

1 Least-square Verfahren im Dreidimensionalen

Berechnet man den quadratischen Fehler (die Summe aus den Quadratischen Abweichungen der Ranges von einem Punkt zu den Beacons zu den gemessenen Ranges) über ein Meshgrid von -10(m) bis +10(m) mit einer Schrittweite von 0.5(m) erhält man die Grafik aus Abbildung 1. Die Z-Koordinate wurde für alle X/Y Paare auf 0 gesetzt.

Die Koordinaten der Beacons sind:

$$\vec{B}_1 = (0 \ 0 \ 0)^T \quad ; \quad \vec{B}_2 = (2 \ 0 \ 0)^T \quad ; \quad \vec{B}_3 = (1 \ 0 \ 1)^T$$

Die gemessene Entfernung zu allen Beacons ist zunächst 1 m. Daraus ergibt sich eine Position der Antenne von $\vec{P} = (1 \ 0 \ 0)^T$. Der rote Punkt in allen folgenden Grafiken ist die Startposition, von der das least-square Verfahren »losläuft«, während der grüne Punkt die Endposition ist, gegen die der Algorithmus konvergiert. In der Grafik ist zu sehen, dass es nur ein Minimum des quadratischen Fehlers

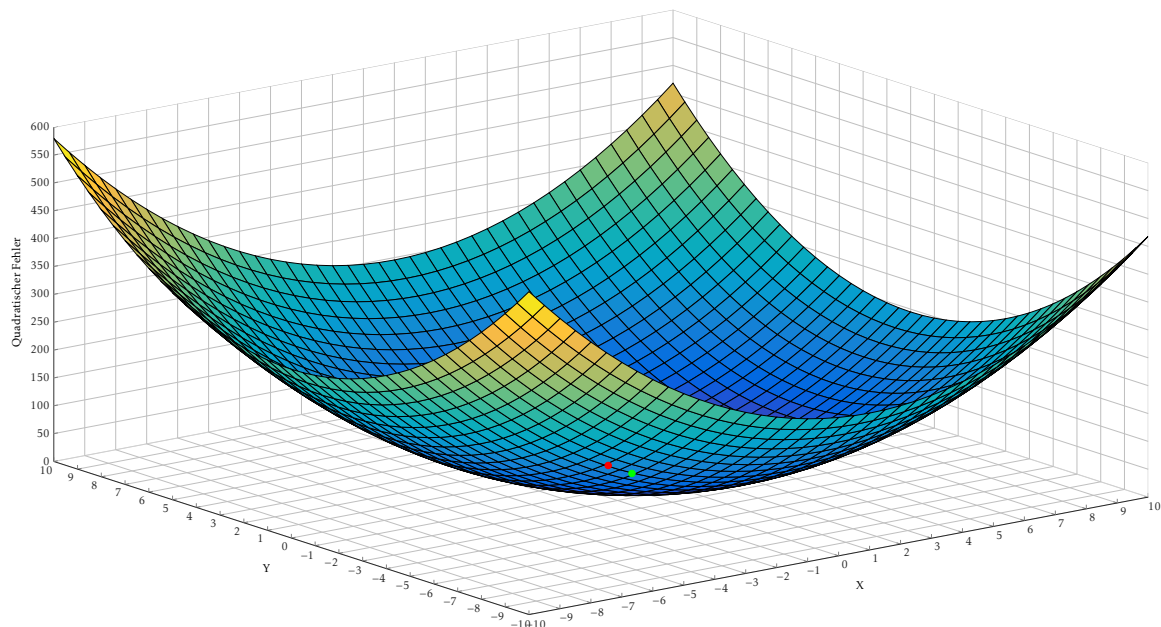


Abbildung 1: Quadratischer Fehler über die X und Y Koordinaten bei Z=0 mit 1 m gemessener Entfernung zu allen Beacons

gibt, in das der least-square laufen kann. Dies ist zu erwarten, da sich alle drei aufgespannte Kugeln von den Beacons und deren Ranges nur in genau einem Punkt schneiden (das Rauschen n ist in der Simulation zunächst 0). Bei ausreichenden Iterationsschritten des Verfahrens, konvergiert es bei jeder Startposition gegen die Position \vec{P} .

