K=2 FPs w/ 2D scenario

1. \( r_0^2 = x_0^2 + y_0^2 \) \( (x_0=0, \ y_0=0) \)

2. \( r_1^2 = (x-x_1)^2 + (y-y_1)^2 \)
   \[= x_1^2 - 2x_1x_1 + x_1^2 + y_1^2 - 2y_1y_1 + y_1^2 \]

3. \( x = \frac{-y_1}{x_1} \frac{r_0^2 - r_1^2 - x_1^2 - y_1^2}{2x_1} \)

Substitute this into 6 → quadratic equation for y

\[(A^2 + 1) y^2 + 2AB y + B^2 - r_0^2 = 0 \]

\[y = -\frac{AB}{A^2 + 1} \pm \sqrt{(\frac{AB}{A^2 + 1})^2 - \frac{B^2 - r_0^2}{A^2 + 1}} \]

→ general two solutions (as expected)

using 3 \( x = A y + B \)

Remark: 3 defines a straight line of possible FP positions
Least Squares Method

- requires K ≥ 3 FP's in 2D scenarios
- similar to the analytical method
- K-2 linear equations for the two variables x and y can be obtained:

\[ r_{0} = x^2 + y^2 \quad (x_0 = 0, y_0 = 0) \]

\[ r_{n} = (x-x_n)^2 + (y-y_n)^2 = x^2 - 2x x_n + x_n^2 + y^2 - 2y y_n + y_n^2 \]

\[ r_0^2 - r_n^2 = -2x x_n - 2y y_n + x_n^2 + y_n^2 \]

- set up an (over-) determined linear system of equations

\[
\begin{bmatrix}
  x_1^2 + y_1^2 - r_1^2 - r_0^2 \\
  \vdots \\
  x_K^2 + y_K^2 - r_K^2 - r_0^2
\end{bmatrix}
= \begin{bmatrix}
  x_1^2 + y_1^2 \\
  \vdots \\
  x_K^2 + y_K^2
\end{bmatrix}
\]

- Remark: Each of the linear equations \( 1-0 \) define a straight line of possible \( n-1 \) positions

\( 1-0 \) - \( 2-0 \) - \( 3-0 \)
Least Squares Pseudosolution of Overdetermined Linear Systems of Equations

\[ A \cdot x = B \]

Minimise the error \( \| A \cdot x - B \| ^2 \)

The error becomes minimum if the error vector \((A \cdot x - B)\) is orthogonal to the subspace spanned by the columns of \(A\)

\[ A^T \cdot (A \cdot x - B) = 0 \]

\[ A^T \cdot A \cdot x - A^T \cdot B = 0 \]

\[ A^T \cdot A \cdot x = A^T \cdot B \]

\[ x = (A^T \cdot A)^{-1} \cdot A^T \cdot B \]
Newton-Raphson Method

- Idea: Linearize the range equation around an initial estimate

\[
(x(n), y(n))
\]

\[
x_n = \sqrt{(x-x_n)^2 + (y-y_n)^2}
\]

\[
X \sqrt{(x(n)-x(n))^2 + (y(n)-y(n))^2}
\]

\[
x_{n+1} = x_n + \Delta x(n)
\]

\[
y_{n+1} = y_n + \Delta y(n)
\]

\[
x_{n+1} = x_n + \frac{x(n)-x_n}{r_n(n)} \Delta x(n) + \frac{y(n)-y_n}{r_n(n)} \Delta y(n)
\]

\[
x_{n+1} = x_n + \left[\begin{array}{c}
\frac{x(n)-x_n}{r_n(n)} \\
\frac{y(n)-y_n}{r_n(n)}
\end{array}\right] \left[\begin{array}{c}
\Delta x(n) \\
\Delta y(n)
\end{array}\right]
\]

