
Projekt Romerschiff Documentation

Release 1.0

Moritz Guenther

August 16, 2016

1	Auswertung	3
1.1	Auswertung	3
1.2	Übersichtskarten und Fahrtstrecken	8
1.3	Software documentation	9
2	Index und Suche / Index and search	15
	Bibliography	17
	Python Module Index	19

Note:

English This website contains both German and English content. While the software is documented in English as it is customary in programming, the analysis and the results are mostly show in German.

Deutsch Diese Webseite enthält Inhalte in deutscher und in englischer Sprache. Der Programmcode und die zugehörige Dokumentation sind in Englisch verfasst, die Auswertung der Ergebnisse ist auf Deutsch präsentiert.

Auswertung

1.1 Auswertung

1.1.1 Der Einfluss von Messunsicherheiten auf die Auswertung

Folgend gibt es Überlegungen zu den Fehlerquellen und deren Einfluss auf die Messungen mit dem NX2 bzw. auf die Auswertung. Dieser Text ist nicht in sich vollständig, sondern eine Ergänzung zu dem Kapitel „Technische Auswertung“ aus dem Buch „Lusoria Rhenana“ vom Verlag XYZ.

LOG

Das Log misst in guter Näherung zwischen 0,5 und 7kn linear, so dass eine Kalibrierung bei z.B. 3kn auch die anderen Geschwindigkeiten adequat beschreibt. Einzig bei Geschwindigkeit kleiner als 0,2-0,3kn reißt das Messsignal des Logs ab, da das Rädchen dann nicht mehr schnell genug angeströmt wird. Ruderschläge erzeugen natürlich nur in der Zugphase der Riemen Vortrieb, danach folgt 1-2 Sekunden eine Verlangsamung des Schiffes. Diese Spitzen bzw. Minima sind allerdings klein gegen die durchschnittliche Geschwindigkeit und mitteln sich über 1-2 Minuten annähernd weg. Die gleichmäßige Antriebskraft durch Ruderschläge ist subjektiv durch die Schlagmänner, die den Takt vorgeben und die übrigen Ruderer beeinflusst. Hin- und Rückweg einer Kalibrierungsfahrt wurden daher kurz hintereinander ohne eine Änderung der Sitzpositionen durchgeführt. Eine systematische Beeinflussung, wenn z.B. die gleiche Schlagfrequenz durch mehr Kraftaufwand bei Gegenwind / Strömung angestrebt wird, ist nicht auszuschließen. Daher wurden mehrere Kalibrierungsfahrten mit unterschiedlichen Schlagmännern, Strecken und insbesondere bei milden Konditionen unternommen. Es gibt einen zeitlichen Versatz der Geschwindigkeiten zum GPS (siehe dort), welcher aber aufgrund der gleichmäßigen Bewegung während der Kalibrierung keine Rolle spielt. Ein unterschiedlicher Anströmwinkel des Logs ergibt sich nur bei variierender Beladung des Schiffes, da die Holzkonstruktion der Halterung mit einem “Schuh” versehen war, die nur eine Einbauposition über dem Bugsporn zulässt. Im Fall der Testwoche in 2012 wurde darüber hinaus das Log auch über Nacht die ganze Zeit fest montiert im Boot belassen.

Kontrolle der Log-Kalibration

Um den Wert der oben berechneten Proportionalitätskonstanten zu überprüfen und sicher zu stellen, dass das Log linear reagiert, vergleichen wir nun die BSP und die SOG für einen wesentlich größeren Zeitraum. Dafür wählen wir alle Messpunkte aus, die folgende drei Bedingungen erfüllen: Die BSP ist positiv, das Boot also in Fahrt. Für sehr kleine Geschwindigkeiten (unter 0,4 kn) reagiert das Log noch nicht, deshalb müssen diese Zeiten herausgefiltert werden. Der Kompasskurs und die Bewegungsrichtung weichen um maximal 15° voneinander ab. Bei starken Drift, wie sie z.B. entstehen kann, wenn hart am Wind gesegelt wird, unterscheiden sich BSP und SOG schon dadurch, dass die SOG die gesamte Bewegung des Schiffes darstellt, das Log aber nur die Bewegungskomponente parallel zum Kiel misst. Die Geschwindigkeit ändert sich nur wenig. Die SOG wird aus der Differenz der GPS Position momentan und wenige Sekunden zuvor berechnet. Diese Art der Differenzbildung führt zu einer Verzögerung von einigen Sekunden

bevor eine veränderte Geschwindigkeit sich in der SOG niederschlägt. Selbstverständlich enthält dieser Datensatz auch die Punkte aus der automatischen Logkalibration.

