Numerički račun savijanja putanja svjetlosti zbog gravitacije i nelinearne elektrodinamike

Kičić, Ivica

Master's thesis / Diplomski rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:217:887680

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

Repository of the Faculty of Science - University of Zagreb

SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO-MATEMATIČKI FAKULTET
FIZIČKI ODSJEK

Ivica Kičić

Numerički račun savijanja putanja svjetlosti zbog gravitacije i nelinearne elektrodinamike

Diplomski rad

Zagreb, 2016.

SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO-MATEMATIČKI FAKULTET
FIZIČKI ODSJEK

SMJER: Istraživački studij fizike

Ivica Kičić

Diplomski rad

Numerički račun savijanja putanja svjetlosti zbog gravitacije i nelinearne elektrodinamike

Voditelj diplomskog rada: doc. dr. sc. Maro Cvitan

Ocjena diplomskog rada: _____

Povjerenstvo: 1. _____

2. _____

3. _____

Datum polaganja: _____

Zagreb, 2016.

Zahvaljujem mentoru doc. dr. sc. Mari Cvitanu na pomoći, strpljenju i podršci prilikom pisanja ovog rada. Zahvaljujem dr. sc. Petru Mimici na korisnim savjetima i diskusiji o akrecijskim diskovima. Zahvalio bih svima koji su sudjelovali u mom obrazovanju i svojoj obitelji na konstantnoj podršci.

Sažetak

Masivne zvijezde i crne rupe svojom gravitacijom utječu na putanje svjetlosti u njihovoj blizini. Trajektorija svjetlosti može se izračunati koristeći rezultate opće teorije relativnosti. U ovom radu koristimo metodu praćenja zraka svjetlosti (*ray tracing*) u zakrivljenom prostorvremenu oko neutronskih zvijezda i crnih rupa. Promatramo kako gravitacija utječe na sliku akrecijskih diskova. Uz to, promatramo dodatan efekt savijanja putanja svjetlosti u blizini izrazito jako magnetiziranih neutronskih zvijezda ($\sim 10^{11}$ T na površini) zbog nelinearne elektrodinamike.

Numerical calculation of light deflection due to gravity and nonlinear electrodynamics

Abstract

Massive stars and black hole deflect nearby light with their gravity. The light trajectory can be calculated using the results of the general theory of relativity. In this work we perform the ray tracing of light rays in curved spacetime around neutron stars and black holes. We analyze how gravity affects the images of accretion disks. Furthermore, we analyze the additional effect of deflection of light in the vicinity of strongly magnetized neutron stars ($\sim 10^{11}$ T on the surface) due to nonlinear electrodynamics.

Sadržaj

1 Uvod **1**

2 Gravitacijsko savijanje putanja svjetlosti **3**
2.1 Metrika 3
2.1.1 Ravna metrika 4
2.1.2 Schwarzschildova metrika 4
2.1.3 Kerrova metrika 5
2.2 Paralelni transport, geodezici i gibanje svjetlosti 6

3 Ray tracing **7**
3.1 Kamera 7
3.2 Integracija geodezika 8
3.3 Provjera točnosti integracije geodezika 9

4 Crtanje akrecijskog diska **11**
4.1 Pitanje promatrača i izvora svjetlosti 11
4.2 Model diska 13
4.3 Postupak za jednu zraku svjetlosti 13
4.4 Rezultati 14
4.5 Shakura-Sunyaev model diska 16

5 Nelinearna elektrodinamika **18**
5.1 Efektivna metrika 19
5.2 Euler-Heisenbergov Lagrangian u najnižem redu 21
5.3 Potpuni Euler-Heisenbergov Lagrangian 22
5.4 Intenzitet efekta nelinearne elektrodinamike 24
5.5 Magnetsko polje 24
5.6 Rezultati 26
5.7 Ograničenja i procjena nepouzdanosti računa 28
5.8 Komentar o energetskoj gustoći magnetskog polja 29

6 Zaključak **30**

Dodaci **31**

A Runge-Kutta-Fehlberg integracija 4./5. reda **31**

B Automatsko deriviranje **33**

C Implementacijski detalji **35**
C.1 Ubrzavanje i unapređivanje crtanja adaptivnom metodom 37

D Dodatne slike putanja svjetlosti i diskova **38**

E Primjer kôda za crtanje diska **41**

1 Uvod

Masivna tijela stvaraju oko sebe gravitacijsko polje kojima privlače okolne objekte. To privlačenje se odnosi i na svjetlost, jer akceleracija koju objekt prima zbog gravitacijskog polja ne ovisi o njegovoj masi. Za pravilan opis tog efekta koristimo opću teoriju relativnosti, Einsteinovu teoriju gravitacije. Ta teorija tvrdi da je potrebno na svemir gledati kao na 4D objekt, gdje je vrijeme stavljeno praktički u ravnopravan položaj s tri prostorne koordinate. Teorija zatim kaže da se gravitacija manifestira kao savijanje tog 4D prostorvremena, a ne kao standardna sila. Odnosno, masivna tijela savijaju prostorvrijeme, a zatim se druga tijela (poput svjelosti, manjih tijela i dr.) gibaju u slobodnom padu, "pravocrtno" u tom zakrivljenom prostoru. Takav pristup se pokazao izrazito uspješnim u objašnjavanju i predviđanju mjerenih veličina te je riješio razne probleme u kojima se newtonovska teorija gravitacije ispostavila nedovoljnom ili netočnom.

Ovaj rad se orijentira na pitanje savijanja svjetlosti u blizi masivnih tijela poput neutronskih zvijezda i crnih rupa. Sastoji se od dva dijela. U prvom dijelu govorimo o prikazivanju akrecijskog diska u blizini rotirajuće crne rupe, gdje vodimo računa o Dopplerovom efektu, intenzitetu i polarizaciji. Ograničavamo se na (infinitezimalno) tanke i neprozirne diskove. Koristimo proceduru opisanu u [1] te implementiramo isti model diska kao u navedenom članku. Kao dodatan primjer, crtamo pojednostavljenu verziju Shakura-Sunyaev modela diska [2], gdje pretpostavljamo da je cijeli disk opisan istim formulama (istim režimom), da je neproziran i da spektar odgovara zračenju crnog tijela. Jednom kada je napisan program za crtanje takvih diskova, po potrebi se lagano mogu implementirati i drugi modeli.

Drugi dio rada se odnosi na nelinearnu elektrodinamiku i efekt dodatnog savijanja svjetlosti zbog prisustva izrazito jakih magnetskih polja u blizini određenih neutronskih zvijezda. Magnetska polja koja se pojavljuju mogu biti veća i od 10^9 T (za usporedbu, najveće polje ikad napravljeno na Zemlji je reda veličine 10^3 T [3]). Mi konkretno proučavamo kut otklona svjetlosti za zvijezdu SGR 1806-20, najmagnetiziraniju otkrivenu zvijezdu, s površinskim magnetskim poljem od $B = 2 \times 10^{11}$ T [4]. Razni autori su već proučavali utjecaj nelinearne elektrodinamike na putanju svjetlosti [5–10], ali se ili ograničavaju na slabija polja $B \ll B_c = 4.4 \times 10^9$ T, gdje je B_c kritično magnetsko polje (vidi poglavlje 5), ili se ograničavaju na ravno prostorvrijeme. U ovom radu pokušavamo izračunati kako bi se svjetlost ponašala u zakrivljenom prostoru s poljima jakosti $B > B_c$. Očekujemo da će efekt biti jači u zakrivljenom prostoru, jer, zbog gravitacijskog savijanja, svjetlost provodi više vremena u područjima jakog polja.

Program za računanje trajektorija i za crtanje diskova napisan je u C++-u. Osmišljen i realiziran je tako da bude neovisan o konkretnoj metrici i koordinatnom sustavu koji se koriste, gdje je god to bilo moguće. Na taj način je primjerice dozvoljena mogućnost eventualnog proučavanja proširenja opće teorije relativnosti. U tom slučaju,

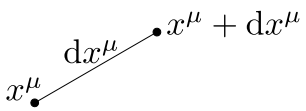
za unos nove metrike bilo bi dovoljno implementirati funkciju za račun matrice metrike $g_{\mu\nu}$ i inverza $g^{\mu\nu}$, dok se objekti poput Christoffelovih simbola $\Gamma^{\rho}{}_{\mu\nu}$ automatski računaju pomoću *automatskog deriviranja* (vidi Dodatak B). Naravno, u prvom dijelu rada gdje govorimo o akrecijskim diskovima, modeli diskova su usko vezani uz same metrike pa bi poopćavanje na druge metrike moglo biti vrlo složeno. Inače, ograničavanje na fiksnu metriku (poput primjerice Kerrove, rotirajuće crne rupe) bi moglo dati nešto brži ili precizniji program. No, nama nije bio cilj imati što moguće bržu implementaciju. Osim toga, budući da radimo s nelinearnom elektrodinamikom, koja efektivno mijenja metriku, ograničavanje na određenu metriku ne bi imalo smisla.

Rad je strukturiran na sljedeći način. Na početku, u 2. poglavlju, govorimo općenito o gravitaciji, metrici i gravitacijskom savijanju putanja svjetlosti. U 3. poglavlju kažemo par riječi o ray tracing-u, kako postavljamo kameru i kako točno numerički računamo trajektorije. U 4. poglavlju objašnjavamo postupak crtanja akrecijskih diskova, koji je zakompliciran činjenicom da radimo u savijenom prostorvremenu. Detaljno opisujemo postupak crtanja diska iz [1], uključujući računanje polarizacije. Prikazujemo ovisnost ukupne polarizacije o kutu gledanja. Na kraju ukratko opisujemo Shakura-Sunyaev model diska te prikazujemo distribuciju temperature i intenziteta. U 5. poglavlju govorimo o nelinearnoj elektrodinamici, o pojmu efektivnog Lagrangiana i efektivne metrike. Analiziramo jačinu nelinearnog efekta te ga koristimo za računanje otklona svjetlosti u blizini magnetiziranih neutronskih zvijezda. Rezultat uspoređujemo s efektom gravitacijskog savijanja svjetlosti. Na kraju dajemo zaključak i komentiramo kako bi se ova tema dalje mogla razraditi.

2 Gravitacijsko savijanje putanja svjetlosti

Gravitacijsko savijanje putanja svjetlosti opisujemo općom teorijom relativnosti. Radi se o klasičnoj teoriji baziranoj na ideji da svemir treba promatrati kao 4D objekt, gdje se uz tri prostorne dimenzije dodaje i vremenska, na način da su sve četiri dimenzije ravnopravne – u smislu da se u formulama ne ističe niti jedna komponenta odvojeno. Osim toga, važan koncept opće teorije relativnosti je invarijantnost formula na izbor koordinatnog sustava. Primjerice, formule moraju imati istu formu (i moraju davati iste rezultate) neovisno o tome jesmo li odlučili koristiti Kartezijev koordinatni sustav ili sferni. Opća relativnost se stoga koristi tenzorskim formalizmom. Najvažniji tenzor i općenito najvažniji objekt u općoj teoriji relativnosti je metrika $g_{\mu\nu}$, koja govori o duljinama infinitezimalnih linija. Informacija o duljinama je baza za daljnju raspravu o zakrivljenosti prostora.

2.1 Metrika



Slika 2.1: Poznavanje duljine općenite infitezimalne dužine je osnova za rad sa zakrivljenim prostor(vremen)om.

Metrika je 2-tenzor koji definira normu $\mathrm{d}s$ općenite infinitezimalne linije $\mathrm{d}x^\mu$ (slika 2.1):

$$\mathrm{d}s^2 = g_{\mu\nu}\mathrm{d}x^\mu \mathrm{d}x^\nu,$$

gdje se podrazumijeva tzv. *Einsteinova konvencija o sumaciji*. Predznak vrijednosti $\mathrm{d}s^2$ govori o kauzalnosti te linije[1]. Za $\mathrm{d}s^2 < 0$, kažemo da je interval $\mathrm{d}x^\mu$ *vremenski*, za $\mathrm{d}s^2 = 0$ da je *svjetlosni*, a za $\mathrm{d}s^2 > 0$ *prostorni*. Trajektorije masivnih čestica mogu biti sastavljene isključivo od vremenskih intervala, a bezmasenih od svjetlosnih ili vremenskih (ako su usporene zbog interakcije s drugim česticama). Odnosno, označimo li s $x^\mu(\lambda)$ trajektoriju neke čestice (u Kartezijevom sustavu bi vrijedilo $x^0 = ct$, $x^1 = x$, $x^2 = y$, $x^3 = z$), tada nužno vrijedi

$$\mathrm{d}s^2(\lambda) = g_{\mu\nu}(x(\lambda))\,\mathrm{d}x^\mu(\lambda)\,\mathrm{d}x^\nu(\lambda) \leq 0,$$

ili drugim riječima, uvedemo li oznaku $k^\mu = \mathrm{d}x^\mu/\mathrm{d}\lambda$, tada je

$$k^2 = g_{\mu\nu}k^\mu k^\nu \leq 0,$$

gdje se jednakost postiže samo ako je čestica bezmasena.

[1]Koristimo konvenciju predznaka (-+++).

2.1.1 Ravna metrika

U radu radimo s tri metrike - ravnom, Schwarzschildovom (nerotirajuća crna rupa) i Kerrovom metrikom (rotirajuća crna rupa). Najjednostavnija, ravna metrika $g_{\mu\nu}$ i njen inverz $g^{\mu\nu}$ u Kartezijevom koordinatnom sustavu glase:

$$g_{\mu\nu} = g^{\mu\nu} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Ista metrika zapisana u sfernom sustavu glasi:

$$g_{\mu\nu} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & r^2 & 0 \\ 0 & 0 & 0 & r^2 \sin^2\theta \end{pmatrix}, \qquad g^{\mu\nu} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/r^2 & 0 \\ 0 & 0 & 0 & 1/r^2 \sin^2\theta \end{pmatrix}.$$

2.1.2 Schwarzschildova metrika

Osnovna netrivijalna metrika je Schwarzschildova metrika [11], koja opisuje gravitacijsko polje sferno simetrične, nenabijene i nerotirajuće crne rupe. U sfernom sustavu ($x^\mu = (ct, r, \theta, \varphi)^\mu$) ona glasi:

$$\begin{aligned} g_{\mu\nu} &= \begin{pmatrix} -(1-\frac{r_s}{r}) & 0 & 0 & 0 \\ 0 & (1-\frac{r_s}{r})^{-1} & 0 & 0 \\ 0 & 0 & r^2 & 0 \\ 0 & 0 & 0 & r^2\sin^2\theta \end{pmatrix}, \\ g^{\mu\nu} &= \begin{pmatrix} -(1-\frac{r_s}{r})^{-1} & 0 & 0 & 0 \\ 0 & (1-\frac{r_s}{r}) & 0 & 0 \\ 0 & 0 & \frac{1}{r^2} & 0 \\ 0 & 0 & 0 & \frac{1}{r^2\sin^2\theta} \end{pmatrix}, \end{aligned} \tag{2.1}$$

gdje je $r_s = 2GM/c^2$ Schwarzschildov radijus, $G = 6.674 \times 10^{-11}\,\mathrm{N\,m^2\,kg^{-2}}$ gravitacijska konstanta, a M masa crne rupe.

Veličina r_s definira horizont crne rupe — jednom kada neko tijelo pređe horizont i uđe u crnu rupu, više nikako ne može van. Matematički, za $r < r_s$, koordinatama t i r se promijeni predznak u metrici, zbog čega u nekom smislu možemo reći da su im se zamijenile uloge — r koordinata postaje "vremenska" i ne možemo utjecati na smjer gibanja te koordinate. Da bismo dobili osjećaj redova veličine, za masu Sunca ($M_\odot = 1.989 \times 10^{30}\,\mathrm{kg}$), Schwarzschildov radijus iznosi samo $r_s = 3\,\mathrm{km}$. Stoga, za Sunce konkretno, efekti opće teorije relativnosti nisu veliki ni blizu površine ($R_\odot = 695\,700\,\mathrm{km}$) [12]. S druge strane, neutronske zvijezde imaju radijus koji je tek oko 3 puta veći od njihovog Schwarzschildovog radijusa, zbog čega su efekti opće teorije

relativnosti u blizini površine vrlo značajni. Za masivne zvijezde vrijedi ista metrika (2.1), koja je valjana samo izvan zvijezde. Zakrivljenost prostora unutar zvijezde nas u ovom radu neće zanimati.

2.1.3 Kerrova metrika

Za opis crnih rupa s velikim angularnim momentom, ili drugih brzo rotirajućih tijela potrebno je koristiti Kerrovu metriku [11, 13]. Ona je zapisana u modificiranom sfernom sustavu, tzv. Boyer-Lindquist koordinatnom sustavu $x^\mu = (ct, r, \theta, \varphi)^\mu$:

$$\begin{aligned} x &= \sqrt{r^2 + a^2}\sin\theta\cos\varphi, \\ y &= \sqrt{r^2 + a^2}\sin\theta\sin\varphi, \\ z &= r\cos\theta, \end{aligned} \tag{2.2}$$

gdje je $a = J/M$ angularni moment po jedinici mase crne rupe. Kerrova metrika u gore navedenim koordinatama glasi:

$$g_{\mu\nu} = \begin{pmatrix} -(1 - \frac{r_s r}{\Sigma}) & 0 & 0 & -\frac{r_s a r \sin^2\theta}{\Sigma} \\ 0 & \frac{\Sigma}{\Delta} & 0 & 0 \\ 0 & 0 & \Sigma & 0 \\ -\frac{r_s a r \sin^2\theta}{\Sigma} & 0 & 0 & (r^2 + a^2 + \frac{r_s a^2 r \sin^2\theta}{\Sigma})\sin^2\theta \end{pmatrix}, \tag{2.3}$$

$$g^{\mu\nu} = \begin{pmatrix} -\frac{A}{\Sigma\Delta} & 0 & 0 & -\frac{r_s a r}{\Sigma\Delta} \\ 0 & \frac{\Delta}{\Sigma} & 0 & 0 \\ 0 & 0 & \frac{1}{\Sigma} & 0 \\ -\frac{r_s a r}{\Sigma\Delta} & 0 & 0 & \frac{\Delta - a^2\sin^2\theta}{\Sigma\Delta\sin^2\theta} \end{pmatrix},$$

gdje su uvedene pokrate $\Sigma = r^2 + a^2\cos^2\theta$, $\Delta = r^2 - r_s r + a^2$, $A = (r^2 + a^2)\Sigma + r_s a^2 r \sin^2\theta$, $r_s = 2GM/c^2$.

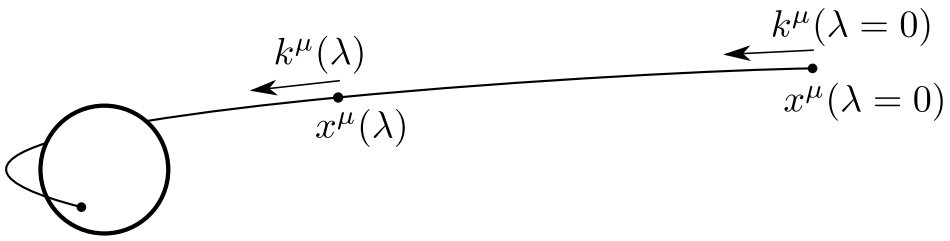
Za razliku od prijašnjih metrika, ova metrika ima miješani član $g_{t\varphi} = g_{\varphi t}$, čime postaje djelomično jasno zašto se uvodi pojam zakrivljenog prostorvremena, a ne samo zakrivljenog prostora. Naime, zbog tog miješanog člana, rotirajuća crna rupa ima izvan horizonta tzv. ergosferu, regiju unutar koje bi za česticu ili promatrača koji stoji na mjestu (ima fiksne r, θ, φ koordinate) vrijedilo $k^2 = (\mathrm{d}x/\mathrm{d}\lambda)^2 > 0$, što nije kauzalno dozvoljeno. U ovom slučaju to znači da cijeli prostor rotira zajedno s crnom rupom toliko brzo da nije moguće stajati na mjestu. Konkretno, u ergosferi ni sama se svjetlost ne može gibati suprotno rotaciji crne rupe.

Horizont rotirajuće crne rupe nalazi se na

$$r_{\text{horizon}} = \frac{r_s + \sqrt{r_s^2 - 4a^2}}{2}, \tag{2.4}$$

gdje je $r_s = 2MG/c^2$ Schwarzschildov radijus. Radijus ergosfere ovisi o θ koordinati te iznosi

$$r_{\text{ergosphere}} = \frac{r_s + \sqrt{r_s^2 - 4a^2\cos^2\theta}}{2}.$$



Slika 2.2: Skica problema od interesa — kako uz zadanu početnu poziciju svjetlosti $x^\mu(\lambda = 0)$ i zadani smjer $k^\mu(\lambda = 0)$ izračunati cijelu trajektoriju $x^\mu(\lambda)$.

Iz jednadžbe (2.4) vidimo da je r_{horizon} nedefiniran za $a > r_s/2$. Vjeruje se stoga da je brzina rotacije ograničena s $a = r_s/2$. U našem radu ćemo promatrati $a = 0.998 r_s/2$ i $a = 0.5 r_s/2$.

2.2 Paralelni transport, geodezici i gibanje svjetlosti

U ravnom prostoru, čestica na koju ne djeluje nikakva sila giba se pravocrtno, ona zadržava svoj smjer kretanja duž svoje trajektorije. U zakrivljenom prostoru nije odmah jasno što znači gibati se ravno, kao što nije ni u ravnom prostoru koristimo li netrivijalni koordinatni sustav (primjerice, sferni sustav).

Ono što možemo dobro definirati jest kako se određeni vektor a^μ mijenja ako ga mičemo duž zadane krivulje $x^\mu(\lambda)$, gdje je λ parametar krivulje. Ta se operacija zove *paralelni transport*. Označimo s $k^\mu = \mathrm{d}x^\mu/\mathrm{d}\lambda$. Tada se paralelni transport vektora a^μ duž krivulje $x^\mu(\lambda)$ opisuje formulom [14]:

$$\frac{\mathrm{d}a^\rho}{\mathrm{d}\lambda} = -\Gamma^\rho{}_{\mu\nu} a^\mu k^\nu, \tag{2.5}$$

gdje je $\Gamma^\rho{}_{\mu\nu}$ *Christoffelov simbol*:

$$\Gamma^\rho{}_{\mu\nu} = \frac{1}{2} g^{\rho\sigma} \left(\partial_\mu g_{\nu\sigma} + \partial_\nu g_{\mu\sigma} - \partial_\sigma g_{\mu\nu} \right). \tag{2.6}$$

Gibanje čestica u slobodnom padu (čestica na koje ne djeluju nikoje sile osim gravitacije), opisujemo *geodezicima*. Geodezik je krivulja kojoj se tangenta ne mijenja ako se paralelno transportira duž same krivulje. Drugačije rečeno, to je krivulja $x^\mu(\lambda)$ koja se dobije paralelnim transportom vlastite tangente $k^\mu = \mathrm{d}x^\mu/\mathrm{d}\lambda$:

$$\frac{\mathrm{d}k^\rho}{\mathrm{d}\lambda} = -\Gamma^\rho{}_{\mu\nu} k^\mu k^\nu. \tag{2.7}$$

Tako, definiramo li početnu poziciju x^μ i početni smjer k^μ neke čestice (npr. fotona), pomoću jednadžbe (2.7) možemo dobiti cijelu njenu trajektoriju. Slika 2.2 grafički prikazuje problem računanja trajektorije.

