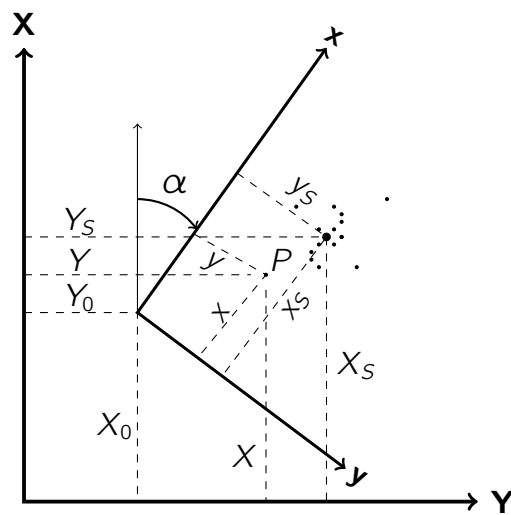


Helmerttransformation

Programm für Windows auf Python-Basis
Version 1.01

Norbert Fuhrmann, Kerpen



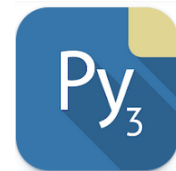
Programminformation

Das Programm ist in Standardpython geschrieben. Python ist eine höhere, interpretierende Programmiersprache von der <Python Software Foundation>. In Python geschriebener Code lässt sich in diversen Betriebssystemen anwenden. Der Python-Interpreter <IDLE Shell 3.11.0> für Windows kann im Internet unter <https://www.python.org/downloads/> heruntergeladen werden.



Das Programm <Helmertransform.pyw> ist mit Hilfe des Python-Interpreters auf einen PC mit **Windows** lauffähig. Oder man nimmt das kompilierte Programm <Helmertransform.exe>. Dieses Programm ist ausschließlich für Windows gedacht; dies wegen der Ausgabe am besten auf dem Desktop. Es wurde mit <pyinstaller> Version 5.6.2 aus dem Python-Code erzeugt. Unter <Android> ist es nicht einsatzfähig; für die Steuerung des Programms wäre eine Funktion erforderlich, die von einem android-kompatiblen Python-Interpreter nicht unterstützt wird.

Nach Programmaufruf können Daten manuell in eine Tabelle eingetragen und editiert werden, wobei die Eingangswerte in der Tabelle nach einem erfolgreichen Berechnungsdurchlauf durch eine Datenausgabe in eine Datei gesichert werden kann. Oder aber die Eingabe der Daten erfolgt von vornherein über eine Datei mit entsprechender Codierung der Elemente.



Ordnungsgemäße Berechnungen bzw. Durchläufe werden angezeigt, wenn sich ein **grüner Haken** im Display neben der Programmüberschrift zeigt **und keine** Fehlermeldung angezeigt wird.

Eingabewerte können editiert werden. Es zeigt sich ein Warnzeichen, wenn eine Taste betätigt wurde und eine neue Berechnung erfolgen muss. Blassgelb hinterlegte Felder zeigen Ergebnisse, die nicht editiert werden können.


Eine Punkte- und Vektorenübersicht der Restklaffen ermöglicht eine visuelle Begutachtung der Transformation. Eine Druckausgabe dieser Grafik ist nicht möglich. Eine dauerhafte Dokumentation dürfte mit einem <screenshot> möglich sein¹. Allerdings sollten screenshots, und damit eine dauerhafte Dokumentation, nur dann gemacht werden, wenn sich ein grüner Haken am Display neben der Programmüberschrift zeigt und **keine Fehlermeldung** erscheint; nur um fehlerhafte Dokumentationen zu vermeiden. Dies gilt ebenso bei der Übernahme von Daten- bzw. Printdateien.

¹Beispiele für Windows: <microsoft swift>

Helmerttransformation

Eine Helmerttransformation dient der Überführung von Koordinaten eines Startsystems (System A) in ein Zielsystem (System B) mit Hilfe bekannter Koordinaten in beiden Systemen (identische Punkte, Pass- oder Stützpunkte) durch die vorangehende Ermittlung von Transformationsparametern, wobei unvermeidliche Abweichungen minimiert werden sollen. Mit den Parametern können die Koordinaten der zu transformierenden Punkte für das Zielsystem berechnet werden. Eine 3-Parametertransformation besteht aus 2 Verschiebungen und einer Drehung, die 4-Parametertransformation aus 2 Verschiebungen, einer Drehung und einer Maßstabsanpassung. Weitere Parameter sind denkbar, werden hier aber nicht behandelt.

Durch mit Überbestimmungen erzeugte Transformationsparameter können die Zielkoordinaten der Passpunkte nicht übereinstimmend berechnet werden; es entstehen Differenzen, die als Restklaffen bezeichnet werden. Deshalb werden oder können die Koordinaten der zu transformierenden Punkte durch eine entsprechende Verteilung angepasst werden. Die Verteilung der Restklaffen auf umzuformende Punkte ist kein Bestandteil der Helmerttransformation mit Minimumsbedingungen, sondern mehr oder weniger willkürlich. Hier werden streckenabhängige Verteilungen angeboten, nämlich proportional mit den Gewichten $1/s$, $1/(s*\sqrt{s})$ bzw. $1/s^2$. Näheres entnehme man dem Kapitel: Formelentwicklung.

Mit dem Button  besteht die Möglichkeit, ausgewählte Passpunkte in den Eingabefeldern bis zur minimalsten Anzahl der Passpunkte (= 2) deaktiv zu stellen, was andererseits zur Fehlersuche und Ausschaltung fehlerhafter Messungen durch die hilfreiche Restklaffenanzeige möglich ist. Inaktiv gestellte Punkte können auch wieder reaktiviert werden. Deaktive Passpunkte werden auf roten Hintergrund angezeigt. Die dabei angegebenen Restklaffen werden mit den ermittelten Parametern der aktiven Passpunkte berechnet, sodass die Abweichungen klarer hervortreten, weil die Elemente der deaktivierten Passpunkte nicht in die Parameterberechnung einfließen.

Eine Limitierung von jeweils maximal 10 Punkten bei manueller Eingabe dürfte aus praktischen Gründen vernünftig sein. Bei Überschreitung der Limits werden Defaultwerte eingesetzt. Ansonsten wäre anschließend die Datei der Datenausgabe zu editieren und einzulesen. Eine Limitierung der Anzahl der Punkte gilt nicht bei der Eingabe von Dateien.

Programmbeschreibung

Ein Maßstabsfaktor kann Eigenschaften einer Korrektur wie auch einer Reduktion haben: Ein Korrekturfaktor berichtigt die ursprüngliche Messung in Annäherung an den wahren Wert (z. B. Kalibrierungswerte), ein Reduktionsfaktor behandelt z. B. die Horizontierung einer Strecke oder eine Abbildungsverzerrung (Verebnung).

Dem hier vielfach in der Eingabe vorkommenden Reduktionsfaktor r (oder auch Maßstabsfaktor m) bei Strecken bzw. Koordinaten liegt folgende Definition zu Grunde:

$$\text{reale örtliche, horizontierte Strecke (hor.)} \times r = \text{Strecke in einer Abbildung (z.B. UTM)}$$

Beispiel: Eine gemessene Strecke wird korrigiert durch Additionskonstante, instrumentell bedingtem Maßstab; eine Strecke wird reduziert durch Horizontierung, Höhenlage, Abbildung, Abbildungsmaßstab.)

