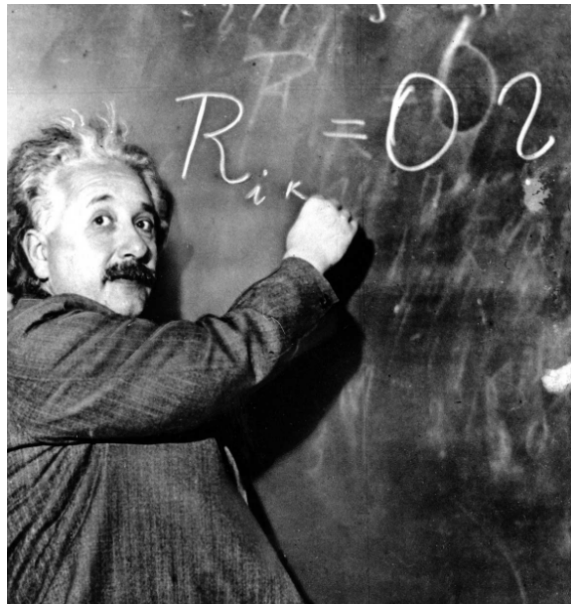


Gekromde oppervlakken en geodeten

Ralf Simons

15-11-2023

Dit document is een aanvulling op de cursus van Nico Roos over de Algemene Relativiteitstheorie van Albert Einstein.



Inhoud

1	Introductie	4
2	Wiskunde van gekromde oppervlakken	5
2.1	Gaussiaanse kromming	5
2.2	Coördinaten	6
2.3	Metrische tensor	6
2.4	Christoffelsymbolen	8
2.5	Riemann-tensor	10
2.6	Ricci-tensor en scalaire kromming	12
2.7	Geodeten	14
3	Verschillende oppervlakken	15
3.1	Oppervlakken door rotatie van $f(r)$ rond z-as	15
3.1.1	Vlak	18
3.1.2	Kegel	20
3.1.3	Logaritmische trechter	22
3.1.4	Opgave	27
3.2	Oppervlakken van bol en donut	28
3.2.1	2-sfeer	28
3.2.2	2-torus	32
4	Geodeten zonder wiskunde	36
5	Literatuur	37

6	Appendix	38
6.1	Methode van Euler	38
6.2	Python-code	40
6.2.1	Harmonische oscillator	40
6.2.2	Vlak	41
6.2.3	Kegel	42
6.2.4	Logaritmische trechter	43
6.2.5	2-sfeer	44
6.2.6	2-torus	46

1 Introductie

In de eerste helft van 2023 heeft Nico Roos in een cursus de algemene relativiteitstheorie van Albert Einstein toegelicht. De einstein-vergelijking vat deze theorie samen:

$$R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \quad (1)$$

De linkerkant van de veldvergelijking beschrijft de kromming of vervorming van de ruimtetijd. De grootte die rechts staat is de energie-impuls-tensor ($T_{\mu\nu}$), en is een object dat zegt hoeveel massa en energie er op een bepaalde plaats in de ruimte is. Het object links wordt ook wel genoteerd als volgt en noemt men de einstein-tensor:

$$G_{\mu\nu} = R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} \quad (2)$$

Als deze niet in de hele ruimte nul is dan is de ruimte gekromd. Als ergens op een bepaalde plaats in de ruimte een zwaar object staat (bijvoorbeeld een ster, zoals onze zon), zit op die plaats veel massa en energie. De einstein-vergelijking zegt dan dat op die plaats (en in de nabijheid ervan) de kromming ook groot moet zijn. Die kromming zorgt ervoor dat de baan van objecten (zoals bijvoorbeeld de aarde) niet meer rechtdoor is, maar wordt afgebogen.

De einstein-tensor (2) is in feite alleen maar afhankelijk van de metriek ofwel metrische tensor $g_{\mu\nu}$. Met kennis van de metriek kun je de Christoffelsymbolen, de Riemann-tensor, de Ricci-tensor en de scalaire kromming berekenen.

In dit document wordt uitgelegd hoe je deze grootheden voor eenvoudige twee-dimensionale oppervlakken¹ berekent. Ook gaan we kijken wat de "meest rechte weg" (geodeet) op deze oppervlakken is.

In het volgende hoofdstuk wordt eerst de wiskunde van gekromde oppervlakken herhaald.

¹normale vlak met poolcoördinaten, kegel, logaritmische trechter, 2-sfeer en 2-torus

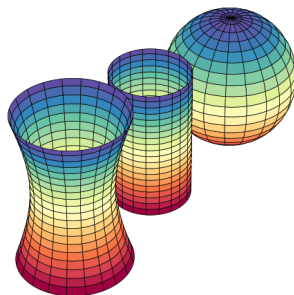
2 Wiskunde van gekromde oppervlakken

2.1 Gaussiaanse kromming

De Gaussiaanse kromming of Gauss-kromming K van een punt op een oppervlak is het product van de hoofdkrommingen, k_1 en k_2 , van dit gegeven punt. De hoofdkrommingen meten op een bepaald punt van een oppervlak, hoe dit oppervlak op dat punt in verschillende richtingen in verschillende mate buigt. k_1 en k_2 zijn de maximale en minimale waarden van deze kromming. Het teken van de Gauss-kromming wordt gebruikt om het oppervlak te karakteriseren:

- $K = k_1 \cdot k_2 > 0$ (elliptisch)
Oppervlakken met een positieve kromming “buigen in ieder punt in twee richtingen dezelfde kant op (bol-bol)”.
- $K = k_1 \cdot k_2 = 0$ (euclidisch)
Oppervlakken met een kromming van nul zijn voorwerpen die in tenminste één richting plat zijn. Deze eigenschap maakt dat deze voorwerpen “uit te vouwen zijn” tot in het platte vlak.
- $K = k_1 \cdot k_2 < 0$ (hyperbolisch)
Oppervlakken met een negatieve kromming “buigen in ieder punt in twee richtingen verschillende kanten op (bol-hol)”.

Hoe bereken je de hoofdkrommingen k_1 en k_2 ? Die directe omgeving van een punt op het oppervlak kun je zowel in de richting van de maximale kromming als ook in de richting van de minimale kromming met een cirkel beschrijven. De radiussen van deze cirkels zijn r_1 en r_2 . Het is dan: $k_1 = 1/r_1$ en $k_2 = 1/r_2$.

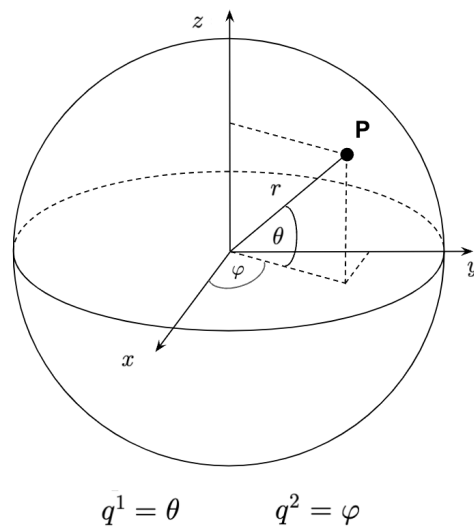
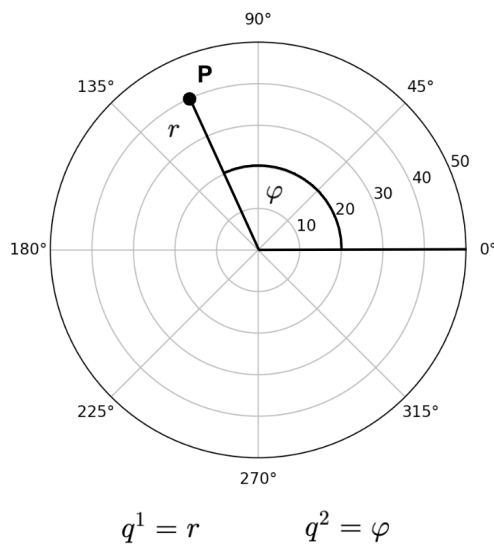


Van links naar rechts: Voor de hyperboloïde is $K < 0$. De cilinder is vlak ($K = 0$) en de sfeer heeft een positieve Gaussiaanse kromming $K > 0$.

2.2 Coördinaten

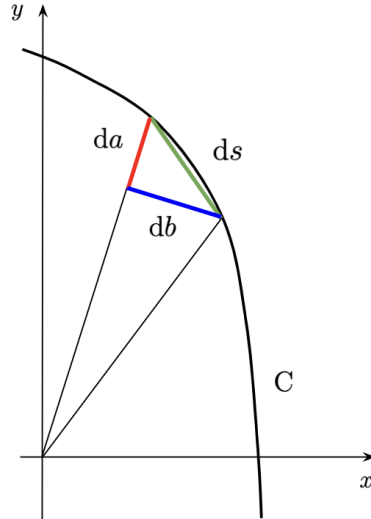
We gaan alleen maar naar tweedimensionale objecten (oppervlakken) kijken. Deze kunnen door twee coördinaten beschreven worden. Omdat het zowel hoeken als ook afstanden kunnen zijn, gebruiken we algemeen de notatie q^1 en q^2 . De boven-indices betekenen niet "machtsverheffen" maar geven aan dat het contravariante componenten zijn. Twee voorbeelden:

- Wanneer je een punt P in een vlak met poolcoördinaten wilt beschrijven dan kies je voor $q^1 = r$ en voor $q^2 = \varphi$. Hierbij is r de afstand van P tot de oorsprong O en φ is de hoek tussen de x-as en de lijn van O naar P.
- Voor de punten op een boloppervlak heb je twee hoeken nodig: de breedtegraad $q^1 = \theta$ en de lengtegraad $q^2 = \varphi$.



2.3 Metrische tensor

Wanneer je de lengte van een lijn op een vlak wilt bepalen dan neem je gewoon een meetliniaal (duimstok) en legt deze naast de lijn. Maar hoe meet je afstanden op een gekromd oppervlak? Hiervoor heb je de metrische tensor nodig. De componenten van de metrische tensor g_{ij} geven de informatie hoe lang een lijnelement ds is in afhankelijkheid van de coördinaten q^1 en q^2 .



We kijken naar een kromme C en berekenen de lengte van ds bij benadering met de stelling van Pythagoras: $ds^2 = da^2 + db^2$
De kleine stukjes da en db hangen als volgt af van kleine veranderingen dq^1 respectievelijk dq^2 :

$$da = \sqrt{g_{11}} dq^1 \quad \text{en} \quad db = \sqrt{g_{22}} dq^2 \quad (3)$$

Dit is gelijk aan:

$$\begin{aligned} da^2 &= g_{11} dq^1 dq^1 \\ db^2 &= g_{22} dq^2 dq^2 \end{aligned}$$

g_{11} en g_{22} zijn componenten van de metrische tensor. Het is nu:

$$ds^2 = g_{11} dq^1 dq^1 + g_{22} dq^2 dq^2$$

In het meest algemene geval zou je ook nog gemengde termen met $dq^1 dq^2$ ofwel $dq^2 dq^1$ kunnen hebben. Hierbij horen componenten van de metriek: g_{12} en g_{21} . Heel algemeen kunnen we schrijven:

$$ds^2 = g_{11} dq^1 dq^1 + g_{12} dq^1 dq^2 + g_{21} dq^2 dq^1 + g_{22} dq^2 dq^2$$

Met de kennis van de componenten van de metrische tensor g_{ij} zijn we dus in staat om de lengte ds op een gekromd oppervlak te berekenen:

$$ds^2 = \sum_{i=1}^2 \sum_{j=1}^2 g_{ij} dq^i dq^j \quad (4)$$

Voor de twee-dimensionale oppervlakken die we later gaan onderzoeken zijn g_{12} en g_{21} altijd nul.

Wat we ook nog nodig hebben zijn de componenten g^{ij} van de inverse metrische tensor. Voor de voorbeelden die later volgen zijn:

$$g^{11} = \frac{1}{g_{11}} \quad \text{en} \quad g^{22} = \frac{1}{g_{22}}$$

De waarden van g^{12} en g^{21} zijn nul.

Nog een paar algemene opmerkingen:

- Omdat we alleen naar twee-dimensionale objecten kijken worden Latijnse indices gebruikt $(i, j, k, l, \dots)$ ².
- In dit document wordt de einstein-sommatieconventie niet gebruikt.

2.4 Christoffelsymbolen

Geometrisch kunnen Christoffelsymbolen Γ^i_{jk} geïnterpreteerd worden als een beschrijving hoe basisvectoren $\vec{e}_j$ in een bepaald coördinatensysteem veranderen.

$$\Gamma^i_{jk} = \frac{\partial \vec{e}_j}{\partial q^k} \cdot \vec{e}^i \quad (5)$$

De basisvectoren kunnen veranderen doordat het coördinatensysteem kromlijng is (dat kan ook in een vlakke ruimte) of doordat de geometrie van de ruimte zelf gekromd is. De Christoffelsymbolen beschrijven beide gevallen.