In einer realen Situation wird der Range-Messung ein Rauschen überlagert. Würde man beispielsweise statt 1m bei beiden Ranges 1.2m messen, ergebe sich die Grafik aus Abbildung 2 für den quadratischen Fehler. Es gibt nun zwei lokale Minima, die beide gleich gute Lösungen darstellen. Bei dem Startpunkt $\vec{P}_{\text{start},1} = (-1 \ -1 \ 0)^T$ berechnet sich die Lösung zu $\vec{P}_{\text{sol},1} = (1 \ -0.6 \ 0)^T$ (Abbildung 2). Für $\vec{P}_{\text{start},2} = (1 \ 1 \ 0)^T$ dagegen, läuft der Algorithmus in die Lösung $\vec{P}_{\text{sol},2} = (1 \ 0.6 \ 0)^T$ (siehe Abbil-

dung 3). Man sieht, dass die Auswahl der Startposition wichtig ist, da davon ab-

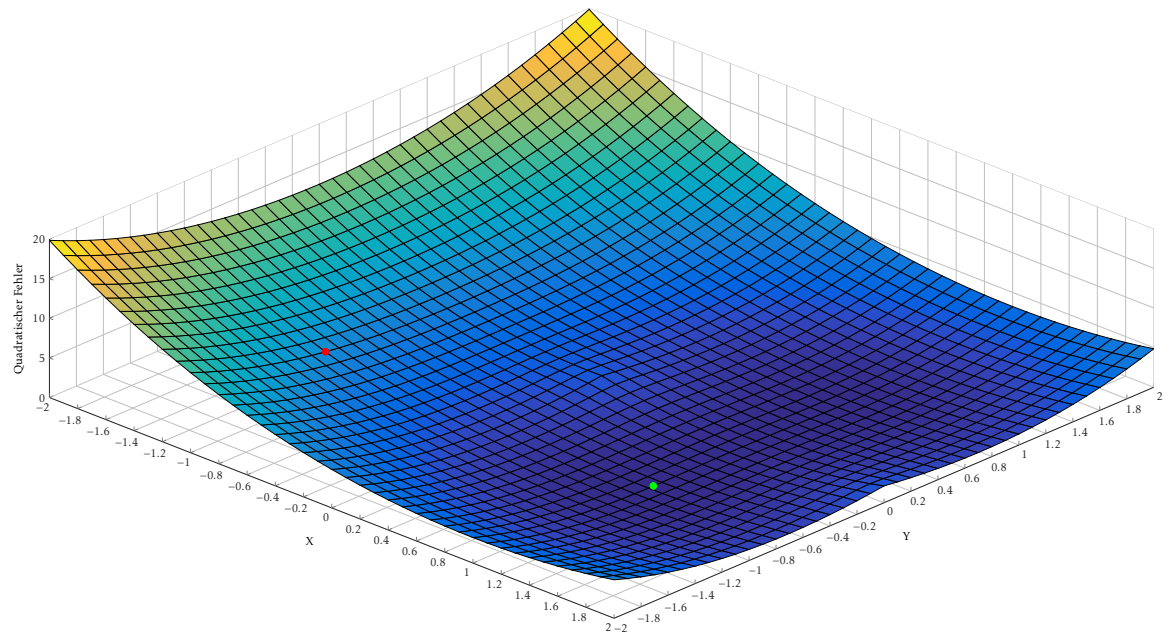


Abbildung 2: Quadratischer Fehler über X-Y Koordinaten bei Z=0 mit 1.2 m gemessener Entfernung zu allen Beacons und einer Startposition von $\vec{P}_{\text{start}} = (-1 \ -1 \ 0)$

