Masarykova univerzita v Brně

Přírodovědecká fakulta

Diplomová práce

Brno 1999

David Dvořák

Masarykova univerzita v Brně
Přírodovědecká fakulta
Katedra teoretické fyziky

Diplomová práce

Programový modul pro optimalizaci Petrovovy klasifikace

David Dvořák

Vedoucí DP: Prof. RNDr. Jan Horský, DrSc.

Brno
1999

Považuji za povinnost poděkovat Prof. RNDr. Janu Horskému, DrSc. za vedení závěrečné práce a Dr. Milanu Štefaníkovi za cenné rady udělené během zpracování zadaného problému.

Prohlašuji, že jsem závěrečnou práci zpracoval samostatně a uvedl veškerou použitou literaturu.

V Brně, 8. dubna 1999 *David Dvořák*

Obsah

1

Předmluva

Počítače hrají ve fyzice a matematice důležitou roli již od svého vzniku. V počátcích se využívala především jejich schopnost rychlých numerických výpočtů, ale se stále se zdokonalujícím technickým a softwarovým zázemím jsou nasazovány na stále složitější úkoly, čímž podstatně ulehčují práci v mnoha oborech lidské činnosti. Ve fyzice jsou v současnosti nasazovány jak v experimentální, tak i v teoretické oblasti. V experimentální fyzice se používají k řízení celého pokusu (např. u experimentů prováděných ve švýcarském CERNu) a jeho následného statistického zpracování. V teoretické fyzice mají své místo jak numerické, tak i symbolické výpočty, které umožňují kromě základních matematických operací také operace derivování, integrování apod., přičemž se pracuje s abstraktními symboly. Je to dáno tím, že mnoho abstraktních výpočtů spočívá v úpravách výrazů na jiné prostřednictvím jednoznačně předepsaných postupů. Tyto činnosti se však dají velice dobře algoritmizovat a počítač, pracující podle programu, pak prakticky nemůže udělat chybu.

Diplomová práce je zaměřena na vytvoření programu pro optimalizaci Petrovovy klasifikace prostoročasů. K vyřešení tohoto úkolu byl vybrán softwarový produkt *Maple V release 4*, který patří spolu s programem *Mathematica* k současným špičkám v oblasti symbolických výpočtů. Jeho velkou předností je programovatelnost, při níž lze využít již nadefinované funkce a procedury. Těchto vlastností je v práci maximálně využito, aby byly nové procedury co nejefektivnější a tedy i dostatečně rychlé.

Diplomová práce měla být doplněním práce Aleše Michálka, který naprogramoval modul pro práci s tenzorovými objekty pod *Maple V release 3*. Tento modul je však s verzí *4* nekompatibilní, a proto je na něm předkládaná práce nezávislá (využívá standardního mapleovského balíku pro práci s tenzory, který nebyl ve starší verzi přítomen).

Hlavní text práce je rozdělen na dvě části. V první je vyložena nejnutnější část teorie k pochopení celého problému a ve druhé jsou uvedeny programové procedury, potřebné k výpočtu Petrovovy klasifikace metrik.

2

Matematické základy

Základní vlastností a klíčem k pochopení podstaty gravitace je univerzálnost. Označujeme tím tu vlastnost, že všechna tělesa padají v gravitačním poli se stejným zrychlením, nezávisle na svém složení a hmotnosti. Tento fakt byl znám již od doby Galilea, ale teprve Albert Einstein pochopil, že je to ta nejdůležitější vlastnost gravitace, vyvodil z toho fyzikální důsledky a v letech 1908–1915 tak vybudoval obecnou teorii relativity (OTR), která je dodnes tou nejlepší teorií gravitace. Einstein zobecnil univerzálnost gravitačního působení na všechny fyzikální jevy v principu ekvivalence, který je základním kamenem obecné teorie relativity. Tento princip nám říká, že *gravitační pole je pro všechny fyzikální děje v každém místě lokálně ekvivalentní situaci, kdy není žádné gravitační pole, ale vztažná soustava se v tomto bodě pohybuje s příslušným zrychlením (tj. je neinerciální)*.

Jinými slovy, obecná teorie relativity je schopna popsat fyzikální jevy v libovolné vztažné soustavě (speciální teorie relativity (STR) platí pro inerciální vztažné soustavy (IVS)). Další analýzou dostáváme, že v libovolném gravitačním poli lze zavést řadu lokálně inerciálních vztažných soustav (v každém bodě) v nichž je stav beztíže a všechny fyzikální děje v nich probíhají lokálně podle zákonů STR bez gravitace. Ve skutečném (nehomogenním) gravitačním poli však obecně nelze spojit jednotlivé IVS v jednu globální inerciální soustavu – prostoročas není rovinný, ale zakřivený. Univerzálnost gravitačního působení vyjádřená v principu ekvivalence tak vede k souvislosti mezi gravitací a geometrií.

2.1. Základní pojmy

Cílem této kapitoly je stručný výklad potřebných vztahů a veličin nutných k popisu Petrovovy klasifikace prostoročasů. Všechny zde uvedené vztahy jsou převzaty z knihy *Matematická teorie černých děr* od S. Chandrasekhara [5]. Detailnější popis matematického aparátu OTR nalezne čtenář např. v [3], [6], [7].

Diferencovatelná varieta

Popisem vlastností Riemannova prostoročasu se zabývá matematická disciplína diferenciální geometrie. Základním objektem je diferencovatelná varieta. Než se ale přistoupí k její definici, je nutné nadefinovat pojmy diferencovatelný atlas a lokální soustava souřadnic.

Nechť n je přirozené číslo, množina N nechť je sjednocením všech svých podmnožin $V_i, i \in I$, tj. $N = \cup_{i\in I} V_i$, a nechť $\exists \forall\, V_i$ vzájemně jednoznačné zobrazení φ_i množiny V_i na otevřenou množinu $U_i \subset \mathbf{R}^n$. Nechť je pro $\forall i,j \in I$ množina $\varphi_i(V_i \cap V_j)$ otevřená a zobrazení $\varphi_j \circ \varphi_i^{-1}$ diferencovatelné zobrazení této množiny na množinu $\varphi_j(V_i \cap V_j)$. Pak je na N dána diferencovatelná struktura a systém $\{(V_i, \varphi_i)\}_{i\in I}$ je diferencovatelný atlas. Každá dvojice $(V_i, \varphi_i), i \in I$ je mapa tohoto atlasu. V_i je souřadnicové okolí a zobrazení φ_i lokální soustava souřadnic na V_i.

N je Hausdorffův prostor, jestliže ke $\forall A, B \in N$; $A \neq B$, $\exists\, U, V \subset N$ $(M,\ N$ jsou otevřené množiny$)$ tak, že $A \in U$, $B \in V$, $U \cap V = 0$. Jinými slovy, je možné libovolné dva body oddělit otevřenými množinami.

Nyní se konečně dostáváme k definici variety: *diferencovatelná varieta* je prostor N s diferencovatelnou strukturou a spočetnou bází, který je Hausdorffův. Dimenze variety je n. Spočetnou bází se myslí fakt, že z atlasu variety lze vybrat konečný systém map, které pokrývají celý prostor N.

Tečné vektory

K zavedení tenzorů je nutné nadefinovat tečný a duální prostor. Tečný prostor nadefinujeme pomocí tečných vektorů.

Tečný vektor $\mathbf{u}$ variety N v bodě $A \in V$ je operátor, který každé diferencovatelné funkci f na N přiřadí číslo $\mathbf{u}f$, které je v souřadnicích dáno vztahem

$$\mathbf{u}f = \frac{\partial f(a^1, \dots, a^n)}{\partial x^i} u^i,$$

kde $[a^1, \dots, a^n] = \varphi(A)$ jsou souřadnice bodu A a $u^1, \dots, u^n$ jsou reálná čísla, nezávislá na f. Číslo $\mathbf{u}f$ je derivace funkce f podle vektoru $\mathbf{u}$. Všechny tečné vektory variety N v bodě A tvoří tečný prostor, který označujeme $T_A N$.

Pro $\mathbf{u}$ platí, že $\mathbf{u} = u^i \partial / \partial x^i$ a pro dvě diferencovatelné funkce f, h v bodě A platí

$$\mathbf{u}(f+h) = \mathbf{u}f + \mathbf{u}h,$$
$$\mathbf{u}(fh) = (\mathbf{u}f)h + f(\mathbf{u}h).$$

1-formy

1-forma ω v bodě A variety N je lineární zobrazení tečného prostoru $T_A N$ na množinu reálných čísel

$$\omega : T_A N \to R^1.$$

Jinak řečeno, libovolnému tečnému vektoru v bodě A přiřazuje 1-forma ω číslo $\omega(X)$, což se zapisuje vztahem

$$\omega(X) = \langle \omega, X \rangle.$$

1-formy splňují vlastnosti vektorového prostoru

$$\begin{aligned} \langle \omega, \alpha X + \beta Y \rangle &= \alpha \langle \omega, X \rangle + \beta \langle \omega, X \rangle, \\ (\alpha\omega)(X) &= \alpha \langle \omega, X \rangle, \\ (\omega + \pi)(X) &= \langle \omega, X \rangle + \langle \pi, X \rangle. \end{aligned}$$

pro libovolný tečný vektor $X \in T_A N$ a libovolná reálná čísla α, β; ω, π – dvě 1-formy. Jelikož 1-formy splňují tyto vlastnosti, tvoří vektorový prostor $T_A^* N$, který se nazývá duální prostor k $T_A N$. 1-formám se také říká *kovariantní* vektory.

Tenzory a tenzorové operace

Tenzor je jedním z nejdůležitějších matematických objektů v OTR. Je možné ho nadefinovat několika způsoby. V obecné relativitě se však vychází z pojmu diferencovatelná varieta, na níž se zavede tečný prostor, k němu odpovídající duální prostor forem a pomocí nich se definuje tenzor [5], [6].

Tenzor $\mathbf{T}$ typu (r, s) nebo stupně $(r+s)$ v bodě A je element součinu prostorů

$$T_A(r, s) = \underbrace{T_A \otimes \cdots \otimes T_A}_{r} \otimes \underbrace{T_A^* \otimes \cdots \otimes T_A^*}_{s},$$

který zobrazuje libovolnou uspořádanou množinu r 1-forem a s vektorů $(\sigma^1, \ldots, \sigma^r; v_1, \ldots, v_s)$ v bodě A na reálné číslo. Tenzor $u_1 \otimes \cdots \otimes u_r \otimes \tau^1 \otimes \cdots \otimes \tau^s$ tedy zobrazuje tuto uspořádanou množinu na úžení

$\langle\sigma^1 u_1\rangle\ldots\langle\sigma^r u_r\rangle\langle\tau^1 v_1\rangle\ldots\langle\tau^s v_s\rangle$, přičemž toto zobrazení je lineární v každém argumentu. Pomocí báze $\{e_a\}$, $\{\omega^b\}$ je možné zapsat libovolný tenzor $\mathbf{T}$ typu (r,s) ve tvaru sumy tenzorových součinů

$$\mathbf{T} = T^{a_1\ldots a_r}_{b_1\ldots b_s} e_{a_1} \otimes \ldots \otimes e_{a_r} \otimes \omega^{b_1} \otimes \ldots \otimes \omega^{b_s},$$

kde všechny indexy probíhají od 1 do n. Koeficienty $T^{a_1\ldots a_r}_{b_1\ldots b_s}$ s *kovariantními* indexy $b_1\ldots b_s$ a *kontravariantními* indexy $a_1\ldots a_r$ se nazývají *komponenty* tenzoru $\mathbf{T}$.