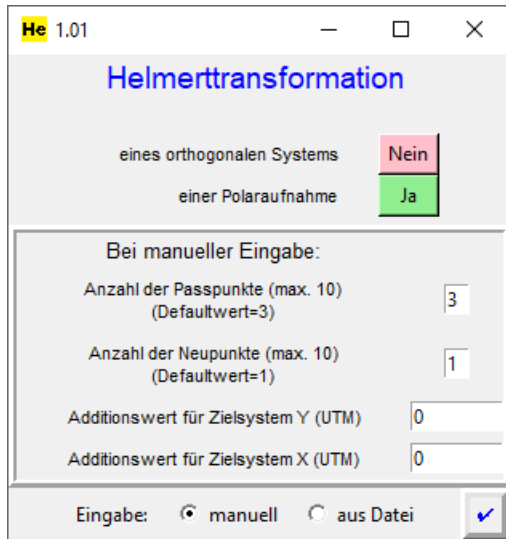


Abbildung 1

Entrée-Fenster, Beispiel der Wahl einer Polaraufnahme

Die Daten können manuell eingegeben oder per Datendatei (*.txt) eingelesen werden und später nach der Berechnung in einer Datenausgabe (*.txt) wieder aktuell gespeichert werden. Ob das umzuformende System eine polare Aufnahme oder orthogonale Koordinaten sind, muss entsprechend im Entrée-Fenster ausgewählt werden. Beim Einlesen einer Datei entscheidet das Programm aufgrund der Codierung in den *.txt-Dateien welche Daten einzulesen sind.

Bei einer manuellen Eingabe muss vorher angegeben werden, wie viele Passpunkte und umzuformende Punkte vorhanden sind, damit die Maske korrekt aufgerufen wird. Bei einer manuellen Eingabe ist die Maske leer, bei einer Dateneingabe entsprechend besetzt. Eine Eingabe durch eine Datei erfordert keine Angabe einer Punktzahl. Diese wird dann automatisch ermittelt

Die Syntax für die Dateieingaben in den *.txt Dateien sind in der nachfolgenden Aufstellung (vgl. auch die Beispieldaten auf Seite 16) genannt. Die Elemente werden durch den Separator ; getrennt. Kommas als Dezimalzeichen sind erlaubt, werden aber nach einem Berechnungsdurchlauf in der Dateiausgabe in einen Punkt umgewandelt, der in Python benutzt wird.

Codierungen der Dateieingabe:

01;							(Projektbezeichnung)
C bzw. 99; oder Leerzeile							(Kommentar, wird nicht bearbeitet)
10; PKZ; Ayi;	Axi;			BYi;	BXi;		(Koordinaten Passpunkte, System A und B)
20; PKZ; Ajj;	Axj;						(Koordinaten zu transformierende Punkte System A)
30; PKZ; ARichtung i;	AZenitdistanz i;	AStrecke i;		BYi;	BXi;		(Polarelemente System A, Koordinaten Sytem B)
40; PKZ; ARichtung j;	AZenitdistanz j;	AStrecke j;					(zu transformierende Polarelemente System B)

Codierungen der Dateiausgabe:

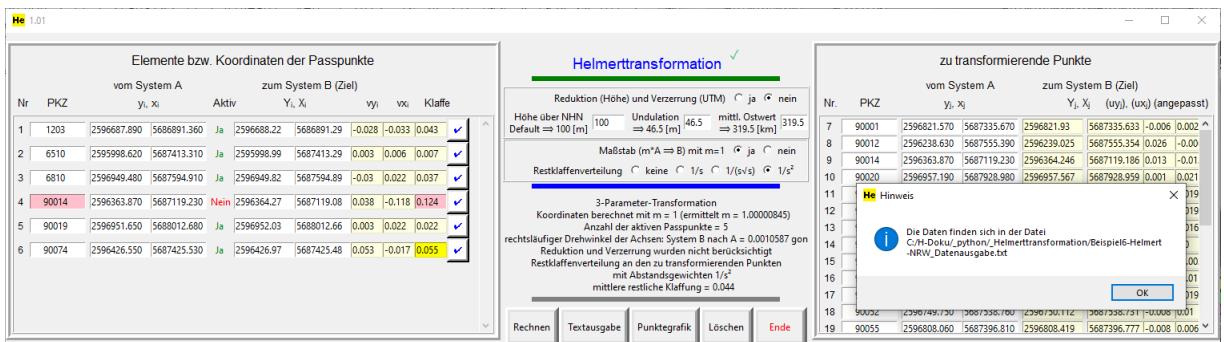
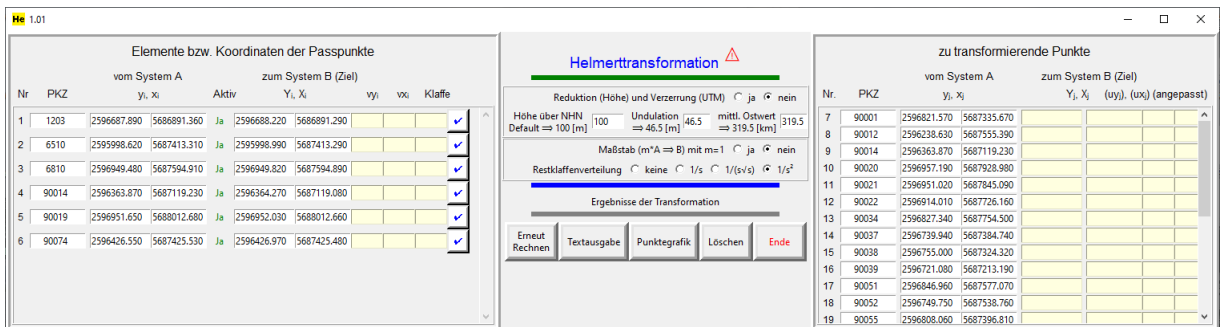
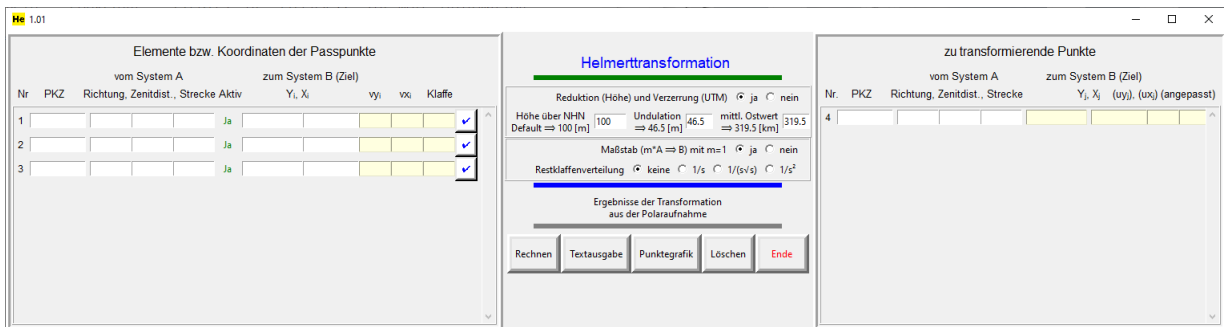
Bei einer erneuten Eingabe werden sie überlesen. Sie haben auf den neuen Berechnungsdurchlauf keinen Einfluss.

- C bzw. 99; oder Leerzeile (Kommentar, wird nicht bearbeitet)
- 02; Berechnungsdatum
- 11; PKZ; Ayi; Axi; BYi; BXi; vyi; vxi; Klaffe; (Passpunkte, System A, B)
- 21; PKZ; Ajj; Axj; BYj; BXj; uyj; uxj; (zu transformierende Punkte, System A, B [berechnet])
- 31; PKZ; Aylokal i; Axlokal i; BYi; BXi; vyi; vxi; Klaffe; (System A(lokal Kood., System B [berechnet])
- 41; PKZ; Aylokal j; Axlokal j; BYj; BXj; uyj; uxj; (System A(lokal Koord., System B [berechnet])

Die Eingabemaske ist in 3 Teile gegliedert. Links stehen die Elemente bzw. Koordinaten der Passpunkte, rechts die der umzuformenden Punkte. In der Mitte befinden sich die Voreingaben zur Art der Transformation, die anschließenden Ergebnisse und weiter unten die Buttons zur Wahl der Rechengänge und Ausgaben.