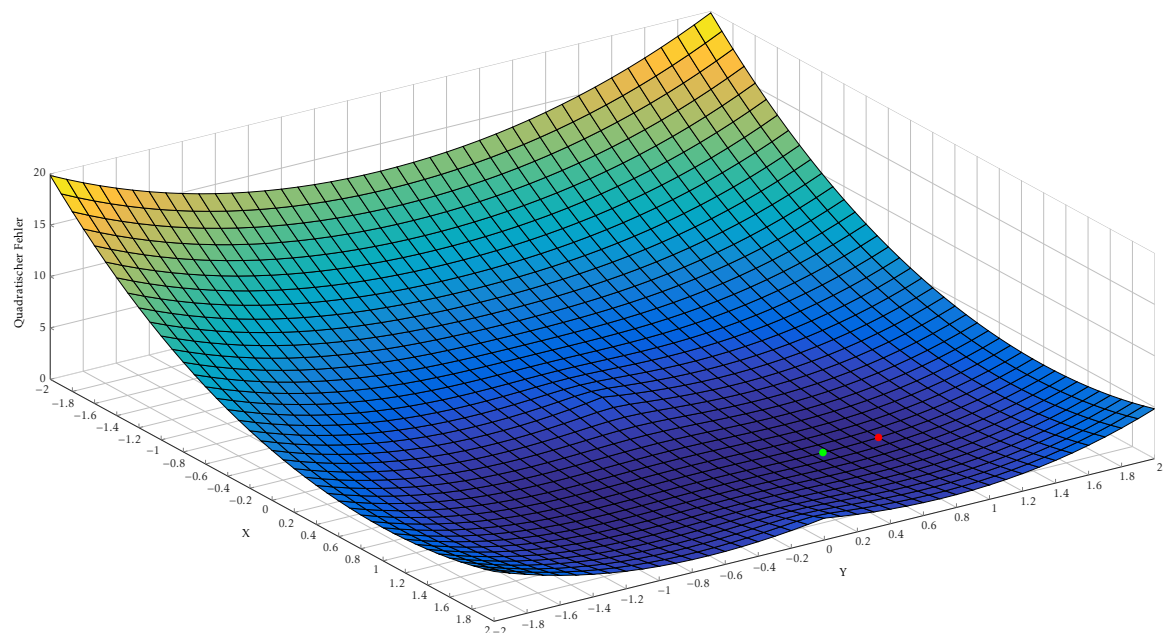


Abbildung 3: Quadratischer Fehler über X-Y Koordinaten bei Z=0 mit 1.2 m gemessener Entfernung zu beiden Beacons und einer Startposition von $\vec{P}_{\text{start}} = (1 \ 1 \ 0)$

hängt in welches Minimum man läuft. Wählt man eine ungünstige Startposition,

kann das Verfahren auch gegen $\pm\infty$ divergieren. Dies liegt daran, dass der Gradient am Startpunkt ins Unendliche zeigt. Eine Darstellung dieser Problematik ist mir nicht gelungen, da eine vierdimensionale Darstellung nötig wäre. Für die folgenden Simulationen wird deswegen komplett in ein zweidimensionales Szenario übergegangen.

2 Least-square Verfahren im Zweidimensionalen

Szenario mit 2 Beacons

Die Beacons und die Antenne werden auf die Positionen

$$\vec{B}_1 = (1 \ 0)^T \quad ; \quad \vec{B}_2 = (-1 \ 0)^T \quad ; \quad \vec{P}_{\text{ant}} = (0 \ 0)^T$$

Abbildung 4 zeigt den Verlauf des Quadratischen Fehlers im Intervall $[-2; +2]$ für alle X- und Y-Werte bei einer Rangemessung von 1m zu jedem Beacon (ideale Messung). In Abbildung 5 werden die gemessenen Radien zu beiden Beacons zu 1.5m