Windmesser

Es ist zwischen Fehlern im gemessenen Windwinkel und der Windstärke zu unterscheiden. Der Windmesser selbst hat von Herstellerseite einige bekannte Offsets, welche in das NX2 Race zur Korrektur eingegeben wurde. Der größte Fehler beruht daher im Einbau des Windmessers. Dieser wurde jeden Tag neu befestigt und entlang des Vorderstevens ausgerichtet. Am Ende des Tages wurde der Einbau zudem kontrolliert, um nachträgliche Veränderungen zu bemerken bzw. ausschließen zu können. Dabei stellte sich allerdings heraus, dass die ganze Konstruktion sehr robust war, so dass hier eine Abweichung von mehr als 1-2° nicht zu erwarten ist. Ein Stampfen des Schiffes oder eine Seitenlage beim Segeln, d.h. eine Veränderung der Drehachse des Windmessers, könnte zwar ebenfalls die Winkelmessung beeinflusst haben, doch zeigte sich die Konstruktion des Twin Fin Windmesser bereits bei Windgeschwindigkeiten von 1-2 kn als wenig anfällig, so dass dieser Fehler in der Größenordnung des vorherigen liegen sollte. Bei kleineren Windstärken oder sogar Windstille ist die Messung zudem ohne Bedeutung, da keine Segelwirkung erzielt wird. Das Problem mit der Messung der Windstärke liegt weniger in dem Wert an sich als darin die Wirkung des Windes auf das Schiff zu erfassen. Besonders bei böigem Wind reagiert das Schiff zu träge, um die Bootsgeschwindigkeit direkt zu korrelieren. Bei Momentaufnahmen würde somit das Schiff fast still liegen, obwohl gerade ein starker Windstoß gemessen wird oder aber das Schiff hätte noch Vortrieb während der Wind (z.B. in der Nähe von Bäumen) plötzlich abreißt. Hierfür wurde der Wind mit einem Gauß-Profil geglättet, um eine durchschnittliche Windstärke über einen längeren Zeitraum zu erhalten. Dieses Profil ist verschoben, denn natürlich ist nur der Wind der vorangegangenen Sekunden maßgeblich für die momentane Geschwindigkeit des Schiffes.

Kompass

Die Ausrichtung des Kompasses senkrecht zum Kiel stellte sicher, dass es keine einbaubedingten Offsets gab. Ebenso konnten an dieser Einbauposition Einflüsse durch Magnetfelder ausgeschlossen werden, da hier weder Eisen noch Batterien in der unmittelbaren Nähe vorhanden waren. Bei den Schiffen selbst handelt es sich zudem ja um Holzschiffe, in denen es nur an kritischen Stellen Eisennägel gibt. Die ausgewählten Stellen zur Befestigung des Kompasses besitzen selbst nur Holzverbindungen und ein wenig Abstand zu Eisennägeln. Die Befestigung selbst wurde mit Klebeband realisiert. Der schwierigste Teil der Kalibrierung ist wie üblich trotz Wind und Strömung mit dem Ruderantrieb einen gleichmäßigen Kreisradius einzuhalten. Im Rahmen der Messgenauigkeit wurden keine Korrekturen vorgenommen. Allerdings wurde auch bei einer Testfahrt auf einer Segeljacht mit dem NX2 bei guten Konditionen (geschütztes Hafenbecken, Motor mit konstanter Leistung) bei der ersten Kalibrierung keine nennenswerte Abweichung gefunden, so dass über die ganzen Messfahrten der Korrekturfaktor der aller ersten Kalibrierungsfahrt unter ebenso günstigen Bedingungen im Yachthafen in Wedel 2006 beibehalten wurde. Eine größere Abweichung zwischen den Gewässern der Testfahrten durch den Unterschied von geographischen zu magnetischen Nordpol gab es nicht.

GPS

Bei einer unmittelbaren Auswertung der Anzeigen Kompasskurs und GPS Kurs bleibt jedoch das Problem, dass es eine Zeitversetzung gibt. Die Kursrichtung durch zwei von einander getrennten GPS Positionsbestimmungen ist erst 2-3 s verzögert verfügbar. Dies ist auch beim Vergleich von Geschwindigkeit im Wasser durch das Log und Geschwindigkeit über Grund durch das GPS beobachtbar. Das GPS ist außerdem träge. Um die Ungenauigkeit der Ortsbestimmungen (95% der Zeit sind diese auf 10 m genau) und gelegentlich Ausreißer zu kompensieren, wird über die vorangegangenen Geschwindigkeiten der letzten Sekunden gemittelt. Bei Geschwindigkeitsfahrten werden daher die absoluten Spitzenwerte am Ende eines einzelnen Riemenzuges nur im Log aufgezeichnet, während das GPS diese Ausreißer mit in die Durchschnittsgeschwindigkeit einrechnet. Die Abweichungen betragen aber normalerweise maximal 0,1 bis 0,2 kts

Server

Die Messdaten können aus dem Binärformat in ein CSV (column separated values) Zwischenformat exportiert werden, welches sich mit jedem handelsüblichen Editor oder Tabellenkalkulationsprogramm öffnen lässt. Die Anzahl der gültigen Stellen ist hierbei jedoch von Herstellerseite vorgegeben. Während eigentlich alle Winkel und Geschwindigkeiten mit vier bzw. zwei Nachkommastellen mehr als ausreichend sind, ergeben sich bei der Positionsbestimmung durch das GPS mittels Longitude und Latitude Koordinaten mit sechs gültigen Ziffern leichte Sprünge wenn man den gefahrenen Kurs über eine Karte plottet. Dies führt beim Projizieren des gefahrenen Kurses auf elektronisches Kartenmaterial zu "Treppenstufen".