Abbildung 2

Oben: Leere Maske zur manuellen Eingabe - **Mitte:** Maske nach Einlesen einer Datei und Festlegung der Transformationsart zur weiteren manuellen Berichtigung/Ergänzung, Beispiel zu LVA NRW [5] S. 31-33 **Unten:** Abgeschlossener Berechnungsdurchlauf





Zur manuellen Eingabe:

Dezimalzahlen müssen durch einen Punkt und nicht mit einem Komma eingegeben werden (pythonspezifisch)². Die Eingabe von Buchstaben ist nicht erlaubt. Durch eine programmspezifische Ausnahmebehandlung würde eine Berechnung nur durchgeführt werden, wenn eine Umwandlung der Eingabe in Gleitkommazahlen (ohne Buchstaben, Komma oder sonstige Zeichen) möglich wäre. Ansonsten müssen Veränderungen in der Eingabe erfolgen. Bei der Dateneingabe über eine Datei können auch Kommas statt Punkte eingegeben werden; sie werden beim Einlesevorgang automatisch in Punkte umgewandelt. Leerfelder müssen besetzt werden (deshalb ist auch eine präzise Angabe der Passpunkte bzw. umzuformenden Punkte erforderlich).

Reduktions- und Korrektionswerte sind durch Defaultwerte in der Eingabe vorgegeben, die bei Notwendigkeit verändert werden können. Bei leeren Feldern für Zenitdistanzen bei Streckenreduktionen werden automatisch Winkel von 100 gon erzeugt.

Punktkennzeichen (PKZ) mit maximal 10 Zeichen sind alphanumerisch. Punkte können deaktiviert und wieder reaktiviert werden. Bei einem deaktivierten Punkt wird das Feld des PKZ rot hinterlegt. Bei einer Datenausgabe werden die Koordinaten dieses Punktes nicht gelöscht, sondern mit einer Codierung 99 versehen.

Durch den Button <Punktegrafik> kann zur besseren Übersicht und zum Verständnis der Eingabefelder eine wieder schließbare Zeichnung im Display geöffnet werden. Die Grafik sollte durch den unteren, roten Button geschlossen werden.

Ein gerade durchgeführter Berechnungsablauf wird neben der Titelüberschrift durch einen grünen Haken  angezeigt. Ein Warnzeichen  taucht bei einer nachträglichen Betätigung einer Taste auf. Dann ist ein erneuter Berechnungsablauf **stets erforderlich**, da sonst im Zusammenhang fehlerhafte, nicht zusammen gehörige Angaben im Display erscheinen können.

Blassgelbe Eingabefelder dienen **ausschließlich einer Anzeige**. Eingaben darin haben keinerlei Einfluss auf die Auswertung. Rot hinterlegte Felder deuten auf Fehler oder Abweichungen hin, die aber keinen Einfluss auf die Berechnungsergebnisse haben. Die größte Klaffe bei den aktiven Passpunkten wird in gelb angezeigt. Die Ausgabe erfolgt in [0,1 mgon] bzw. [mm], wobei rechtsseitige Nullen nicht angezeigt werden.

²Eine Einstellung des Punktes im Ziffernblock der Tastatur für Windows ist sinnvoll. Man wähle dazu in Windows: >Einstellungen > Zeit und Sprache > Sprache > Sprache hinzufügen > Deutsch-Optionen > Deutsch(Punkt). Hinweise im Internet, einen Punkt im Ziffernblock der Tastatur mit Tastatortreiber anderer Länder (z.B. Schweiz) erzeugen zu wollen, wechseln nach Tastendruck zwar das Komma in einen Punkt, haben aber die Eigenschaft andere Sonderzeichen der Tastatur zu verändern. Deshalb kann es sinnvoll sein, softwaremäßig ausschließlich nur das Komma im Ziffernblock zu ändern: Die herunter ladbare Software PatchKeyboard_6.1 vermag das und erzeugt einen Tastatortreiber <Deutsch(Punkt)>, der zu wählen wäre.

Elemente bzw. Koordinaten der Passpunkte										
Nr	PKZ	vom System A		Aktiv	zum System B (Ziel)		v_{y_i}	v_{x_i}	Klaffe	
		y_i, x_i	Y_i, X_i							
1	P1	20.03	30.72	Ja	413.6	377.6	0.005	-0.04	0.04	✓
2	P2	80.34	362.24	Ja	487.6	706.4	0.029	0.06	0.067	✓
3	P3	360.23	123.43	Ja	757.3	456.2	-0.002	0.024	0.024	✓
4	P4	300.07	351.23	Ja	706.6	686.2	-0.031	-0.045	0.054	✓

He Hinweis

Die Daten finden sich in der Datei
 C:/H-Doku/_python/_Helmertransformation/Beispiel1-Helmer
 -Benning_Datenausgabe.txt

OK

Abbildung 3

Helmertransformation ✓

Reduktion (Höhe) und Verzerrung (UTM) ja nein

Höhe über NHN Default ⇒ 100 [m] Undulation ⇒ 46.5 [m] mittl. Ostwert ⇒ 319.5 [km]

Maßstab ($m^*A \Rightarrow B$) mit $m=1$ ja nein

Restklaffenverteilung keine 1/s 1/(s√s) 1/s²

4-Parameter-Transformation, berechnet mit dem Maßstabsfaktor = 0.99988411
 Anzahl der aktiven Passpunkte = 4
 rechtsläufiger Drehwinkel der Achsen: System B nach A = 2.636528 gon
 Reduktion und Verzerrung wurden nicht berücksichtigt.
 Koordinaten berechnet mit $m = 0.99988411$
 keine Anpassung der Restklaffen an den zu transformierenden Punkten
 mittlere restliche Klaffung = 0.069

Rechnen Textausgabe Punktgrafik Löschen Ende

Abbildung 4

zu transformierende Punkte							
Nr.	PKZ	vom System A		zum System B (Ziel)			
		y_j, x_j	Y_j, X_j	(u_{y_j})	(u_{x_j})	(angepasst)	
5	S	190.1675	216.9050	591.275	556.6	0.0	0.0
6	N	180	200	580.418	540.132	0.0	0.0