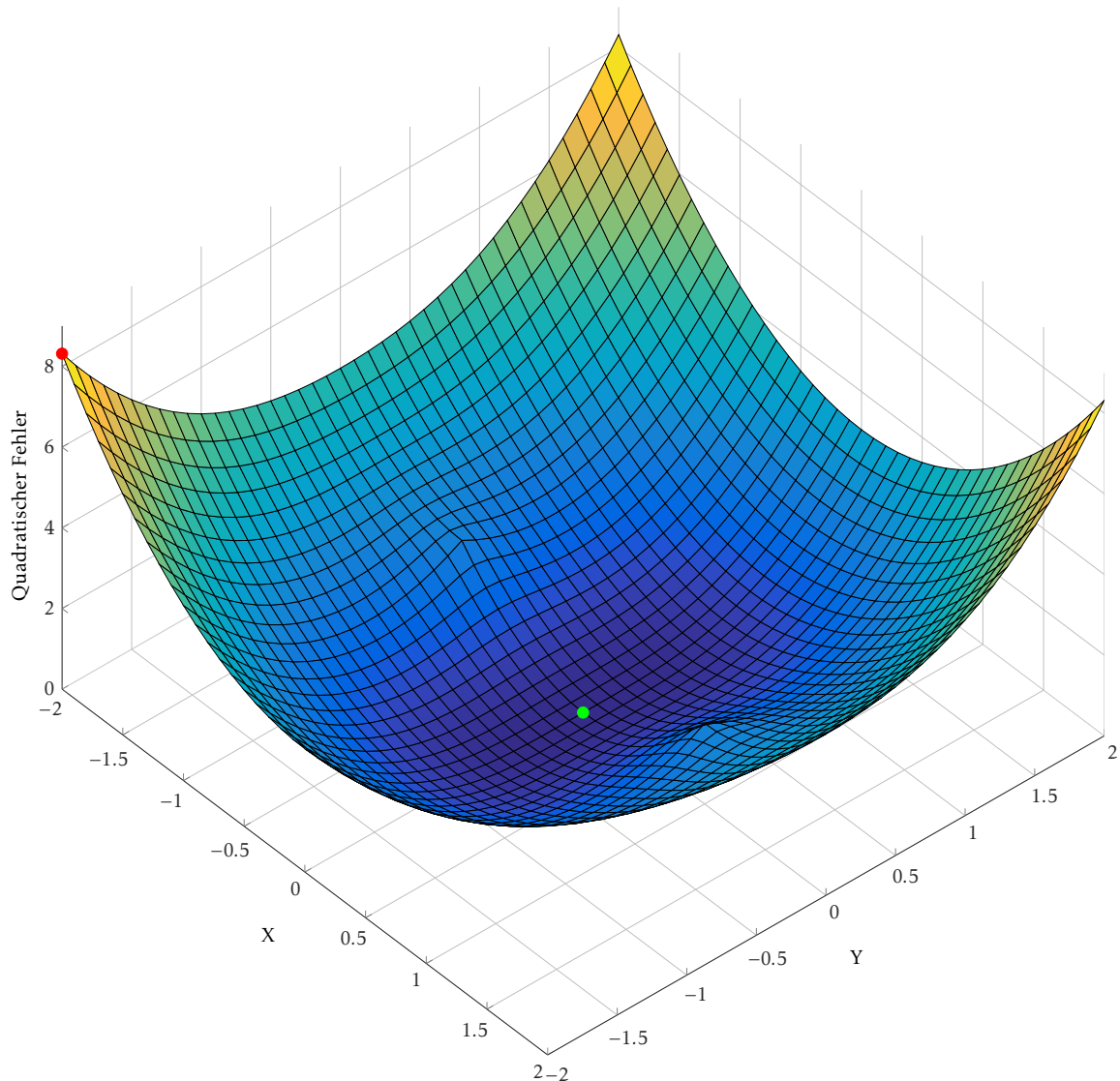


Abbildung 4: Quadratischer Fehler über X-Y Koordinaten bei $Z=0$ mit 1 m gemessener Entfernung zu beiden Beacons und einer Startposition von $\vec{P}_{\text{start}} = (-1 \ -1 \ 0)$

gesetzt. Man erhält nun wieder zwei Minima, die beide potentielle Lösungen sind. Wählt man den Startpunkt $(2 \ 2)^T$ konvergiert man ins Minimum $(0 \ 1.2)^T$ (Abbildung 5), für den Startpunkt $(-2 \ -2)^T$ dagegen in das Minimum $(0 \ -1.2)^T$ (Abbildung 6). Für das zweidimensionale Verfahren ist es mir nicht gelungen, ei-

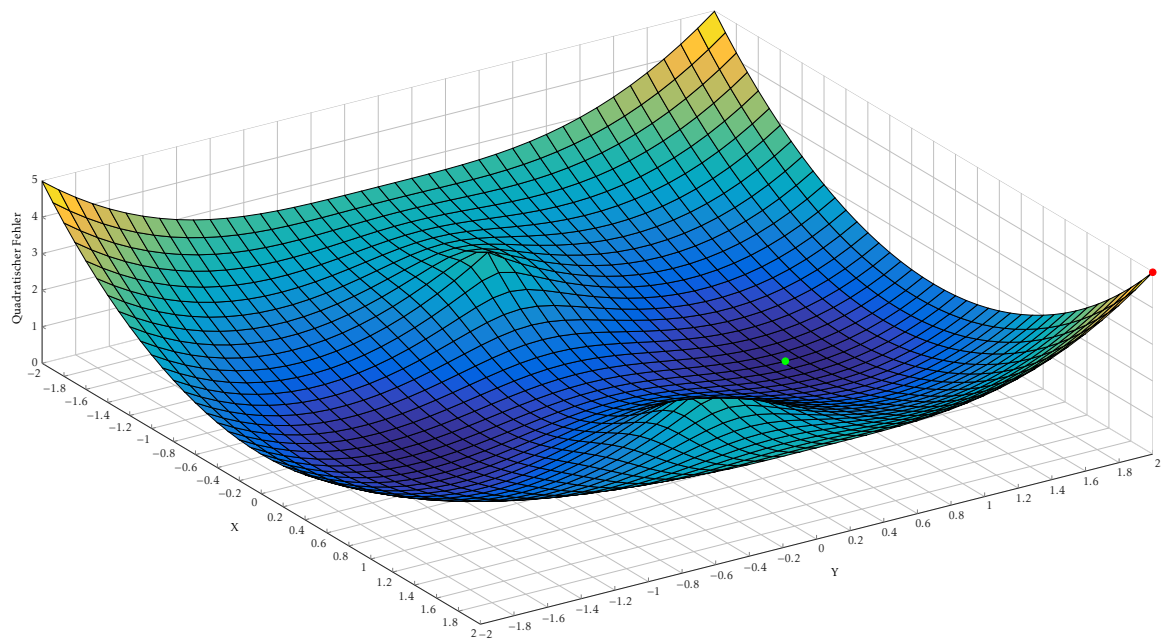


Abbildung 5: Quadratischer Fehler über X-Y Koordinaten im zweidimensionalen mit 2 Beacons