Polardiagramm

Wie bereits angedeutet wird bei der Erstellung eines Polardiagramms bei diversen Größen gemittelt, hierbei verschmiert das Ergebnis natürlich ein wenig. Angenommen es gäbe nur genau zwei Messpunkte in dem Intervall bei 3 kn und 150°, einer davon bei einer Windstärke von 3,5 kn mit 155° und der andere nur bei 2,5 kn mit 145° aufgetragen, dann wäre es nicht verwunderlich, wenn ersterer eine höhere Bootsgeschwindigkeit ergibt als letzterer. Auch wenn der Zusammenhang nicht linear ist, dürfte das mittlere Ergebnis in diesem Beispiel durchaus dicht am eigentlich gewünschten Wert von 3 kn mit 150° liegen. Aber was wenn nun einer der beiden Werte fehlt? Dann wird dieser Fehler für dieses eine Intervall maximal. Aus diesem Grund gibt es zu jedem Polardiagramm noch ein Histogramm, welches die Anzahl der Messwerte wiedergibt und somit die Aussagekraft in dem jeweiligen Bin anzeigt. Bei der Erstellung des Polardiagrammes wird zudem vorher gefiltert. Da das Schiff aufgrund seines Gewichtes träge ist und das Segel optimal ausgerichtet werden muss, dauert es jeweils eine Weile bis das Schiff bei konstantem Kurs und konstanter Windrichtung und Windstärke seine optimale Endgeschwindigkeit erreicht. Es ist leicht einzusehen, dass man innerhalb dieser Zeit keine drastischen Änderungen haben möchte, sondern warten muss bis tatsächlich die Endgeschwindigkeit erreicht wurde. Ansonsten könnte der Skipper das Schiff aus voller Fahrt gegen die Strömung und gegen den Wind drehen und würde im ersten Moment noch Vortrieb messen bis das Schiff aufgestoppt hat und danach wieder zurückgetrieben wird. Ebenso könnte ein plötzlich aufkommender starker Wind bei einem treibenden Schiff im ersten Augenblick den Eindruck erwecken, dass dieses weiterhin treibt und gar nicht auf den Wind reagiert. Das ist mitnichten so und bereits das Schanzkleid bietet eine große Angriffsfläche. Bevor das Polardiagramm auf eine Seite gefaltet wird, kann der Unterschied in den Schmetterlingsflügeln auf Back- und Steuerbordseite als Qualität der Messwerte und der Filterung angesehen werden. Mit genügend Messpunkten und starker Filterung müssten beide Seiten identisch sein, da auch die Schiffe in sehr guter Näherung symmetrisch gebaut sind.

Drift

Bei achterlichem Wind und ohne Strömung gibt es keinen seitlichen Versatz, daher ist diese Windstellung ein Maß für die Genauigkeit der Messung. Das NX2 ist vergesslich, d.h. ohne einen angeschlossenen Computer mit Festplatte werden die aktuellen Daten aufbereitet und kurz im Display angezeigt. Dabei ist davon auszugehen, dass die Trägheit des GPS, Geschwindigkeiten erst nach einer zweiten Ortsbestimmung, also mit 3-4 Sekunden Verzögerung, auszurechnen, eine Ungenauigkeit bei schnell und eng gefahrenen Kurven darstellt, denn die Log und Kompassdaten sind weniger alt.

Ruderschlagliste

Der Protokollant wurde häufiger gewechselt, z.T. war auch niemand verfügbar, so dass die Liste von der Technik nebenher geführt wurde. Es konnte bereits während der Fahrt beobachtet werden, dass die Qualität sehr stark personenbezogen war. Bei längeren eintönigen Strecken, aber auch bei waghalsigen Manövern kam es immer wieder vor, dass nach einer vollen Minute zu spät in die nächste Zeile gewechselt wurde. Außerdem wurden Manöver nicht immer kleinteilig notiert. Bei der parallelen Führung durch die Technik musste zudem teilweise im Kopf mitgezählt werden und nachgetragen werden, wenn gerade andere Handgriffe zu tun waren. Auch wenn es theoretisch möglich ist kurze Geschwindigkeitszuwächse im Wasser, die durch das Log gemessen werden, als Zugphase beim Rudern zu identifizieren, wurde auf einen solchen Vergleich verzichtet. Die Segelstart-/endzeiten sollten hingegen akkurat sein, da sie

auch im Log des NX2 verzeichnet wurden. Ebenso wurde bei Geschwindigkeitsfahrten auf eine höhere Genauigkeit geachtet. Um Manöver für eine spätere Auswertung wiederzufinden, ist keine sekundengenaue Übersicht erforderlich.