Tenzorové pole r-krát kovariantní a s-krát kontravariantní na varietě N je zobrazení, které každému bodu $X\in N$ přiřadí tenzor typu (s,r) na $T_X N$.

Součet tenzorů

Součet tenzorů je operace, která je definována pouze pro tenzory téhož typu. Platí $T_1+T_2\to T_3$, kde T_1, T_2, T_3 jsou tenzory typu (r,s). Pro složky platí toto pravidlo:

$$A^{\alpha\ldots\delta}_{\lambda\ldots\nu} + B^{\alpha\ldots\delta}_{\lambda\ldots\nu} = C^{\alpha\ldots\delta}_{\lambda\ldots\nu},$$

přičemž $A^{\alpha\ldots\delta}_{\lambda\ldots\nu}\in T_1$, $B^{\alpha\ldots\delta}_{\lambda\ldots\nu}\in T_2$, a $C^{\alpha\ldots\delta}_{\lambda\ldots\nu}\in T_3$. Je vidět, že výsledný tenzor je stejného typu a operace sčítání tenzorů je komutativní.

Násobení skalárem

Jedná se o opreaci $T\cdot T_1\to T_2$, kde T_1, T_2 jsou tenzory typu (r,s) a T je skalár (tenzor typu $(0,0)$). Při této operaci je každá složka tenzoru vynásobena skalárem a nemění se při ní řád tenzoru:

$$(a)A^{\alpha\ldots\delta}_{\lambda\ldots\nu} = a\cdot A^{\alpha\ldots\delta}_{\lambda\ldots\nu}; \qquad a\in T,\ A^{\alpha\ldots\delta}_{\lambda\ldots\nu}\in T_1.$$

Tenzorový součin

Je operace $T_1\otimes T_2\to T_3$, přičemž T_1 je tenzor typu (r,s), T_2 je typu (u,v) a výsledný tenzor T_3 je typu $(r+u,s+v)$. Pro složky platí vztahy:

$$A^{\alpha\ldots}_{\lambda\ldots}D^{\varrho\ldots}_{\varphi\ldots} = D^{\varrho\ldots}_{\varphi\ldots}A^{\alpha\ldots}_{\lambda\ldots}, \qquad \text{(komutativita)}$$

$$A^{\alpha\ldots}_{\lambda\ldots}(D^{\varrho\ldots}_{\varphi\ldots}E^{\zeta\ldots}_{\xi\ldots}) = (A^{\alpha\ldots}_{\lambda\ldots}D^{\varrho\ldots}_{\varphi\ldots})E^{\zeta\ldots}_{\xi\ldots}, \qquad \text{(asociativita)}$$

$$(A^{\alpha\ldots}_{\lambda\ldots}+B^{\alpha\ldots}_{\lambda\ldots})D^{\varrho\ldots}_{\varphi\ldots} = A^{\alpha\ldots}_{\lambda\ldots}D^{\varrho\ldots}_{\varphi\ldots}+B^{\alpha\ldots}_{\lambda\ldots}D^{\varrho\ldots}_{\varphi\ldots}, \qquad \text{(distributivita)}$$

kde $A^{\alpha\dots}_{\lambda\dots} \in T_1$, $D^{\varrho\dots}_{\varphi\dots} \in T_2$. Součin tenzorů je operace, která mění řád tenzoru.

2.2. Relativistické objekty

V této části uvedu základní tenzorové veličiny, které jsou potřebné k zavedení *Einsteinových gravitačních rovnic*. V celé této kapitole se používá souřadnicová báze.

V obecné teorii relativity se pracuje s tenzory na varietě. V konkrétních fyzikálních případech se pak varietou myslí prostoročas (např. tvořící náš vesmír) pro definovaný zdroj gravitačního pole. Podle toho se pak příslušná řešení Einsteinových rovnic dělí na *vnitřní* a *vnější* řešení.

Geometrie prostoročasu je určena přítomností hmoty, která jej zakřivuje. Toto zakřivení pak definuje vzdálenost mezi body a způsob pohybu hmotných těles v prostoročase. Vzdálenost je definována pomocí intervalu

$$\mathrm{d}s^2 = g_{ij}\,\mathrm{d}x^i \otimes \mathrm{d}x^j .$$

Symboly $\mathrm{d}x^i$ jsou lineární formy, které ve fyzikálním smyslu představují délkový element v příslušném souřadném systému, g_{ij} je metrický tenzor, kterým je určena geometrie prostoročasu. V celé práci se dále uvažuje signatura metrického tenzoru rovna -2.

Pro další počítání je nutné znát lineární konexi. V OTR se složkám konexe v souřadnicové bázi říká Christoffelovy symboly. Tyto složky se netransformují jako tenzor, proto není lineární konexe tenzorem.

Další objekty jsou svázány jak s metrikou, tak s lineární konexí. Jsou to tenzor torze lineární konexe $\mathbf{T}$ a Riemannův tenzor křivosti $\mathbf{R}$, které popisují geometrické vlastnosti konkrétní lineární konexe. Pokud platí $\mathbf{T}=0$, jedná se o lineární konexi bez torze. Jestliže je $\mathbf{R}=0$, pak se jedná o plochou lineární konexi.

Z Riemannova tenzoru křivosti je pak odvozen Ricciho tenzor, který vznikne úžením. Úžením Ricciho tenzoru pak dostaneme Ricciho skalární křivost R.

Dále se definuje Einsteinův tenzor, který tvoří Einsteinovy rovnice gravitačního pole a Weylův tenzor, který představuje část Riemannova tenzoru s nulovou stopou, používaný v Petrovově klasifikaci metrik.

Christoffelovy symboly

Definují se vztahem

$$\Gamma^i_{jk} = \frac{1}{2} g^{il} \quad \frac{\partial g_{lj}}{\partial x^k} + \frac{\partial g_{lk}}{\partial x^j} - \frac{\partial g_{jk}}{x^l}$$

a jsou tzv. doplnitelnou strukturou, která umožňuje zavést pojem *kovariantní derivace*, což je derivace vztahující se na zakřivený prostoročas.

Lieova derivace

Tímto názvem se označuje Lieova závorka $[\mathbf{X}, \mathbf{Y}]$, na kterou se pohlíží jako na operaci diferencování. Výraz

$$\mathcal{L}_\mathbf{X} \mathbf{Y} = [\mathbf{X}, \mathbf{Y}] = -[\mathbf{Y}, \mathbf{X}] = -\mathcal{L}_\mathbf{Y} \mathbf{X}$$

je *Lieova derivace* vektoru $\mathbf{Y}$ ve směru vektoru $\mathbf{X}$. V obecném případě je Lieova derivace $\mathcal{L}_\mathbf{X} \mathbf{Y}$ tenzorového pole $\mathbf{T}$ typu (r, s) opět tenzor téhož typu.

Lieova derivace splňuje následující vlastnosti:

1. Lieova derivace skalárního pole f

$$\mathcal{L}_\mathbf{X} f = \mathbf{X} f = \mathrm{d} f(\mathbf{X}),$$

2. Lieova derivace vektorového pole $\mathbf{Y}$

$$\mathcal{L}_\mathbf{X} \mathbf{Y} = [\mathbf{X}, \mathbf{Y}],$$

3. Lieova derivace tenzorového pole je lineární operátor, jehož působení na tenzorový součin se řídí Leibnitzovým pravidlem

$$\mathcal{L}_\mathbf{X} (\mathbf{S} \otimes \mathbf{T}) = \mathcal{L}_\mathbf{X} \mathbf{S} \otimes \mathbf{T} + \mathbf{S} \otimes \mathcal{L}_\mathbf{X} \mathbf{T},$$

   kde $\mathbf{S}$, $\mathbf{T}$ jsou libovolná tenzorová pole.

Kovariantní derivace

Jelikož obyčejná parciální derivace tenzorů nemá tenzorový charakter (netransformuje se jako tenzor), je potřeba nalézt takovou kalibraci, která tuto potíž beze zbytku odstraní. Derivace, jejímž působením na tenzor dostaneme objekt tenzorového charakteru, se nazývá *kovariantní derivace*. Je definována vztahem

$$T^{ij\dots}_{kl\dots;m} = \frac{\partial T^{ij\dots}_{kl\dots}}{\partial x^m} + \Gamma^i_{nm} T^{nj\dots}_{kl\dots} + \Gamma^j_{nm} T^{in\dots}_{kl\dots} + \dots$$
$$- \Gamma^n_{km} T^{ij\dots}_{nl\dots} - \Gamma^n_{lm} T^{ij\dots}_{kn\dots} - \dots .$$

Vlastnosti Riemannova a Ricciho tenzoru

Riemannův tenzor křivosti je definován následujícím způsobem:

$$R^j_{lnm} = \Gamma^j_{lm,n} - \Gamma^j_{ln,m} + \Gamma^j_{kn}\Gamma^k_{lm} - \Gamma^j_{km}\Gamma^k_{ln},$$

přičemž je zde zavedeno označení $\Gamma^j_{lm,n} = \partial\Gamma^j_{lm}/\partial x^n$, které je jen jiným způsobem zápisu parciální derivace.

Některé důležité vlastnosti:

$$R_{jkmn} + R_{jmnk} + R_{jnkm} = 0,$$
$$R_{jkmn} = R_{mnjk},$$
$$R_{jkmn} = -R_{kjmn} = -R_{jknm} = R_{kjnm}.$$

Ricciho tenzor získáme úžením z Riemannova tenzoru:

$$R_{ij} = g^{kl} R_{ikjl} = g^{lk} R_{ljki} = R_{ji}.$$

Je vidět, že Ricciho tenzor je symetrický. Ricciho skalár dostaneme úžením Ricciho tenzoru:

$$R = g^{ij} R_{ij}.$$

Weylův tenzor

Je definován jako část Riemannova tenzoru křivosti, která má nulovou stopu. Má tedy stejné vlastnosti symetrie:

$$C_{ijkl} = R_{ijkl} - \frac{1}{2}(g_{ik}R_{jl} + g_{jl}R_{ik} - g_{jk}R_{il} - g_{il}R_{jk}) +$$
$$+ \frac{1}{6}(g_{ik}g_{jl} - g_{jk}g_{il})R.$$

Riemannův tenzor má dvacet nezávislých komponent, tenzory Ricciho a Weylův mají každý po deseti nezávislých komponentách.

Einsteinovy rovnice

Einsteinovy gravitační rovnice mají následující tvar:

$$G_{ij} - \Lambda g_{ij} = R_{ij} - \frac{1}{2} g_{ij} R - \Lambda g_{ij} = -\frac{8\pi G}{c^4} T_{ij},$$

kde T_{ij} je tenzor energie-hybnosti elektromagnetického pole a Λ je kosmologická konstanta. V dalším textu uvažujeme případ, kdy $\Lambda = 0$.

2.3. Formalismus tetrád

Standardní způsob řešení úloh v OTR spočívá v zavedení lokálního souřadnicového systému, přičemž se souřadnice vybírají tak, aby řešení bylo co nejjednodušší. Existuje ale také druhý způsob řešení, který je založen na tom, že se vybere *tetrádní báze* ze čtyř nezávislých vektorových polí a všechny potřebné veličiny se transformují do této vybrané báze. Dále se sledují rovnice, kterým tyto veličiny vyhovují. Tento formalismus se nazývá tetrádní.

