

<https://helda.helsinki.fi>

General Purpose Ray Tracing and Polarized Radiative Transfer in General Relativity

Pihajoki, Pauli

2018-08-10

Pihajoki , P , Mannerkoski , M , Nattila , J & Johansson , P H 2018 , ' General Purpose Ray Tracing and Polarized Radiative Transfer in General Relativity ' , Astrophysical Journal , vol. 863 , no. 1 , 8 . <https://doi.org/10.3847/1538-4357/aacea0>

<http://hdl.handle.net/10138/240782>

<https://doi.org/10.3847/1538-4357/aacea0>

cc_by

publishedVersion

Downloaded from Helda, University of Helsinki institutional repository.

This is an electronic reprint of the original article.

This reprint may differ from the original in pagination and typographic detail.

Please cite the original version.



General Purpose Ray Tracing and Polarized Radiative Transfer in General Relativity

Pauli Pihajoki¹ , Matias Mannerkoski¹, Joonas Näätäla² , and Peter H. Johansson¹ 

¹ Department of Physics, University of Helsinki, Gustaf Hällströmin katu 2a P.O. Box 64, FI-00014 University of Helsinki, Finland; pauli.pihajoki@iki.fi

² Nordita, KTH Royal Institute of Technology and Stockholm University, Roslagstullsbacken 23 SE-10691 Stockholm, Sweden

Received 2018 April 12; revised 2018 June 6; accepted 2018 June 20; published 2018 August 6

Abstract

Ray tracing is a central tool for constructing mock observations of compact object emission and for comparing physical emission models with observations. We present ARCMANCER, a publicly available general ray-tracing and tensor algebra library, written in C++ and providing a Python interface. ARCMANCER supports Riemannian and semi-Riemannian spaces of any dimension and metric, and has novel features such as support for multiple simultaneous coordinate charts, embedded geometric shapes, local coordinate systems, and automatic parallel propagation. The ARCMANCER interface is extensively documented and user friendly. While these capabilities make the library well suited for a large variety of problems in numerical geometry, the main focus of this paper is in general relativistic polarized radiative transfer. The accuracy of the code is demonstrated in several code tests and in a comparison with GRTRANS, an existing ray-tracing code. We then use the library in several scenarios as a way to showcase the wide applicability of the code. We study a thin variable-geometry accretion disk model and find that polarization carries information of the inner disk opening angle. Next, we study rotating neutron stars and determine that to obtain polarized light curves at better than a $\sim 1\%$ level of accuracy, the rotation needs to be taken into account both in the spacetime metric and in the shape of the star. Finally, we investigate the observational signatures of an accreting black hole lensed by an orbiting black hole. We find that these systems exhibit a characteristic asymmetric twin-peak profile both in flux and polarization properties.

Key words: accretion, accretion disks – gravitation – gravitational lensing: strong – methods: numerical – polarization – radiative transfer

1. Introduction

Fully covariant radiative transfer in general relativity (GR) presents distinct complications. Due to gravity, the path of a wave front of radiation is curved even in vacuum. This leads to gravitational lensing, which causes measurable effects from the scales of the cosmic microwave background (Weinberg et al. 2013) and galaxy clusters (Treu 2010) all the way to supermassive black holes (SMBHs) in the centers of galaxies (Luminet 1979), and down to single neutron stars (Pechenick et al. 1983). Similarly, the rotation of spacetime itself, such as around rotating Kerr black holes, can cause an observable rotation of the direction of polarization of light. This phenomenon is known as (gravitational) Faraday rotation (Connors & Stark 1977; Stark & Connors 1977; Ishihara et al. 1988). Finally, the observed intensity is also dependent on the relative position and velocity of the observer with respect to the elements of the emitting, absorbing, and scattering medium—typically an astrophysical plasma—through which the light has propagated (e.g., Gammie & Leung 2012). This dependence is responsible for such effects as Doppler (de-)boosting, via velocities of the emitter and observer, and gravitational and cosmological redshifts, via relative positions in spacetime.

A full (classical) solution of the polarized radiative transfer problem in GR requires simultaneously solving the Einstein field equations, the magnetohydrodynamic equations of motion of the radiating and interacting matter, and the curved-space Maxwell equations. This is a formidable undertaking, also in terms of computational resources, and significant progress has been made only relatively recently (see Kelly et al. 2017 and references therein). The problem becomes less taxing by assuming that the radiation field makes a negligible contribution to both the spacetime curvature and the motion of the

interacting medium. In this case, the underlying spacetime structure and the state of the interacting medium can be specified by either analytic means or a separate numerical computation. However, even in this case, the full curved-space Maxwell equations need to be solved in the entire computational domain, which is still a computationally demanding task.

The situation is considerably simplified by the fact that the exact time-dependent behavior of the electromagnetic \mathbf{E} and \mathbf{B} fields is not usually required, and knowledge of the radiative energy flow, i.e., the specific intensity, is enough. In this case, it suffices to solve the usual radiative transfer equation while taking into account GR. There are numerous approaches with different limitations to solving the radiative transfer equation, such as Monte Carlo (MC) methods or the method of characteristics (see, e.g., Baron & Hauschildt 2004 and references therein). One of the perhaps conceptually simplest approaches is to use ray tracing. In ray tracing, a mock observation can be constructed by connecting the observer to the emitting region through null geodesics (when plasma effects are unimportant; otherwise, see, e.g., Gedalin & Melrose 2001 and Broderick & Blandford 2003) through either analytic or numerical means. The bending of these geodesics captures the lensing effects of the gravitational field. The relativistic polarized radiative transfer equation can then be solved along these geodesics to capture the remaining relativistic effects. This process, called ray-tracing, is computationally efficient and naively parallelizable, enabling high-resolution mock observations to be computed in seconds or minutes on a standard desktop computer. However, straightforward ray-tracing methods are limited to problems where scattering is not dominant. This is due to the fact that strong scattering couples all directions and spatial locations of the solution, whereas simple ray-tracing only samples the rays

reaching the observer. In addition, if the radiative eigenmodes propagate differently due to strong plasma effects, then all eigenmodes must be propagated separately and similarly, radiative transfer must be computed multiple times, increasing the workload considerably.

Despite these limitations, using ray-tracing to compute mock observations of highly relativistic objects has a relatively long history. Already in Cunningham & Bardeen (1972), the light curve of a star orbiting around a black hole was computed, followed by studies of the effects of gravity on the observed accretion disk spectra (Cunningham 1975, 1976). Polarization effects of relativistic motion and strong gravity in the Kerr solution were studied using ray-tracing in Connors & Stark (1977), Stark & Connors (1977), and Connors et al. (1980). The first resolved mock observation of an accretion flow around a black hole was computed via ray-tracing remarkably early as well, in Luminet (1979). Following these pioneering studies, the ray-tracing approach was quickly adopted in investigations of a great variety of relativistic phenomena, including, but not limited to, hot spots and accretion columns on rotating neutron stars (Pechenick et al. 1983; Riffert & Meszaros 1988), the general mock observation problem in the Kerr spacetime (Viergutz 1993), details of the resolved black hole accretion disk structure (Fukue & Yokoyama 1988; Bromley et al. 2001), accretion disk hot spots (Karas et al. 1992), accretion disk microlensing (Rauch & Blandford 1991; Jaroszynski et al. 1992), accretion disk line profiles (Chen et al. 1989; Ebisawa et al. 1991), optical caustics (Rauch & Blandford 1994), and the shadow cast by the black hole event horizon (Falcke et al. 2000).

In particular, the topics of the black hole shadow and accretion flow as well as the observable polarization properties of neutron stars are currently especially relevant. The interest in black hole shadows and accretion flows is warranted by the recent progress in programs for interferometric observations at the event horizon scales of Sgr A*, the Milky Way SMBH, and the SMBH in M87, the dominant galaxy of the Virgo cluster. The event horizon is approached both in the submillimeter wavelengths, via the Event Horizon Telescope (EHT) VLBI program (Doeleman et al. 2009), and in optical wavelengths, via the VLTI GRAVITY instrument (Eisenhauer et al. 2008). The surging interest is evident also in the number of recent studies focusing on the black hole shadow and accretion flow modeling using ray-tracing, especially in the context of Sgr A* (e.g., Atamurotov et al. 2016; Broderick et al. 2016; Chael et al. 2016; Dexter 2016; García et al. 2016; Vincent et al. 2016; Gold et al. 2017; Porth et al. 2017; Mościbrodzka & Gammie 2018).

Likewise, accurate modeling of the observable properties of neutron stars is timely due to the current and near-future increase in X-ray-sensitive space missions such as *NICER* (Gendreau et al. 2012) and *eXTP* (Zhang et al. 2016), the latter being also sensitive to polarization. In anticipation, a number of recent papers have applied the ray-tracing approach to model observations of neutron stars (e.g., Bauböck et al. 2015a, 2015b; Miller & Lamb 2015; De Falco et al. 2016; González Caniulef et al. 2016; Ludlam et al. 2016; Nättilä & Pihajoki 2018; Vincent et al. 2018).

It is evident even from the short review above that ray-tracing is an important numerical tool, especially for general relativistic radiative transfer in a variety of astrophysical situations. However, the numerical means to compute curves

has an even wider applicability in the sense that in addition to the path of light, curves also represent the timelines of massive particles and observers in a spacetime. Furthermore, it is often convenient to have various tensorial quantities such as local Lorentz frames parallel transported (or more generally, Fermi–Walker transported) along curves. It is also necessary to perform various algebraic computations involving tensor quantities, often mixing different coordinate systems.

To help facilitate numerical studies requiring curve and tensor manipulations in any (semi-)Riemannian context, which naturally includes GR, we have implemented ARCMANCER (Pihajoki et al. 2018),³ a publicly available general ray-tracing and tensor algebra library. From an astrophysical point of view, ARCMANCER is useful for varied such tasks as radiative transfer and mock observations, computing the paths of massive charged particles in curved spacetimes, or calculating the orbits of extreme mass-ratio inspirals (EMRIs). However, the ARCMANCER library offers capabilities beyond purely physically motivated applications. It can compute all kinds of curves, both geodesic and externally forced, on Riemannian and semi-Riemannian manifolds of any dimension and metric, using multiple simultaneously defined coordinate charts to circumnavigate coordinate singularities and to facilitate easy input and output of data in any preferred coordinate system. ARCMANCER can also be used to define tensors of any rank and to perform tensor algebra, as well as, for example, to automatically parallel propagate tensorial quantities along curves. This last feature is particularly useful for problems of radiative transfer, which we will mainly focus on in this paper.

In this paper, we present an overview of the ARCMANCER library and its implementation. We show the results of various code tests to establish the accuracy of the code and present several astrophysical applications using the ARCMANCER library. In this paper, the main focus of the tests and applications is on general relativistic polarized radiative transfer using ray-tracing exclusively. The generality of the ARCMANCER library makes it straightforward to use for more general purpose radiative transfer methods such as MC radiative transfer or hybrid MC–ray-tracing schemes, and the application of these will be demonstrated in future works.

This paper is organized as follows. In Section 2, we present an overview of the ARCMANCER library and its capabilities. In Section 3, we discuss how the various mathematical objects and functionalities provided by the ARCMANCER library are implemented. For convenience, these differential geometric concepts are briefly reviewed in Appendix A, to which Section 3 cross-references. Section 4 describes the implementation details of the radiative transfer scheme implemented in ARCMANCER. In Section 5, we present a series of numerical tests, measuring the accuracy of the numerics implemented in ARCMANCER. These include a test of the radiative transfer features, where the results obtained with ARCMANCER are compared to another recent general relativistic code GRTRANS (Dexter 2016). In Section 6, the ARCMANCER code is applied to various astrophysical phenomena in order to showcase the versatility of the code. Finally, in Section 7, we give concluding remarks and discuss some future prospects concerning the ARCMANCER library and the ray-tracing approach in astrophysics. This paper has several appendices. Appendix A presents a highly condensed review of the various

³ Codebase: <https://bitbucket.org/popiha/arcmanecer>.

differential geometric concepts used in the code. Appendix B presents the general relativistic polarized radiative transfer equation used in ARCMANCER, and how it relates to the usual flat-space equation. Appendix C presents the built-in manifolds and coordinate systems available in ARCMANCER. The reader interested mainly in a broad overview of the code and its astrophysical applications is urged to browse Sections 2, 5.2.2, and 6. Those interested in technical details may want to read through Sections 3–5, and the appendices as well.

Throughout the paper, we use a system of units where $G = c = 1$, unless explicitly otherwise specified. For Lorentzian spacetimes, we use a metric signature $(+---)$ in the paper, although ARCMANCER supports other signatures as well. The abstract index notation (see Appendix A.2) is assumed throughout.

2. Overview of the Code

The ARCMANCER library consists of a core library, written in modern C++, a Python interface, and a suite of example C++-programs and Python scripts. The core library code, Python interface, and example programs are all thoroughly documented. The code, the examples, and instructions for installation and getting started are all freely available at the code repository, <https://bitbucket.org/popiha/arcmancer>.⁴

The underlying idea behind the ARCMANCER library is to provide all of the mathematical tools needed to perform a large variety of relativistic computations that require numerical tensor algebra and curve propagation. In addition, the library and the Python interface are designed with easy extensibility in mind. These design decisions make it possible to use ARCMANCER for a wide variety of astrophysical problems, including, for example, particle dynamics and radiative transfer, as well as for problems in applied mathematics.

These design goals give ARCMANCER some distinct advantages compared to existing “pure” ray-tracing codes such as GRTRANS (Dexter 2016), GYOTO (Vincent et al. 2011), KERTAP (Chen et al. 2015), GRAY (Chan et al. 2013, 2017), or ASTORAY (Shcherbakov & McKinney 2013). Namely, ARCMANCER can work in any dimension and with metric spaces that are either Lorentzian, as in GR, or purely Riemannian. For Lorentzian geometry, all types of geodesics—null, spacelike, and timelike—are supported, as well as general curves of indeterminate classification. ARCMANCER can also work with spaces for which the geometry, through the metric, is available only numerically, such as from a numerical relativity simulation. In addition, ARCMANCER supports any number of simultaneous coordinate systems with automatic conversion of all quantities between coordinate systems. The use of multiple coordinate systems makes it possible to input and output data in whatever coordinates are most convenient for the given problem. Furthermore, the simultaneous use of multiple coordinates makes it possible for ARCMANCER to avoid coordinate singularities and to automatically choose the numerically most optimal coordinate system for propagating a curve (see Section 3.5).

ARCMANCER provides full support for tensorial quantities of any contra- or covariant rank (see Section 3.2). This support is built on top of the Eigen Linear Algebra Library and includes

all of the usual tensor operations such as sums, products, contractions of indices, and the raising and lowering of indices with the metric. All of these operations are checked at compile time so that mathematically malformed operations, such as mixing points and vectors or contracting two similar indices, are automatically detected. In addition, ARCMANCER can automatically parallel transport all tensor quantities along curves, so that, e.g., smooth local coordinates can be constructed for an observer undergoing arbitrary geodesic motion. This functionality also supports Fermi–Walker transport for accelerating observers and fully general transport for, e.g., accelerating and rotating observers.

ARCMANCER also provides support for including user-defined embedded geometry (see Section 3.4). This feature can be used, for example, to model surfaces of optically thick or solid astrophysical objects, such as planets, photospheres of stars or neutron stars, or optically thick accretion disks. The surfaces are easy to define through level sets and can be given tangential vector fields, which represent movement along the surface, such as in the case of a rotating surface of a neutron star or an optically thick accretion disk.

Finally, while ARCMANCER comes with a suite of built-in spacetimes, coordinates, geometries, and radiation models, the library is designed to be easily extensible by the user. Several examples showcasing this easy extensibility are bundled together with the ARCMANCER library. These examples include such programs as simple black hole and neutron star imagers, as well as a full postprocessor for two-dimensional data produced by the GR magnetohydrodynamics (GRMHD) code HARM (Gammie et al. 2003; Noble et al. 2006).

In the following, we will discuss in more detail how the C++ library implements the mathematical concepts required for the wide variety of applications described above.

3. Implementation of Differential Geometry and Ray-tracing

The main aim of the ARCMANCER implementation is to provide the user with C++/Python objects that match the mathematical objects of differential geometry (see Appendix A) as closely as possible. This approach makes converting mathematical formulae to code straightforward. It also has the additional benefit of eliminating errors stemming from code that expresses mathematically invalid operations. These include, for example, assigning to the components of a point from the components of a vector or a one-form, since all can be expressed as a tuple of n numbers, or assigning to the components of a vector from the components of a vector defined at a different point, in a different chart, or even defined on a different manifold. Likewise, for tensorial quantities, an error such as contracting two similar indices is easily made if working in terms of pure components.

The implementation in ARCMANCER guarantees that all programmed operations correspond to mathematically valid statements. This feature eliminates a large set of logical errors of the kind described above—a major benefit, since currently there are no codebase analysis tools able to identify errors of this kind.

In the following, we describe how the mathematical objects are implemented in the ARCMANCER code. To make the exposition easier to follow, we have provided a list of the most important C++ classes of the ARCMANCER library together with their descriptions in Figure 1. The Python interface

⁴ The version of the code used to produce the results in this paper is version 0.2.0, which corresponds to the commit identifier 38b0879909990746f28e09fb1f94167063608be9 at the master branch of the repository.

MetricSpace<int n> Represents an n -dimensional (semi-)Riemannian manifold M . All quantities are defined with respect to a <code>MetricSpace</code> .	Chart A name tag for a chart $\phi : M \rightarrow \mathbb{R}$. Actual chart functions ϕ are supplied with the <code>MetricSpace</code> .
ManifoldPoint<MetricSpace> Represents a point on the <code>MetricSpace</code> . Can transform coordinates between all known <code>Charts</code> .	Tensor<MetricSpace,Indices...> Represents a tensor on a <code>MetricSpace</code> . The <code>Indices...</code> is an arbitrary combination of tags <code>Cov</code> and <code>Cnt</code> for co- and contravariant indices, respectively.
ParametrizedCurve<MetricSpace,TransportedType> A curve on a <code>MetricSpace</code> , parametrized by a real number. The optional <code>TransportedType</code> is used to specify the type of object to be parallel transported along the curve. A force function can also be specified. If unspecified, the curve is a geodesic.	Surface<MetricSpace> An embedded surface inside <code>MetricSpace</code> , defined as the zero set of a user-defined function. Provides coordinate-independent access to the value and gradient of the function.
LorentzFrame<MetricSpace> Represents the local coordinate system of an observer in a four-dimensional Lorentzian <code>MetricSpace</code> . Convenient for generating initial conditions and computing observed values of physical quantities.	ImagePlane<MetricSpace,DataType> Sets up a 2-dimensional rectangular plane of initial conditions of curves passing orthogonally through. Iterates a user-defined computation over the initial values, and stores the resulting array of values of type <code>DataType</code> .

Figure 1. Short descriptions of the most often-used classes in the ARCMANCER library.

provides corresponding counterparts to these classes, together with some additional convenience classes. A listing of these can be found in the documentation accompanying the code.

3.1. Manifolds and Charts

The most fundamental object in the ARCMANCER library is `MetricSpace<n,Signature>`, representing an n -dimensional (semi-)Riemannian manifold of a given signature. Defining a new `MetricSpace` requires the specification of dimensionality n , one or more charts, and functions returning the components of the metric tensor field and its derivatives in each chart. For four-dimensional semi-Riemannian spaces, the metric signature must also be specified. ARCMANCER supports both timelike (+---) and spacelike (-+++ signatures).

A chart is represented as a class `chart`, which in the current implementation only contains a description and serves to give meaning to a tuple of coordinate numbers. The points on the manifold are implemented as a class `ManifoldPoint<MetricSpace>`. These can be constructed by specifying n coordinates and the corresponding chart. After this, the components of the point can be requested in any available chart, and the object itself behaves much like the mathematical idea of a point on a manifold (see Appendix A.1).

To transform the components of tensorial objects, the transition functions and their Jacobians between the charts must also be specified. For N different charts, this would naively require $N(N - 1)$ transition functions and Jacobians to be implemented. The amount of work increases quadratically.

However, when the domains of charts ϕ_i , ϕ_j , and ϕ_k overlap suitably, the transition function from i -coordinates to j -coordinates fulfills

$$\phi_j \circ \phi_i^{-1} = \phi_j \circ \phi_k^{-1} \circ \phi_k \circ \phi_i^{-1}, \quad (1)$$

and the Jacobian $J_{i \rightarrow j} = d(\phi_j \circ \phi_i^{-1})$ decomposes similarly,

$$J_{i \rightarrow j} = J_{k \rightarrow j} J_{i \rightarrow k}. \quad (2)$$

The ARCMANCER library uses the properties (1) and (2) to build a directed graph of charts, wherein each chart is a node, and the Jacobians and transition functions define the edges. This makes it possible to introduce N charts while supplying only the minimum number of $2(N - 1)$ transition functions and Jacobians to make the graph connected. Then, when the components of a point or a tensorial quantity are requested in a different chart, the code walks through the graph, building the transition function and Jacobian piece by piece using Equations (1) and (2).

For a listing of the built-in metric spaces and chart implementations provided with ARCMANCER, see Appendix C.

3.2. Tensors

The tensor algebra and calculus for tensors of arbitrary rank (see Appendix A.2) are provided by the `Tensor<MetricSpace,Indices...>` template class. Here, `MetricSpace` is the base manifold and `Indices...` is an arbitrary combination of index tags `Cov` and `Cnt`, for covariant or contravariant index, respectively. The implementation is

pointwise, using a set of n^{k+l} components in a given chart to specify a tensor of rank (k, l) on a manifold M with $\dim(M) = n$ at a given `ManifoldPoint`.

As such, similarly to a `ManifoldPoint`, defining a tensor at a point requires the input of n^{k+l} components and the corresponding chart. After this, the chart is abstracted away in the sense that algebraic operations between tensors defined at the same point can be performed irrespective of the chart the tensors were originally defined in. The `Tensor` class provides all of the usual algebraic tensor operations: sum of tensors of the same rank, tensor product, contraction, and in addition, raising and lowering of the indices using the underlying `MetricSpace` structure. The implementation checks all operations for index correctness at compile time so that, e.g., no contraction between indexes of same type is allowed. In addition, during runtime, all operands are inspected to ensure that they are defined at the same base point. These checks guarantee that operations expressed in code correspond to mathematical operations that are well defined.

The `Tensor` class also provides some elements of tensor calculus. Namely, the class automatically computes the derivatives required for parallel transporting a tensor along a general curve. Given a curve tangent vector u^a , the class can compute the contractions with $\Gamma_{bc}^a u^c$ required in the parallel transport Equation (39).