Abbildung 5
 Beispiele aus BENNING
 [1]
 S. 264-266

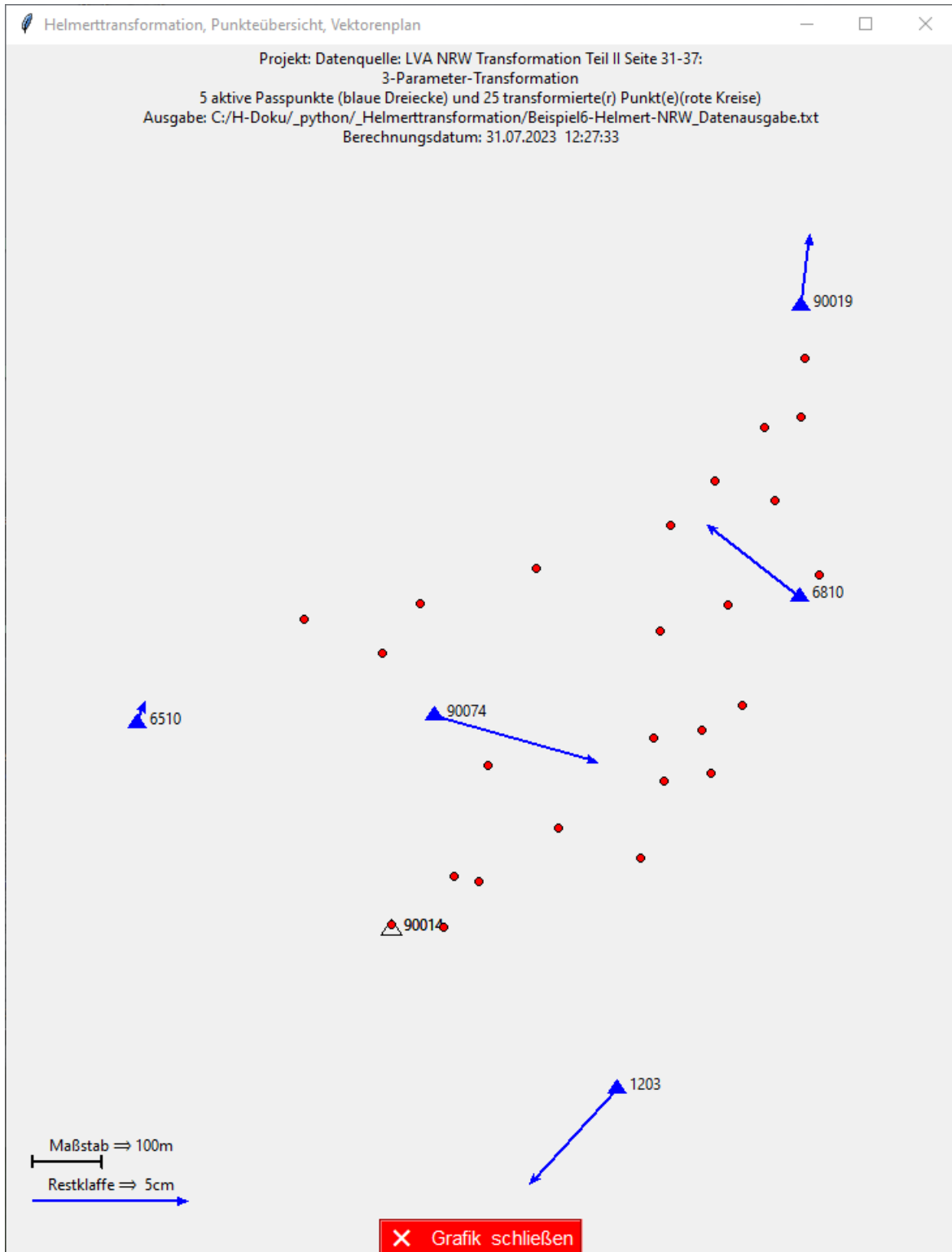


Abbildung 6
zu Beispiel aus LVA NRW [5] S. 34-37

Elemente bzw. Koordinaten der Passpunkte										
Nr	PKZ	vom System A			zum System B (Ziel)					
		y_i, x_i	Aktiv	Y_i, X_i	v_{y_i}	v_{x_i}	Klaffe			
1	1203	2596687.890	5686891.360	Nein	2596688.22	5686891.29	-0.027	-0.056	0.062	✓
2	6510	2595998.620	5687413.310	Ja	2595998.99	5687413.29	0.004	-0.007	0.008	✓
3	6810	2596949.480	5687594.910	Ja	2596949.82	5687594.89	-0.02	0.002	0.02	✓
4	90014	2596363.870	5687119.230	Nein	2596364.27	5687119.08	0.039	-0.136	0.142	✓
5	90019	2596951.650	5688012.680	Ja	2596952.03	5688012.66	0.017	0.005	0.017	✓
6	90074	2596426.550	5687425.530	Nein	2596426.97	5687425.48	0.057	-0.033	0.066	✓

Abbildung 7

4-Parameter-Transformation mit von 6 auf 3 aktive Passpunkte reduzierte Transformation, zu Beispiel aus LVA NRW [5] S. 34-37

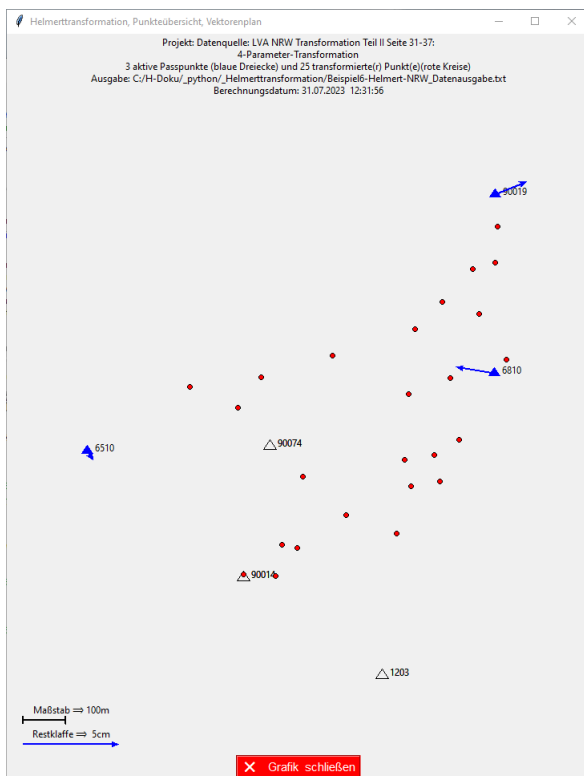


Abbildung 8

Grafik zum obigen Beispiel

Als extrapolierte Punkte (siehe [4]) in einer Helmertransformations werden die zu transformierenden Punkte bezeichnet, die außerhalb eines die Stützpunkte umhüllenden Polygons liegen. Deren Kennzeichnung ist wenig aussagekräftig, weil die Form des Stützpunktgebietes einen großen Einfluss auf das Ergebnis hat, außerdem werden Clusterbildungen nicht hinreichend erkannt.

Eine wichtige Kontrolle ist die Ansicht der Grafik der Punkte. Sind die Passpunkte gleichmäßig verteilt? Wurde eine Clusterbildung vermieden, insbesondere bei umzuformenden Punkten für die Restklaffenverteilung? Ist das Umformungsgebiet möglichst nicht langgestreckt?

Ausreißer bei den zu transformierenden Punkten werden einer Abstandskontrolle zum Schwerpunkt unterworfen; dies im Vergleich zum Streckenmedian der Passpunkte zu dessen Schwerpunkt.

Zur Fehlersuche wurde auf die Implementierung von Data-Snooping (1968) oder Cook-Abstand (1977) verzichtet, weil die Angaben der Restklaffen und die Möglichkeit der unmittelbaren Deaktivierung von Stützpunkten für eine Fehlersuche ausreichend erscheinen. Auch deshalb, weil hier keine hybriden Messungen vorliegen und keine a-priori-Genauigkeiten verwendet werden. Eine wichtige Kontrolle ist die grafische bzw. durch Augenschein, um auf eine gewogene, gut gestreute Punktverteilung zu achten. Die maximale Restklaffe wird in der Tabelle durch einen verstärkten Gelbton kenntlich gemacht. Eine Deaktivierung dieses Punktes mit einem anschließenden, erneutem Durchlauf mit den neu ermittelten Transformationsparametern der anderen Punkte zeigt die quasiwahre Restklaffe (in rotem Hintergrund) dieses auffälligen Punktes. Weitere Punkte können deaktiviert werden, aber auch die Zurücknahme einer Deaktivierung ist möglich.

Die Redundanzen zeigen bei einer Überbestimmung inwieweit die Eingangswerte kontrolliert sind. Geringe Redundanzwerte sprechen für eine geringe Kontrolle, höhere für eine bessere. Alle Redundanzen im gleichen Bereich sprechen für ein ausgewogenes Ergebnis. Die Redundanzen bei der Polaraufnahme beziehen sich auf die Koordinaten der in das örtliche, orthogonale System umgeformten Punkte. Alle haben demnach das Gewicht 1.

Die Aussage von Y_0 und X_0 ist gering, je nachdem ob man durch Konstanten gekürzte Koordinaten zur Transformation benützt oder nicht. Ein direkter Einfluss auf das Koordinatenergebnis besteht nicht.