Zavedení tetrád

V každém bodě prostoročasu zavedeme bázi ze čtyř kontravariantních vektorů

$$e^i_{(a)} \quad (a = 1, 2, 3, 4), \tag{1}$$

kde index v závorkách čísluje vektory tetrády a nazývá se tetrádním indexem na rozdíl od tenzorových indexů, které se píší bez závorek. S kontravariantními vektory (1) jsou spjaty kovariantní vztahem

$$e_{(a)i} = g_{ik} e^k_{(a)},$$

kde veličina g_{ik} označuje metrický tenzor. Dále platí vztahy

$$e^i_{(a)} e^{(b)}_i = \delta^{(b)}_{(a)},$$
$$e^i_{(a)} e^{(a)}_j = \delta^i_j,$$
$$e^i_{(a)} e_{(b)i} = \eta_{(a)(b)},$$

kde $\eta_{(a)(b)}$ je konstantní symetrická matice. Předpokládá se, že vektory $e^i_{(a)}$ představují ortonormovanou tetrádu; v tom případě je $\eta_{(a)(b)}$ diagonální matice s elementy $+1$, -1, -1, -1. Dále platí, že

$$\begin{aligned} \eta^{(a)(b)}\eta_{(b)(c)} &= \delta^{(a)}_{(c)}, \\ \eta^{(a)(b)}e_{(a)i} = e^{(b)}_i, \quad \eta_{(a)(b)}e^{(a)}_i &= e_{(b)i}, \\ e_{(a)i}e^{(a)}_j &= g_{ij}. \end{aligned}$$

Jestliže provedeme transformaci vektorového či tenzorového pole na tetrádu, dostaneme *tetrádní komponenty* vektoru či tenzoru:

$$\begin{aligned} A_{(a)} &= e_{(a)j}A^j = e^j_{(a)}A_j, \\ A^{(a)} &= \eta^{(a)(b)}A_{(b)} = e^{(a)}_j A^j = e^{(a)j}A_j, \\ A^i &= e^i_{(a)}A^{(a)} = e^{(a)i}A_{(a)}, \\ T_{(a)(b)} &= e^i_{(a)}e^j_{(b)}T_{ij} = e^i_{(a)}T_{i(b)}, \\ T_{ij} &= e^{(a)}_i e^{(b)}_j T_{(a)(b)} = e^{(a)}_i T_{(a)j}. \end{aligned}$$

Derivace ve směru a Ricciho koeficienty

Kontravariantní vektory $e_{(a)}$ jako tečné vektory umožňují popsat směrové derivace

$$e_{(a)} = e^i_{(a)}\frac{\partial}{\partial x^i};$$

zavedeme označení $\phi_{,(a)} = e^i_{(a)}(\partial\phi/\partial x^i) = e^i_{(a)}\phi_{,i}$, kde ϕ je libovolné skalární pole. V obecnějším případě máme

$$\begin{aligned} A_{(a),(b)} &= e^i_{(b)}\frac{\partial}{\partial x^i}A_{(a)} = e^i_{(b)}\frac{\partial}{\partial x^i}e^j_{(a)}A_j = \\ &= e^i_{(b)}\nabla_{\partial_i}\left[e^j_{(a)}A_j\right] = e^i_{(b)}\left[e^j_{(a)}A_{j;i} + A_k e^k_{(a);i}\right]. \end{aligned}$$

Odtud pak dostáváme:

$$A_{(a),(b)} = e^j_{(a)}A_{j;i}e^i_{(b)} + e_{(a)k;i}e^i_{(b)}e^k_{(c)}A^{(c)}. \tag{2}$$

Zavedeme-li označení $\gamma_{(c)(a)(b)} = e^k_{(c)} e_{(a)k;i} e^i_{(b)}$, můžeme přepsat rovnici (2) do tvaru

$$A_{(a),(b)} = e^j_{(a)} A_{j;i} e^i_{(b)} + \gamma_{(c)(a)(b)} A^{(c)}. \tag{3}$$

Veličiny $\gamma_{(a)(b)(c)}$ se nazývají *Ricciho koeficienty*. Tyto koeficienty jsou antisymetrické v prvních dvou indexech:

$$\gamma_{(c)(a)(b)} = -\gamma_{(a)(c)(b)}.$$

Rovnice (3) lze pak přepsat do tvaru

$$e^i_{(a)} A_{i;j} e^j_{(b)} = A_{(a),(b)} - \eta^{(n)(m)} \gamma_{(n)(a)(b)} A_{(m)}.$$

K nalezení Ricciho koeficientů nejsou potřebné kovariantní derivace, takže není potřeba znát Christoffelovy symboly. Zavedeme-li

$$\begin{aligned} \lambda_{(a)(b)(c)} &= e_{(b)i,j} \left[e^i_{(a)} e^j_{(c)} - e^j_{(a)} e^i_{(c)} \right] = \\ &= \; e_{(b)i,j} - e_{(b)j,i} \;\; e^i_{(a)} e^j_{(c)} \end{aligned} \tag{4}$$

a změníme-li v rovnici (4) obyčejné derivace kovariantními (v důsledku symetrie afinních koeficientů), dostaneme

$$\lambda_{(a)(b)(c)} = [e_{(b)i;j} - e_{(b)j;i}] e^i_{(a)} e^j_{(c)} = \gamma_{(a)(b)(c)} - \gamma_{(c)(b)(a)}.$$

Rozřešíme-li tyto rovnice vzhledem ke γ, obdržíme vztah

$$\gamma_{(a)(b)(c)} = \frac{1}{2} \;\; \lambda_{(a)(b)(c)} + \lambda_{(c)(a)(b)} - \lambda_{(b)(c)(a)} \;\; ,$$

odkud je zřejmé, že k určení Ricciho koeficientů je nutné provést pouze obyčejné derivace. Pro koeficienty λ dále platí tato symetrie:

$$\lambda_{(a)(b)(c)} = -\lambda_{(c)(b)(a)}.$$

Komutační relace a strukturní koeficienty

Strukturní koeficienty $C^{(c)}_{(a)(b)}$ jsou definovány pomocí Lieovy závorky bázových vektorů, kterým přiřazuje tečný vektor podle vztahu

$$[e_{(a)}, e_{(b)}] = C^{(c)}_{(a)(b)} e_{(c)}. \tag{5}$$

V obecném případě máme 24 různých konstant a tyto konstanty jsou antisymetrické v indexech (a) a (b). Strukturní koeficienty souvisí s Ricciho koeficienty vztahem

$$C^{(c)}_{(a)(b)} = \gamma^{(c)}_{(b)(a)} - \gamma^{(c)}_{(a)(b)}.$$

Jestliže do rovnic (5) dosadíme místo strukturních konstant Ricciho koeficienty, dostaneme 24 *komutačních vztahů*.

Ricciho identity

Provedeme-li projekci Ricciho identity $e_{(a)i;kl} - e_{(a)i;lk} = R_{mikl} e^m_{(a)}$ do tetrádní báze, obdržíme vztah pro Riemannův tenzor v tetrádních složkách:

$$\begin{aligned} R_{(a)(b)(c)(d)} = & -\gamma_{(a)(b)(c),(d)} + \gamma_{(a)(b)(d),(c)} + \\ & + \gamma_{(b)(a)(f)} \left[\gamma_{(c)}{}^{(f)}{}_{(d)} - \gamma_{(d)}{}^{(f)}{}_{(c)} \right] + \\ & + \gamma_{(f)(a)(c)} \gamma_{(b)}{}^{(f)}{}_{(d)} - \gamma_{(f)(a)(d)} \gamma_{(b)}{}^{(f)}{}_{(c)}. \end{aligned} \tag{6}$$

V důsledku antisymetrie Ricciho koeficientů přes hlavní dva indexy existuje 36 rovnic (6).

2.4. Formalismus Newmana-Penrose

Jedná se o tetrádní formalismus ve speciálně vybrané bázi. Zavedli jej Newman a Penrose [8] v roce 1962 a jeho podstata spočívá v tom, že za základ vzali izotropní bázi namísto ortonormované báze vektorů. NP-formalismus umožňuje řešit problémy týkající se černých děr s větší efektivitou a vyhovuje také pro řešení problému Petrovovy klasifikace pomocí počítače.

Izotropní báze a spinové koeficienty

V základech formalismu Newmana-Penrose je izotropní báze složená z páru reálných vektorů $\mathbf{l}$, $\mathbf{n}$ a z páru komplexně sdružených vektorů $\mathbf{m}$, $\overline{\mathbf{m}}$. Tyto vektory musí splňovat tři podmínky:

1. ortogonality: $\mathrm{l}^a \cdot \mathrm{m}_a = \mathrm{l}^a \cdot \overline{\mathrm{m}}_a = \mathrm{n}^a \cdot \mathrm{m}_a = \mathrm{n}^a \cdot \overline{\mathrm{m}}_a = 0,$
2. izotropnosti: $\mathrm{l}^a \cdot \mathrm{l}_a = \mathrm{n}^a \cdot \mathrm{n}_a = \mathrm{m}^a \cdot \mathrm{m}_a = \overline{\mathrm{m}}^a \cdot \overline{\mathrm{m}}_a = 0,$

3. normování: $\mathrm{l}^a \cdot \mathrm{n}_a = 1, \quad \mathrm{m}^a \cdot \overline{\mathrm{m}}_a = -1.$

Z toho důvodu je základní matice $\eta_{(a)(b)}$ konstantní symetrickou maticí tvaru:

$$\eta_{(a)(b)} = \eta^{(a)(b)} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \end{pmatrix},$$

jestliže se zavede následující označení tetrády:

$$\mathbf{e_1} = \mathbf{l}, \ \mathbf{e_2} = \mathbf{n}, \ \mathbf{e_3} = \mathbf{m}, \ \mathbf{e_4} = \overline{\mathbf{m}}.$$

Odpovídající kovariantní báze má tvar

$$\mathbf{e^1} = \mathbf{e_2} = \mathbf{n}, \ \mathbf{e^2} = \mathbf{e_1} = \mathbf{l}, \ \mathbf{e^3} = -\mathbf{e_4} = -\overline{\mathbf{m}}, \ \mathbf{e^4} = -\mathbf{e_3} = -\mathbf{m}.$$

Bázové vektory, které jsou brány jako směrové derivace mají ve formalismu Newmana-Penrose speciální označení:

$$\begin{aligned} \mathbf{e_1} &= \mathbf{e^2} = D, & \mathbf{e_2} &= \mathbf{e^1} = \Delta, \\ \mathbf{e_3} &= -\mathbf{e^4} = \delta, & \mathbf{e_4} &= -\mathbf{e^3} = \delta^*. \end{aligned}$$

Různé Ricciho koeficienty, které se teď nazývají *spinové koeficienty*, mají tato označení (tady a v dalším textu již není prováděno uzávorkování tetrádních koeficientů):

$$\begin{aligned} \kappa &= \gamma_{311}, & \varrho &= \gamma_{314}, & \varepsilon &= (\gamma_{211} + \gamma_{311})/2, \\ \sigma &= \gamma_{313}, & \mu &= \gamma_{243}, & \gamma &= (\gamma_{212} + \gamma_{342})/2, \\ \lambda &= \gamma_{244}, & \tau &= \gamma_{312}, & \alpha &= (\gamma_{214} + \gamma_{344})/2, \\ \nu &= \gamma_{242}, & \pi &= \gamma_{241}, & \beta &= (\gamma_{213} + \gamma_{343})/2. \end{aligned}$$

Komplexně sdružené veličiny se dostanou záměnou indexů 3 za indexy 4 a obráceně.

