

calc,3d,arrows

tdplot_{screen}coords/.style = x = (1cm,0cm),y = (0cm,1cm),z = (-1cm,-1cm)

Spidercam

Andreas Stumpf, Sönke Bartelo

April 2020

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Um automatisierte Kameraführungen in großen und schwer zugänglichen Orten, wie z.B. Stadien, zu ermöglichen werden Kameras an Seilen befestigt. In diesem Dokument soll eine solche Kamerasteuerung geplant und durchgeführt werden. Dabei wird eine theoretische Herleitung der Ortskoordinaten einer solchen Konstruktion durchgeführt und anschließend an einem Model verifiziert. Ziel ist es entweder durch Vorgabe eines Endpunktes eine automatisierte Bewegungssteuerung zum definierten Punkt, oder eine Verfolgung einer Solltrajektorie zu erreichen.

2 Herleitung der inversen Kinematik Andi

2.1 Betrachtung im Zweidimensionalen

2.1.1 Definition der Punkte

Um das Anfangsproblem zu vereinfachen wird die Herleitung der Längenänderung des Seils eines Motors zunächst im zweidimensionalen durchgeführt. Das Übersetzen der Funktionen in den dreidimensionalen Raum ist damit trivial. Um eine Durchführbarkeit innerhalb der geometrischen Begrenzungen zu garantieren, wird ein Koordinatensystem so in den Raum gelegt, sodass sich für zwei der drei Eckpunkte ein x- Wert von Null ergibt. Damit lassen sich Begrenzungsfunktionen zwischen den Eckpunkten aufstellen, die eine einfache Berechnung der Machbarkeit zulassen. Der theoretische Aufbau der Konstruktion sieht dann wie folgt aus:

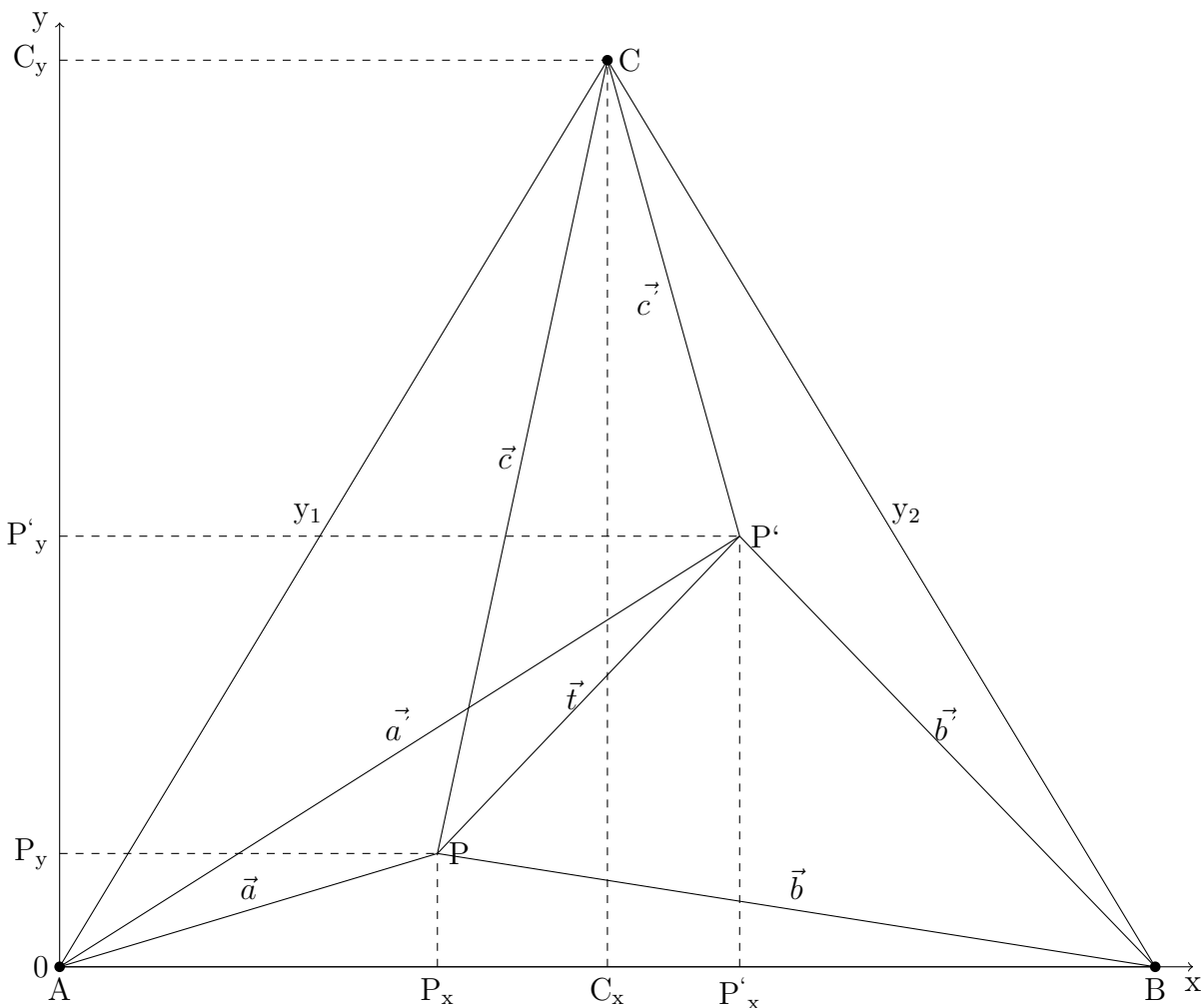


Abbildung 1: Darstellung der Transformation eines Punktes

2.1.2 Raumbegrenzung

Bevor eine Längenänderung ermittelt werden kann, muss zunächst garantiert werden, dass alle Punkte der Bewegung erreicht werden können. Da zuvor zwei der drei Eckpunkte auf die x- Achse gelegt wurden kann nun einfach ermittelt werden, ob alle Punkte unterhalb der Verbindungsgeraden \overline{AC} und \overline{CB} liegen. Dafür werden folgende Geradengleichungen aufgestellt:

$$y_1 = m_1 * x + b_1 \quad (1)$$

$$y_2 = m_2 * x + b_2 \quad (2)$$

Mit $b_1=0$ und $m_1=\frac{\Delta y}{\Delta x}$

$$y_1 = \frac{c_y - a_y}{c_x - a_x} * x \quad (3)$$

Mit $m_2=\frac{\Delta y}{\Delta x}$ ergibt sich y_2 zu:

$$y_2 = \frac{c_y - b_y}{c_x - b_x} * x + b_2 \quad (4)$$

Da die Koordinaten von Eckpunkt B bekannt sind, kann b_2 mit Hilfe von m_2 ermittelt werden.

$$m_2 = \frac{y_2 - y_1}{x_2 - x_1} \quad (5)$$

Mit $b_2=y_2, x_2=0$ und $m_2=\frac{c_y-b_y}{c_x-b_x}$ folgt:

$$b_2 = \frac{(c_y - c_x) * (c_y - b_y)}{c_x - b_x} \quad (6)$$

Da nun beide Geradengleichungen definiert sind, kann überprüft werden, ob alle Punkte der Bewegung folgende Bedingungen erfüllen:

$$p_x < c_x \wedge p_y < y_1(p_x) \vee p_x \geq c_x \wedge p_y < y_2(p_x) \quad (7)$$

2.1.3 Berechnung der Längenänderungen

Ist garantiert, dass alle Punkte der Bewegung realisierbar sind kann die Berechnung der Seillängenänderung erfolgen. Die Längenänderung der Steuerseile kann geometrisch bestimmt werden. Dabei wird davon ausgegangen, dass die aktuelle Kameraposition P, sowie die Montagepositionen der Aktuatoren A, B und C bekannt sind.