Formelentwicklung

Streckenreduktion auf UTM-Bezugsebene:

$$S_{[\text{örtlich}]} = \frac{\sqrt{((Ost_2 - Ost_1)^2 + (Nord_2 - Nord_1)^2)} = S_{[UTM]}}{\left(1 + \frac{(Ost_{m[km]} - 500)^2}{2R_{[km]}^2} - \frac{h_{NHN[m]} + h_{Undulation[m]}}{R_{[km]} * 1000}\right) * 0.9996} = r$$

$Ost_i, Nord_i$ = UTM-Koordinaten

$S_{[\text{örtlich}]}, S_{[UTM]}$ = Strecke im örtlichen System, in UTM

r = Reduktionsfaktor

$R = 6381 [km]$, mittlerer Erdradius für die mittlere Breite in Deutschland

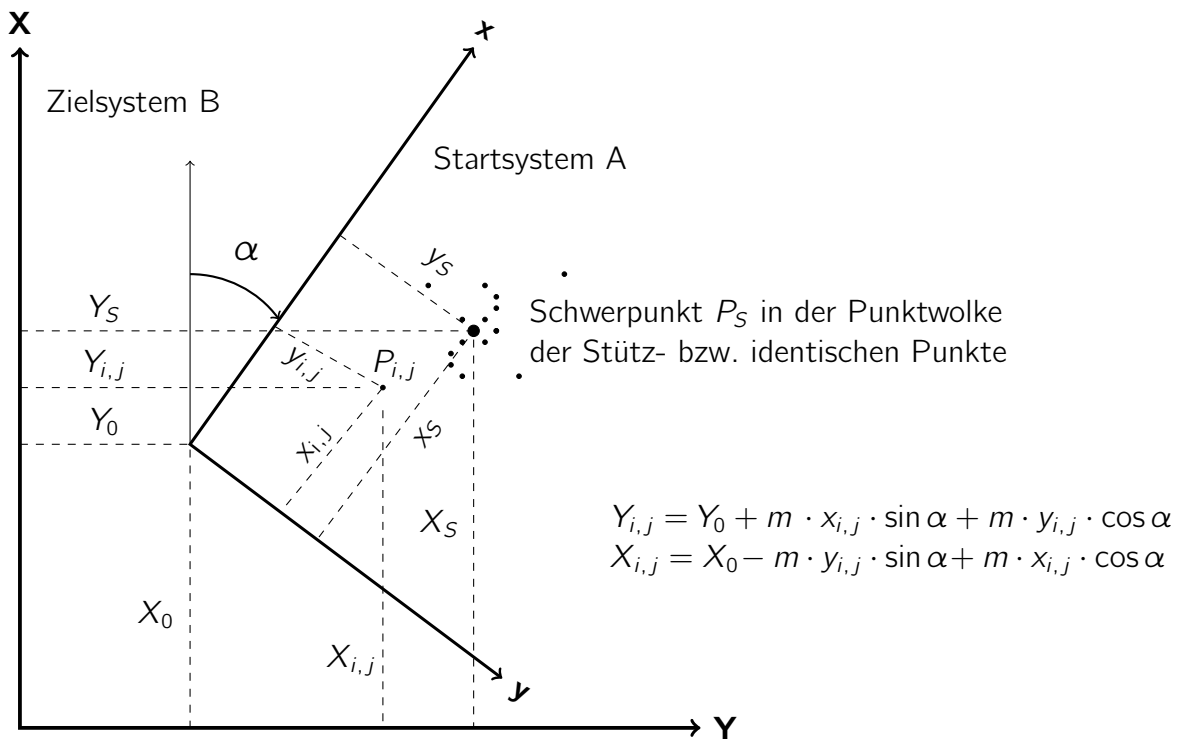
$S_{[\text{örtlich}]} \cdot r = S_{[UTM]}$

Bei vorgegebener polaren Aufnahme im System A erfolgt eine Umrechnung polarer in orthogonale Koordinaten mit:

$$y_i = s_i \cdot \sin(\text{Zenitdistanz}_i) \cdot \sin(\text{Richtung}_i)$$

$$x_i = s_i \cdot \sin(\text{Zenitdistanz}_i) \cdot \cos(\text{Richtung}_i)$$

Das System A ist das **A**usgangs- oder Startsystem, dass u. U. die ursprünglichen, umzuformenden Elemente oder Messergebnisse enthält. Das System B ist das Zielsystem, i.d.R. ein übergeordnetes System, in das hinein transformiert werden soll.



Notation³:

y, x	Startsystem A
Y, X	Zielsystem B
i	Punktindex Passpunkte
j	Punktindex umzuformende Punkte
\bar{y}	Mittelwert
\hat{y}	Schätzwert, berechneter Wert
y'	reduzierter Wert
y_S	Schwerpunkt, auf den Schwerpunkt bezogen
y_{iS}	lokaler, vom Punkt i auf den Schwerpunkt reduzierter Wert
m	Maßstab

³Die Ziffer **1** bzw. der Kleinbuchstabe **l** bzw. der Großbuchstabe **L** lassen sich vor allem im Druck nur schwer unterscheiden. Wird das nicht beachtet – besonders in Lehrbüchern – führt das beim Leser leicht zu Missverständnissen.

Falls erforderlich wird deshalb hier ein Einheitsvektor mit **e**, eine Einheitsmatrix mit **E**, eine Beobachtung mit **l**, ein Beobachtungsvektor mit **l**, eine Residue mit ε , ein Residuenvektor mit **ε** bezeichnet.

Auch die Vertauschung von X und Y in den Lehrbüchern erschweren das Verständnis. Chaotischer kann Lehre nicht sein!

<p>Benning [1] Syntax aus der Statistik</p> $y = X \cdot \beta + \varepsilon$	<p>Niemeier [2] Syntax aus der klassischen Ausgleichsrechnung</p> $l + v = A \cdot x$	<p>Helmerttransformation</p>
<p>Passpunkte im Startsystem A (Ausgang) = $P_i(y_i, x_i)$ und im Zielsystem B = $P_i(Y_i, X_i)$ und zu transformierende Punkte $P_j(y_j, x_j)$ in das Zielsystem B = $P_j(Y_j, X_j)$</p>		
<p>$Y_i = Y_0 + o \cdot x_i + a \cdot y_i$ (Ansatz Benning, 4-Parameter) $X_i = X_0 - o \cdot y_i + a \cdot x_i$</p> <div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px; width: 45%;"> $y = X \cdot \beta + \varepsilon$ </div> <div style="border: 1px solid black; padding: 5px; width: 45%;"> $l + v = A \cdot x$ </div> </div> <div style="display: flex; justify-content: space-around; margin-top: 20px;"> <div style="text-align: center;"> $y_{(2n,1)} = \begin{bmatrix} Y_1 \\ X_1 \\ \vdots \\ Y_n \\ X_n \end{bmatrix}, \beta = \begin{bmatrix} Y_0 \\ X_0 \\ o \\ a \end{bmatrix}$ </div> <div style="text-align: center;"> $A_{(2n,4)} = X = \begin{bmatrix} 1 & 0 & x_1 & y_1 \\ 0 & 1 & -y_1 & x_1 \\ 1 & 0 & x_i & y_i \\ 0 & 1 & -y_i & x_i \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & x_n & y_n \\ 0 & 1 & -y_n & x_n \end{bmatrix}$ </div> </div>		<p>Translations und Transformationsparameter Y_0, X_0, o, a</p> <p>$v_i = -\varepsilon_i$ bzw. $v = -\varepsilon$</p> <p>$i =$ Index Stützpunkt $j =$ Index umzuformender Punkt</p> <p>$A = X, \beta = x, y = l$</p> <p>$N = A^T A, n = A^T l$</p>
<p>Reduktion der Koordinaten auf die Schwerpunkte (Mittelwerte) der Passpunkte</p>		
$Y_S = \frac{1}{n} \sum_{i=1}^n Y_i, \quad X_S = \frac{1}{n} \sum_{i=1}^n X_i$ $y_S = \frac{1}{n} \sum_{i=1}^n y_i, \quad x_S = \frac{1}{n} \sum_{i=1}^n x_i$ $Y_S = Y_0 + a \cdot y_S + o \cdot x_S, \quad X_S = X_0 - o \cdot y_S + a \cdot x_S$ $Y_{iS} = Y_i - Y_S, \quad X_{iS} = X_i - X_S$ $y_{iS} = y_i - y_S, \quad x_{iS} = x_i - x_S$ $Y_i = Y_S + a \cdot y_{iS} + o \cdot x_{iS}$ $X_i = X_S - o \cdot y_{iS} + a \cdot x_{iS}$		<p>Schwerpunkte y_S, x_S, Y_S, X_S $n =$ Anzahl der aktiven Passpunkte</p> <p>Reduktionen auf den Schwerpunkt bezogen</p>
<div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px; width: 45%;"> $y = X_S \cdot \hat{\beta} + \varepsilon$ </div> <div style="border: 1px solid black; padding: 5px; width: 45%;"> $l + v = A_S \cdot \hat{x}$ </div> </div> <div style="display: flex; justify-content: space-around; margin-top: 20px;"> <div style="text-align: center;"> $\hat{\beta} = \begin{bmatrix} Y_S \\ X_S \\ o \\ a \end{bmatrix}$ </div> <div style="text-align: center;"> $A_S_{(2n,4)} = \begin{bmatrix} 1 & 0 & x_{1S} & y_{1S} \\ 0 & 1 & -y_{1S} & x_{1S} \\ 1 & 0 & x_{iS} & y_{iS} \\ 0 & 1 & -y_{iS} & x_{iS} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & x_{nS} & y_{nS} \\ 0 & 1 & -y_{nS} & x_{nS} \end{bmatrix}$ </div> </div>		<p>$v_i = -\varepsilon_i$ bzw. $v = -\varepsilon$</p> <p>$A_S = X_S, \hat{\beta} = \hat{x}, y = l$</p> <p>$N_S = A_S^T A_S, n_S = A_S^T l$</p> $N_S = \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & \Sigma & 0 \\ 0 & 0 & 0 & \Sigma \end{bmatrix}$ <p>wobei $\Sigma \hat{=} \sum (x_{iS}^2 + y_{iS}^2)$</p>