Zodra je de componenten van de metrische tensor kent kun je de Christoffelsymbolen berekenen:

$$\Gamma^i_{jk} = \frac{1}{2} \sum_{l=1}^2 g^{il} \cdot \left(\frac{\partial g_{jl}}{\partial q^k} + \frac{\partial g_{kl}}{\partial q^j} - \frac{\partial g_{jk}}{\partial q^l} \right) \quad (6)$$

Hierbij kun je trouwens gebruik maken van de eigenschap dat de Christoffelsymbolen niet veranderen bij het verwisselen van de onderindices. Het is:

$$\Gamma^i_{jk} = \Gamma^i_{kj} \quad (7)$$

²Voor een volledige ruimtetijd dus één tijddimensie en drie ruimtedimensies gebruik je Griekse indices.

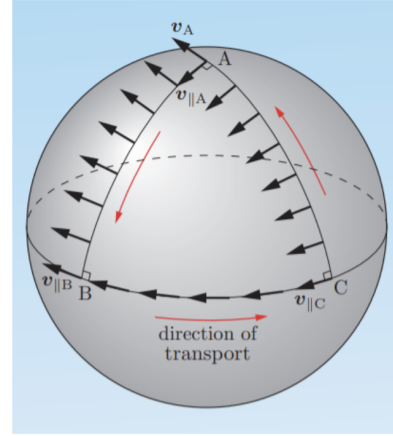
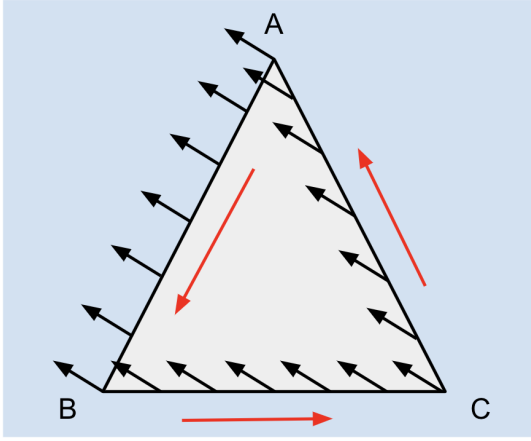
Voor twee-dimensionale oppervlakken zijn er $2^3 = 8$ Christoffelsymbolen. Deze heb je nodig om de componenten van de Riemann-tensor te berekenen en de geodetenvergelijkingen op te stellen (zie hoofdstuk 2.5 en 2.7).

De 8 Christoffelsymbolen zijn dan als volgt:

$$\begin{aligned}
\Gamma^1_{11} &= \frac{1}{2} \left[g^{11} \left(\frac{\partial g_{11}}{\partial q^1} + \frac{\partial g_{11}}{\partial q^1} - \frac{\partial g_{11}}{\partial q^1} \right) + g^{12} \left(\frac{\partial g_{12}}{\partial q^1} + \frac{\partial g_{12}}{\partial q^1} - \frac{\partial g_{11}}{\partial q^2} \right) \right] \\
\Gamma^1_{12} &= \frac{1}{2} \left[g^{11} \left(\frac{\partial g_{11}}{\partial q^2} + \frac{\partial g_{21}}{\partial q^1} - \frac{\partial g_{12}}{\partial q^1} \right) + g^{12} \left(\frac{\partial g_{12}}{\partial q^2} + \frac{\partial g_{22}}{\partial q^1} - \frac{\partial g_{12}}{\partial q^2} \right) \right] \\
\Gamma^1_{21} &= \Gamma^1_{12} \\
\Gamma^1_{22} &= \frac{1}{2} \left[g^{11} \left(\frac{\partial g_{21}}{\partial q^2} + \frac{\partial g_{21}}{\partial q^2} - \frac{\partial g_{22}}{\partial q^1} \right) + g^{12} \left(\frac{\partial g_{22}}{\partial q^2} + \frac{\partial g_{22}}{\partial q^2} - \frac{\partial g_{22}}{\partial q^2} \right) \right] \\
\\
\Gamma^2_{11} &= \frac{1}{2} \left[g^{21} \left(\frac{\partial g_{11}}{\partial q^1} + \frac{\partial g_{11}}{\partial q^1} - \frac{\partial g_{11}}{\partial q^1} \right) + g^{22} \left(\frac{\partial g_{12}}{\partial q^1} + \frac{\partial g_{12}}{\partial q^1} - \frac{\partial g_{11}}{\partial q^2} \right) \right] \\
\Gamma^2_{12} &= \frac{1}{2} \left[g^{21} \left(\frac{\partial g_{11}}{\partial q^2} + \frac{\partial g_{21}}{\partial q^1} - \frac{\partial g_{12}}{\partial q^1} \right) + g^{22} \left(\frac{\partial g_{12}}{\partial q^2} + \frac{\partial g_{22}}{\partial q^1} - \frac{\partial g_{12}}{\partial q^2} \right) \right] \\
\Gamma^2_{21} &= \Gamma^2_{12} \\
\Gamma^2_{22} &= \frac{1}{2} \left[g^{21} \left(\frac{\partial g_{21}}{\partial q^2} + \frac{\partial g_{21}}{\partial q^2} - \frac{\partial g_{22}}{\partial q^1} \right) + g^{22} \left(\frac{\partial g_{22}}{\partial q^2} + \frac{\partial g_{22}}{\partial q^2} - \frac{\partial g_{22}}{\partial q^2} \right) \right]
\end{aligned}$$

2.5 Riemann-tensor

Hoe weten we of de ruimte(tijd) vlak of gekromd is? De methode van het parallel transport over een gesloten curve geeft antwoord op deze vraag. Wij kijken naar een vlakdriehoek en een boldriehoek. Een vector wordt parallel langs de kanten van de driehoek verschoven. Je start bij A, gaat dan naar B en uiteindelijk naar C. Bij de vlakke driehoek verandert de vector na een omloop niet van richting. Op de boldriehoek wijst de vector bij terugkeer naar A in een andere richting.



Je kunt het verschil tussen de startvector en eindvector bij het punt A met behulp van de Riemann-tensor uitrekenen:

$$R^l_{ijk} = \frac{\partial \Gamma^l_{ik}}{\partial q^j} - \frac{\partial \Gamma^l_{ij}}{\partial q^k} + \sum_{m=1}^2 \Gamma^m_{ik} \Gamma^l_{mj} - \sum_{m=1}^2 \Gamma^m_{ij} \Gamma^l_{mk} \quad (8)$$

Wanneer voor alle punten in de ruimte de componenten R^l_{ijk} gelijk nul zijn dan is de ruimte vlak. Zodra er een component verschillend van nul is, dan is de ruimte gekromd.

De volgende eigenschap van de Riemann-tensor is nuttig bij het berekenen van alle componenten:

$$R^l_{ijk} = -R^l_{ikj} \quad (9)$$

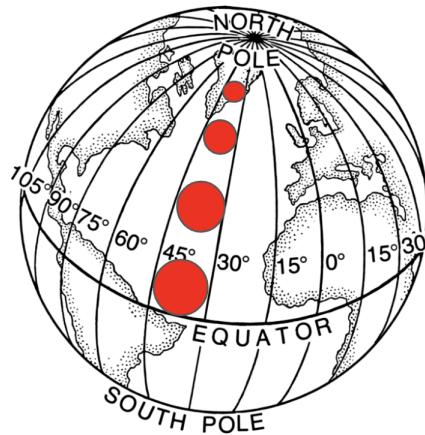
De 16 componenten van de Riemann-tensor zijn dan als volgt:

$$\begin{aligned}
R^1_{111} &= \frac{\partial \Gamma^1_{11}}{\partial q^1} - \frac{\partial \Gamma^1_{11}}{\partial q^1} + \Gamma^1_{11}\Gamma^1_{11} + \Gamma^2_{11}\Gamma^1_{21} - \Gamma^1_{11}\Gamma^1_{11} - \Gamma^2_{11}\Gamma^1_{21} \\
R^1_{112} &= \frac{\partial \Gamma^1_{12}}{\partial q^1} - \frac{\partial \Gamma^1_{11}}{\partial q^2} + \Gamma^1_{12}\Gamma^1_{11} + \Gamma^2_{12}\Gamma^1_{21} - \Gamma^1_{11}\Gamma^1_{12} - \Gamma^2_{11}\Gamma^1_{22} \\
R^1_{121} &= -R^1_{112} \\
R^1_{122} &= \frac{\partial \Gamma^1_{12}}{\partial q^2} - \frac{\partial \Gamma^1_{12}}{\partial q^2} + \Gamma^1_{12}\Gamma^1_{12} + \Gamma^2_{12}\Gamma^1_{22} - \Gamma^1_{12}\Gamma^1_{12} - \Gamma^2_{12}\Gamma^1_{22} \\
R^1_{211} &= \frac{\partial \Gamma^1_{21}}{\partial q^1} - \frac{\partial \Gamma^1_{21}}{\partial q^1} + \Gamma^1_{21}\Gamma^1_{11} + \Gamma^2_{21}\Gamma^1_{21} - \Gamma^1_{21}\Gamma^1_{11} - \Gamma^2_{21}\Gamma^1_{21} \\
R^1_{212} &= \frac{\partial \Gamma^1_{22}}{\partial q^1} - \frac{\partial \Gamma^1_{21}}{\partial q^2} + \Gamma^1_{22}\Gamma^1_{11} + \Gamma^2_{22}\Gamma^1_{21} - \Gamma^1_{21}\Gamma^1_{12} - \Gamma^2_{21}\Gamma^1_{22} \\
R^1_{221} &= -R^1_{212} \\
R^1_{222} &= \frac{\partial \Gamma^1_{22}}{\partial q^2} - \frac{\partial \Gamma^1_{22}}{\partial q^2} + \Gamma^1_{22}\Gamma^1_{12} + \Gamma^2_{22}\Gamma^1_{22} - \Gamma^1_{22}\Gamma^1_{12} - \Gamma^2_{22}\Gamma^1_{22} \\
\\
R^2_{111} &= \frac{\partial \Gamma^2_{11}}{\partial q^1} - \frac{\partial \Gamma^2_{11}}{\partial q^1} + \Gamma^1_{11}\Gamma^2_{11} + \Gamma^2_{11}\Gamma^2_{21} - \Gamma^1_{11}\Gamma^2_{11} - \Gamma^2_{11}\Gamma^2_{21} \\
R^2_{112} &= \frac{\partial \Gamma^2_{12}}{\partial q^1} - \frac{\partial \Gamma^2_{11}}{\partial q^2} + \Gamma^1_{12}\Gamma^2_{11} + \Gamma^2_{12}\Gamma^2_{21} - \Gamma^1_{11}\Gamma^2_{12} - \Gamma^2_{11}\Gamma^2_{22} \\
R^2_{121} &= -R^2_{112} \\
R^2_{122} &= \frac{\partial \Gamma^2_{12}}{\partial q^2} - \frac{\partial \Gamma^2_{12}}{\partial q^2} + \Gamma^1_{12}\Gamma^2_{12} + \Gamma^2_{12}\Gamma^2_{22} - \Gamma^1_{12}\Gamma^2_{12} - \Gamma^2_{12}\Gamma^2_{22} \\
R^2_{211} &= \frac{\partial \Gamma^2_{21}}{\partial q^1} - \frac{\partial \Gamma^2_{21}}{\partial q^1} + \Gamma^1_{21}\Gamma^2_{11} + \Gamma^2_{21}\Gamma^2_{21} - \Gamma^1_{21}\Gamma^2_{11} - \Gamma^2_{21}\Gamma^2_{21} \\
R^2_{212} &= \frac{\partial \Gamma^2_{22}}{\partial q^1} - \frac{\partial \Gamma^2_{21}}{\partial q^2} + \Gamma^1_{22}\Gamma^2_{11} + \Gamma^2_{22}\Gamma^2_{21} - \Gamma^1_{21}\Gamma^2_{12} - \Gamma^2_{21}\Gamma^2_{22} \\
R^2_{221} &= -R^2_{212} \\
R^2_{222} &= \frac{\partial \Gamma^2_{22}}{\partial q^2} - \frac{\partial \Gamma^2_{22}}{\partial q^2} + \Gamma^1_{22}\Gamma^2_{12} + \Gamma^2_{22}\Gamma^2_{22} - \Gamma^1_{22}\Gamma^2_{12} - \Gamma^2_{22}\Gamma^2_{22}
\end{aligned}$$

2.6 Ricci-tensor en scalaire kromming

De Ricci-tensor R_{ik} beschrijft de verandering van de grootte van een oppervlak-element wanneer dit element langs geodeten beweegt.

Hier een eenvoudig voorbeeld: De lengtegraden op de aarde zijn geodeten. Je kunt nu een schijf langs deze geodeten van de evenaar naar de noordpool verschuiven. De Ricci-tensor houdt bij hoe de grootte van deze schijf verandert. In dit voorbeeld krimpt de schijf. Algemeen gaat het om de krimp of groei van een “volume” dat langs geodeten beweegt.