3.3. Curves

Functionality for working with curves γ , including geodesics, is provided by the class `ParametrizedCurve<MetricSpace, TransportedType>` along with a convenience subclass `Geodesic`. Curves are implemented as sequences of points (λ, p) on a manifold, where λ is the curve parameter and $p \in M$. More concretely, the implementation is based on an ordered queue of objects of type `ParametrizedPoint<MetricSpace, TransportedType>`, which combines a `ManifoldPoint` with a real value λ specifying the position along the curve. In addition, the `ParametrizedPoint` can include any arbitrary object A of type `TransportedType` to be parallel transported along a geodesic or Fermi–Walker transported along a forced curve. The only requirement is that the object be representable as a (chart-dependent) tuple of real numbers and that a function $D_A(u^a, \Gamma_{bc}^a u^c, f^a)$ yielding the derivatives $dA(\gamma(\lambda))/d\lambda$ is provided. The function D_A depends externally on the current tangent vector of the curve u^a , the contractions $\Gamma_{bc}^a u^c$, and optionally the force f^a . As mentioned above, the `Tensor` class provides the derivative function automatically, and as such, arbitrary tensors can be parallel transported along all generic curves without any extra programming effort.

In practice, a curve is computed by specifying the initial conditions in some given chart. These consist of the initial point $(\lambda_0, p_0) \in \mathbb{R} \times M$, the components of the curve tangent vector $u^a(p_0) \in T_{p_0}M$, the components $A(p_0)$ of the possible parallel transported object, and an optional force function f^a . The ARCMANCER library then computes points along the curve for the desired interval $I \subset \mathbb{R}$ containing λ_0 by solving the set of equations (see Appendix A.4)

$$\frac{d\gamma(\lambda)}{d\lambda} = u^a, \quad (3)$$

$$\frac{du^a(\lambda)}{d\lambda} = -\Gamma_{bc}^a u^b u^c + f^a, \quad (4)$$

$$\frac{dA(\lambda)}{d\lambda} = D_A(u^a, \Gamma_{bc}^a u^c, f^a) \quad (5)$$

in a suitable chart (see Section 3.5 for details on the chart selection).

ARCMANCER computes the solution using the integration methods offered by the ODEINT C++ library (Ahnert & Mulansky 2011). The default method is the Dormand–Prince fifth-order Runge–Kutta method (Dormand & Prince 1980), which offers error estimation and automatic step-size adjustment, as well as a fair numerical performance in most cases. The absolute and relative error tolerances and step-size and iteration limits are fully user-configurable. After the computation is finished, the `ParametrizedCurve` class provides access to the solution in any chart and for any $\lambda \in I$. Internally, this is achieved through a cubic spline interpolation.

3.4. Surfaces

An interface for implementing hypersurfaces is available through the class `Surface<MetricSpace>`. Surfaces are useful for representing solid or highly optically thick objects, or regions of interest. Examples include not only the surfaces of neutron stars, white dwarfs, or planets, but also black hole event horizons, optically thick accretion disks, or the limits of computational domains. The ARCMANCER implementation of surfaces is based on the concept of level hypersurfaces (see Appendix A.5).

A new surface is implemented by supplying a real-valued function S taking a `ManifoldPoint` as an argument, as well as the gradient $\partial_a S$. The surface is then defined as the set of points $\{p \in M | S(p) = 0\}$. In addition, a tangent vector field t^a on the surface must be defined. This field is primarily used to represent the four-velocity field of observers fixed on the surface and is required for, e.g., computations involving rotating neutron stars (see Section 6.2).

The ARCMANCER library automatically detects intersections of curves with surfaces and numerically finds the exact (to within tolerance) intersection point. The intersections are found by examining the sign of the product $S(p_{k+1})S(p_k)$ for two successive points p_{k+1} and p_k on a curve. If the product is negative, the two points must lie in different regions bounded by the surface. The exact intersection point is then found using the so-called Henon’s trick (Henon 1982). The “trick” consists of changing the independent variable γ , the curve parameter, in Equations (3)–(4) to S , or the value of the surface function. The transformed equations read

$$\frac{d\lambda}{dS} = (u^b \partial_b S)^{-1}, \quad (6)$$

$$\frac{d\gamma(S)}{dS} = (u^b \partial_b S)^{-1} u^a, \quad (7)$$

$$\frac{du^a(S)}{dS} = (u^b \partial_b S)^{-1} (-\Gamma_{bc}^a u^b u^c + f^a), \quad (8)$$

$$\frac{dA(S)}{dS} = (u^b \partial_b S)^{-1} D_A(u^a, \Gamma_{bc}^a u^c, f^a). \quad (9)$$

These equations can then be numerically propagated for a single step of length $-S(p_{k+1})$ starting from the point p_{k+1} to yield the intersection point to within numerical tolerance.

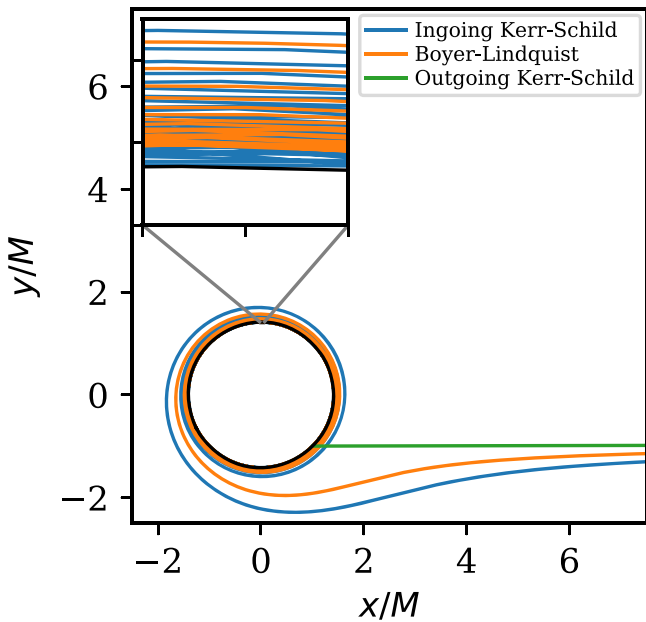


Figure 2. A null geodesic emanating from near the event horizon of an extremal Kerr black hole, shown in the outgoing and ingoing Kerr–Schild coordinates as well as Boyer–Lindquist (BL) coordinates, in the xy projection. For BL coordinates, the transformation $(x, y, z) = \sqrt{M^2 + a^2}(\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)$ was used. The black line shows the location of the event horizon. The inset shows a zoomed-in region from near the event horizon.

3.5. Automatic Chart Selection

Perhaps the most novel and interesting feature of ARCMAN-CER is the possibility to use multiple coordinate charts simultaneously and seamlessly. The most immediate benefit is that objects can be input and output in any available chart, with all transformations handled automatically by ARCMAN-CER. However, there are important computational benefits to the free selection of coordinate charts as well. The most obvious benefit is the fact that a given problem may be much easier to solve numerically in some specific coordinates compared to others. This is illustrated in Figure 2, where the same null geodesic in an extremal Kerr spacetime is shown in the outgoing Kerr–Schild (KS) coordinates, the ingoing KS coordinates, and the Boyer–Lindquist (BL) coordinates (see Appendix C.2.2). From the figure, it is easy to appreciate how in the outgoing KS coordinates the geodesic is essentially straight, and long integration steps can be taken. On the other hand, in the ingoing KS coordinates and the BL coordinates, the geodesic twists around the event horizon at an increasing rate as the event horizon is approached. The magnitudes of the derivatives with respect to the curve parameter increase correspondingly, making the problem eventually numerically impossible to solve.

The possibility to simultaneously use multiple charts makes it possible to avoid the coordinate singularities present in any single chart, such as the pole singularity in any spherical coordinate system or the coordinate singularity at the event horizon present in the usual Schwarzschild coordinates. In addition, using multiple charts makes it possible to switch the chart used for solving the equations of motion for a curve on the fly, useful for situations such as the one depicted in Figure 2. It is not obvious which chart is to be preferred, which

is why ARCMAN-CER currently implements several heuristics for automatically choosing the numerically optimal chart.

The first heuristic consists of finding a chart ϕ_i where the matrix of the components of the metric $\mathbf{G} = (g_{ab})$ has the largest inverse condition number, defined as the ratio of the smallest and largest singular values of the matrix, i.e., $\text{cond}^{-1}(\mathbf{G}) = \sigma_{\min}/\sigma_{\max}$. This is based on two key observations. First, floating point addition and subtraction between numbers of different magnitude cause a loss of precision. Second, the equations of motion for a curve and for parallel transport along it, Equations (3)–(5), contain a mix of the components of the metric and its derivatives on the right-hand side. As such, it would be intuitively advantageous to perform the computations in a chart where the matrix formed by the metric has eigenvalues that span as small a range as possible. This is achieved by maximizing the inverse condition number.

In some cases, the condition number of the metric is not enough to detect a computationally awkward chart. For example, in the case of a Kerr black hole, the condition number cannot differentiate between the ingoing and outgoing KS charts. However, as is seen in Section 5.1.2, using one over the other can cause a large difference in computation time and accuracy for radial geodesics, depending on whether they are falling toward or emanating from the event horizon. As such, a further heuristic is needed.

If the condition number heuristic does not separate two promising charts, the ARCMAN-CER code next tries to minimize the maximal absolute value of the intrinsic derivatives, $-\Gamma_{bc}^a u^b u^c$, of the curve tangent vector u^a . As such, this heuristic needs to know the current curve tangent vector u^a , unlike the condition number test, for which only the current point is required. For Cartesian coordinates in a Euclidean or Minkowskian space, $\Gamma_{bc}^a \equiv 0$, so in effect this procedure looks for the most Cartesian-like chart in which the metric looks most Euclidean (or Minkowskian) in the direction of the current curve tangent vector u^a .

Formal proofs of the performance of these heuristics are beyond the scope of this work, but the numerical results in Section 5 indicate that they work reasonably well.

3.6. Local Lorentz Frames

For four-dimensional Lorentzian manifolds, ARCMAN-CER provides a functionality to construct local Lorentz frames (see Appendix A.6) through the class `LorentzFrame<MetricSpace>`. The user supplies a timelike vector e_t and two spacelike vectors e_z and e_x . From these, a complete Lorentz frame $\{E_t, E_x, E_y, E_z\}$ is constructed by first normalizing e_t to yield E_t and then orthonormalizing e_z and e_x sequentially. Finally, E_y is defined by the remaining orthogonal direction through $E_y^a = \pm \epsilon_{bcd}^a E_t^b E_z^c E_x^d$, where ϵ_{abcd} is the Levi–Civita tensor, with the sign depending on the desired handedness (positive for a right-handed frame).

The `LorentzFrame` object can be automatically parallel transported along a `ParametrizedCurve`. In addition, `Tensor` objects can be constructed from components given with respect to a `LorentzFrame`. Likewise, the components of any `Tensor` can be extracted in a given `LorentzFrame` as well; see Equation (41).

3.7. Image Plane Generation

To produce mock observations, an observational instrument must be emulated somehow. For ray-tracing purposes, this usually means using an image plane. The image plane is positioned near the object of interest, and only the rays intersecting the plane orthogonally are considered. These rays are then assumed to propagate in vacuum all the way to the distant observer. This approximation ignores atmospheric and instrumental effects, but these can be modeled afterwards using dedicated tools if necessary.

There are three main sources of error when generating the image plane: required deviations from perpendicularity, perturbations caused by the curvature of the space, and the assumption of vacuum propagation. The assumption of perpendicularity is typically excellent. For distant objects, the maximum deviation from perpendicularity $\Delta\theta$ is approximately equal to the observed angular size of the object, or $\Delta\theta \sim L/(2D)$, where L is the linear extent of the source perpendicular to the line of sight and D is the distance. For example, in the case of Sgr A* (Sagittarius A), we have $\Delta\theta \sim 10^{-11}$, and for a typical galactic neutron star $\Delta\theta \sim 10^{-17}$. The effects of the remaining spacetime curvature at the image plane location can be estimated by looking at the bending angle β that the image plane rays will make when propagated to infinity. Sufficiently far away from the object so that the Schwarzschild metric can be used, this angle turns out to be (e.g., Beloborodov 2002) $\beta \sim 2GM/(c^2R)$, where M is the total mass of the observed object and R is the radial distance of the image plane from the object. Thus, for $R \gtrsim 10^4 GM/c^2$, we have $\beta \lesssim 2 \times 10^{-4}$, and so the effects of residual curvature are negligible. The assumption of propagation in vacuum is typically valid for objects that are not situated at cosmological distances as far as light bending is concerned. However, corrections for effects such as extinction, frequency dispersion, or Faraday rotation may need to be added in further postprocessing.

In many ray-tracing codes, the construction of image planes is achieved by assuming a flat space and explicitly constructing the starting points and tangent vectors for a planar configuration of geodesics (Broderick 2004; Cadeau et al. 2007; Dexter & Agol 2009; Vincent et al. 2011; Dexter 2016; Chan et al. 2017). ARCMANCER provides a general purpose tool for constructing plane-parallel initial conditions for Lorentzian spacetimes in the class `ImagePlane<MetricSpace, DataType>`. The user specifies a `LorentzFrame` at the center of the plane and the extent and the resolution (number of grid points) of the plane in the local E_x and E_y directions. The local Lorentz frame is then parallel transported to the desired grid points via spacelike geodesics, using the ARCMANCER curve propagation functionality. Initial conditions for curves passing through the plane are set up by assigning the tangent vectors $u^a(0)$ to be spatially parallel to the parallel transported E_z vector. The collection of parallel transported frames defines a best local approximation to a flat plane that is threaded by orthogonal curves and corrects the effect of the bending β caused by the curvature to first order. Thus, the ARCMANCER `ImagePlane` can safely be used in regions where the curvature is small but non-negligible. The method is also general purpose in the sense that it works similarly in any coordinate system and only requires specifying a local Lorentz frame at one point.

4. Implementation of Radiative Transfer

4.1. Fluid and Radiation Models

Radiative transfer functionality in ARCMANCER is built with flexibility in mind. For this purpose, the interface declares two types of functions. The first type of function is `FluidFunction<MetricSpace, FluidData>`, which maps points on the base manifold `MetricSpace` to a user-defined set of fluid variables `FluidData`, which represent local material properties such as temperature or density. The only restriction is that `FluidData` must include a single bulk fluid four-velocity w^a and a single reference direction (often magnetic field) t^a orthogonal to w^a .

The second type of function is `RadiationFunction<FluidData>`, which computes the Stokes emissivity vector \mathbf{J} and the response matrix \mathbf{M} (see Appendix B) from the given `FluidData`, local fluid rest-frame frequency ν , and the rest-frame angle θ between the reference direction t^a and the current direction of the light ray (the tangent vector k^a).

This approach makes implementing different fluid and radiation models rather straightforward. For example, the fluid variables for a given point can be obtained from a GR magnetohydrodynamics (GRMHD) simulation or from an analytic model. The ARCMANCER suite includes an example application which reads outputs from the HARM GRMHD code (Gammie et al. 2003; Noble et al. 2006) and computes mock observations using a thermal synchrotron radiation model based on the results in Dexter (2016). See Section 5.2.2 for computational results.

4.2. Solving the Radiative Transfer Equation

With ARCMANCER, a radiative transfer problem (see Appendix B) is solved by first propagating a set of curves γ_i (typically geodesics, unless plasma effects are significant) along which the radiative transfer equation, Equation (52), is to be solved as a curve integral. Usually, the most convenient approach is to use an `ImagePlane` and let ARCMANCER propagate the set of initial conditions backwards in time through the region of interest. Each propagated curve must include a parallel transported `PolarizationFrame`, a pair of two orthogonal spacelike vectors $\mathcal{P} = \{v^a, h^a\}$, also orthogonal to the geodesic and the four-velocity of the observer, representing the vertical and horizontal linear polarization basis vectors of the observer at one end $\gamma_i(\lambda_{\text{obs}}) = p_{\text{obs}}$ of the curve. If using an `ImagePlane`, these can be conveniently obtained from the E_x and E_y vectors of the local Lorentz frame at each point.

The four-velocity $u^a(p_{\text{obs}})$ of the observer \mathcal{O} at p_{obs} , the four-velocity $w^a(p)$ of the fluid at each point $p = \gamma_i(\lambda)$, and the curve tangents $k^a(p)$ and $k^a(p_{\text{obs}})$ define a connection between the photon frequency ν_0 observed by \mathcal{O} at p_{obs} and the corresponding photon frequency ν in the local rest frame of the fluid at p . This is given by the redshift factor

$$\mathcal{G} = \frac{u_a(p_{\text{obs}})k^a(p_{\text{obs}})}{w_a(p)k^a(p)} = \frac{\nu_0}{\nu}. \quad (10)$$

The initial conditions are set by defining initial invariant specific intensities $\{\mathcal{I}_{\nu_0,i/\mathcal{G}} = \mathbf{I}_{\nu_0/\mathcal{G}}(\nu_0/\mathcal{G})^{-3}\}$ at the other end p_{start} of the curve, one for each observed frequency $\nu_{0,i}$ of interest. Often these can be set to zero, but, for example, in the case of radiation emanating from optically thick or solid surfaces, the initial intensity can be non-zero. Solving the

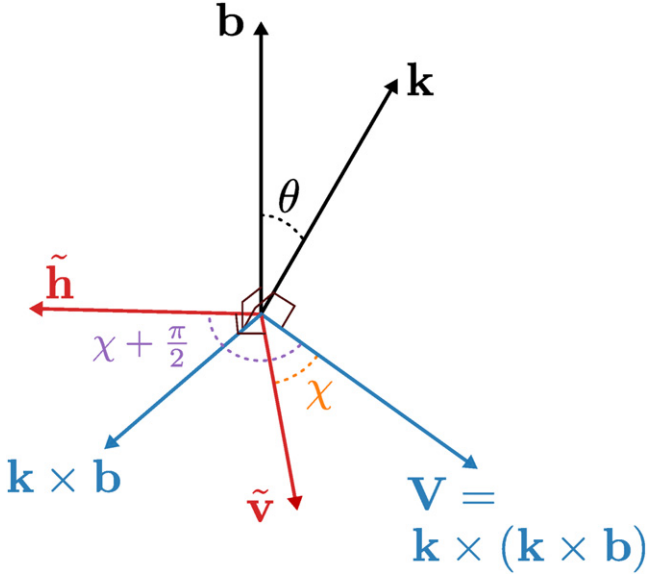


Figure 3. Definition of the angles θ and χ in the three-dimensional rest frame of the fluid. Also shown are the local reference direction \mathbf{b} , the direction of the geodesic \mathbf{k} and the local vertical polarization direction \mathbf{V} . The vectors $\tilde{\mathbf{v}}$ and $\tilde{\mathbf{h}}$ are the spatial parts of \tilde{v}^a and \tilde{h}^a .

radiative transfer equation itself proceeds in a manner following Shcherbakov & Huang (2011). See Figure 3 for a diagram of all the vectors and angles.

At each point $p \in M$ during the calculation, the ARCMAN-CER library evaluates the given FluidFunction to obtain the fluid four-velocity w^a and the rest of the fluid parameters in the rest frame of the fluid. This includes the local reference direction b^a , which typically is the direction of the local magnetic field. From these, the angle $\theta(w; b, k)$ between the reference direction b^a and the light ray tangent k^a as seen in the fluid rest frame is computed using Equation (45). This angle is required by some radiation models, such as synchrotron emission models. The reference direction also defines the local vertical direction of polarization $\mathbf{V} = \mathbf{k} \times (\mathbf{k} \times \mathbf{b})$, where \mathbf{k} and \mathbf{b} are the spatial parts of k^a and b^a , respectively.

The next step is to project the parallel transported polarization frame \mathcal{P} to the fluid rest frame using the screen projection operator, Equation (44), yielding $\tilde{\mathcal{P}} = \{\tilde{v}^a, \tilde{h}^a\}$, where

$$\tilde{v}^a = P_{\perp}(w, k)_b^a v^b, \quad (11)$$

$$\tilde{h}^a = P_{\perp}(w, k)_b^a h^b. \quad (12)$$

Now we can compute the angle χ between the projected parallel transported polarization frame $\{\tilde{v}^a, \tilde{h}^a\}$ and the polarization frame of the fluid, defined by \mathbf{V} , from

$$\tan \chi = \frac{-V_a \tilde{h}^a}{V_a \tilde{v}^a}, \quad (13)$$

where $V^a = (0, \mathbf{V})$.

Next, the angle θ and the fluid parameters are passed to the RadiationFunction to obtain the Stokes emissivity and the response (Müller) matrix \mathbf{M}_{ν} in the fluid rest frame. These are related to the parallel transported and projected polarization frame $\tilde{\mathcal{P}}$ using the angle χ and the transformation properties of the Stokes components under rotation (e.g., Chandrasekhar 1960). The emissivity vector \mathbf{J}_{ν} and response matrix \mathbf{M}_{ν} are transformed via $\mathbf{J}_{\nu} \mapsto R(\chi)\mathbf{J}_{\nu}$ and $\mathbf{M}_{\nu} \mapsto R(\chi)\mathbf{M}\mathbf{R}(-\chi)$,

where

$$R(\chi) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \cos(2\chi) & -\sin(2\chi) & 0 \\ 0 & \sin(2\chi) & \cos(2\chi) & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (14)$$

gives the transformation of Stokes vectors under rotations of the polarization plane. Finally, it can be shown that the Stokes components in any two polarization frames \mathcal{P} and $\tilde{\mathcal{P}}$ related by a screen projection are equal, so that the radiative transfer equation to be solved along the geodesic is

$$\frac{d\mathcal{I}_{\nu}}{d\lambda} = L \frac{u_a(p_{\text{obs}})k^a(p_{\text{obs}})}{\nu_0} (\mathcal{J}_{\nu} - \mathcal{I}_{\nu}\mathcal{M}_{\nu}), \quad (15)$$

where

$$\mathcal{J}_{\nu} = \nu^{-2}R(\chi)\mathbf{J}_{\nu}, \quad (16)$$

$$\mathcal{M}_{\nu} = \nu R(\chi)\mathbf{M}_{\nu}R(-\chi), \quad (17)$$

$$\nu = \nu_0/\mathcal{G}, \quad (18)$$

and L is the unit of length. For example, in problems related to black holes, a typical choice is $L = GM/c^2$, where M is the black hole mass. Internally, Equation (15) is solved using the ODEINT Runge–Kutta–Fehlberg eighth-order method. However, for problems where the optical thickness is large, Equation (15) can become stiff, and an implicit method would provide better performance.