Testgewässer

Die Versuchsgewässer waren in ihren Eigenschaften unterschiedlich, was insbesondere einen Vergleich der beiden Navis Lusoriae erschwert. Während bereits 2005 und 2006 mit der Regina auf der Naab und der Donau in Regensburg getestet wurde, waren die Altrheinarme in Wörth am Rhein und Germersheim bei der Rhenana 2011 und 2012 eher langsam fließende Gewässer. Insbesondere die Segelstrecken waren auf der Donau durch die Strömung, den Wind, die Flussufer und die Berufsschiffahrt vorgegeben, während bei den späteren Versuchsreihen zumindest etwas ausgedehntere Flächen zum Manövrieren und insbesondere mehrere Segeltage mit unterschiedlichen Windverhältnissen zur Verfügung standen.

Strömungskorrektur

Auch wenn die Korrekturen der Strömungssimulation grundsätzlich in die richtige Richtung zielen sollten, ergibt sich hierdurch dennoch eine weitere Fehlerquelle, da das Modell abhängig von diversen Parametern und Annahmen ist, sodass nicht von einer 100% Übereinstimmung mit den tatsächlichen Strömungen im Testgebiet ausgegangen werden kann. Bei der Victoria 2008 fanden die Testfahrten zum größten Teil auf dem Domsee in Ratzeburg statt, wobei auch einzelne längere Fahrten auf den Ratzeburger See unternommen wurden. Dieses Gewässer war am besten für die Erprobung eines römischen Flusskriegsschiffes geeignet, da es quasi keine Strömung gab und die Einschränkungen durch nahe Ufer und andere Wasserfahrzeuge die meiste Zeit vernachlässigt werden konnten.

1.1.2 Vergleich der Rudereigenschaften von Lusoria und Victoria

Versuchsablauf

In diesem Test wurden die *Lusoria Rhenana* und die *Victoria* nacheinander von der gleichen Mannschaft gerudert. Mit jedem Boot wurden drei Testfahrten absolviert, die erste bei einem angenehmen Reisetempo, die zweite mit mehr Kraftaufwand und die dritte mit voller Leistung. Um die möglichst gleichen Bedingungen zu haben, war die Fahrtstrecke immer dieselbe. Auf dem Rückweg konnten sich die Ruderer ausruhen und die *Lusoria Rhenana* bzw. die *Victoria* wurde von einem Motorboot zurück zur Ausgangsposition geschleppt.

War die Strecke immer die gleiche?

Gerudert wurde immer von Westen nach Osten. Der Startpunkt wurde auf dem Wasser abgeschätzt, die Länge der Fahrtstrecke mit dem Log und mit dem GPS gemessen. Tatsächlich liegen die Fahrtstrecken fast genau übereinander (*Lusoria Rhenana* und *Victoria*).

Welche Rolle spielte der Wind?

Die *Lusoria Rhenana* und *Victoria* wurden beide gerudert. Auf beiden Schiffen war der Mast gesetzt, aber keine Rah angeschlagen. Aus vorausgegangenen Versuchen wissen wir, dass schon Mast und Rah allein eine messbare Segelwirkung entfalten können. Das gilt ebenso für das Schanzkleid des Schiffes, insbesondere bei Seitenwind. Am Tag des Tests schwankte die Windgeschwindigkeit zwischen 2,5 und 5,0 Knoten. Die erste Fahrt der *Victoria* lag auf Raumschotkurs (der Wind kam schräg von hinten), die anderen beiden fanden mehr oder weniger bei halbem Wind (Wind von der Seite) statt. Das bedeutet, dass der Wind auf der ersten Fahrt die *Victoria* leicht geschoben hat. Ebenso war eine kleine Drift zu messen, welche den Kiel des Schiffes zur Fahrtrichtung zwischen 5°-10° versetzte. Bei gesetztem Segel hätte diese Windgeschwindigkeit für eine Fahrtgeschwindigkeit um 2,0 Knoten ausgereicht [*Victoria*] (S. 146). Ohne Segel bewirkte der Wind sicherlich deutlich weniger, zumal er seitlich auf den Rumpf auftraf (Beim

Segeln würde man das Segel nach dem Wind ausrichten; in diesem Versuch hingegen waren Rumpf, Mast und Rah statisch.). Zudem steigt der Widerstandsbeiwert für höhere Geschwindigkeiten stark an: Um eine Geschwindigkeit von 6 Knoten zu erreichen, ist eine ca. 15 mal größere Leistung erforderlich als um 2 Knoten zu erreichen [Victoria] (S. 87). Die Leistung, die der Wind erbringt, wäre im Verhältnis zu der Ruderleistung für die gemessene Geschwindigkeit selbst dann zu vernachlässigen, wenn das Segel gesetzt gewesen wäre.