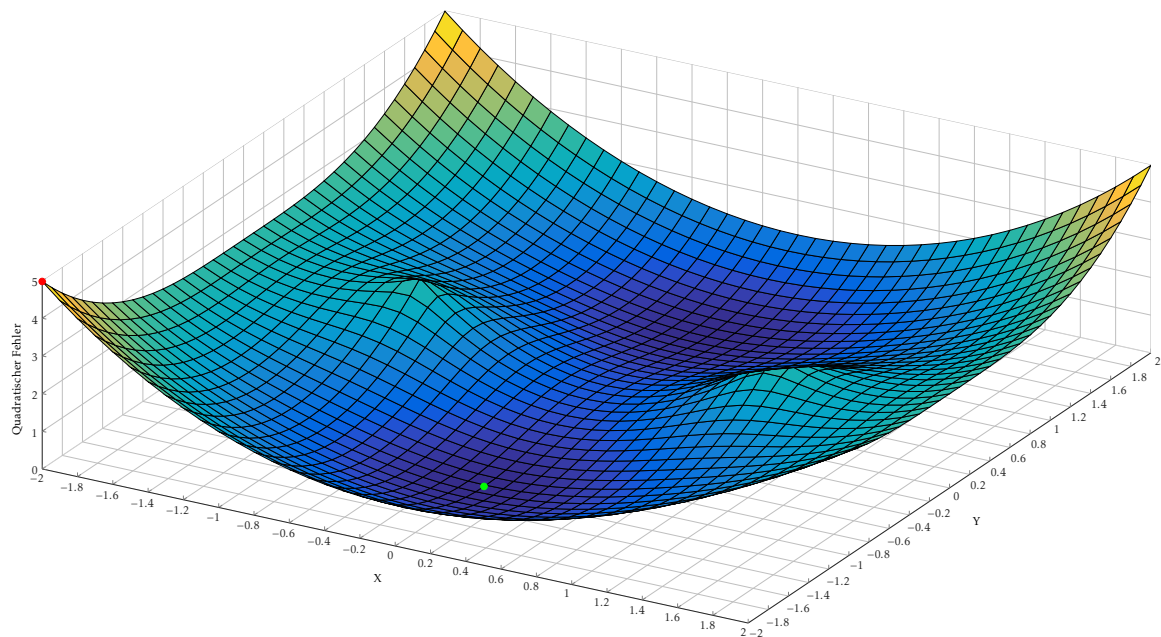


Abbildung 6: Quadratischer Fehler über X-Y Koordinaten im zweidimensionalen mit 2 Beacons

ne Startposition zu finden, bei der die Lösung divergiert.

Szenario mit 3 Beacons

Um die Problematik mit den zwei potentiellen Lösungen zu umgehen, kann nun

ein dritter Beacon $\vec{B}_3 = (1 \ 1)^T$ hinzugefügt werden. Der gemessene Range zu diesem Beacon wird auf 1.2m gesetzt, alle anderen Beacons und Ranges bleiben gegenüber Abschnitt 2 unverändert. Wie man in Abbildung 7 sieht, gibt es nun nur noch ein Minimum. Startet man nun wieder in der Position $(2 \ 2)^T$ so findet man das Minimum $(-0.1 \ 1.12)^T$ (Abbildung 7). Bei der Startposition $(-2 \ -2)^T$, bei der vorher ein anderes Minimum gefunden wurde, konvergiert man nun in das gleiche Minimum.