Unit vector pointing from FPk towards initial estimate at NI position projection

\[
\Delta(n)
\]
set up an (over-) determined linear system of equations for computing the correction \((ax, ay)_1\) requires \(K \geq 2\) FP's in 2D scenarios

\[
\begin{pmatrix}
\frac{x(n) - x_0}{r_0(n)} & \frac{y(n) - y_0}{r_0(n)} \\
\vdots & \vdots \\
\frac{x(n) - x_{K-1}}{r_{K-1}(n)} & \frac{y(n) - y_{K-1}}{r_{K-1}(n)}
\end{pmatrix} \cdot 
\begin{pmatrix}
\Delta x(n) \\
\Delta y(n)
\end{pmatrix} = 
\begin{pmatrix}
\frac{x_0 - x_0(n)}{r_0(n)} \\
\vdots \\
\frac{x_{K-1} - x_{K-1}(n)}{r_{K-1}(n)}
\end{pmatrix}
\]

\[R(n) \cdot \Delta(n) = B(n)\]

\[\text{least squares pseudo solution}\]

\[\Delta(n) = \left( R^T(n) \cdot R(n) \right)^{-1} \cdot R^T(n) \cdot B(n)\]

\[\text{obtain updated estimates}\]

\[x(n+1) = x(n) + \Delta x(n)\]
\[y(n+1) = y(n) + \Delta y(n)\]

\[\text{repeat this procedure iteratively until desired accuracy/stop criterion is reached, typically good convergence}\]
Dilution of Precision (DOP)

- Definition

\[
DOP = \frac{\text{standard deviation of position error}}{\text{standard deviation of range error}}
\]

- Assumption: measurement errors are so small that the linearized model (e.g., Jacob-Newton method) holds in very good approximation.

- Let \((x, y)\) be the true position of the \(N\) and \(r\) be the true range.

Then, the measured range which seems to be due to a position \((\hat{x}, \hat{y})\) of the \(N\) reads

\[
\hat{r} = r + \frac{x-x_k}{r_k} \frac{\Delta x}{\Delta x} + \frac{y-y_k}{r_k} \frac{\Delta y}{\Delta y}
\]

- Set up an (over-)determined linear system of equations

\[
\begin{pmatrix}
\frac{x-x_0}{r_0} & \frac{y-y_0}{r_0} \\
\vdots & \vdots \\
\frac{x-x_{N-1}}{r_{N-1}} & \frac{y-y_{N-1}}{r_{N-1}}
\end{pmatrix}
\begin{pmatrix}
\Delta x \\
\Delta y
\end{pmatrix}
= 
\begin{pmatrix}
\hat{r} - r_0 \\
\hat{r} - r_1 \\
\vdots \\
\hat{r} - r_{N-1}
\end{pmatrix}
\]

\[
A \Delta = 1
\]

Position error measurement error independent and identically distributed zero mean, standard deviation \(\sigma\)
• Least squares pseudosolution

\[ \Delta = (R^T \cdot R)^{-1} \cdot R^T \cdot B \]

• Correlation matrix of the position error

\[ E^e \{ \Delta \cdot \Delta^T \} = E^e \{ (R^T \cdot R)^{-1} \cdot R^T \cdot B \cdot \cdot \cdot \cdot \} \]

\[ = (R^T \cdot R)^{-1} \cdot R^T \cdot \frac{E^e \{ B \cdot B^T \} \cdot R \cdot (R^T \cdot R)^{-1}}{\sigma^2} \]

\[ = \sigma^2 (R^T \cdot R)^{-1} \cdot R^T \cdot R \cdot (R^T \cdot R)^{-1} \]

\[ = \sigma^2 (R^T \cdot R)^{-1} \]

• Standard deviation of the position error

\[ \sigma_p = \sqrt{\text{trace} \{ \sigma^2 (R^T \cdot R)^{-1} \}} \]

\[ = \sigma \sqrt{\text{trace} \{ (R^T \cdot R)^{-1} \}} \]

\[ = \sigma \cdot \text{DOP} = \sqrt{\text{trace} \{ (R^T \cdot R)^{-1} \}} \]