5. Code Tests

5.1. Curves, Parallel Transport, and Chart Selection

5.1.1. Geodesic Propagation

The accuracy of the basic curve propagation functionality (Section 3.3) was verified by investigating curves on a two-dimensional spherical surface. The computations were performed both in two dimensions, using the intrinsic spherical coordinate chart (θ, ϕ) , Equation (53), and in a three-dimensional Euclidean slice at $t = 0$ of the Minkowski space, using the spherical coordinates $(0, r, \theta, \phi)$, Equation (55). To force the curve to stay on the surface of a sphere in the three-dimensional case, a constraint force $f(u^a) = (0, -u^a u_a, 0, 0)$ was specified. Here, u^a is the curve tangent, in three-dimensional spherical coordinates.

Numerical convergence was estimated using a single geodesic curve $\gamma(\lambda)$ passing through $(\theta, \phi) = (\pi/2, 0)$ at $\lambda = 0$ with a tangent vector $u^a = (\dot{\theta}, \dot{\phi}) = (1.0, 0.3)$. The initial values were chosen so as to avoid a purely polar or equatorial geodesic, but were otherwise chosen arbitrarily. The geodesic was computed several times using a range of equal relative and absolute numerical tolerances ϵ_{rel} and ϵ_{abs} from 10^{-20} to 10^{-2} in 40 steps. The differences between the numerical results and the known analytical solution are shown in Figure 4. We see that both in the intrinsic two-dimensional and the constrained three-dimensional case, the numerical curves converge toward the analytical solution linearly with the tolerance parameters. The convergence saturates at tolerance parameters $\sim 10^{-15}$ when the relative precision floor of the double precision floating point numbers is reached.

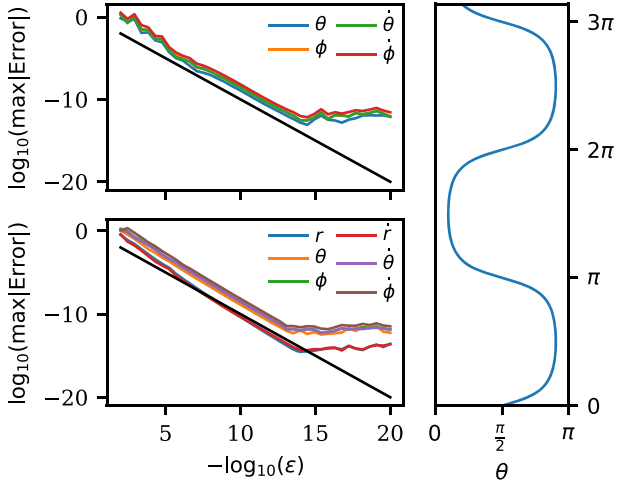


Figure 4. Differences between the analytic and the numerically computed geodesic on a spherical surface, fulfilling $(\theta(0), \phi(0)) = (\pi/2, 0)$ and $(\dot{\theta}(0), \dot{\phi}(0)) = (1.0, 0.3)$. Top-left panel: results computed using the intrinsic two-dimensional metric. Shows maximal numerical errors along the curve in the coordinate position (θ, ϕ) and the tangent vector $(\dot{\theta}, \dot{\phi})$ as a function of the tolerance $\epsilon_{\text{abs}} = \epsilon_{\text{rel}} = \epsilon$. Bottom-left panel: results computed in a three-dimensional space using a constraint force. Shows maximal numerical errors along the curve in the coordinate position (r, θ, ϕ) and the tangent vector $(\dot{r}, \dot{\theta}, \dot{\phi})$ as a function of the tolerance. Right panel: the orbit of the curve in the two-dimensional spherical coordinate chart. The solid black line segments in the top and bottom left-hand panels are guides to the eye and show the identity function $f(\epsilon) = \epsilon$.

5.1.2. Parallel Propagation in the Kerr Spacetime

The functionality for parallel transporting tensorial quantities (see Sections 3.2 and 3.3) along a curve was assessed in the context of a Kerr spacetime (see Section C.2.2) with a near-extremal non-dimensional spin parameter $\chi = 0.99$ and mass $M = 1$. First, the initial conditions $\gamma(0) = (t_0, r_0, \theta_0, \phi_0) = (0, 10, 1, 0)$ and $u^a(0) = (1, 0, 0.01, 0.03)$ were fixed in the BL coordinates (see Equation (56)). These initial values were chosen to yield a generic timelike geodesic, and to avoid special cases such as equatorial geodesics, but were otherwise chosen arbitrarily. The geodesic was then augmented by including the metric g_{ab} and a Lorentz frame $\{E_r, E_x, E_y, E_z\}$ as quantities to be parallel transported. The geodesic was then computed until $\lambda = 990$ to yield several complete orbits around the black hole, using tolerances $\epsilon_{\text{abs}} = \epsilon_{\text{rel}} = 10^{-10}$. Finally, the parallel transported values were evaluated for accuracy by comparing to analytic expectations.

Figure 5 shows the orbit of the geodesic. It also depicts magnitudes of the maximum difference $\max|\Delta g_{ab}|$ of the components of the parallel transported metric with respect to the analytic expression, both computed in the ingoing KS chart. Also shown are the absolute values of all the pairwise inner products of the parallel propagated Lorentz frame which should be identically zero. From the figure, we see that the errors in all of these conserved quantities increase in a secular fashion, while the single step errors are below the set numerical tolerance. This is an expected and well-known behavior for non-symplectic numerical integration methods, such as the fifth-order Dormand–Prince scheme used in ARCMANCER, which does not respect the geometric structure of the phase space (Hairer et al. 2008). Symplectic methods for the inseparable Hamiltonians occurring in geodesic propagation have been discovered recently (Pihajoki 2015), but these are not yet available in ODEINT. In general, the secular

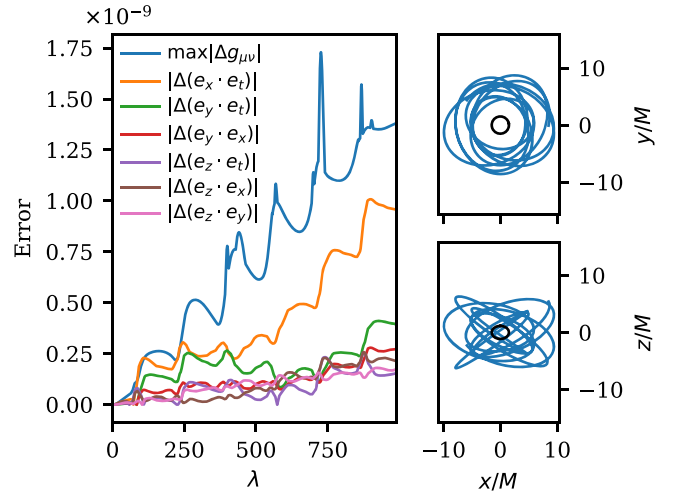


Figure 5. Left panel: the absolute values of the errors accumulated during the parallel transport of the metric g_{ab} and a local Lorentz frame $\{E_r, E_x, E_y, E_z\}$, computed in the ingoing Kerr–Schild coordinates. Right panels: the orbit of the geodesic along which the parallel transport was computed, shown in ingoing Kerr–Schild coordinates using xy (top) and xz projections (bottom). The black circle shows the location of the event horizon.

accumulation of integration error poses no problem for the applications we demonstrate in this paper. However, for integrations over long periods of time, such as for computing dynamics of massive particles orbiting a black hole, a symplectic method for inseparable Hamiltonians might need to be implemented.

The accuracy and performance of both the curve propagation and parallel transport functionality were also assessed as a function of the geodesic and the coordinate chart. To this end, we set up an image plane at $(r_0 = 10^5, \theta_0 = 50^\circ)$ in the BL coordinates of a Kerr spacetime with $\chi = 0.95$ and $M = 1$. From the image plane, null geodesics were propagated backwards from $\lambda = 0$ to $\lambda = -2r_0$ or until intersection with a surface slightly outside the event horizon, defined by $r = 1.03r_H$, where r_H is the event horizon radius. This radius was chosen since the computation in the BL and ingoing KS coordinates must be terminated before the event horizon itself (see Figure 2). The geodesics were computed three times, each time fixing the chart (automatic chart selection disabled) to either ingoing KS, outgoing KS, or the BL chart. Standard tolerances of $\epsilon_{\text{rel}} = \epsilon_{\text{abs}} = 10^{-10}$ were used. We then computed the maximal absolute errors in the value of the curve Hamiltonian, $H(x, k) = k_a k^a = 0$, and the trace g^a_a of the parallel transported metric along the geodesics and plotted these on the image plane, in addition to the number of integration steps N . The results are shown in Figure 6.

From the figure, it is evident that the outgoing KS coordinates offer significantly better numerical performance than the ingoing KS coordinates or the BL coordinates. This is not surprising, since the outgoing KS chart is adapted to radially outgoing null geodesics. As a consequence, the more radial the geodesic is, the more nearly a straight line it is in the outgoing KS chart. In the figure, this can be seen as the remarkable decrease in the maximal error and the number of computational steps for the geodesics starting near the origin of the image plane (see also Figure 2). On the other hand, the BL coordinates are seen to perform significantly worse. This is related to both the fact that the geodesic “wraps around” the black hole near the event horizon (see Figure 2), but also the

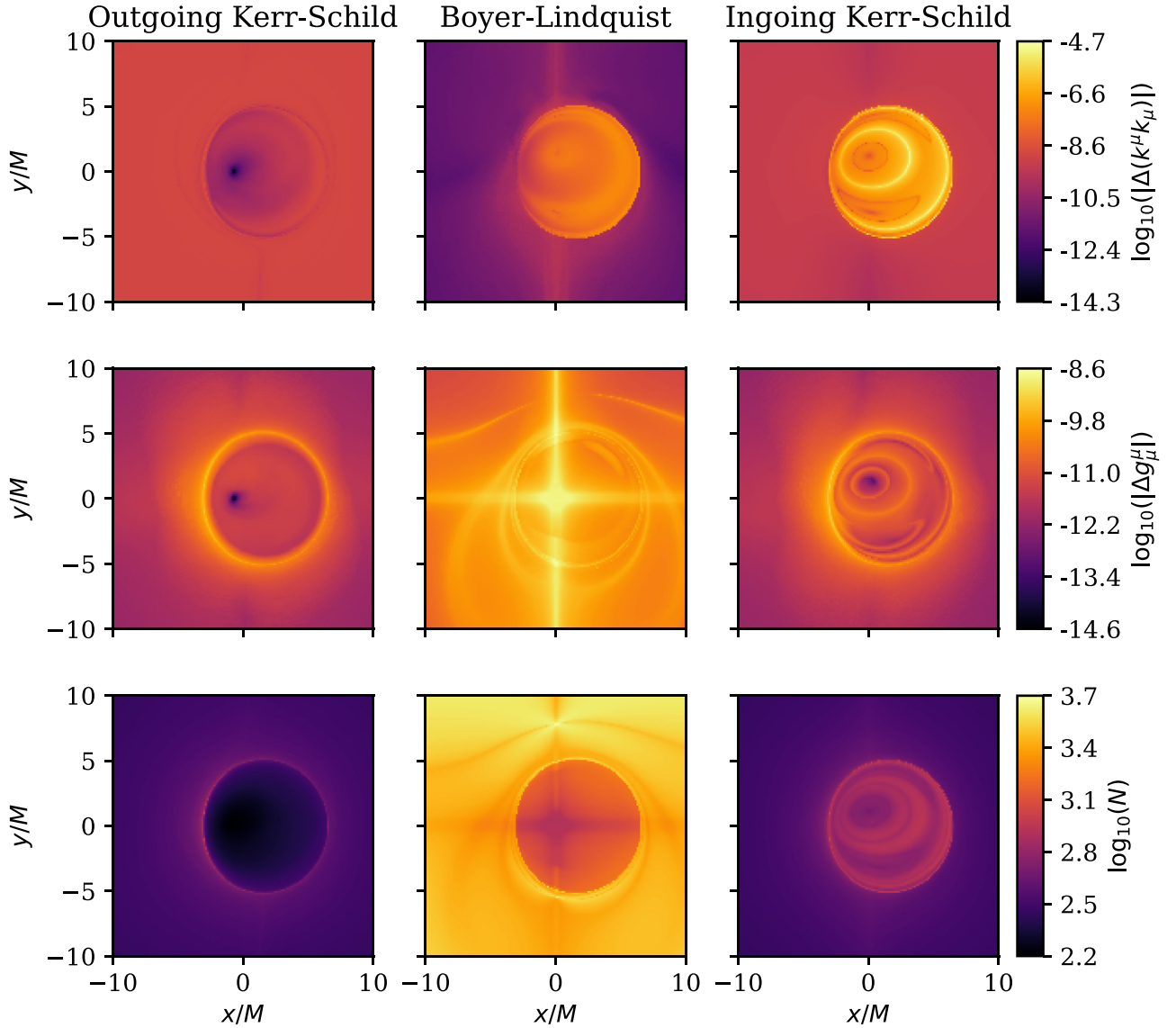


Figure 6. Maximum absolute errors and the number of computational steps taken along null geodesics computed in a Kerr spacetime with a dimensionless spin parameter $\chi = 0.95$. The errors are shown on an image plane situated at $r_0 = 10^5 M$ and an inclination of $\theta_0 = 50^\circ$, in Boyer–Lindquist coordinates. Geodesics were computed from $\lambda = 0$ to $\lambda = -2r_0$ or until intersection with a surface at $r = 1.03r_H$, where r_H is the Kerr event horizon radius. The computation was performed three times with the chart fixed to either the outgoing Kerr–Schild (left column) or Boyer–Lindquist (middle column) coordinates, or the ingoing Kerr–Schild coordinates (right column). Top row: maximum error in the value of the Hamiltonian $H = k_a k^a$, where k^a is the tangent of the geodesic. Middle row: maximum error in the trace g^a_a of the parallel transported metric tensor. Bottom row: number of steps, N , taken by the curve integration routine.

fact that the condition number (the ratio of the maximum to minimum singular value) of the matrix of the metric components scales as $\mathcal{O}(r^2)$ (see Section 3.5). In addition, the coordinates are singular at the poles. All of these factors combine to make the BL chart the most numerically disadvantageous of the three. Finally, the ingoing KS chart fares worse than the BL chart for the conservation of the Hamiltonian, but better for the trace of the metric tensor and number of steps taken. This is understandable, since these coordinates are adapted to radially ingoing null geodesics, and outgoing geodesics “wrap around” the black hole near the event horizon twice as fast compared to the BL coordinates. This is partly offset by the fact that the condition number of the metric components is better behaved than for the BL coordinates. The “wraparound” behavior is suppressed near the poles of the black hole, which in the figure can be seen as the slight decrease in the error of the metric trace around the

“north” pole of the black hole for the BL and the ingoing KS coordinates.

The accuracy in general is seen to be consistent with the given numerical tolerances. The outgoing KS chart in particular provides excellent accuracy, with results much better than even the set tolerances for nearly radial geodesics. In addition, there is a factor of ~ 10 difference in the number of steps taken between the outgoing KS chart and the BL chart, which was also directly reflected in the computational time. The results strongly suggest that the outgoing KS metric should be preferred in all codes computing mock observations using geodesics emanating from the vicinity of a Kerr black hole. Likewise, for studies of radiation scattering from a black hole, ingoing KS coordinates should be used for computing the incoming radiation and outgoing KS coordinates for the scattered, outgoing radiation.

5.2. Radiation Tests

We assessed the accuracy and convergence properties of the ARCMANCER radiative transfer functionality by post-processing a general relativistic magnetohydrodynamics (GRMHD) simulation and by comparing to an existing polarized radiative transfer code GRTRANS (Dexter 2016). To facilitate an easy comparison, we used the same simulation data used to test GRTRANS, as the data are conveniently distributed with the GRTRANS code.⁵ The simulation data used were computed with the GRMHD code HARM (Gammie et al. 2003; Noble et al. 2006) and describe an axisymmetric optically thin accretion flow around a Kerr black hole with a dimensionless angular momentum of $\chi = 0.9375$. The black hole mass M and its accretion rate \dot{M} were set to the GRTRANS defaults for HARM, $M = 4 \times 10^6 M_\odot$ and $\dot{M} = 1.57 \times 10^{15} \text{ g s}^{-1} = 2.49 \times 10^{-11} M_\odot \text{ yr}^{-1}$. Assuming in addition that the source is at a distance of $D = 8 \text{ kpc}$, these values approximate Sgr A* (Dexter 2016), although for this particular data set at the observed frequency of $\nu = 230 \text{ GHz}$ the total computed flux of $\sim 13 \text{ Jy}$ (see below) is roughly three times too large compared to the observed value of $\sim 3\text{--}4 \text{ Jy}$ (e.g., Bower et al. 2015). The radiative model was taken to be relativistic thermal synchrotron radiation, using the updated formulae in Dexter (2016). The electron and proton temperatures in the plasma were assumed equal, and the ideal gas equation of state was assumed, also corresponding to GRTRANS.

All mock observations were computed using a square image plane with physical dimensions $x, y \in [-L, L]$, where $L = 13 M$, in the local Lorentz frame of a stationary observer with $E_t = (1, 0, 0, 0)$ (see Section 3.6 and Appendix A.6.1). The negative z -axis is pointed toward the origin of the BL coordinates. The image plane was set at a distance $r = 10^4 M$ and an inclination of $\theta = 50^\circ$ (in BL coordinates), as in Dexter (2016). Geodesics from the image plane were then computed backwards in time from the image plane and the radiative transfer computed at the observed frequency of $\nu = 230 \text{ GHz}$ along these geodesics to form the final image. Numerical tolerances for both the geodesic computation and the radiative transfer computation were set to 10^{-10} . These tolerances guarantee that the accuracy during radiation transfer is governed by the chosen sampling rate $\Delta\lambda/\lambda_{\text{max}}$, where λ_{max} is the total range of the affine parameter over which the radiation transfer is computed. This ensures that the convergence and comparison results are not affected by the characteristics of the sampling induced by time-step control.

5.2.1. Flux Convergence

First, we investigated the convergence of the total flux in the Stokes variables $I_\nu = (I_\nu, Q_\nu, U_\nu, V_\nu)$ as the maximum step size $\Delta\lambda$ in the affine parameter and the image size P in pixels per side were varied. For each pixel i , we computed the observed flux

$$\mathbf{F}_{\nu,i} = \mathbf{I}_\nu(x_i, y_i) \frac{\Delta x \Delta y}{D^2}, \quad (19)$$

where $\Delta x = \Delta y = 2L/P$ is the physical size of each pixel and D is the (non-cosmological) distance of the target. From the

pixel-by-pixel fluxes, the total integrated fluxes $F_\nu = \sum_i \mathbf{F}_{\nu,i}$ were then computed.

The convergence results are shown in Figure 7. The general trend is that the benefits of a smaller step size saturate quickly for smaller-sized images, where the spatial sampling noise dominates. Similarly, increasing the image size is only effective up to the point where the noise from sampling of the small-scale structures starts to dominate. For this particular case, the benefit of increasing the image size beyond $P = 316$ pixels per side is already marginal. At this size, a 0.1% convergence is achieved at a maximum relative step size of $\Delta\lambda/\lambda_{\text{max}} = 3 \times 10^{-3}$.

5.2.2. Comparison to GRTRANS

In addition to ensuring the consistency and convergence of the ARCMANCER results, we performed a comparison to a publicly available radiative transfer code GRTRANS using the same HARM data set as above. Both codes were used to compute a square image 400 pixels wide, as above. GRTRANS was configured to take 2000 steps, which according to Dexter (2016) should net a relative accuracy for total flux at the 10^{-3} level. Similarly, ARCMANCER was constrained to take steps of at most $\Delta\lambda/\lambda_{\text{max}} = 3 \times 10^{-3}$, which should guarantee a relative accuracy of better than 10^{-3} by the convergence results above.

The resulting Stokes intensity maps computed with ARCMANCER are shown in Figure 8. In addition, Figure 9 shows the relative differences in the Stokes intensities as computed by ARCMANCER versus GRTRANS. In the left panel of Figure 9, we see that the unpolarized intensity predicted by ARCMANCER is consistently higher, and there is a clear difference in the polarized results, especially in the Q and U components.

The reason for this discrepancy was traced to two separate numerical issues. First, GRTRANS uses values for the gravitational constant G and Boltzmann constant k_B that were truncated to three significant figures, while in ARCMANCER the CODATA 2014 (Mohr et al. 2016) values are used up to the known experimental precision. Second, GRTRANS uses an approximation for computing the cylindrical Bessel functions used in the relativistic thermal synchrotron radiation model, Equations (B4) and (B14) in Dexter (2016). The GRTRANS code, as well as some other codes, such as that in Mościbrodzka & Gammie (2018), use first-order approximations for the cylindrical Bessel functions, but at least in this example case the approximations are not always valid throughout.

If the same physical constants and Bessel function approximations are used in ARCMANCER, the agreement both in unpolarized and polarized intensities is excellent, as can be seen in the right panel of Figure 9. The unpolarized and polarized total intensities agree with GRTRANS to a relative level of $\sim 10^{-3}$, and the pixel-by-pixel errors are below percent level on average. There are a small number of high difference outliers located either at regions where the absolute intensity values are very small or at the strongly lensed rings of emissivity. The former outliers are caused by numerical noise and the latter mainly by spatial sampling noise, since the small-scale structure in these rings is not resolved while simultaneously the emission is highly boosted, amplifying the differences. However, it can be seen from the results in Figure 7 and the numerical total fluxes tabulated in Table 1 that these pixels make no significant difference in the observed integrated fluxes.

⁵ The data is found in the file `dump040`, found online at <https://github.com/jadexter/grtrans/blob/master/dump040>.