Zápis Weylova, Riemannova a Ricciho tenzoru v NP-formalismu

Weylův tenzor je definován jako bezestopá část Riemannova tenzoru:

$$C_{abcd}=R_{abcd}-\frac{1}{2}(\eta_{ac}R_{bd}-\eta_{bc}R_{ad}-\eta_{ad}R_{bc}+\eta_{bd}R_{ac})+ \\ +\frac{1}{6}(\eta_{ac}\eta_{bd}-\eta_{ad}\eta_{bc})R,$$

kde R_{ab} jsou tetrádní komponenty Ricciho tenzoru a R je skalární křivost:

$$R_{ac}=\eta^{bd}R_{abcd}, \qquad R=\eta^{ab}R_{ab}=2(R_{12}-R_{34}).$$

Dále platí vztahy:

$$\eta^{ad}C_{abcd}=C_{1bc2}+C_{2bc1}-C_{3bc4}-C_{4bc3}=0, \\ C_{1234}+C_{1342}+C_{1423}=0.$$

V NP-formalismu se deset nezávislých komponent Weylova tenzoru popisuje pěti komplexními skaláry:

$$\begin{aligned} \Psi_0&=-C_{1313}=-C_{pqrs}l^p m^q l^r m^s, \\ \Psi_1&=-C_{1213}=-C_{pqrs}l^p n^q l^r m^s, \\ \Psi_2&=-C_{1342}=-C_{pqrs}l^p m^q \overline{m}^r n^s, \\ \Psi_3&=-C_{1242}=-C_{pqrs}l^p n^q \overline{m}^r n^s, \\ \Psi_4&=-C_{2424}=-C_{pqrs}n^p \overline{m}^q n^r \overline{m}^s. \end{aligned} \tag{7}$$

Nyní se věnujme Ricciho tenzoru. Deset nezávislých komponent Ricciho tenzoru se v NP-formalismu zapisuje pomocí těchto skalárů:

$$\begin{aligned} \Phi_{00}&=-\frac{1}{2}R_{11}, & \Phi_{22}&=-\frac{1}{2}R_{22}, & \Phi_{02}&=-\frac{1}{2}R_{33}, \\ \Phi_{20}&=-\frac{1}{2}R_{44}, & \Phi_{11}&=-\frac{1}{4}(R_{12}+R_{34}), & \Phi_{01}&=-\frac{1}{2}R_{13}, \\ \Phi_{10}&=-\frac{1}{2}R_{14}, & \Phi_{12}&=-\frac{1}{2}R_{23}, & \Phi_{21}&=-\frac{1}{2}R_{24}, \end{aligned}$$

$$\Lambda=\frac{1}{24}R=\frac{1}{12}(R_{12}-R_{34}).$$

Komutační vztahy a strukturní konstanty

Vyjdeme z komutačních vztahů pro bázové vektory

$$[\mathbf{e}_a, \mathbf{e}_b] = (\gamma_{cba} - \gamma_{cab})\mathbf{e}^c = C^c_{ab}\mathbf{e}_c. \tag{8}$$

Pro jeden případ provedu podrobný popis:

$$\begin{aligned}[\Delta, D] = [\mathbf{n}, \mathbf{l}] = [\mathbf{e_2}, \mathbf{e_1}] = (\gamma_{c12} - \gamma_{c21})\, \mathbf{e}^c = \\ = -\gamma_{121}\mathbf{e^1} + \gamma_{212}\mathbf{e^2} + (\gamma_{312} - \gamma_{321})\, \mathbf{e^3} + (\gamma_{412} - \gamma_{421})\, \mathbf{e^4} = \\ = -\gamma_{121}\Delta + \gamma_{212}D - (\gamma_{312} - \gamma_{321})\delta^* - (\gamma_{412} - \gamma_{421})\delta.\end{aligned}$$

Tato komutační relace pak pomocí spinových koeficientů dostane následující tvar:

$$\begin{aligned}[\Delta, D] \equiv \Delta D - D\Delta = \\ = (\gamma + \gamma^*)D + (\varepsilon + \varepsilon^*)\Delta - (\tau^* + \pi)\delta - (\tau + \pi^*)\delta^*.\end{aligned}$$

Podobným způsobem se odvodí tyto vztahy:

$$\begin{aligned}\delta D - D\delta = (\alpha^* + \beta - \pi^*)D + \kappa\Delta - (\varrho^* + \varepsilon - \varepsilon^*)\delta - \sigma\delta^*, \\ \delta\Delta - \Delta\delta = -\nu^* D + (\tau - \alpha^* - \beta)\Delta + (\mu - \gamma + \gamma^*)\delta + \lambda^*\delta^*, \\ \delta^*\delta - \delta\delta^* = (\mu^* - \mu)D + (\varrho^* - \varrho)\Delta + (\alpha - \beta^*)\delta + (\beta - \alpha^*)\delta^*.\end{aligned}$$

Srovnají-li se tyto vztahy se vzorcem (8), dostane se následující vyjádření strukturních konstant pomocí spinových koeficientů:

$$\begin{aligned}
C^1_{21} &= \gamma + \gamma^*, & C^1_{31} &= \alpha^* + \beta - \pi^*, \\
C^1_{32} &= -\nu^*, & C^1_{43} &= \mu^* - \mu, \\
C^2_{21} &= \varepsilon + \varepsilon^*, & C^2_{31} &= \kappa, \\
C^2_{32} &= \tau - \alpha^* - \beta, & C^2_{43} &= \varrho^* - \varrho, \\
C^3_{21} &= -(\tau^* + \pi), & C^3_{31} &= -(\varrho^* + \varepsilon - \varepsilon^*), \\
C^3_{32} &= \mu - \gamma + \gamma^*, & C^3_{43} &= \alpha - \beta^*, \\
C^4_{21} &= -(\tau + \pi^*), & C^4_{31} &= -\sigma, \\
C^4_{32} &= \lambda^*, & C^4_{43} &= \beta - \alpha^*.
\end{aligned}$$

Ricciho identity a jim odpovídající rovnice

Ricciho identity jsou v NP-formalismu odvozeny následujícím způsobem:

$$\begin{aligned}-\Psi_0=C_{1313}=R_{1313}=\gamma_{133,1}-\gamma_{131,3}+&\\ &+\gamma_{133}(\gamma_{121}+\gamma_{431}-\gamma_{413}+\gamma_{431}+\gamma_{134})-\\ &-\gamma_{131}(\gamma_{433}+\gamma_{123}-\gamma_{213}+\gamma_{231}+\gamma_{132}).\end{aligned} \tag{9}$$

Použijí-li se již odvozené vztahy pro derivace ve směru a spinové koeficienty, lze rovnici (9) přepsat do tvaru

$$D\sigma-\delta\kappa=\sigma(3\varepsilon-\varepsilon^*+\varrho+\varrho^*)+\kappa(\pi^*-\tau-3\beta-\alpha^*)+\Psi_0.$$

V NP-formalismu stačí zapsat systém osmnácti rovnic (nemusí se vypisovat komplexně sdružené). Nyní tento systém uvedu, pričemž u každé rovnice bude vypsána komponenta Riemannova tenzoru, která s ní souvisí.

$$\begin{aligned}
D\varrho-\delta^*\kappa&=(\varrho^2+\sigma\sigma^*)+\varrho(\varepsilon+\varepsilon^*)-\kappa^*\tau-\\
&\quad-\kappa(3\alpha+\beta^*-\pi)+\Phi_{00}, && [R_{1314}]\\
D\sigma-\delta\kappa&=\sigma(\varrho+\varrho^*+3\varepsilon-\varepsilon^*)-\\
&\quad-\kappa(\tau-\pi^*+\alpha^*+3\beta)+\Psi_0, && [R_{1313}]\\
D\tau-\Delta\kappa&=\varrho(\tau+\pi^*)+\sigma(\tau^*+\pi)+\tau(\varepsilon-\varepsilon^*)-\\
&\quad-\kappa(3\gamma+\gamma^*)+\Psi_1+\Phi_{01}, && [R_{1312}]\\
D\alpha-\delta^*\varepsilon&=\alpha(\varrho+\varepsilon^*-2\varepsilon)+\beta\sigma^*-\beta^*\varepsilon-\kappa\lambda-\\
&\quad-\kappa^*\gamma+\pi(\varepsilon+\varrho)+\Phi_{10}, && [\tfrac{1}{2}(R_{3414}-R_{1214})]\\
D\beta-\delta\varepsilon&=\sigma(\alpha+\pi)+\beta(\varrho^*-\varepsilon^*)-\kappa(\mu+\gamma)-\\
&\quad-\varepsilon(\alpha^*-\pi^*)+\Psi_1, && [\tfrac{1}{2}(R_{1213}-R_{3413})]\\
D\gamma-\Delta\varepsilon&=\alpha(\tau+\pi^*)+\beta(\tau^*+\pi)-\gamma(\varepsilon+\varepsilon^*)-\varepsilon(\gamma+\gamma^*)\\
&\quad+\tau\pi-\nu\kappa+\Psi_2+\Phi_{11}-\Lambda, && [\tfrac{1}{2}(R_{1212}-R_{3412})]\\
D\lambda-\delta^*\pi&=(\varrho\lambda+\sigma^*\mu)+\pi(\pi+\alpha-\beta^*)-\nu\kappa^*-\\
&\quad-\lambda(3\varepsilon+\varepsilon^*)+\Phi_{20}, && [R_{2441}]\\
D\mu-\delta\pi&=(\varrho^*\mu+\sigma\lambda)+\pi(\pi^*-\alpha^*+\beta)-\\
&\quad-\mu(\varepsilon+\varepsilon^*)-\nu\kappa+\Psi_2+2\Lambda, && [R_{2431}]
\end{aligned}$$