Compare: Cramer Rao lower bound \((CRLB)\) for the case of independent and identically zero mean Gaussian distributed range measurement errors with variance \(\sigma^2\)
$k = 4 \ , \ F_0$
Time Difference of Arrival (TDOA)
FPs in the focal points of the hyperboloids
**Assumptions**

- For simplicity, 2D-scenario, extension to 3D-scenarios straightforward.
- Choose coordinates such that FPO is at the origin.
- Typical situation: synchronized clocks at the FPs (transmitters) but no time reference at the PR (receiver), e.g., GPS uses synchronized atomic clocks at the satellites.
- Choose FPO as the reference and measure K-1 range differences

\[ \Delta R_k = R_k - R_0 \quad k=1, \ldots, K-1 \]

(Note: equivalent to delay differences, "Time Difference of Arrival", TDOA.)
Time Difference of Arrival (TDMA)

- Each measured range difference

\[ \Delta r_n = r_n - r_0 = \sqrt{(x-x_n)^2 + (y-y_n)^2} - \sqrt{x^2 + y^2} \]

defines a hyperboloid of possible NT positions

\[ \Rightarrow \] find the intersection point of the hyperboloids, various algorithms to solve this nonlinear (in general overdetermined) system of equations

- In 2D scenarios 2 unknowns x and y
  - 3 FPs \[ \Rightarrow \] 2 equations but in general two intersection points, ambiguity might be resolved by some side information
  - 4 FPs \[ \Rightarrow \] one unique intersection point (if no measurement error)

- as compared to time of arrival methods one more FP is required (the time at the NT is one additional unknown which needs to be determined implicitly)
Analytical Method

• requires \( k = 3 \) FPs in 2D scenario

• idea: introduce \( r_0 \) as an additional unknown

• first step: compute \( x \) and \( y \) as linear functions of \( r_0 \)

\[
\frac{x^2 + y^2 - 2x x_n - 2y y_n + x_n^2 + y_n^2}{r_0^2} = \delta r_n^2 + 2\delta r_n r_0 + r_0^2
\]

\[
\frac{(x - x_n)^2 + (y - y_n)^2}{r_0^2} = \delta r_n^2 + 2\delta r_n r_0 + r_0^2
\]

= two linear system of equations

\[
2 \begin{pmatrix}
  x_n & y_n \\
  x_n & y_n
\end{pmatrix} \cdot \begin{pmatrix}
  x \\
  y
\end{pmatrix} = \begin{pmatrix}
  x_n^2 + y_n^2 - 2x_n r_0 - \delta r_n^2 \\
  x_n^2 + y_n^2 - 2y_n r_0 - \delta r_n^2
\end{pmatrix}
\]

from this \( x \) and \( y \) can be obtained as linear functions of \( r_0 \)

• second step: substituting these \( x \) and \( y \) being linear functions of \( r_0 \) into

\[
r_0^2 = x^2 + y^2
\]

one obtains a quadratic equation for \( r_0 \)

\( r_0 \) can be easily solved and with the \( r_0 \) obtained \( r_0 \) the final results for \( x \) and \( y \) can be obtained.
Least Squares Method

- Requires \( K \geq 4 \) FPs in 2D scenarios

- Similar to the analytical method. K-1 linear equations for the three variables \( x, y, \) and \( r_0 \) can be obtained:

\[
2x_k - 2yy_k + x_k^2 + y_k^2 = \Delta r_k^2 + 2\Delta r_k r_0
\]

- Set up an (over-) determined linear system of equations:

\[
2 \begin{pmatrix} x_1 & y_1 & \Delta r_1 \\ \vdots & \vdots & \vdots \\ x_{K-1} & y_{K-1} & \Delta r_{K-1} \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ r_0 \end{pmatrix} = \begin{pmatrix} x_1^2 + y_1^2 - \Delta r_1^2 \\ \vdots \\ x_{K-1}^2 + y_{K-1}^2 - \Delta r_{K-1}^2 \end{pmatrix}
\]

\[
X = (A^T \cdot A)^{-1} \cdot A^T \cdot B
\]