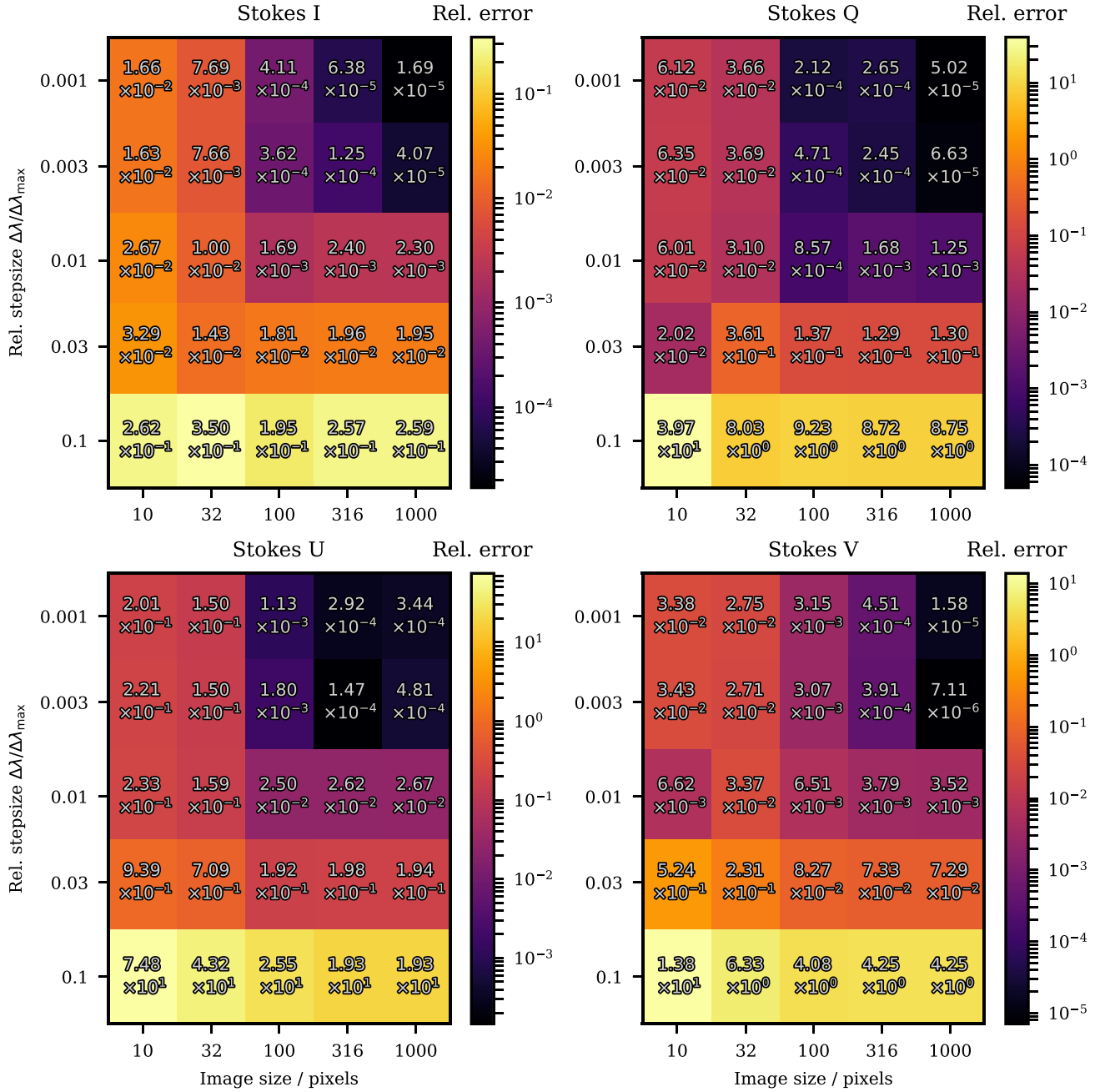


Figure 7. Convergence of total Stokes fluxes (see Equation (19)) at $\nu = 230 \times 10^9$ Hz over a mock observation image (see text) when the relative integration step size $\Delta\lambda/\lambda_{\max}$ and mock image size P (width and height, in pixels) are varied. All differences are relative to a data set computed with $P = 1500$ and $\Delta\lambda/\lambda_{\max} = 10^{-4}$. Differences are shown as density maps with the numerical value inset, with one map for each Stokes flux component, I , Q , U , and V .

For an interesting test case, we also ran the same test scenario with all polarization effects disabled. That is, we set $j_{\{Q,U,V\}} = \alpha_{\{Q,U,V\}} = r_{\{Q,U,V\}} = 0$ in \mathbf{J} and \mathbf{M} so that only the unpolarized degrees of freedom were propagated. As shown in Table 1, the resulting total flux is $\sim 13\%$ higher than in the polarized case. This suggests that creating mock observations of unpolarized flux can be misleading if polarization effects are completely ignored.

6. Applications

The capability of ARCMANCER to compute radiation transfer through an emitting and absorbing relativistic fluid (plasma) was showcased in the previous section. In the following, we

present further applications of ARCMANCER in different scenarios. The focus is on leveraging the capability of ARCMANCER to work with all kinds of emitting and absorbing surfaces, both moving and stationary.

6.1. Effects of Thin Accretion Disk Geometry

ARCMANCER makes it easy to compute mock observations of emitting surfaces with different user-defined geometries. Here, this feature is demonstrated through a toy model by computing the changes on the observed spectropolarimetric features caused by varying the opening half-angle β of a geometrically thin but optically thick accretion disk around a Kerr black hole.

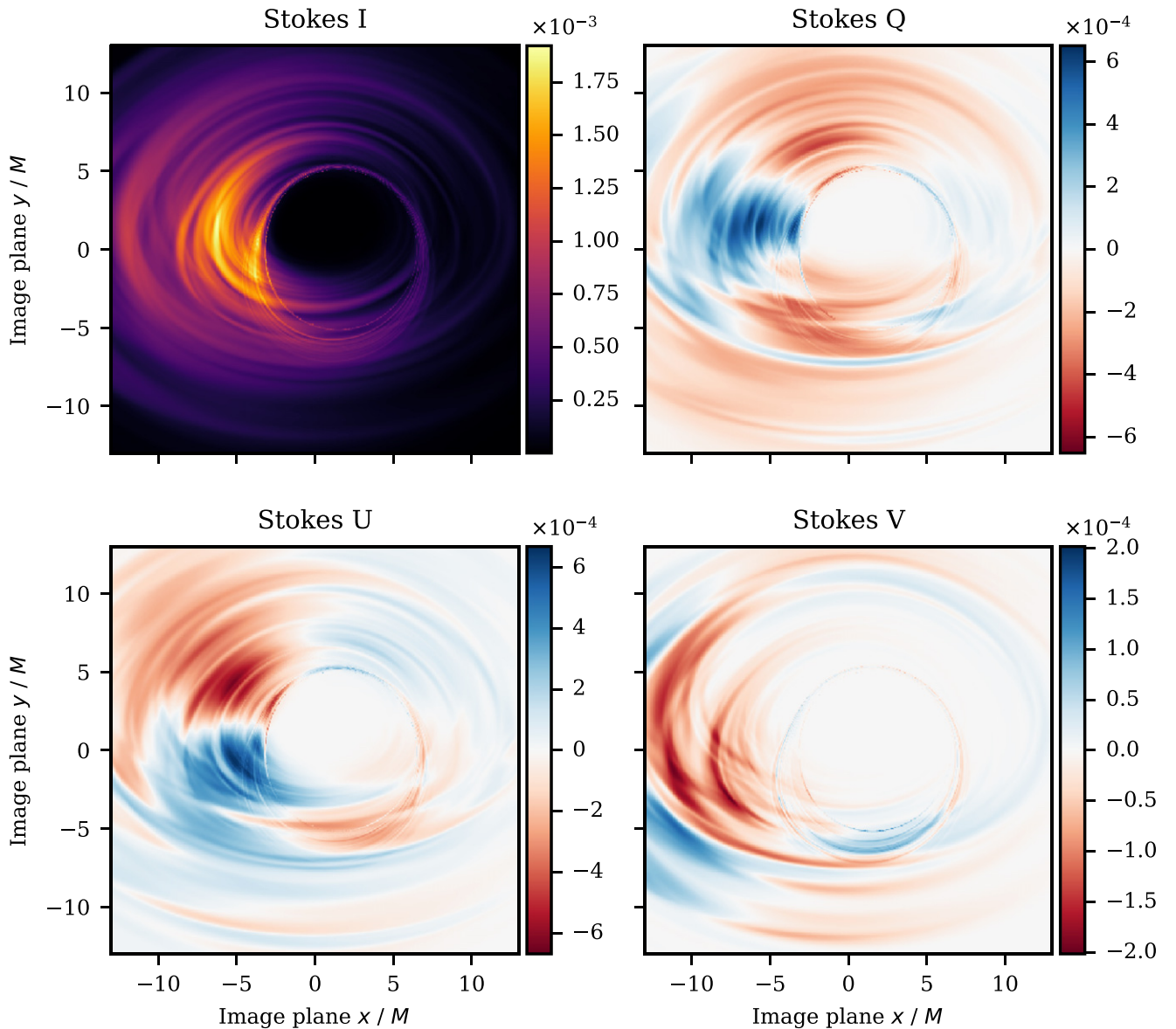


Figure 8. Observed specific Stokes intensities for an accretion flow around a rotating black hole with mass $M = 4 \times 10^6 M_\odot$, dimensionless spin parameter $\chi = 0.9375$, and accretion rate of $\dot{M} = 2.49 \times 10^{-11} M_\odot \text{ yr}^{-1}$. The mock observation was computed at $\nu = 230 \times 10^9 \text{ Hz}$, using ARCMANCER. The x and y axes are in units of GM/c^2 , whereas the intensity values are given in cgs units of $\text{erg s}^{-1} \text{ Hz}^{-1} \text{ cm}^{-2} \text{ sr}^{-1}$.

Often (e.g., Vincent et al. 2011; Bambi 2012; Psaltis & Johannsen 2012; Dexter 2016), a thin disk is modeled in mock observation simulations as an infinitely thin equatorial plane around the black hole. For α -disk models (Shakura & Sunyaev 1973; Novikov & Thorne 1973), this is in many cases a satisfactory approximation. This is because the maximal angle made by the disk photosphere and the symmetry plane is $\beta \sim \arctan(0.2\dot{m})$, computed in the Schwarzschild coordinates, where $\dot{m} = \dot{M}/\dot{M}_{\text{Edd}}$ is the black hole accretion rate in units of the Eddington accretion rate \dot{M}_{Edd} . However, for an accretion rate of $\dot{m} = 0.3$, this maximal angle is already $\beta \sim 4^\circ$, which can be expected to have observable consequences. This is since the maximal β in the Shakura–Sunyaev solution is found at $r = (27/2)M$, where M is the black hole mass, which is in the bright inner region of the disk.

Instead of the geometry of the α -disk model, which has a photospheric surface profile dependent on the accretion rate, we use a disk defined by a hyperbolic surface in the outgoing KS

coordinates,

$$S(t, x, y, z) = \frac{x^2 + y^2 - a^2}{r_{\text{min}}^2} - \frac{z^2}{r_{\text{min}}^2 \tan^2 \beta} - 1, \quad (20)$$

where $a = M\chi$ is the normalized angular momentum of the black hole, β is the half-opening angle of the hyperboloid, and r_{min} sets the inner boundary of the disk, here fixed to the innermost stable circular orbit (ISCO) of the black hole. The choice of this surface is motivated by the intention to investigate only the effects of geometry on the observable properties while keeping the emission properties of the disk otherwise fixed.

To compute the mock observation, we first set up an image plane with physical dimensions $x, y \in [-40M, 40M]$ at a distance of $r = 10^4 M$ (in BL coordinates). From this surface, null geodesics were propagated backwards until they intersected the disk surface or the event horizon. A

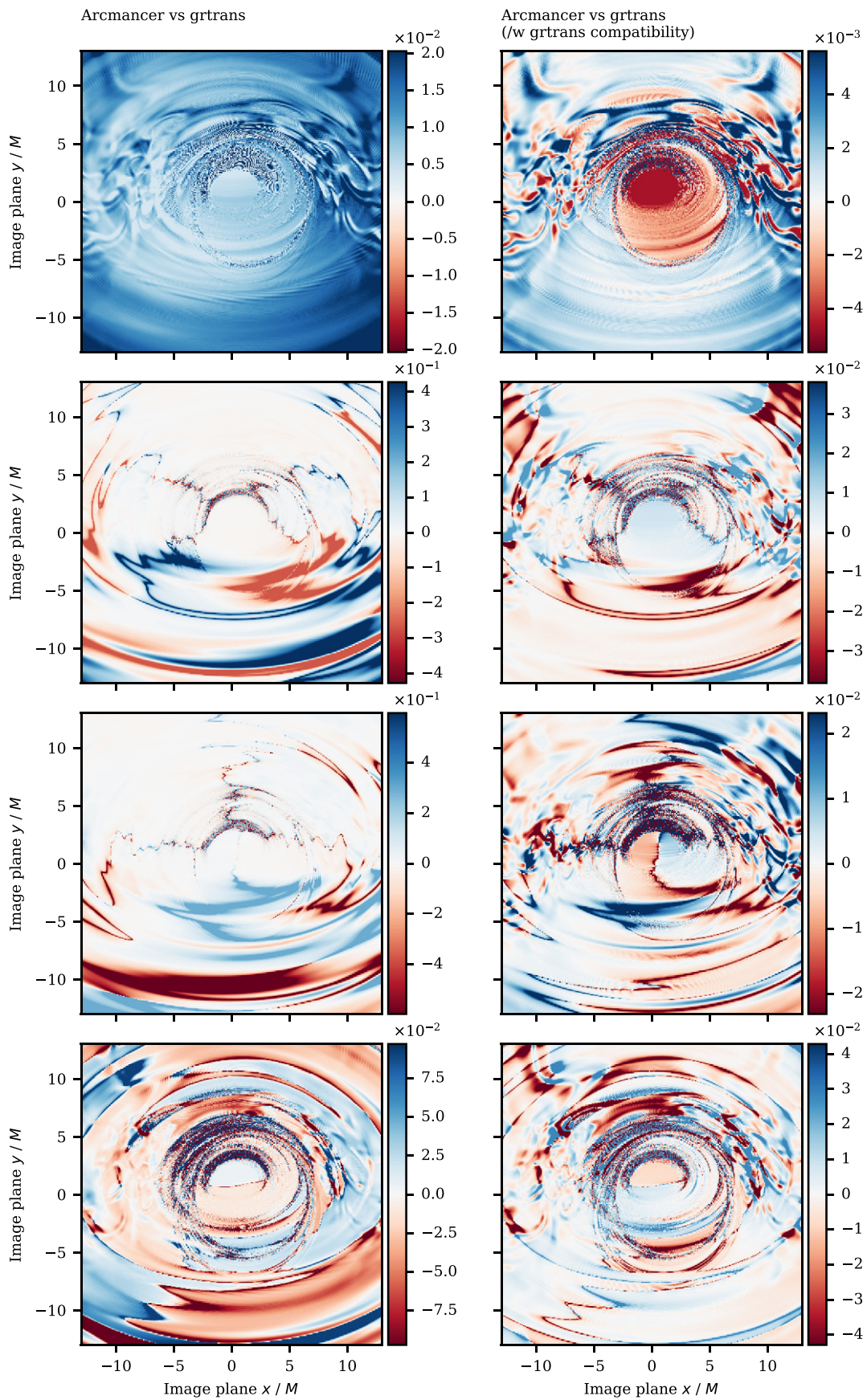


Figure 9. Left column: pixel-by-pixel relative difference between the Stokes intensities computed using ARCMANCER and GRTRANS, Winsorized to 95th percentile. ARCMANCER results computed using exact Bessel functions (see text). Right column: same as above, but using the GRTRANS approximations for the Bessel functions and fundamental constants in ARCMANCER. Note the very different color bar normalization between the left and right columns.

Table 1

Total Integrated Fluxes as Computed with ARCMANCER in the Test Scenario Depicted in Figures 8 and 9, Relative to the Values Obtained with GRTRANS

I	Q	U	V
ARCMANCER			
1.0083	0.98675	1.1671	1.0054
With GRTRANS compatibility			
1.0009	1.0029	0.99690	1.0023
With no polarization			
1.1264	0	0	0

PolarizationFrame was parallel transported with the geodesic to enable Faraday rotation effects to be captured. Points that hit the disk were given a blackbody spectrum $B_\nu(T)$ with a temperature T matching the Novikov–Thorne disk model (Novikov & Thorne 1973; Page & Thorne 1974), using mass $M = 10 M_\odot$, accretion rate $\dot{m} = 0.3 \dot{M}_{\text{Edd}}$, and dimensionless viscosity parameter $\alpha = 0.1$. The intensity and the linear polarization of the point were computed based on the electron scattering atmosphere model given in Chandrasekhar (1960), using the impact angle θ between the geodesic and the disk normal, computed in the rest frame of the rotating disk surface. The exact solution requires solving an integral equation. We instead used the Padé approximants

$$\frac{I_\nu(\mu)}{S_\nu} = \frac{1 + 2.3\mu - 0.3\mu^2}{2 \times 1.19167} \quad (21)$$

$$P = 0.117126 \frac{1 + 14.9165\mu - 15.8923\mu^2}{1 + 22.2420\mu + 44.8893\mu^2}, \quad (22)$$

where $\mu = \cos(\theta)$ for the intensity I_ν , normalized by the source function S_ν (in this example, $S_\nu = B_\nu(T)$), and the polarization fraction P , respectively. Both approximations are accurate to within 2% over the range $\mu \in [0, 1]$. It should be noted that the combination of a blackbody spectrum and a beamed intensity profile is not fully self-consistent, since a genuine blackbody emitter is isotropic. However, the combination serves to illustrate the effects of an anisotropically emitting surface. In addition, the spectral shape for thin accretion disks around stellar mass black holes is in any case well described using a diluted blackbody (Davis et al. 2005).

To construct the image from these data, instead of running full radiation transfer, the radiation was assumed to propagate in vacuum. This is not a particularly good assumption physically, since the thin disks are expected to have a tenuous, hot corona (e.g., Liang & Price 1977; Czerny & Elvis 1987), but it was made so as to not add additional uncertainties and keep the focus on the effects of changing disk geometry. The values of the intensity and polarization were directly transferred to the image plane after scaling the intensity by the redshift factor and rotating the polarization to match the rotation of the parallel transported PolarizationFrame. This computation was repeated for seven values of the disk opening angle from $\beta = 0^\circ.001$ to $\beta = 25^\circ$ and three observer inclination angles $i = 10^\circ$, 35° , and 60° . The results are collected in Figures 10 and 11, which show mock images of the two extreme cases ($\beta = 0^\circ.001$, $\beta = 25^\circ$) and the polarization spectra for all the computation runs.

The effect of the disk opening angle is clearly seen in Figure 10, which shows specific intensity maps as seen by an observer at an inclination of $i = 60^\circ$ for the extreme opening angles of $\beta = 0^\circ.001$ and $\beta = 25^\circ$. The intensity patterns differ significantly, with most of the emission coming from the opposite side of the disk for the disk with the larger opening angle. In addition, the structure of the ring caused by radiation that has traveled around the black hole once is noticeably changed by the increased disk thickness (cf. Luminet 1979). Despite the visual differences, Figure 11 shows that the shape of the spectra obtained from the integrated emission is hardly changed at all, and as such, the shape of the observed spectrum is not very sensitive to the disk geometry in this example.

Figure 10 also shows a significant difference in polarization patterns, with the large opening angle disk exhibiting a large asymmetry between the upper and lower halves of the mock observation image. This is caused by a purely geometrical effect, wherein the geodesics emanating from the opposite side of the disk from the observer’s point of view are more closely aligned with the local disk surface normal. For the geodesics coming from the observer’s side of the disk, the situation is the opposite. The polarization fraction of the electron scattering atmosphere model is strongly dependent on the angle of the geodesic with respect to the disk normal, with stronger polarization for lower incidence angles. The graphs of the degree of polarization,

$$P = \frac{\sqrt{Q^2 + U^2 + V^2}}{I}, \quad (23)$$

and the polarization angle,

$$\psi = \frac{1}{2} \arctan(U/Q), \quad (24)$$

in Figure 11 show that, unlike for intensity, the polarization asymmetry does not average out. Indeed, for the largest observer inclination (60°) shown in Figure 11, we see that there is a strong dependency of the degree of net polarization on the disk opening angle β . A similar but weaker effect is seen also for the observer inclinations $i = 10^\circ$ and $i = 35^\circ$. Figure 11 also shows the behavior of the net polarization angle ψ . With all observer inclinations, a similar behavior of rotation of the polarization angle at high photon energies is seen. However, for these model parameters, the rotation mainly occurs at the high energy end of the spectrum, where the exponential cutoff makes the effect hard to observe in practice.

The changes in polarization with observation frequency described above, for $\beta \sim 0$, are consistent with those of Schnittman & Krolik (2009), who studied an infinitely thin disk using a Monte Carlo approach. However, for a physically more realistic result, the accretion disk corona, as well as the radiation returning to and reflecting from the disk, needs to be taken into account, as in Schnittman & Krolik (2010). In addition, here we have shown that the geometry of the optically thick part of the disk cannot be ignored, which was an assumption in Schnittman & Krolik (2010). Combining the effects of the geometry with the effects of the corona and the returning radiation is straightforward using ARCMANCER and will be investigated in a future work.

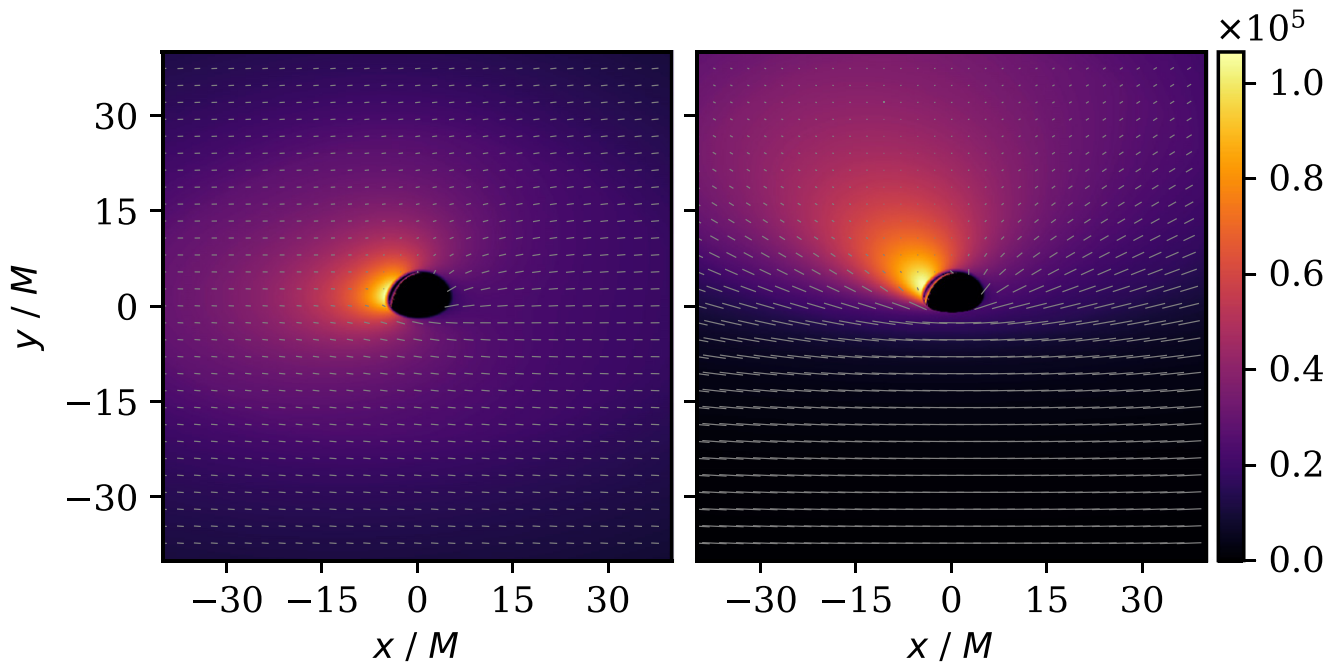


Figure 10. Specific intensity maps at $\nu = 8 \times 10^{17}$ Hz of a thin Novikov–Thorne model around a $10 M_{\odot}$ Kerr black hole with a dimensionless spin parameter $\chi = 0.7$. The disk model is computed with $\alpha = 0.1$ and $m = 0.3$, and the observer inclination is $i = 60^{\circ}$ from the disk symmetry axis. The disk opening angles are $\beta = 0.001$ and $\beta = 25^{\circ}$ for the left and the right panels, respectively. The direction of the observed linear polarization is shown by the gray lines, with the degree of linear polarization proportional to the length of the line. Both panels are computed with a resolution of 600 pixels per side. The intensities are in cgs units, i.e., $\text{erg s}^{-1} \text{cm}^{-2} \text{Hz}^{-1} \text{sr}^{-1}$.