Osim za definiranje geodezika, paralelni transport će nam poslužiti kod računanja polarizacije svjetla prilikom crtanja akrecijskog diska. Naime, budući da je foton u slobodnom padu, njegov vektor polarizacije se "ne mijenja", nego samo prati foton duž trajektorije.

3 Ray tracing

Za zadanu metriku, jednadžbe (2.6) i (2.7) opisuju kako će se proizvoljna zraka svjetlosti gibati. Zamislimo li kameru postavljenu na određenu poziciju i usmjerenu u određenom smjeru (obično prema središtu zvijezde), simuliranjem gibanja zraka svjetlosti možemo izračunati što bi kamera snimila. Naime, za svaki *piksel* slike, zamislimo po jednu zraku svjetlosti, koju simuliramo unatrag u vremenu (točnije, $\mathrm{d}\lambda < 0$) kako bismo odredili s kojeg dijela neba, zvijezde ili akrecijskog diska ta zraka dolazi. Takva se metoda zove *ray tracing* i česta je u računalnoj grafici. Koristi se primjerice pri iscrtavaju scena koje sadrže reflektivne objekte, objekte različitih indeksa loma i slično. Iako ustvari zrake svjetlosti simuliramo unatrag u vremenu, ponekad ćemo se možda izražavati kao da idu unaprijed, što možemo jer su sve jednadžbe gibanja simetrične na inverziju vremena $t \to -t$ (u slučaju rotirajuće crne rupe potrebno je još okrenuti smjer rotacije).

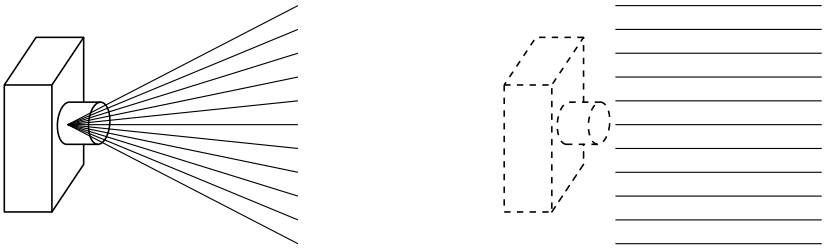
U određenim metrikama, jednadžbe gibanja se mogu riješiti analitički, u nekima se može smanjiti broj nezavisnih varijabli (primjerice, ako postoje konstante gibanja), ali mi smo odlučili napraviti univerzalni program koji ne ovisi o metrici koja je korištena. To nam dozvoljava proučavanje dodatnih efekata, konkretno nelinearne elektrodinamike, zbog koje se fotoni ne gibaju po geodezicima (barem ne onima definiranim metrikom $g_{\mu\nu}$, vidi poglavlje 5.2).

3.1 Kamera

Napisani program podržava par načina kontroliranja početnih uvjeta zraka svjetlosti. Prvo, moguće je ručno specificirati početnu poziciju i smjer fotona. To je korisno za dobivanje konkretnih brojeva, uspoređivanje rezultata s drugim radovima i crtanje raznih grafova.

Drugi način je postavljanje zamišljene kamere u prostor i crtanje slike koju bi kamera snimila (tzv. perspektivna projekcija). Za kameru je potrebno specificirati poziciju, poziciju točke u koju gleda, kut gledanja te vektor koji određuje vertikalni smjer (recimo ako želimo da je z-os vertikalna, onda će taj vektor biti jednak $\hat{z}$).

Treći način je simuliranje beskonačno daleke kamere (ortogonalna projekcija), koja je vrlo slična običnoj kameri, samo što se umjesto kuta gledanja specificira razmak između susjednih zraka svjetlosti. Primjerice, crtamo li 100×100 sliku zvijezde radijusa $10\,\mathrm{km}$, dobar razmak bi bio $0.5\,\mathrm{km}$. Slika 3.1 prikazuje razliku između perskeptivne i ortogonalne projekcije.



Slika 3.1: Prikaz perskeptivne (lijevo) i ortogonalne projekcije (desno).

3.2 Integracija geodezika

Kao što je rečeno, jednadžbe (2.6) i (2.7) nam govore kako dobiti putanju zadane zrake svjetlosti. Tu se radi konkretno o sustavu 8 diferencijalnih jednadžbi:

$$\frac{\mathrm{d}k^\rho}{\mathrm{d}\lambda} = -\Gamma^\rho{}_{\mu\nu} k^\mu k^\nu. \tag{3.1a}$$

$$\frac{\mathrm{d}x^\rho}{\mathrm{d}\lambda} = k^\rho, \tag{3.1b}$$

gdje je $\Gamma^\rho{}_{\mu\nu}$ funkcija vektora pozicije x. Slike D.1, D.2, D.3 i D.4 u dodatku prikazuju primjere putanja svjetlosti u ekvatorijalnoj ravnini za slučaj nerotirajuće i maksimalno rotirajuće crne rupe.

U slučaju da nas zanima i paralelni transport polarizacije, ne možemo samo simulirati jednadžbu (2.5), jer integraciju radimo unatrag ($\mathrm{d}\lambda < 0$) pa ne znamo unaprijed izvornu polarizaciju. Zato radimo paralelni transport baze svih vektora. Počinjemo od četiri vektora δ^μ_0, δ^μ_1, δ^μ_2 i δ^μ_3, koje kompaktno zapisujemo kao matricu $\mathcal{M}_\alpha{}^\mu$, te njih paralelno transportiramo duž geodezika. Time dobivamo 16 novih diferencijalnih jednadžbi:

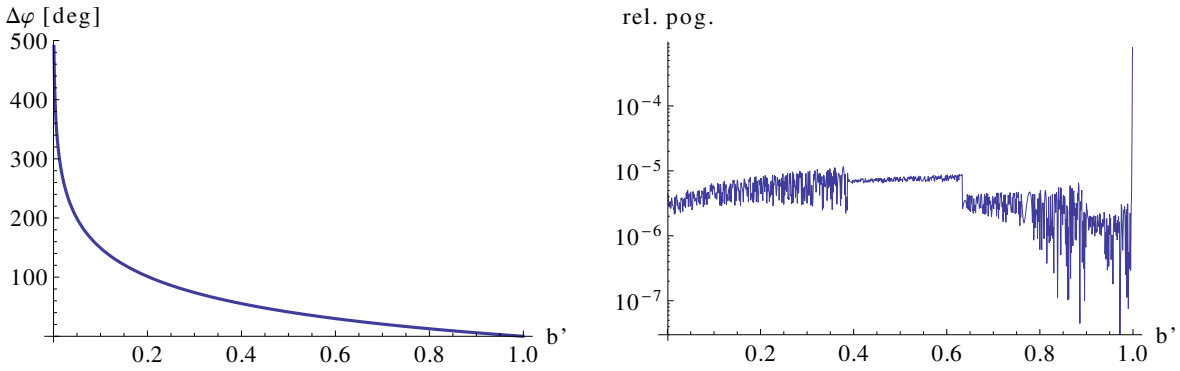
$$\frac{\mathrm{d}\mathcal{M}_\alpha{}^\rho}{\mathrm{d}\lambda} = -\Gamma^\rho{}_{\mu\nu}\mathcal{M}_\alpha{}^\mu k^\nu, \qquad \mathcal{M}_\alpha{}^\mu(\lambda = 0) = \delta^\mu_\alpha. \tag{3.2}$$

Na kraju transformiramo početni vektor polarizacije E^μ u konačni izrazom:

$$E^\mu_{\text{observer}} = E^\alpha_{\text{source}} (\mathcal{M}^{-1})_\alpha{}^\mu. \tag{3.3}$$

Sustav (3.1) (po potrebi proširen jednadžbama (3.2)) rješavamo adaptivnom numeričkom integracijom Runge-Kutta-Fehlberg 4./5. reda (vidi Dodatak A). Adaptivna integracija prilagođava korak integracije $\mathrm{d}\lambda$ s obzirom na zakrivljenost prostora i koordinatnog sustava. Na taj način daleko od crne rupe radimo korake reda veličine nekoliko km, dok u blizini korak smanjujemo na red veličine cm. U slučaju sfernog ili Boyer-Lindquist koordinatnog sustava (2.2) radimo dodatne ručne korekcije na korak $\mathrm{d}\lambda$ kada je komponenta $x^2 \equiv \theta$ u blizini $\theta = 0$ ili $\theta = \pi$, kako bi smanjili numeričke greške koje dolaze zbog izraza poput $1/\sin\theta$ i $\cot\theta$. Također, ručno smanjujemo $\mathrm{d}\lambda$ u blizini neutronske zvijezde ili diska.

Parametri integracije su $\varepsilon = 10^{-7}$, $\mathrm{d}\lambda_{\text{max}} = 10$, $\mathrm{d}\lambda_{\text{min}} = 10^{-6}$ (vidi Dodatak A), gdje je Schwarzschildov radijus fiksiran na $r_s = 0.1$, odnosno masa na $M = 0.05$ (vidi početak Dodatka C). Integracija se prekida za $r > r_\infty = 1000 r_s$.



Slika 3.2: Lijevo: ovisnost kuta otklona $\Delta\varphi$ o relativnom ulaznom parametru b' (3.4) za $s = 1$, $b' \in [0.001, 0.999]$, u stupnjevima. Desno: relativna razlika numeričkog i analitičkog rezultata. Numerički rezultat odstupa oko $0.01\% - 0.1\%$ za slučajeve kada zraka svjetlosti prolazi vrlo blizu ili vrlo daleko crnoj rupi. Po potrebi se preciznost može dodatno povećati. Račun je napravljen za $a = 0.5a_{\max} = 0.25r_s$, za $b' \in [0.002, 0.999]$. Za $b' = 0.001$ numerika dobije bitno različit rezultat, što je djelomično posljedica toga što se račun prekida čim se zraka nađe unutar $10^{-5}r_s$ udaljenosti od horizonta.

U slučaju računanja paralelnog transporta, uključenje jednadžbi (3.2) usporava račun više od 5 puta[2]. Kako bi ubrzali račun, ray tracing radimo prvo bez matrice $\mathcal{M}_\alpha{}^\mu$, i samo ako zraka svjetlosti pogodi metu od interesa (konkretno, disk, vidi poglavlje 4), ponavljamo račun uključujući matricu $\mathcal{M}_\alpha{}^\mu$.

3.3 Provjera točnosti integracije geodezika

Algoritam i kôd testiramo usporedbom rezultata s analitičkim izrazima iz [15]. Promatramo Kerrovu metriku i svjetlosne geodezike koji leže u ekvatorijalnoj ravnini ($\theta = \pi/2 = const$) i koji počinju i završavaju u beskonačnosti ($r \to \infty$). Promatramo ovisnost kuta otklona[3] $\Delta\varphi = \varphi_1 - \varphi_0 - \pi$ o ulaznom parametru b (udaljenosti originalnog pravca gibanja fotona od središta crne rupe). Označimo s b_c^s vrijednost ulaznog parametra koja razdvaja one vrijednosti b za koje geodezik upada u crnu rupu i one za koje ostaje slobodan. Sa $s = 1$ ćemo označiti slučaj kada se foton na početku giba u smjeru rotacije crne rupe (odnosno, $b > 0$), a sa $s = -1$ suprotno ($b < 0$). Radi preglednosti uvodimo mapiranje

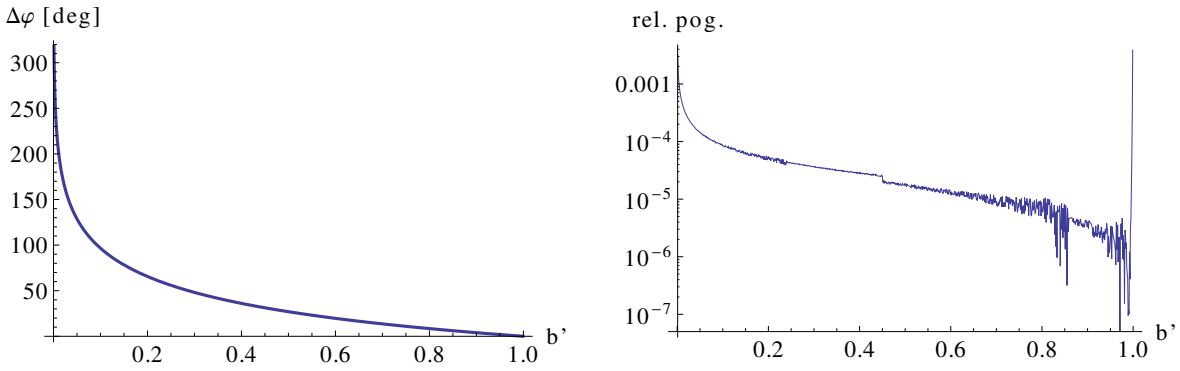
$$b' = 1 - \frac{b_c^s}{b}, \qquad b = \frac{b_c^s}{1 - b'}, \tag{3.4}$$

kojim mapiramo interval $|b| \in (|b_c^s|, \infty)$ u $b' \in (0, 1)$.

Slika 3.2 (lijevo) pokazuje izračunati kut otklona $\Delta\varphi$ u ovisnosti o modificiranom ulaznom parametru b' za $s = 1$. Desna slika prikazuje relativno odstupanje numerič-

[2]Spora operacija je računanje kontrakcije Christoffelovog simbola i odgovarajućih vektora, dakle jednadžba (3.1a) i 4 puta kompleksnija jednadžba (3.2), dok je u odnosu na njih (3.1b) vremenski zanemariva.

[3]Oduzimamo π koji je rezultat prolaska geodezika s jedne strane zvijezde na drugu. Taj član bi bio prisutan i u ravnom prostoru.



Slika 3.3: Analogno slici 3.2, za $s = -1$.

kog i analitičkog rezultata $\Delta\varphi_{\text{numerical}}/\Delta\varphi_{\text{analytical}} - 1$. U većini intervala $(0, 1)$ slaganje je odlično. Odstupanje nastaje oko $b' \sim 0$, odnosno $b \sim b_c^s$, zbog singulariteta u metrici (2.3) u blizini horizonta. Odstupanje oko $b' = 1$, to jest $b \to \infty$, nastaje jer algoritam prekida integraciju čim foton bude dovoljno daleko od centra. Krajnji kut φ se računa iz vektora brzine k^μ, a ne x^μ, jer se tako dobije točniji rezultat. Naime, kada prekinemo integraciju (jer je $r \gg r_s$), možemo pretpostaviti da se foton nastavlja gibati ravno, pa za $\lambda \to -\infty$ imamo $x^\mu(\lambda) \approx x^\mu(\lambda_{\text{final}}) + (\lambda - \lambda_{\text{final}})k^\mu \approx \lambda k^\mu$. Slika 3.3 prikazuje istu stvar, za $s = -1$.

4 Crtanje akrecijskog diska

Akrecijski diskovi su astronomski objekti sastavljeni od plina koji rotira oko centralnog gravitirajućeg objekta. Primjeri su aktivne galaktičke jezgre (AGN, *active galactic nuclei*), protoplanetarni diskovi ili diskovi koji nastanu kada crna rupa ili druga mala zvijezda uzima materijal susjedne zvijezde. Materija koja pada prema centralnom objektu, ovisno o njenom angularnom momentu ili upada na centralni objekt ili samo orbitira. U slučaju plina, zbog viskoznosti (trenja između susjednih slojeva plina), brži sloj ubrzava sporiji sloj, odnosno sporiji sloj usporava brži, zbog čega materija polako pada prema centru. Zbog tog trenja, akrecijski disk se grije te zrači. Ako je disk neproziran, zračenje će odgovarati termalnom zračenju (Planckovom zakonu), a ako je proziran zračenje će odgovarati primjerice bremsstrahlungu, sinkrotronskom i Comptonovom zračenju [16].

Kao što se može očekivati, analiza diska je poprilično kompleksna. Mora se uzeti u obzir masa centralnog objekta, brzina upada vanjske materija (u smislu, masa po jedinici vremena), sastav te materije i na kraju, vrsta centralnog objekta (crna rupa ili zvijezda). Modeli pokušavaju što detaljnije predvidjeti strukturu diska, poput gustoće materije, temperature, debljine, zatim intenziteta zračenja, spektra, oscilacija, magnetskih polja i drugog. Primjerice, sama viskoznost nije trivijalna. Naime, molekularna viskoznost je desetak redova veličine preslaba da bi opisala mjerene veličine, zbog čega se moraju proučavati efektivne viskoznosti zbog turbulencija i magnetskih polja.

U ovom radu ne ulazimo u fiziku samih diskova, već govorimo samo o prikazivanju postojećih jednostavnih modela korištenjem ray tracing metode. Baziramo se na članku [1], gdje se crta infinitezimalno tanki akrecijski disk AGN-a, konkretno tanka X-ray korona iznad diska. Za vanjski radijus diska se uzima eksperimentalni rezultat $r = 10r_s$, a za unutarnji se uzima najniža stabilna orbita r_{ISCO}. Distribucija intenziteta i polarizacije su opisani u poglavlju 4.2. Nakon toga, ukratko opisujemo Shakura-Sunyaev model diska, za koji isto pretpostavljamo da je neproziran. Ograničavamo se dakle na neprozirne diskove, jer se za njih bitno jednostavnije opisuju intenziteti zračenja.

4.1 Pitanje promatrača i izvora svjetlosti

Prije nego što opišemo strukturu i svojstva diska, navest ćemo neke detalje oko crtanja. Budući da radimo u zakrivljenom prostoru, moramo dobro definirati promatrača i izvor svjetlosti. Kod promatrača osim pozicije bitna je i brzina, budući da ona utječe na Dopplerov efekt. Osim toga, ukoliko je promatrač unutar područja gdje je metrika bitno različita od metrike Minkowskog, nije trivijalno definirati u kojem smjeru treba poslati svjetlosnu zraku za koji piksel slike[4]. Sa strane izvora svjetlosti, također je potrebno voditi računa o Dopplerovom efektu, i općenito o distribuciji intenziteta s

obzirom na izlazni kut svjetlosti (kut u odnosu na površinu diska).

Ti problemi se rješavaju uvođenjem lokalnih koordinatnih sustava, tzv. *tetrada*. Za promatrača pretpostavljamo ZAMO (*zero angular momentum observer*) sustav, koji ima fiksnu r i θ koordinatu i angularni moment jednak nuli. To ne znači da je φ koordinata fiksna, već da se promatrač vrti oko centralnog objekta jednako brzo kao sam prostor[5]. Konkretno, vektor brzine promatrača glasi $u^\mu = (1/\alpha, 0, 0, \omega/\alpha)^\mu$, gdje je $\alpha = \Sigma\Delta/A$, a $\omega = ar_s r/A$. Odnos između vektora u Boyer-Lindquist koordinatama i u ZAMO tetradi glasi [1]:

$$\begin{pmatrix} \hat{k}^t \\ \hat{k}^r \\ \hat{k}^\theta \\ \hat{k}^\varphi \end{pmatrix} = \begin{pmatrix} \alpha & 0 & 0 & 0 \\ 0 & \sqrt{\frac{\Sigma}{\Delta}} & 0 & 0 \\ 0 & 0 & \rho & 0 \\ -\omega\sqrt{\frac{A}{\Sigma}}\sin\theta & 0 & 0 & \sqrt{\frac{A}{\Sigma}}\sin\theta \end{pmatrix} \begin{pmatrix} k^t \\ k^r \\ k^\theta \\ k^\varphi \end{pmatrix}. \tag{4.1}$$

Za izvor svjetlosti, odnosno dio diska kojeg je pogodila inverzna zraka iz promatrača, aproksimiramo da se nalazi u Keplerovoj orbiti, odnosno da se giba brzinom $u^\mu_K = (\gamma, 0, 0, \gamma\Omega_K)^\mu$, gdje je

$$\Omega_K = \frac{M^{1/2}}{r^{3/2} + aM^{1/2}},$$
$$\gamma = (-g_{tt} - 2g_{t\varphi}\Omega_K - g_{\varphi\varphi}\Omega_K^2)^{-1/2}.$$

Ω_K je kutna brzina Keplerove orbite, a g_{ij} članovi Kerrove metrike (2.3). Matrica transformacije iz tetrade lokalnog promatrača u Keplerovoj orbiti u Boyer-Lindquist koordinate glasi:

$$\begin{pmatrix} k^t \\ k^r \\ k^\theta \\ k^\varphi \end{pmatrix} = \begin{pmatrix} \gamma & 0 & 0 & \tau\lambda \\ 0 & \sqrt{\frac{\Delta}{\Sigma}} & 0 & 0 \\ 0 & 0 & -\frac{1}{\sqrt{\Sigma}} & 0 \\ \gamma\Omega_K & 0 & 0 & \tau \end{pmatrix} \begin{pmatrix} \hat{k}^1 \\ \hat{k}^2 \\ \hat{k}^3 \\ \hat{k}^4 \end{pmatrix}, \tag{4.2}$$

gdje je

$$\lambda = -\frac{g_{t\varphi} + g_{\varphi\varphi}\Omega_K}{g_{tt} + g_{t\varphi}\Omega_K},$$
$$\tau = (g_{tt}\lambda^2 + 2g_{t\varphi}\lambda + g_{\varphi\varphi})^{-1/2}.$$

Tetrada (4.2) nam omogućuje da se stavimo u sustav koji se giba zajedno s pogođenim dijelom diska. Na taj način možemo dobro definirati kut upada svjetlosti, distribucije intenziteta i polarizacije s obzirom na kut itd.

[4]Što se tiče početnog smjera zraka svjetlosti, u kodu ipak pretpostavljamo da je promatrač fiksiran i dovoljno daleko od centra crne rupe. No, i dalje vodimo posebnog računa o komponenti k^t, kako bismo osigurali $k^2 = 0$.

[5]Tu, dakle, govorimo o Kerrovoj metrici. U slučaju Schwarzschildove metrike i φ koordinata je fiksna. U slučaju neke drugačije metrike bilo bi potrebno definirati drugu tetradu.

4.2 Model diska

Za disk pretpostavljamo da je beskonačno tanak, neproziran i da se prostire od najbliže stabilne kružne orbite r_{ISCO} (*innermost stable circular orbit*), do nekog radijusa r_1 po izboru. Za intenzitet svjetla pretpostavljamo takozvani *power-law*, kakav prate emisije X-zraka kvazara:

$$I_\nu(\nu, \mu, r) \propto \frac{1}{r^n} \frac{w(\mu)}{\nu^{\Gamma-1}}, \tag{4.3}$$

gdje je μ kosinus kuta između zrake svjetlosti i normale na disk (u sustavu koji se giba zajedno s tim dijelom diska), $w(\mu)$ funkcija kutne ovisnosti intenziteta, ν frekvencija, Γ takozvani *indeks fotona*, a n veličina koja definira radijalnu ovisnost intenziteta. Uzimamo vrijednosti $\Gamma = 2$ i $n = 3$, iako se to naravno može mijenjati po potrebi.