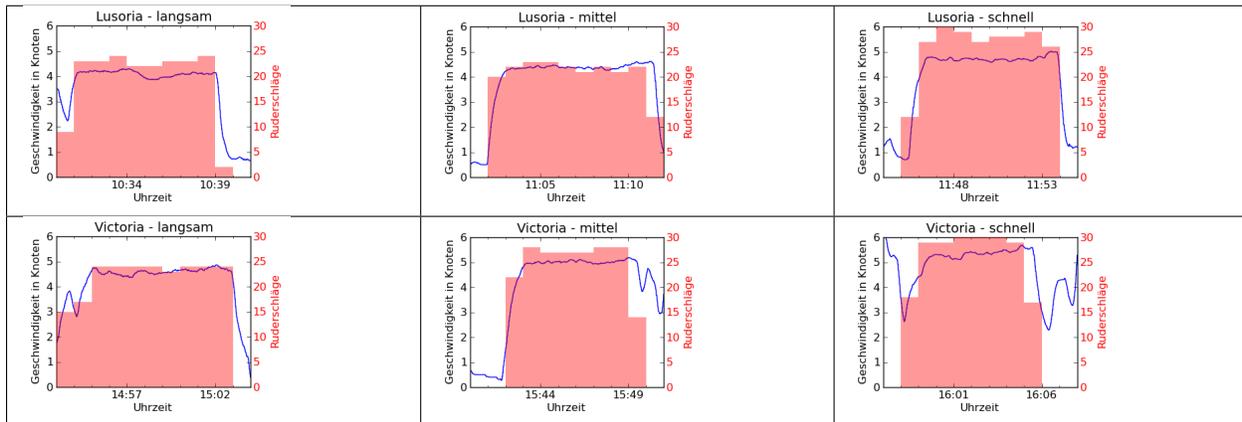
Während der Fahrten mit der *Lusoria Rhenana* schwankte die Windrichtung zwischen einem Raumschotkurs (schräg von hinten) und vor dem Wind (von hinten). Die Windgeschwindigkeit war ähnlich wie auf der *Victoria*. Auch hier gilt, dass die erreichte Geschwindigkeit viel kleiner gewesen wäre, wenn die *Lusoria Rhenana* gesegelt statt gerudert worden wäre. Trotzdem unterstützte der Wind natürlich die Ruderer, zumal er in diesem Fall aus der richtigen Richtung auf Mast und Rah traf und es zumindest keinen Fahrtwind gab, der gebremst hätte.

Welche Geschwindigkeit wurde erreicht?

In der folgenden Tabelle zeigen wir die erreichten Geschwindigkeiten (kn = Knoten) und die mittlere Ruderfrequenz. Während der Fahrt wurde die Ruderschlagzahl pro Minute notiert. Die mittlere Ruderfrequenz berechnen wir aber nur aus den ganzen Minuten. Wenn die Fahrt z.B. um 11:46:27 beginnt, dann wissen wir nicht, wie viele Ruderschläge in den ersten 27 Sekunden dieser Minute gemacht wurden (mit langsamerem Tempo) und wie viele in den letzten 33 Sekunden (mit vollem Tempo).

Schiff	Tempo	Uhrzeit	Geschwindigkeit	mittlere Ruderfrequenz
Lusoria	angenehm	10:31:00-10:39:04	4.1 kn	22.8
Lusoria	schnell	11:03:21-11:11:27	4.4 kn	22.0
Lusoria	maximal	11:46:27-11:53:55	4.7 kn	28.0
Victoria	angenehm	14:54:32-15:49:33	4.6 kn	23.9
Victoria	schnell	15:42:44-15:49:33	5.0 kn	27.5
Victoria	maximal	15:59:00-16:05:39	5.3 kn	29.5

Im Folgenden zeigen wir die Daten. Die gemessene Geschwindigkeit über Grund ist mit der blauen Linie angezeigt, die Anzahl der Ruderschläge pro Minute mit roten Balken (rechte Achse).



Was lernen wir aus diesen Versuchen?

Während der Testfahrten wurden die Ruderer auch sportmedizinisch untersucht, um die erbrachte Leistung zu beurteilen. Dabei wurde der Puls aller 20 Ruderer während der Fahrt überwacht und im Anschluss eine Lactat-Messung durchgeführt, die ein Maß für die Übersäuerung der Muskeln ist. Das Ziel ist u.a. eine Abschätzung, welche Geschwindigkeiten wie lange zumutbar sind, d.h. welche Strecke man in 8-10 Stunden mit einem solchen Schiff unter Ruder zurücklegen könnte. Dies wird an anderer Stelle ausgewertet. Ein paar Dinge lassen sich aber direkt aus der Tabelle und aus den Grafiken ablesen:

- Man muss nicht unbedingt mehr Ruderschläge machen, um schneller voranzukommen. Die zweite Fahrt der *Lusoria Rhenana* erreichte eine höhere Geschwindigkeit mit weniger Ruderschlägen. Dabei musste natürlich jeder Ruderschlag lang und mit mehr Kraft durchgezogen werden. Gefühlt ist dies für die Mannschaft anstrengender, denn auch die Wirkung ist größer.
- Die *Victoria* war in diesen Versuchen schneller. Zusammen mit der Wettfahrt einen Tag vorher, die natürlich notgedrungen zeitgleich, d.h. mit unterschiedlichen Mannschaften durchgeführt werden musste, bestätigt dieses die Widerstandsmessungen der Schiffsbauversuchsanstalt, welche mit verkleinerten Modellen der beiden Schiffe der *Victoria* die besseren Rumpfeigenschaften bescheinigt.

Note: Da beide Schiffe nacheinander gerudert wurden, könnten natürlich die Faktoren Training (besserer Takt) oder Ermüdung (trotz 2 Stunden Pause zur Regeneration) einen Einfluss auf das Endergebnis gehabt haben. Eine Gegenprobe mit umgekehrter Reihenfolge wurde aber nicht durchgeführt.

1.2 Übersichtskarten und Fahrtstrecken

Auf den folgenden Seiten könnt ihr euch die Fahrtroute der Römerschiffe für jeden einzelnen Tag anschauen. Ich habe die Daten nach Jahren sortiert. Zu jeder Fahrt, die wir gemacht haben, gibt es einen Link der die Route direkt in [Google maps](#) anzeigt und einen um die kml Datei herunterzuladen, falls ihr [Google Earth](#) bei Euch auf dem Rechner installiert habt.

1.2.1 Victoria 2008

Dies sind die Testfahrten der Victoria auf dem Ratzeburger See.

Bootsbauer

Studenten Uni Hamburg

Obladener Kohorte

Studenten HSU

1.2.2 Lusoria 2011

Diese Fahrten wurden in Germersheim am Rhein durchgeführt.

Studenten Trier - Team 1

Studenten Trier - Team 2

Studenten der HSU

1.2.3 Tests in 2012

Studenten Trier

In diesem Jahr wird die Lusoria noch einmal getestet. Dieses Mal haben wir einen grosseren Mast dabei.

Tests in Haltern

Diese Fahrten wurden mit verschiedenen Teams auf der Lusoria und der Victoria gemacht. Beide Teams wurden von Sportwissenschaftlern begleitet, die die Leistungsfähigkeit der Ruderer untersucht haben.

1.3 Software documentation

The analysis software we use for this project is written in `python`. The software is available at <https://github.com/hamogu/NX2>.

1.3.1 Main module

exception `NX2.NX2.NX2InterpolationWarning` (*column, maxgap=inf*)
Warning class for interpolation of data columns.

exception `NX2.NX2.NX2RowingWarning`
Warning class for warning related to the matching of rowing data files.

class `NX2.NX2.NX2Table` (**args, **kwargs*)
Basic catalog object.

add_rowing_old_format (*filename, verbose=True*)
add rowing and sailing data

Input filename for cvs file in format as used in 2008

datetime ()
Return an `np.array` of `datetime.datetime` object for each data point
Useful for plotting, so `matplotlib` can label the x-axis correctly.

fit_BSP_corr ()
fit a linear correlation between BSP and SOG

Only points which fulfill the following criteria are used:

- `BSP > 0`
- `np.abs(COG-HDC) < 15`
- small gradients in speed

Returns

- **myoutput** (*scipy.odr.odrpack.Output object*) – contains the fit results
- **con** (*boolean array*) – index array of points usef for fitting

minutes_index ()
return an index array to access exactly one field per minute.

Unfortunately, the NX2 time intervals are not exactly separated by 1s, some values are double or missing. This functions return an index array of those entries where the minute changes, i.e. the first entry within each minute.

plot_course (*scale=50, n=300*)
Return an x/y overview plot of boat path, wind and speeds

plot_polar (*fct*=<function median>, *speedbins*=array([0., 2., 4., 6., 8., 10., 12.]), *anglebins*=array([0., 15.001, 30.002, 45.003, 60.004, 75.005, 90.006, 105.007, 120.008, 135.009, 150.01, 165.011, 180.012]), *color*=['r', 'g', 'b', 'y', 'k', 'c', 'orange'])

Return a polar plot

plot_speeds (*t1*=(0, 0, 0), *t2*=(23, 59, 59))

Return a figure that shows BSP, SOG and rowing (if possible)

time ()

Return an np.array of `datetime.time` object for each data point

Useful for plotting, so matplotlib can label the x-axis correctly.

when (*t1*=(0, 0, 0), *t2*=(23, 59, 59))