$$
\begin{aligned}
D\nu-\Delta\pi &= \mu(\pi+\tau^*)+\lambda(\pi^*+\tau)+\pi(\gamma-\gamma^*)- \\
&\quad -\nu(3\varepsilon+\varepsilon^*)+\Psi_3+\Phi_{21}, && [R_{2421}] \\
\Delta\lambda-\delta^*\nu &= -\lambda(\mu+\mu^*+3\gamma-\gamma^*)+ \\
&\quad +\nu(3\alpha+\beta^*+\pi-\tau^*)-\Psi_4, && [R_{2442}] \\
\delta\varrho-\delta^*\sigma &= \varrho(\alpha^*+\beta)-\sigma(3\alpha-\beta^*)+\tau(\varrho-\varrho^*)+ \\
&\quad +\kappa(\mu-\mu^*)-\Psi_1+\Phi_{01}, && [R_{3143}] \\
\delta\alpha-\delta^*\beta &= (\mu\varrho-\lambda\sigma)+\alpha\alpha^*+\beta\beta^*-2\alpha\beta+\gamma(\varrho-\varrho^*)+ \\
&\quad +\varepsilon(\mu-\mu^*)-\Psi_2+\Phi_{11}+\Lambda, && [\tfrac{1}{2}(R_{1234}-R_{3434})] \\
\delta\lambda-\delta^*\mu &= \nu(\varrho-\varrho^*)+\pi(\mu-\mu^*)+\mu(\alpha+\beta^*)+ \\
&\quad +\lambda(\alpha^*-3\beta)-\Psi_3+\Phi_{21}, && [R_{2443}] \\
\delta\nu-\Delta\mu &= (\mu^2+\lambda\lambda^*)+\mu(\gamma+\gamma^*)-\nu^*\pi+ \\
&\quad +\nu(\tau-3\beta-\alpha^*)+\Phi_{22}, && [R_{2423}] \\
\delta\gamma-\Delta\beta &= \gamma(\tau-\alpha^*-\beta)+\mu\tau-\sigma\nu-\varepsilon\nu^*- \\
&\quad -\beta(\gamma-\gamma^*-\mu)+\alpha\lambda^*+\Phi_{12}, && [\tfrac{1}{2}(R_{1232}-R_{3432})] \\
\delta\tau-\Delta\sigma &= (\mu\sigma+\lambda^*\sigma)+\tau(\tau+\beta-\alpha^*)- \\
&\quad -\sigma(3\gamma-\gamma^*)-\kappa\nu^*+\Phi_{02}, && [R_{1332}] \\
\Delta\varrho-\delta^*\tau &= -(\varrho\mu^*+\sigma\lambda)+\tau(\beta^*-\alpha-\tau^*)+ \\
&\quad +\varrho(\gamma+\gamma^*)+\nu\kappa-\Psi_2-2\Lambda, && [R_{1324}] \\
\Delta\alpha-\delta^*\gamma &= \nu(\varrho+\varepsilon)-\lambda(\tau+\beta)+\alpha(\gamma^*-\mu^*)+ \\
&\quad +\gamma(\beta^*-\tau^*)-\Psi_3, && [\tfrac{1}{2}(R_{1242}-R_{3442})]
\end{aligned}
$$

2.5. Petrovova klasifikace

Již bylo uvedeno, že v dané tetrádě je Weylův tenzor zcela popsán pomocí pěti komplexních skalárů $\Psi_0, \ldots, \Psi_4$. Tyto skaláry obecně závisí na tetrádě, kterou ovšem lze transformovat. Vzniká tak otázka, které z těchto skalárů lze vynulovat, vybereme-li výhodnější orientaci tetrády. Odpověď na tuto otázku nás přivede k algebraické klasifikaci Weylova tenzoru, kterou zavedl A. Z. Petrov [9].

Možné transformace tetrád:

1. Rotace třídy I, při níž se zachovává vektor $\mathbf{l}$,
2. Rotace třídy II, při níž se zachovává vektor $\mathbf{n}$,
3. Rotace třídy III, při které se nemění směry $\mathbf{l}$ a $\mathbf{n}$; provádí rotaci vektoru $\mathbf{m}$ (a $\overline{\mathbf{m}}$) o úhel θ v rovině $(\mathbf{m}, \overline{\mathbf{m}})$.

Transformace, které odpovídají těmto požadavkům a splňují podmínky ortonormality, mají tento tvar:

Transf.	$\mathbf{l}$	$\mathbf{n}$	$\mathbf{m}$	$\overline{\mathbf{m}}$
I.	$\mathbf{l}$	$\mathbf{n}+a^*\mathbf{m}+a\overline{\mathbf{m}}+aa^*\mathbf{l}$	$\mathbf{m}+a\mathbf{l}$	$\overline{\mathbf{m}}+a^*\mathbf{l}$
II.	$\mathbf{l}+b^*\mathbf{m}+b\overline{\mathbf{m}}+bb^*\mathbf{n}$	$\mathbf{n}$	$\mathbf{m}+b\mathbf{n}$	$\overline{\mathbf{m}}+b^*\mathbf{n}$
III.	$A^{-1}\mathbf{l}$	$A\mathbf{n}$	$\mathrm{e}^{\mathrm{i}\theta}\mathbf{m}$	$\mathrm{e}^{-\mathrm{i}\theta}\overline{\mathbf{m}}$

kde a, a^*, b, b^* jsou komplexně sdružené funkce a A, θ jsou funkce reálné.

Jednotlivé transformace působící na Weylovské skaláry Ψ_i:

Rotace I. třídy:

$$\Psi_0 \to \Psi_0,$$
$$\Psi_1 \to \Psi_1 + a^*\Psi_0,$$
$$\Psi_2 \to \Psi_2 + 2a^*\Psi_1 + (a^*)^2\Psi_0,$$
$$\Psi_3 \to \Psi_3 + 3a^*\Psi_2 + 3(a^*)^2\Psi_1 + (a^*)^3\Psi_0,$$
$$\Psi_4 \to \Psi_4 + 4a^*\Psi_3 + 6(a^*)^2\Psi_2 + 4(a^*)^3\Psi_1 + (a^*)^4\Psi_0.$$

Rotace II. třídy:

$$\Psi_0 \to \Psi_0 + 4b\Psi_1 + 6b^2\Psi_2 + 4b^3\Psi_3 + b^4\Psi_4,$$
$$\Psi_1 \to \Psi_1 + 3\Psi_2 + 3b^2\Psi_3 + b^3\Psi_4,$$
$$\Psi_2 \to \Psi_2 + 2b\Psi_3 + b^2\Psi_4,$$
$$\Psi_3 \to \Psi_3 + b\Psi_4,$$
$$\Psi_4 \to \Psi_4.$$

Rotace III. třídy:

$$\begin{aligned}
&\Psi_0 \to A^{-2}\,\mathrm{e}^{2\mathrm{i}\theta}\Psi_0,\\
&\Psi_1 \to A^{-1}\,\mathrm{e}^{\mathrm{i}\theta}\Psi_1,\\
&\Psi_2 \to \Psi_2,\\
&\Psi_3 \to A\,\mathrm{e}^{-\mathrm{i}\theta}\Psi_3,\\
&\Psi_4 \to A^2\,\mathrm{e}^{-2\mathrm{i}\theta}\Psi_4.
\end{aligned}$$

Při změně tetrád dochází samozřejmě také k transformaci spinových koeficientů.

Definice Petrovovy klasifikace

Nechť je $\Psi_4 \neq 0$. (Jestliže je $\Psi_4 = 0$, pak jej lze rotací I převést na $\Psi_4 \neq 0$, pokud není prostor konformně plochý, kdy jsou všechny Weylovské skaláry rovny nule.) Nyní provedeme rotaci třídy II s parametrem b. Weylovské skaláry se tak transformují do tvaru:

$$\begin{aligned}
&\Psi_0^{(1)} \to \Psi_0 + 4b\Psi_1 + 6b^2\Psi_2 + 4b^3\Psi_3 + b^4\Psi_4,\\
&\Psi_1^{(1)} \to \Psi_1 + 3\Psi_2 + 3b^2\Psi_3 + b^3\Psi_4,\\
&\Psi_2^{(1)} \to \Psi_2 + 2b\Psi_3 + b^2\Psi_4,\\
&\Psi_3^{(1)} \to \Psi_3 + b\Psi_4,\\
&\Psi_4^{(1)} \to \Psi_4,
\end{aligned}$$

kde horní index označuje skalár po transformaci. Je zřejmé, že rotací třídy II lze vynulovat skalár Ψ_0, jestliže je b kořenem rovnice

$$\Psi_4 b^4 + 4\Psi_3 b^3 + 6\Psi_2 b^2 + 4\Psi_1 b + \Psi_0 = 0. \tag{10}$$

Tato rovnice má vždy čtyři kořeny a jim odpovídající nové směry vektoru $\mathbf{l}$ dané vztahem $\mathbf{l} + b^*\mathbf{m} + b\overline{\mathbf{m}} + bb^*\mathbf{n}$, se nazývají *hlavní izotropní přímky* Weylova tenzoru. Jestliže jsou stejné dva nebo víc kořenů, pak je tenzor *algebraicky speciální*, jinak je tenzor *algebraicky obecný*. Různé varianty rovnosti kořenů jsou pak základem Petrovovy klasifikace.

Typ I. V tomto případě jsou všechny čtyři kořeny v rovnici (10) různé. Označme je b_1, b_2, b_3, b_4. Potom rotací třídy II s parametrem $b = b_1$

(dosazením kteréhokoliv kořene dostaneme tentýž výsledek) transformujeme Ψ_0 na nulu. Pak lze rotací třídy I (která nemá vliv na Ψ_0) převést na nulu* i Ψ_4. Skaláry Ψ_1, Ψ_2, Ψ_3 zůstanou různé od nuly a invariantní vůči rotaci třídy III, která nemá vliv na Ψ_0 a Ψ_4.

Typ II. Rovnice (10) má dva stejné kořeny $b_1 = b_2$ ($\neq b_3 \neq b_4$, $b_3 \neq b_4$). V tomto případě má nejen rovnice (10), ale i její derivace podle b

$$\Psi_4 b^3 + 3\Psi_3 b^2 + 3\Psi_2 b + \Psi_1 = 0$$

kořen $b = b_2$. Nyní rotací II s parametrem $b = b_1$ ($= b_2$) můžeme vynulovat Ψ_0 a Ψ_1. Potom rotací I (nemá vliv na Ψ_0, Ψ_1) může být vynulován skalár Ψ_4. Nenulové zůstanou pouze Ψ_2 a Ψ_3.

Typ D. Rovnice (10) má dva různé dvojnásobné kořeny b_1 a b_2. Rotací třídy II a potom rotací třídy I mohou být vynulovány skaláry Ψ_0, Ψ_1, Ψ_3 a Ψ_4. Různým od nuly zůstane pouze skalár Ψ_2 (invariantní vůči rotaci III).

Typ III. Jsou stejné tři kořeny rovnice (10): $b_1 = b_2 = b_3 \neq b_4$. Rotací třídy II s parametrem $b = b_1$ ($= b_2 = b_3$) mohou být vynulovány skaláry Ψ_0, Ψ_1 a Ψ_2. Rotací I lze dále vynulovat skalár Ψ_4, takže jediným nenulovým zůstane Ψ_3 (může změnit znaménko při rotaci III).

Typ N. Řešením rovnice (10) je jeden čtyřnásobný kořen b. Rotací třídy II lze vynulovat skaláry Ψ_0, Ψ_1, Ψ_2 a Ψ_3. Nenulovým Weylovským skalárem zůstane pouze Ψ_4.

Typ 0. Jedná se o konformně plochý prostor, v němž jsou všechny Weylovské skaláry rovny nule.

Průběh degenerace jednotlivých typů Petrova je dobře znázorněn na Penroseově diagramu (obr. 1).