Polarizaciju radimo na sljedeći način - prvo izračunamo vektor polarizacije izvora $\boldsymbol{E}_{\text{source}}$ u lokalnom sustavu, koristeći činjenicu da je vektor okomit na valni vektor svjetlosti i na ravninu diska:

$$\hat{E}^\mu_{\text{source}} = \frac{1}{\sqrt{(\hat{k}^2)^2 + (\hat{k}^4)^2}} (0, -\hat{k}^4, 0, \hat{k}^2).$$

Nakon toga, preko (3.3) izračunamo vektor polarizacije E^μ_{observer}, pomoću kojeg izračunamo Stokesove parametre:

$$\begin{aligned}
\chi &= \arctan(-\hat{E}^\theta_{\text{observer}} / \hat{E}^\varphi_{\text{observer}}), \\
Q &= I_{\nu,\text{observer}}\, \delta \cos 2\chi, \\
U &= I_{\nu,\text{observer}}\, \delta \sin 2\chi,
\end{aligned}$$

gdje je χ kut polarizacije, $\delta = \delta_{\text{source}}$ stupanj polarizacije ($0 \leq \delta \leq 1$), koji je očuvan duž geodezika, a $I_{\nu,\text{observer}}$ intenzitet svjetla iz perspektive promatrača. Ovdje dakle radimo samo s linearnom polarizacijom. Napomenimo da je u [1] pogrešno napisano $2\chi = \ldots$ umjesto $\chi = \ldots$ (jednadžba (17) u članku), zbog čega nam se razlikuju grafovi ovisnosti ukupnog χ i δ o kutu gledanja θ_{camera} (vidi sliku 4.4). Na kraju, odnos intenziteta za promatrača i iz izvora glasi:

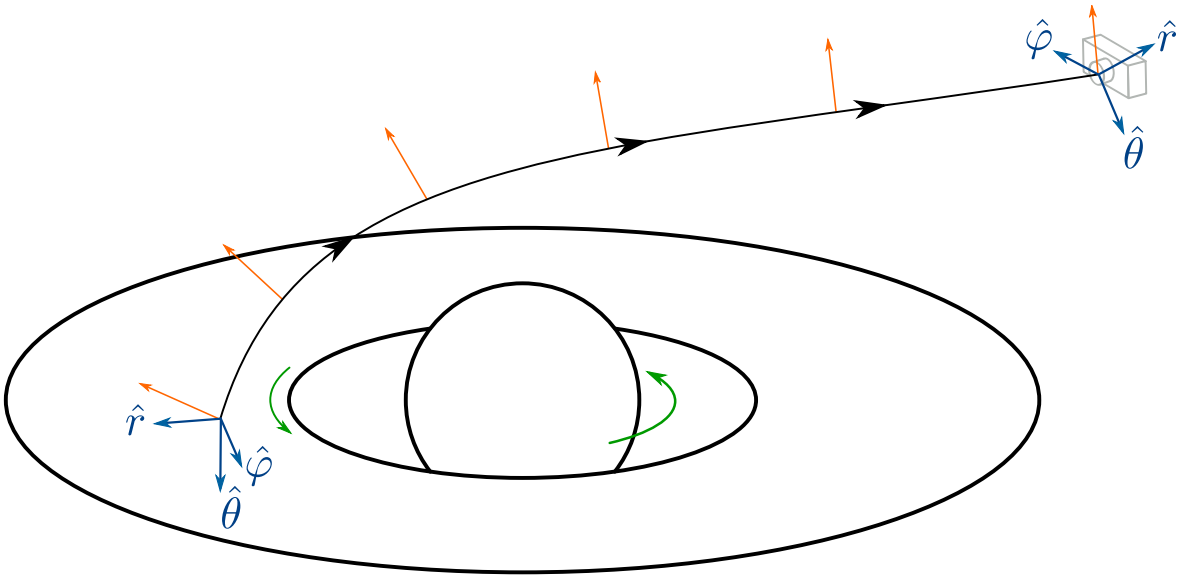
$$I_{\nu,\text{observer}}\, \mathrm{d}\nu_{\text{observer}} = g^4 I_{\nu,\text{source}}\, \mathrm{d}\nu_{\text{source}},$$

$$g = \frac{\nu_{\text{observer}}}{\nu_{\text{source}}} = \frac{(-u^\mu k_\mu)_{\text{observer}}}{(-u^\mu k_\mu)_{\text{source}}},$$

gdje je u^μ 4-brzina izvora, odnosno promatrača, a k_μ valni vektor fotona. Faktor g zovemo *redshift* faktor. Članak [1] koristi faktor g^3 umjesto g^4, jer je greškom izostavljen faktor $\mathrm{d}\nu$ koji dolazi od činjenice da je $I(\nu)$ distribucija po ν. Konačno, vrijednosti δ_{source} i $w(\mu)$ uzimamo iz [17], tablica XXIV.

4.3 Postupak za jednu zraku svjetlosti

Cijeli algoritam za crtanje jednog piksela, odnosno postupak koji treba napraviti za jednu zraku svjetlosti glasi (vidi sliku 4.1):



Slika 4.1: Skica algoritma za crtanje jednog piksela, opisanog u tekstu. Narančastom bojom je označen vektor polarizacije. Plavom bojom su označeli lokalni koordinatni sustavi promatrača i elementa diska.

(i) odrediti početnu poziciju x^μ i smjer zrake svjetlosti k^μ,

(ii) integrirati jednadžbu geodezika (2.7) unatrag u vremenu ($\mathrm{d}\lambda < 0$) kako bi se saznalo koji je izvor te zrake svjetlosti,

(iii) ako je zraka "udarila" u crnu rupu ili otišla "u beskonačnost" ($r > r_\infty$), intenzitet je 0 i račun je gotov,

(iv) inače, prebacujemo se u sustav koji se giba zajedno s pogođenim elementom diska (jednadžba (4.2)),

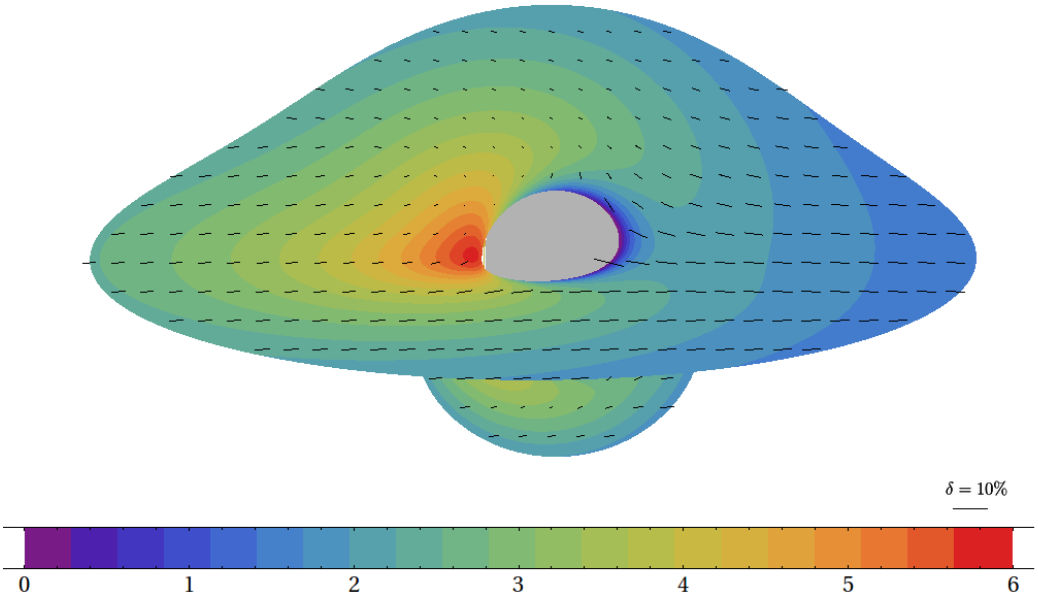
(v) računamo intenzitet svjetla, intenzitet i vektor polarizacije za zadani model,

(vi) vraćamo se u Boyer-Lindquist sustav, vektor polarizacije paralelno transportiramo natrag do kamere (ili radimo proceduru opisanu jednadžbom (3.3)),

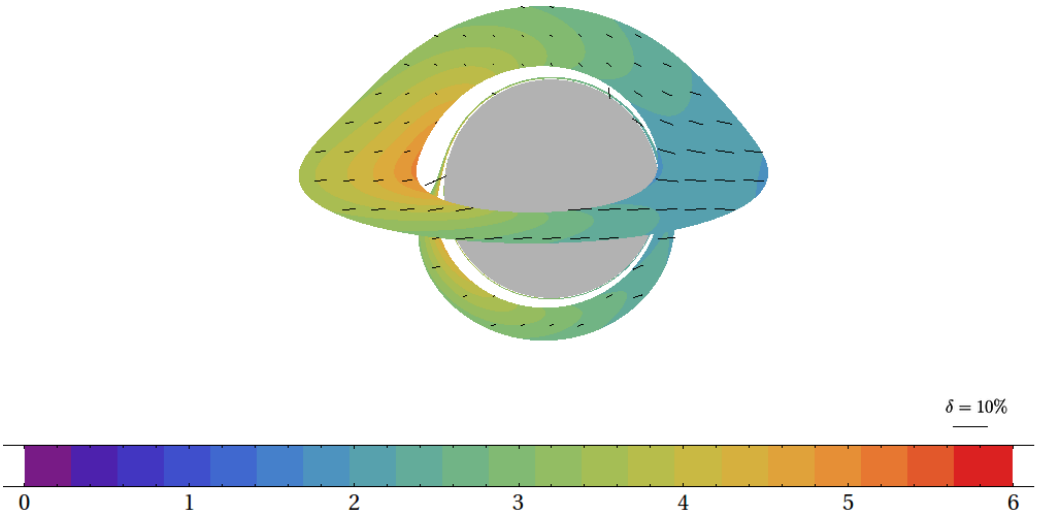
(vii) prebacujemo se u sustav promatrača (kamere) kako bismo izračunali Stokesove parametre polarizacije (jednadžba (4.1)).

4.4 Rezultati

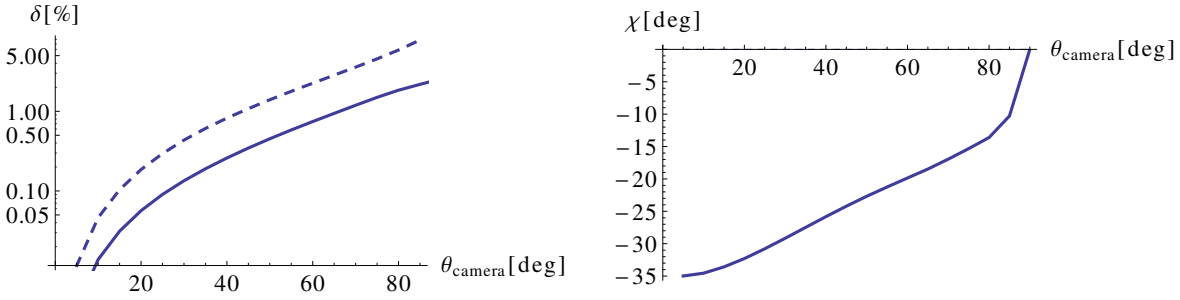
Slike 4.2 i 4.3 prikazuju disk i intenzitet radijacije za različite parametre veličine i angularnog momenta crne rupe. Vidimo da je Dopplerov efekt vrlo značajan. Naime, lijeva strana je 3 do 4 reda veličine svjetlija od desne u slučaju slike 4.2 ($a = 0.998 a_{\mathrm{max}}$), a oko 1 do 2 reda veličine u slučaju slike 4.3 ($a = 0.5 a_{\mathrm{max}}$). Budući da je brzina rotacije diska oko crne rupe više-manje neovisna o modelu, tj. uvijek približno jednaka



Slika 4.2: Logaritamski prikaz inteziteta zračenja (u slobodnim jedinicama) za model diska iz [1], za $r_{\text{ISCO}} \leq r \leq 10r_s$, $a = 0.998r_s/2$, $\theta_{\text{camera}} = 75°$. Crne linije označavaju intenzitet i smjer polarizacije (do na rotaciju π). U donjem desnom kutu je označena skala za $\delta = 10\%$. Crna rupa označena je sivom bojom. Zbog Dopplerovog efekta, lijeva strana diska (koja se giba prema kameri) je 3 do 4 reda veličine svjetlija od desne strane (koja se giba od kamere). Ova slika odgovara slici 2 iz [1].



Slika 4.3: Slično slici 4.2, ova slika prikazuje disk za $r_{\text{ISCO}} \leq r \leq 5r_s$, $a = 0.5r_s/2$, $\theta_{\text{camera}} = 75°$. Odgovara slici 3 iz [1].



Slika 4.4: Ovisnost ukupnih polarizacijskih veličina δ i χ o kutu gledanja θ_{camera}, za $a = 0.998 r_s/2$ slučaj. Puna linija predstavlja rezultat s Kerrovom metrikom, a isprekidana s ravnom. U slučaju ravne metrike kut χ iščezava. Kada je kamera iznad diska (odnosno, blizu z-osi, $\theta_{\text{camera}} \approx 0$), kut χ nije dobro definiran jer je $\delta \approx 0$. Ova slika je ispravljena verzija slike 4 iz [1].

brzini Keplerove orbite, ovakva nesimetričnost intenziteta će se pojavljivati i kod drugih modela[6].

Prikazane slike se odnose samo za $\theta_{\text{camera}} = 75°$. Za jednu sliku sumacijom po svim pikselima možemo dobiti ukupnu polarizaciju. Grafovi na slici 4.4 prikazuju ovisnost δ i χ o kutu gledanja θ_{camera}.

4.5 Shakura-Sunyaev model diska

Jedan od najpoznatijih i najizučavanijih modela diska je Shakura-Sunyaev model [2] iz 1973. godine. Radi se o modelu aksijalno simetričnog stacionarnog i uglavnom neprozirnog diska. Parametri modela su masa centralnog gravitirajućeg objekta M, unutarnji radijus diska r_0. (odgovara ili radijusu zvijezde ili r_{ISCO} u slučaju crne rupe), stopa upada materije $\dot{M}$ i parametar viskoznosti α. Ako je centralni objekt rotirajuća crna rupa, disk će ovisiti i o parametru rotacije a. Relativistički izračun su napravili Novikov i Thorne [19].

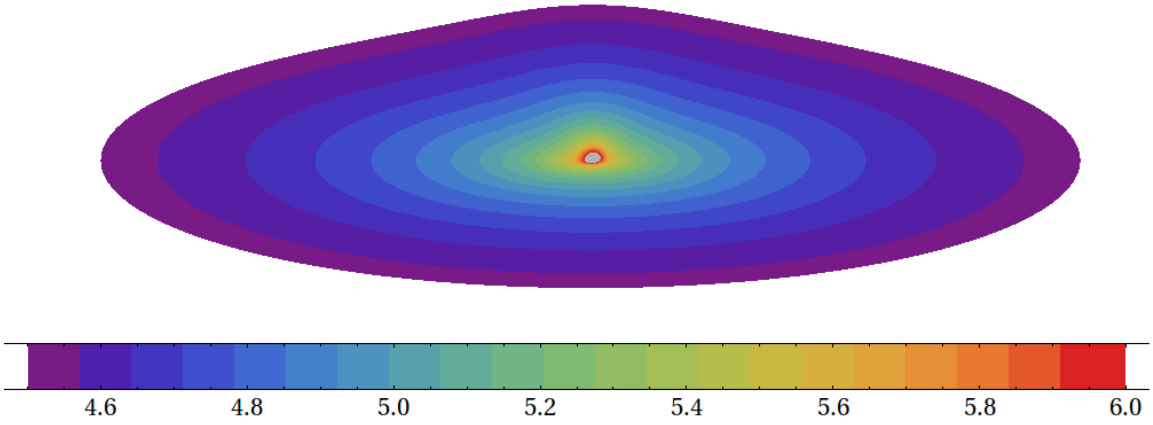
Ovdje radimo pojednostavljenu verziju modela. Dok model predviđa tri režima ponašanja diska (unutarnji, srednji i vanjski), mi ćemo pretpostaviti da se cijeli disk može opisati režimom vanjskog sloja diska. Vanjski sloj je vrlo neproziran i zračenje mu odgovara zračenju crnog tijela. Temperatura i debljina diska u ovisnosti o koordinati r zadani su sljedećim izrazima [19]:

$$T = (8 \times 10^7\,\text{K})(\alpha^{-1/5} M_*^{-1/2} \dot{M}_{0*}^{3/10}) r_*^{-3/4} (\mathscr{A}^{-2}\mathscr{B}^{-4}\mathscr{D}^{-3}\mathscr{E}^{1}\mathscr{Q}^{6})^{1/20} \tag{4.4a}$$

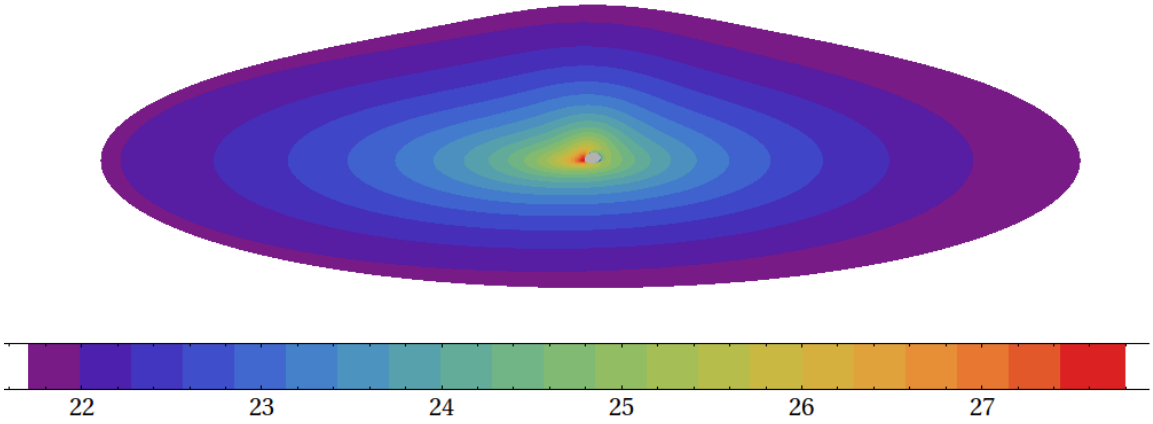
$$H = (9 \times 10^2\,\text{cm})(\alpha^{-1/10} M_*^{3/4} \dot{M}_{0*}^{3/20}) r_*^{9/8} (\mathscr{A}^{38}\mathscr{B}^{-44}\mathscr{C}^{20}\mathscr{D}^{-23}\mathscr{E}^{-19}\mathscr{Q}^{6})^{1/40} \tag{4.4b}$$

gdje je $M_* = M/(3M_\odot)$, $\dot{M}_{0*} = \dot{M}/(10^{14}\,\text{kg/s})$, $r_* = r/M$. Vrijednosti $\mathscr{A}$, $\mathscr{B}$, $\mathscr{C}$, $\mathscr{D}$, $\mathscr{E}$ i $\mathscr{Q}$ su definirane u [19] (jednadžbe (5.4.1a–j)). Zatvoreni oblik za $\mathscr{Q}$ se može naći u [20] (jednadžba (35)).

[6]Zanimljivo je da su autori filma *Interstellar* (2014) namjerno isključili Dopplerov efekt (i za intenzitet i za boju), kako slika akrecijskog diska ne bi bila previše zbunjujuća publici. Vidi raspravu u poglavlju 4.2 te sliku 15 u [18].



Slika 4.5: Logaritamska skala temperature (u kelvinima) diska opisanog Shakura-Sunyaev modelom s parametrima opisanima u tekstu.



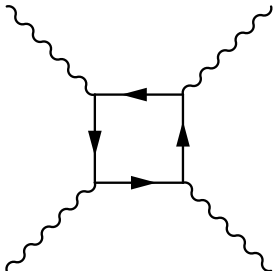
Slika 4.6: Slično slici 4.5, logaritamska skala intenziteta (u slobodnim jedinicama). Zbog distribucije temperature i Dopplerovog efekta, skoro sav intenzitet dolazi od uskog područja diska u neposrednoj blizini crne rupe.

Crno tijelo zrači intenzitetom

$$I(T) = \sigma T^4, \quad \sigma = \frac{2\pi^5}{15}\frac{k_B^4}{c^2 h^3}, \tag{4.5}$$

gdje je k_B Boltzmannova konstanta, a h Planckova konstanta. Taj intenzitet se množi s faktorom crvenog pomaka g^4, gdje je $g = \nu_{\text{observer}}/\nu_{\text{source}}$ omjer frekvencija koju vidi promatrač i koju vidi izvor.

Slika 4.5 pokazuje distribuciju temperature po disku, a slika 4.6 logaritamsku skalu intenziteta. Korišteni su parametri $M = 10M_\odot$, $\dot{M} = 10^{-14}M_\odot/\text{year}$, $\alpha = 0.1$, tako da ukupni luminozitet otprilike odgovara otprilike luminozitetu Sunca ($\sim 4 \times 10^{26}$ W). Disk počinje na $r = r_{\text{ISCO}}$ i završava na $r = 100r_s$.



Slika 5.1: Najniži red doprinosa nelinearnoj elektrodinamici, Feynmanov dijagram s četiri vanjska fotona i jednom petljom. Ovaj dijagram dovodi do efektivnog klasičnog (Euler-Heisenbergovog) Lagrangiana (5.17).

5 Nelinearna elektrodinamika

Klasično, Maxwellove jednadžbe, koje opisuju evoluciju elektromagnetskog polja, su linearne. Tako se dva vala u vakuumu propagiraju jedan neovisno o drugome, ili se pak val propagira neovisno o postojećem polju koje se nalazi u tom prostoru[7]. Kvantna teorija polja pak predviđa da elektromagnetizam nije linearan, tj. da postoji direktna interakcija između fotona i u vakuumu, kao posljedica netrivijalnih Feynmanovih dijagrama. Kako je elektromagnetsko polje u biti sastavljeno od fotona, za očekivati je da zraka svjetlosti može interagirati s fotonima polja kroz koje prolazi te da time polje može utjecati na putanju ili svojstva početne zrake.

No, interakcija između fotona je vrlo slaba, djelomično zbog relativno velike mase elektrona $m_e = 0.511\,\text{MeV}$ u odnosu na tipične energije fotona. Naime, najniža kvantna korekcija, Feynmanov *1-loop* dijagram s četiri vanjska fotona (slika 5.1), postaje značajna tek na poljima jakosti

$$B_c = m_e^2 c^2 / e\hbar = 4.4 \times 10^9\,\text{T}, \tag{5.1}$$

gdje je c brzina svjetlosti, e naboj elektrona, $\hbar$ reducirana Planckova konstanta. Veličinu B_c zovemo *kritičnim magnetskim poljem*. Kao što je rečeno u uvodu, to je polje 6 redova veličine jače od najjačeg polja stvorenog u laboratoriju. Unatoč tome, takva magnetska polja se zbilja pojavljuju u prirodi, u blizini nekih neutronskih zvijezda. Mi promatramo zvijezdu SGR 1806-20, čije površinsko magnetsko polje iznosi $B = 2 \times 10^{11}\,\text{T}$ [4]. U ovom ćemo radu promatrati koliko je značajan efekt nelinearne elektrodinamike u blizini te konkretne zvijezde. Usporedit ćemo savijanje svjetlosti koje dolazi od nelinearne elektrodinamike sa standardnim gravitacijskim savijanjem svjetlosti.