6.2. Neutron Stars

Another natural application of user-definable surfaces is the imaging of neutron stars. A solid surface is an excellent approximation for the radiating atmosphere of a neutron star, since the atmospheric thickness is on the order of ~ 10 cm, whereas the radii of the neutron stars are in the range of ~ 10 km (see e.g., Potekhin 2014 for a review). Thus, terminating geodesics on the top of the atmosphere and using a separate atmospheric model to provide the (angle-dependent) specific intensity and polarization as initial conditions are attractive possibilities.

The use of a numerical geodesic propagation code such as ARCMANCER is further warranted due to the fact that a rotating neutron star is not exactly spherical but oblate, and the spacetime near the star cannot be exactly described by the Kerr metric (Stergioulas 2003; Bradley & Fodor 2009; Urbanec et al. 2013). Both complications are difficult to take into account when using fully analytic approaches, such as in, e.g., Pechenick et al. (1983), Strohmayer (1992), Miller & Lamb (1998), Poutanen & Gierliński (2003), and Lamb et al. (2009b), where the neutron star is modeled as a spherical surface in a Schwarzschild spacetime. The reason is twofold: the intersections of geodesics with the oblate surface are much more involved to compute (but not impossible; see Morsink et al. 2007; Lo et al. 2013; Miller & Lamb 2015; Stevens et al. 2016), and since Carter’s constant (Carter 1968) of the Kerr solution is not available, the geodesics themselves cannot be analytically solved even in quadrature. Another benefit of using a fully covariant approach throughout is that the pitfalls of trying to combine special relativistic and general relativistic effects separately in an ad hoc way (as done in e.g., Lo et al. 2013) are avoided. For example, see Nättilä & Pihajoki (2018) and Lo et al. (2018) for a thorough discussion of an error in the calculation of the observed flux in the ad hoc approach that has

gone undetected for years. Finally, incorporating polarization in an analytic geodesic propagator is only possible for Kerr (and Schwarzschild) spacetimes, but even then it is not trivial (see Viironen & Poutanen 2004; Dexter 2016). However, polarization data for this application are critical, since for small hot spots there is a severe degeneracy in the unpolarized pulse profile between the spot colatitude θ_s and the observer inclination i (Poutanen & Gierliński 2003).

In this section, we use ARCMANCER to assess the effects of the oblateness of the neutron star surface and the deviation of the neutron star spacetime from the simple Schwarzschild spacetime on the radiative transfer calculation. For this purpose, we use the AlGendy–Morsink (AGM) form of the Butterworth–Ipser spacetime (AlGendy & Morsink 2014; see also Appendix C.3). The AGM spacetime describes the surroundings of a rotating neutron star, taking into account the oblate shape of the star. The spacetime is parametrized by the dimensionless rotational parameter $\bar{\Omega} = \Omega R_e^{3/2} M^{-1/2}$ and the compactness parameter $x = M/R_e$, where Ω is the angular velocity of the rotation as seen by a distant observer, and M and R_e are the mass and the equatorial radius of the star, respectively. The oblate shape of the star is obtained from Equation (69).

As an example case, we studied a rotating neutron star with mass $M = 1.6 M_{\odot}$, equatorial radius $R_e = 12$ km, and rotational frequency of $\nu = 700$ Hz, with $\Omega = 2\pi\nu$. The high value of the spin was chosen to accentuate the effects of oblateness, yielding from Equation (70) a flattening of $f = 1 - R_p/R_e \sim 0.09$, where R_p and R_e are the polar and equatorial radii of the star, respectively. However, the high spin value is still within the observed range for neutron stars (Hessels et al. 2006). Similarly, the mass and the radius are well within the observed and inferred limits (Özel & Freire 2016; Steiner et al. 2016; Alsing et al. 2018). Using

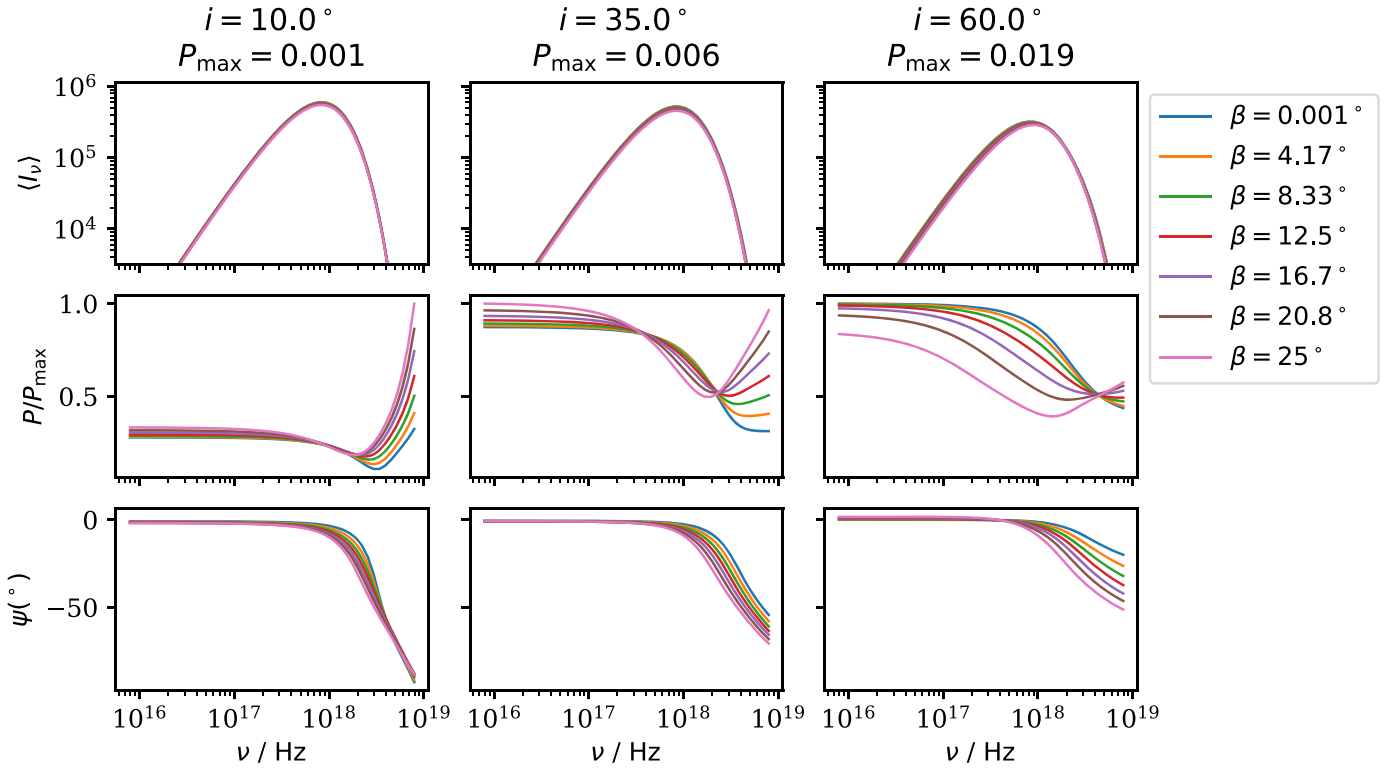


Figure 11. The mean intensity in cgs units (top panel), normalized degree of polarization (middle panel), and the polarization angle (bottom panel) as a function of frequency, obtained from the integrated Stokes intensities of mock images of a Novikov–Thorne disk around a Kerr black hole, with parameters as in Figure 10. The lines correspond to the different disk opening angles β , shown in the legend.

ARCMANCER, we computed surface maps of the flux and polarization characteristics, again assuming that the emission originates from an electron scattering atmosphere, using Equations (21) and (22). This is a good approximation for neutron stars where the emission originates from thermonuclear outbursts on the surface (see e.g., Suleimanov et al. 2011 and the references therein). However, we note that the results can be extrapolated on a more qualitative level to shock-heated accretion-powered hot spots as well (see, e.g., Basko & Sunyaev 1976; Lyubarskii & Syunyaev 1982; Viironen & Poutanen 2004). Otherwise, the radiation transfer is computed as in Section 6.1.

The computations were repeated three times: for an oblate star using the AGM spacetime (hereafter, AGM+Obl), for an oblate star using the Schwarzschild spacetime (Sch+Obl), and for a spherical star in the Schwarzschild spacetime (Sch+Sph). The results are shown in Figures 12–15. Figure 12 shows the behavior of I_ν/S_ν , the specific intensity divided by the source function, and the polarization over the star surface, computed using the AGM spacetime at observer inclinations of $i = 20^\circ$, 50° , and 90° . The combination of Doppler boosting and the strong angular dependence of the electron scattering atmosphere yields an intensity that varies significantly over the neutron star surface. The net polarization is high only near the edges, where the impact angle is large. Figure 12 also shows two possible paths of constant colatitude hot spots, assuming that the star is rotating around the vertical axis. From the figure, it is then easy to appreciate that a rotating hot spot should exhibit a large periodic variation in the observed polarization angle. This variation can be directly seen in Figure 15, which is consistent with the results in Viironen & Poutanen (2004).

Figure 13 shows the difference in normalized intensity I_ν/S_ν , degree of polarization, and polarization angle when the computation is performed using the AGM metric versus the Schwarzschild metric (i.e., AGM+Obl versus Sch+Obl). The effects of the rotation become significant only near the star, and consequently, the differences stay moderate for the most part, below $\sim 10\%$ for the intensity and below $\sim 2\%$ for the degree of polarization. There are areas of larger differences, but these are concentrated on the edges of the visible disk of the neutron star, and their total area is small. The differences in the polarization angle are larger, around $\sim 20^\circ$ overall. There are very large differences near the point where the radiation was emitted toward the zenith in the frame of the neutron star surface, but this area corresponds to vanishing polarization, and as such, these differences are unobservable.

In contrast, Figure 14 displays the same differences but between computations performed using an oblate star versus a spherical star, both in Schwarzschild spacetimes (i.e., Sch+Obl versus Sch+Sph). The spherical star was given a radius equal to the equatorial radius of the oblate star. In this case, the differences in all quantities are much more pronounced. This is not a surprise, since a change in the shape of the star affects the redshift distribution on the surface due to variations in local surface gravity. These differences become even more evident when one looks at Figure 15, which shows two examples of light curves and the time-varying degree of polarization and polarization angle for a rotating hot spot. First, the pulse and polarization profiles closely match those obtained by Viironen & Poutanen (2004) for the Sch+Sph case and confirm that the observational degeneracy in unpolarized flux between observer inclination and spot colatitude is lifted by the polarization measurements. However, from the figure, it can be seen that the

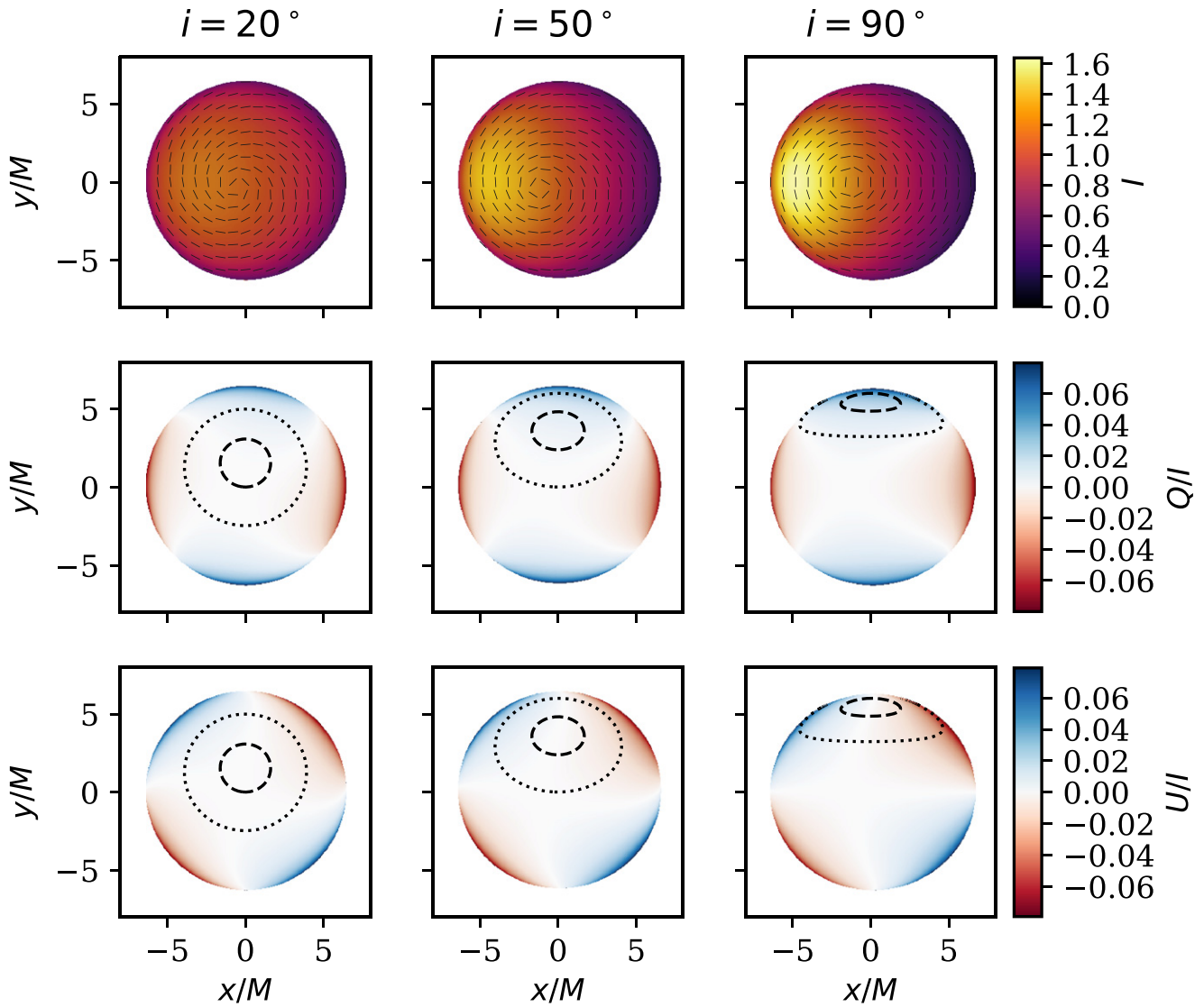


Figure 12. Surface maps of an oblate rotating neutron star with an equatorial radius $R_e = 12$ km, mass $M = 1.6 M_\odot$, and rotational frequency $\nu = 700$ Hz, computed with ARCMANCER. The top row shows I_ν/S_ν , or the specific intensity normalized with the source function (see Equation (21)) as a color map, while the direction of linear polarization is indicated by black lines. The middle and bottom rows show the linear polarization fractions Q/I and U/I , respectively. The dashed and dotted lines indicate contours of constant colatitudes of 20° and 50° , respectively (cf. Figure 15). Columns from left to right show results with observer inclinations of $i = 20^\circ$, 50° , and 90° with respect to the rotational axis of the star.

approximation of a spherical star produces results that differ significantly both in intensity and polarization properties from the result obtained using an oblate surface. In addition, there is a small but non-negligible difference between the results obtained using the AGM metric versus a plain Schwarzschild metric. Similar results for the unpolarized flux were obtained already in Psaltis & Özel (2014), although for an isotropically emitting atmosphere.

Based on our preliminary study, we can conclude that the error introduced when computing the polarization angle with the Schwarzschild spacetime approximation is largest when both the observer inclination and the spot colatitude are small. Likewise, the error in the degree of polarization is largest when the spot is near the equator, i.e., spot colatitude is close to $\sim 90^\circ$. We conclude that to obtain polarized pulse profiles that are accurate to below the $\sim 1\%$ level, it is necessary that the rotation and the geometric shape of the star are both accurately modeled. In practice this means that the analytic results based

on the Schwarzschild spacetime such as in, e.g., Weinberg et al. (2001), Viironen & Poutanen (2004), Lamb et al. (2009a), Lo et al. (2013), and Miller & Lamb (2015) should be used with caution. However, to actually reach the $\sim 1\%$ level of accuracy, other systematic errors in, e.g., modeling the emission from the neutron star and its surrounding environment would also need to be resolved.

6.3. Binary Black Holes

To further explore the possibility of using arbitrary metrics and multiple surfaces that may also move, we consider a toy model of an accreting black hole with a secondary black hole companion. To set up the problem, we use an approximative metric, constructed using the outgoing KS form of the Kerr metric, Equations (57) and (60). In the limit of zero spin, $a = 0$, the metric is

$$g_{ab} = \eta_{ab} + F(M, r)l_a(\mathbf{x})l_b(\mathbf{x}), \quad (25)$$

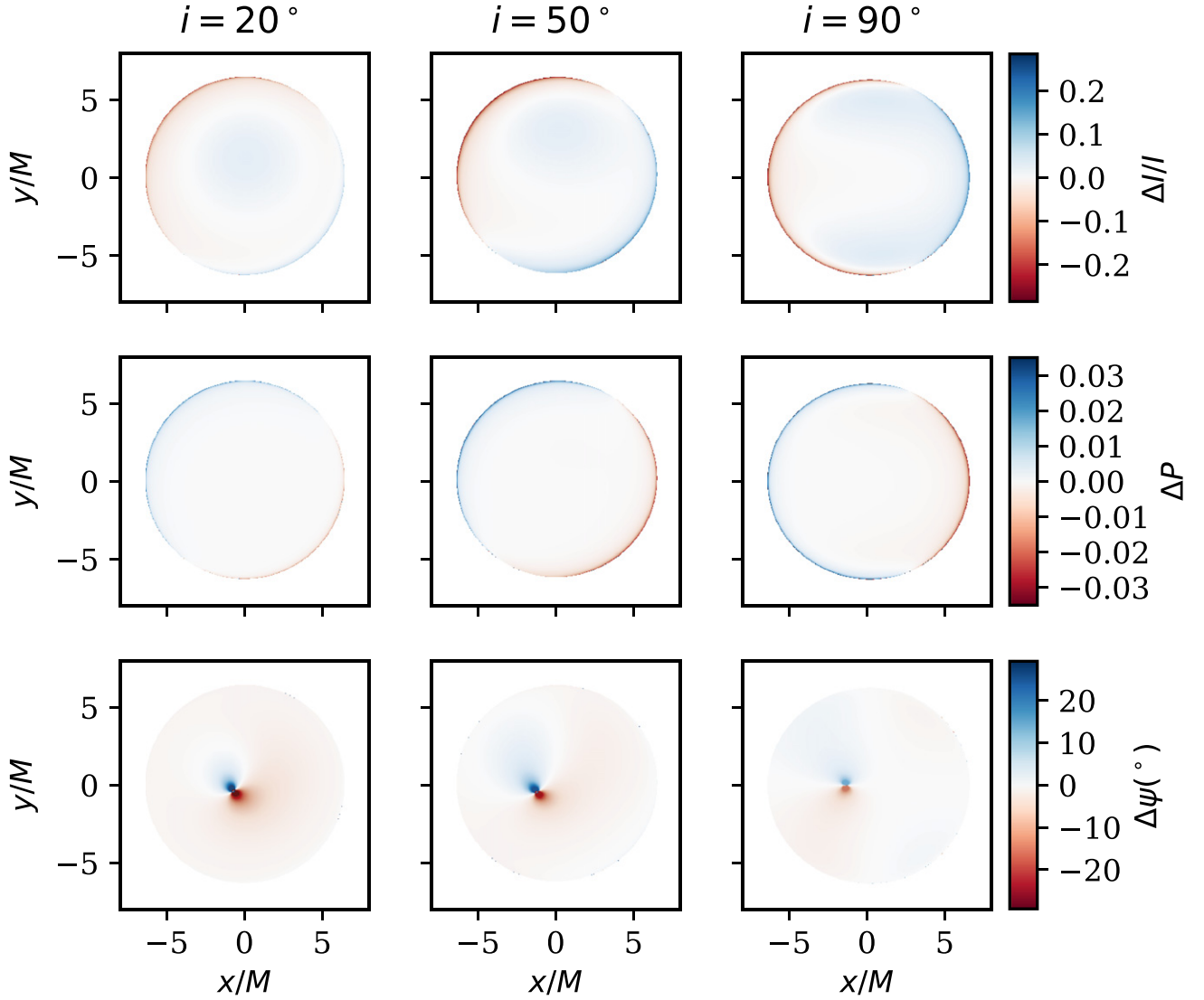


Figure 13. Surface maps for a rotating neutron star with $R_e = 12$ km, $M = 1.6 M_\odot$, and $\nu = 700$ Hz, computed with ARCMANCER. The maps indicate relative differences between a solution using the AlGendy–Morsink spacetime vs. using the Schwarzschild spacetime. The same oblate shape of the neutron star is used for both metrics. The rows show the relative difference in normalized intensity (see Equation (21)) I_ν/S_ν (top row), degree of polarization P (middle row), and polarization angle ψ (bottom row). Columns from left to right show results with observer inclinations of $i = 10^\circ$, 50° , and 90° .

where M is the mass of the black hole, $\mathbf{x} = (x, y, z)$,

$$F(M, r) = -\frac{2M}{r}, \quad (26)$$

$$l_a(\mathbf{x}) = \left(-1, \frac{x}{r}, \frac{y}{r}, \frac{z}{r}\right), \quad (27)$$

and $r^2 = x^2 + y^2 + z^2$. To this form we add a perturbation representing a distant second black hole moving at a slow coordinate velocity. Taking the spatial position of the second black hole to be a function $\mathbf{x}_2(t)$ of the coordinate time, we set

$$g_{ab} = \eta_{ab} + F(M_1, r)l_a(\mathbf{x})l_b(\mathbf{x}) + F(M_2, r_2)l_a(\mathbf{x} - \mathbf{x}_2)l_b(\mathbf{x} - \mathbf{x}_2), \quad (28)$$

where M_1 and M_2 are the masses of the primary and secondary black holes, respectively, \mathbf{x}_2 is the spatial position of the secondary black hole, and $r_2^2 = (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2$.