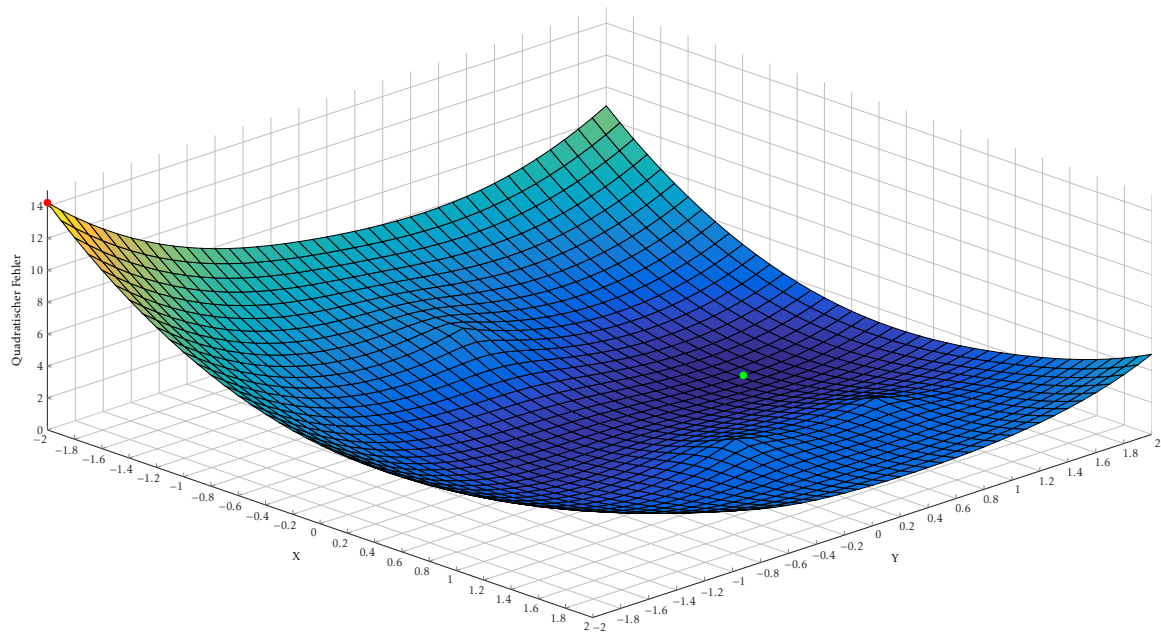


Abbildung 7: Quadratischer Fehler über X-Y Koordinaten im zweidimensionalen mit 3 Beacons

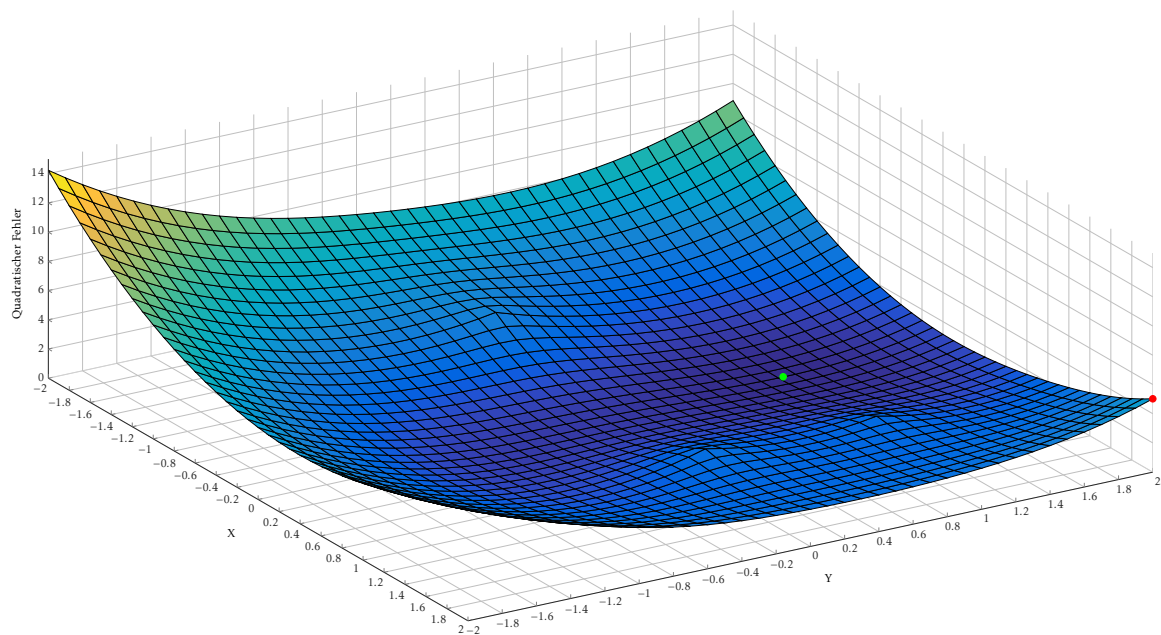


Abbildung 8: Quadratischer Fehler über X-Y Koordinaten im zweidimensionalen mit 3 Beacons

3 Ergebnisse

Die Simulation hat gezeigt, dass falsche Minima im Quadratischen Fehler beim least-square Verfahren durch zusätzliche Beacons umgehen lassen. Es ist nach wie vor nicht klar, wieso das Verfahren im dreidimensionalen bei manchen Startpositionen divergiert. Allerdings wurde festgestellt, dass die Häufigkeit des Divergiereins mit folgenden Methoden verringert werden kann:

- günstiges Setzen der Beacon-Positionen (gut räumlich verteilt, nicht in der gleichen Richtung zur Antenne!)
- verwenden von mehr als drei Beacons
- Kalkulieren der Startposition durch geometrisches Mittel der Beaconpositionen
- Nachführen der Startposition mit einer aktuell kalkultierten Position

Für den Test der aufgeführten Methoden wurde ein Testtreiber implementiert, der aus Beaconpositionen und Position der Antenne die korrekte Signalstärke berechnet. Dieser Signalstärke wird anschließend ein gaußverteiltes, weißes Rauschen mit variabler Varianz überlagert. Die Werte werden dann wie bei dem Wifi-Treiber in die indoor Navigationssoftware über eine POSIX Pipe übergeben. Befindet sich die Antenne innerhalb des von den Beacons aufgespannten Raumes und ist die Varianz des Rauschens gering genug, so divergiert das Verfahren nicht und eine Positionslösung kann bestimmt werden.