Potrebno je napomenuti da cijeli izračun koji radimo podrazumijeva da polja ne variraju puno na skali Comptonove valne duljine elektrona [9, 21] ($\lambda_c = 3.86 \times 10^{-13}\,\text{m}$) te da se ne mijenjaju puno na skali Comptonovog vremena ($\tau_c = 1.29 \times 10^{-21}\,\text{s}$). Za polja koja promatramo pretpostavljamo da su ti uvjeti zadovoljeni

[7] U materijalima elektrodinamika često nije linearna, ovdje nas zanima samo vakuum.

— polja su statična, a prostorno se mijenjaju otprilike za $6 \times 10^{-6}\,\mathrm{T}/\lambda_c$. Također, račun je valjan samo za niskoenergetske fotone $\hbar\omega \ll m_e c^2$ (u što primjerice možemo uključiti vidljivu svjetlost).

Kvantni doprinosi, odnosno netrivijalni Feynmanovi dijagrami, utječu na propagaciju svjetlosti tako što efektivno mijenjaju Lagrangian sustava. Zbog toga će svjetlost vidjeti neku efektivnu metriku $\tilde{g}_{\mu\nu}$, umjesto standardne $g_{\mu\nu}$. Naš je zadatak stoga vidjeti kako metrika ovisi o općenitom Lagrangianu te uvrstiti Lagrangian koji dolazi od kvantnih korekcija.

5.1 Efektivna metrika

Napravit ćemo izvod efektivne metrike za općeniti Lagrangian oblika

$$\mathcal{L} = \mathcal{L}(\mathcal{F}, \mathcal{G}), \tag{5.2}$$

gdje su $\mathcal{F}$ i $\mathcal{G}$ relativističke invarijante elektromagnetskog tenzora $F_{\mu\nu}$:

$$\mathcal{F} = \frac{1}{4} F_{\mu\nu} F^{\mu\nu} = \frac{1}{2}(\boldsymbol{B}^2 - \boldsymbol{E}^2), \tag{5.3}$$

$$\mathcal{G} = \frac{1}{4} F_{\mu\nu} \tilde{F}^{\mu\nu} = -\boldsymbol{E} \cdot \boldsymbol{B}, \qquad \tilde{F}^{\mu\nu} = \frac{1}{2} \varepsilon^{\mu\nu\alpha\beta} F_{\alpha\beta}. \tag{5.4}$$

Izvod baziramo na [7] za disperzijsku relaciju[8], a na kraju se pozivamo na [8, 22] da je krajnji rezultat da se fotoni zbilja gibaju u efektivnoj metrici. Izvod radimo za ravno prostorvrijeme ($g_{\mu\nu} = \eta_{\mu\nu}$). Za zakrivljeno prostorvrijeme ćemo iskoristiti *minimalno vezanje*, gdje svako pojavljivanje $\eta_{\mu\nu}$ samo zamijenimo s $g_{\mu\nu}$. Takav pristup je korišten u primjerice [6, 8]. Kasnije ćemo komentirati koliko je ta aproksimacija velika i kako bi izgledale gravitacijske korekcije.

Izvod započinjemo uvrštavanjem Lagrangiana $\mathcal{L}(\mathcal{F}, \mathcal{G})$ u Euler-Lagrangeove jednadžbe

$$\partial_\mu \frac{\partial \mathcal{L}}{\partial(\partial_\mu A_\nu)} - \frac{\partial \mathcal{L}}{\partial A_\nu} = 0,$$

gdje je A_ν elektromagnetski potencijal ($F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu$). Dobivamo jednadžbu polja

$$\partial_\mu h^{\mu\nu} = 0, \tag{5.5}$$

$$h^{\mu\nu} = \mathcal{L}_F F^{\mu\nu} + \mathcal{L}_G \tilde{F}^{\mu\nu},$$

gdje smo s $\mathcal{L}_F$ i $\mathcal{L}_G$ označili derivacije $\mathcal{L}$ po $\mathcal{F}$ i $\mathcal{G}$, redom. Polje koje mi promatramo se sastoji od dvije komponente, pozadinskog elektromagnetskog polja (radi preglednosti zadržavamo za to oznaku $F_{\mu\nu}$) te polja vala $\phi_{\mu\nu}$:

$$F_{\mu\nu} \to F_{\mu\nu} + \phi_{\mu\nu}.$$

[8]Napomena: Različiti članci koriste različite definicije za $\mathcal{F}$ i $\mathcal{G}$. Kako bismo lakše implementirali Lagrangian $\mathcal{L}$, preuzeli smo definicije iz [21], koje se razlikuju u predznaku od definicija iz [7].

Linearizacijom jednadžbe (5.5) po $\phi_{\mu\nu}$ dobivamo:

$$\frac{\partial h^{\mu\nu}}{\partial F_{\alpha\beta}}\partial_\mu \phi_{\alpha\beta}(x) = 0. \tag{5.6}$$

Za polje $\phi_{\mu\nu}$ koristimo ansatz za ravni val:

$$\phi_{\mu\nu}(x) = \varepsilon_{\mu\nu}(k)e^{-ik\cdot x}, \tag{5.7a}$$
$$\varepsilon_{\mu\nu}(k) = k_\mu \varepsilon_\nu(k) - k_\nu \varepsilon_\mu(k). \tag{5.7b}$$

Uvrštavanjem u jednadžbu (5.6) dobivamo:

$$M^{\mu\nu}\varepsilon_\nu = 0, \tag{5.8}$$

$$M^{\mu\nu} \stackrel{\text{def}}{=} \frac{\partial h^{\lambda\mu}}{\partial F_{\rho\nu}} k_\lambda k_\rho = \mathcal{L}_F(k^2 g^{\mu\nu} - k^\mu k^\nu) + \mathcal{L}_{FF} a^\mu a^\nu + \mathcal{L}_{FG}(a^\mu \tilde{a}^\nu + \tilde{a}^\mu a^\nu) + \mathcal{L}_{GG}\tilde{a}^\mu \tilde{a}^\nu,$$

$$a^\mu \stackrel{\text{def}}{=} F^{\mu\nu}k_\nu, \quad \tilde{a}^\mu \stackrel{\text{def}}{=} \tilde{F}^{\mu\nu}k_\nu,$$

gdje smo derivacije po $F_{\rho\nu}$ pretvorili u derivacije po $\mathcal{F}$ i $\mathcal{G}$ koristeći

$$\frac{\partial \mathcal{F}}{\partial F_{\mu\nu}} = F^{\mu\nu}, \quad \frac{\partial \mathcal{G}}{\partial F_{\mu\nu}} = \tilde{F}^{\mu\nu}.$$

Rješenja za $\varepsilon_\mu(k)$ tražimo uvrštavanjem općenite kombinacije četiri linearno nezavisna vektora. Primjerice, jedan takav odabir su vektori $a_\mu, \tilde{a}_\mu, k_\mu$ i b_μ:

$$b_\mu \stackrel{\text{def}}{=} F_{\mu\nu}a^\nu,$$
$$\varepsilon_\mu(k) = \alpha a_\mu + \beta \tilde{a}_\mu + \gamma k_\mu + \delta b_\mu.$$

Uvrštavanjem u jednadžbu (5.8) dobije se:

$$\begin{aligned} M^{\mu\nu}\varepsilon_\nu =& \alpha\left(\mathcal{L}_F k^2 a^\nu + \mathcal{L}_{FF} a^2 a^\nu + \mathcal{L}_{FG} a^2 \tilde{a}^\nu + \mathcal{L}_{FG}(a\cdot\tilde{a})a^\nu + \mathcal{L}_{GG}(a\cdot\tilde{a})\tilde{a}^\nu\right) \\ &+\beta\left(\mathcal{L}_F k^2 \tilde{a}^\nu + \mathcal{L}_{FF}(a\cdot\tilde{a})a^\nu + \mathcal{L}_{FG}(a\cdot\tilde{a})\tilde{a}^\nu + \mathcal{L}_{FG}\tilde{a}^2 a^\nu + \mathcal{L}_{GG}\tilde{a}^2\tilde{a}^\nu\right) \\ &+\gamma \times 0 \\ &+\delta(k^2 b^\nu - (k\cdot b)k^\nu) = 0. \end{aligned}$$

Budući da su vektori $a^\nu, \tilde{a}^\nu, k^\nu$ i b^ν linearno nezavisni, gornja jednadžba je zadovoljena samo ako faktor uz svaki od navedenih vektora iščezava. Time dobijemo:

$$\begin{aligned} &a^\nu \to \alpha\left(\mathcal{L}_F k^2 + \mathcal{L}_{FG}\mathcal{G}k^2 + \mathcal{L}_{FF}a^2\right) + \beta\left(\mathcal{L}_{FF}\mathcal{G}k^2 - 2\mathcal{L}_{FG}\mathcal{F}k^2 + \mathcal{L}_{FG}a^2\right) = 0, \\ &\tilde{a}^\nu \to \alpha\left(\mathcal{L}_{GG}\mathcal{G}k^2 + \mathcal{L}_{FG}a^2\right) + \beta\left(\mathcal{L}_F k^2 + \mathcal{L}_{FG}\mathcal{G}k^2 - 2\mathcal{L}_{GG}\mathcal{F}k^2 + \mathcal{L}_{GG}a^2\right) = 0, \\ &k^\nu \to \delta\mathcal{L}_F(b\cdot k) = 0, \\ &b^\nu \to \delta k^2 \mathcal{L}_F = 0, \end{aligned} \tag{5.9}$$

gdje smo iskoristili

$$a\cdot\tilde{a} = \mathcal{G}k^2, \quad \tilde{a}^2 = a^2 - 2\mathcal{F}k^2, \quad a\cdot k = a\cdot b = \tilde{a}\cdot k = \tilde{a}\cdot b = 0.$$

Tražimo netrivijalna rješenja ($\varepsilon_\mu \neq 0$). Prvo, iz faktora uz k^ν možemo zaključiti da je $\delta = 0$, zato što je općenito $\mathcal{L}_F \neq 0$ i $b \cdot k \neq 0$. Nadalje, sustav ne ovisi o faktoru γ pa ćemo taj član odbaciti (član γk_μ ne doprinosi faktoru $\varepsilon_{\mu\nu}$, vidi jednadžbu (5.7b)). Konačno, sustav (5.9) će imati rješenje ako determinanta sustava prve dvije jednadžbe iščezava. Time dobijemo sljedeći izraz:

$$k^2 = a^2 \lambda_\pm, \tag{5.10}$$

gdje je

$$\lambda_\pm = \frac{\mathcal{L}_F(\mathcal{L}_{GG} + \mathcal{L}_{FF}) - 2\mathcal{F}(\mathcal{L}_{FF}\mathcal{L}_{GG} - \mathcal{L}_{FG}^2) \pm \sqrt{\delta}}{2[\mathcal{G}^2(\mathcal{L}_{FF}\mathcal{L}_{GG} - \mathcal{L}_{FG}^2) + 2\mathcal{L}_F(\mathcal{L}_{GG}\mathcal{F} - \mathcal{L}_{FG}\mathcal{G}) - \mathcal{L}_F^2]}, \tag{5.11}$$

$$\delta = \left[2\mathcal{F}(\mathcal{L}_{FF}\mathcal{L}_{GG} - \mathcal{L}_{FG}^2) + \mathcal{L}_F(\mathcal{L}_{GG} - \mathcal{L}_{FF})\right]^2 + \left[2\mathcal{G}(\mathcal{L}_{FF}\mathcal{L}_{GG} - \mathcal{L}_{FG}^2) - 2\mathcal{L}_F\mathcal{L}_{FG}\right]^2.$$

Faktori λ_- i λ_+ se odnose na različita stanja vektora ε_μ. U našem slučaju, faktor λ_+ se odnosi na polarizaciju okomitu magnetskom polju (onu čije je magnetsko polje okomito na vanjsko magnetsko polje), a λ_- na paralelnu. Disperzijska relacija (5.10) se može zapisati kao [8, 22]

$$(\eta^{\mu\nu} + \lambda_\pm F^{\mu\rho}F_\rho{}^\nu)k_\mu k_\nu = 0, \tag{5.12}$$

čime se uvodi efektivna metrika

$$\tilde{\eta}^{\mu\nu} = \eta^{\mu\nu} + \lambda_\pm F^{\mu\rho}F_\rho{}^\nu. \tag{5.13}$$

Sada koristimo minimalno vezanje, i metriku $\eta^{\mu\nu}$ poopćavamo na $g^{\mu\nu}$:

$$\tilde{g}^{\mu\nu} = g^{\mu\nu} + \lambda_\pm F^{\mu\rho}F_\rho{}^\nu. \tag{5.14}$$

Zrake svjetlosti se, dakle, propagiraju kroz prostor opisan metrikom $\tilde{g}^{\mu\nu}$ na način opisan jednadžbama (2.7) i (2.6), gdje se svako pojavljivanje metrike g zamijeni s $\tilde{g}$. Kovarijantni oblik metrike $\tilde{g}_{\mu\nu}$ se definira kao inverz matrice $\tilde{g}^{\mu\nu}$:

$$\tilde{g}^{\mu\rho}\tilde{g}_{\rho\nu} = \delta^\mu_\nu.$$

5.2 Euler-Heisenbergov Lagrangian u najnižem redu

Lagrangian koji dolazi od dijagrama na slici 5.1 poznat je kao Euler-Heisenbergov Lagrangian [23]. Prije nego što ga navedemo u potpunom obliku, pogledajmo za početak najniži red po $\mathcal{F}$ i $\mathcal{G}$ [5], kako bismo dobili osjećaj redova veličine:

$$\mathcal{L} = \mathcal{L}_0 + \Delta\mathcal{L} = -\frac{1}{\mu_0}\mathcal{F} + \eta\left(4\mathcal{F}^2 + 7\mathcal{G}^2\right), \quad \eta = \frac{2\alpha^2\hbar^3}{45m_e^4c^5\mu_0^2} \tag{5.15}$$

gdje je μ_0 permeabilnost vakuuma, α konstanta fine strukture, m_e masa elektrona. Jednadžba (5.15) je valjana samo za $B \ll B_c$. Uvrstimo li taj Lagrangian u jedn. (5.11) te uzmemo samo najniži red po $\mathcal{F}$ i $\mathcal{G}$ dobit ćemo

$$\begin{aligned} \lambda_+ &\approx 1.85 \times 10^{-23}\,\mathrm{T}^{-2}, \\ \lambda_- &\approx 1.06 \times 10^{-23}\,\mathrm{T}^{-2}, \end{aligned} \tag{5.16}$$

gdje se oba faktora sporo mijenjaju s obzirom na polje za $B \ll B_c$. Uvrstimo li sada $B = B_c$ u jednadžbu (5.14), faktor $\lambda_\pm F^{\mu\rho}F_\rho{}^\nu$ će iznositi tek oko 3×10^{-4}, a to je već preveliko polje da bi Lagrangian (5.15) bio valjan. Dakle, treba nam polje $B \gg B_c$ da bi efekt bio značajan. Zbog toga će nam biti potreban potpuni Euler-Heisenbergov Lagrangian, bez razvoja po $\mathcal{F}$ i $\mathcal{G}$. Napomenimo da razni autori analiziraju efekte koristeći samo slaba polja i Lagrangian (5.15) [5–8], ali nas je zanimalo je li otklon svjetlosti zbog nelinearne elektrodinamike usporediv s gravitacijskim otklonom, za što smo uzeli najveće polje otkriveno u prirodi, konkretno $B = 2 \times 10^{11}$ T na površini zvijezde SGR 1806-20.

5.3 Potpuni Euler-Heisenbergov Lagrangian

U prethodnom smo poglavlju naveli doprinos nelinearne elektrodinamike u prvom redu po $\mathcal{F}$ i $\mathcal{G}$ (jednadžba (5.15)), valjan za $B \ll B_c$, gdje je B_c kritično magnetsko polje (5.1). Kako se to ispostavilo nedovoljno da bi efekt bio zanimljiv, proučit ćemo potpuni Lagrangian kako bismo mogli proučavati polja $B \gtrsim B_c$. Zapisan u integralnom obliku, on glasi ($c = \hbar = \mu_0 = 1$) [21]:

$$\Delta\mathcal{L} = -\frac{\alpha}{2\pi} \lim_{\varepsilon,\eta\to 0^+} \int_{i\eta}^{\infty+i\eta} \frac{\mathrm{d}s}{s} e^{-(m^2-i\varepsilon)s} \left[ab\coth(eas)\cot(ebs) - \frac{a^2-b^2}{3} - \frac{1}{(es)^2} \right], \tag{5.17}$$

gdje su a i b zadani preko:

$$\begin{aligned} a &= \sqrt{\sqrt{\mathcal{F}^2+\mathcal{G}^2}+\mathcal{F}}, & \mathcal{F} &= \frac{1}{2}(\boldsymbol{B}^2-\boldsymbol{E}^2) = \frac{1}{2}(a^2-b^2), \\ b &= \sqrt{\sqrt{\mathcal{F}^2+\mathcal{G}^2}-\mathcal{F}} = \frac{\mathcal{G}}{a}, & \mathcal{G} &= -\boldsymbol{E}\cdot\boldsymbol{B} = \pm ab. \end{aligned}$$

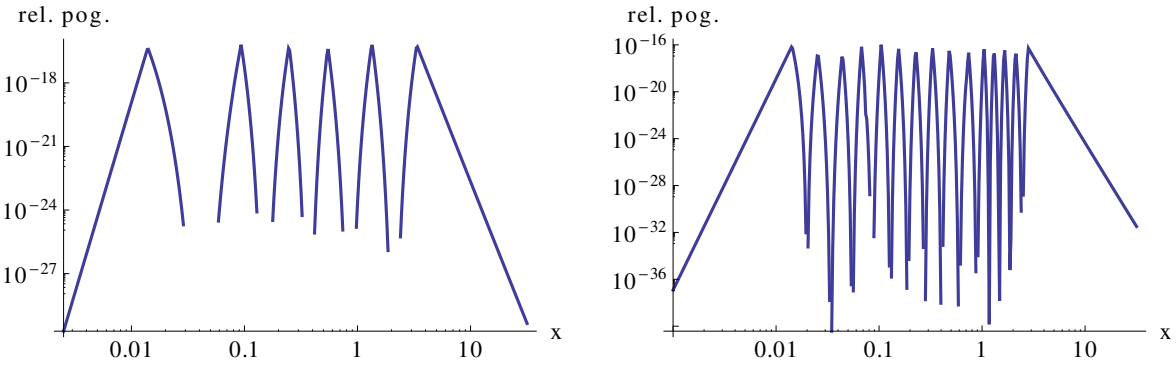
Navedeni integral je komplicirano numerički integrirati zbog raznih divergencija koje se pojavljuju. Koristimo, stoga, sljedeći razvoj [21]:

$$\mathrm{Re}\,\Delta\mathcal{L} = -\frac{\alpha}{\pi^2} ab \sum_{n=1}^{\infty} [a_n + d_n], \tag{5.18a}$$

$$\mathrm{Im}\,\Delta\mathcal{L} = \frac{\alpha}{2\pi} ab \sum_{n=1}^{\infty} \frac{1}{n} \coth\left(\frac{n\pi a}{b}\right) \exp\left(-\frac{nB_c}{b}\right), \tag{5.18b}$$

$$\begin{aligned} a_n &= \frac{\coth(n\pi b/a)}{n} \left\{ \mathrm{Ci}\left(\frac{nB_c}{a}\right)\cos\left(\frac{nB_c}{a}\right) + \mathrm{si}\left(\frac{nB_c}{a}\right)\sin\left(\frac{nB_c}{a}\right) \right\}, \\ d_n &= -\frac{\coth(n\pi a/b)}{2n} \left\{ \exp\left(\frac{nB_c}{b}\right)\mathrm{Ei}\left(-\frac{nB_c}{b}\right) + \exp\left(-\frac{nB_c}{b}\right)\mathrm{Ei}\left(\frac{nB_c}{b}\right) \right\}. \end{aligned} \tag{5.19}$$

Gore navedene formule se mogu numerički izvrjedniti. Problem jedino stvara spora konvergencija reda za realni dio Lagrangiana (imaginarni konvergira brzo zbog eksponencijalne funkcije). Asimptotski, izraz $a_n + d_n$ pada kao $\mathcal{O}(1/n^3)$, ili kao $\mathcal{O}(1/n^4)$ za slučaj $\boldsymbol{E} = \boldsymbol{0}$. Ako član reda pada kao $1/n^3$, ograničimo li se na prvih



Slika 5.2: Relativna greška implementirane aproksimacije izraza u vitičastoj zagradi za koeficijent a_n (lijevo) i d_n (desno) u ovisnosti o parametru $x = a/(nB_c)$, odnosno $x = b/(nB_c)$ (vidi jednadžbe (5.19)). Aproksimacija se sastoji od podjele domene na 7, odnosno 16 intervala i korištenjem Taylorovih redova ili Padé aproksimacija, po jedan za svaki interval. Za dijelove u kojima nedostaju linije, greška aproksimacije je premala da bi se pouzdano izračunala koristeći račun s ∼30 decimala. Najveća relativna greška iznosi u oba slučaja oko 10^{-16}.

N članova, greška će biti reda veličine $1/N^2$. Odnosno, želimo li izračunati rezultat na relativnu preciznost od 10^{-6}, potrebno je izračunati 10^4 članova.

Zbog toga je jasno da se $\Delta\mathcal{L}$ (i potrebne derivacije) neće moći računati u svakom koraku integracije geodezika (jednadžba (2.7)), jer bi to usporilo račun tisućama puta. Umjesto toga, $\Delta\mathcal{L}$ i derivacije smo izračunali za fiksne vrijednosti $\mathcal{F}$ i $\mathcal{G}$, dok smo rezultate za općenite vrijednosti dobili interpolacijom[9].

Cijeli red je moguće izračunati numerički u paketima poput *Wolfram Mathematice*, no takav se pristup pokazao sporim i nepraktičnim. Kako bismo dobili što precizniji izračun tih redova te izračunali $\Delta\mathcal{L}$ i derivacije u što više točaka, implementirali smo gornje formule u C++-u. Za to je prvo bilo potrebno implementirati specijalne funkcije. Zapravo, bilo je dovoljno računati cijele izraze u vitičastim zagradama u izrazima za a_n i d_n odjednom. Naime, zagrada u članu a_n je funkcija jedne varijable $x = a/(nB_c)$, odnosno $x = b/(nB_c)$ za d_n[10]. Stoga su za njih uz pomoć *Mathematice* određene aproksimativne formule u ovisnosti o x. Formule se temelje na Taylorovim redovima i Padé aproksimacijama u okolini nekoliko različitih fiksnih točaka. Ovisno o iznosu varijable x, koristi se odgovarajući razvoj. Time smo postigli relativnu preciznost od 10^{-16} za sve vrijednosti x (vidi sliku 5.2), što se zbog grešaka zaokruživanja u kôdu pogoršalo na otprilike 10^{-15}.

Vrijednosti $\Delta\mathcal{L}$ i derivacija su izračunate na preciznost od otprilike 10^{-10}, za što je bilo potrebno i do 10^6 članova reda. Pomoću njih su izračunati koeficijenti $\lambda_\pm$,

[9] [21] govori kako bi se mogla ubrzati konvergencija reda (5.18a), ali se interpolacija činila kao dovoljno rješenje i prije svega jednostavnije.