Select a subset of the table

Parameters **t2** (*t1*,) – Start and end time in the form (h, m, s)

write_geojson (*filename*)

write geojson file from a NX2 object

Parameters **filename** (*string*) – file name or path for output

write_kml (*filename*, *verbose*=True)

write a kml file from an NX2 object

Parameters **filename** (*string*) – file name or path for output

NX2.NX2.read_NX2 (*self*, *filename*, *date*, *corr_bsp*=1.0, *origin*=None, *timeoffset*=2, *verbose*=True)

read in csv data and initialize table

Parameters

- **filename** (*string*) – filename as string or other input compatible with asciitable
- **date** (tuple of integers (day, month, year)) – date of measurement
- **corr_bsp** (*float*) – multiplicative correction factor for BSP
- **origin** (*tuple*) – (lat, lon) in deg of x,y origin default: lat, lon at first datapoint
- **timeoffset** (*float*) – hours to be added to convert UT to local

NX2.NX2.remove_Danube_current (*data*)

read Danube current simulation and transform data basis system

This procedure read a current simulation of the Danube current in the region north of Regensburg, where the Navis Lusoria was tested in 2006. It adds 2 columns to the NX2 table, that contain the current in the x,y coordinate system (measured west-> east and south-> north) at each position of the ship. Then, the speed over ground (contained in SOG and COG) is transformed into a coordinate system that moves with the water. Thus, after this procedure, the SOG is not longer the “speed over ground”, but instead the “speed over flowing river” and the COG is the “course over flowing river”! Similarly, the TWS and the TWA are transformed into the same coordinate system of the flowing river.

The purpose of this is to correct SOG, COG, TWS and TWA in such a way that the usual procedures for plotting the polar diagram and the drift are applicable.

Parameters **data** (NX2Table) – The dataset to be modified

NX2.NX2.write_leg (*data*, *kmlFile*, *ind*, *name*='', *style*='#yellowLine', *skip*=1)

write one leg of the journal to a kml file.

This does not write complete kml files, neither does it open the file!

Parameters

- **data** (*NX2 instance*) – *LAT* and *LON* are taken from this instance.
- **kmlFile** (*file handle*) – file to write data
- **ind** (*index array*) – indexed values are written in this leg
- **name** (*string , optional*) – Name of this leg in kml file
- **style** (*string , optional*) – name of a line style defined in the kml header
- **skip** (*ind*) – Skips *skip* values before a new position is written. Use for coarser, but smaller files.

1.3.2 Polar plots

Module for polar plots

This module collects all functions necessary for polar plots, ranging from the gridding for angles and wind speeds into an array (`NX2.polar.grid()`) to the plotting (`NX2.polar.plot()`). This also includes some of the filters that are mainly used for polar plots (e.g. `NX2.polar.sail()`).

`NX2.polar.grid` (*angle, wind, bsp, speedbins, anglebins, fig=None*)

Plot a grid of data in bins according to wind angle and wind speed.

This makes a grid of plots showing the distribution of the BSP in each angle and wind bin.

Parameters

- **angle** (*np.ndarray*) – Wind angles in degrees
- **wind** (*np.ndarray*) – wind speed in kn
- **bsp** (*np.ndarray*) – Boat speed in kn
- **speedbins** (*ndarray*) – bin boundaries for speed binning
- **anglebins** (*ndarray*) – bin boundaries for angle binning. Make sure that 180. is included in last bin and not on the boundary.
- **fig** (*matplotlib.figure instance*) – If None, a new figure instance is created.

`NX2.polar.group` (*angle, wind, bsp, speedbins, anglebins, fct=<function median>*)

Group data in bins according to wind angle and wind speed.

Parameters

- **angle** (*np.ndarray*) – Wind angles in degrees
- **wind** (*np.ndarray*) – wind speed in kn
- **bsp** (*np.ndarray*) – Boat speed in kn
- **speedbins** (*ndarray*) – bin boundaries for speed binning
- **anglebins** (*ndarray*) – bin boundaries for angle binning. Make sure that 180. is included in last bin and not on the boundary.
- **fct** (*function*) – Given all bsp values in one (speedbin,anglebin) select on value to be used. Common examples are `np.median` or `np.mean`

Returns polar – This contains the data array with one speed for each (speedbin, anglebin)

Return type `ndarray`(`[len(speedbins)+1, len(anglebins)]`)

`NX2.polar.near_const` (*arr*, *max_diff=0.01*)
mark regions with small gradient in *arr*

This is essentially `np.abs(np.diff(arr)) < max_diff` with one element added, so that input and output have the same number of elements.

Parameters

- **arr** (*1-dim array*) –
- **max_diff** (*np.float*) – maximum allowed gradient between two elements

`NX2.polar.plot` (*ax*, *polardata*, *speedbins*, *anglebins*, *color=['r', 'g', 'b', 'y', 'k', 'c', 'orange']*)
Make a polar plot and label it for data in bins.