Im Testtreiber kann auch ein Trajektorienfile geladen werden. In diesem File werden X-, Y- und Z-Koordinaten der Antenne festgehalten und nacheinander als Position im Treiber gesetzt und die dazugehörigen Signalstärken ausgegeben. So kann simuliert werden, wie sich die Positionslösung der Navigationssoftware bei bewegen der Empfangsantenne verhält. Als Testfall wurde im folgenden eine lineare Trajektorie, die im Punkt $(10 \ 6 \ 3)^T$ beginnt und im Punkt $(-10 \ -6 \ -3)^T$ endet. Für das Additive White Gaussian Noise (AWGN) wurde eine Standardabweichung $\sigma = 0.3 \text{ dBm}$ angenommen. Die erhaltenen Werte vom *TestDriver* werden in der Navigationssoftware noch von einem gleitendem Mean-Filter mit der Fenstergröße 5 geglättet. Es werden vier Beacons mit den Positionen

$\vec{B}_1 = (-1 \ 1 \ 1)^T$ $\vec{B}_2 = (-3 \ 3 \ -3)^T$ $\vec{B}_3 = (8 \ -8 \ -5)^T$ $\vec{B}_4 = (8 \ 4 \ 0)^T$ verwendet. Abbildung 9 bis Abbildung 11 zeigen die berechneten und realen Positionswerte für die drei Raumachsen. Der Peak bei rund 160 s ist einem Divergieren

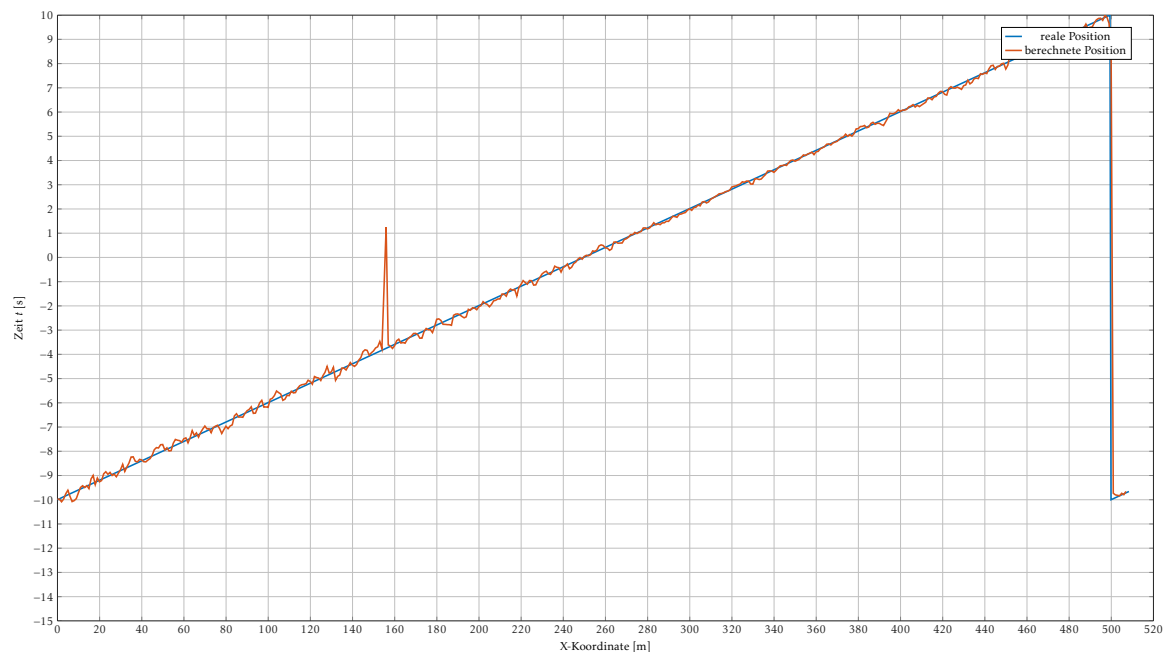


Abbildung 9: Reale X-Koordinate und X-Koordinate der Lösung des least-square Verfahrens über der Zeit

der Lösung zuzuschreiben. Während dem ganzen Szenario ist die Positionslösung nur einmal divergiert, was als akzeptabel angenommen werden kann. Da die Standardabweichung des Rauschens frei erfunden wurde, geben diese Plots zunächst keine quantitative Aussage über die Güte der Lösung. Eine weitere Messung mit realen Messwerten ist dazu nötig.