[10] Zbog divergencija oblika $1/a$ i $1/b$ bilo je ne samo dovoljno, već poželjno tretirati cijelu vitičastu zagradu u a_n ili d_n kao jednu specijalnu funkciju. Na taj način nije bilo problema oko uvrštavanja $a = 0$ ili $b = 0$, to jest $\boldsymbol{E} = \boldsymbol{0}$ ili $\boldsymbol{B} = \boldsymbol{0}$.

na otprilike istu preciznost. Koeficijenti $\lambda_\pm$ su izračunati za 10^6 vrijednosti $\mathcal{F}$, dok je za $\mathcal{G}$ uvijek uzeto $\mathcal{G} = 0$ (ograničavamo se na isključivo magnetska polja, vidi poglavlje 5.5). Na kraju je implementirana najjednostavnija, linearna interpolacija, zbog koje se relativna preciznost smanjila na $10^{-8} - 10^{-9}$ za $B \gtrsim 10^{-3} B_c$. Po potrebi se može napraviti bolja interpolacija.

Kao što je spomenuto u nekoliko navrata, za račun faktora $\lambda_\pm$, osim vrijednosti $\mathcal{L}$, potrebno je izračunati i derivacije $\mathcal{L}_F, \mathcal{L}_G, \mathcal{L}_{FF}, \mathcal{L}_{GG}$ i $\mathcal{L}_{FG}$ (vidi jednadžbu (5.11)). Ručno deriviranje gornjih izraza bi bilo vrlo mukotrpno i sklono greškama. Zbog toga smo koristili *automatsko deriviranje* (vidi Dodatak B), koje automatski i "analitički" računa tražene derivacije. Uz te derivacije, koje su potrebne u izračunu koeficijenata $\lambda_\pm$, zbog ray tracing-a potrebno je izračunati i Christoffelove simbole za efektivnu metriku $\tilde{g}_{\mu\nu}$ (jednadžba (5.14)). Dakle, moramo znati sve derivacije $\partial_\rho \tilde{g}_{\mu\nu}$. Naravno, i tome pristupamo automatskim deriviranjem. Jedino ručno deriviranje koje radimo je mijenjanje iz prostornih derivacija faktora $\lambda_\pm$ u derivacije po $\mathcal{F}$ i $\mathcal{G}$:

$$\partial_\mu \lambda_\pm = \frac{\partial \lambda_\pm}{\partial \mathcal{F}} \partial_\mu \mathcal{F} + \frac{\partial \lambda_\pm}{\partial \mathcal{G}} \partial_\mu \mathcal{G},$$

zato što se derivacije $\lambda_\pm$ po $\mathcal{F}$ i $\mathcal{G}$ isto mogu izračunati unaprijed. Dakle, u konačnici unaprijed računamo $\lambda_\pm$ te derivacije $\partial\lambda_\pm/\partial\mathcal{F}$ i $\partial\lambda_\pm/\partial\mathcal{G}$. Derivacije računamo automatskim deriviranjem jednadžbe (5.11).

5.4 Intenzitet efekta nelinearne elektrodinamike

Grafovi na slikama 5.3 i 5.4 prikazuju ovisnost faktora $\lambda_\pm$ i izraza $\lambda_\pm B^2$ o jakosti magnetskog polja. Vidimo da paralelna polarizacija skoro pa uopće nije perturbirana nelinearnim efektom, dok ortogonalna počinje biti utjecajna na poljima od $B = 10^{11}\,\mathrm{T}$, koja mi i proučavamo (magnetsko polje zvijezde SGR 1806-20 je $B_0 = 2 \times 10^{11}\,\mathrm{T} \approx 45 B_c$).

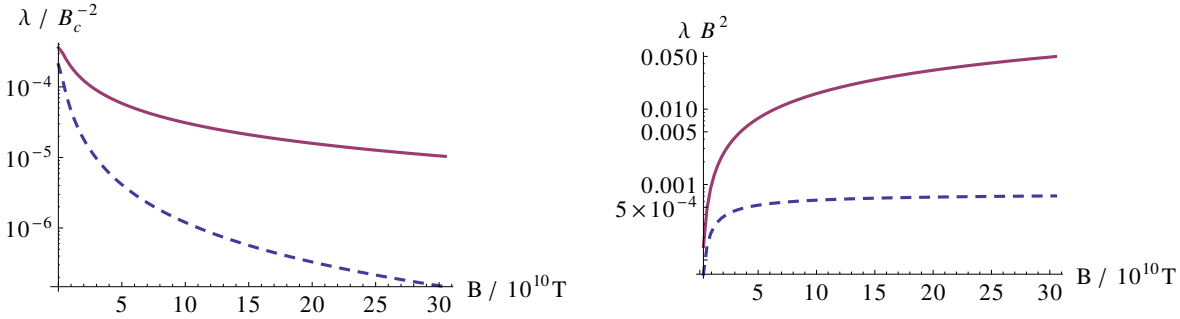
Usporedimo li Schwarzschildovu metriku (2.1) s efektivnom metrikom (5.14) za $B = B_0$, za koje je $\lambda B^2 \approx 0.03$, vidimo da efekt nelinearne elektrodinamike neće biti velik. Možemo očekivati dodatno savijanje putanja svjetlosti za svega par stupnjeva.

5.5 Magnetsko polje

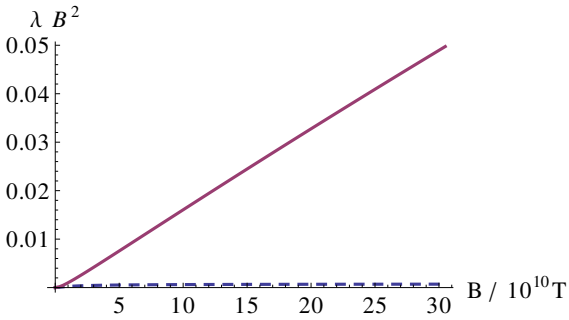
U radu se zbog jednostavnosti ograničavamo na isključivo magnetska polja:

$$\mathcal{G} = 0, \qquad b = 0.$$

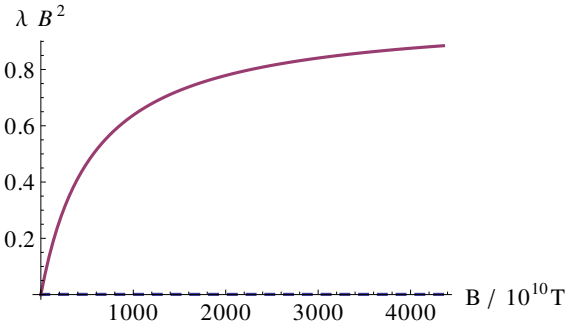
Električno polje (ako ima komponentu paralelnu magnetskom polju) bi uvelo dodatne efekte poput stvaranja elektron-pozitron parova i raspada fotona na više manjeenergetskih fotona [24]. Kada bi se išli uključiti u ray tracing, ugrubo bi se mogli opisati smanjivanjem intenziteta fotona, u smislu da za $b \neq 0$ foton ima konačno vrijeme života.



Slika 5.3: Ovisnost faktora λ_+ (puna crta) i λ_- (isprekidana) o jakosti magnetskog polja B u jedinicama B_c. Lijevi graf pokazuje $\lambda_\pm$ u ovisnosti o polju B, a desni $\lambda_\pm \times B^2$, što predstavlja intenzitet nelinearnog efekta (vidi jednadžbu (5.14)). Vidimo da je polarizacija vezana uz λ_- zanemarivo perturbirana.



Slika 5.4: Ovisnost $\lambda_\pm \times B^2$ o magnetskom polju B, kao i desni graf na slici 5.3, ali bez logaritamske skale. Vidimo da utjecaj na jednu polarizaciju uz λ_+ raste otprilike linearno s B, dok je za drugu cijeli efekt zanemariv.



Slika 5.5: Vrijednost $\lambda_\pm B^2$ za vrlo velike iznose magnetskog polja. Vidi raspravu u poglavlju 5.7. Napomena: Za $B \gg 100 B_c$, vjerojatno je potrebno uračunati i više dijagrame.

Potrebno je napomenuti da se formula (5.18a) vjerojatno ne smije pojednostaviti unaprijed uvrštavanjem $b = 0$, jer bi to onemogućilo automatsko deriviranje po $\mathcal{G}$. Zapravo se u kodu uvijek stavlja $\mathcal{G} = (10^{-10}B_c)^2$, kako bi se izbjegli izrazi poput $0/0$ (vidi napomenu na kraju Dodatka B). Za imaginarni dio Lagrangiana (5.18b) i derivacije od značaja[11] može se pokazati da svi iščezavaju čim je $b = 0$.

Magnetsko polje neutronske zvijezde modeliramo kao magnetski dipol. U ravnom prostoru ono glasi:

$$\boldsymbol{B}(\boldsymbol{r}) = \frac{\mu_0}{4\pi}\left(\frac{3\boldsymbol{r}(\boldsymbol{m}\cdot\boldsymbol{r})}{r^5} - \frac{\boldsymbol{m}}{r^3}\right), \tag{5.20}$$

gdje je $\boldsymbol{m}$ magnetski moment. Za Schwarzschildovu metriku promatramo magnetski dipol kojemu je magnetski moment u smjeru z-osi [25]:

$$F_{r\varphi} = \frac{\mu\sin^2\theta\,(f(r) - rf'(r))}{r^2}, \quad F_{\theta\varphi} = -\frac{\mu f(r)\sin 2\theta}{r}, \tag{5.21}$$

$$f(r) = \frac{3r^3}{r_s^3}\left[\log\left(1 - \frac{r_s}{r}\right) + \frac{r_s}{r}\left(1 + \frac{r_s}{2r}\right)\right].$$

Iznos vektora $\boldsymbol{m}$, odnosno konstantu μ određujemo tako što zahtijevamo da je najveće magnetsko polje na površini neutronske zvijezde jednako zadanom B_0. Kako magnetsko polje nije relativistička invarijanta, umjesto magnetskog polja zahtijevamo $\mathcal{F} = \mathcal{F}_0 = B_0^2/2$ (vidi jednadžbu (5.3)).

Napomenimo na kraju da napisani program nije ograničen na konkretan oblik magnetskog polja, a dobrim dijelom čak ni na to da $\mathcal{G}$ mora biti 0.

5.6 Rezultati

Računamo utjecaj nelinearne elektrodinamike za zvijezdu SGR 1806-20, s parameterima [26][12]

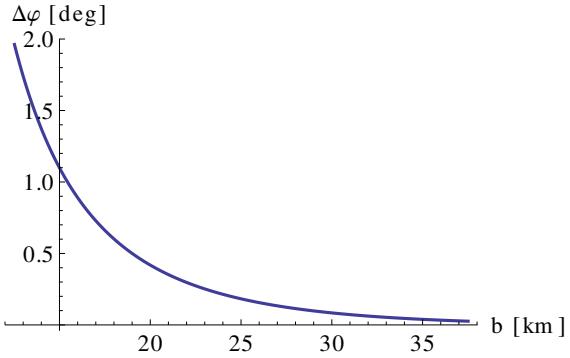
$$B_0 = 2\times 10^{11}\,\mathrm{T}, \quad M = 1.25M_\odot, \quad R = 12.4\,\mathrm{km}. \tag{5.22}$$

U analizi se radi jednostavnosti ograničavamo na ekvatorijalnu ravninu, jer bi u suprotnom dolazilo do kontinuiranog razdvajanja zrake svjetlosti na polarizaciju okomitu i paralelnu s obzirom na magnetsko polje, kao posljedica $\lambda_- \neq \lambda_+$. U ekvatorijalnoj ravnini je pak kut između polarizacije i magnetskog polje konstantan[13].

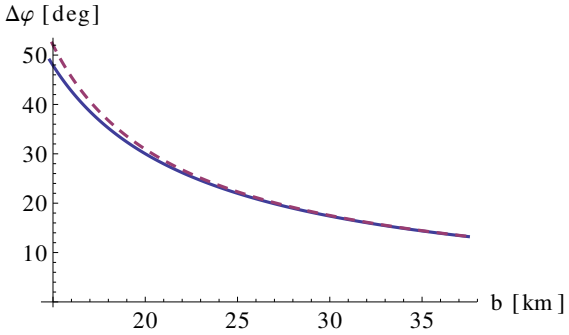
[11] Može se pokazati da niti d_n nije potrebno računati, jer su sve vrijednosti i potrebne derivacije jednake 0 za $b = 0$. Unatoč tome, Lagrangian $\Delta\mathcal{L}$ je u potpunosti implementiran u kôdu, samo što je za $\mathrm{Im}\,\Delta\mathcal{L}$ posebno potrebno naglasiti da iščezava, kako bismo bili sigurni da radimo s realnim brojevima.

[12] Navedeni članak koristi polje $B = 2.4\times 10^{11}$ T za procjenu mase i radijusa zvijezde. Mi ćemo unatoč tome koristiti $B = 2\times 10^{11}$ T.

[13] Monte Carlo metodom bismo mogli simulirati i efekt kontinuiranog razdvajanja svjetlosti, tako što krenemo od kamere od jedne zrake svjetlosti, koja se u ovisnosti o jačini magnetskog polja češće ili rjeđe razdvaja na dvije zrake različitih polarizacija. Pritom bi trebalo voditi računa o paralelnom transportu vektora polarizacije, kao što je napravljeno pri crtanju akrecijskog diska u poglavlju 4.



Slika 5.6: Ovisnost kuta otklona $\Delta\varphi$ o ulaznom parametru b za fotone u ekvatorijalnoj ravnini, uslijed efekta nelinarne elektrodinamike u ravnom prostoru.



Slika 5.7: Ovisnost kuta otklona o ulaznom parametru računajući samo efekt gravitacije (puna crta) te i efekt gravitacije i nelinearne elektrodinamike (isprekidana crta).

Graf na slici 5.7 pokazuje ovisnost kuta otklona $\Delta\varphi$ s obzirom na ulazni parametar zrake za Schwarzschildovu metriku, sa i bez magnetskog polja, uz parametre (5.22). Graf na slici 5.6 pokazuje otklon $\Delta\varphi$ koji dolazi od nelinearne elektrodinamike, ako se zvijezda promatra u ravnom prostoru.

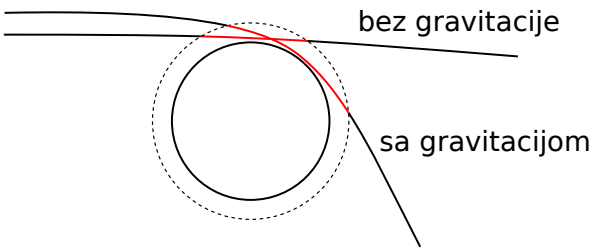
Iz grafova vidimo da najveći otklon zbog isključivo gravitacije iznosi $48.98°$, otklon zbog isključivo elektromagnetskog polja $1.95°$, a uključujući oba efekta oko $52.41° = 48.98° + 3.43°$. Iako tih par stupnjeva zvuči malo, dobro je napomenuti da Sunce savija svjetlost za otprilike $0.0005°$, što je dovoljno za mjerenje savijanja putanja svjetlosti[14]. Uočavamo da uključivanje gravitacije i zakrivljenosti prostorvremena pospješuje efekt nelinearne elektrodinamike[15].

Postoji više argumenata za tu razliku. Prvo, distribucije magnetskog polja nisu u potpunosti iste u oba slučaja. Naime, magnetsko polje je postavljeno tako da *najveće* površinsko polje bude 2×10^{11} T, što se u konačnici odnosi na polove ($\theta = 0, \pi$), dok polje na ekvatoru bude 1.0×10^{11} T u ravnom slučaju, odnosno 1.1×10^{11} T u slučaju Schwarzschildove metrike, za $r = r_N$. No, u slučaju zakrivljenog prostora, polje pada brže nego u ravnom. Tako se ona izjednače već na $r = 1.3r_N$ na $B = 4.7 \times 10^{10}$ T, dok primjerice za $r = 2r_N$ polje iznosi $B = 1.25 \times 10^{10}$ T za ravan prostor te $B = 1.13 \times 10^{10}$ T za zakrivljeni. U svakom slučaju, ta razlika u polju stvara tek 10% prednosti zakrivljenome prostoru.

Drugi razlog, spomenut i u uvodu, je činjenica da zbog gravitacijskog savijanja, svjetlost provodi više vremena u blizini neutronske zvijezde, čime ona dulje vrijeme provodi u područjima jakog magnetskog polja (vidi sliku 5.8). To je pogotovo bitno jer polje pada kao $1/r^3$ pa mali pomak od zvijezde bitno umanjuje utjecaj magnetskog polja.

[14]Naravno, direktno mjerenje savijanja putanja svjetlosti daleke neutronske zvijezde nije izvedivo.

[15]Napomena: Oduzimanje dvije linije iz grafa 5.7 radi uspoređivanja s grafom 5.6 bi dalo razliku veću od $3.43°$, ali to ne bi bio dobar pristup. Naime, vidljivi radijusi u tim slučajevima nisu isti pa se grafovi ne mogu naivno oduzimati ili uspoređivati.



Slika 5.8: Vjerojatno najznačajniji efekt zbog kojeg gravitacijsko savijanje putanja svjetlosti pospješuje savijanje od nelinearne elektrodinamike. Kada se uključi efekt gravitacije, zraka svjetlosti će provesti više vremena u području jakog magnetskog polja (označeno ugrubo crvenom linijom).

Na kraju, treba napomenuti da razlika može nastati i zbog problema definiranja veličina u zakrivljenom prostoru, poput radijusa zvijezde. Općenito bi za formalniju analizu trebalo provjeriti na koji su način određeni radijus i iznos magnetskog polja, odnosno je li u tim izvodima uračunat i efekt gravitacije.

Osim najvećeg kuta otklona, možemo promotriti i vidljivi radijus zvijezde. Pod vidljivim radijusom zvijezde mislimo na granični ulazni parametar b_0, takav da za $b < b_0$ svjetlost završava na zvijezdi, a za $b > b_0$ svjetlost odlazi u beskonačnost. Naime, zbog savijanja (konkretno, privlačenja) svjetlosti, zvijezda će izgledati veća nego što jest. Radijus zvijezde je $12.40\,\mathrm{km}$. U ravnom prostoru, elektrodinamika će promijeniti vidljivi radijus na $12.50\,\mathrm{km}$. Gravitacija sama poveća vidljivi radijus na $14.80\,\mathrm{km}$, a oba efekta zajedno na $14.93\,\mathrm{km}$. Gledano relativno, povećanje vidljivog radijusa nakon uključivanja elektrodinamičkog efekta je jednako u oba slučaja, oko 0.8%.

5.7 Ograničenja i procjena nepouzdanosti računa

Poopćenje efektivne metrike za ravnu metriku (5.13) na onu za zakrivljenu metriku (5.14) je aproksimacija i opravdana samo za slabo zakrivljene prostore. Točan račun bi morao uračunati vezanje gravitacijskih veličina poput Riccijevog skalara R, Riccijevog tenzora $R_{\mu\nu}$ i Riemannovog tenzora $R_{\mu\nu\rho\sigma}$ s elektromagnetskim poljem $F^{\mu\nu}$ i $\tilde{F}^{\mu\nu}$. No, ispostavlja se da su te korekcije izrazito male. Za najnižu korekciju, $\mathcal{O}(RF^2)$, relativna korekcija na brzinu svjetlosti u blizini crne rupe mase Sunca je tek 10^{-34}, a na kut otklona u blizini Sunca 10^{-47} [27]. Slično bi, dakle, bilo i za neutronsku zvijezdu koju mi promatramo. Korekcije su male zato što su osjetljive na zakrivljenost tek na skali Comptonove valne duljine. Više korekcije vjerojatno ne bi dale ništa značajniji rezultat, no mogle bi se izračunati iz efektivnog Euler-Heisenbergovog Lagrangiana u redu $\mathcal{O}(RF^4)$ [28] ili u općenitom redu po elektromagnetskom polju [29].

Iako smo pokazali da je gravitacijska korekcija vrlo mala, pogledajmo za svaki slučaj numeriče vrijednosti članova Schwarzschildove metrike $g^{\mu\nu}$ (2.1) i efektivne metrike $\tilde{g}^{\mu\nu}$ (5.14). Pogledat ćemo kada matrica $\tilde{g}^{\mu\nu}$ postane neinvertibilna, konkretno kada član $\tilde{g}^{rr}$ postane blizak 0, zbog čega cijela matrica nema dobro uvjetovan

inverz.

Za početak primijetimo da je radijus neutronske zvijezde 3 puta veći od njenog Schwarzschildovog radijusa ($r_N \approx 12.4\,\text{km}$, $r_s = 2GM/c^2 \approx 3.69\,\text{km}$), što znači da su članovi metrike jednaki $g_{tt} \approx 1.42$ i $g_{rr} \approx 0.70$ za $r = r_N$. Vidimo stoga da prostor u blizini površine zvijezde nije ekstremno zakrivljen, iako makroskopski značajno jest.

Promotrimo sada član efektivne metrike $\tilde{g}^{rr}$. Za slučaj magnetskog dipola u Schwarzschildovoj metrici (5.21), za $\theta = \pi/2$, $r = r_N$ dobivamo:

$$\tilde{g}^{rr} \approx 0.70 - 0.15\lambda_{\pm}B^2.$$

Po rezultatima na slikama 5.4 i 5.5, u našem slučaju smo vrlo daleko od singulariteta $g^{rr} = 0$. No, zanimljivo je vidjeti što bi bilo kada bi neutronska zvijezda bila (vjerojatno nefizikalno) gušća, primjerice 3 puta, tako da radijus bude $r_N \approx 2.3r_s$, umjesto $r_N \approx 3.4r_s$. Tada bi član $\tilde{g}^{rr}$ iznosio:

$$\tilde{g}^{rr} \approx 0.6 - 1.6\lambda_{\pm}B^2,$$

što bi bilo blizu nuli već za polje iznosa $B \approx 3 \times 10^{12}\,\text{T}$. U tom slučaju ili naše formule jednostavno postanu netočne, ili se čak događa nešto analogno crnoj rupi. U svakom slučaju, možemo pretpostaviti da je naš račun valjan u režimu unutar kojeg ga koristimo, barem na red veličine. Za veću točnost (makar radi provjere) bi bilo dobro uzeti dijagrame višeg reda [30, 31], dok su gravitacijske popravke na Euler-Heisenbergov Lagrangian [28, 29] vjerojatno u potpunosti zanemarive.