- Least squares pseudosolution

\[
X = (A^T \cdot A)^{-1} \cdot A^T \cdot B
\]
Jaus-Newton Method

The idea: linearize the nong differentiable equations around an initial estimate \((x(n), y(n))\)

\[
\Delta x_n \approx \sqrt{(x-x_n)^2 + (y-y_n)^2} - \sqrt{x^2 + y^2} \\
\approx \sqrt{(x(n)-x_n)^2 + (y(n)-y_n)^2} - \sqrt{x(n)^2 + y(n)^2} \\
+ \frac{2 \Delta x_n}{\Delta x(n)} \left( \frac{x-x(n)}{\Delta x(n)} \right) \Delta x(n) \\
+ \frac{2 \Delta x_n}{\Delta y(n)} \left( \frac{y-y(n)}{\Delta y(n)} \right) \Delta y(n) \\
= \Delta x_n(n) \\
+ \left( \frac{x(n)-x_n}{r_n(n)} - \frac{x(n)}{r_0(n)} \right) \Delta x(n) \\
+ \left( \frac{y(n)-y_n}{r_n(n)} - \frac{y(n)}{r_0(n)} \right) \Delta y(n)
\]

\[
\text{Normalize: } r_n(n) = \sqrt{(x(n)-x_n)^2 + (y(n)-y_n)^2} \\
r_0(n) = \sqrt{x(n)^2 + y(n)^2}
\]
* Set up an (over-)determined linear system of equations for computing the correction \((dx, dy)\) requires \(K \geq 3\) FP2s in 2D scenarios.

\[
\begin{pmatrix}
\frac{x(n)-x_0}{\tau_0(n)} & \frac{y(n)-y_0}{\tau_0(n)} & \frac{y(n)}{\tau_0(n)} \\
\vdots & \vdots & \vdots \\
\frac{x(n)-x_{K-1}}{\tau_{K-1}(n)} & \frac{y(n)-y_{K-1}}{\tau_{K-1}(n)} & \frac{y(n)}{\tau_{K-1}(n)} \\
\end{pmatrix} \cdot \begin{pmatrix} \Delta x(n) \\ \Delta y(n) \end{pmatrix} = \begin{pmatrix} \Delta x_0(n) \\ \vdots \\ \Delta x_{K-1}(n) - \Delta x_{K-1}(n) \end{pmatrix}
\]

\[
A(n) \cdot \Delta(n) = B(n)
\]

* Least squares pseudosolution

\[
\Delta(n) = (A^T(n) \cdot A(n))^{-1} \cdot A^T(n) \cdot B(n)
\]

* Obtain updated estimates

\[
x(n+1) = x(n) + \Delta x(n) \\
y(n+1) = y(n) + \Delta y(n)
\]

* Repeat this procedure iteratively until desired accuracy/stop criterion is reached, typically good convergence.
Angle of Arrival (AOA)
(no measurement errors)

draw 10 cm

intersection point of rectangles

for x > 2, fp = approx.
Least Squares Method

requires \( K \geq 2 \) FPs in 2D scenarios

\[
\tan (\beta_n) = \frac{y - y_n}{x - x_n}
\]

\[
(x - x_n) \tan (\beta_n) = y - y_n
\]

\[
x \tan (\beta_n) - y = x_n \tan (\beta_n) - y_n
\]

set up an (over-)determined linear system of equations

\[
\begin{pmatrix}
\tan (\beta_0) & -1 \\
\vdots & \vdots \\
\tan (\beta_{K-1}) & -1 \\
\end{pmatrix}
\begin{pmatrix}
x \\
y \\
\end{pmatrix}
= \begin{pmatrix}
x_0 \tan (\beta_0) - y_0 \\
\vdots \\
x_{K-1} \tan (\beta_{K-1}) - y_{K-1}
\end{pmatrix}
\]

least squares pseudosolution

\[
x = (A^T A)^{-1} A^T \cdot B
\]

Remark: also here a Gauss-Newton method based on linearization is possible. This is especially useful for the construction of hybrid techniques exploiting different kinds of measurements.