The metric (28) is not a solution of the vacuum Einstein field equations, for which no exact dynamic binary black hole solution is known.⁶ For example, the metric (28) does not contain the gravitational wave component expected from the motion of multiple gravitating bodies. However, in the limit $M_2 \sim 0$ and $dx_2(t)/dt \sim 0$ for all t , the perturbation caused by the secondary is small and remains small, and the gravitational wave component is negligible, and in this sense the approximation is reasonable. For black hole binary systems with smaller separations and larger velocities, a discretized metric from a full GR simulation should be used with ARCMANCER. More accurate analytical approximations, such as from Mundim et al. (2014), can also yield satisfactory accuracy

⁶ However, there are a number of known static solutions for multiple black holes. Examples include any number of Schwarzschild black holes in a collinear configuration (Israel & Khan 1964) or any number of maximally charged Reissner–Nordström black holes in any configuration (Papapetrou 1945; Majumdar 1947).

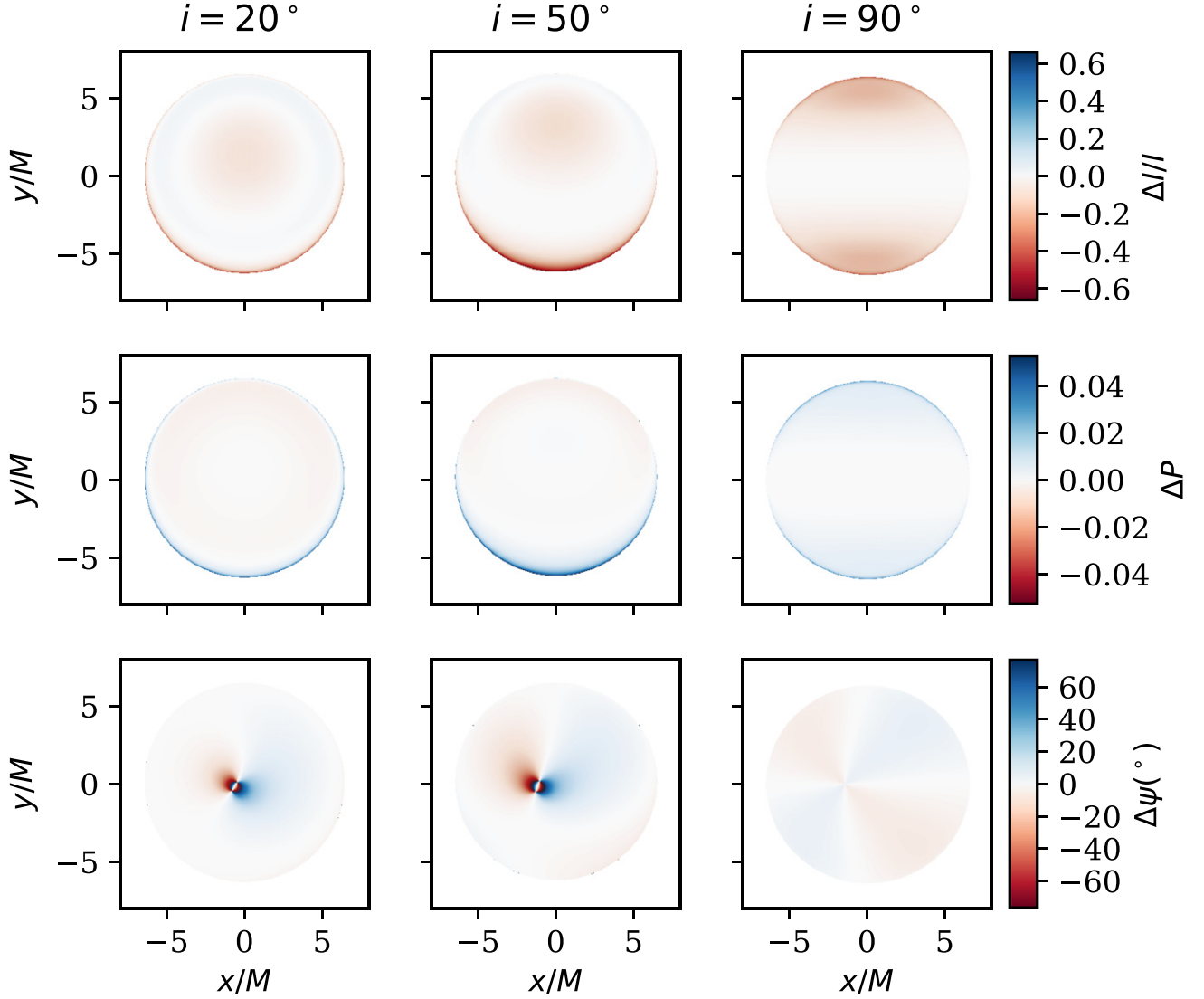


Figure 14. Same as Figure 13, but showing the relative differences between an oblate neutron star surface, corresponding to the rotational rate of $\nu = 700$ Hz, and a spherical surface, both in the Schwarzschild spacetime. The oblate surface has a flattening of $f \sim 0.09$, so that the polar radius is $\sim 91\%$ of the equatorial radius. The spherical surface has a radius equal to the equatorial radius of the oblate surface.

(Sadiq et al. 2018), but the approximative analytical metrics are on the other hand algebraically complex.

We set $M_1 = 5 \times 10^6 M_\odot$, representing an SMBH, and $M_2 = 0.05M_1$, which falls in the intermediate mass black hole (IMBH) range. Otherwise, we set up the system as in Section 6.1, by placing an image plane with physical dimensions $[-50M_1, 50M_1]^2$ at $R = 10^7 M_1$ at an observer inclination of $i = 60^\circ$. The secondary black hole is set on a rectilinear coordinate path $\mathbf{x}_2(t) = \mathbf{x}_{2,0} + t\mathbf{v}_2$, where

$$\begin{aligned} \mathbf{x}_{2,0} = & (L \sin i - \delta \cos i \cos \phi_v, -\delta \sin \phi_v, \\ & \times L \cos i + \delta \sin i \cos \phi_v), \end{aligned} \quad (29)$$

$$\mathbf{v}_2 = v_2(-\cos i \sin \phi_v, \cos \phi_v, \sin i \sin \phi_v). \quad (30)$$

Here, δ is the apparent offset of the secondary's path, $L = \sqrt{r_{2,0}^2 - \delta^2}$ is the orthogonal distance from the primary to the image plane, $r_{2,0}$ is the minimum distance between the black holes, v_2 is the velocity of the secondary, and ϕ_v is the angle between the path of the secondary and the image plane

x -axis. For this particular example, we set $r_{2,0} = 10^3 M_1$, $\delta = -4M_1$, $\phi_v = 25^\circ$, and $\nu = 3.162 \times 10^{-2}$. These initial conditions approximately correspond to an IMBH on a circular orbit around an SMBH, a situation that could possibly follow a merger of a more massive galaxy with a dwarf galaxy (Graham & Scott 2013). In order to have something to make a mock observation of, the primary black hole was given an infinitely thin Novikov–Thorne accretion disk, with $\alpha = 0.1$ and an accretion rate $\dot{m} = 0.01$ in units of the Eddington accretion rate. Geodesics were then propagated backwards in time from the image plane starting at 150 different values of the coordinate time, evenly distributed in $[-3000M_1, 3000M_1]$. For each set of geodesics, mock images, integrated fluxes, and polarization fraction and angle were computed.

The resulting light curves are shown in Figure 17, with selected resolved frames shown in Figure 16. The main effect of the passing secondary is a strong enhancement by a factor of ~ 2 of the observed flux from the accretion disk of the primary, caused by gravitational lensing. The flux curve has a clearly

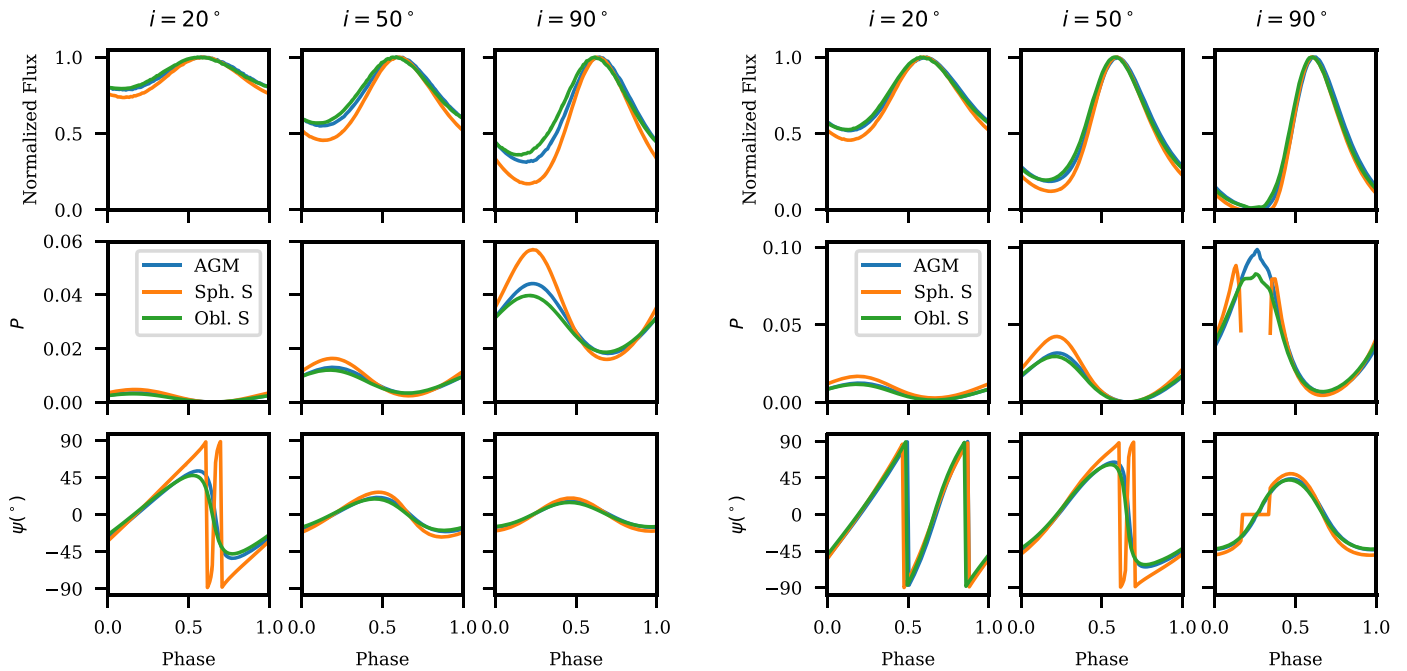


Figure 15. Pulse profiles for thermonuclear-powered hot spots rotating with the surface of a neutron star with $R_e = 12$ km, $M = 1.6 M_\odot$, and $\nu = 700$ Hz. The spot has an angular radius of 5° and a constant colatitude of either 20° (left panel) or 50° (right panel). In both panels, the top row shows the integrated flux normalized to the maximum value, assuming a constant spot temperature. The middle and bottom rows show the integrated polarization fraction and polarization angle, respectively. Three cases are shown: oblate star in the AGM spacetime (blue curve), oblate star in the Schwarzschild spacetime (green curve), and spherical star in the Schwarzschild spacetime (orange curve).

non-sinusoidal shape, where, after the main peak, there is a pronounced shoulder. The double-peaked structure results from the lensing of the two main visible arcs of the primary accretion disk. The difference is that the major peak has a larger contribution from the Doppler-boosted side of the primary accretion disk. This asymmetry is also clearly visible in the curves for polarization fraction P and angle ψ ; see Equations (23) and (24). The polarization fraction curve shows a clear two-peaked shape, with a sharp peak followed by a sharp trough. The polarization angle mirrors this behavior, with a maximum rotation of $\sim 7^\circ$.

The light curves were also computed with a smaller value of observer inclination of $i = 5^\circ$, also shown in Figure 17. The results show that the double-peaked structure of the light curve is more evident toward $i = 0^\circ$, whereas the relative amount of polarized flux becomes significantly smaller. Both effects are to be expected considering the increased symmetry when $i \rightarrow 0^\circ$. The changes in the degree of polarization and the polarization angle are more pronounced as well, but due to the negligible relative amount of polarized flux, these are unlikely to be detectable.

Over longer timescales, the recurrent lensing by the secondary produces a periodic signal, which can be clearly observable over the baseline brightness of the primary accretion disk, as seen from Figure 17. However, the signal is strongly non-sinusoidal, which may reduce observability in periodicity searches based on periodogram techniques. On the other hand, if a series of accretion disk lensing events was observed, it should be possible to use lensing mock observation simulations to obtain independent constraints on the secondary black hole mass and the orbital parameters.

Finally, we note the interesting fact that the double-peaked light curve is reminiscent of the light curve of the periodic binary blazar OJ 287, which exhibits a long succession of

strongly non-sinusoidal double-peaked outbursts every ~ 12 years (Sillanpaa et al. 1988; Valtonen et al. 2008). Many different physical mechanisms for the outbursts have been proposed, such as tidally enhanced accretion rate (Sillanpaa et al. 1988), accretion disk impacts (Lehto & Valtonen 1996; Pihajoki 2016), and changes in the relativistic jet geometry (Katz 1997; Villata et al. 1998). Accretion disk lensing adds yet another possible outburst mechanism.

7. Conclusions

In this paper, we have presented ARCMANCER, a C++/Python library for the numerical computation of curves and tensor algebra in arbitrary Riemannian and semi-Riemannian spaces. The library is designed to be easy to extend as well as to incorporate in new or existing applications. ARCMANCER offers several novel and useful features. Many of these are built around ARCMANCER's seamless support of multiple simultaneous coordinate charts. For example, ARCMANCER offers automatic conversion of coordinates and tensors of arbitrary rank between different charts. This conversion works even in the case where no explicit transformation is provided between two given charts, as long as the graph formed by all of the available charts and transformations contains a path connecting the two charts. The coordinate chart support is also used in the library to automatically pick the numerically most appropriate chart to integrate the equations of motion for curves. ARCMANCER can also be used for numerical tensor algebra, supporting all usual tensor operations for tensors of arbitrary rank and dimension. In addition, ARCMANCER can parallel propagate arbitrary tensors and user-defined quantities along curves. In the four-dimensional case, the ARCMANCER library contains a suite of tools designed for solving problems of general relativistic radiative transfer using the ray-tracing approach. These include

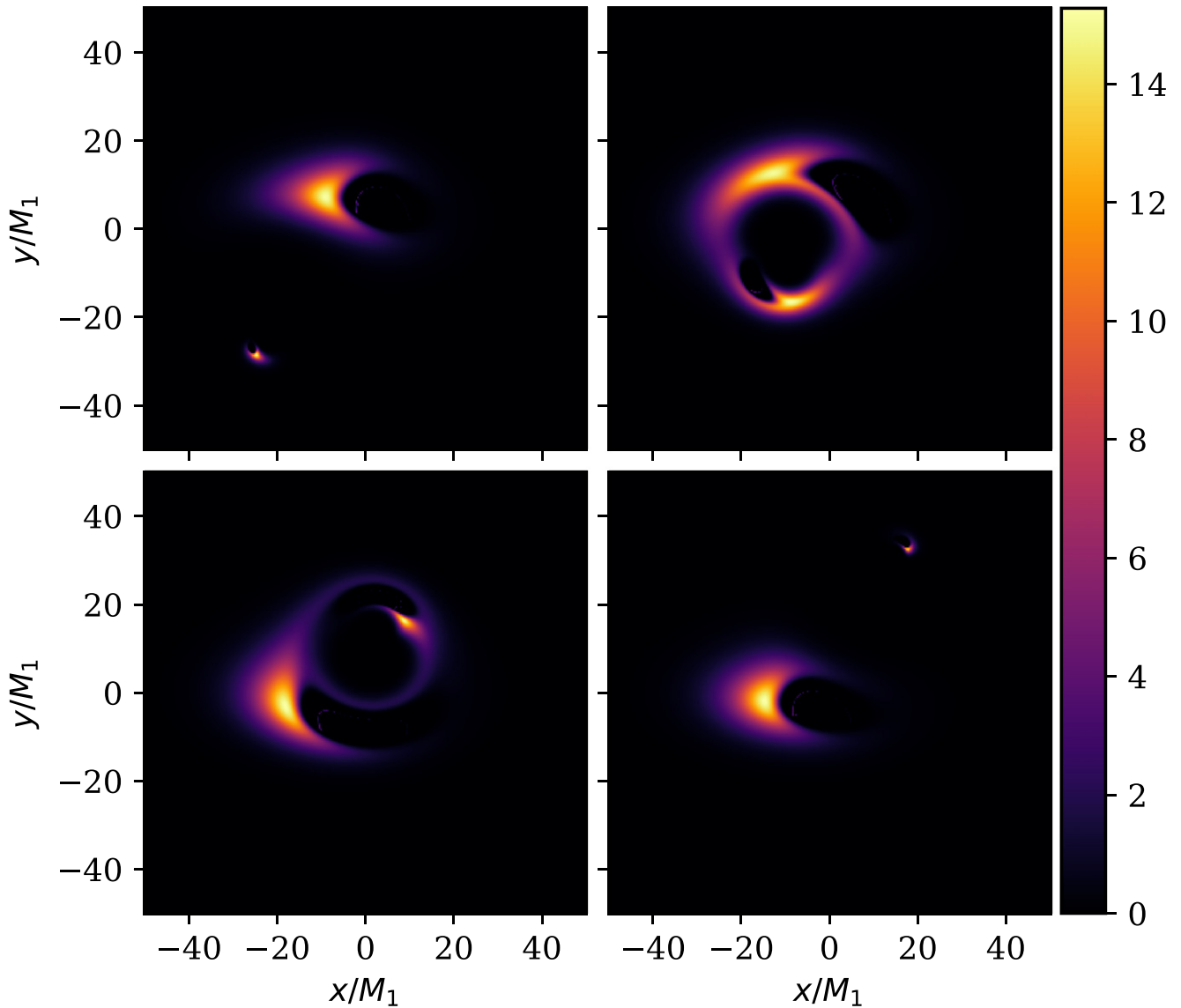


Figure 16. Resolved mock observations at $\nu = 5 \times 10^{16}$ Hz of a simulated accreting binary black hole system with a primary of mass $M_1 = 5 \times 10^6 M_\odot$ and a secondary of mass $M_2 = 0.05M_1$ (see the text for orbital parameters). The primary has a geometrically thin and optically thick Novikov–Thorne accretion disk. The images show intensity maps of the unpolarized Stokes I from the accretion disk, with the coordinate time increasing from left to right and top to bottom. The x and y axes are in units of GM_1/c^2 , and the intensities are in cgs units.

a coordinate-invariant method for generating image planes, easy interface for supplying user-defined fluid and radiation models, and a support for geometric objects, which can be used, for example, to model radiating surfaces or define limits of computational domains. For convenience, the library can also work with a Lorentzian metric signature. All of these features are thoroughly documented in the code itself, in the documentation automatically generated from the code and via several example applications provided with the library. In addition, the library Web site⁷ provides instructions for installation and getting started.

In this presentation of the ARCMANCER library, we have included a description of the internal workings of the code, as well as numerous tests of the accuracy of the code. The ARCMANCER code was found to fulfill theoretically expected convergence properties. It also produced very similar results to

an existing ray-tracing code, GRTRANS, when applied to a demanding mock observation scenario of a hot accretion flow around a Kerr black hole. Notably, the code tests demonstrated the critical importance of choosing the right coordinate system for the chosen problem and the necessity of being able to change coordinate systems during the numerical evolution of the problem.

The code tests were followed by applications of the code to a variety of astrophysical scenarios, showcasing the flexibility of the ARCMANCER code. The first example application was an investigation of the effect of the opening angle of an optically thick but geometrically thin accretion disk to its observable properties. While the unpolarized flux was essentially invariant with respect to the disk opening angle, the degree of polarization and angle of polarization were found to significantly depend on it.