5.8 Komentar o energetskoj gustoći magnetskog polja

Magnetska polja o kojima govorimo su izrazito jaka i možemo se pitati kolika je njihova gustoća energije u usporedbi s gustoćom materije neutronske zvijezde. Za SGR 1806-20, računamo li prosječnu gustoću po $\rho = M/(4\pi R^3/3)$, dobit ćemo $\rho \approx 3 \times 10^{17}\,\text{kg/m}^3$. Gustoća mase magnetskog polja glasi $u = B^2/(2\mu)/c^2$ (tu smo iskoristili $E = mc^2$) te iznosi oko $u \approx 10^{11}\,\text{kg/m}^3$. Dakle, u našem slučaju je stvaranje gravitacijskog polja direktno zbog "težine" magnetskog polja zanemarivo. No, kako je $u \sim B^2$, sto puta jače polje bi svojom energijom savijalo prostor i mijenjalo metriku otprilike jednakog reda veličine kao i nelinearna elektrodinamika, barem što se tiče najniže kvantne korekcije, dijagrama na slici 5.1 (jačina nelinearnog efekta pri $B \gg 100B_c$ saturira u konačnu vrijednost, vidi sliku 5.5). Za usporedbu, gustoća u centru Sunca iznosi $1.6 \times 10^5\,\text{kg/m}^3$ [12].

6 Zaključak

Napravili smo numerički račun i analizu savijanja putanja svjetlosti zbog gravitacijskog savijanja prostorvremena te zbog efekata nelinearne elektrodinamike. Gravitacijsko savijanje smo iskoristili za vizualizaciju jednostavnog modela akrecijskog diska opisanog u [1] te za vizualizaciju pojednostavljene verzije Shakura-Sunyaev modela diska [2]. Napravili smo osnovni program za crtanje općenitih diskova, što uključuje račun Dopplerovog efekta i gravitacijske dilatacije, prelaženje iz Boyer-Lindquist koordinata u sustav gibanja dijela diska (keplerovska tetrada), kao i prelazak u sustav ZAMO promatrača. Uz pomoć toga, bez većih promjena mogu se implementirati i drugi modeli diska, uključujući i prozirne diskove.

U drugom dijelu rada, proučavali smo kut otklona za svjetlost koja prolazi blizu jako magnetiziranih neutronskih zvijezdi, a kao primjer smo uzeli najmagnetiziraniju otkrivenu zvijezdu SGR 1806-20, koja ima površinsko magnetsko polje od $B = 2 \times 10^{11}\,\mathrm{T}$. Za takvo je polje bilo potrebno iskoristiti puni Euler-Heisenbergov Lagrangian (jednadžbe (5.17), odnosno (5.18a)), a ne samo primjerice prvi red po $\mathcal{F}$ i $\mathcal{G}$.

Provjerili smo mijenja li se intenzitet efekta nelinearne elektrodinamika ako ga proučavamo u zakrivljenom prostoru (Schwarzschildovoj metrici), umjesto u ravnom. Dobiveno je da se maksimalni kut otklona povećava oko 70%, s otprilike 2.0° u ravnom na 3.4° u savijenom. Potrebno je napomenuti da ti brojevi jako ovise o masi i radijusu zvijezde. Ovisnost o iznosu magnetskog polja je otprilike linearna u danom režimu.

Pritom, za proučavanje nelinearne elektrodinamike u zakrivljenom prostorvremenu smo koristili minimalno vezanje, gdje se na kraju izvoda efektivne metrike ravna metrika $\eta^{\mu\nu}$ samo zamijeni sa zakrivljenom $g^{\mu\nu}$. Argumentirali smo da su gravitacijske korekcije izrazito male, otprilike 30 redova veličine manje od običnog efekta. No, budući da radimo s poljima 50-ak puta većim od kritičnog polja $B_c = 4.4 \times 10^9\,\mathrm{T}$, za precizniju analizu bilo bi poželjno provjeriti doprinos viših redova kvantnih korekcija, pogotovo ako bi se analiza nelinearne elektrodinamike koristila za određivanje svojstava neutronskih zvijezda iz mjerenih veličina.

Dodaci

Dodatak A Runge-Kutta-Fehlberg integracija 4./5. reda

Neka je zadana diferencijalna jednadžba i početni uvjet oblika

$$\frac{\mathrm{d}y}{\mathrm{d}\lambda} = f(\lambda,\, y), \quad y(\lambda_0) = y_0,$$

gdje je y nepoznata funkcija (skalar ili vektor), a $f(\lambda,\, y)$ funkcija koja opisuje kako se y mijenja u ovisnosti o parametru λ i samoj vrijednosti funkcije y. Vrijednosti λ_0 i y_0 označavaju početne vrijednosti.

Zanima nas kako integrati numerički gornju jednadžbu što preciznije i efikasnije. Najjednostavnija metoda je Eulerova metoda. Uzmimo neki mali $h > 0$ i uvedimo oznaku $\lambda_n = \lambda_0 + nh$, $y_n = y(\lambda_n)$. Eulerova metoda kaže da uz poznavanje y_n, sljedeći element y_{n+1} računamo trivijalno na način:

$$y_{n+1} = y_n + hf(\lambda_n,\, y_n).$$

Takav pristup ima točnost od $\mathcal{O}(h)$. Preciznost se može poboljšati npr. Runge-Kutta metodom 4. reda, dakle na točnost $\mathcal{O}(h^4)$:

$$\begin{aligned}
k_1 &= hf(\lambda_n,\, y_n) \\
k_2 &= hf(\lambda_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_1) \\
k_3 &= hf(\lambda_n + \frac{1}{2}h,\, y_n + \frac{1}{2}k_2) \\
k_4 &= hf(\lambda_n + h,\, y_n + k_3) \\
y_{n+1} &= y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)
\end{aligned}$$

No, u slučaju kada funkcija $f(\lambda, y)$ jako ovisi o λ ili y, gornja će metoda biti ili prespora ili preneprecizna. Metodu možemo popraviti ručnim podešavanjem parametra h ovisno o situaciji. Često je pouzdanije (i lakše) imati automatsko podešavanje.

Runge-Kutta-Fehlberg [32] je jedna takva adaptivna metoda integracije. Bazira se na tome da istovremeno računa korak i u 4. i 5. redu po h te na temelju ta dva rezultata zaključi treba li se h smanjiti radi preciznosti ili se smije povećati radi

efikasnosti. Korak integracije glasi:

$$
\begin{aligned}
k_1 &= hf(\lambda_n,\, y_n) \\
k_2 &= hf(\lambda_n + \frac{1}{4}h,\, y_n + \frac{1}{4}k_1) \\
k_3 &= hf(\lambda_n + \frac{3}{8}h,\, y_n + \frac{3}{32}k_1 + \frac{9}{32}k_2) \\
k_4 &= hf(\lambda_n + \frac{12}{13}h,\, y_n + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3) \\
k_5 &= hf(\lambda_n + h,\, y_n + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4) \\
k_6 &= hf(\lambda_n + \frac{1}{2}h,\, y_n - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5) \\
y^{(4)}_{n+1} &= y_n + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5 \\
y^{(5)}_{n+1} &= y_n + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6
\end{aligned}
$$

Mjeru pogreške R dobijemo preko $R = |y^{(4)}_{n+1} - y^{(5)}_{n+1}|/h$, koju uspoređujemo s očekivanom preciznosti ε. Ako je $R > \varepsilon$, ponavljamo računanje trenutačnog koraka s korakom veličine δh, gdje je $\delta = 0.84(\varepsilon/R)^{1/4}$. Ako je $R \leq \varepsilon$, prihvaćamo izračunati korak te, ako je $\delta > 1$, povećavamo korak na δh. Na kraju, korak h ćemo uvijek držati unutar nekih granica h_{min}, kako ne bismo imali problema s brzinom i greškama zaokruživanja, i h_{max}, kako se ne bi u jednom koraku preskočio cijeli objekt od interesa (crna rupa ili disk).

Dodatak B Automatsko deriviranje

Automatsko deriviranje je vrlo praktičan način računanja derivacija općenitih funkcija zadanih u analitičkom obliku. Za razliku od nepreciznog i ponekad nepouzdanog numeričkog deriviranja, poput aproksimacija

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} + \mathcal{O}(h^2), \quad \text{za neki } h > 0,$$

automatsko deriviranje *analitički* računa derivaciju zadane funkcije. Jedino što trebamo znati unaprijed su derivacije svih pomoćnih funkcija koje se pojavljuju, poput $\sin x, \cos x, e^x$ itd.

Metoda se temelji na tome što uvrštavanje $x + \varepsilon$ umjesto x, za male ε daje derivaciju funkcije $f(x)$:

$$f(x + \varepsilon) = f(x) + \varepsilon f'(x) + \mathcal{O}(\varepsilon^2).$$

Ako *definiramo* algebru brojeva oblika $a + b\varepsilon$, $\varepsilon^2 = 0$, tada će gornja jednadžba postati egzaktna:

$$f(x + \varepsilon) = f(x) + \varepsilon f'(x).$$

Ili općenito:

$$f(a + b\varepsilon) = f(a) + b\varepsilon f'(a).$$

Primjerice,

$$\begin{aligned}
\sin(a + b\varepsilon) &= \sin(a)\cos(b\varepsilon) + \cos(a)\sin(b\varepsilon) = \sin(a) + b\varepsilon\cos(a) \\
\cos(a + b\varepsilon) &= \ldots = \cos(a) - b\varepsilon\sin(a) \\
\exp(a + b\varepsilon) &= \exp(a) + b\varepsilon\exp(a)
\end{aligned}$$

Još nam nedostaju osnovne aritmetičke operacije:

$$\begin{aligned}
(a + b\varepsilon) + (c + d\varepsilon) &:= (a + c) + (b + d)\varepsilon \\
(a + b\varepsilon) - (c + d\varepsilon) &:= (a - c) + (b - d)\varepsilon \\
(a + b\varepsilon) \cdot (c + d\varepsilon) &:= ac + (ad + bc)\varepsilon \\
\frac{a + b\varepsilon}{c + d\varepsilon} &:= \frac{a}{c} + \frac{bc - ad}{c^2}\varepsilon
\end{aligned}$$

Pomoću takvih operacija moguće je izračunati derivaciju općenite analitičke funkcije, samo treba uvrstiti $x + 1 \cdot \varepsilon$ u funkciju i pročitati ε član rezultata.

Više derivacije se mogu dobiti na sličan način, npr. drugi red se dobije preko algebre $a + b\varepsilon + c\varepsilon^2$, $\varepsilon^3 = 0$:

$$f(x + 1 \cdot \varepsilon + 0 \cdot \varepsilon^2) = f(x) + f'(x)\varepsilon + \frac{1}{2}f''(x)\varepsilon^2.$$

Također, parcijalne derivacije se dobivaju uvođenjem različitih konstanti ε. Primjerice, za druge parcijalne derivacije potrebna je algebra nad brojevima oblika

$$a + \sum_i a_i\varepsilon_i + \sum_{i \le j} a_{ij}\varepsilon_i\varepsilon_j, \quad \varepsilon_i\varepsilon_j\varepsilon_k = 0.$$

Derivacije se dobivaju na sljedeći način:

$$\begin{aligned} f(x_1+\varepsilon_1, x_2+\varepsilon_2, \ldots, x_n+\varepsilon_n) = f(x_1, x_2, \ldots, x_n) \\ + \frac{\partial f}{\partial x_1}\varepsilon_1 + \frac{\partial f}{\partial x_2}\varepsilon_2 + \cdots + \frac{\partial f}{\partial x_n}\varepsilon_n \\ + \frac{1}{2}\left(\frac{\partial^2 f}{\partial x_1^2}\varepsilon_1^2 + \frac{\partial^2 f}{\partial x_2^2}\varepsilon_2^2 + \cdots + \frac{\partial^2 f}{\partial x_n^2}\varepsilon_n^2\right) \\ + \sum_{i<j} \frac{\partial^2 f}{\partial x_i \partial x_j}\varepsilon_i\varepsilon_j \end{aligned}$$

Što više redova derivacije računamo, teže je definirati sve potrebne operacije algrebre. Zbog načina kako radi, za očekivati je da automatsko deriviranje uspori račun otprilike onoliko puta kolika je dimenzionalnost algebre. Primjerice, za druge parcijalne derivacije po dvije varijable, očekuje se da je račun 6 puta sporiji.

Automatsko deriviranje je lagano implementirati u jezicima u kojima nije potrebno unaprijed definirati tip podatka s kojim neki dio kôda radi. Naime, implementiramo li gore navedenu algebru, umjesto običnih brojeva u željenu funkciju uvrstimo brojeve iz algebre, čime dobijemo tražene derivacije. Konkretno, algebru (za primjer druge derivacije) dobijemo uvođenjem strukture oblika

$$\langle a,\, a_1,\, \ldots,\, a_n,\, a_{11},\, \ldots,\, a_{nn}\rangle \stackrel{\text{def}}{=} a + \sum_i a_i\varepsilon_i + \sum_{i\le j} a_{ij}\varepsilon_i\varepsilon_j.$$

Primjerice, za $n = 1$:

$$\langle a, b, c\rangle \stackrel{\text{def}}{=} a + b\varepsilon + c\varepsilon^2.$$

Sve potrebne aritmetičke operacije i osnovne funkcije nad tom strukturom definiramo na gore opisan način.

Ovakav pristup je pogotovo dobar u jezicima koji podržavaju *predloške* (*template*), gdje se funkcije isto mogu definirati bez unaprijed definiranog tipa podatka, ali se posebno kompajliraju za svaki slučaj primjene. Na taj način automatsko deriviranje često daje jednako efikasan i precizan kod kao da su formule za derivacije bile ručno napisane. Naravno, ukoliko je brzina izvršavanja od velike važnosti, automatsko deriviranje otežava primjenu *low-level* optimizacija (vektorizacija, SIMD...). U takvom slučaju bi moglo biti koristi ručno implementirati i optimizirati formule za derivacije traženih funkcija, pogotovo ako nas zanima isključivo vrijednost derivacije, a ne i same funkcije.

Potrebno je napomenuti da automatsko deriviranje može dovesti do neodređenih izraza poput $0/0$, $0 \cdot \infty$ i sličnih. Primjerice, takvi bi se izrazi pojavili za $f(x) = \sin(x)/x$ za $x = 0$. U takvim se slučajevima mogu ručno napraviti provjere koje bi odozdo ograničile x prije uvrštavanja u funkciju $f(x)$.

Dodatak C Implementacijski detalji

Ovdje navodimo neke informativne implementacijske detalje koji nisu od glavne važnosti za rad. Prvo, recimo par riječi o mjernim jedinicima. U kôdu pretpostavljamo $c = \varepsilon_0 = \mu_0 = 1$ te $G = 1$. Masu crne rupe (neutronske zvijezde) fiksiramo na $M = 0.05$. Fiksiranjem mase definiramo odnos veličine između koordinate r (npr. $r_s = 0.1$) i koordinata θ, ϕ, što nam je bitno radi računanja pogreške R u koraku integracije (vidi Dodatak A). Uz definirane numeričke vrijednosti konstanti c, μ_0, G i M sve jedinice poput metra, sekunde, kilograma i tesle su jedinstveno definirane, što znači da možemo definirati potrebne fizikalne konstante: masu elektrona m_e, Planckovu konstantu $\hbar$ i kritično magnetsko polje B_c. Primijetimo da $\hbar$ *ne* fiksiramo na 1, zbog čega je bilo potrebno pažljivo implementirati formule za Euler-Heisenbergov Lagrangian, red (5.18). Kako se faktori $\lambda_\pm$ računaju unaprijed i spremaju u datoteku, njihove vrijednosti zapisujemo bezdimenzionalno u jedinicama kritičnog magnetskog polja B_c.

Kao što je u uvodu spomenuto, program je napisan tako da se lagano mogu dodavati druge metrike. Rečeno je kako je potrebno definirati samo funkcije za računanje metrike $g_{\mu\nu}$ (`get_metric_ll`) i inverza $g^{\mu\nu}$ (`get_metric_uu`) te da će se odgovarajući Christoffelovi simboli računati automatskim deriviranjem. Takav pristup je ipak djelomično prespor, jer *compiler* ne može toliko dobro optimizirati automatsko deriviranje. Naime, često većina Christoffelovih simbola iščezava, što compiler ne uspije prepoznati ili iskoristiti. Po potrebi se stoga mogu ručno implementirati funkcije za račun Christoffelovih simbola $\Gamma^\rho{}_{\mu\nu}$ (`get_christoffel_ull`) i za $\Gamma_{\rho\mu\nu}$ (`get_christoffel_lll`), što i jest napravljeno za slučaj Kerrove, Schwarzschildove i ravne metrike (varijanta $\Gamma_{\rho\mu\nu}$ se koristi radi lakšeg i efikasnijeg implementiranja jednadžbe geodezika (2.7) za efektivnu metriku (5.14)). Kako ne bismo morali ručno voditi brige o tome poziva li se automatska ili ručno napisana funkcija `get_christoffel_ull`, koristimo takozvani *Curiously recurring template pattern* [33]. Dakle, koristimo nasljeđivanje struktura / klasa kako bismo mogli dijeliti isti kôd između različitih metrika, ali obavezno u tom slučaju izbjegavamo virtualne funkcije kako ne bismo utjecali na brzinu izvršavanja programa. Na kraju, kao što je vidljivo iz gornjih imena funkcija, u nazivima varijabli ili funkcija koje predstavljaju tenzore radi preglednosti koristimo sufikse `"l"` za kovarijantne indekse i `"u"` za kontravarijantne.

U cijelom kodu koristimo dvostruku preciznost (`double`), što se pokazalo dovoljnim za naše potrebe. Jednostruka preciznost bi vjerojatno dovodila do prevelikih grešaka zaokruživanja. Prilikom računanja faktora $\lambda_\pm$, gdje se zbraja do 10^6 članova reda, dakle brojeva u padajućem poretku, imamo rizik od velikih grešaka zaokruživanja. Zbog toga je implementiran Kahanov algoritam za sumiranje [34], koji na pametan način ispravlja tu grešku. Uočena je relativna promjena u vrijednosti faktora $\lambda_\pm$ od 10^{-12}, što ipak nije toliko bitna razlika. Alternativno se mogla koristiti

četverostruka preciznost (`__float128` u GCC-u) za varijablu zbroja.

Što se tiče točnosti integracije geodezika kada je kamera postavljena blizu z-osi, numeričke greške postaju uočljive za otprilike $\theta_{\text{camera}} \lesssim 0.001°$. Taj granični kut jako ovisi o parametrima integracije $\mathrm{d}\lambda_{\text{min}}$, $\mathrm{d}\lambda_{\text{max}}$ i ε. Zanimljivo je da smanjivanje $\mathrm{d}\lambda_{\text{min}}$ ili ε često ne dovodi do točnijih slika, već suprotno, greške su obično veće.

Za prevođenje je korišten g++ compiler, s postavkama `-O3` i `-march=native`. Razina optimizacije `-O3` se pokazala u nekim slučajevima bolja, u nekim lošijima od `-O2`. Program se može za nekoliko dodatnih desetaka posto ubrzati korištenjem `-ffast-math`, što dozvoljava *unsafe* optimizacije koje mogu utjecati na konačan rezultat (zbog singulariteta u metrici u blizini crne rupe, potrebno je pomoću `-fno-finite-math-only` ipak isključiti optimizacije koje pretpostavlju da su svi brojevi u računu konačni). Takve bi optimizacije u našem slučaju bile dozvoljene svugdje osim za Kahanov algoritam, kojeg bi one vjerojatno poništile i svele na obično sumiranje. Dodatno, u kôdu koristimo određene nove mogućnosti sljedećeg standarda C++-a, poznatog pod nazivom C++1z (očekuje se da će biti objavljen 2017. g., kada bi bio preimenovan u C++17). S obzirom na konkretne mogućnosti koje koristimo, koristi li se g++ compiler, potrebno je imati verziju 6 ili veću te koristiti postavku `-std=c++1z`.

Velik dio koda je napisan koristeći predloške (`template`), u smislu da se često ne rade nikakve pretpostavke o tome koji se koordinatni sustav koristi, kojeg tipa varijable su same koordinate, zatim koja se metrika koristi itd. Takav pristup je bio nužan za dijelove koji se baziraju na automatskom deriviranju, ali dovodi do relativno dugog vremena prevođenja.

Kako bismo osigurali točnost koda, napisani su razni *unit test*-ovi. Provjeravaju se automatsko deriviranje, konverzije između sustava, specijalizirane funkcije za Christoffelove simbole, račun punog EH Lagrangiana i dr. Također, uspoređuje se akceleracija geodezika (2.7) za slučaj punog EH Lagrangiana s posebno implementiranom funkcijom koja pretpostavlja da je polje slabo i da je $\lambda_{\pm}$ fiksna veličina (jednadžba (5.16)).

Budući da je vrijeme crtanja jedne slike relativno veliko (reda veličine jedne minute za pola milijuna piksela), napravljen je *cache* sustav, tako da se svi konačni rezultati geodezika (x^μ, k^μ, po potrebi $\mathcal{M}_\alpha{}^\mu$ i dr.) spremaju u datoteku. To je prije svega korisno ako se konfigurira vizualizacija, npr. ako mijenjamo koju veličinu prikazujemo (intenzitet, polarizacija, faktor crvenog pomaka...) ili u kojim će se bojama prikazivati. Naravno, različite datoteke bi se koristile za različite postavke sustava. Za ubrzavanje procesa iscrtavanja napravljena je i adaptivna metoda opisana u sljedećem poglavlju.

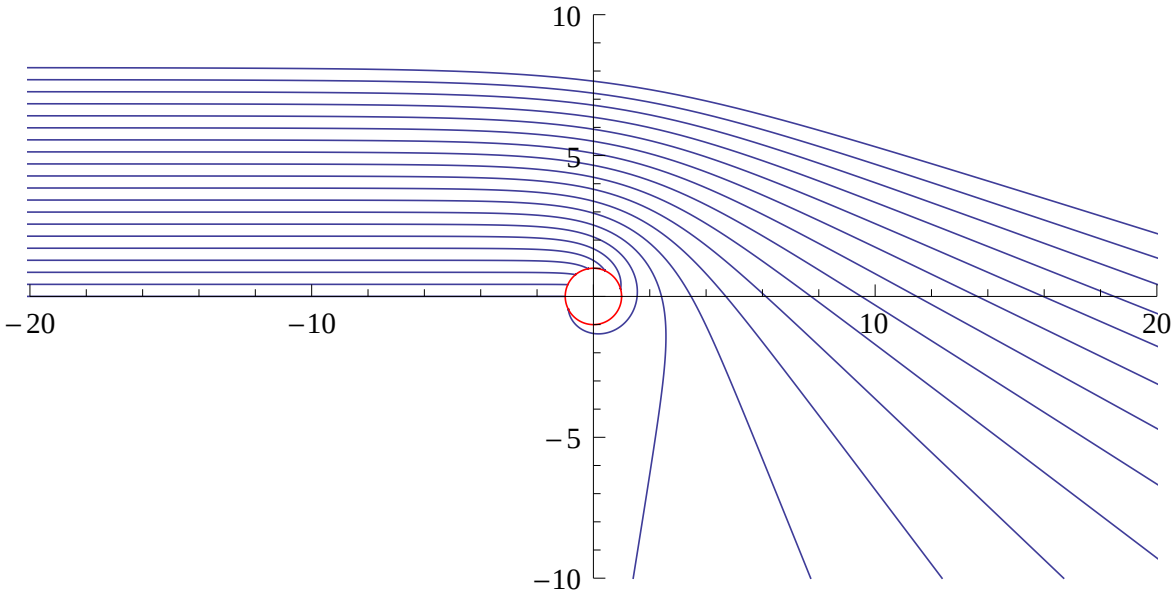
C.1 Ubrzavanje i unapređivanje crtanja adaptivnom metodom

Crtamo li sliku (primjerice 4.2) na način da za svaki piksel računamo zasebni geodezik, vrijeme crtanja će biti veliko. Naime, na našoj platformi (4 fizičke, odnosno 8 virtualnih jezgara radnog takta $2.2\,\mathrm{GHz}$), brzina računanja trajektorija bez polarizacije iznosi oko $10\,000\,\mathrm{ray/s}$ (može se bitno ubrzati smanjivanjem preciznosti računa). U slučaju polarizacije brzina padne za otprilike faktor 2, ovisno o površini diska na slici. Brzina računa trajektorije u slučaju nelinearne elektrodinamike je tek oko $1000\,\mathrm{ray/s}$. U svakom slučaju, crtanje slike visoke rezolucije može biti vremenski zahtjevno.