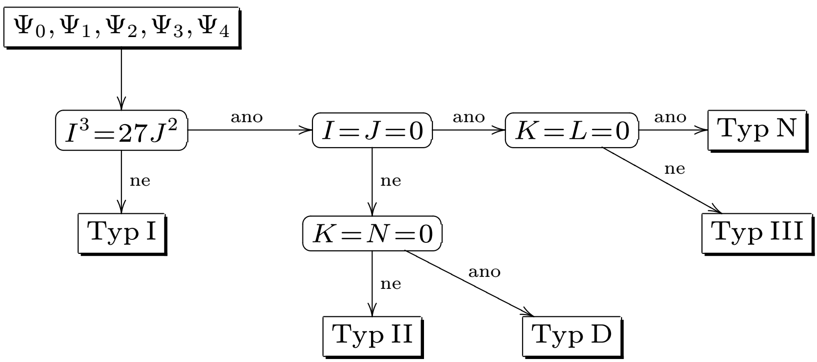
* Kořeny rovnice $\Psi_0(a^*)^4 + 4\Psi_1(a^*)^3 + 6\Psi_2(a^*)^2 + 4\Psi_3 a^* + \Psi_4 = 0$ jsou inverzní ke kořenům rovnice (10)

$$
\begin{tikzcd}
 & & \mathrm{I} \arrow[dl] \arrow[d, Rightarrow] \\
 & \mathrm{II} \arrow[r] \arrow[dl] \arrow[d, Rightarrow] & \mathrm{D} \arrow[d] \\
\mathrm{III} \arrow[r] & \mathrm{N} \arrow[r] & 0
\end{tikzcd}
$$

Obr. 1: Penroseův diagram

Algoritmus na zjištění Petrovova typu z $\Psi_0, \ldots, \Psi_4$ [4] je založen na porovnání těchto pěti skalárů:

$$
\begin{aligned}
I &\equiv \Psi_0\Psi_4 - 4\Psi_1\Psi_3 + 3\Psi_2^2, \\
J &\equiv \begin{vmatrix} \Psi_4 & \Psi_3 & \Psi_2 \\ \Psi_3 & \Psi_2 & \Psi_1 \\ \Psi_2 & \Psi_1 & \Psi_0 \end{vmatrix}, \\
K &\equiv \Psi_1\Psi_4^2 - 3\Psi_4\Psi_3\Psi_2 + 2\Psi_3^3, \\
L &\equiv \Psi_2\Psi_4 - \Psi_3^2, \\
N &\equiv 12L^2 - \Psi_4^2 I.
\end{aligned}
$$



Obr. 2: Vývojový diagram na určení Petrovova typu

Jestliže je Ψ_4 roven nule, pak jsou v definicích invariantů navzájem zaměněny $\Psi_0 \rightleftharpoons \Psi_4$ a $\Psi_1 \rightleftharpoons \Psi_3$. Vlastní porovnání a určení Petrovova typu je uvedeno ve schématu (obr. 2). Pomocí tohoto algoritmu se bude počítat Petrovova klasifikace.

Jestliže je $\Psi_0 = \Psi_4 = 0$, pak se definuje skalár

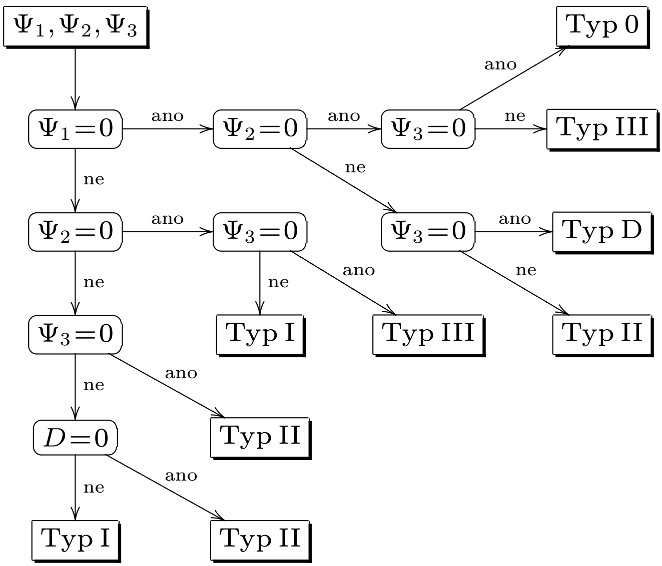
$$D \equiv 9\Psi_2^2 - 16\Psi_1\Psi_3,$$

a určení Petrovova typu probíhá podle schématu na obrázku 3.

Považuji za důležité dále zmínit skutečnost, že změnou parametru v metrice může dojít i ke změně Petrovova typu. Příkladem může být Schwarzschildova metrika, u níž pro $r \to \infty$ dostáváme typ 0.

Petrovovu klasifikaci lze samozřejmě provést i výpočtem v souřadnicové bázi (viz. [7]). Prvním krokem je výpočet komponent Weylova tenzoru, které se potom definovaným způsobem převedou na tenzor druhého řádu a porovnáním jeho vlastních hodnot dostáváme Petrovovu klasifikaci.

Z algoritmického hlediska je však mnohem výhodnější ji provádět pomocí Newman-Penrosova formalismu.



Obr. 3: Vývojový diagram na určení Petrovova typu, jestliže $\Psi_0 = \Psi_4 = 0$

3

Procedury

Tato kapitola je psána jako příručka uživatele, který pracuje s mapleovským modulem pro Petrovovu klasifikaci, jenž je vlastním výsledkem této práce. Procedury jsou roztříděny do dvou základních skupin podle matematických veličin s nimiž pracují. První část popisuje funkce, které pracují s tenzorovými veličinami v souřadnicové reprezentaci. Ve druhé části se nachází procedury, které odpovídají veličinám počítaným pomocí Newman-Penrosova tetrádního formalismu. Ve všech procedurách se využívá mapleovských knihoven *tensor* a *linalg*, proto je nutné je vyvolat ještě před samotným modulem.

3.1. Globální proměnné

Použitím globálních proměnných lze dosáhnout toho, že uvedené procedury jsou z uživatelského hlediska co nejjednodušší. Jejich použitím se totiž odstraní zadávání vstupních parametrů funkcí. Jelikož jsou tyto proměnné potřeba ve více procedurách, považuji za důležité, aby se o nich případný uživatel dověděl více, protože jejich nevědomé předefinování by vedlo ke zcela chybným výsledkům.

Na tomto místě je též vhodné podotknout, že při zjednodušování těchto proměnných je důležité správné přiřazení, aby další použité procedury počítaly se zjednodušeným tvarem dané proměnné. To lze zařídit tím, že se vstupní proměnné přiřadí totožné jméno. Pro lepší orientaci je uveden následující příklad: Je potřeba zjednodušit hodnoty Riemannova tenzoru d_RM. K tomu se použije níže uvedená procedura *jed*, která provádí zadané zjednodušování. Správné přiřazení vypadá následovně:

```
d_RM := jed(d_RM, simplify, 0):
```

Seznam globálních proměnných

d_g – základní proměnná, které je přiřazen vstupní metrický tenzor. Pracují s ní všechny funkce, pomocí nichž dostáváme tenzorové komponenty v souřadnicové reprezentaci. Je důležitá při snižování indexů.

d_ginv – této proměnné je přiřazen kontravariantní metrický tenzor, který je používán ke zvyšování indexů.

d_elmag – globální proměnná, které je přiřazen tenzor elektromagnetického pole.

d_tetrads – základní proměnná pro Newman-Penroseův (NP) formalismus. Je v ní uložena vstupní kontravariantní tetráda $[-1, 1]$, s níž se počítá ve všech dalších funkcích NP formalismu.

d_tetrads_i – proměnná, které je pomocí funkce *i_tetrad* přiřazena vypočtená kovariantní tetráda $[-1, -1]$. Případné pojmenování výstupu této funkce nemá z hlediska funkcí pracujících s touto veličinou žádný význam (jelikož z požadavku co největší jednoduchosti neobsahují žádné vstupní hodnoty).

d_gamma – této proměnné je přiřazen výstup z procedury *Riccikoef*. Je důležitá pro procedury *NPr* a *spin_koef*.

d_Psi – globální proměnná, v níž jsou uloženy hodnoty komplexních skalárů Ψ Weylova tenzoru. Vypočítává se pomocí procedury *NPr* a využívají ji funkce *petr*, *RIEMANN* a *RICCI*.

d_Phi – obsahuje hodnoty komplexních skalárů Φ Ricciho tenzoru. Určuje ji procedura *NPr* a využívají ji funkce *RIEMANN* a *RICCI*.

d_Lambda – hodnota skaláru křivosti v tetrádní reprezentaci určená procedurou *NPr*. Využívá ji funkce *RS*.

d_RM – globální označení Riemannova tenzoru křivosti v tetrádní reprezentaci.

d_C – označení Weylova tenzoru v tetrádní reprezentaci.

d_RC – globální označení Ricciho tenzoru v tetrádní reprezentaci.

d_R – označení Ricciho skaláru.

3.2. Funkce pro tenzorové veličiny

display_t

Syntaxe:

display_t(tenzor)

Popis:

Jedná se o základní funkci, která provádí výpis nenulových složek veličin, které mají strukturu vícenásobných vektorových polí. Tato pole mohou mít souřadnicový i tetrádní charakter, přičemž jediné omezení spočívá v tom, že musí být 2., 3. nebo 4. řádu, což ovšem pro dále prezentované funkce zcela postačuje. Toto omezení je způsobeno mapleovskou definicí objektu tenzor. Funkce byla napsána proto, že *Maple V release 4* neobsahuje funkci na výpis složek libovolných tenzorů a navíc funkce, která v *Maple* provádí výpis (jde o funkci *displayGR*) tenzorových komponent, nepodává úplnou informaci o daném tenzoru.

Výstupními parametry jsou:

- *name* – jméno tenzoru, jehož vlastnosti se vypisují,
- *coordinates* – souřadnicová soustava, v níž jsou složky daného tenzoru počítány,
- *character* – určuje typ tenzoru, tj. zda je tenzor v daném indexu kovariantní [−1] nebo kontravariantní [1],
- *rank of tensor* – označuje řád tenzoru,
- *non-zero komponents* – vypisuje nenulové komponenty zadaného tenzoru.

Příklad použití:

Výpis složek metrického tenzoru:

```
display_t(d_g);
```

jed

Syntaxe:

jed(tenzor, procedura, fce)

Popis:

Funkce zjednodušuje a upravuje komponenty tenzorů 2., 3. a 4. řádu, přičemž vstupními parametry jsou:

- *tenzor* – libovolný tenzor 2.–4. řádu,
- *procedura* – zjednodušující procedura,
- *fce* – pravidlo, podle kterého se provádí zjednodušení.

Pro jednotlivé procedury existují tato pravidla (viz. [2]):

- *simplify* – provádí obecné zjednodušení
  - *0* – pokud potřebujeme obecné zjednodušení,
  - *power* – zjednodušuje mocniny, exponenciály a logaritmy,
  - *radical* – zjednodušuje výrazy s mocninami s lomeným exponentem,
  - *sqrt* – zjednodušuje druhé mocniny nebo mocniny druhých odmocnin,
  - *trig* – zjednodušuje mocniny trigonometrických funkcí,
- *convert* – procedura pro konverzi výrazu na jinou *formu*
  - *exp* – konvertuje trigonometrické funkce do ekvivalentní formy, která obsahuje exponenciály,
  - *ln* – konvertuje cyklometrické funkce do tvarů, které obsahují pouze logaritmy,
  - *expln* – kombinace exp a ln,
  - *expsincos* – konvertuje trigonometrické funkce do tvarů obsahujících sin a cos a všechny hyperbolické trigonometrické funkce do tvarů obsahujících exponenciály,
  - *sincos* – konvertuje trigonometrické funkce do tvarů obsahujících jenom sin, cos, sinh a cosh,
  - *tan* – konvertuje trigonometrické funkce do tvarů obsahujících pouze tangens,
  - *trig* – konvertuje všechny exponenciály do tvarů obsahujících sin, cos, sinh a cosh podle Eulerových pravidel,

- *expand* – provede roznásobení výrazu a jeho úpravu na součet členů,
- *factor* – provede rozklad polynomu na součin kořenových činitelů.