Die Länge des Seils a zu Motor A kann bestimmt werden als Subtraktion des Vektors zu Motor A und des Vektors zu P.

Die Länge des Seils a' zu Motor A kann bestimmt werden als Subtraktion des Vektors zu Motor A und des Vektors zu P.

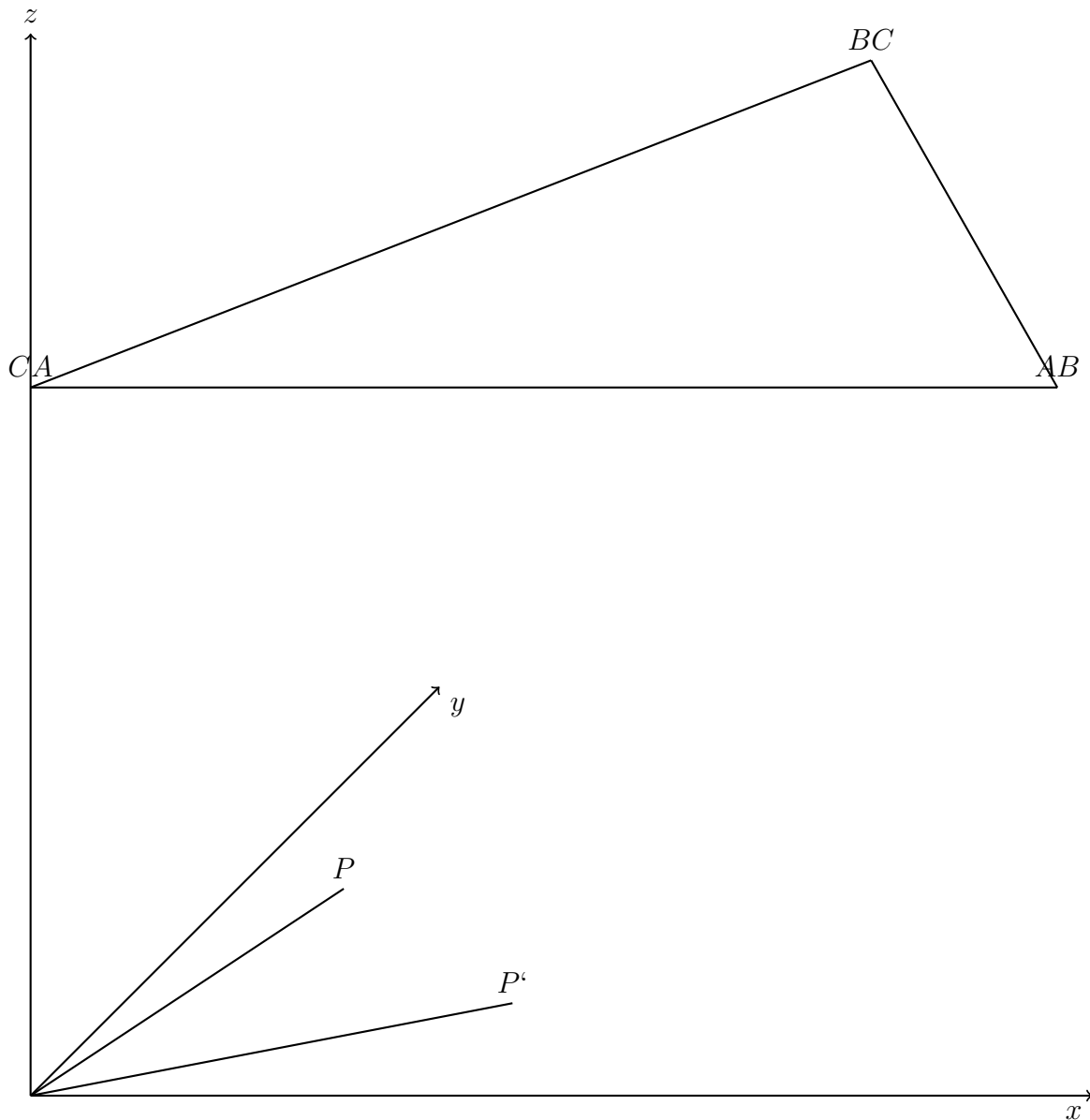
Damit ergibt sich die Längenänderung des Seils zu Motor A als Differenz der Längen a und a' . Analog dazu ergeben sich die Längenänderungen der Seile zu Motor B und C. Soll ein Punkt mit drei Aktuatoren in der Ebene bewegt werden, so ergibt sich die Längenänderung Δn in Abhängigkeit der Montageposition des Aktuators und der angestrebten Kameraposition zu:

$$\Delta n = |\vec{n}| - |\vec{n}| = \sqrt{(P'_x - N_x)^2 + (P'_y - N_y)^2} \quad (8)$$

2.2 Betrachtung im Dreidimensionalen

2.2.1 Definition der Punkte

Die Positionen der Motoren bleiben nach der Betrachtung im Zweidimensionalen identisch, jedoch wird jeder Motorposition eine Z- Koordinate hinzugefügt, sodass sich die Eckpunkte nun im Format $N=(N_x \ N_y \ N_z)$ befinden. Die x- und y- Komponenten bleiben bestehen, um sämtliche Vereinfachungen aus dem Zweidimensionalen übernehmen zu können.



2.2.2 Raumbegrenzung

Die Berechnung der räumlichen Begrenzung einer Bewegung im dreidimensionalen Raum kann äquivalent zur Bewegung im Zweidimensionalen durchgeführt werden. Die Begrenzung wird

jedoch um folgende Bedingung erweitert:

$$P_z < A_z \wedge P_z > 0 \quad (9)$$

Damit muss nun jeder Punkt einer Bewegung folgende Bedingungen erfüllen:

$$p_x < c_x \wedge p_y < y_1(p_x) \wedge P_z < A_z \wedge P_z > 0 \vee p_x \geq c_x \wedge p_y < y_2(p_x) \wedge P_z < A_z \wedge P_z > 0 \quad (10)$$

2.2.3 Berechnung der Längenänderungen

Die Berechnung der Längenänderung ergibt sich analog zum Zweidimensionalen, wird jedoch um die Z- Komponente erweitert.

$$\Delta n = |\vec{n}| - |\vec{n}| = \sqrt{(P'_x - N_x)^2 + (P'_y - N_y)^2 + (P'_z - N_z)^2} \quad (11)$$



Abbildung 2: Die Mechanische Konstruktion mit Vereinfachungen.

3 Herleitung der inversen Kinematik

3.1 Mechanische Konstruktion

Vereinfacht wird im folgenden Angenommen, dass sich die gesamte Anordnung der Kameraposition zunächst durch ein einfaches, gleichseitiges Dreieck annehmen lässt. Die Entscheidung zu Gunsten eines Dreiecks basiert auf der Tatsache, dass zur Befestigung einer Plattform im dreidimensionalen Raum mindesten drei Befestigungspunkte benötigt werden. Daraus resultiert die Wahl von drei Seilen zur Steuerung.

3.2 Inverse Kinematik

Zunächst soll an dieser Stelle die Pose der Kamera zu bestimmen. Diese besteht aus den kartesischen Koordinaten

$$\vec{p} = \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix}, \quad (12)$$

mit den jeweiligen Koordinaten in x,y und z Richtung. Die Parametrierung der Winkel erfolgt über Kardan-Winkel (Roll, Pitch, Yaw). Diese zeichnet sich dadurch aus, dass die Drehung um x-, y- und z-Achse jeweils um die alten Achsen durchgeführt wird. Daraus ergibt sich Bestimmung der Lage

$$\vec{w} = \begin{pmatrix} \phi_e \\ \theta_e \\ \psi_e \end{pmatrix}, \quad (13)$$

jeweils bezogen auf den Endeffektor-Frame.