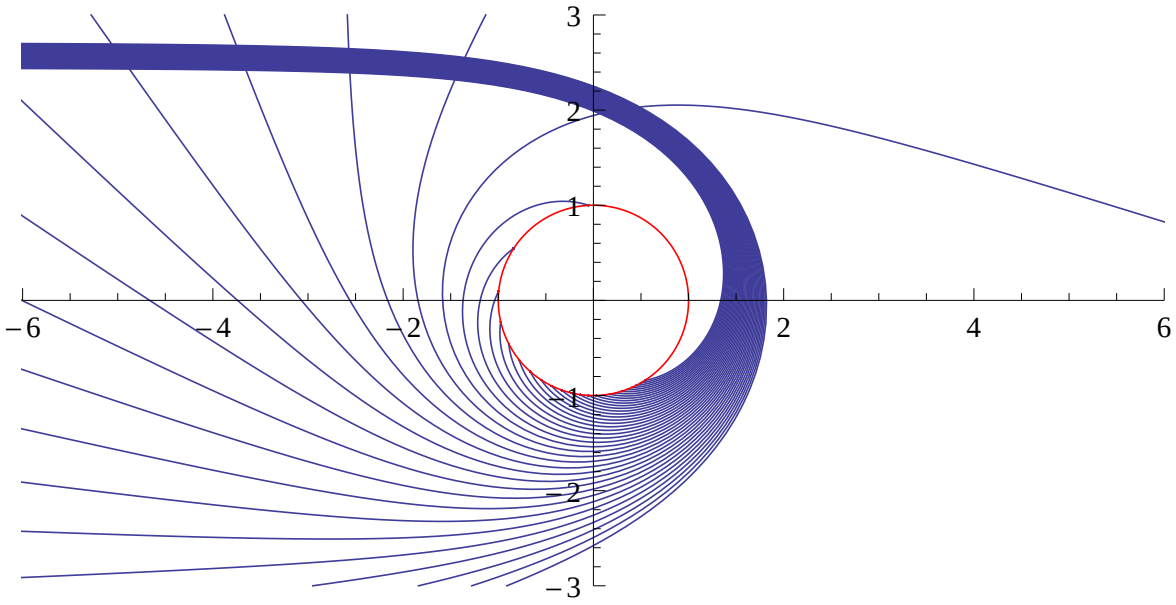
No, ubrzo uočimo da susjedni pikseli vjerojatno nemaju bitno različit rezultat. Stoga možemo napraviti adaptivnu metodu crtanja, tako da se prvo slika nacrta ugrubo, npr. svaki 64. piksel. Time se dobije mreža izračunatih piksela. Za svaki kvadratić u toj mreži provjerimo jesu li svi pikseli završili na istoj meti (crna rupa, disk ili nebo). Ako jesu, gotovi smo (u slučaju diska ćemo htjeti nešto veću rezoluciju od 64×64 blokova, tipa 4×4), a ako nisu, kvadratić dijelimo na četiri manja te računamo sve nove geodezike koje još nismo izračunali. Također, kvadratić dijelimo ako uočimo veliku udaljenost krajnjih pozicija x^μ, čime forsiramo detaljnije iscrtavanje područja s velikom zakrivljenosti (okolina crne rupe). Taj postupak podjele ponavljamo do unaprijed zadane granice. Time smo ubrzali crtanje Full HD slike (1920×1080) diska bez polarizacije s nekoliko minuta na desetak sekundi.

Osim brzine, to nam omogućuje detaljno crtanje rubova, to jest *anti-aliasing* — ako podjelu nastavimo i na kvadratiće manje od 1×1 piksela, dobit ćemo sliku s glatkim rubovima. Primjer takvog crtanja je prikazan na slikama D.5 i D.6. Gore opisano postupak podjele sadrži glavni princip rada, dok je konkretna implementacija nešto kompleksnija.

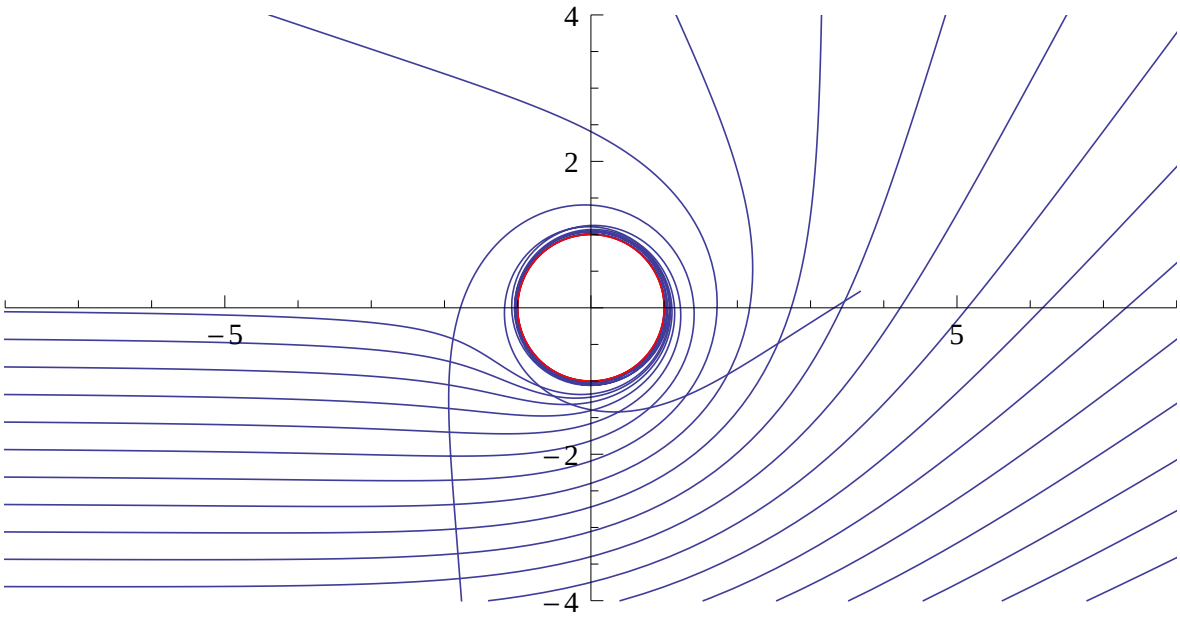
Dodatak D Dodatne slike putanja svjetlosti i diskova



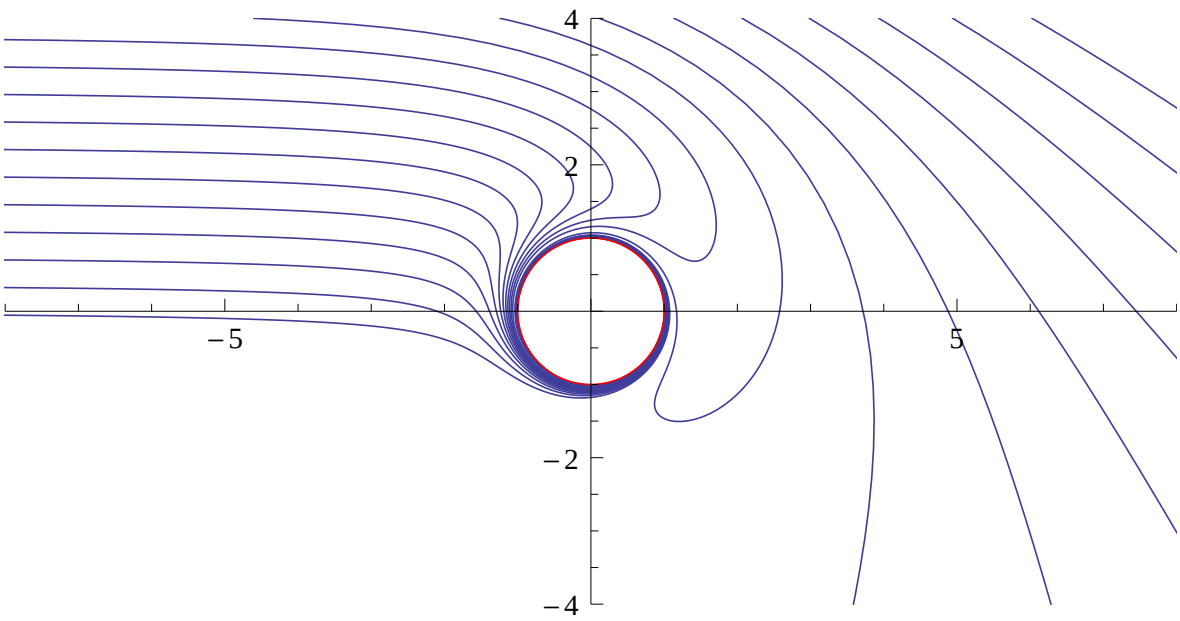
Slika D.1: Putanje svjetlosti u blizini nerotirajuće crne rupe (jednadžba (2.1)). Zrake svjetlosti dolaze slijeva iz beskonačnosti. Što je zraka svjetlosti bliža crnoj rupi, kut otklona će biti veći. Ako zraka prolazi preblizu, zabit će se u crnu rupu.



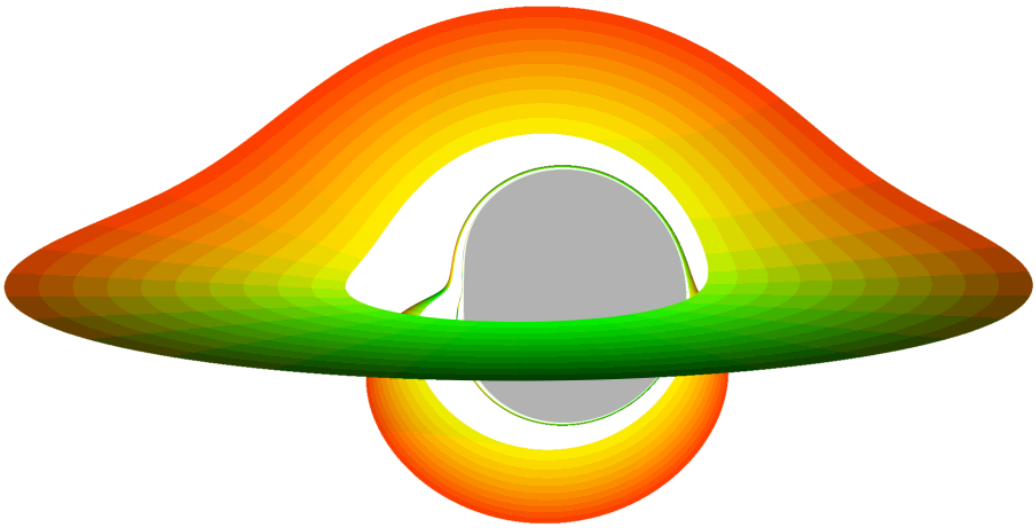
Slika D.2: Primjer okreta svjetlosti za $180°$ i $360°$ za nerotirajuću crnu rupu.



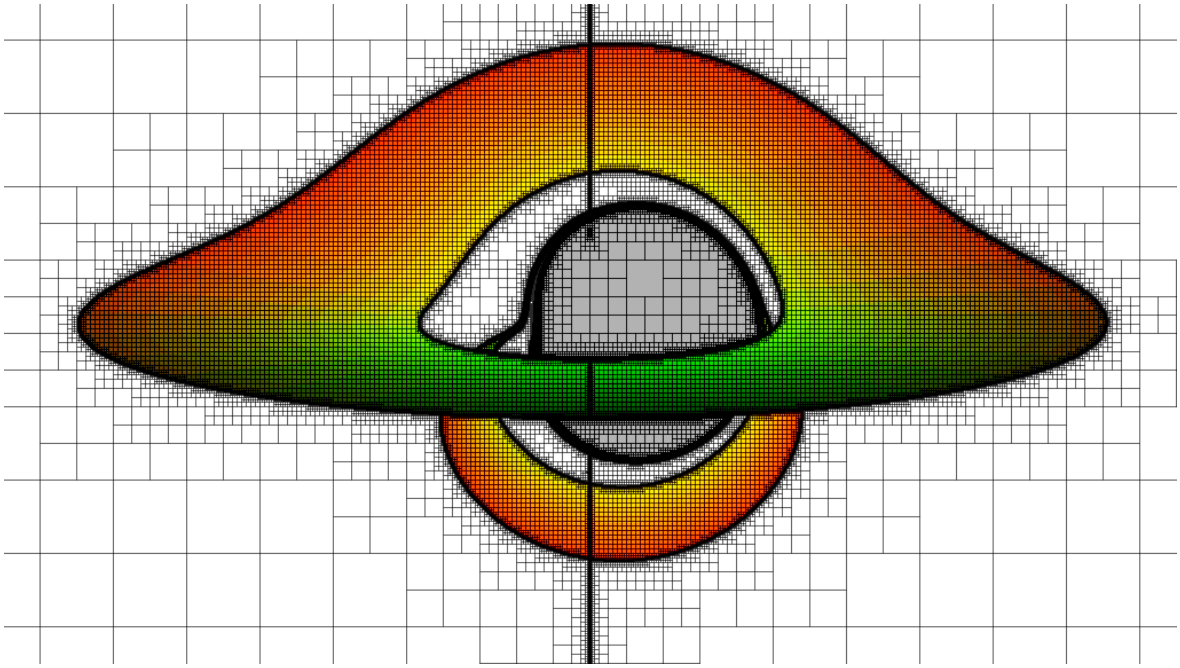
Slika D.3: Putanje svjetlosti u blizini maksimalno rotirajuće crne rupe (jednadžba (2.3)), za zrake u smjeru rotacije crne rupe. Crna rupa se rotira u suprotnom smjeru kazaklje na satu. Uočavamo kako crna rupa zavrti svjetlost u smjeru vlastite rotacije.



Slika D.4: Putanje svjetlosti za zrake u suprotnom smjeru rotacije crne rupe. Uočavamo da za određene ulazne parametre zraka svjetlost prvo biva privučena od crne rupe, a kada se dovoljno približi, crna rupa ju povuče u suprotnom smjeru.



Slika D.5: Prikaz nefizikalnog diska za $3r_s \leq r \leq 10r_s$, $a = 0.998r_s/2$. Crna rupa prikaza je sivom bojom. Boje na disku prikazuju koordinate φ i r konačne lokacije. Uočavamo više slika istog diska, što je posljedica izrazito jakog savijanja putanja svjetlosti, odnosno vrlo velikog kuta otklona za svjetlost koja prolazi blizu crne rupe. Rezolucija slike je 1024×576, ali je glatkoća rubova postignuta detaljnijim crtanjem tih dijelova, vidi sliku D.6.



Slika D.6: Prikaz *quadtree* strukture crtanja za sliku D.5 (vidi poglavlje C.1). Svaki vrh nacrtanih kvadrata predstavlja jedan geodezik. Gustoća kvadrata stoga odražava gustoću svjetlosnih zraka korištenih pri crtanju. U ovom slučaju, broj dijeljenja ispod veličine jednog piksela je ograničen na 4, što je povećalo broj geodezika s oko $38\,000$ (kada nema crtanja ispod veličine jednog piksela) na $250\,000$. Za usporedbu, slika ima ukupno $\sim 590\,000$ piksela. Primijetite da se u tih $250\,000$ geodezika velik broj njih nalazi u blizini crne rupe, zbog čega je prosječan broj koraka integracije po zraci svjetlosti relativno velik.

Dodatak E Primjer kôda za crtanje diska

Pripremili smo pojednostavljeni oblik programa, koji samo crta sliku analognu slici D.5 (bez *anti-aliasing*-a, tj. adaptivne metode). Navedeni kôd uključuje Kerrovu metriku, prelaženje iz Kartezijevih u Boyer-Linquist koordinate, RGF45 metodu integracije, projektivnu perspektivu, određivanje boja ovisno o pogođenoj meti te spremanje slike u TGA format. Originalni kôd je desetak puta veći od ovoga.

```
/* Preporučene postavke prevođenja (za g++):
 *   g++ -std=c++11 -O3 -march=native -Wall -Wextra
 *   (za starije verzije g++-a, umjesto -std=c++11 koristite -std=c++0x)
 *
 * Imena .h i .cpp označenih s ORIGINAL označavaju lokacije u potpunom kodu. */
#include <cassert>
#include <cstdio>
#include <cstdlib>
#include <cmath>
#include <tuple>

enum Target {
  TARGET_BLACK_HOLE,
  TARGET_DISK,
  TARGET_INFINITY,
  TARGET_TOO_MANY_STEPS
};

/* Za fizikalne konstante pretpostavljamo G = c = 1! */
// ORIGINAL: include/physical_constants.h, include/parameters.h
constexpr double M = 0.05;
constexpr double a = 0.998 * M;
constexpr double aa = a * a;
constexpr double rs = 2 * M;  // rs = 2 G M / c^2

constexpr double disk_inner_r = 3 * rs;
constexpr double disk_outer_r = 10 * rs;

constexpr int MAX_N = 100000;
constexpr double epsilon = 1e-6;
constexpr double dlambda_initial = rs;
constexpr double dlambda_min = rs / 100000;
constexpr double dlambda_max = rs * 100;
constexpr double max_r = 1100 * rs;

/* Pomoćne funkcije. */
template <typename T> inline T sqr(const T &x) { return x * x; }


/****************************** STRUKTURE *********************************/

struct BoyerLindquist {  // ORIGINAL: include/coordinate.h
  double t;
  double r;
  double theta;
  double phi;