Parameters

- **ax** (*matplotlib.axis instance*) – axis where the plot should be added. Will usually be a polar axis.
- **polardata** (*ndarray* (`[len(speedbins)+1, len(anglebins)]`)) – Array with the values to be plotted in individual bins.
- **speedbins** (*ndarray*) – bin boundaries for speed binning
- **anglebins** (*ndarray*) – bin boundaries for angle binning in deg Make sure that 180. is included in last bin and not on the boundary.
- **color** (*array*) – matplotlib colors used for plotting the line in the diagram

`NX2.polar.plot_half_circle` (*ax*, *theta*, *r*, ***kwargs*)
extends *theta* and *r* to touch the final bin boundaries, then plots

Parameters

- **ax** (*mpl.Axis instance*) –
- **theta** (*np.ndarray*) – angles in degrees
- **r** (*np.ndarray*) – radius values for plot
- **plt.plot for a list of accepted keyword arguments.** (See) –

`NX2.polar.sail` (*data*)
mark sailing only regions

This function determines times with the sail up and no rowing. It also cut off the very beginning and end of each sailing phase, because in those times the sail was not set perfectly anyway.

Parameters *data* (*NX2 data*) –

Returns *sail* – True, if vessel was sailing AND not rowing.

Return type array of boolean

`NX2.polar.setup_plot` (*fig*, *axtuple=111*, *maxr=5.0*, *returnall=False*)
setup a polar plot (axis, labels etc.)

Parameters

- **fig** (*matplotlib figure instance*) –
- **axtuple** (*string or number*) – defines how the axis contains is placed in *fig*
- **returnall** (*boolean*) – If true, return additional output

Returns

- **aux_ax1** (*matplotlib*) – Plot in this auxiliary axis (angle in degrees)
- **ax** (*matplotlib.axis instance*)

1.3.3 Math module

Module with a few mathematical definitions

This module collects a few simple, mathematical functions to treat data smoothing or the handling of angles.

`NX2.math.bearingdiff180(a, b)`

Calculate the smallest difference between two bearings.

There are two complications here compared to the usual $a-b$:

- The angle wraps around at 360 deg.
- A-priori it's not clear if the shortest difference between two bearings is clockwise or anti-clockwise.

Parameters **b** ($a,$) – angles in degrees

Returns **out** – smallest difference between a and b in the range $[-180,180]$

Return type float

`NX2.math.smooth_expdec(data, t_e)`

smooth an array with an exponential decay

Parameters

- **data** (*np.ndarray*) – input array to be smoothed
- **t_e** (*int*) – decay timescale of exponential in number of bins (for NX2 data, i.e. seconds)

Returns **out** – smoothed array

Return type np.ndarray

`NX2.math.smooth_gauss(data, width)`

smooth an array with a gauss

Parameters

- **data** (*np.ndarray*) – input array to be smoothed
- **t_e** (*int*) – decay timescale of exponential in number of bins (for NX2 data, i.e. seconds)

Returns **out** – smoothed array

Return type np.ndarray

Index und Suche / Index and search

- genindex
- search

Bibliography

[Victoria] 18. Aßkamp, C. Schäfer (Hrsg.): Projekt Römerschiff, Koehler, Hamburg, 2008

n

`NX2.math`, 13
`NX2.NX2`, 9
`NX2.polar`, 11

A

add_rowing_old_format() (NX2.NX2.NX2Table method), 9

B

bearingdiff180() (in module NX2.math), 13

D

datetime() (NX2.NX2.NX2Table method), 9

F

fit_BSP_corr() (NX2.NX2.NX2Table method), 9

G

Germersheim, 8

grid() (in module NX2.polar), 11

group() (in module NX2.polar), 11

H

Hamburg, 8

L

Lusoria, 8

M

minutes_index() (NX2.NX2.NX2Table method), 9

N

near_const() (in module NX2.polar), 11

NX2.math (module), 13

NX2.NX2 (module), 9

NX2.polar (module), 11

NX2InterpolationWarning, 9

NX2RowingWarning, 9

NX2Table (class in NX2.NX2), 9

P

plot() (in module NX2.polar), 12

plot_course() (NX2.NX2.NX2Table method), 9

plot_half_circle() (in module NX2.polar), 12

plot_polar() (NX2.NX2.NX2Table method), 9

plot_speeds() (NX2.NX2.NX2Table method), 10

R

Ratzeburg, 8

read_NX2() (in module NX2.NX2), 10

remove_Danube_current() (in module NX2.NX2), 10

S

sail() (in module NX2.polar), 12

setup_plot() (in module NX2.polar), 12

smooth_expdec() (in module NX2.math), 13

smooth_gauss() (in module NX2.math), 13

T

time() (NX2.NX2.NX2Table method), 10

V

Victoria, 8

W

when() (NX2.NX2.NX2Table method), 10

write_geojson() (NX2.NX2.NX2Table method), 10

write_kml() (NX2.NX2.NX2Table method), 10

write_leg() (in module NX2.NX2), 10