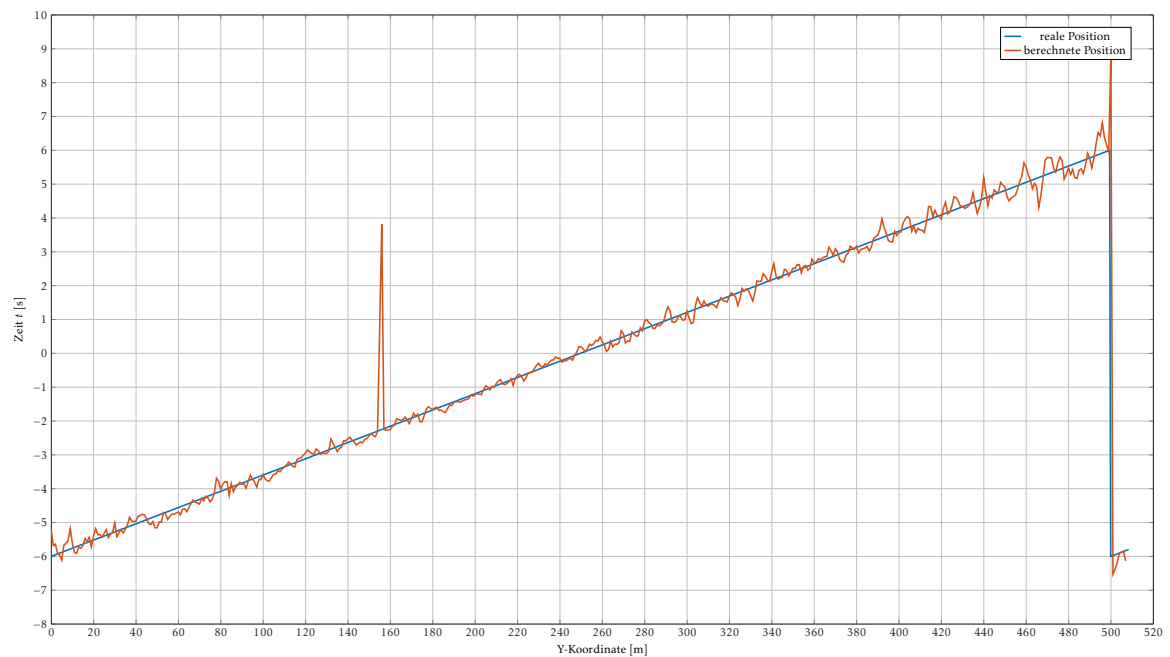


Abbildung 10: Reale Y-Koordinate und Y-Koordinate der Lösung des least-square Verfahrens über der Zeit

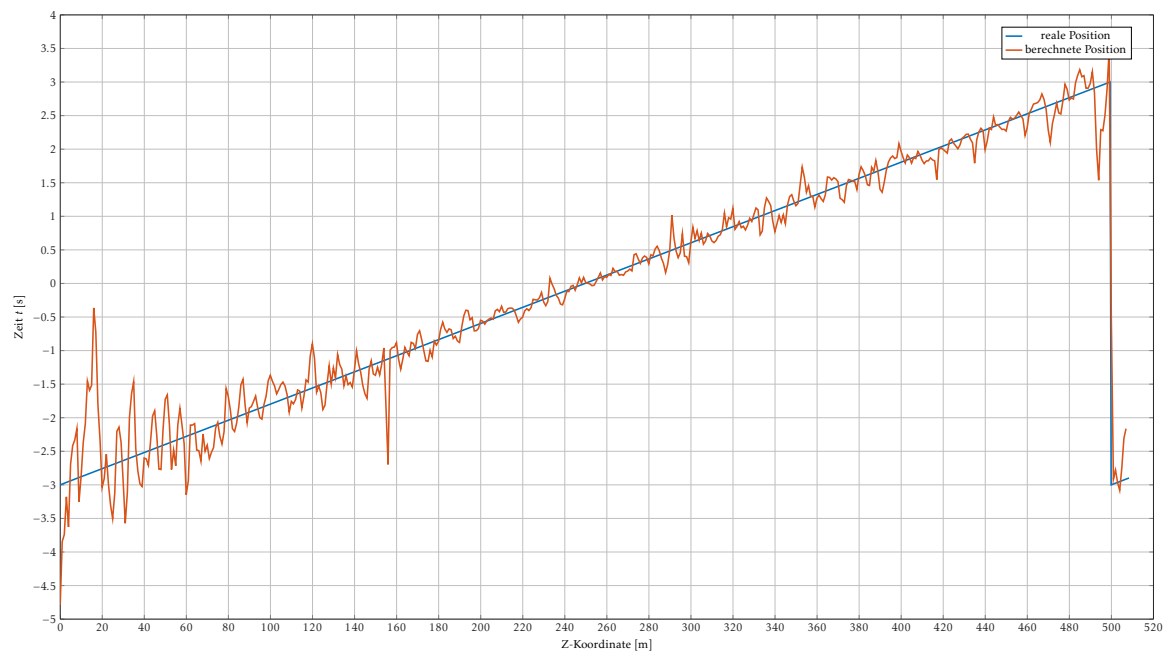


Abbildung 11: Reale Z-Koordinate und Z-Koordinate der Lösung des least-square Verfahrens über der Zeit