Zodra je de componenten van de Riemann-tensor kent kun je de componenten van de Ricci-tensor berekenen:

$$R_{ik} = \sum_{j=1}^2 R^j_{ijk} \quad (10)$$

Of helemaal uitgeschreven:

$$R_{11} = R^1_{111} + R^2_{121}$$

$$R_{12} = R^1_{112} + R^2_{122}$$

$$R_{21} = R^1_{211} + R^2_{221}$$

$$R_{22} = R^1_{212} + R^2_{222}$$

De scalaire kromming R geeft aan in welke mate een oppervlak verschilt van de vlakke ruimte. Ook hier weer een eenvoudig voorbeeld: Wij kijken eerst naar een vlakke schijf met omtrek C (plaatje links). De oppervlakte van deze schijf is $A_{vlak} = C^2/4\pi$. Dan plaatsen wij een schijf met dezelfde (!) omtrek C op een sfeer (plaatje rechts). Het oppervlak A_{sfeer} is groter dan A_{vlak} . De scalaire kromming van de vlakke schijf is $R_{vlak} = 0$. Daarentegen geldt voor het "keppeltje": $R_{sfeer} > 0$.



Wiskundig gezien is R de contractie van de Ricci-tensor met de contravariante (inverse) versie van de metrische tensor g^{ik} :

$$R = \sum_{i=1}^2 \sum_{k=1}^2 g^{ik} R_{ik} \quad (11)$$

Of weer helemaal uitgeschreven:

$$R = g^{11} R_{11} + g^{12} R_{12} + g^{21} R_{21} + g^{22} R_{22}$$

Voor twee-dimensionale objecten is de scalaire kromming R het dubbele van de gaussiaanse kromming K (zie hoofdstuk 2.1):

$$R = 2 \cdot K \quad (12)$$

2.7 Geodeten

Een deeltje waarop geen kracht werkt kiest in de ruimte(tijd) de meest rechte weg tussen twee punten. De lijn tussen deze twee punten noemt je de geodeet. Om de geodeet op een twee-dimensionaal oppervlak te vinden moet je de oplossingen van de volgende vergelijkingen vinden:

$$\frac{d^2 q^i}{dt^2} + \sum_{j=1}^2 \sum_{k=1}^2 \Gamma^i_{jk} \frac{dq^j}{dt} \frac{dq^k}{dt} = 0 \quad (13)$$

Wanneer we de vergelijkingen (13) helemaal uitschrijven dan krijgen we twee gekoppelde differentiaalvergelijkingen van de tweede orde:

$$\frac{d^2 q^1}{dt^2} + \Gamma^1_{11} \frac{dq^1}{dt} \frac{dq^1}{dt} + \Gamma^1_{12} \frac{dq^1}{dt} \frac{dq^2}{dt} + \Gamma^1_{21} \frac{dq^2}{dt} \frac{dq^1}{dt} + \Gamma^1_{22} \frac{dq^2}{dt} \frac{dq^2}{dt} = 0 \quad (14)$$

$$\frac{d^2 q^2}{dt^2} + \Gamma^2_{11} \frac{dq^1}{dt} \frac{dq^1}{dt} + \Gamma^2_{12} \frac{dq^1}{dt} \frac{dq^2}{dt} + \Gamma^2_{21} \frac{dq^2}{dt} \frac{dq^1}{dt} + \Gamma^2_{22} \frac{dq^2}{dt} \frac{dq^2}{dt} = 0 \quad (15)$$

In het algemeen is het bijna onmogelijk om een analytische oplossing voor (14) en (15) te vinden. Maar je kunt de oplossing benaderen door middel van een numerieke analyse. De methode van Euler werkt heel goed (zie hoofdstuk 6.1). Ik heb voor ieder twee-dimensionaal oppervlak een klein python-programma geschreven dat geodeten berekent en grafisch in 3D weergeeft (zie hoofdstukken 6.2.2 t/m 6.2.6).

3 Verschillende oppervlakken

3.1 Oppervlakken door rotatie van $f(r)$ rond z-as

We gaan eerst naar oppervlakken kijken die ontstaan door de rotatie van een functie $f(r)$ rond de z-as. De afstand r tot de z-as loopt van 0 tot oneindig ($0 \leq r < \infty$) en de azimut φ ligt tussen $-\pi$ en $+\pi$ ($-\pi < \varphi \leq +\pi$). De coördinaten van zo een oppervlak zijn dus $q^1 = r$ en $q^2 = \varphi$. De cartesische coördinaten x, y en z zijn:

$$x = r \cdot \cos \varphi$$

$$y = r \cdot \sin \varphi$$

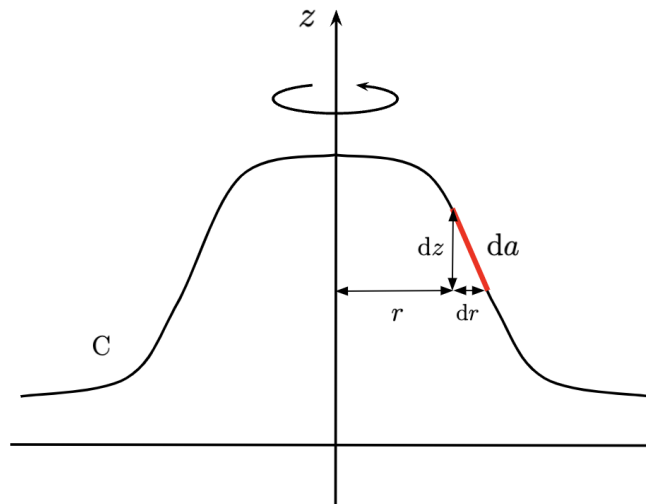
$$z = f(r)$$

Heel algemeen onderzoeken we nu wat de metrische tensor, de Christoffelsymbolen en de geodeten-vergelijkingen van deze klasse oppervlakken zijn. Beginnen we met de metriek: Een kleine afstand ds op het oppervlak kunnen we weer met behulp van de stelling van Pythagoras berekenen: $ds^2 = da^2 + db^2$. Het stukje da loopt radiaal en db staat er haaks op in azimutale richting.

Voor de berekening van da hebben we de afgeleide $f'(r)$ nodig:

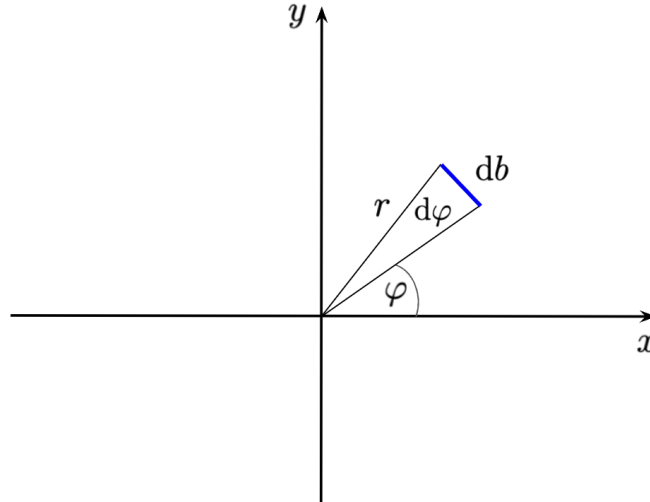
$$\frac{dz}{dr} = \frac{df(r)}{dr} = f'(r) \quad \Rightarrow \quad dz = f'(r) \cdot dr$$

$$da^2 = dz^2 + dr^2 = f'(r)^2 \cdot dr^2 + dr^2$$



Het andere lijnstukje db is een kleine draaiing $d\varphi$ op een afstand r rond de z -as:

$$db = r \cdot d\varphi \quad \Rightarrow \quad db^2 = r^2 \cdot d\varphi^2$$



Samenvattend geldt dus:

$$ds^2 = da^2 + db^2 = (f'(r)^2 + 1) dr^2 + r^2 d\varphi^2 \quad (16)$$

Vanaf nu ga ik vaak in plaats van de indices 1 en 2 gewoon de betreffende coördinaat (bv. r of φ) als index gebruiken. De componenten van de metrische tensor g_{ij} zijn:

$$g_{11} = g_{rr} = f'(r)^2 + 1$$

$$g_{12} = g_{r\varphi} = 0$$

$$g_{21} = g_{\varphi r} = 0$$

$$g_{22} = g_{\varphi\varphi} = r^2$$

Hieruit kun je makkelijk de componenten van de inverse metrische tensor g^{ij} bepalen:

$$g^{11} = g^{rr} = \frac{1}{f'(r)^2 + 1}$$

$$g^{12} = g^{r\varphi} = 0$$

$$g^{21} = g^{\varphi r} = 0$$

$$g^{22} = g^{\varphi\varphi} = \frac{1}{r^2}$$

Met kennis van de metriek kunnen we de 8 Christoffelsymbolen berekenen (zie vergelijking 6). $\Gamma^1_{12}, \Gamma^1_{21}, \Gamma^2_{11}$ en Γ^2_{22} zijn nul. Maar de volgende 4 Christoffelsymbolen zijn verschillend van nul:

$$\begin{aligned}\Gamma^1_{11} &= \Gamma^r_{rr} = \frac{f'(r) \cdot f''(r)}{f'(r)^2 + 1} \\ \Gamma^1_{22} &= \Gamma^r_{\varphi\varphi} = -\frac{r}{f'(r)^2 + 1} \\ \Gamma^2_{12} &= \Gamma^\varphi_{r\varphi} = \frac{1}{r} \\ \Gamma^2_{21} &= \Gamma^\varphi_{\varphi r} = \frac{1}{r}\end{aligned}$$

De differentiaalvergelijkingen voor de geodeten kunnen we nu opstellen (zie (14) en (15)):

$$\frac{d^2r}{dt^2} + \frac{f'(r) \cdot f''(r)}{f'(r)^2 + 1} \cdot \left(\frac{dr}{dt}\right)^2 - \frac{r}{f'(r)^2 + 1} \cdot \left(\frac{d\varphi}{dt}\right)^2 = 0 \quad (17)$$

$$\frac{d^2\varphi}{dt^2} + \frac{2}{r} \cdot \frac{dr}{dt} \cdot \frac{d\varphi}{dt} = 0 \quad (18)$$

Deze geodeten-vergelijkingen gaan we nu voor drie tweedimensionale oppervlakken toepassen: het gewone vlak, de kegel en de logaritmische trechter. Ook berekenen we de Riemann-tensor, de Ricci-tensor en de scalaire kromming voor deze oppervlakken.

3.1.1 Vlak

De beschrijving van een vlak met poolcoördinaten r en φ wordt vaak toegepast. De straal r loopt van nul tot oneindig en de hoek φ van $-\pi$ naar $+\pi$. Neem je ook nog de z -richting erbij dan kun je de punten in een vlak met de volgende cartesische coördinaten weergeven:

$$\begin{aligned}x &= r \cdot \cos \varphi \\y &= r \cdot \sin \varphi \\z &= f(r) = 0\end{aligned}$$

Met behulp van vergelijking (16) kunnen we onmiddellijk de metriek bepalen:

$$ds^2 = dr^2 + r^2 d\varphi^2 \quad (19)$$

Dus de componenten van de metrische tensor en de inverse tensor zijn (zie pagina 16):

$$\begin{aligned}g_{11} &= g_{rr} = 1 \\g_{12} &= g_{r\varphi} = 0 \\g_{21} &= g_{\varphi r} = 0 \\g_{22} &= g_{\varphi\varphi} = r^2\end{aligned}$$

$$\begin{aligned}g^{11} &= g^{rr} = 1 \\g^{12} &= g^{r\varphi} = 0 \\g^{21} &= g^{\varphi r} = 0 \\g^{22} &= g^{\varphi\varphi} = \frac{1}{r^2}\end{aligned}$$

Met de formules op pagina 17 voor de Christoffelsymbolen zijn deze voor het vlak makkelijk te berekenen ($\Gamma^1_{11} = \Gamma^r_{rr} = 0$):

$$\begin{aligned}\Gamma^1_{22} &= \Gamma^r_{\varphi\varphi} = -r \\ \Gamma^2_{12} &= \Gamma^\varphi_{r\varphi} = \frac{1}{r} \\ \Gamma^2_{21} &= \Gamma^\varphi_{\varphi r} = \frac{1}{r}\end{aligned}$$

De volgende stap is het berekenen van de Riemann-tensor, de Ricci-tensor en de scalaire kromming (zie vergelijkingen (8), (10) en (11)).