Příklad použití:

Provedeme obecné zjednodušení komponent Riemannova tenzoru křivosti. Přiřazení je nutné proto, aby se dále počítalo se zjednodušeným tvarem tenzoru.

```
Rm := jed(Rm, simplify, 0):
```

ds

Syntaxe:
ds()

Popis:

Tato funkce slouží k zadání metrického tenzoru. Je interakční, což znamená, že uživatel postupně zadává všech deset složek metrického tenzoru. Funkce využívá mapleovskou knihovní funkci *entermatrix*. Pro určení metriky je ve funkci nastavena globální proměnná *d_g*. Proto není potřeba provádět přiřazení dané funkce (metrika se vždy vyvolá pomocí *d_g*).

Příklad použití:

Zadání Schwarzschildovy metriky:

```
ds();
```

g_inv

Syntaxe:
g_inv()

Popis:

Funkce provádí výpočet kontravariantního metrického tenzoru. Její výstup je zapsán do globální proměnné *d_ginv*.

Příklad použití:

Výpočet kontravariantní Schwarzschildovy metriky:

```
g_inv():
```

create_tensor

Syntaxe:
create_tensor(p, type)

Popis:

Funkce umožňuje vytvořit libovolný tenzor druhého řádu. Opět využívá standardní funkce *entermatrix*. Jelikož zde nejsou nastaveny žádné globální veličiny (lze předpokládat, že bude použita vícekrát a globálním nastavením by došlo k přepsání předchozího tenzoru), je nutné provést přiřazení (viz. příklad). Vstupními parametry jsou

- *p* – parametr, který určuje vlastnosti symetrie zadávaného tenzoru:
  - *symmetric* – vytvořený tenzor bude symetrický,
  - *antisymmetric* – vytvořený tenzor bude antisymetrický,
  - *general* – vytvoří obecný tenzor,
- *type* – určuje typ tenzoru:
  - $[-1,-1]$ – tenzor bude kovariantní,
  - $[1,1]$ – tenzor bude kontravariantní,
  - $[-1,1]$ – jedná se o smíšený tenzor (kovariantní v prvním a kontravariantní v druhém indexu).

Speciálním případem této funkce je funkce *ds()*, která umožňuje zadat jenom metrický tenzor.

Příklad použití:

Vytvoření antisymetrického kovariantního tenzoru F

```
F := create_tensor(antisymmetric, [-1,-1]);
```

raise, lower

Syntaxe:

raise(contra_metric, tensor, $i_1, i_2, \ldots$)
lower(cov_metric, tensor, $i_1, i_2, \ldots$)

Popis:

Pomocí těchto funkcí provádíme zvedání a snižování indexů již nadefinovaných tenzorů. Jedná se o standardní mapleovské funkce obsažené ve verzi 4 a 5. Je nutné je zde uvést, protože tyto operace patří mezi základní. Význam jednotlivých vstupních parametrů je následující:

- *contra_metric* – kontravariantní metrický tenzor,
- *cov_metric* – kovariantní metrický tenzor,
- *tensor* – tenzorový objekt, na němž chceme operaci vykonat,
- $i_1, i_2, \ldots$ – číslo indexu, u něhož má být daná operace provedena (může jich být samozřejmě i více, jedná-li se o tenzor vyššího řádu).

Příklad použití:

Provedeme zvednutí prvního a třetího indexu Riemannova tenzoru *Rm*, přičemž kontravariantní metrika má označení *d_ginv*.

```
raise(d_ginv, Rm, 1, 3):
```

ideal

Syntaxe:

ideal(u)

Popis:

Provádí výpočet tenzoru energie-hybnosti ideální kapaliny, který je dán vztahem [7]:

$$T_{ik} = (p + \varepsilon)u_i u_k - p g_{ik},$$

kde p je tlak, ε je napětí a u je čtyřrychlost. Vstupním parametrem je:

- *u* – čtyřvektor rychlosti.

Funkce využívá globální proměnnou *d_g*, která obsahuje kovariantní metrický tenzor.

Příklad použití:

```
u := [1, 0, 0, 0]:
tid := ideal(u);
```

t_elmag

Syntaxe:

t_elmag()

Popis:

Funkce provádí výpočet tenzoru elektromagnetického pole F_{ik} ze zadaného čtyřpotenciálu elektromagnetického pole A_k podle rovnice

$$F_{ik} = \frac{\partial A_k}{\partial x^i} - \frac{\partial A_i}{\partial x^k}.$$

Je interakční, takže po vyvolání stačí zadat komponenty příslušného čtyřpotenciálu. Výstup je ukládán do globální veličiny *d_elmag*, která je potřebná pro funkci *T_eh_elm*.

Příklad použití:

```
t_elmag();
```

T_eh_elm

Syntaxe:

T_eh_elm()

Popis:

Funkce zavádí tenzor energie-hybnosti elektromagnetického pole podle vztahu

$$T_{ik} = \frac{1}{4\pi} \quad -F_{il}F_{k}^{l} + \frac{1}{4}F_{lm}F^{lm}g_{ik} \quad ,$$

přičemž se předpokládá, že F_{ik} je již zadané pomocí funkce *t_elmag*. Dále je nutné, aby byl zadán kovariantní metrický tenzor g_{ik} (funkce *ds()*) a kontravariantní metrický tenzor g^{ik}, který se ve funkci používá pro zvedání indexů.

Příklad použití:

```
TEH := T_eh_elm():
TEH := jed(TEH, simplify, 0):
```

coords

Popis:

Globální proměnná, která udává souřadný systém, v němž se provádí výpočet. Je nutné ji použít již na počátku výpočtů, protože s ní pracují úplně všechny procedury. Dále je potřeba dávat pozor na případné kolizní situace, které nastanou, pokud se někde uprostřed výpočtu *coords* předefinuje a dále se počítá s předchozími veličinami.

Příklad použití:

```
coords := [t, r, theta, phi];
```

tensorsGR

Syntaxe:

tensorsGR(coords, d_g, d_ginv, det_met, C1, C2, Rm, Rc, R, G, C)

Popis:

Standardní mapleovská procedura na výpočet nejdůležitějších tenzorů z obecné teorie relativity. Vstupním parametrem je pouze metrika *d_g* a soustava souřadnic *coords*. Ostatní veličiny uváděné v závorce jsou funkcí vypočteny.

Význam jednotlivých parametrů:

- *coords* – souřadnicová soustava,
- *d_g* – kovariantní metrický tenzor,
- *d_ginv* – inverzní (kontravariantní) metrika,
- *det_met* – determinant metriky,
- *C1* – Christoffelovy symboly prvního druhu,
- *C2* – Christoffelovy symboly druhého druhu,
- *Rm* – Riemannův tenzor křivosti,
- *Rc* – Ricciho tenzor,
- *R* – Ricciho skalár,
- *G* – Einsteinův tenzor,
- *C* – Weylův tenzor.

Před zadáním procedury je nutné mít nadefinované souřadnice a metriku!

Příklad použití:

```
tensorsGR(coords,d_g,d_ginv,det_met,C1,C2,Rm,Rc,R,G,C):
```

display_allGR

Syntaxe:

display_allGR(coords, d_g, d_ginv, det_met, C1, C2, Rm, Rc, R, G, C)

Popis:

Funkce provádí výpis nenulových komponent všech veličin vypočtených pomocí *tensorsGR*. Jedná se o standardní mapleovskou proceduru. Význam formálních parametrů je stejný jako u funkce *tensorsGR*.

Příklad použití:

```
display_allGR(coords,d_g,d_ginv,det_met,C1,C2,Rm,Rc,R,G,C);
```

einstein_eq

Syntaxe:
einstein_eq(G, energ_hyb)

Popis:

Funkce vypisuje Einsteinovy rovnice gravitačního pole, přičemž vstupními parametry jsou

- *G* – Einsteinův tenzor,
- *energ_hyb* – libovolný tenzor energie-hybnosti.

Einsteinův tenzor je nutné vypočítat pomocí mapleovské funkce *tensors_GR*, ve které je použito odlišné definice Ricciho tenzoru (úžení Riemannova tenzoru provádí přes první a čtvrtý index, čímž dochází ke změně znaménka). Ve funkci je proto před každou komponentou Einsteinova tenzoru znaménko minus. Pokud by se výpočet Riemannova tenzoru provedl úžením přes první a třetí index, jak je běžné, dostali bychom ve výpise rovnice s opačným znaménkem a na to je třeba dávat pozor.

Příklad použití:

```
einstein_eq(G, TEH);
```

3.3. Funkce pro tetrádní veličiny

tetrad

Syntaxe:
tetrad()

Popis:

Funkce interaktivně definuje izotropní tetrády, které jsou nutné pro výpočet Petrovovy klasifikace. Je definována jako funkce *ds()*, ale výstupní tenzorový tvar je smíšeného typu $[-1,1]$. Této funkci je přiřazena globální veličina *d_tetrads*, která se využívá ve všech dalších funkcích, a proto je nutné dávat pozor na její případné předefinování. Pro zadanou tetrádu se dále předpokládá, že splňuje vlastnosti, které jsou na ni kladené v kapitole 2.4.

Příklad použití:

```
tetrad();
```

i_tetrad

Syntaxe:
i_tetrad()

Popis:

Funkce vypočítá k výše zadané izotropní tetrádě její kovariantní tvar. Algoritmus je založen na vztahu z kapitoly 2.4. Přiřazená globální veličina je *d_tetrads_i*.

Příklad použití:

```
i_tetrad();
```

Riccikoef

Syntaxe:

Riccikoef()

Popis:

Funkce provádí výpočet Ricciho koeficientů γ, které jsou důležité pro výpočet Riemannova tenzoru v tetrádním tvaru. Pro dosažení kompaktnějších (jednodušších) tvarů je vhodné po výpočtu použít zjednodušující proceduru *jed*. Ricciho koeficienty jsou na výstupu uloženy jako kovariantní tenzory třetího řádu (tj. typ $[-1, -1, -1]$). Před použitím je nutné, aby byla nadefinována kontravariantní a kovariantní tetráda, protože funkce používá globální proměnné *d_tetrads* a *d_tetrads_i*. Výsledek je přiřazen globální proměnné *d_gamma*, takže při následném zjednodušování je nutné provést přiřazení do této proměnné.

Příklad použití:

```
Riccikoef():
```

spin_koef

Syntaxe:

spin_koef()

Popis:

Funkce provádí výpočet spinových koeficientů, které přímo dostáváme přiřazením z Ricciho koeficientů. Výstup funkce je řešen dvěma způsoby: napřed je proveden výpis jednotlivých spinových koeficientů a ty jsou potom přiřazeny jako vektor zadané veličině (v příkladu je to veličina SP).

Příklad použití:

```
SP := spin_koef();
```

NPr

Syntaxe:

NPr()

Popis:

Jedná se o nejdůležitější proceduru z hlediska výpočtu Petrovovy klasifikace (jestliže se vychází z tetrádní báze). Výstupními parametry jsou globální proměnné *d_Psi*, *d_Phi*, *d_Lambda*, které jsou použity v procedurách na výpočet Riemannova, Weylova a Ricciho tenzoru. V proceduře se řeší Newman-Penrosovy rovnice, takže je časově nejnáročnější.