In the next application, ARCMANCER was used to study observational properties of hot spots on rotating neutron stars. We compared three different commonly used models. In the

⁷ <https://bitbucket.org/popuha/arcmanecer>

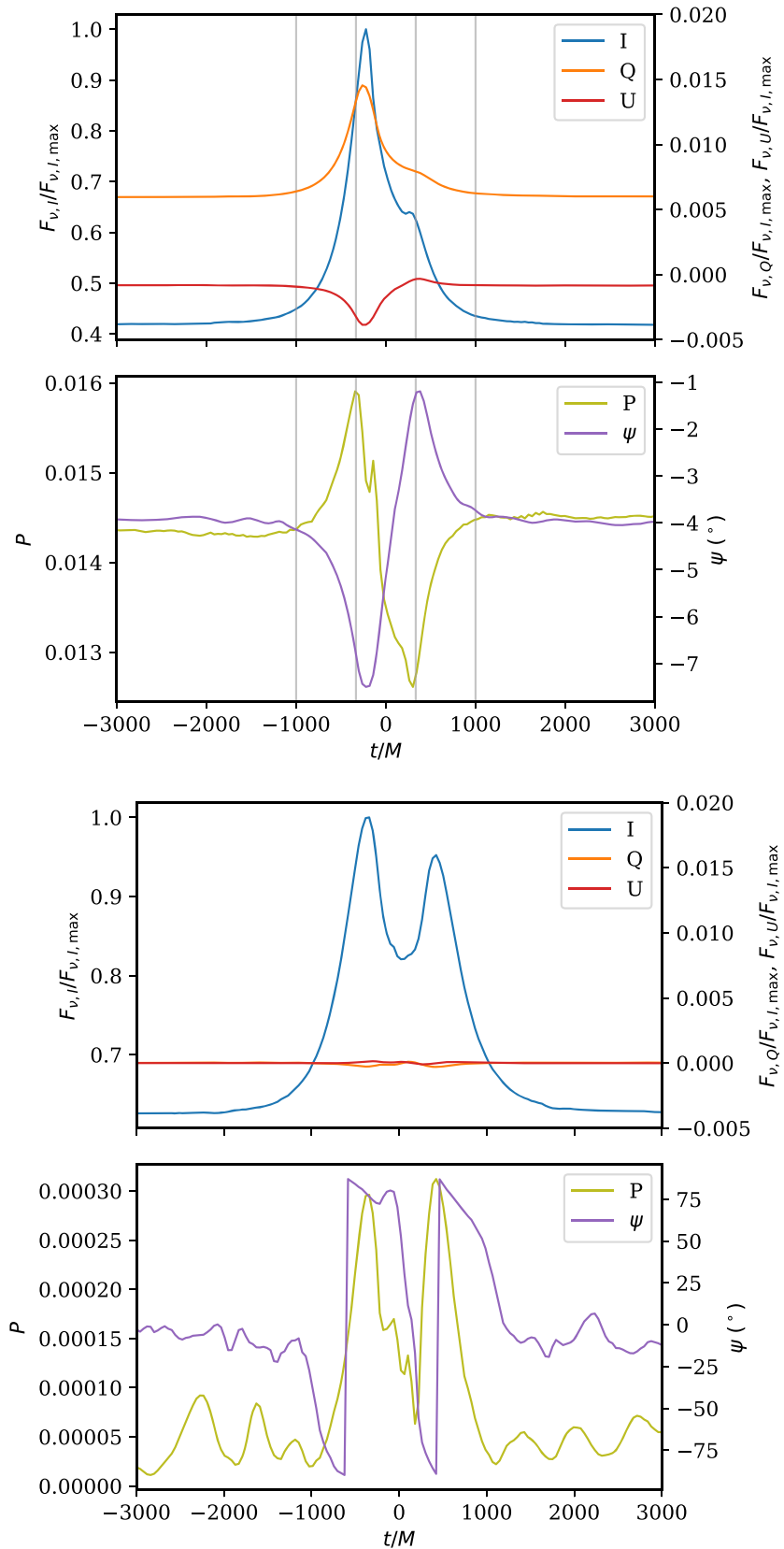


Figure 17. Polarized light curves at $\nu = 5 \times 10^{16}$ Hz of a simulated accreting binary black hole system (see text) for observer inclinations of $i = 60^\circ$ (top two panels) and 5° (bottom two panels). The horizontal time axis is shown in units of GM_1/c^3 . Gray vertical lines indicate the times of the resolved frames shown in Figure 16. Top and third panels show the integrated polarized fluxes of Stokes components I , Q , and U relative to the respective maximum flux, as a function of coordinate time. Second and bottom panels show the polarization fraction P and polarization angle ψ as a function of coordinate time. For the bottom panel, the Q and U curves have been smoothed with a nine-point second-order Savitzky–Golay filter (Savitzky & Golay 1964) to reduce the numerical noise in P and ψ curves caused by the very small net polarization.

AGM+Obl model, the neutron star surface was modeled using a physical oblate shape with an exterior spacetime metric that took the oblate shape and rotation into account. This physically accurate model was compared with two more approximate models: Sch+Obl, in which the exterior metric was changed to a Schwarzschild metric, and Sch+Sph, in which in addition the shape of the neutron star was taken to be spherical. Our results show that the oblate shape of the star makes a large contribution to the shape of both polarized and unpolarized flux curves and must be taken into account. However, we also find that in order to obtain polarized light curves with accuracies better than the $\sim 1\%$ level, the Schwarzschild metric must be abandoned in favor of more physically motivated alternatives.

Finally, we used ARCMANCER to create mock observations of an accreting binary black hole system, consisting of a primary black hole with an accretion disk, together with an orbiting secondary black hole. The application demonstrated how ARCMANCER can easily handle a more complex geometry where light rays can be terminated on multiple surfaces (two event horizons and one accretion disk), some of which may move (the secondary event horizon). We found that the lensing caused by the orbiting secondary can produce clearly observable changes in the observed polarized and unpolarized flux from the accretion disk of the primary black hole. However, the changes in polarized flux are strongly dependent on the observer inclination due to the geometry of the simple α -disk model we used.

In the future, we expect to use the ARCMANCER library to build a comprehensive radiative transfer application for investigating complex accretion flows around compact objects. In addition, the capabilities of the library itself will be extended. Planned features include built-in support for outputs of other GRMHD codes besides HARM, support for easy serialization of the code data structures, and more built-in radiation and fluid models. The coordinate chart system will also be enhanced with support for defining domains for the charts and improving the numerical behavior of chart-dependent operations such as curve interpolation.

Ray tracing is expected to become even more important in the future, driven by the increase in observational capabilities, especially with respect to polarized light, together with the ongoing prodigious increase in computational resources. We are confident that ARCMANCER will prove to be a highly useful and adaptable tool in this upcoming era.

P.P., M.M., and P.H.J. acknowledge support from the Academy of Finland, grant No. 274931. This research has made use of NASA's Astrophysics Data System.

Software: Eigen (<http://eigen.tuxfamily.org/>), Boost (<http://www.boost.org/>) Pybind11 (<https://github.com/pybind/pybind11>), Numpy (van der Walt et al. 2011), Scipy (<https://www.scipy.org/>), Matplotlib (Hunter 2007).

Appendix A Differential Geometry and General Relativity

The ARCMANCER library has capabilities beyond ray tracing and radiative transfer in four-dimensional Lorentzian spacetimes. The library offers a variety of tools for computational differential geometry in Riemannian or semi-Riemannian manifolds of arbitrary dimension, within the constraints of available memory and computing power. In the following, we

give a short, self-contained review of the concepts of differential geometry that are implemented and used in the ARCMANCER library. For practical reasons, the exposition is kept brief and mathematical details are omitted where possible. The discussion is styled after a number of texts, namely Lee (2013, 2006), O'Neill (1983) and Choquet-Bruhat et al. (1982), in which the interested reader can find the omitted details.

A.1. Manifolds and Coordinate Charts

The basic building block of differential geometry is the manifold M , which can be intuitively understood as a space which locally “looks like” \mathbb{R}^n , the n -dimensional Euclidean space. More concretely, each manifold comes with an atlas of charts (coordinate systems) $\phi_i: M \supset U_i \rightarrow \mathbb{R}^n$, defined on open sets U_i of M . Using a chart ϕ , an abstract point $p \in M$ is transformed into its coordinate representation $(x^1(p), \dots, x^n(p)) \in \mathbb{R}^n$, where $x^j = \pi^j \circ \phi$ is the projection to the j th coordinate. We say that the dimension of M is $\dim(M) = n$ and use the shorthand $\phi = (x^1, \dots, x^n)$. A change in coordinates then corresponds to the transition map $\phi_i \circ \phi_j^{-1}: \mathbb{R}^n \rightarrow \mathbb{R}^n$, which changes a tuple of n coordinates to another tuple of n coordinates describing the same point. ARCMANCER requires the manifold M to be differentiable, meaning that the maps $\phi_i \circ \phi_j^{-1}$ are differentiable. However, in the following we assume smooth (infinitely differentiable) manifolds.

A.2. Tensors

The tangent space of M at point p , written $T_p M$, is the space of all vectors tangent to M at p . If $\dim(M) = n$, then $T_p M$ is an n -dimensional real vector space. The dual space of $T_p M$, written $T_p^* M$ and called the cotangent space of M at p , is the space of all linear maps $\omega_p: T_p M \rightarrow \mathbb{R}$. These linear maps are called one-forms. For each chart $\phi = (x^1, \dots, x^n)$, at point p we can define the vectors $\partial_i|_p := \left. \frac{\partial}{\partial x^i} \right|_p$, called coordinate vectors.

The vector $\partial_i|_p$ points toward the direction where the i th coordinate increases at p . The complete set of coordinate vectors, $\{\partial_1|_p, \dots, \partial_n|_p\}$, forms a basis for $T_p M$, called the coordinate basis. Any vector $v \in T_p M$ can be written in terms of its components in this special basis as $v = v^i \partial_i|_p$, where now $v^i \in \mathbb{R}$ are the components of v in the chart ϕ and the Einstein summation convention has been assumed. Likewise, there exists a set of one-forms, $\{dx^1|_p, \dots, dx^n|_p\}$, called the coordinate one-forms, which forms a basis for $T_p^* M$ and makes it possible to write any one-form $\theta \in T_p^* M$ in terms of its components as $\theta = \theta_i dx^i$. The coordinate vectors and one-forms obey $dx^i|_p(\partial_j|_p) = \delta^i_j$, where δ^i_j is the Kronecker delta symbol. Together, the coordinate vectors and one-forms are called the coordinate frame. The tangent and cotangent spaces can naturally have bases other than the coordinate bases as well (see Appendix A.6), but the coordinate basis is the default basis used in ARCMANCER.

A rank (k, l) tensor can then be defined as a multilinear map

$$T: \overbrace{T_p^* M \times \dots \times T_p^* M}^{k \text{ times}} \times \overbrace{T_p M \times \dots \times T_p M}^{l \text{ times}} \rightarrow \mathbb{R}.$$

In general, the factors $T_p M$ and $T_p^* M$ may be in any order, and the ordering is important, but for convenience, we will use the ordering given above. A rank $(0, 0)$ corresponds to a scalar quantity, rank $(1, 0)$ tensors are equivalent to vectors, and rank $(0, 1)$ tensors to one-forms. The components of a tensor at p in the coordinate

basis can be directly found from

$$T_{i_{k+1}\dots+i_{k+l}}^{i_1\dots i_k} = T(dx^{i_{k+1}}|_p, \dots, dx^{i_{k+l}}|_p, \partial_{i_1}|_p, \dots, \partial_{i_k}|_p). \quad (31)$$

Using the components, a tensor can be locally defined as an expansion

$$T = T_{i_{k+1}\dots+i_{k+l}}^{i_1\dots i_k} \partial_{i_1}|_p \otimes \dots \otimes \partial_{i_k}|_p \otimes dx^{i_{k+1}}|_p \otimes \dots \otimes dx^{i_{k+l}}|_p. \quad (32)$$

If instead of the chart $\phi = (x^1, \dots, x^n)$ we wish to use another (overlapping) chart $\psi = (y^1, \dots, y^n)$ to represent the tensor T , the components $T_{i_{k+1}\dots+i_{k+l}}^{i_1\dots i_k} \in \mathbb{R}$ of a tensor must be transformed accordingly. The new components turn out to be

$$T_{i_{k+1}\dots+i_{k+l}}^{i_1\dots i_k} = T_{j_{k+1}\dots+j_{k+l}}^{j_1\dots j_k} J_{j_1}^{i_1} \dots J_{j_k}^{i_k} (J^{-1})_{i_{k+1}}^{j_{k+1}} \dots (J^{-1})_{i_{k+l}}^{j_{k+l}}, \quad (33)$$

where $J_j^i = \partial(y^i \circ \phi^{-1}) / \partial x^j$ are the components of the Jacobian J of the function $\psi \circ \phi^{-1}$ at point p , and similarly for the inverse of the Jacobian J^{-1} .

The definitions above generalize to vector, one-form, and tensor fields, which can be understood as functions which, for each point p of M , pick a specific vector, one-form, or tensor, respectively. In physics-oriented GR literature, all tensorial quantities are usually tensor fields. In addition, a convention called the abstract index notation (Penrose & Rindler 1987) is often used. In this convention, for example, a rank (2, 3) tensor field T can be written as T_{cd}^{ab} . The number and ordering of the upper and lower indexes is taken only to signify the number and ordering of the factors $T_p M$ and $T_p^* M$ in the definition, Equation (32), of the tensor. This approach makes it possible to write all coordinate-invariant tensor operations tersely, without specifying any underlying basis. In this paper, we use the abstract index notation wherever possible.

A.3. Metric

A manifold may have a special rank (0, 2) tensor field called the metric, usually written g_{ab} . The metric defines the inner product of vectors $\langle v^a w^a \rangle = g_{ab} v^a w^b$ and consequently a norm $\|v^a\| = \sqrt{|g_{ab} v^a v^b|}$ on each $T_p M$. Intuitively, the metric defines the distance between nearby points x and $x + \Delta x$ as the norm of the tangent vector approximated by Δx . In physics, the components of the metric are often written in the form of a line element ds^2 , essentially an expansion in terms of coordinate basis tensors, as $ds^2 = g_{ij} dx^i \otimes dx^j$. The signature of the metric is the pair (p, q) of the number of positive and negative eigenvalues of the matrix of components of g_{ab} (in any basis), respectively, assuming $p + q = n$. If $q = 0$, the metric and the manifold are said to be Riemannian. For signatures of $(n - 1, 1)$ or $(1, n - 1)$, the metric and the manifold are said to be Lorentzian. In particular, GR is defined in terms of a four-dimensional Lorentzian manifold. In this paper, Lorentzian metrics are always assumed to be of type $(1, n - 1)$, and in particular for GR, this implies the $(+---)$ metric convention.

A choice of metric also defines a unique metric-compatible zero-torsion connection ∇ , often called the Levi-Civita connection. A connection in general can be roughly said to characterize which vectors of the tangent spaces of two nearby points are to be considered equal, or alternatively, to yield a vector field $\nabla_X Y$ describing the rate of change of the vector field Y in the direction of the vector field X , called the (natural) covariant derivative of Y with respect to X . Written in terms of

the coordinate vector fields, the connection is $\nabla_{\partial_i}(\partial_j) = \Gamma_{ij}^k \partial_k$, where Γ_{ij}^k are the Christoffel symbols (of the second kind). This work only uses the Levi-Civita connection, the zero-torsion property of which can be written as $\Gamma_{ij}^k = \Gamma_{ji}^k$, and the metric compatibility as $\nabla_c g_{ab} = 0$. In terms of the metric, the Christoffel symbols read

$$\Gamma_{ij}^k = \frac{1}{2} g^{km} (\partial_i g_{jm} + \partial_j g_{im} - \partial_m g_{ij}). \quad (34)$$

The covariant derivative can be generalized for any tensor field T . Using the Christoffel symbols, the covariant derivative of T with respect to X can be written in component form as

$$\begin{aligned} \nabla_X T &= X^i \nabla_{\partial_i} T_{b_1 \dots b_l}^{a_1 \dots a_k} \\ &= X^i (\partial_i T_{b_1 \dots b_l}^{a_1 \dots a_k} + T_{b_1 \dots b_l}^{c_1 \dots a_k} \Gamma_{c_1 i}^{a_1} \\ &\quad + \dots + T_{b_1 \dots b_l}^{a_1 \dots c_k} \Gamma_{c_k i}^{a_k} - T_{c_1 \dots b_l}^{a_1 \dots a_k} \Gamma_{b_l i}^{c_1} \\ &\quad - \dots - T_{b_1 \dots c_l}^{a_1 \dots a_k} \Gamma_{b_l i}^{c_l}). \end{aligned} \quad (35)$$

In the abstract index notation, the covariant derivative is written simply as $\nabla_X T = X^c \nabla_c T_{b_1 \dots b_l}^{a_1 \dots a_k}$. The term covariant derivative of T , without additional qualifiers, is often used to refer only to the vector field independent part $\nabla_c T_{b_1 \dots b_l}^{a_1 \dots a_k}$. If $\nabla_X T = 0$, we say that T is parallel (transported) along the vector field X . This concept can be extended to the case where X is a tangent vector field of a curve γ ; see below.

For semi-Riemannian spaces, the metric also divides vectors at a point into three categories. A vector $v \in T_p M$ is said to be timelike if the inner product $g_{ab} v^a v^b$ is positive, null if it is zero, and spacelike otherwise. The same classification can be extended to vector fields V if the sign of the inner product is the same everywhere on M .

A.4. Geodesics and Other Curves

For ray-tracing mock observations, the notion of curves and especially geodesics on a manifold is essential. A curve γ can be thought of as a map $\gamma: \mathbb{R} \supset I \rightarrow M$. The curve defines a vector field u^a through $u(\lambda) = d\gamma(\lambda)/d\lambda$ along the curve. This vector field can be understood as the velocity vector field of the curve. The curve itself is a solution of

$$\frac{du^a(\lambda)}{d\lambda} = \nabla_{u^b} u^a = u^b \nabla_b u^a. \quad (36)$$

If in particular $u^b \nabla_b u^a = 0$, the tangent vector of the curve is parallel transported along the curve, and we say that the curve is a geodesic. Equation (36) can also be written as

$$u^b \nabla_b u^a = f^a, \quad (37)$$

where the vector field f^a is analogous to a force that causes the curve to deviate from a straight path along the manifold. For geodesics, $f^a = 0$, corresponding to the notion of geodesics as the straightest possible paths. Written in terms of components in a specific chart, Equation (37) reads

$$\frac{du^a(\lambda)}{d\lambda} = -\Gamma_{bc}^a u^b u^c + f^a. \quad (38)$$

In this form, the equation can be solved numerically, as long as the solution stays within the domain of the chart.

Generic tensorial quantities can also be parallel transported along the curve. For a rank $k + l$ tensor field $T_{b_1 \dots b_l}^{a_1 \dots a_k}$, the

equation of parallel transport is

$$\nabla_u^c T_{b_1 \dots b_l}^{a_1 \dots a_k} = u^c \nabla_c T_{b_1 \dots b_l}^{a_1 \dots a_k} = 0, \quad (39)$$

which in component form can be found from Equation (35) with the substitution $X = u^a$. Equation (39) can likewise be directly solved numerically.

Finally, curves for which the tangent vector field u^a is always timelike, spacelike, or null, are called timelike, spacelike, or null, respectively. This characterization is important for geodesics, for which the tangent vector field can never change their timelike, spacelike or null character. As such, geodesics always fall in one of these categories.

A.5. Level Hypersurfaces

Manifolds may contain many kinds of submanifolds. A particularly useful class of submanifolds are those defined by level sets of functions, called level hypersurfaces. If $S: M \rightarrow \mathbb{R}$ is a smooth map, then the sets $S_c := S^{-1}(c) = \{p \in M | S(p) = c\}$ are submanifolds of M for each c in the codomain of S if $dS|_p \neq 0$ for all $p \in S_c$. We can subsume the constant c in the definition of the function S and take $c = 0$, which is always assumed in the ARCMANCER code. Furthermore, with a slight abuse of notation, we use the function S defining the hypersurface to refer to the hypersurface itself.

A level hypersurface divides the manifold M into two disconnected subsets corresponding to regions where the value of S is either negative or positive. In some cases, these may be conveniently taken to be the ‘‘outside’’ and the ‘‘inside’’ of a region bounded by the level hypersurface.

A.6. Local Frames

Tensor fields can also be expressed in terms of bases other than the coordinate basis. A local frame defined in an open set $U \supset M$ consists of n vector fields $\{E_1, \dots, E_n\}$ that form a basis for $T_p M$ at each $p \in U$. For each local frame, there is a corresponding local coframe of one-form fields $\{\omega^1, \dots, \omega^n\}$ for which $\omega^i(E_j) = \delta_j^i$. The components of a rank (k, l) tensor $T_{b_1 \dots b_l}^{a_1 \dots a_k}$ in terms of these bases can then be defined through (see Equation (31))

$$T_{j_1 \dots j_l}^{i_1 \dots i_k} = T(\omega^{i_1}, \dots, \omega^{i_k}, E_{j_1}, \dots, E_{j_l}). \quad (40)$$

The transformation from the components of a tensor in a coordinate frame to components in terms of a local frame can be found by first writing the local (co)frame in terms of the coordinate frame as $E_i = \sum_\alpha E_i^\alpha \partial_\alpha$ and $\omega^j = \sum_\beta \omega_\beta^j dx^\beta$. Using the multilinearity of an arbitrary (k, l) tensor $T_{b_1 \dots b_l}^{a_1 \dots a_k}$ in Equation (40) then yields

$$T_{j_1 \dots j_l}^{i_1 \dots i_k} = \sum_{\beta_1} \dots \sum_{\alpha_l} \omega_{\beta_1}^{i_1} \dots \omega_{\beta_k}^{i_k} E_{j_1}^{\alpha_1} \dots E_{j_l}^{\alpha_l} \times T(dx^{\beta_1}, \dots, dx^{\beta_k}, \partial_{\alpha_1}, \dots, \partial_{\alpha_l}), \quad (41)$$

where $T_{\alpha_1 \dots \alpha_l}^{\beta_1 \dots \beta_k} = T(dx^{\beta_1}, \dots, dx^{\beta_k}, \partial_{\alpha_1}, \dots, \partial_{\alpha_l})$ are the components of the tensor in the coordinate frame. From a computational perspective, it is useful to note that the set of numbers E_i^α can be interpreted as a matrix, for which the corresponding set ω_β^j forms an inverse matrix. Equation (41) is seen to resemble Equation (33), with these matrices taking the place of the Jacobians.

Orthonormal local frames define a set of convenient projections of $T_p M$ onto orthogonal complements at each point p . A collection of k non-null basis vectors $\mathcal{K} = \{E_{ij} \in T_p M | j = 1, \dots, k\}$ defines a projection onto the subspace of $T_p M$ spanned by \mathcal{K} through

$$P_{\parallel}(E_{i_1}, \dots, E_{i_k})_b^a = \sum_{j=1}^k \frac{E_{ij}^a (E_{ij})_b}{E_{ij}^c (E_{ij})_c}. \quad (42)$$

Similarly, a projection onto the orthogonal complement of this subspace is defined via

$$P_{\perp}(E_{i_1}, \dots, E_{i_k})_b^a = \delta_b^a - \sum_{j=1}^k \frac{E_{ij}^a (E_{ij})_b}{E_{ij}^c (E_{ij})_c}. \quad (43)$$

For Lorentzian spaces, direct projections with respect to a null vector k^a do not work. However, given a unit timelike vector u^a , a projection onto the space orthogonal to both u^a and k^a can be given through

$$\begin{aligned} P_{\perp}(u, k)_b^a &= \delta_b^a - \frac{u^a u_b}{u^c u_c} - \frac{s^a s_b}{s^c s_c} \\ &= \delta_b^a - \frac{u^a k_b}{K} + \frac{k^a u_b}{K} + \frac{k^a k_b}{K^2}, \end{aligned} \quad (44)$$

where $s^a = P_{\perp}(u)_b^a k^b$ and $K = u_a k^a$. In four dimensions, the operator $P_{\perp}(u, k)_b^a$ is a projection onto a two-dimensional surface on which the observed components of polarization are defined. In this context, $P_{\perp}(u, k)_b^a$ is known as the screen projection operator (Gammie & Leung 2012).