Alle componenten van de Riemann-tensor R^l_{ijk} zijn nul. Hieruit volgt dat ook de componenten van de Ricci-tensor R_{ik} en de scalaire kromming R nul zijn.

$$\begin{aligned} R^l_{ijk} &= 0 \\ R_{ik} &= 0 \\ R &= 0 \end{aligned}$$

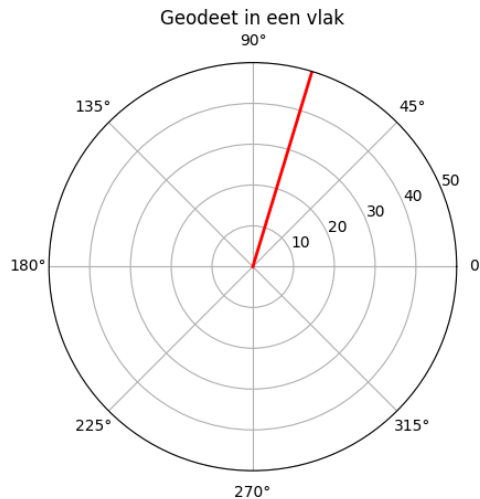
Dit betekent zoals verwacht dat het gewone vlak geen kromming heeft. Met behulp van de vergelijkingen (17) en (18) worden de geodeten-vergelijkingen voor het vlak:

$$\frac{d^2 r}{dt^2} - r \cdot \left(\frac{d\varphi}{dt} \right)^2 = 0 \quad (20)$$

$$\frac{d^2 \varphi}{dt^2} + \frac{2}{r} \cdot \frac{dr}{dt} \cdot \frac{d\varphi}{dt} = 0 \quad (21)$$

Dit zijn gekoppelde differentiaalvergelijkingen van de tweede orde. We weten dat de geodeet in een vlak gewoon een rechte lijn moet zijn. Maar dit kun je niet zomaar aan deze twee vergelijkingen aflezen. Ik heb daarom naar een numerieke oplossing met de methode van Euler gezocht (zie hoofdstuk 6.1). De betreffende Python-code kun je in hoofdstuk 6.2.2 vinden.

Zoals verwacht is inderdaad de geodeet een rechte lijn:



De geodeten-vergelijkingen (20) en (21) kunnen we nog iets anders schrijven:

- We gebruiken de punt-notatie voor de afgeleide naar de tijd t (bv. $\frac{dr}{dt} = \dot{r}$).
- Voor $\frac{d\varphi}{dt}$ schrijven we ω .
- De vergelijkingen worden met een massa m vermenigvuldigt.
- Vergelijking (21) vermenigvuldigen we nog met r .

Dan worden (20) en (21):

$$m \ddot{r} - m r \omega^2 = 0 \quad \Rightarrow \quad \underbrace{m \ddot{r}}_{\text{radiale kracht}} = \underbrace{m r \omega^2}_{\text{centrifugale kracht}} \quad (22)$$

$$m r \dot{\omega} + 2 m \dot{r} \omega = 0 \quad \Rightarrow \quad \underbrace{m r \dot{\omega}}_{\text{eulerkracht}} = \underbrace{-2 m \dot{r} \omega}_{\text{corioliskracht}} \quad (23)$$

De geodeten-vergelijkingen voor een vrij deeltje in een vlak met pool-coördinaten worden dus vergelijkingen met schijnkrachten (radiale kracht, centrifugale kracht, eulerkracht en corioliskracht). Let op: Een schijnkracht geeft een deeltje geen echte versnelling. Maar omdat we een kromlijngig coördinatenstelsel gebruiken lijkt het dat het deeltje deze schijnkrachten voelt.

3.1.2 Kegel

We gaan nu naar een kegeloppervlak kijken. De top van de kegel ligt op de z -as met een afstand h tot het xy -vlak. De helling wordt beschreven met de parameter σ :

$$\begin{aligned} x &= r \cdot \cos \varphi \\ y &= r \cdot \sin \varphi \\ z &= h - \sigma \cdot r \end{aligned}$$

Met behulp van vergelijking (16) kunnen we weer de metriek bepalen:

$$ds^2 = (\sigma^2 + 1) dr^2 + r^2 d\varphi^2 \quad (24)$$

De componenten van de metrische tensor en de inverse tensor zijn dan (zie pagina 16):

$$\begin{aligned}g_{11} &= g_{rr} = \sigma^2 + 1 \\g_{12} &= g_{r\varphi} = 0 \\g_{21} &= g_{\varphi r} = 0 \\g_{22} &= g_{\varphi\varphi} = r^2\end{aligned}$$

$$\begin{aligned}g^{11} &= g^{rr} = \frac{1}{\sigma^2 + 1} \\g^{12} &= g^{r\varphi} = 0 \\g^{21} &= g^{\varphi r} = 0 \\g^{22} &= g^{\varphi\varphi} = \frac{1}{r^2}\end{aligned}$$

Met de formules op pagina 17 voor de Christoffelsymbolen zijn deze voor de kegel eenvoudig te berekenen ($\Gamma^1_{11} = \Gamma^r_{rr} = 0$):

$$\begin{aligned}\Gamma^1_{22} &= \Gamma^r_{\varphi\varphi} = -\frac{r}{\sigma^2 + 1} \\ \Gamma^2_{12} &= \Gamma^\varphi_{r\varphi} = \frac{1}{r} \\ \Gamma^2_{21} &= \Gamma^\varphi_{\varphi r} = \frac{1}{r}\end{aligned}$$

Wanneer je alle componenten van de Riemann-tensor R^l_{ijk} berekent (zie pagina 11) dan krijg je als resultaat dat alle nul zijn. Hieruit volgt dat ook de componenten van de Ricci-tensor R_{ik} en de scalaire kromming R nul zijn.

$$\begin{aligned}R^l_{ijk} &= 0 \\ R_{ik} &= 0 \\ R &= 0\end{aligned}$$

Dit betekent dat de kegel in feite vlak is. Wanneer je de kegel van de top naar de rand opensnijdt dan kun je deze plat op een tafel neerleggen.

Met behulp van de vergelijkingen (17) en (18) worden de geodeten-

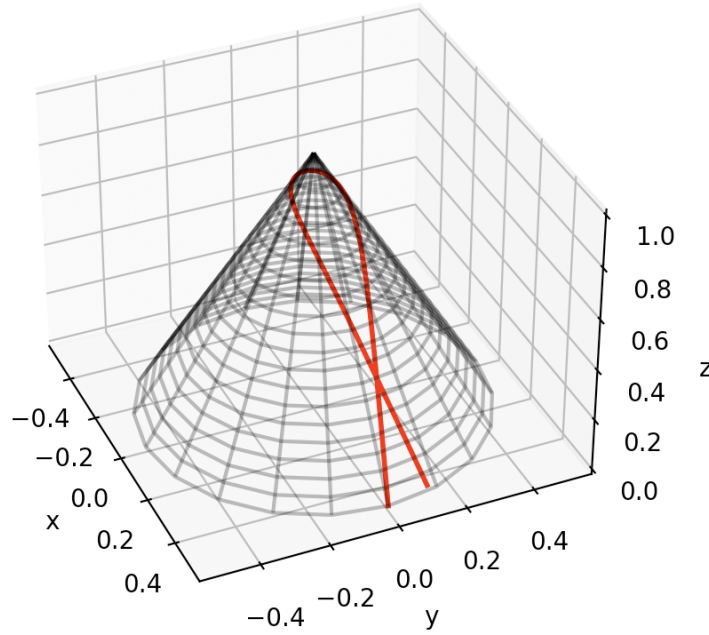
vergelijkingen voor de kegel:

$$\frac{d^2 r}{dt^2} - \frac{r}{\sigma^2 + 1} \cdot \left(\frac{d\varphi}{dt} \right)^2 = 0 \quad (25)$$

$$\frac{d^2 \varphi}{dt^2} + \frac{2}{r} \cdot \frac{dr}{dt} \cdot \frac{d\varphi}{dt} = 0 \quad (26)$$

Met de Python-code in hoofdstuk 6.2.3 kunnen we numerieke oplossingen voor deze gekoppelde differentiaalvergelijkingen vinden. Hier zien we een voorbeeld van een geodeet op een kegel-oppervlak (met $h = 1, \sigma = 2, r_0 = h/\sigma, \dot{r}_0 = -10, \varphi_0 = 0, \dot{\varphi}_0 = 5$):

Geodeet op een kegel-oppervlak



3.1.3 Logaritmische trechter

Het vlak en de kegel hebben geen kromming: Alle componenten van de Riemann-tensor, de Ricci-tensor en de scalaire kromming zijn nul. We gaan nu naar een voorbeeld kijken waar dit niet meer het geval is: de logaritmische trechter. Deze ontstaat door de functie $f(r) = \ln r$

rond de z-as te laten draaien. De cartesische coördinaten zijn dan:

$$\begin{aligned}x &= r \cdot \cos \varphi \\y &= r \cdot \sin \varphi \\z &= \ln r\end{aligned}$$

Met behulp van vergelijking (16) kunnen we weer de metriek bepalen:

$$ds^2 = \left(\frac{1}{r^2} + 1 \right) dr^2 + r^2 d\varphi^2 \quad (27)$$

De componenten van de metrische tensor en de inverse tensor zijn dan (zie pagina 16):

$$\begin{aligned}g_{11} &= g_{rr} = \frac{1}{r^2} + 1 \\g_{12} &= g_{r\varphi} = 0 \\g_{21} &= g_{\varphi r} = 0 \\g_{22} &= g_{\varphi\varphi} = r^2\end{aligned}$$

$$\begin{aligned}g^{11} &= g^{rr} = \frac{r^2}{r^2 + 1} \\g^{12} &= g^{r\varphi} = 0 \\g^{21} &= g^{\varphi r} = 0 \\g^{22} &= g^{\varphi\varphi} = \frac{1}{r^2}\end{aligned}$$

Met de formules op pagina 17 voor de Christoffelsymbolen zijn deze voor de logaritmische trechter eenvoudig te berekenen:

$$\begin{aligned}\Gamma^1_{11} &= \Gamma^r_{rr} = -\frac{1}{r^3 + r} \\ \Gamma^1_{22} &= \Gamma^r_{\varphi\varphi} = -\frac{r^3}{r^2 + 1} \\ \Gamma^2_{12} &= \Gamma^\varphi_{r\varphi} = \frac{1}{r} \\ \Gamma^2_{21} &= \Gamma^\varphi_{\varphi r} = \frac{1}{r}\end{aligned}$$

Nu wordt het interessant: Van de 16 componenten van de Riemann-tensor R^l_{ijk} (vergelijking (8)) zijn 4 componenten verschillend van nul:

$$\begin{aligned} R^1_{212} &= R^r_{\varphi r \varphi} = -\frac{r^2}{(r^2 + 1)^2} \\ R^1_{221} &= R^r_{\varphi \varphi r} = \frac{r^2}{(r^2 + 1)^2} \\ R^2_{112} &= R^\varphi_{rr \varphi} = \frac{1}{r^2(r^2 + 1)} \\ R^2_{121} &= R^\varphi_{r \varphi r} = -\frac{1}{r^2(r^2 + 1)} \end{aligned}$$

Dit betekent direct dat dit oppervlak niet vlak is. We gaan nu verder en berekenen de Ricci-tensor R_{ik} (zie vergelijking (10)). De componenten $R_{12} = R_{r\varphi}$ en $R_{21} = R_{\varphi r}$ zijn nul.

$$\begin{aligned} R_{11} &= R_{rr} = -\frac{1}{r^2(r^2 + 1)} \\ R_{22} &= R_{\varphi\varphi} = -\frac{r^2}{(r^2 + 1)^2} \end{aligned}$$

De scalaire kromming R (vergelijking (11)) is:

$$R = -\frac{2}{(r^2 + 1)^2}$$

Wat valt op wanneer we naar R kijken?

- De scalaire kromming R is altijd negatief!
- Omdat $R = 2 \cdot K$ (zie vergelijking (12)) geldt voor de Gaussiaanse kromming K :

$$K = -\frac{1}{(r^2 + 1)^2}$$

- Voor zeer kleine afstanden r tot de z -as wordt $R = -2$ (en hiermee $K = -1$). De logaritmische trechter gedraagt zich daar als een pseudo-sfeer³.

³Een pseudo-sfeer heeft een constante negatieve kromming. De normale sfeer heeft overal een constante positieve kromming.

- Voor zeer grote afstanden r wordt R (en K) gelijk nul. Dus de trechter is daar vlak.