Příklad použití:

```
NPr();
```

petr

Syntaxe:

petr()

Popis:

Funkce určuje Petrovův typ podle klasifikace pěti komplexních skalárů Ψ, které plně popisují Weylův tenzor. Schémata algoritmů, podle nichž se typ určuje jsou uvedena v kapitole 2.5. Ve funkci je použita globální proměnná *d_Psi* (proto se nezadává žádný vstupní parametr), která je výsledkem funkce *Psi_koef* či *NPr*.

Příklad použití:

```
petr();
```

petrLim

Syntaxe:

petrLim(vel, lim)

Popis:

Tato funkce provádí Petrovovu klasifikaci pro případ limitní hodnoty některé proměnné, nacházející se v metrice. Má dva vstupní parametry:

- *vel* – proměnná veličina, pro niž se počítá limitní hodnota,
- *lim* – limitní hodnota.

Funkce je napsána proto, že některé limity závisí na parametru, přičemž s měnícím se parametrem dochází též ke změně Petrovova typu.

Příklad použití:

```
petrLim(r, infinity);
```

t_tetrad

Syntaxe:

t_tetrad(C)

Popis:

Provádí transformaci složek Weylova tenzoru do tetrádního tvaru. Vstupem je Weylův tenzor spočtený pomocí funkce *tensorsGR*.

Příklad použití:

```
Weyl_tetr := t_tetrad(C):
```

Psi_koef

Syntaxe:

Psi_koef(C)

Popis:

Pomocí této funkce se z Weylova tenzoru C v tetrádním tvaru určí komplexní skaláry Ψ, které jsou nutné pro výpočet Petrovovy klasifikace. Používá se jen v případě, že vycházíme z metriky *d_g* a tetrádní tvar Weylova tenzoru počítáme užitím funkce *WEYL*. Pokud výpočet začínáme v tetrádní reprezentaci, o určení těchto koeficientů se stará funkce *NPr*. Výstup je přiřazen globální proměnné *d_Psi*, se kterou již přímo pracuje funkce určující Petrovovu klasifikaci.

Příklad použití:

```
Psi_koef(Weyl_tetr);
```

RIEMANN

Syntaxe:
RIEMANN()

Popis:

Provádí výpočet komponent Riemannova tenzoru křivosti v tetrádní reprezentaci. Výsledek výpočtu je přiřazen do globální proměnné *d_RM*. Protože se k výpočtu používají skaláry Ψ Weylova tenzoru a Φ Ricciho tenzoru, je nutné nejprve použít funkci *NPr*, která tyto veličiny produkuje. Je též vhodné získaný výsledek dále zjednodušit pomocí funkce *jed*.

Příklad použití:

```
RIEMANN():
```

WEYL

Syntaxe:
WEYL()

Popis:

Funkce provádí výpočet komponent Weylova tenzoru v tetrádní reprezentaci. Její výstup je zapsán do globální proměnné *d_C*. Protože používá skaláry Ψ, je nutné napřed vyvolat funkci *NPr*. V dalším kroku je vhodné použít zjednodušující funkci *jed*, která uvede výsledek v kompaktnějším tvaru.

Příklad použití:

```
WEYL():
```

RICCI, RS

Syntaxe:

RICCI()
RS()

Popis:

Funkce *RICCI* vypočítá nenulové komponenty Ricciho tenzoru v tetrádním tvaru a přiřadí je do globální proměnné *d_RC*, která má tvar kovariantního tenzoru $[-1,-1]$. Před jejím použitím je nutné zavolat funkci *NPr*, neboť se ve výpočtu využívá jejích výsledků.

Funkce *RS* určuje hodnotu Ricciho skaláru křivosti R, kterou přiřadí do globální proměnné *d_R*.

Příklad použití:

```
RICCI():
RS();
```

4

Závěr

Úkolem diplomové práce bylo vytvoření programového modulu pro Petrovovu klasifikaci prostoročasů. Zadaný úkol byl realizován pomocí programu *Maple V release 4*.

Výsledkem práce je programový modul, který umožňuje určit Petrovův typ s minimálními nároky na znalosti uživatele (myšleno ve vztahu k výpočetní technice). Výsledný modul je přeložen pro čtvrtou i pro pátou verzi *Maple V*, která byla uvolněna během řešení problému. Nové funkce, kterých modul využívá k řešení Petrovovy klasifikace, jsou založeny na *Newman-Penrosově* (NP) tetrádním formalismu.

Petrovův typ lze však získat i řešením ve standardní souřadnicové bázi, přičemž potřebné veličiny (Weylův tenzor) se po výpočtu převedou do báze tetrádní. Tím bylo využito i standardních mapleovských funkcí, které vytvořený programový modul doplňuje o funkce pracující v NP-tetrádní bázi.

Funkce jsou optimalizovány tak, aby bylo jejich zadávání jednoduché a výpočet co nejkratší. Proto jsou téměř ve všech využity globální proměnné, jejichž nadefinováním bylo dosaženo efektivnějšího vyvolávání funkcí. Jediným stinným bodem takového přístupu je fakt, že uživatel musí dávat pozor na případné předefinování takové veličiny, a v případě zjednodušování výrazů (které je u složitějších metrik nutné) na správné přiřazení. Zkrácení času nutného pro výpočet je ve většině případů dosaženo tím, že se uvnitř algoritmů nepoužívá zjednodušování výrazů. Při řešení bylo totiž zjištěno, že je efektivnější napřed provést výpočet a až potom výsledný výraz zjednodušit. Samozřejmě platí, že čím je metrika složitější, tím náročnější výpočty jsou (ve funkcích je větší množství nenulových členů, které se na výpočtu podílí) a výpočetní čas tím narůstá. Na současných počítačích je však doba výpočtu pro všechny testované metriky přijatelná.

Otevřeným problémem předkládané práce zůstává algoritmus procedury, která by řešila „výpočet“ izotropních tetrád, jenž jsou vstupními veličinami nutnými k celému dalšímu výpočtu Petrovovy klasifikace.

5

Dodatky

5.1. Příloha

Veškeré výpočty prováděné pomocí programového modulu pro zjišťování Petrovovy klasifikace prostoročasů (*modul4.m* pro *Maple V release 4* či *modul5.m* pro *Maple V release 5*) byly kontrolovány pomocí mapleovského modulu *GRTensorII* pro práci s relativistickými objekty. Tento modul není standardní součástí *Maple V*, ale lze jej získat jako free software na adrese

`http://astro.queensu.ca/~grtensor/GRHome.html`

včetně manuálů k použití. K další kontrole bylo použito databáze známých metrik, ve které jsou uvedeny jak typy podle Petrova, tak i Segrèho klasifikace Ricciho tenzoru. Tato databáze se nachází na adrese

`http://edradour.symbcomp.eurj.br/`

Součástí diplomové práce je také disketa, na níž je umístěn příslušný modul pro Petrovovu klasifikaci, přičemž
- *modul4.m* je určen pro práci v prostředí *Maple V release 4*,
- *modul5.m* je určen pro práci v prostředí *Maple V release 5*.

Dále jsou na disketě umístěny příkladové mapleovské pracovní listy (worksheets), ve kterých je řešena Petrovova klasifikace několika známých metrik (např. Schwarzschildova, Kerrova, Gödelova a další).

5.2. Příklad na použití modulu

Uvedu příklad, jak použít modul ke zjištění Petrovova typu zadané metriky. Za metriku bylo vybráno Kerrovo řešení.

Výpočet začíná načtením potřebných programových balíků, které jsou součástí *Maple V* a jež využívá samotný modul pro Petrovovu klasifikaci. Jedná se o moduly *tensor* a *linalg*.

```
>restart;
>with(tensor):
>with(linalg):
```

Nyní se vyvolá samotný modul pro Petrovovu klasifikaci (v parametru funkce *read* je nutno zadat úplnou cestu).

```
>read (‘modul4.m‘):
```

Teď se provede nadefinování báze, v níž se bude celý výpočet provádět

```
>coords := [t,r,theta,phi]:
```

Definice proměnných, které budou sloužit ke snadnějšímu zápisu NP-tetrád

```
>Delta := r^2-2*M*r+a^2:
>Sigma := sqrt(r^2+a^2*cos(theta)^2):
```

Zadání kontravariantní NP-tetrády

```
>tetrad();
enter element 1,1> (r^2+a^2)/Delta;
enter element 1,2> 1;
enter element 1,3> 0;
enter element 1,4> a/Delta;
enter element 2,1> (r^2+a^2)/(2*Sigma^2);
enter element 2,2> -Delta/(2*Sigma^2);
enter element 2,3> 0;
enter element 2,4> a/(2*Sigma^2);
enter element 3,1> I*a*sin(theta)*sqrt(2)/(r*I*a*cos(theta));
enter element 3,2> 0;
enter element 3,3> 1/((r+I*a*cos(theta))*sqrt(2));
enter element 3,4> I*csc(theta)/((r+I*a*cos(theta))*sqrt(2));
enter el. 4,1> -I*a*sin(theta)/((r-I*a*cos(theta))*sqrt(2));
enter element 4,2> 0;
enter element 4,3> 1/((r-I*a*cos(theta))*sqrt(2));
enter el. 4,4> -I*csc(theta)/((r-I*a*cos(theta))*sqrt(2));
```

Výpočet kovariantní NP-tetrády a její zjednodušení

```
>i_tetrad():
>d_tetrads_i := jed(d_tetrads_i, simplify, 0):
```

Výpočet Ricciho koeficientů γ a jejich zjednodušení

```
>Riccikoef():
>d_gamma := jed(d_gamma, simplify, 0):
>d_gamma := jed(d_gamma, factor, 0):
```

Výpočet Ψ koeficientů potřebných pro Petrovovu klasifikaci

```
>NPr():
```

Vlastní určení Petrovova typu

```
>petr();
```

Petrov type D

Petrovův typ pro případ, že se hodnota poloměru r blíží nekonečnu (dostáváme plochý prostoročas) určíme následovně:

```
>petrLim(r, infinity);
```

For r → infinity
Petrov type 0

6

Literatura

[1] L. Boček: *Tenzorový počet*, SNTL, Praha 1976.
[2] J. Buchar a kol.: *Úvod do programového souboru Maple V*, VŠZ Brno 1994.
[3] R. d'Inverno: *Introducing Einstein's Relativity*, Oxford University Press 1992.
[4] R. d'Inverno, R. A. Russel-Clark: *Classification of the Harrison metrics*, J. Math. Phys. 12, 1258 (1971).
[5] S. Chandrasekhar: *The mathematical theory of black holes*, Oxford University Press 1983 (ruský překlad)
[6] D. Kramer, H. Stephani, M. MacCallum, E. Hertl: *Exact solutions of the Einsteins field equations*, Deutscher Verlag der Wissenschaften, Berlin 1980 (ruský překlad)
[7] L. D. Landau, E. M. Lifšic: *Teorie pole*, Nauka, Moskva 1973 (rusky)
[8] E. Newman, R. Penrose, J. Math. Phys 3, 366 (1962)
[9] A. Z. Petrov: *Nové metody v obecné teorii relativity*, Nauka, Moskva 1966 (rusky)
[10] K. Rektorys a kol.: *Přehled užité matematiky*, díl I, PROMETHEUS, Praha 1995

7

Rejstřík