```

```
48    inline double &operator[](int k) { return ((double *)this)[k]; }
49    inline double operator[](int k) const { return ((const double *)this)[k]; }
50
51    /* Aritmetičke operacije nad vektorom, potrebne za integraciju. */
52    friend inline BoyerLindquist operator+(const BoyerLindquist &A,
53                                           const BoyerLindquist &B) {
54      return {A.t + B.t, A.r + B.r, A.theta + B.theta, A.phi + B.phi};
55    }
56    friend inline BoyerLindquist operator*(double c, const BoyerLindquist &B) {
57      return {c * B.t, c * B.r, c * B.theta, c * B.phi};
58    }
59    inline BoyerLindquist operator-() const {
60      return {-t, -r, -theta, -phi};
61    }
62
63    /* Procjena razlike (greške) između dva vektora. */
64    friend inline double numerical_distance(const BoyerLindquist &A,
65                                            const BoyerLindquist &B) {
66      return std::sqrt(sqr(A.t - B.t)
67                     + sqr(A.r - B.r)
68                     + sqr(A.theta - B.theta)
69                     + sqr(A.phi - B.phi));
70    }
71
72    inline bool isfinite(void) const {
73      using std::isfinite;
74      return isfinite(t) && isfinite(r) && isfinite(theta) && isfinite(phi);
75    }
76  };
77
78  struct Vector3 {  // ORIGINAL: include/vector.h
79    double x, y, z;
80
81    friend inline Vector3 operator+(const Vector3 &A, const Vector3 &B) {
82      return {A.x + B.x, A.y + B.y, A.z + B.z};
83    }
84    friend inline Vector3 operator-(const Vector3 &A, const Vector3 &B) {
85      return {A.x - B.x, A.y - B.y, A.z - B.z};
86    }
87    friend inline Vector3 operator*(double c, const Vector3 &B) {
88      return {c * B.x, c * B.y, c * B.z};
89    }
90    inline Vector3 operator-(void) const {
91      return {-x, -y, -z};
92    }
93    inline Vector3 normalized(void) const {
94      return (1 / std::sqrt(x * x + y * y + z * z)) * (*this);
95    }
96
97    friend inline Vector3 cross(const Vector3 &A, const Vector3 &B) {
98      return {
99          A.y * B.z - A.z * B.y,
100         A.z * B.x - A.x * B.z,
101         A.x * B.y - A.y * B.x
102     };
103   }
104 };
105
106 struct Christoffel {
```

```
107   double chr[4][4][4];
108 };
109
110 struct BGRA {
111   unsigned char b, g, r, a;
112 };
113
114
115 /*************************** KERROVA METRIKA *******************************/
116
117 Christoffel get_christoffel_ull(const BoyerLindquist &position) {
118   /* Thomas Müller, Catalogue of Spacetimes, http://arxiv.org/abs/0904.4184v3 */
119   /* ORIGINAL: include/kerr.h */
120   double r = position.r;
121   double rr = sqr(r);
122   double cos_theta = std::cos(position.theta);
123   double sin_theta = std::sin(position.theta);
124   double cot_theta = cos_theta / sin_theta;
125   double sin_cos_theta = sin_theta * cos_theta;
126   double cos2_theta = sqr(cos_theta);
127   double sin2_theta = sqr(sin_theta);
128   double sigma = rr - aa * cos2_theta;
129   double rr_aa = rr + aa;
130   double one_over_Sigma = 1 / (rr + aa * cos2_theta);
131   double one_over_Sigma2 = sqr(one_over_Sigma);
132   double one_over_Sigma3 = one_over_Sigma * one_over_Sigma2;
133   double Delta = rr - rs * r + aa;
134   double one_over_Delta = 1 / Delta;
135
136   /* Notacija: t = t, r = r, h = theta, p = phi. */
137   double rtt = .5 * rs * Delta * sigma * one_over_Sigma3;
138   double ttr = .5 * rs * rr_aa * sigma * one_over_Sigma2 * one_over_Delta;
139   double tth = -rs * aa * r * sin_cos_theta * one_over_Sigma2;
140   double rtp = -.5 * rs * Delta * a * sin2_theta * sigma * one_over_Sigma3;
141   double rrr = (r * aa * sin2_theta - .5 * rs * sigma) * one_over_Sigma
142       * one_over_Delta;
143   double rrh = - aa * sin_cos_theta * one_over_Sigma;
144   double rhh = - r * Delta * one_over_Sigma;
145   double php = cot_theta * (1 + rs * aa * r * sin2_theta * one_over_Sigma2);
146   double htt = -rs * aa * r * sin_cos_theta * one_over_Sigma3;
147   double ptr = .5 * rs * a * sigma * one_over_Sigma2 * one_over_Delta;
148   double pth = -rs * a * r * cot_theta * one_over_Sigma2;
149   double htp = rs * a * r * rr_aa * sin_cos_theta * one_over_Sigma3;
150   double hrr = aa * sin_cos_theta * one_over_Sigma * one_over_Delta;
151   double hrh = r * one_over_Sigma;
152   double hhh = -aa * sin_cos_theta * one_over_Sigma;
153   double thp = rs * a * aa * r * sin2_theta * sin_cos_theta * one_over_Sigma2;
154   double trp = .5 * rs * a * sin2_theta * (aa * cos2_theta * (aa - rr)
155       - rr * (aa + 3 * rr)) * one_over_Sigma2 * one_over_Delta;
156   double prp = (r + .5 * rs * ((sqr(aa * sin_cos_theta) - rr * rr_aa)
157       * one_over_Sigma2 - rr * one_over_Sigma)) * one_over_Delta;
158   double rpp = Delta * sin2_theta * (-r
159       + .5 * rs * aa * sin2_theta * sigma * one_over_Sigma2) * one_over_Sigma;
160   double hpp = -sin_cos_theta * (sqr(rr_aa) - aa * Delta * sin2_theta
161       + rr_aa * rs * aa * r * sin2_theta * one_over_Sigma) * one_over_Sigma2;
162
163   return Christoffel{{
164     {
165       {0, ttr, tth, 0},
```

```
166       {ttr, 0, 0, trp},
167       {tth, 0, 0, 0},
168       {0, trp, thp, 0},
169     }, {
170       {rtt, 0, 0, rtp},
171       {0, rrr, rrh, 0},
172       {0, rrh, rhh, 0},
173       {rtp, 0, 0, rpp},
174     }, {
175       {htt, 0, 0, htp},
176       {0, hrr, hrh, 0},
177       {0, hrh, hhh, 0},
178       {htp, 0, 0, hpp},
179     }, {
180       {0, ptr, pth, 0},
181       {ptr, 0, 0, prp},
182       {pth, 0, 0, php},
183       {0, prp, php, 0},
184     }
185   }};
186 }
187
188
189 /*************************** JEDNADŽBA GEODEZIKA ****************************/
190
191 struct ODEState {  /* Ukupni 8D vektor koji ulazi u dif. jedn. */
192   BoyerLindquist position;
193   BoyerLindquist direction;
194
195   friend inline ODEState operator+(const ODEState &A, const ODEState &B) {
196     return {A.position + B.position, A.direction + B.direction};
197   }
198   friend inline ODEState operator*(double c, const ODEState &B) {
199     return {c * B.position, c * B.direction};
200   }
201 };
202
203 /* Izračun y_{n+1} uz zadan y_n, fiksan h i desnu stranu dif. jedn. f(y). */
204 template <typename Vector, typename RHSFunc>
205 std::pair<Vector, Vector> integration_step_RGF45(
206     const RHSFunc &RHS, const double h, const Vector &u) {
207   // ORIGINAL: include/integration.h
208   Vector k1, k2, k3, k4, k5, k6, out4, out5;
209
210   k1 = h * RHS(u);
211   k2 = h * RHS(u + (1. / 4) * k1);
212   k3 = h * RHS(u + (3. / 32) * k1 + (9. / 32) * k2);
213   k4 = h * RHS(u + (1932. / 2197) * k1
214                  + (-7200. / 2197) * k2
215                  + (7296. / 2197) * k3);
216   k5 = h * RHS(u + (439. / 216) * k1
217                  + (-8) * k2
218                  + (3680. / 513) * k3
219                  + (-845. / 4104) * k4);
220   k6 = h * RHS(u + (-8. / 27) * k1
221                  + 2 * k2
222                  + (-3544. / 2565) * k3
223                  + (1859. / 4104) * k4
224                  + (-11. / 40) * k5);
```

```
225
226   out4 = u + (25. / 216) * k1
227            + (1408. / 2565) * k3
228            + (2197. / 4104) * k4
229            + (-1. / 5) * k5;
230   out5 = u + (16. / 135) * k1
231            + (6656. / 12825) * k3
232            + (28561. / 56430) * k4
233            + (-9. / 50) * k5
234            + (2. / 55) * k6;
235
236   return std::make_pair(out4, out5);  // Rezultat 4. i 5. reda.
237 }
238
239 /* Desna strana dif. jedn. dy/dlambda = f(y). Napomena: Radi jednostavnosti,
240  * u kôdu koristimo dlambda > 0 pa smo tu okrenuli predznake za f(y). */
241 ODEState geodesic_RHS(const ODEState &state) {  // ORIGINAL: include/geodesic.h
242   Christoffel christoffel_ull = get_christoffel_ull(state.position);
243
244   ODEState result;
245   for (int k = 0; k < 4; ++k) {
246     double outer = 0;
247     for (int i = 0; i < 4; ++i) {
248       double inner = 0;
249       for (int j = 0; j < 4; ++j)
250         inner += christoffel_ull.chr[k][i][j] * state.direction[j];
251       outer += state.direction[i] * inner;
252     }
253     result.direction[k] = outer;
254   }
255   result.position = -state.direction;
256   return result;
257 }
258
259 /* Jedan korak RGF45 integracije, uključujući adaptaciju. */
260 std::pair<ODEState, double> advance_geodesic_RGF45(const double min_h,
261                                                    double h,
262                                                    const double max_h,
263                                                    const double epsilon,
264                                                    const ODEState &state) {
265   // ORIGINAL: include/raytracer.h
266   // Ovdje je adaptacija koraka h djelomično modificirana u odnosu na RGF45.
267   constexpr double SAFETY = 0.84;
268   constexpr double SAFETY_INC = 1.2;
269   constexpr double SAFETY_INC_MAX = 3.0;
270
271   int limit = 0;
272   for (;;) {
273     ODEState final4, final5;
274     std::tie(final4, final5) = integration_step_RGF45(geodesic_RHS, h, state);
275     double R = (numerical_distance(final4.position, final5.position)
276               + numerical_distance(final4.direction, final5.direction)) / h;
277     double delta = SAFETY * std::pow(epsilon / R, 0.25);
278     if (R <= epsilon) {
279       double new_h = delta > SAFETY_INC_MAX
280           ? std::min(h * SAFETY_INC_MAX, max_h)
281           : (delta > SAFETY_INC ? std::min(h * delta / SAFETY_INC, max_h) : h);
282       return std::make_pair(final4, new_h);
283     } else if (h == min_h) {
```

```
284       return std::make_pair(final4, h);
285     }
286
287     if (++limit == 20) {
288       fprintf(stderr, "Prevelik broj koraka u advance_geodesic_RGF45!\n");
289       exit(1);
290     }
291
292     h *= delta;
293     if (h < min_h) h = min_h;
294     if (h > max_h) h = max_h;
295   }
296 }
297
298 /* Rješava jednadžbu geodezika dy/dlambda = f(y), uz sve uvjete prekida. */
299 std::pair<ODEState, int> generate_geodesic(const BoyerLindquist &position,
300                                            const BoyerLindquist &direction) {
301   // ORIGINAL: include/raytracer.h
302   ODEState state0{position, direction};  // Trenutačno stanje y(lambda).
303   double dlambda0 = dlambda_initial;     // Korak dlambda.
304
305   for (int n = 0; n < MAX_N; ++n) {
306     ODEState state1;  // Novo stanje i novi korak dlambda.
307     double dlambda1;
308     std::tie(state1, dlambda1) = advance_geodesic_RGF45(
309         dlambda_min, dlambda0, dlambda_max, epsilon, state0);
310
311     if (state1.position.r > max_r)
312       return std::make_pair(state1, TARGET_INFINITY);
313     if (state1.position.r < rs * (1 + 1e-5)
314         || !state1.position.isfinite()
315         || !state1.direction.isfinite()) {
316       return std::make_pair(state1, TARGET_BLACK_HOLE);
317     }
318
319     // Usporavanje u blizini diska ako prelazimo z == 0 ravninu.
320     const auto z0 = std::cos(state0.position.theta) * state0.position.r;
321     const auto z1 = std::cos(state1.position.theta) * state1.position.r;
322     const auto r = state0.position.r;
323     if (((z0 >= 0 && z1 < 0) || (z0 <= 0 && z1 > 0))
324         && 0.5 * disk_inner_r < r && r < 1.5 * disk_outer_r) {
325       if (dlambda0 == dlambda_min) {
326         if (disk_inner_r < r && r < disk_outer_r)
327           return std::make_pair(state1, TARGET_DISK);
328       } else if (0.7 * disk_inner_r < r && r < 1.3 * disk_outer_r) {
329         dlambda0 = std::max(dlambda_min, dlambda0 / 2);
330         continue;  // Smanji dlambda i ponovi korak integracije.
331       }
332     }
333
334     state0 = state1;
335     dlambda0 = dlambda1;
336   }
337
338   return std::make_pair(state0, TARGET_TOO_MANY_STEPS);
339 }
340
341
342 /******************************** KAMERA **********************************/
```

```
343
344 /* Računa smjer zrake za zadanu rel. poziciju pixela x, y (-1 <= x, y <= 1). */
345 class ProjectionCamera {  // ORIGINAL: include/raytracer.h
346  private:
347   Vector3 view, right;
348   double fovy_tan;
349  public:
350   Vector3 eye, center, up;
351   double aspect;
352
353   ProjectionCamera(const Vector3 &_eye, const Vector3 &_center,
354                    const Vector3 &_up, double _aspect, double _fovy)
355         : eye(_eye), center(_center), up(_up), aspect(_aspect) {
356     view = (center - eye).normalized();
357     right = cross(view, up).normalized();
358     up = cross(right, view);
359     fovy_tan = std::tan(_fovy / 180 * M_PI / 2.0);
360   }
361
362   std::pair<Vector3, Vector3> ray(double x, double y) const {
363     Vector3 direction = view
364                       + (fovy_tan * aspect * x) * right
365                       - (fovy_tan * y) * up;
366     return std::make_pair(eye, direction);
367   }
368 };
369
370 void matrix3_inverse(const double m[3][3], double out[3][3]) {
371   // ORIGINAL: include/matrix.h
372   double det = m[0][0] * (m[1][1] * m[2][2] - m[2][1] * m[1][2])
373              - m[0][1] * (m[1][0] * m[2][2] - m[1][2] * m[2][0])
374              + m[0][2] * (m[1][0] * m[2][1] - m[1][1] * m[2][0]);
375   double invdet = 1 / det;
376   out[0][0] = (m[1][1] * m[2][2] - m[2][1] * m[1][2]) * invdet;
377   out[0][1] = (m[0][2] * m[2][1] - m[0][1] * m[2][2]) * invdet;
378   out[0][2] = (m[0][1] * m[1][2] - m[0][2] * m[1][1]) * invdet;
379   out[1][0] = (m[1][2] * m[2][0] - m[1][0] * m[2][2]) * invdet;
380   out[1][1] = (m[0][0] * m[2][2] - m[0][2] * m[2][0]) * invdet;
381   out[1][2] = (m[1][0] * m[0][2] - m[0][0] * m[1][2]) * invdet;
382   out[2][0] = (m[1][0] * m[2][1] - m[2][0] * m[1][1]) * invdet;
383   out[2][1] = (m[2][0] * m[0][1] - m[0][0] * m[2][1]) * invdet;
384   out[2][2] = (m[0][0] * m[1][1] - m[1][0] * m[0][1]) * invdet;
385 }
386
387 /* Pretvaranje pozicije i vektora iz Kartezijevih u Boyer-Lindquist koord. */
388 std::pair<BoyerLindquist, BoyerLindquist> cartesian_to_boyer_lindquist(
389     const Vector3 &pos, const Vector3 &dir) {  // ORIGINAL: include/coordinate.h
390   double ss = sqr(pos.x) + sqr(pos.y);
391   double s = std::sqrt(ss);
392   double zz = sqr(pos.z);
393   double RR_aa = ss + zz - aa;
394   double rr = (RR_aa + std::sqrt(sqr(RR_aa) + 4 * aa * zz)) / 2;
395   double r = std::sqrt(rr);
396   double ra = std::sqrt(rr + aa);
397   double cost = pos.z / r;
398   double sint = s / ra;
399   double cosp = pos.x / s;
400   double sinp = pos.y / s;
401   double inv[3][3], mat[3][3] = {
```

```
    {r / ra * sint * cosp, r * cost * cosp, -r * sint * sinp},
    {r / ra * sint * sinp, r * cost * sinp, r * sint * cosp},
    {cost, -r * sint, 0}
  };
  matrix3_inverse(mat, inv);

  BoyerLindquist position, direction;
  position.t = 0.0;  // Nije bitno.
  position.r = r;
  position.theta = std::atan2(s * r, ra * pos.z);
  position.phi = std::atan2(pos.y, pos.x);

  direction.t = 1.0;  // Approx., u redu ako je kamera daleko od crne rupe.
  direction.r     = inv[0][0] * dir.x + inv[0][1] * dir.y + inv[0][2] * dir.z;
  direction.theta = inv[1][0] * dir.x + inv[1][1] * dir.y + inv[1][2] * dir.z;
  direction.phi   = inv[2][0] * dir.x + inv[2][1] * dir.y + inv[2][2] * dir.z;

  return std::make_pair(position, direction);
}


/********************** GENERIRANJE I SPREMANJE SLIKE ************************/

/* TGA (u osnovnom obliku) je izrazito jednostavan format slike.
 * Boje se spremaju u poretku BGRA, a ne RGBA!. */
void save_TGA(const BGRA *image, int width, int height, const char *filename) {
  // ORIGINAL: src/tga.cpp
  unsigned char header[18] = {0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
  header[12] = width         & 0xFF;
  header[13] = (width >> 8)  & 0xFF;
  header[14] = height        & 0xFF;
  header[15] = (height >> 8) & 0xFF;
  header[16] = 32;  // BPP.
  header[17] = 32;  // (Isključi) vertikalno zrcaljenje.

  FILE *f = fopen(filename, "wb");
  if (f == NULL) exit(10);
  if (fwrite(header, 18, 1, f) != 1) exit(11);
  if (fwrite(image, width * height * 4, 1, f) != 1) exit(12);
  fclose(f);
}

/* Boja jednog px, u ovisnosti o pogođenoj meti i konačnom stanju geodezika. */
BGRA get_color(const ODEState &state, int target) {
  // ORIGINAL: include/render.h
  switch(target) {
    case TARGET_BLACK_HOLE:
      return BGRA{0, 0, 255, 255};
    case TARGET_DISK: {
      double phi = 24 * state.position.phi / (2 * M_PI);
      phi = 2 * M_PI * std::floor(phi) / 24;
      double radial = (disk_outer_r - state.position.r)
                    / (disk_outer_r - disk_inner_r);
      unsigned char r = 255 * (.5 - .5 * std::cos(phi));
      unsigned char g = 255 * (.25 + (int)(radial * 10) / 10. * .75);
      return BGRA{0, g, r, 255};
    }
    case TARGET_INFINITY:
      return BGRA{0, 0, 0, 255};
```

```
461     // case TARGET_TOO_MANY_STEPS:
462     default:
463       return BGRA{255, 0, 255, 255};
464   };
465 }
466
467 int main(void) {
468   const int width = 320;
469   const int height = 3 * width / 4;
470   BGRA *image = new BGRA[width * height];
471   if (image == NULL) exit(1);
472
473   double fovy = 1.0;                              // Kut gledanja.
474   double camera_dist = 1000 * rs;                 // Udaljenost kamere.
475   double camera_theta = 75 * M_PI / 180.;         // Kut pozicije u odnosu na z-os.
476   double cos_theta = std::cos(camera_theta);
477   double sin_theta = std::sin(camera_theta);
478   assert(camera_dist < max_r);
479   // Gdje je kamera (eye), u što gleda (center) i kako je orijentirana (up).
480   Vector3 eye = camera_dist * Vector3{sin_theta, 0, cos_theta};
481   Vector3 up = camera_dist * Vector3{-cos_theta, 0, sin_theta};
482   Vector3 center{0, 0, 0};
483   ProjectionCamera camera(eye, center, up, (double)width / height, fovy);
484
485   // ORIGINAL: include/render.h (generate_image)
486   for (int i = 0; i < height; ++i) {
487     for (int j = 0; j < width; ++j) {
488       double x = double(1 + 2 * j - width) / width;     // -1 <= x <= 1
489       double y = double(1 + 2 * i - height) / height;   // -1 <= y <= 1
490       std::pair<Vector3, Vector3> ray = camera.ray(x, y);
491
492       BoyerLindquist position, direction;
493       std::tie(position, direction) = cartesian_to_boyer_lindquist(
494           ray.first, -ray.second);  // Minus na smjer jer zraku šaljemo unatrag.
495
496       std::pair<ODEState, int> result = generate_geodesic(position, direction);
497       image[i * width + j] = get_color(result.first, result.second);
498     }
499     if (i % 16 == 0) fprintf(stderr, "%.1lf%% ", 100. * (i + 1) / height);
500   }
501   fprintf(stderr, "\n");
502
503   save_TGA(image, width, height, "output.tga");
504   delete []image;
505   return 0;
506 }
```

Literatura

[1] Bin Chen, Ronald Kantowski, Xinyu Dai, Eddie Baron, Prasad Maddumage. Algorithms And Programs For Strong Gravitational Lensing In Kerr Space-time Including Polarization. *Astrophys. J. Suppl.*, 218:4, 2015, arXiv:1505.02714.

[2] N. I. Shakura, R. A. Sunyaev. Black holes in binary systems. Observational appearance. *Astron. Astrophys.*, 24:337–355, 1973.

[3] Wikipedia. Orders of magnitude (magnetic field) — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Orders_of_magnitude_(magnetic_field)&oldid=723344862, 2016. [8. srpnja 2016.].

[4] McGill Pulsar Group. Mcgill online magnetar catalog. http://www.physics.mcgill.ca/~pulsar/magnetar/main.html. [8. srpnja 2016.].

[5] Jin Young Kim, Taekoon Lee. Light bending by nonlinear electrodynamics under strong electric and magnetic field. *JCAP*, 1111:017, 2011, arXiv:1101.3433.

[6] R. R. Cuzinatto, C. A. M. Melo, K. C. Vasconcelos, L. G. Medeiros, P. J. Pompeia. Non-linear effects on radiation propagation around a charged compact object. *Astrophys. Space Sci.*, 359(2):59, 2015, arXiv:1501.06234.

[7] Z. Bialynicka-Birula, I. Bialynicki-Birula. Nonlinear effects in quantum electrodynamics. photon propagation and photon splitting in an external field. *Phys. Rev. D*, 2:2341–2345, Nov 1970.

[8] P. A. Vshivtseva, M. M. Denisov. Mathematical modeling of electromagnetic wave propagation in nonlinear electrodynamics. *Computational Mathematics and Mathematical Physics*, 49(12):2092–2102, 2009.

[9] Felix Karbstein, Rashid Shaisultanov. Photon propagation in slowly varying inhomogeneous electromagnetic fields. *Phys. Rev.*, D91(8):085027, 2015, arXiv:1503.00532.

[10] Wu-yang Tsai, Thomas Erber. Propagation of photons in homogeneous magnetic fields: Index of refraction. *Phys. Rev. D*, 12:1132–1137, Aug 1975.

[11] Thomas Muller, Frank Grave. Catalogue of Spacetimes. 2009, arXiv:0904.4184.

[12] NASA Dr. David R. Williams. Sun fact sheet. http://nssdc.gsfc.nasa.gov/planetary/factsheet/sunfact.html. [8. srpnja 2016.].

[13] Roy P. Kerr. Gravitational field of a spinning mass as an example of algebraically special metrics. *Phys. Rev. Lett.*, 11:237–238, Sep 1963.

[14] R. M. Wald. *General Relativity*. The University of Chicago Press, 1984.

[15] S. V. Iyer, E. C. Hansen. Light's Bending Angle in the Equatorial Plane of a Kerr Black Hole. *Phys. Rev.*, D80:124023, 2009, arXiv:0907.5352.

[16] Scholarpedia. Accretion disks. http://www.scholarpedia.org/w/index.php?title=Accretion_discs&oldid=145813. [8. srpnja 2016.].

[17] Subrahmanyan Chandrasekhar. *Radiative transfer*. Dover publications, New York, 1960.

[18] Oliver James, Eugénie von Tunzelmann, Paul Franklin, Kip S Thorne. Gravitational lensing by spinning black holes in astrophysics, and in the movie interstellar. *Classical and Quantum Gravity*, 32(6):065001, 2015.

[19] I. D. Novikov, K. S. Thorne. Astrophysics of black holes. In C. Dewitt, B. S. Dewitt, editors, *Black Holes (Les Astres Occlus)*, pages 343–450, 1973.

[20] Don N. Page, Kip S. Thorne. Disk-Accretion onto a Black Hole. Time-Averaged Structure of Accretion Disk. *Astrophys. J.*, 191:499–506, 1974.

[21] Ulrich D. Jentschura, Holger Gies, Sree Ram Valluri, Darrell R. Lamm, Ernst Joachim Weniger. QED effective action revisited. *Can. J. Phys.*, 80:267–284, 2002, arXiv:hep-th/0107135.

[22] M. Novello, V. A. De Lorenci, J. M. Salim, Renato Klippert. Geometrical aspects of light propagation in nonlinear electrodynamics. *Phys. Rev.*, D61:045001, 2000, arXiv:gr-qc/9911085.

[23] W. Heisenberg, H. Euler. Consequences of Dirac's theory of positrons. *Z. Phys.*, 98:714–732, 1936, arXiv:physics/0605038.

[24] Jeremy S. Heyl, Lars Hernquist. An Analytic form for the effective Lagrangian of QED and its application to pair production and photon splitting. *Phys. Rev.*, D55:2449–2454, 1997, arXiv:hep-th/9607124.

[25] P. Bakala, E. Šrámková, Z. Stuchlík, G. Török. Dipole magnetic field on a Schwarzschild background and related epicyclic frequencies. In S. Hledík, Z. Stuchlík, editors, *Proceedings of RAGtime 8/9: Workshops on Black Holes and Neutron Stars*, pages 1–10, dec 2007.

[26] Alex T. Deibel, Andrew W. Steiner, Edward F. Brown. Magnetar giant flare oscillations and the nuclear symmetry energy. *Phys. Rev.*, C90(2):025802, 2014, arXiv:1303.3270.

[27] Graham M. Shore. Quantum gravitational optics. *Contemp. Phys.*, 44:503–521, 2003, arXiv:gr-qc/0304059.

[28] F. Bastianelli, O. Corradini, J. M. Davila, C. Schubert. Photon-Graviton Amplitudes from the Effective Action. *Phys. Part. Nucl.*, 43:630–634, 2012, arXiv:1203.1689.

[29] Fiorenzo Bastianelli, Jose Manuel Davila, Christian Schubert. Gravitational corrections to the Euler-Heisenberg Lagrangian. *JHEP*, 03:086, 2009, arXiv:0812.4849.

[30] Gerald V. Dunne. Heisenberg-Euler effective Lagrangians: Basics and extensions. *In *Shifman, M. (ed.) et al.: From fields to strings, vol. 1* 445-522*, 2004, arXiv:hep-th/0406216.

[31] Gerald V. Dunne, Christian Schubert. Closed form two loop Euler-Heisenberg Lagrangian in a selfdual background. *Phys. Lett.*, B526:55–60, 2002, arXiv:hep-th/0111134.

[32] Yanqiu Wang. Runge-kutta method. https://math.okstate.edu/people/yqwang/teaching/math4513_fall11/Notes/rungekutta.pdf. [8. srpnja 2016.].

[33] Wikipedia. Curiously recurring template pattern — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Curiously_recurring_template_pattern&oldid=725390448, 2016. [8. srpnja 2016.].

[34] Wikipedia. Kahan summation algorithm — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Kahan_summation_algorithm&oldid=720674932, 2016. [8. srpnja 2016.].