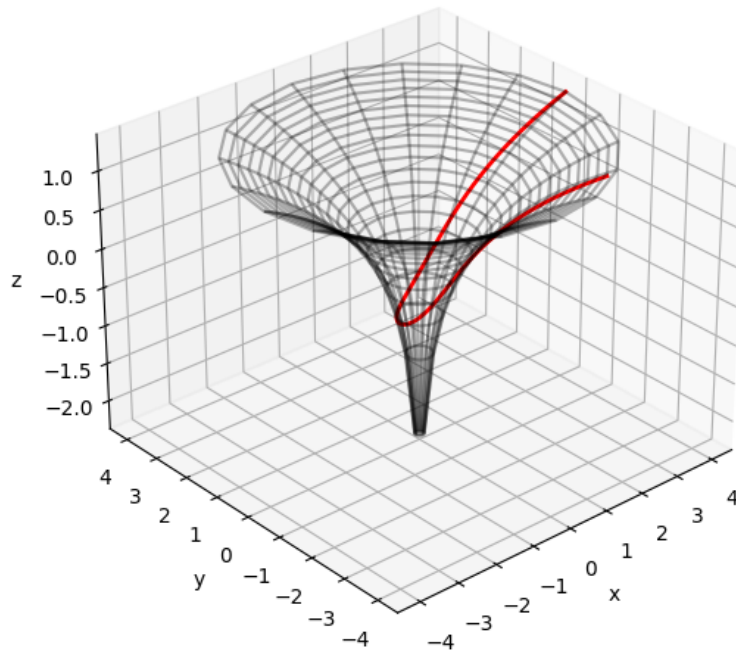
Met behulp van de vergelijkingen (17) en (18) worden de geodeten-vergelijkingen voor de logaritmische trechter:

$$\frac{d^2 r}{dt^2} - \frac{1}{r^3 + r} \cdot \left(\frac{dr}{dt} \right)^2 - \frac{r^3}{r^2 + 1} \cdot \left(\frac{d\varphi}{dt} \right)^2 = 0 \quad (28)$$

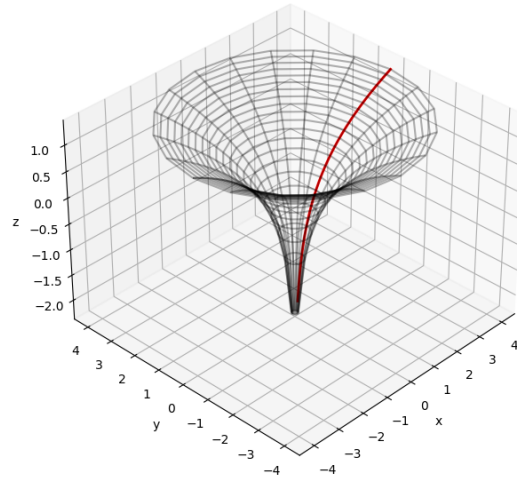
$$\frac{d^2 \varphi}{dt^2} + \frac{2}{r} \cdot \frac{dr}{dt} \cdot \frac{d\varphi}{dt} = 0 \quad (29)$$

Met de Python-code in hoofdstuk 6.2.4 kunnen we numerieke oplossingen voor deze gekoppelde differentiaalvergelijkingen vinden. Hier zien we een voorbeeld van een geodeet op een kegel-oppervlak (met $r_0 = 4, \dot{r}_0 = -10, \varphi_0 = 0, \dot{\varphi}_0 = 0,3$):

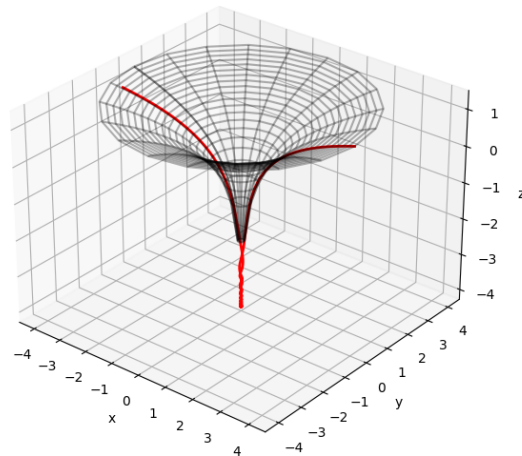
Geodeet op een logaritmische trechter



Wanneer een deeltje precies langs een meridiaan zou vliegen (bv. met de beginvoorwaarden $r_0 = 4, \dot{r}_0 = -10, \varphi_0 = 0, \dot{\varphi}_0 = 0$), dan verdwijnt het deeltje natuurlijk in het gat van de trechter:



Maar wat gebeurt wanneer het deeltje in het begin een kleine azimuthale snelheid $\dot{\varphi}_0$ heeft? Hier zien we een voorbeeld met $\dot{\varphi}_0 = 0,01$:



Het deeltje komt naar een reis in het gat weer terug naar boven. Je zou dus kunnen zeggen dat de logarithmische trechter repulsief is.

3.1.4 Opgave

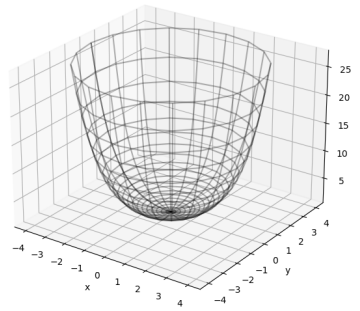
Hier volgt nu een opgave om zelf eens voor een oppervlak de metriek, de Christoffelsymbolen, de componenten van de Riemann- en Ricci-tensor en de scalaire kromming te berekenen. De cartesische coördinaten zijn als volgt. De afstand r tot de z-as loopt van 0 tot oneindig ($0 \leq r < \infty$) en de azimuth φ ligt tussen $-\pi$ en $+\pi$ ($-\pi < \varphi \leq +\pi$):

$$x = r \cdot \cos \varphi$$

$$y = r \cdot \sin \varphi$$

$$z = f(r) = \cosh r$$

Dit wordt hier grafisch weergegeven:



Bewijs dat voor de scalaire kromming R geldt:

$$R = \frac{2 \cdot \sinh r}{r \cdot \cosh^3 r}$$

Hint:

Alleen de volgende componenten van de Riemann-tensor zijn verschillend van nul:

$$R^1_{212} = R^r_{\varphi r \varphi}$$

$$R^1_{221} = R^r_{\varphi \varphi r}$$

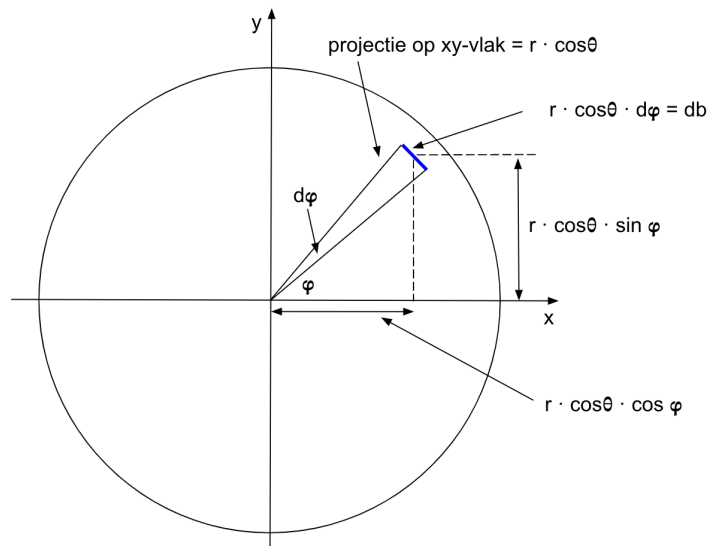
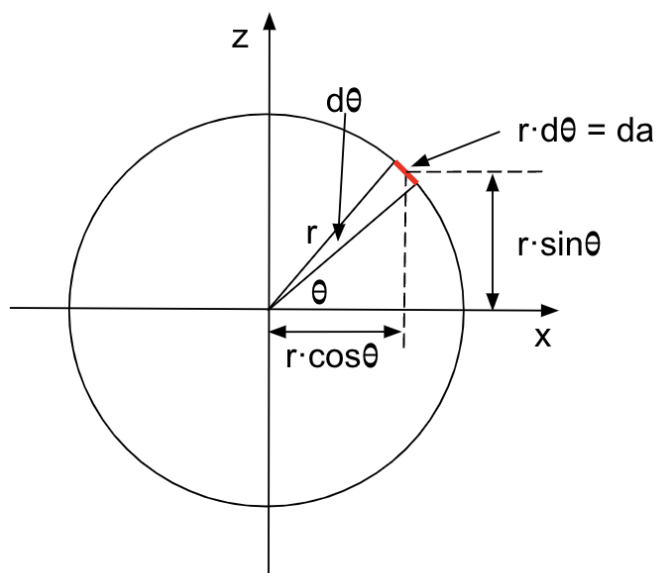
$$R^2_{112} = R^\varphi_{rr \varphi}$$

$$R^2_{121} = R^\varphi_{r \varphi r}$$

3.2 Oppervlakken van bol en donut

3.2.1 2-sfeer

Het eerste diagram laat een sfeer van de zijkant zien. Op het tweede plaatje zie je de sfeer van boven.



De sfeer heeft een straal r . Iedere punt op het oppervlak van de sfeer kan door de radius r en de twee hoeken θ en φ beschreven worden. De hoek θ ($= q^1$) is de breedtegraad en loopt van $-\pi/2$ tot $+\pi/2$ (positieve hoeken = noorderbreedte en negatieve hoeken = zuiderbreedte). De andere hoek φ ($= q^2$) is de lengtegraad en loopt van $-\pi$ tot $+\pi$ (negatieve hoeken = westerlengte en positieve hoeken = oosterlengte).

De cartesische coördinaten van de punten op een sfeer zijn:

$$\begin{aligned}x &= r \cdot \cos \theta \cdot \cos \varphi \\y &= r \cdot \cos \theta \cdot \sin \varphi \\z &= r \cdot \sin \theta\end{aligned}$$

Een klein stukje ds op het oppervlak van de sfeer kunnen we weer met de stelling van Pythagoras berekenen: $ds^2 = da^2 + db^2$. Het radiale stukje $da = r \cdot d\theta$ en het azimutale stukje $db = r \cos \theta d\varphi$ zijn in de diagrammen te zien. Het geldt dus voor ds :

$$ds^2 = r^2 d\theta^2 + r^2 \cos^2 \theta d\varphi^2 \quad (30)$$

De componenten van de metrische tensor en de inverse tensor zijn dan:

$$\begin{aligned}g_{11} &= g_{\theta\theta} = r^2 \\g_{12} &= g_{\theta\varphi} = 0 \\g_{21} &= g_{\varphi\theta} = 0 \\g_{22} &= g_{\varphi\varphi} = r^2 \cos^2 \theta\end{aligned}$$

$$\begin{aligned}g^{11} &= g^{\theta\theta} = \frac{1}{r^2} \\g^{12} &= g^{\theta\varphi} = 0 \\g^{21} &= g^{\varphi\theta} = 0 \\g^{22} &= g^{\varphi\varphi} = \frac{1}{r^2 \cos^2 \theta}\end{aligned}$$

Nu kunnen we de Christoffelsymbolen (zie vergelijking (6)) berekenen. Alle zijn nul behalve de volgende drie:

$$\begin{aligned}\Gamma^1_{22} &= \Gamma^\theta_{\varphi\varphi} = \sin \theta \cdot \cos \theta \\ \Gamma^2_{12} &= \Gamma^\varphi_{\theta\varphi} = -\tan \theta \\ \Gamma^2_{21} &= \Gamma^\varphi_{\varphi\theta} = -\tan \theta\end{aligned}$$

Van de 16 componenten van de Riemann-tensor R^l_{ijk} (vergelijking (8))

zijn 4 componenten verschillend van nul:

$$\begin{aligned} R^1_{212} &= R^\theta_{\varphi\theta\varphi} = \cos^2 \theta \\ R^1_{221} &= R^\theta_{\varphi\varphi\theta} = -\cos^2 \theta \\ R^2_{112} &= R^\varphi_{\theta\theta\varphi} = -1 \\ R^2_{121} &= R^\varphi_{\theta\varphi\theta} = 1 \end{aligned}$$

Het oppervlak van een 2-sfeer is zoals verwacht niet vlak. We gaan nu verder en berekenen de Ricci-tensor R_{ik} (zie vergelijking (10)). De componenten $R_{12} = R_{\theta\varphi}$ en $R_{21} = R_{\varphi\theta}$ zijn nul, maar:

$$\begin{aligned} R_{11} &= R_{\theta\theta} = 1 \\ R_{22} &= R_{\varphi\varphi} = \cos^2 \theta \end{aligned}$$

De scalaire kromming R (vergelijking (11)) is:

$$R = \frac{2}{r^2}$$

Omdat $R = 2 \cdot K$ (zie vergelijking (12)) geldt voor de Gaussiaanse kromming K :

$$K = \frac{1}{r^2} = \frac{1}{r} \cdot \frac{1}{r}$$

De twee factoren $(1/r)$ zijn de hoofdkrommingen k_1 en k_2 .