$$\hat{\mathbf{x}} = (\mathbf{A}_S^T \mathbf{A}_S)^{-1} \mathbf{A}_S^T \mathbf{l} + \mathbf{v}$$

$$\hat{\mathbf{x}} = \mathbf{N}_S^{-1} \mathbf{n}_S$$

$$\mathbf{v} = \mathbf{A}_S \hat{\mathbf{x}} - \mathbf{l}$$

Transformationsparameter

Restklaffen

Berechnung der Transformationsparameter

$$m = \text{Maßstab} = \frac{\text{Strecke (Zielsystem)}}{\text{Strecke (Startsystem)}}$$

$$\hat{o} = m \cdot \sin \alpha, \quad \hat{a} = m \cdot \cos \alpha, \quad m = \sqrt{\hat{o}^2 + \hat{a}^2}, \quad \alpha = \arctan \frac{\hat{o}}{\hat{a}}$$

α = Drehwinkel vom System B zum System A

α = Richtungswinkel der x-Achse (System A) gegen die X-Achse (System B)

$$\hat{o} = \frac{\sum_{i=1}^n (Y_{iS} * x_{iS}) - \sum_{i=1}^n (X_{iS} * y_{iS})}{\sum_{i=1}^n (y_{iS}^2 + x_{iS}^2)} \quad \hat{a} = \frac{\sum_{i=1}^n (Y_{iS} * y_{iS}) + \sum_{i=1}^n (X_{iS} * x_{iS})}{\sum_{i=1}^n (y_{iS}^2 + x_{iS}^2)}$$

$$o' = \frac{\hat{o}}{m} = \sin \alpha, \quad a' = \frac{\hat{a}}{m} = \cos \alpha$$

$$\hat{Y}_0 = Y_S - o \cdot x_S - a \cdot y_S$$

$$\hat{X}_0 = X_S + o \cdot y_S - a \cdot x_S$$

Translationsunbekannte Y_0, X_0

Die Translationsunbekannten werden ausgleichsunabhängig mitunter bei nahezu achsparallelen Systemen berechnet durch : $Y_0 = Y_S - y_S, \quad X_0 = X_S - x_S$ (so auch in [5])

Transformation, Restklaffen

$$\hat{Y}_i \text{ (gerechnet)} = \hat{Y}_S + \hat{a}y_i + \hat{o}x_i$$

$$\hat{X}_i \text{ (gerechnet)} = \hat{X}_S + \hat{a}x_i - \hat{o}y_i$$

$$\hat{Y}_i \text{ (gerechnet)} = \hat{Y}_S + a'y_i + o'x_i$$

$$\hat{X}_i \text{ (gerechnet)} = \hat{X}_S + a'x_i - o'y_i$$

$$\hat{v}_{yi} = Y_i - \hat{Y}_i \text{ (gerechnet)}$$

$$\hat{v}_{xi} = X_i - \hat{X}_i \text{ (gerechnet)}$$

$$\text{Klaffe} = \sqrt{\hat{v}_{yi}^2 + \hat{v}_{xi}^2}$$

4-Parameter-Transformation, konform

Koordinaten nach Transformation

3-Parameter-Transformation, isometrisch

Koordinaten nach Transformation

Restklaffen

Berechnung der Neupunktkoordinaten ins Zielsystem

$$\hat{Y}_j \text{ (gerechnet)} = \hat{Y}_S + \hat{a}y_j + \hat{o}x_j$$

$$\hat{X}_j \text{ (gerechnet)} = \hat{X}_S + \hat{a}x_j - \hat{o}y_j$$

$$\hat{Y}_j \text{ (gerechnet)} = \hat{Y}_S + a'y_j + o'x_j$$

$$\hat{X}_j \text{ (gerechnet)} = \hat{X}_S + a'x_j - o'y_j$$

4-Parameter-Transformation

3-Parameter-Transformation

Fehlerberechnung, Redundanzen

$$s_0^2 = \frac{\mathbf{v}^T \mathbf{v}}{2n-3}, \quad s_0^2 = \frac{\mathbf{v}^T \mathbf{v}}{2n-4}$$

$$\hat{s}_0 = \sqrt{\frac{\sum_{i=1}^n \hat{v}_{y_i}^2 + \sum_{i=1}^n \hat{v}_{x_i}^2}{2n-4}}$$

$$\hat{s}_0 = \sqrt{\frac{\sum_{i=1}^n \hat{v}_{y_i}^2 + \sum_{i=1}^n \hat{v}_{x_i}^2}{2n-3}}$$

$$\hat{m}_{P_i} = \hat{s}_0 \sqrt{2}$$

$$r_{y_i}, r_{x_i}[\%] = 100 \cdot \left(1 - \frac{1}{n} - \frac{x_{iS}^2 + y_{iS}^2}{\sum_{i=1}^n (x_{iS}^2 + y_{iS}^2)} \right)$$

$$r_{y_i}[\%] = 100 \cdot \left(1 - \frac{1}{n} - \frac{(x_{iS} \cdot \cos \alpha - y_{iS} \cdot \sin \alpha)^2}{\sum_{i=1}^n (x_{iS}^2 + y_{iS}^2)} \right)$$

$$r_{x_i}[\%] = 100 \cdot \left(1 - \frac{1}{n} - \frac{(y_{iS} \cdot \cos \alpha + x_{iS} \cdot \sin \alpha)^2}{\sum_{i=1}^n (x_{iS}^2 + y_{iS}^2)} \right)$$

3- und 4-Parameter, empirische Varianz der Gewichtseinheit

4-Parameter, empirische Standardabw. der Gewichtseinheit

3-Parameter, empirische Standardabw. der Gewichtseinheit

m_{P_i} = mittlere restliche Klaffung eines Punktes

Redundanzanteil in %, 4-Parameter

Redundanzanteil in %, 3-Parameter

Redundanzanteil in %, 3-Parameter

Gewichtete Restklaffenverteilung

$$p_{ij} = \frac{1}{s_{ij}} \text{ bzw. } p_{ij} = \frac{1}{s_{ij} * \sqrt{s_{ij}}} \text{ bzw. } p_{ij} = \frac{1}{s_{ij}^2}$$

$$u_{y_j} = \frac{1}{\sum p_{ij}} \sum_{i=1}^n (p_{ij} \cdot \hat{v}_{y_i})$$

$$u_{x_j} = \frac{1}{\sum p_{ij}} \sum_{i=1}^n (p_{ij} \cdot \hat{v}_{x_i})$$

$$u_{y_j}, u_{x_j} = 0 \text{ (keine Restklaffenanpassung)}$$

$$\hat{Y}_j(\text{endgültig}) = \hat{Y}_j(\text{gerechnet}) + u_{y_j}$$

$$\hat{X}_j(\text{endgültig}) = \hat{X}_j(\text{gerechnet}) + u_{x_j}$$

p_{ij} = Abstandsgewicht

s_{ij} = Strecke zwischen i und j

v = Restklaffe

u = Anpassungsbetrag

(Dieser Betrag ist in den endgültigen Koordinaten bereits enthalten)

