THOR - Thunderstorm Hazard Observation and Reporting

Konzept zur Lokalisierung von Gewittern

Konzept mit einem geometrischem Dreieck als Referenz

Berechnung der Position eines Punktes S relativ zu einem regelmäßigen Dreieck

Die Eckpunkte des Dreiecks werden als $A(x1,y1)$, $B(x2,y2)$, und $C(x3,y3)$ definiert.

Gleichungssystem (Dreieck)

Geometrische Eigenschaften (Dreieck)

- Geschwindigkeit der elektromagnetischen Welle: v = 299,792,458 m/s

Berechnung der Eckpunkte des Dreiecks

Die Gleichungen für die Distanz vom Punkt S(x_s,y_s) zu den Eckpunkten A, B , und C lauten:

$$
(x_s - x_1)^2 + (y_s - y_1)^2 = d_{SA}^2
$$

$$
(x_s - x_2)^2 + (y_s - y_2)^2 = d_{SB}^2
$$

$$
(x_s - x_3)^2 + (y_s - y_3)^2 = d_{SC}^2
$$

wobei

$$
d_{SA} = v \cdot t_{SA}
$$

$$
d_{SB} = v \cdot t_{SB}
$$

$$
d_{SC} = v \cdot t_{SC}
$$

Die Gleichungen können nun numerisch gelöst, um S zu finden.

Python

```
import numpy as np
from scipy.optimize import least_squares

# Gegebene Punkte
A = (0, 0)
B = (0.5, 0)
C = (0, 0.5)

# Gegebene Zeiten (in Sekunden)
t_SA = 3e-6
t_SB = 2e-6
t_SC = 4e-6

# Geschwindigkeit der elektromagnetischen Welle
v = 299792458

# Distanzen berechnen
d_SA = v * t_SA
d_SB = v * t_SB
d_SC = v * t_SC

# Koordinaten von A, B, und C
x1, y1 = A
x2, y2 = B
x3, y3 = C

# Gleichungssystem definieren
def equations(p):
    x_s, y_s = p
    eq1 = (x_s - x1)**2 + (y_s - y1)**2 - d_SA**2
    eq2 = (x_s - x2)**2 + (y_s - y2)**2 - d_SB**2
    eq3 = (x_s - x3)**2 + (y_s - y3)**2 - d_SC**2
    return (eq1, eq2, eq3)

# Anfangswert für das Lösen
initial_guess = (0.25, 0.25)

# Lösung der Gleichungen
result = least_squares(equations, initial_guess)

# Koordinaten von S
S_x, S_y = result.x

# Berechnung der Distanz SM und des Winkels theta
M_x = (x1 + x2 + x3) / 3
M_y = (y1 + y2 + y3) / 3
SM = np.sqrt((S_x - M_x)**2 + (S_y - M_y)**2)
```

```
theta = np.degrees(np.arctan2(S_y - M_y, S_x - M_x))

print(S_x, S_y, SM, theta)
```

Berechnung der Distanz SM und des Winkels θ (Dreieck)

Nachdem die Koordinaten von S bestimmt sind:

$$
SM=\sqrt{(x_s - x_m)^2 + (y_s - y_m)^2}
$$

wobei

$$
x_m = \frac{x_1 + x_2 + x_3}{3}
$$

$$
y_m = \frac{y_1 + y_2 + y_3}{3}
$$

und

$$
\theta = \arctan\left(\frac{y_s - y_m}{x_s - x_m}\right)
$$

Konzept mit einem geometrischem Achteck als Referenz

Berechnung der Position eines Punktes S relativ zu einem regelmäßigen Achteck

Geometrische Eigenschaften (Achteck)

- Radius des Umkreises: $R = 1$
- Geschwindigkeit der elektromagnetischen Welle: $v = 299{,}792{,}458\ m/s$

Berechnung der Eckpunkte des Achtecks

Die Eckpunkte eines regelmäßigen Achtecks relativ zum Mittelpunkt M (bei (0,0)) sind:

$$
P_k = (R \cos(\theta_k), R \sin(\theta_k))
$$

wobei

$$
\theta_k = \frac{2\pi k}{8}, k = 0, 1, 2, ..., 7

$$

Gleichungssystem (Achteck)

Die Gleichungen für die Distanz vom Punkt (S (x_s, y_s)) zu den Eckpunkten (P_k) lauten:

$$
(x_s - x_k)^2 + (y_s - y_k)^2 = d_k^2
$$

Die Gleichungen werden nun numerisch gelöst, um *S* zu finden.

Python

```
import numpy as np
from scipy.optimize import least_squares

# Geometrische Eigenschaften
R = 1  # Radius des Umkreises
v = 299792458  # Geschwindigkeit der elektromagnetischen Welle

# Eckpunkte des Achtecks berechnen
P = [(R * np.cos(2 * np.pi * k / 8), R * np.sin(2 * np.pi * k / 8)) for
k in range(8)]

# Beispielzeiten (in Sekunden) zu den Eckpunkten
t = [3e-6, 2.8e-6, 3.2e-6, 2.9e-6, 3.1e-6, 2.7e-6, 3.3e-6, 2.6e-6]

# Distanzen zu den Eckpunkten
d = [v * t_k for t_k in t]

# Gleichungssystem definieren
def equations(p):
    x_s, y_s = p
    return [(np.sqrt((x_s - x_k)**2 + (y_s - y_k)**2) - d_k) for (x_k,
y_k), d_k in zip(P, d)]

# Anfangswert für das Lösen
initial_guess = (0, 0)

# Lösung der Gleichungen
result = least_squares(equations, initial_guess)

# Koordinaten von S
S_x, S_y = result.x

# Distanz SM und Winkel theta berechnen
SM = np.sqrt(S_x**2 + S_y**2)
theta = np.degrees(np.arctan2(S_y, S_x))

print(S_x, S_y, SM, theta)
```

Berechnung der Distanz SM unde des Winkels θ

Nachdem die Koordinaten von S bestimmt sind:

$$
SM=\sqrt{x_s^2+y_s^2}
$$

$$
\theta = \arctan\left(\frac{y_s}{x_s}\right)
$$