De geodeten-vergelijkingen voor de sfeer zijn (zie vergelijking (13)):

$$\frac{d^2\theta}{dt^2} + \sin \theta \cdot \cos \theta \cdot \left(\frac{d\varphi}{dt} \right)^2 = 0 \quad (31)$$

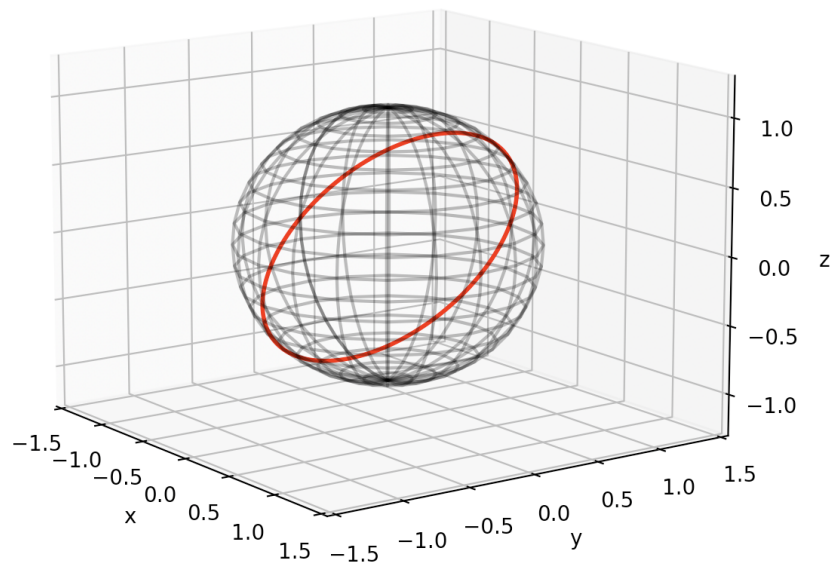
$$\frac{d^2\varphi}{dt^2} - 2 \tan \theta \cdot \frac{d\theta}{dt} \cdot \frac{d\varphi}{dt} = 0 \quad (32)$$

In hoofdstuk 6.2.5 kun je weer een programma vinden om de geodeten-vergelijkingen van de sfeer numeriek op te lossen. De meest voor de hand liggende geodeten op een sfeer zijn de evenaar en alle lengtegraden. Deze kun je bv. met de volgende beginvoorwaarden realiseren:

Geodeet	θ_0	$\dot{\theta}_0$	φ_0	$\dot{\varphi}_0$
Evenaar	0	0	0	1
Lengtegraden	0	1	tussen $-\pi$ en $+\pi$	0

Op een sfeer zijn alle grootcirkels⁴ geodeten. Dit kunnen we bv. zien wanneer we de volgende beginvoorwaarden kiezen: $\theta_0 = 0, \dot{\theta}_0 = 1, \varphi_0 = 0$ en $\dot{\varphi}_0 = 1$. Het deeltje start op de evenaar onder een hoek van 45° . Dit levert het volgende plaatje op:

Geodeet op een sfeer

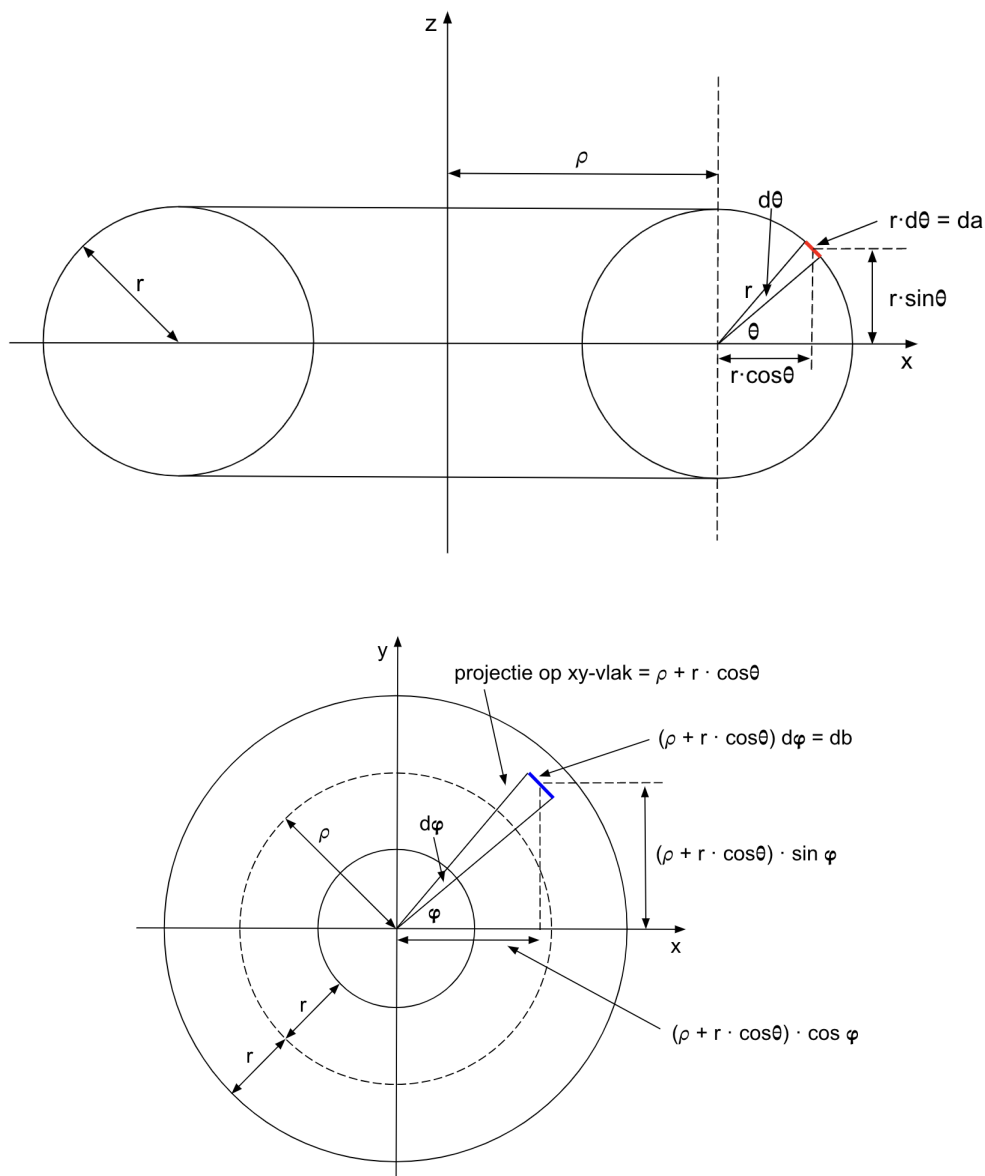


Een geodeet op een bol "bijt" altijd in zijn eigen staart. Op een andere oppervlak (bijv. torus) hoeft dat niet het geval te zijn.

⁴Het middelpunt van een grootcirkel valt samen met het middepunt van de sfeer.

3.2.2 2-torus

Het eerste diagram laat een torus van de zijkant zien. Op het tweede plaatje zie je de torus van boven.



De torus heeft een grote straal ρ . Deze gaat van de oorsprong van het coördinatenstelsel naar de middellijn van de torus. De kleine straal r beschrijft de afstand van de middellijn naar de rand van de torus. Iedere punt op het oppervlak van de torus kan door de radii ρ en r en de twee hoeken θ en φ beschreven worden. De hoek θ ($= q^1$) heeft de rol van de breedtegraad en loopt van $-\pi$ tot $+\pi$ (positieve hoeken = noorderbreedte en negatieve hoeken = zuiderbreedte). De andere hoek φ ($= q^2$) is de lengtegraad en loopt ook van $-\pi$ tot $+\pi$ (negatieve hoeken = westerlengte en positieve hoeken = oosterlengte).

Het verband met de cartesische coördinaten is als volgt:

$$\begin{aligned}x &= (\rho + r \cdot \cos \theta) \cdot \cos \varphi \\y &= (\rho + r \cdot \cos \theta) \cdot \sin \varphi \\z &= r \cdot \sin \theta\end{aligned}$$

Een klein stukje ds op het oppervlak van de torus kunnen we weer met de stelling van Pythagoras berekenen: $ds^2 = da^2 + db^2$. Het radiale stukje $da = r \cdot d\theta$ en het azimuthale stukje $db = (\rho + r \cos \theta)d\varphi$ zijn in de diagrammen te zien. Het geldt dus voor ds :

$$ds^2 = r^2 d\theta^2 + (\rho + r \cos \theta)^2 d\varphi^2 \quad (33)$$

De componenten van de metrische tensor en de inverse tensor zijn dan:

$$\begin{aligned}g_{11} &= g_{\theta\theta} = r^2 \\g_{12} &= g_{\theta\varphi} = 0 \\g_{21} &= g_{\varphi\theta} = 0 \\g_{22} &= g_{\varphi\varphi} = (\rho + r \cos \theta)^2\end{aligned}$$

$$\begin{aligned}g^{11} &= g^{\theta\theta} = \frac{1}{r^2} \\g^{12} &= g^{\theta\varphi} = 0 \\g^{21} &= g^{\varphi\theta} = 0 \\g^{22} &= g^{\varphi\varphi} = \frac{1}{(\rho + r \cos \theta)^2}\end{aligned}$$

Nu kunnen we de Christoffelsymbolen (zie vergelijking (6)) berekenen. Alle zijn nul behalve de volgende drie:

$$\begin{aligned}\Gamma^1_{22} &= \Gamma^{\theta}_{\varphi\varphi} = \frac{1}{r} \cdot \sin \theta \cdot (\rho + r \cos \theta) \\ \Gamma^2_{12} &= \Gamma^{\varphi}_{\theta\varphi} = -\frac{r \cdot \sin \theta}{(\rho + r \cos \theta)} \\ \Gamma^2_{21} &= \Gamma^{\varphi}_{\varphi\theta} = -\frac{r \cdot \sin \theta}{(\rho + r \cos \theta)}\end{aligned}$$

Van de 16 componenten van de Riemann-tensor R^l_{ijk} (vergelijking (8))

zijn 4 componenten verschillend van nul:

$$\begin{aligned}
R^1_{212} &= R^{\theta}_{\varphi\theta\varphi} = \frac{1}{r} \cdot \cos \theta \cdot (\rho + r \cos \theta) \\
R^1_{221} &= R^{\theta}_{\varphi\varphi\theta} = -\frac{1}{r} \cdot \cos \theta \cdot (\rho + r \cos \theta) \\
R^2_{112} &= R^{\varphi}_{\theta\theta\varphi} = -\frac{r \cdot \cos \theta}{\rho + r \cos \theta} \\
R^2_{121} &= R^{\varphi}_{\theta\varphi\theta} = \frac{r \cdot \cos \theta}{\rho + r \cos \theta}
\end{aligned}$$

Het oppervlak van een 2-torus is dus zoals ook bij de 2-sfeer niet vlak. We gaan nu verder en berekenen de Ricci-tensor R_{ik} (zie vergelijking (10)). De componenten $R_{12} = R_{\theta\varphi}$ en $R_{21} = R_{\varphi\theta}$ zijn nul.

$$\begin{aligned}
R_{11} &= R_{\theta\theta} = \frac{r \cdot \cos \theta}{\rho + r \cos \theta} \\
R_{22} &= R_{\varphi\varphi} = \frac{1}{r} \cdot \cos \theta \cdot (\rho + r \cos \theta)
\end{aligned}$$

De scalaire kromming R (vergelijking (11)) is:

$$R = \frac{2 \cos \theta}{r \cdot (\rho + r \cos \theta)}$$

Omdat $R = 2 \cdot K$ (zie vergelijking (12)) geldt voor de Gaussiaanse kromming K :

$$K = \frac{\cos \theta}{r \cdot (\rho + r \cos \theta)} = \frac{1}{r} \cdot \frac{\cos \theta}{\rho + r \cos \theta}$$

De eerste factor ($k_1 = 1/r$) geeft de hoofdkromming van de lengtegraad weer. De tweede factor is de hoofdkromming $k_2 = \cos \theta / (\rho + r \cos \theta)$.

Het volgende filmpje laat heel goed zien hoe de hoofdkromming k_2 op een torus verandert wanneer we langs een lengtegraad lopen:

<https://www.youtube.com/watch?v=VEPqKOPttbU>

De geodeten-vergelijkingen voor de torus zijn (zie vergelijking (13)):

$$\frac{d^2\theta}{dt^2} + \frac{1}{r} \cdot \sin \theta \cdot (\rho + r \cos \theta) \cdot \left(\frac{d\varphi}{dt}\right)^2 = 0 \quad (34)$$

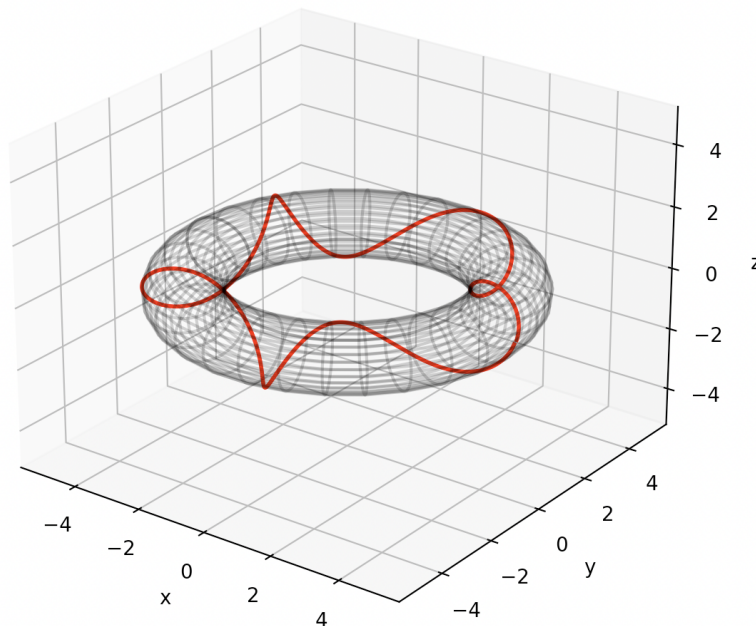
$$\frac{d^2\varphi}{dt^2} - \frac{2r \sin \theta}{\rho + r \cos \theta} \cdot \frac{d\theta}{dt} \cdot \frac{d\varphi}{dt} = 0 \quad (35)$$

In hoofdstuk 6.2.6 kun je weer een programma vinden om de geodetenvergelijkingen van de torus numeriek op te lossen. De meest voor de hand liggende geodeten op een torus zijn de twee evenaren (buiten en binnen) en alle lengtegraden. Deze kun je bv. met de volgende beginvoorwaarden realiseren ($t_{\text{end}} = 7$):

Geodeet	θ_0	$\dot{\theta}_0$	φ_0	$\dot{\varphi}_0$
Buitenevenaar	0	0	0	1
Binnenevenaar	π	0	0	1
Lengtegraden	0	1	tussen $-\pi$ en $+\pi$	0

Wanneer beide beginvoorwaarden voor de hoeksnelheden $\dot{\theta}_0$ en $\dot{\varphi}_0$ verschillend van nul zijn, krijg je geodeten die op een meer complexe wijze rond de torus lopen. Je hebt bv. geodeten die zowel de buiten- als ook binnenevenaar kruisen en in zich gesloten zijn. De figuur onderaan laat een geodeet zien die 5 keer rond de torus gaat (met $\rho = 4, r = 1, \theta_0 = 0, \dot{\theta}_0 = 100, \varphi_0 = 0, \dot{\varphi}_0 = 10$):

Geodeet op een torus



4 Geodeten zonder wiskunde

In hoofdstuk 2.7 hebben we kennis gemaakt met de differentiaalvergelijkingen om geodeten op een gekromd oppervlak te berekenen. De oplossingen van deze vergelijkingen zijn vaak alleen maar met behulp van een numerieke methode te berekenen (zie hoofdstuk 6.1).