Anhang – 3-Parameter Helmerttransformation (Redundanzen)

$$Y_i = Y_0 + \overbrace{x_i \cdot \sin \alpha + y_i \cdot \cos \alpha}^{a_i} = Y_0 + a_i \quad (\text{Transformationsgleichungen})$$

$$X_i = X_0 + \overbrace{-y_i \cdot \sin \alpha + x_i \cdot \cos \alpha}^{b_i} = X_0 + b_i \quad \text{Ansatz analog zu Benning [1] S. 260f}$$

$$i = 1, \dots, n$$

$$\frac{\delta Y_i}{\delta Y_0} = 1; \quad \frac{\delta Y_i}{\delta \alpha} = \frac{\delta a_i}{\delta \alpha} = x_i \cdot \cos \alpha - y_i \cdot \sin \alpha = c_i$$

$$\frac{\delta X_i}{\delta X_0} = 1; \quad \frac{\delta X_i}{\delta \alpha} = \frac{\delta b_i}{\delta \alpha} = -y_i \cdot \cos \alpha - x_i \cdot \sin \alpha = d_i$$

$$\mathbf{A} = \mathbf{X} = \begin{bmatrix} 1 & 0 & x_1 \cdot \cos \alpha - y_1 \cdot \sin \alpha \\ 0 & 1 & -y_1 \cdot \cos \alpha - x_1 \cdot \sin \alpha \\ 1 & 0 & c_1 \\ 0 & 1 & d_1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & c_n \\ 0 & 1 & d_n \end{bmatrix} \quad (\text{Funktionalmatrix})$$

$$\mathbf{A}^T = \begin{bmatrix} 1 & 0 & \dots \\ 0 & 1 & \dots \\ c_1 & d_1 & \dots \end{bmatrix} \begin{bmatrix} n \\ n \\ \sum_{i=1}^n (c_i^2 + d_i^2) \end{bmatrix} = \mathbf{A}^T \mathbf{A} = \mathbf{N}$$

(Normalgleichungsmatrix)

$$h = \sum_{i=1}^n (c_i^2 + d_i^2) = \sum_{i=1}^n (x_i^2 + y_i^2)$$

$$\mathbf{N}^{-1} = \begin{bmatrix} 1/n & & \\ & 1/n & \\ & & 1/h \end{bmatrix} = \text{diag} \{1/n, 1/n, 1/h\}$$

$$\mathbf{H} = \mathbf{A}(\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T = \mathbf{A} \mathbf{N}^{-1} \mathbf{A}^T \quad (\text{Projektionsmatrix})$$

$$\mathbf{N}^{-1} = \begin{bmatrix} 1/n & & \\ & 1/n & \\ & & 1/h \end{bmatrix} \begin{bmatrix} 1 & 0 & \dots \\ 0 & 1 & \dots \\ c_1 & d_1 & \dots \end{bmatrix} = \mathbf{A}^T$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & c_1 \\ 0 & 1 & d_1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & c_n \\ 0 & 1 & d_n \end{bmatrix} \begin{bmatrix} 1/n & 0 & c_1/h \\ 0 & 1/n & d_1/h \\ \vdots & \vdots & \vdots \\ 1/n & 0 & c_n/h \\ 0 & 1/n & d_n/h \end{bmatrix} \begin{bmatrix} \frac{1}{n} + \frac{c_1^2}{h} & \dots & \dots \\ \dots & \frac{1}{n} + \frac{d_1^2}{h} & \dots \\ \vdots & \vdots & \vdots \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{bmatrix} = \mathbf{H}$$

$$r_{ii} = 1 - h_{ii}, \quad r_{yi} = 1 - \left(\frac{1}{n} + \frac{c_i^2}{h}\right), \quad r_{xi} = 1 - \left(\frac{1}{n} + \frac{d_i^2}{h}\right)$$

$$r_{y_i} [\%] = 100 \cdot \left(1 - \frac{1}{n} - \frac{(x_i \cdot \cos \alpha - y_i \cdot \sin \alpha)^2}{\sum_{i=1}^n (x_i^2 + y_i^2)}\right) \quad (\text{Redundanzanteile in } \%)$$

$$r_{x_i} [\%] = 100 \cdot \left(1 - \frac{1}{n} - \frac{(y_i \cdot \cos \alpha + x_i \cdot \sin \alpha)^2}{\sum_{i=1}^n (x_i^2 + y_i^2)}\right)$$

Ausgabe, Beispieldaten und -ausdrucke

Die Daten- bzw. die Druckausgaben finden sich im gleichen Verzeichnis, in dem auch die Eingabedaten (xxx.txt) vorhanden sind. Sie haben jeweils die Ergänzung xxx_Datenausgabe.txt bzw. xxx_Printausgabe.txt. Bei manueller Eingabe ist die Ausgabe in C:/tmp/_Datenausgabe.txt bzw. C:/tmp/_Printausgabe.txt.

Beispiel einer erzeugten Datenausgabe, die wiederum auch editiert zur Dateneingabe verwendet werden kann:

```
C ***** Eingabedaten (manuell oder ursprünglich aus einer Datei) *****
01;Projekt: Beispieldaten zur Helmertransformations. 4
02;Berechnungsdatum: 31.07.2023 12:40:29
99 Lagesystem: ETRS89 (489); Bezugsellipsoid: GRS80; Abbildung: UTM
99 Undulation = 46.5 m, mittl. Höhe ü.NHN = 49.5 m, mittl. UTM-Ostwert[km] = 360
99 Reduktion und Verzerrung (r = 0.99982555) wurden berücksichtigt
C Code 30: Polare bzw. orthogonale Passpunkte
C Code; PKZ; ARichtung i; AZenitdistanz i; AStrecke i; BYi; BXi;
30; 104569; 0.0;100.0777; 91.857;2360502.036;5649727.404;
30; 104778; 11.4171;100.4669; 31.551; 2360466.71; 5649677.6;
30; 104207;103.009 ;102.3127; 6.677; 2360447.08;5649653.125;
30; 104779;192.3501;101.0408; 76.459;2360399.768;5649594.142;
99
C Code 40: Polare zu transformierende Punkte
C Code; PKZ; ARichtung j; AZenitdistanz j; AStrecke j;
40; 2000;300.5159;100.5466; 64.418;
40; 104569;399.9989;100.0784; 91.857;
99
99 ***** Ergebnisse der Transformation *****
99
99 4-Parameter-Transformation
99 Koordinaten berechnet mit m = 1.00003529
99 Anzahl der aktiven Passpunkte = 4
99 rechtsläufiger Drehwinkel der Achsen: System B nach A = 45.144445 gon
99 Standpunktkoordinaten im System B: Y = 2360442.228, X = 5649657.703
99 Reduktion und Verzerrung (r = 0.99982555) wurden berücksichtigt
99 keine Anpassung der Restklaffen an den zu transformierenden Punkten
99 Standardabweichung der Gewichtseinheit = 0.003
99 mittlere restliche Klaffung = 0.004
99
C Code 31: Polare bzw. orthogonale Passpunkte, nach der Transformation
C Code; PKZ; Aylokal i; Axlokal i; BYi; BXi; vyi; vxi; Klaffe;
31; 104569; -5.363; 80.171; 2360502.036; 5649727.404; 0.001; -0.003; 0.003;
31; 104778; 0.264; 19.369; 2360466.71; 5649677.6; -0.001; 0.004; 0.004;
31; 104207; 1.301; -11.985; 2360447.08; 5649653.125; -0.001; 0.000; 0.001;
31; 104779; 3.799; -87.554; 2360399.768; 5649594.142; 0.001; -0.002; 0.002;
99
C Code 41: Polare zu transformierende Punkte, nach der Transformation
C Code; PKZ; Aylokal j; Axlokal j; BYj; BXj; uyj; uxj(angepasst);
41; 2000; -69.766; -11.148; 2360393.690; 5649700.038; 0.000; 0.000;
41; 104569; -5.365; 80.171; 2360502.034; 5649727.408; 0.000; 0.000;
99
```


Printbeispiel mit Daten nach [5] Seiten 31, 32:

*** Helmerttransformation ***

Seite 1

Projekt: Datenquelle: LVA NRW Transformation Teil II Seite 31-37:
 Berechnungsdatum: 31.07.2023

3-Parameter-Transformation

Koordinaten berechnet mit $m = 1$ (ermittelt $m = 1.00000845$)
 Restklaffenverteilung an den zu transformierenden Punkten
 mit Abstandsgewichten $1/s^2$
 Standardabweichung der Gewichtseinheit = 0.031
 mittlere restliche Klaffung = 0.044

Auf die Stützpunkte bezogen:

mittlere Streckenlänge zum Schwerpunkt des lokalen Systems A = 477.357
 maximale Streckenlänge zum Schwerpunkt des lokalen Systems A = 647.169
 Median der Streckenlängen zum Schwerpunkt des lokalen Systems A = 475.868

Auf die zu transformierenden Punkte bezogen:

maximale Streckenlänge zum Schwerpunkt des lokalen Systems A = 581.787
 Der nach <ex> angezeigte Wert ist das Verhältnis von Strecke/Median System A
 wenn es größer als 1,5 ist.

rechtsläufiger Drehwinkel der Achsen: System B nach A = 0.0010587 gon
 Schwerpunkt der Passpunkte, System A, $y_s = 2596602.838$, $x_s = 5687467.558$
 Schwerpunkt der Passpunkte, System B, $Y_s = 2596603.206$, $X_s = 5687467.522$
 Translationsunbekannte $Y_0 = -116.152$, $X_0 = -4.903$

Transformation der Elemente bzw. Koordinaten der gegebenen Passpunkte

	vom System A		zum System B (Ziel)				
Punktkennzeichen	y _i	x _i	Y _i	X _i	v _{yi}	v _{xi}	Klaffe
1203	2596687.890	5686891.360	2596688.220	5686891.290	-0.028	-0.033	0.043
					ry= 54%	rx= 79%	
6510	2595998.620	5687413.310	2595998.990	5687413.290	0.003	0.006	0.007
					ry= 80%	rx= 52%	
6810	2596949.480	5687594.910	2596949.820	5687594.890	-0.030	0.022	0.037
					ry= 79%	rx= 71%	
90019	2596951.650	5688012.680	2596952.030	5688012.660	0.003	0.022	0.022
					ry= 57%	rx= 71%	
90074	2596426.550	5687425.530	2596426.970	5687425.480	0.053	-0.017	0.055
					ry= 80%	rx= 78%	

zu transformierende Punkte des Systems A

	vom System A		zum System B (Ziel, berechnet)			
Punktkennzeichen	y _j	x _j	Y _j	X _j	u _{yj} , u _{xj} (angepasst)	
90001	2596821.570	5687335.670	2596821.930	5687335.633	-0.006	0.002
90012	2596238.630	5687555.390	2596239.025	5687555.354	0.026	-0.006
90014	2596363.870	5687119.230	2596364.246	5687119.186	0.013	-0.012
90020	2596957.190	5687928.980	2596957.567	5687928.959	0.001	0.021
90021	2596951.020	5687845.090	2596951.389	5687845.067	-0.005	0.019
90022	2596914.010	5687726.160	2596914.362	5687726.138	-0.021	0.019
90034	2596827.340	5687754.500	2596827.700	5687754.477	-0.013	0.016
90037	2596739.940	5687384.740	2596740.309	5687384.702	0.003	0.000

usw.

The image shows two side-by-side screenshots of a software interface for Helmert transformations. Both windows have a title bar 'Helmerttransformation' with a green checkmark. The left window is titled '3-Parameter-Transformation' and the right is '4-Parameter-Transformation'. Both have identical input fields: 'Reduktion (Höhe) und Verzerrung (UTM)' with radio buttons for 'ja' and 'nein' (selected), 'Höhe über NHN' (100), 'Undulation' (46.5), and 'mittl. Ostwert' (319.5). Below these are 'Maßstab (m*A => B) mit m=1' with radio buttons for 'ja' and 'nein' (selected), and 'Restklaffenverteilung' with radio buttons for 'keine', '1/s', '1/(s\sqrt{s})', and '1/s^2' (selected). The 3-parameter window shows 'Koordinaten berechnet mit m = 1 (ermittelt m = 1.00000845)', 'Anzahl der aktiven Passpunkte = 5', 'rechtsläufiger Drehwinkel der Achsen: System B nach A = 0.0010587 gon', and 'mittlere restliche Klaffung = 0.044'. The 4-parameter window shows the same parameters but with 'mittlere restliche Klaffung = 0.047'. Both windows have a bottom row of buttons: 'Rechnen', 'Textausgabe', 'Punktgrafik', 'Löschen', and 'Ende'.

Abbildung 9

Ergebnisbeispiel mit Daten aus LVA NRW [5] Seite 31-37

Schlussbemerkungen

Der Quellcode dieses Programmes wird neben der compilierten Fassung ebenfalls zur Verfügung gestellt. Es soll vor allem den vermessungstechnischen oder geodätischen Nachwuchs dazu anregen, sich durchaus mit einem komplexeren Programm auseinander zu setzen. Man kann alles immer besser machen. Eine Haftung und Gewährleistung kann nicht übernommen werden. Wenn man selber programmieren möchte, sind einige Passagen des Programms als Hilfestellungen sicherlich nützlich. Auch sollte man sich nicht scheuen, Python-Erweiterungen wie <NumPy> zu nutzen.

Auch eine Nachfrage bei ChatGPD bietet zumindest eine Lösungsrichtung; nicht alle Vorschläge sind sofort brauchbar. Da ChatGPD von Programmierern geschaffen sein muss, sind dort Fragen und Antworten zur Programmierung erstaunlich gut vertreten.

Python ist eigentlich leicht zu erlernen – eine gewisse Geduld vorausgesetzt –, hat aber als ältere Sprache auch seine zeit- und entwicklungsabhängige Tücken, was sich in der nicht einheitlichen, verschiedenfachen Syntax ausdrückt.

Liste der Beispieldaten

Daten für Eingabe und Printdateien:

- Beispiel 1: orthogonal, entnommen aus BENNING [1], Statistik..., 3. Auflage, S. 264
- Beispiel 2: orthogonal, entnommen aus Formelsammlung Nds 2012 S. 79, 89
- Beispiel 3: orthogonal, entnommen aus NIEMEYER [2]. Ausgleichsrechnung, 2. Aufl. S. 373
- Beispiel 4: orthogonales Extrembeispiel; Polaraufnahme
- Beispiel 5: orthogonales und polares Extrembeispiel
- Beispiel 6: orthogonal, entnommen aus Transformation von Koordinaten und Höhen in der Landesvermessung - Teil II: Praktische Anwendungsfälle, Seite 31-37

Literatur

- [1] BENNING, Statistik in Geodäsie, Geoinformation und Bauwesen, Verlag Wichmann, Heidelberg, 3. Aufl. 2010
- [2] NIEMEYER, Ausgleichsrechnung – Statistische Auswertemethoden, Verlag Walter de Gruyter, 2. Aufl. 2008
- [3] Lenkungsgruppe AFIS–ALKIS–ATKIS Niedersachsen, Formelsammlung unter Berücksichtigung des amtlichen Bezugssystems ETRS89 mit UTM-Abbildung – Entwurf –, Landesbetrieb Landesvermessung und Geobasisinformation Niedersachsen (LGN), Stand: 1.7.2012 [Nds], Seiten 34, 35 und 42
<https://docplayer.org/24975654-Formelsammlung-unter-beruecksichtigung-des-amtlichen-bezugssystems-etr89-mit-utm-abbildung.html>
- [4] Transformation von Koordinaten und Höhen in der Landesvermessung Teil 1: Theoretische Grundlagen LVA NRW, 2. Aufl. 1999, Seiten 51, 52
<http://nbn-resolving.de/urn:nbn:de:hbz:5:2-958>
- [5] Transformation von Koordinaten und Höhen in der Landesvermessung Teil 2: Praktische Anwendungsfälle LVA NRW, 1. Aufl. 1999
<http://nbn-resolving.de/urn:nbn:de:hbz:5:2-973>