

Entwicklung eines Seilberechnungsprogramms für Seilbahnen in Python

mit einer Schnittstelle zu AxisVM

Diplomand



Pascal Schnyder

Einleitung: Bei älteren Seilbahnstützen hat man häufig Probleme, diese nach den neuesten Normen nachzuweisen. Oftmals liegt das Problem darin, dass man früher die Windkraft in der Bemessungssituation "Ausser Betrieb" deutlich kleiner angesetzt hat. Die höhere rechnerische Windkraft führt zusammen mit langen Seilsätteln und verschiedenen langen Seilfeldern zu einem hohen Torsionsmoment um die Hochachse der Stützen. Die Diagonalstäbe alter Stützen kann man mit den daraus entstehenden Druckkräften nicht nachweisen und auch die Verformung übersteigt die Anforderungen in der Norm. In der Realität wird diese Verformung aber dazu führen, dass sich der Tangentialpunkt des Tragseils auf dem Seilsattel mehr zur Stütze hin verschieben wird und sich das Torsionsmoment in der Folge reduziert. Diese Interaktion wird jedoch aktuell in den Berechnungen nicht berücksichtigt. Das Ziel dieser Arbeit ist es deshalb, ein Seilberechnungsprogramm zu entwickeln, welches die Interaktion zwischen Seil und Stütze berücksichtigen kann.

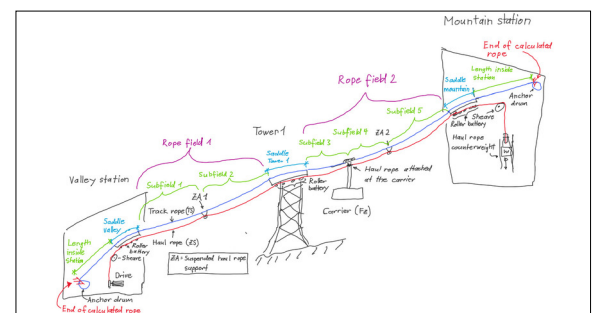
Vorgehen: Um dies umsetzen zu können, musste zuerst ein Seilberechnungsprogramm für Seilbahnen programmiert werden. Für die vorhergehende Arbeit im Rahmen des Masterstudiums wurde bereits ein Seilberechnungsprogramm für eine Seilbahn in Python entwickelt. Dieses Programm wurde jedoch sehr einfach gehalten und konnte nur für die eine Seilbahn, für die es entwickelt wurde, eingesetzt werden. Die Einflüsse von Reibung und Temperatur wurden vernachlässigt und die Seilberechnung erfolgte nur nach der parabolischen Methode. Da es sich ausserdem um eine Seilbahn mit Spannungsgewicht abgespanntem Tragseil handelte, vereinfachte dies die Berechnung erheblich. Dieses Seilberechnungsprogramm soll nun aber alle möglichen Arten von Pendelbahnen, mit unterschiedlicher Anzahl von Stützen, mit einem oder zwei Tragseilen, mit Spannungsgewicht oder fest abgespanntem Tragseil und auch mit Zwischenaufhängungen für das Zugseil, berechnen können. Es zeigte sich schnell, dass die Entwicklung so eines Programms aufgrund der vielen möglichen Konfigurationen sehr komplex ist. Der Fokus der Arbeit wurde deshalb auf die Entwicklung des Seilberechnungsprogramms selbst gelegt und die weiteren Ziele werden später weiterverfolgt.

Ergebnis: Die Entwicklung des Seilberechnungsprogramms erfolgte ausgehend von der einfachsten Seilbahnkonfiguration – ohne Stützen, Kabinen oder Zwischenaufhängungen – und wurde anschliessend schrittweise erweitert. Je nach Anzahl der Stützen und Zwischenaufhängungen führt die Berechnung zu einem System nichtlinearer Gleichungen mit über 100 unbekannten Variablen. Für die Lösung dieser nichtlinearen Gleichungssysteme wurde die Funktion `fsolve` aus der Python-Bibliothek `scipy.optimize` verwendet. Um aber diese grossen, nichtlinearen Gleichungssysteme überhaupt lösen zu können, sind

gute Startwerte notwendig. Der Berechnungslauf erfolgt deshalb schrittweise, wobei die Ergebnisse vom vorhergehenden Schritt die Startwerte des nächsten Schritts sind. Zuerst wird nur das leere Tragseil ohne Zugseil, ohne Kabine, ohne Zwischenaufhängungen und ohne den Einfluss von Reibung und Temperatur berechnet. Im nächsten Schritt werden dann die Temperatur und die Reibung berücksichtigt. Anschliessend werden Zwischenaufhängungen und folgend das Zugseil berücksichtigt. Im letzten Schritt wird dann das Tragseil, das Zugseil und auch die Kabine zusammen in einem Gleichungssystem für die ganze Seilbahn berechnet.

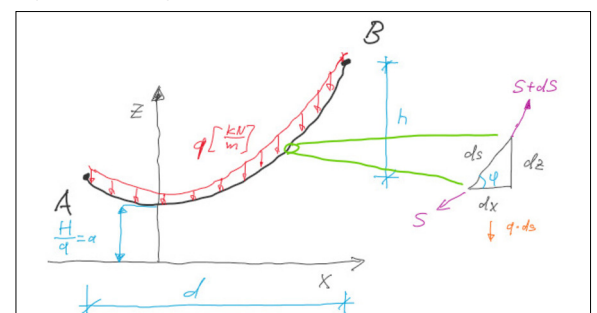
Relevante Elemente einer Seilbahn für die Berechnung der Seilkräfte

Eigene Darstellung



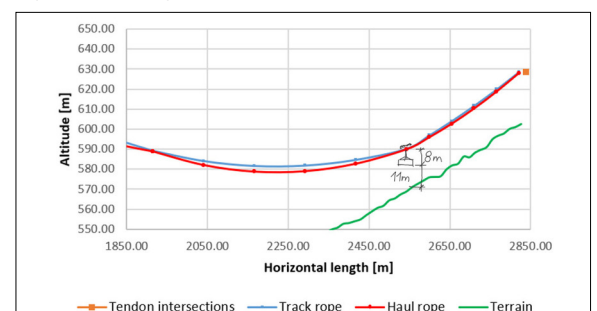
Prinzip der Seilberechnung mittels Kettenlinie

Eigene Darstellung



Darstellung eines Ergebnisses des Seilberechnungsprogramms

Eigene Darstellung



Referenten

Prof. Dr. Ivan Marković,
Prof. Dr. Hans Fritz

Korreferent

Dipl. Masch. Ing. FH
Bernd Populorum,
Garaventa AG

Themengebiet

Civil Engineering

Projektpartner

GTX Engineering
GmbH, Sisikon, Uri