Maar het is ook mogelijk om helemaal zonder wiskunde geodeten op een gekromd oppervlak te bepalen. Wat je hiervoor nodig hebt is bv. een piepschuim-model van het oppervlak, een smalle band en wat speldjes om het band op het model te fixeren. Voor een geodeet is het noodzakelijk dat het band vlak op het oppervlak komt te liggen (in het raakvlak). Wanneer het band niet meer volledig het oppervlak raakt, dan is het geen geodeet meer. Als voorbeeld is hier een geodeet op een torus met 5 windingen te zien:



5 Literatuur

Robert J. A. Lambourne:

Chapter 3 of the book **Relativity, Gravitation and Cosmology:**
Geometry and curved spacetime

Christoffel Symbols: A Complete Guide With Examples:

<https://profoundphysics.com/christoffel-symbols-a-complete-guide-with-examples/>

Christoffel symbols in terms of the metric tensor (unknown author, 2021):

<https://physicspages.com/pdf/Relativity/Christoffel%20symbols%20in%20terms%20of%20the%20metric%20tensor.pdf>

The Riemann Curvature Tensor (Jennifer Cox, 2019):

<https://digitalcommons.latech.edu/cgi/viewcontent.cgi?article=1008&context=mathematics-senior-capstone-papers>

Geodesics on a Cone (Mark L. Irons, 2007):

<http://www.rdrop.com/~half/Creations/Puzzles/cone.geodesics/index.html>

The Curvature and Geodesics of the Torus (Mark L. Irons, 2005):

<http://www.rdrop.com/~half/math/torus/torus.geodesics.pdf>

Torus Geodesics (Joel Feldman, 2007):

https://personal.math.ubc.ca/~feldman/m428/torus_geodesics.pdf

Geodesics on the Torus and other Surfaces of Revolution (Robert T. Jantzen, 2010):

<https://arxiv.org/pdf/1212.6206.pdf>

The Euler method for second order ODE's (Jeffrey Chasnov, 2010):

<https://www.youtube.com/watch?v=QuyBVdDHkZY>

Solving a System of Two Differential Equations Numerically in Python (Hugo de Groot, 2020):

<https://medium.com/analytics-vidhya/solving-a-system-of-two-differential-equations-numerically-in-python-d31844d4ea28>

6 Appendix

6.1 Methode van Euler

De vergelijkingen (14) en (15) zijn (gekoppelde) partiële differentiaalvergelijkingen van de tweede orde. De methode van Euler is de eenvoudigste methode om een numerieke oplossing te berekenen.

Hoe deze methode werkt, wordt nu toegelicht aan de hand van de differentiaalvergelijking voor de harmonische oscillator (de slinger):

$$\ddot{x} + k \cdot x = 0 \quad (36)$$

Hierbij is x de positie van de slinger. De snelheid v is:

$$v = \frac{dx}{dt} = \dot{x}$$

En het geldt voor de versnelling:

$$\ddot{x} = \frac{dv}{dt} = \dot{v}$$

Daarom kunnen we voor de vergelijking (36) schrijven:

$$\ddot{x} = \dot{v} = -k \cdot x \quad (37)$$

De beginvoorwaarden voor de beweging van de oscillator zijn de positie $x(t_0) = x_0$ en de snelheid $v(t_0) = v_0$. Na het verstreken van een klein tijdsinterval Δt kunnen we de positie en snelheid als volgt benaderen:

$$\begin{aligned} t_1 &= t_0 + \Delta t \\ x_1 &= x_0 + \Delta t \cdot v_0 \\ v_1 &= v_0 + \Delta t \cdot \dot{v}_0 = v_0 + \Delta t \cdot (-k \cdot x_0) \end{aligned}$$

Na nog een tijdsinterval Δt geldt:

$$\begin{aligned} t_2 &= t_1 + \Delta t \\ x_2 &= x_1 + \Delta t \cdot v_1 \\ v_2 &= v_1 + \Delta t \cdot \dot{v}_1 = v_1 + \Delta t \cdot (-k \cdot x_1) \end{aligned}$$

Door voortschrijdende iteratie belanden we bij de positie x_i en de

snelheid v_i :

$$t_i = t_{i-1} + \Delta t$$

$$x_i = x_{i-1} + \Delta t \cdot v_{i-1}$$

$$v_i = v_{i-1} + \Delta t \cdot \dot{v}_{i-1} = v_{i-1} + \Delta t \cdot (-k \cdot x_{i-1})$$

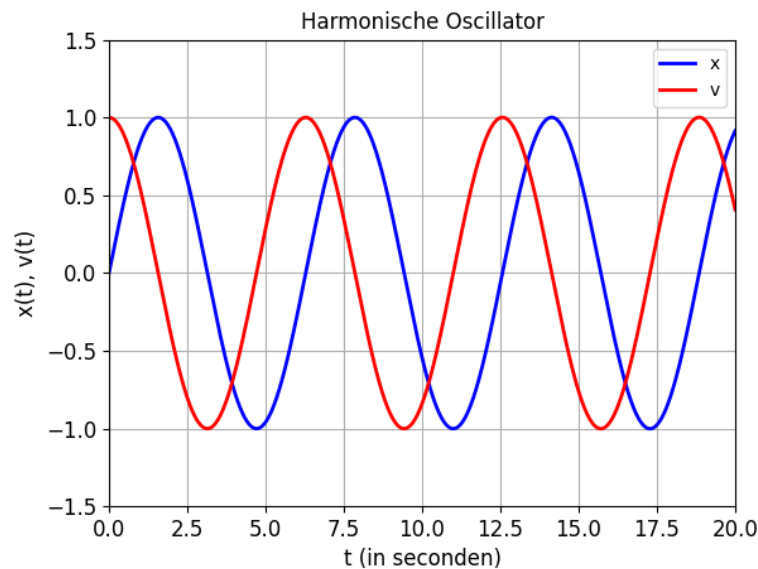
De iteratie stopt bij stap n met de positie x_n en de snelheid v_n :

$$t_n = t_{n-1} + \Delta t$$

$$x_n = x_{n-1} + \Delta t \cdot v_{n-1}$$

$$v_n = v_{n-1} + \Delta t \cdot \dot{v}_{n-1} = v_{n-1} + \Delta t \cdot (-k \cdot x_{n-1})$$

Al de waarden $x_0, x_1, \dots, x_n$ met de bijbehorende tijden $t_0, t_1, \dots, t_n$ beschrijven de beweging van de oscillator. Dit kun je met een klein python-programma berekenen (zie hoofdstuk 6.2.1) en dan grafisch weergeven. Hoe kleiner je Δt kiest hoe beter de oplossing wordt. Het is niet moeilijk om deze methode toe te passen op gekoppelde differentiaalvergelijkingen.



6.2 Python-code

6.2.1 Harmonische oscillator

```
# Harmonic oscillator
#
import numpy as np
import matplotlib.pyplot as plt
Dt = 0.00001          # timestep Delta t
x_start = 0           # initial x value (position)
v_start = 1           # initial v value (velocity)
t_start = 0           # starttime
t_end = 20            # endtime
n_steps = int(round((t_end-t_start)/Dt)) # number of timesteps
x_arr = np.zeros(n_steps + 1) # create an array of zeros for x
v_arr = np.zeros(n_steps + 1) # create an array of zeros for v
t_arr = np.zeros(n_steps + 1) # create an array of zeros for t
t_arr[0] = t_start      # add starttime to array
x_arr[0] = x_start      # add initial value of x to array
v_arr[0] = v_start      # add initial value of v to array
k = 1                  # k = (angular frequency)^2
#
# Euler method
for i in range (1, n_steps + 1):
    # calculate t at next timestep:
    t_arr[i] = t_arr[i-1] + Dt
    # calculate x at next timestep:
    x_arr[i] = x_arr[i-1] + Dt * v_arr[i-1]
    # calculate v at next timestep:
    v_arr[i] = v_arr[i-1] + Dt * (-k * x_arr[i-1])
#
# plotting the result
fig = plt.figure()
plt.plot(t_arr, x_arr, linewidth = 2, color='blue', label = 'x')
plt.plot(t_arr, v_arr, linewidth = 2, color='red', label = 'v')
plt.title('Harmonische Oscillator', fontsize = 12)
plt.xlabel('t (in seconden)', fontsize = 12)
plt.ylabel('x(t), v(t)', fontsize = 12)
plt.xticks(fontsize = 12)
plt.yticks(fontsize = 12)
plt.grid(True)
plt.axis([t_start, t_end, -1.5, 1.5])
plt.legend()
plt.show()
#
# end
```

6.2.2 Vlák

```
# Geodesic in a plane
# r = distance from origin
# phi = azimuth (0 < phi < 2*pi)
# x = r * cos(phi)
# y = r * sin(phi)
#
import numpy as np
import matplotlib.pyplot as plt
Dt = 0.0001          # timestep Delta t
r_start = 0.01        # initial r value (distance to origin)
drdt_start = 10       # initial velocity in radial direction
phi_start = 0         # initial phi value
dphidt_start = 1e4    # initial change of phi
t_start = 0           # starttime
t_end = 2*np.pi      # endtime
n_steps = int(round((t_end-t_start)/Dt)) # number of timesteps
r_arr = np.zeros(n_steps + 1)          # array of zeros for r
drdt_arr = np.zeros(n_steps + 1)        # array of zeros for drdt
phi_arr = np.zeros(n_steps + 1)         # array of zeros for phi
dphidt_arr = np.zeros(n_steps + 1)      # array of zeros for dphidt
r_arr[0] = r_start                      # initial value of r
drdt_arr[0] = drdt_start                 # initial value of drdt
phi_arr[0] = phi_start                   # initial value of phi
dphidt_arr[0] = dphidt_start             # initial value of dphidt
#
# Calculating geodesic with Euler method
for i in range (1, n_steps + 1):
    r = r_arr[i-1]
    drdt = drdt_arr[i-1]
    phi = phi_arr[i-1]
    dphidt = dphidt_arr[i-1]
    # calculate Christoffelsymbols:
    Gamma_rpp = - r
    Gamma_prp = 1 / r
    # calculate r at next timestep:
    r_arr[i] = r_arr[i-1] + Dt * drdt
    # calculate drdt at next timestep:
    drdt_arr[i] = drdt + Dt * (- Gamma_rpp * (dphidt**2))
    # calculate phi at next timestep:
    phi_arr[i] = phi + Dt * dphidt
    # calculate dphidt at next timestep:
    dphidt_arr[i] = dphidt + Dt * (- 2 * Gamma_prp * dphidt*drdt)
#
# plotting the result
fig = plt.figure()
plt.axes(projection='polar')
plt.polar(phi_arr, r_arr, 'red', linewidth = 2)
plt.title('Geodeet in een vlak', fontsize = 12)
plt.grid(True)
plt.axis([0, 2*np.pi, 0, 50])
#
plt.show()
#
# end
```

6.2.3 Kegel

```
# Geodesic on a cone-surface
# r = distance from z-axis
# phi = azimuth (-pi < phi < pi)
# x = r * cos(phi)
# y = r * sin(phi)
# z = h - sigma * r (h = height of apex, sigma = steepness)
#
import numpy as np
import matplotlib.pyplot as plt
Dt = 0.0001          # timestep Delta t
h = 1.0              # h = height op apex above xy-plane
sigma = 2.0          # sigma = steepness
r_start = h / sigma  # initial r value (at the edge of cone)
drdt_start = -10.0   # initial change rate of r
phi_start = 0.0       # initial phi value
dphidt_start = 5.0    # initial change rate of phi
t_start = 0.0         # starttime
t_end = 0.098         # endtime
n_steps = int(round((t_end-t_start)/Dt)) # number of timesteps
r_arr = np.zeros(n_steps + 1)          # array of zeros for r
drdt_arr = np.zeros(n_steps + 1)        # array of zeros for drdt
phi_arr = np.zeros(n_steps + 1)         # array of zeros for phi
dphidt_arr = np.zeros(n_steps + 1)      # array of zeros for dphidt
r_arr[0] = r_start                      # initial value of r
drdt_arr[0] = drdt_start                 # initial value of drdt
phi_arr[0] = phi_start                   # initial value of phi
dphidt_arr[0] = dphidt_start             # initial value of dphidt
#
# Calculating geodesic with Euler method
for i in range (1, n_steps + 1):
    r = r_arr[i-1]
    drdt = drdt_arr[i-1]
    phi = phi_arr[i-1]
    dphidt = dphidt_arr[i-1]
    # calculate Christoffelsymbols:
    Gamma_rpp = - r / (sigma**2 + 1)
    Gamma_prp = 1 / r
    # calculate r at next timestep:
    r_arr[i] = r + Dt * drdt
    # calculate drdt at next timestep:
    drdt_arr[i] = drdt + Dt * (- Gamma_rpp * (dphidt**2))
    # calculate phi at next timestep:
    phi_arr[i] = phi + Dt * dphidt
    # calculate dphidt at next timestep:
    dphidt_arr[i] = dphidt + Dt * (- 2 * Gamma_prp * dphidt*drdt)
#
# plotting the result in 3D
fig = plt.figure()
ax = plt.axes(projection='3d')
plt.title("Geodeet op een kegel-oppervlak")
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
ax.set_zlim(0,h)
```

```

#
# plot the cone
r_cone, phi_cone = np.mgrid[0:h/sigma:21j, -np.pi:np.pi:21j]
x_cone = r_cone * np.cos(phi_cone)
y_cone = r_cone * np.sin(phi_cone)
z_cone = h - sigma * r_cone
ax.plot_wireframe(x_cone, y_cone, z_cone, color='black', alpha=0.3)
#
# plot the geodesic
x_geodesic = r_arr * np.cos(phi_arr)
y_geodesic = r_arr * np.sin(phi_arr)
z_geodesic = h - sigma * r_arr
ax.plot3D(x_geodesic, y_geodesic, z_geodesic, 'red', linewidth=2)
#
plt.show()
#
# end

```

6.2.4 Logarithmische trechter

```

# Geodesic on a funnel-surface
# r = distance from z-axis
# phi = azimuth (-pi < phi < pi)
# x = r * cos(phi)
# y = r * sin(phi)
# z = ln(r)
#
import numpy as np
import matplotlib.pyplot as plt
Dt = 0.00001          # timestep Delta t
r_start = 4           # initial r value
drdt_start = -10      # initial change rate of r
phi_start = 0          # initial phi value
dphidt_start = 0.3    # initial change rate of phi
t_start = 0.0          # starttime
t_end = 0.97          # endtime
n_steps = int(round((t_end-t_start)/Dt)) # number of timesteps
r_arr = np.zeros(n_steps + 1)          # array of zeros for r
drdt_arr = np.zeros(n_steps + 1)        # array of zeros for drdt
phi_arr = np.zeros(n_steps + 1)         # array of zeros for phi
dphidt_arr = np.zeros(n_steps + 1)      # array of zeros for dphidt
r_arr[0] = r_start                      # initial value of r
drdt_arr[0] = drdt_start                 # initial value of drdt
phi_arr[0] = phi_start                   # initial value of phi
dphidt_arr[0] = dphidt_start             # initial value of dphidt
#
# Calculating geodesic with Euler method
for i in range(1, n_steps + 1):
    r = r_arr[i-1]
    drdt = drdt_arr[i-1]
    phi = phi_arr[i-1]
    dphidt = dphidt_arr[i-1]
    # calculate Christoffelsymbols:
    Gamma_rrr = - 1 / (r**3 + r)

```

```

Gamma_rpp = - r**3 / (r**2 + 1)
Gamma_prp = 1 / r
# calculate r at next timestep:
r_arr[i] = r + Dt*drdt
# calculate drdt at next timestep:
drdt_arr[i] = \
drdt + Dt * (- Gamma_rrr*(drdt**2) - Gamma_rpp*(dphidt**2))
# calculate phi at next timestep:
phi_arr[i] = phi + Dt * dphidt
# calculate dphidt at next timestep:
dphidt_arr[i] = dphidt + Dt * (- 2 * Gamma_prp * dphidt*drdt)
#
# plotting the result in 3D
fig = plt.figure()
ax = plt.axes(projection='3d')
plt.title("Geodeet op een logaritmische trechter")
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
#
# plot the funnel
r_funnel, phi_funnel = np.mgrid[0.1:4:21j, -np.pi:np.pi:21j]
x_funnel = r_funnel * np.cos(phi_funnel)
y_funnel = r_funnel * np.sin(phi_funnel)
z_funnel = np.log(r_funnel)
ax.plot_wireframe(x_funnel, y_funnel, z_funnel, color='black',
                  alpha=0.3)
#
# plot the geodesic
x_geodesic = r_arr * np.cos(phi_arr)
y_geodesic = r_arr * np.sin(phi_arr)
z_geodesic = np.log(r_arr)
ax.plot3D(x_geodesic, y_geodesic, z_geodesic, 'red', linewidth=2)
#
plt.show()
#
# end

```

6.2.5 2-sfeer

```

# Geodesic on a 2-sfeer
# theta = latitude (-pi/2 < theta < pi/2)
# phi = longitude (-pi < phi < pi)
# The radius of the 2-sfeer is 1.
# x = r * cos(theta) * cos(phi)
# y = r * cos(theta) * sin(phi)
# z = r * sin(theta)
#
import numpy as np
import matplotlib.pyplot as plt
Dt = 0.0001          # timestep Delta t
theta_start = 0      # initial theta value
dthetadt_start = 1   # initial change rate of theta
phi_start = 0        # initial phi value

```

```

dphidt_start = 1          # initial change rate of phi
t_start = 0               # starttime
t_end = np.sqrt(2)*np.pi # endtime
n_steps = int(round((t_end-t_start)/Dt)) # number of timesteps
theta_arr = np.zeros(n_steps + 1)      # array of zeros for theta
dthetadt_arr = np.zeros(n_steps + 1)    # array of zeros for dthetadt
phi_arr = np.zeros(n_steps + 1)         # array of zeros for phi
dphidt_arr = np.zeros(n_steps + 1)      # array of zeros for dphidt
theta_arr[0] = theta_start               # initial value of theta
dthetadt_arr[0] = dthetadt_start         # initial value of dthetadt
phi_arr[0] = phi_start                   # initial value of phi
dphidt_arr[0] = dphidt_start             # initial value of dphidt
#
# Calculating geodesic with Euler method
for i in range(1, n_steps + 1):
    theta = theta_arr[i-1]
    dthetadt = dthetadt_arr[i-1]
    phi = phi_arr[i-1]
    dphidt = dphidt_arr[i-1]
    # calculate Christoffelsymbols:
    Gamma_tpp = np.sin(theta) * np.cos(theta)
    Gamma_ptp = - np.tan(theta)
    # calculate theta at next timestep:
    theta_arr[i] = theta + Dt*dthetadt
    # calculate dthetadt at next timestep:
    dthetadt_arr[i] = dthetadt + Dt*(- Gamma_tpp * (dphidt**2))
    # calculate phi at next timestep:
    phi_arr[i] = phi + Dt*dphidt
    # calculate dphidt at next timestep:
    dphidt_arr[i] = dphidt + Dt*(- 2*Gamma_ptp) * dthetadt*dphidt
#
# plotting the result in 3D
fig = plt.figure()
ax = plt.axes(projection='3d')
plt.title("Geodeet op een sfeer")
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
ax.set_xlim(-1.5,1.5)
ax.set_ylim(-1.5,1.5)
ax.set_zlim(-1.25,1.25)
#
# plot a 2-sphere
theta_sphere, phi_sphere = np.mgrid[-np.pi/2:np.pi/2:21j, -np.pi:np
                                     .pi:21j]
x_sphere = np.cos(theta_sphere) * np.cos(phi_sphere)
y_sphere = np.cos(theta_sphere) * np.sin(phi_sphere)
z_sphere = np.sin(theta_sphere)
ax.plot_wireframe(x_sphere, y_sphere, z_sphere, color='black',
                  alpha=0.3)
#
# plot the geodesic
x_geodesic = np.cos(theta_arr)*np.cos(phi_arr)
y_geodesic = np.cos(theta_arr)*np.sin(phi_arr)
z_geodesic = np.sin(theta_arr)
ax.plot3D(x_geodesic, y_geodesic, z_geodesic, 'red', linewidth=2)

```

```
#
plt.show()
#
# end
```

6.2.6 2-torus

```
# Geodesic on a torus
# theta = latitude (-pi < theta < pi)
# phi = longitude (-pi < phi < pi)
# major radius = rho
# minor radius = r
# x = (rho + r * cos(theta)) * cos(phi)
# y = (rho + r * cos(theta)) * sin(phi)
# z = r * sin(theta)
#
import numpy as np
import matplotlib.pyplot as plt
Dt = 0.00001          # timestep Delta t
rho = 4.0              # major radius of torus
r = 1.0               # minor radius of torus
theta_start = 0.0      # initial theta value
dthetadt_start = 100.0 # initial change rate of theta
phi_start = 0.0        # initial phi value
dphidt_start = 10.0    # initial change rate of phi
t_start = 0.0          # starttime
t_end = 0.351          # endtime
n_steps = int(round((t_end-t_start)/Dt)) # number of timesteps
theta_arr = np.zeros(n_steps + 1)      # array of zeros for theta
dthetadt_arr = np.zeros(n_steps + 1)    # array of zeros for dthetadt
phi_arr = np.zeros(n_steps + 1)         # array of zeros for phi
dphidt_arr = np.zeros(n_steps + 1)      # array of zeros for dphidt
theta_arr[0] = theta_start               # initial value of theta
dthetadt_arr[0] = dthetadt_start         # initial value of dthetadt
phi_arr[0] = phi_start                   # initial value of phi
dphidt_arr[0] = dphidt_start             # initial value of dphidt
#
# Calculating geodesic with Euler method
for i in range (1, n_steps + 1):
    theta = theta_arr[i-1]
    dthetadt = dthetadt_arr[i-1]
    phi = phi_arr[i-1]
    dphidt = dphidt_arr[i-1]
    # calculate Christoffelsymbols:
    Gamma_tpp = (1/r) * np.sin(theta) * (rho+r*np.cos(theta))
    Gamma_ptp = - r * np.sin(theta) / (rho+r*np.cos(theta))
    # calculate theta at next timestep:
    theta_arr[i] = theta + Dt * dthetadt
    # calculate dthetadt at next timestep:
    dthetadt_arr[i] = dthetadt + Dt * (- Gamma_tpp * (dphidt**2))
    # calculate phi at next timestep:
    phi_arr[i] = phi + Dt*dphidt
    # calculate dphidt at next timestep:
    dphidt_arr[i] = dphidt + Dt * (- 2*Gamma_ptp) * dthetadt*dphidt
```

```

#
# plotting the result in 3D
fig = plt.figure()
ax = plt.axes(projection='3d')
plt.title("Geodeet op een torus")
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
ax.set_zlim(-(rho+r),(rho+r))
#
# plot the torus
theta_torus, phi_torus = np.mgrid[-np.pi:np.pi:31j, -np.pi:np.pi:
                                     31j]
x_torus = (rho+r*np.cos(theta_torus))*np.cos(phi_torus)
y_torus = (rho+r*np.cos(theta_torus))*np.sin(phi_torus)
z_torus = r*np.sin(theta_torus)
ax.plot_wireframe(x_torus, y_torus, z_torus, color='black', alpha=0
                  .2)
#
# plot the geodesic
x_geodesic = (rho+r*np.cos(theta_arr)) * np.cos(phi_arr)
y_geodesic = (rho+r*np.cos(theta_arr)) * np.sin(phi_arr)
z_geodesic = r * np.sin(theta_arr)
ax.plot3D(x_geodesic, y_geodesic, z_geodesic, 'red', linewidth=2)
#
plt.show()
#
# end

```

Het bovenstaande Python-programma maakt een geodeet met 5 windingen op een torus. De volgende tabel geeft de beginvoorwaarde voor $\dot{\varphi}_0$ weer voor geodeten met n windingen ($2 \leq n \leq 10$). Hierbij is altijd $\rho = 4$, $r = 1$, $\theta_0 = 0$, $\varphi_0 = 0$ en $\dot{\theta}_0 = 100,0$.

n	2	3	4	5	6	7	8	9	10
$\dot{\varphi}_0$	14,69	13,33	11,59	10,00	8,71	7,67	6,84	6,16	5,60
t_{end}	0,25	0,25	0,30	0,35	0,45	0,50	0,55	0,60	0,65