A.6.1. Lorentz Frames and Observers

In the case of a four-dimensional Lorentzian manifold, an orthonormal local frame is often called a (local) Lorentz frame or a tetrad. The tetrad can in general consist of any permissible combination of null, spacelike, and timelike vector fields. However, in this work, we use the term Lorentz frame to mean a mutually orthonormal combination of one timelike vector, E_t , and three spacelike vectors, E_x , E_y , and E_z , defined at a point. Since parallel transport preserves inner products, a parallel transported Lorentz frame is still a valid Lorentz frame.

Observers in GR are characterized by a timelike curve γ , a world line. An observer’s rest frame can then be defined as a Lorentz frame for which the timelike basis vector is given by the observer’s four-velocity, or $E_t = d\gamma(\lambda)/d\lambda$, and the spatial triad $\{E_x, E_y, E_z\}$ defines the observer’s choice of spatial coordinate system. The values of physical quantities as measured by the observer are obtained by expressing them in the observer’s rest-frame basis.

For example, an important feature of GR is that angles between vectors at a point is observer dependent. The angle θ between vectors x^a and y^a as measured by an observer with a four-velocity u^a is then

$$\cos \theta(u; x, y) = \frac{-x_{\perp}^a y_{\perp a}}{\sqrt{|(x_{\perp}^b x_{\perp b})(y_{\perp}^c y_{\perp c})|}}, \quad (45)$$

where $x_{\perp}^a = P_{\perp}(u)_b^a x^b$ and $y_{\perp}^a = P_{\perp}(u)_b^a y^b$.

Appendix B Radiative Transfer

B.1. Geometric Optics

Briefly, a propagating monochromatic radiation front can be modeled as a congruence of curves, each perpendicular to the surface of constant phase. This is possible in the limit where the wavelength of the radiation is much smaller than the scale of variations in the radiation front (curvature, amplitude, polarization) and much smaller than the local “radius of curvature of the space” $\propto |R_{bcd}^a|^{-1/2}$, where R_{bcd}^a is the Riemann curvature tensor.

Furthermore, if the contribution of matter is insignificant, i.e., $T^{ab} \sim 0$, the result is that the propagation of radiation can be modeled by solving an equation of radiative transfer along null geodesics. If the integrated contribution of the intervening matter is non-negligible, the radiation front normals will not be geodesics in general, and there will be a matter-dependent forcing term f^a in Equation (37) (see, e.g., Broderick & Blandford 2003, 2004).

B.2. Equation of Radiative Transfer

The classical equation of radiative transfer in Cartesian coordinates is (Mihalas & Mihalas 1984)

$$\left(\frac{\partial}{c \partial t} + \hat{\mathbf{n}} \cdot \nabla \right) I_\nu(\mathbf{x}, t; \hat{\mathbf{n}}) = j_\nu(\mathbf{x}, t; \hat{\mathbf{n}}) - \alpha_\nu(\mathbf{x}, t; \hat{\mathbf{n}}) I_\nu(\mathbf{x}, t; \hat{\mathbf{n}}), \quad (46)$$

where j_ν is the total emission coefficient (emissivity) and α_ν the total absorption coefficient. Equation (46) describes the change in specific intensity at a frequency ν in the direction $\hat{\mathbf{n}}$ at a point \mathbf{x} and time t . Often in astrophysical cases, the one-dimensional time-independent case suffices, in which case Equation (46) reduces to a more commonly seen form,

$$\frac{dI_\nu(s)}{ds} = j_\nu(s) - \alpha_\nu(s) I_\nu(s), \quad (47)$$

where s is the distance along a radiation front normal. Equations (46) and (47) do not take into account interference, quantum effects, or, most importantly, polarization.

Polarization can be included by making the specific intensity I_ν vector-valued by introducing the Stokes intensities $\mathbf{I}_\nu = (I_\nu, Q_\nu, U_\nu, V_\nu)$. Similarly, j is replaced by the vector

$$\mathbf{J}_\nu = (j_{I,\nu}, j_{Q,\nu}, j_{U,\nu}, j_{V,\nu}) \quad (48)$$

of Stokes emissivities and α by the response (or Müller) matrix

$$\mathbf{M}_\nu = \begin{pmatrix} \alpha_{I,\nu} & \alpha_{Q,\nu} & \alpha_{U,\nu} & \alpha_{V,\nu} \\ \alpha_{Q,\nu} & \alpha_{I,\nu} & r_{V,\nu} & -r_{U,\nu} \\ \alpha_{U,\nu} & -r_{V,\nu} & \alpha_{I,\nu} & r_{Q,\nu} \\ \alpha_{V,\nu} & r_{U,\nu} & -r_{Q,\nu} & \alpha_{I,\nu} \end{pmatrix}, \quad (49)$$

where the α -coefficients represent absorption effects, and the r -coefficients relate to Faraday conversion and rotation. It should be noted that different conventions for \mathbf{M} exist, varying by the sign of r_U . The one-dimensional polarized equation of radiative transfer is then

$$\frac{d\mathbf{I}_\nu(s)}{ds} = \mathbf{J}_\nu - \mathbf{M}_\nu(s) \mathbf{I}_\nu(s). \quad (50)$$

The general relativistic generalization of Equation (50) is (e.g., Gammie & Leung 2012)

$$\frac{dN^{ab}}{d\lambda} = J^{ab} + H^{abcd} N_{cd}, \quad (51)$$

where λ is the affine parameter along the curve representing the propagating radiation front, N^{ab} is the (complex-valued) polarization tensor, J^{ab} is the emissivity tensor, and H^{abcd} is the response tensor.

Directly solving Equation (51) requires integrating the 16 real independent components of N^{ab} . This number can be reduced to four by parallel transporting a polarization frame along the geodesic. The frame consists of two orthogonal spacelike vectors that are also orthogonal to the geodesic and the observer four-velocity. Expressing all quantities in this frame using the screen projection operator (44), Equation (51) can be written as

$$\frac{d\mathcal{I}_\nu}{d\lambda} = \mathcal{J}_\nu - \mathcal{M}_\nu \mathcal{I}_\nu, \quad (52)$$

where $\mathcal{I}_\nu = \nu^{-3} I_\nu$, $\mathcal{J} = C \nu^{-2} J_\nu$, $\mathcal{M} = C \nu^{-1} M$, and C is a constant related to the parametrization of the curve. An observer with a four-velocity u^a at one end of the curve, where the tangent is k^a , has $C = u_a k^a / \nu_0$ for an observed frequency ν_0 .

Appendix C

Built-in Manifold and Chart Support

ARCMANCER contains a number of predefined metric spaces and spacetimes together with commonly used coordinate charts for convenience. The number of implemented spaces and charts is expected to grow, but the selection at the time of writing is given in the following. We list all of the charts and the representations of the metric, either as a line element or in matrix form, in these charts for each implemented space. All Lorentzian spacetimes are shown with the (+---) metric convention.

C.1. Riemannian Manifolds

C.1.1. Two-sphere

Spherical coordinates (θ, ϕ)—ARCMANCER uses two copies of the spherical chart (θ, ϕ) to cover the entire two-sphere. The metric, in either copy, is given by the usual

$$ds^2 = d\theta^2 + \sin^2 \theta d\phi^2. \quad (53)$$

C.2. Semi-Riemannian Manifolds

C.2.1. Minkowski Spacetime

Cartesian coordinates (t, x, y, z)—The Cartesian Minkowski coordinates are often in the literature denoted by η_{ab} , and we use the same convention. The line element is diagonal, given by

$$ds^2 = dt^2 - dx^2 - dy^2 - dz^2. \quad (54)$$

Spherical coordinates (t, r, θ, ϕ)—

$$ds^2 = dt^2 - dr^2 - r^2(d\theta^2 + \sin^2 \theta d\phi^2) \quad (55)$$

C.2.2. Kerr Spacetime

The Kerr spacetime (Kerr 1963) is an important solution to the Einstein field equations, representing a rotating black hole exactly and other rotating fluid bodies of finite size asymptotically. The Kerr spacetime is parametrized by the mass M and the angular momentum J . Usually, J is given through the normalized spin parameter $a = J/M$, in which case $a \in [0, M]$, or through the dimensionless spin parameter χ , so that $\chi \in [0, 1]$. The solution reduces to the Schwarzschild spacetime when $\chi = 0$ and further to the flat Minkowski spacetime when $M = 0$.

Boyer–Lindquist coordinates (t, r, θ, ϕ)—Perhaps the clearest representation of the Kerr metric in BL coordinates is through the matrix form

$$(g_{ab}) = \begin{pmatrix} \frac{2Mr}{\rho^2} & 0 & 0 & \frac{2Mr a \sin^2 \theta}{\rho^2} \\ 0 & -\frac{\rho^2}{\Delta} & 0 & 0 \\ 0 & 0 & -\rho^2 & 0 \\ \frac{a \sin^2 \theta}{\rho^2} & 0 & 0 & -\sin^2 \theta \left(r^2 + a^2 + \frac{2Mr a^2 \sin^2 \theta}{\rho^2} \right) \end{pmatrix}, \quad (56)$$

where $\rho^2 = r^2 + a^2 \cos^2 \theta$ and $\Delta = r^2 - 2Mr + a^2$. The BL form of the metric is singular when $\Delta = 0$ or $\rho^2 = 0$. The former condition corresponds to $r_{\pm} = m^2 \pm \sqrt{m^2 - a^2}$, which gives the locations of the inner and outer event horizons, where the curvature is not singular. The condition $\rho^2 = 0$ implies $r = 0$ and $\cos \theta = 0$, and corresponds to the curvature singularity.

$$(g_{ab}) = \begin{pmatrix} e^{2\nu} - B^2 \bar{r}^2 \omega^2 \sin^2 \theta e^{-2\nu} & 0 & 0 & B^2 \bar{r}^2 \omega \sin^2 \theta e^{-2\nu} \\ 0 & -e^{2(\zeta-\nu)} & 0 & 0 \\ 0 & 0 & -e^{2(\zeta-\nu)} \bar{r}^2 & 0 \\ B^2 \bar{r}^2 \omega \sin^2 \theta e^{-2\nu} & 0 & 0 & -B^2 \bar{r}^2 \sin^2 \theta e^{-2\nu} \end{pmatrix}, \quad (61)$$

Cartesian KS coordinates (t, x, y, z)—The Cartesian KS coordinates come in two flavors, ingoing and outgoing, adapted to null geodesics that move radially inwards or outwards, respectively. As such, they are a generalization of the ingoing and outgoing Eddington–Finkelstein coordinates for a Schwarzschild black hole. Of the two variants, only the ingoing form is typically seen in the literature. The metric in the ingoing KS coordinates is most easily written as a sum,

$$g_{ab} = \eta_{ab} - F(x, y, z; a) l_a l_b, \quad (57)$$

where η is the Cartesian Minkowski metric,

$$F(x, y, z; a) = \frac{2Mr^3}{r^4 + a^2 z^2}, \quad (58)$$

$$(l_a) = \left(1, \frac{rx + ay}{a^2 + r^2}, \frac{ry - ax}{a^2 + r^2}, \frac{z}{r} \right), \quad (59)$$

and r is defined implicitly through $x^2 + y^2 + z^2 = r^2 + a^2(1 - z^2/r^2)$. The vector l_a is null with respect to both

η_{ab} and g_{ab} . The metric in the outgoing coordinates is similarly given as a sum as in Equation (57), but with

$$(l_a) = \left(-1, \frac{rx - ay}{a^2 + r^2}, \frac{ry + ax}{a^2 + r^2}, \frac{z}{r} \right). \quad (60)$$

When numerically computing geodesics near a Kerr black hole, it is crucial that the right form of the KS coordinates is chosen. If the geodesic approaches the hole, the ingoing chart should be used, and for a geodesic going away from the hole, the outgoing chart should be used. Failure to do so has significant computational penalties, as can be seen in Section 5.1.2. Note that this implies that for a geodesic coming in toward a Kerr black hole, passing by it, and then leaving, both ingoing and outgoing charts should be used with a change of coordinates near the closest approach.

C.3. AlGendy–Morsink (AGM) Spacetime

AGM coordinates (t, \bar{r}, θ, ϕ)—The AGM spacetime and coordinates (AlGendy & Morsink 2014) actually refer to a specific choice of a Butterworth–Ipser (BI) spacetime, with the accompanying coordinate chart (Butterworth & Ipser 1976). The BI spacetime is a general representation of the spacetime outside an axisymmetric rotating fluid body. The AGM spacetime is a special case representing the spacetime around a physically realistic oblate rotating neutron star. The AGM representation is accurate up to second order in the dimensionless rotation parameter $\bar{\Omega} = \Omega R_e^{3/2} M^{-1/2}$, where Ω is the rotational angular velocity of the star as seen by a distant observer, R_e is the equatorial radius of the star, and M is the mass of the star. In BI coordinates, the generic axisymmetric metric reads

where ν , B , ω , and ζ are the so-called metric functions or potentials. The AGM metric is defined by using

$$\nu = \ln \frac{1 - M/(2\bar{r})}{1 + M/(2\bar{r})} + \left(\frac{\beta}{3} - q P_2(\cos \theta) \right) \left(\frac{M}{\bar{r}} \right)^3, \quad (62)$$

$$B = 1 - \left(\frac{M}{2\bar{r}} \right)^2 + \beta \left(\frac{M}{\bar{r}} \right)^2, \quad (63)$$

$$\omega = \frac{2j}{\bar{r}} \left(\frac{M}{\bar{r}} \right)^2 \left(1 - \frac{3M}{\bar{r}} \right), \quad (64)$$

$$\zeta = \ln \left[1 - \left(\frac{M}{2\bar{r}} \right)^2 \right] + \beta \frac{4P_2(\cos \theta) - 1}{3}, \quad (65)$$

where P_2 is the second-order Legendre polynomial. The constants q and β are the dimensionless moments of energy density and pressure, respectively, and $j = J/M^2$ is the dimensionless angular momentum. AlGendy & Morsink (2014) found these constants to be well described across

various neutron star parameters and equations of state by the approximate relations

$$j = (1.136 - 2.53x + 5.6x^2)\bar{\Omega}, \quad (66)$$

$$q = -0.11x^{-2}\bar{\Omega}^2, \quad (67)$$

$$\beta = 0.4454x\bar{\Omega}^2, \quad (68)$$

where $x = M/R_e$ is called the compactness (parameter). Along with the metric potentials, AlGendy & Morsink (2014) also derived an equation for the shape of the surface of a rotating neutron star, used in Section 6.2,

$$R(\theta) = R_e[1 + (-0.788 + 1.030x)\bar{\Omega}^2 \cos^2 \theta]. \quad (69)$$

From this, the flattening, also called the oblateness, of the star is given by

$$f = \frac{R(\pi/2) - R(0)}{R(\pi/2)} = (0.788 - 1.030x)\bar{\Omega}^2. \quad (70)$$

It should be noted that the quantities R_e and $R(\theta)$ are not defined in terms of \bar{r} , but instead in terms of a radial coordinate $r = Be^{-\nu\bar{r}}$.

C.4. Hartle–Thorne Spacetime

The Hartle–Thorne spacetime (Hartle & Thorne 1968) describes the spacetime around a rotating oblate star. The original derivation was based on a perturbation of the non-rotating Schwarzschild spacetime. In ARCMANCER, we use instead a version based on a perturbation of the rotating Kerr spacetime, given in Glampedakis & Babak (2006).

Glampedakis–Babak (GB) coordinates (t, r, θ, ϕ)—The GB coordinates are based on the BL coordinates of the Kerr spacetime. In these coordinates, in a parametrization used by Bauböck et al. (2012), the metric is given by

$$g_{ab} = g_{ab}^{\text{Kerr}} + \eta\chi^2 h_{ab}, \quad (71)$$

where g_{ab}^{Kerr} is the Kerr metric in BL coordinates (see Equation (56)), and

$$h^{00} = -(1 - 2M/r)^{-1} [(1 - 3\cos^2\theta)\mathcal{F}_1(r)]$$

$$h^{11} = -(1 - 2M/r)[(1 - 3\cos^2\theta)\mathcal{F}_1(r)], \quad (72)$$

$$h^{22} = r^{-2} [(1 - 3\cos^2\theta)\mathcal{F}_2(r)]$$


$$h^{33} = r^{-2} \sin^2\theta [(1 - 3\cos^2\theta)\mathcal{F}_2(r)]. \quad (73)$$

Here, M is the mass of the star, $\chi = J/M$ is the dimensionless angular momentum and η parametrizes the mass quadrupole moment $q = -\chi^2(1 + \eta)$, so that $\eta = 0$ corresponds to the quadrupole moment of the Kerr spacetime. The functions $\mathcal{F}_{1,2}$ are given in Glampedakis & Babak (2006).

ORCID iDs

Pauli Pihajoki  <https://orcid.org/0000-0003-1758-1908>

Joonas Nättilä  <https://orcid.org/0000-0002-3226-4575>

Peter H. Johansson  <https://orcid.org/0000-0001-8741-8263>

References

Ahnert, K., & Mulansky, M. 2011, in AIP Conf. Proc. 1389, Numerical Analysis and Applied Mathematics ICNAAM 2011, ed. T. E. Simos et al. (Melville, NY: AIP), 1586
AlGendy, M., & Morsink, S. M. 2014, *ApJ*, 791, 78

Alsing, J., Silva, H. O., & Berti, E. 2018, *MNRAS*, 478, 1377
Atamurotov, F., Ghosh, S. G., & Ahmedov, B. 2016, *EPJC*, 76, 273
Bambi, C. 2012, *ApJ*, 761, 174
Baron, E., & Hauschildt, P. H. 2004, *A&A*, 427, 987
Basko, M. M., & Sunyaev, R. A. 1976, *MNRAS*, 175, 395
Bauböck, M., Özel, F., Psaltis, D., & Morsink, S. M. 2015a, *ApJ*, 799, 22
Bauböck, M., Psaltis, D., & Özel, F. 2015b, *ApJ*, 811, 144
Bauböck, M., Psaltis, D., Özel, F., & Johannsen, T. 2012, *ApJ*, 753, 175
Beloborodov, A. M. 2002, *ApJL*, 566, L85
Bower, G. C., Markoff, S., Dexter, J., et al. 2015, *ApJ*, 802, 69
Bradley, M., & Fodor, G. 2009, *PhRvD*, 79, 044018
Broderick, A., & Blandford, R. 2003, *MNRAS*, 342, 1280
Broderick, A., & Blandford, R. 2004, *MNRAS*, 349, 994
Broderick, A. E. 2004, PhD thesis, California Institute of Technology
Broderick, A. E., Fish, V. L., Johnson, M. D., et al. 2016, *ApJ*, 820, 137
Bromley, B. C., Melia, F., & Liu, S. 2001, *ApJL*, 555, L83
Butterworth, E. M., & Ipser, J. R. 1976, *ApJ*, 204, 200
Cadeau, C., Morsink, S. M., Leahy, D., & Campbell, S. S. 2007, *ApJ*, 654, 458
Carter, B. 1968, *PhRv*, 174, 1559
Chael, A. A., Johnson, M. D., Narayan, R., et al. 2016, *ApJ*, 829, 11
Chan, C.-k., Medeiros, L., Özel, F., & Psaltis, D. 2017, arXiv:1706.07062
Chan, C.-k., Psaltis, D., & Özel, F. 2013, *ApJ*, 777, 13
Chandrasekhar, S. 1960, Radiative Transfer (New York: Dover)
Chen, B., Kantowski, R., Dai, X., Baron, E., & Maddumage, P. 2015, *ApJS*, 218, 4
Chen, K., Halpern, J. P., & Filippenko, A. V. 1989, *ApJ*, 339, 742
Choquet-Bruhat, Y., DeWitt-Morette, C., & Dillard-Bleick, M. 1982, Analysis, Manifolds, and Physics, Analysis, Manifolds, and Physics No. Part 1 (Amsterdam: North-Holland)
Connors, P. A., & Stark, R. F. 1977, *Natur*, 269, 128
Connors, P. A., Stark, R. F., & Piran, T. 1980, *ApJ*, 235, 224
Cunningham, C. 1976, *ApJ*, 208, 534
Cunningham, C. T. 1975, *ApJ*, 202, 788
Cunningham, C. T., & Bardeen, J. M. 1972, *ApJL*, 173, L137
Czerny, B., & Elvis, M. 1987, *ApJ*, 321, 305
Davis, S. W., Blaes, O. M., Hubeny, I., & Turner, N. J. 2005, *ApJ*, 621, 372
De Falco, V., Falanga, M., & Stella, L. 2016, *A&A*, 595, A38
Dexter, J. 2016, *MNRAS*, 462, 115
Dexter, J., & Agol, E. 2009, *ApJ*, 696, 1616
Doeleman, S., Agol, E., & Backer, D. 2009, Astronomy, Vol. 2010, astro2010: The Astronomy and Astrophysics Decadal Survey, 2010, 68
Dormand, J., & Prince, P. 1980, *JCoAM*, 6, 19
Ebisawa, K., Mitsuda, K., & Hanawa, T. 1991, *ApJ*, 367, 213
Eisenhauer, F., Perrin, G., Brandner, W., et al. 2008, *Proc. SPIE*, 7013, 70132A
Falcke, H., Melia, F., & Agol, E. 2000, *ApJL*, 528, L13
Fukue, J., & Yokoyama, T. 1988, *PASJ*, 40, 15
Gammie, C. F., & Leung, P. K. 2012, *ApJ*, 752, 123
Gammie, C. F., McKinney, J. C., & Tóth, G. 2003, *ApJ*, 589, 444
García, F., Ranea-Sandoval, I. F., & Johannsen, T. 2016, *A&A*, 587, A141
Gedalin, M., & Melrose, D. B. 2001, *PhRvE*, 64, 027401
Gendreau, K. C., Arzoumanian, Z., & Okajima, T. 2012, *Proc. SPIE*, 8443, 844313
Glampedakis, K., & Babak, S. 2006, *CQGra*, 23, 4167
Gold, R., McKinney, J. C., Johnson, M. D., & Doeleman, S. S. 2017, *ApJ*, 837, 180
González Caniulef, D., Zane, S., Taverna, R., Turolla, R., & Wu, K. 2016, *MNRAS*, 459, 3585
Graham, A. W., & Scott, N. 2013, *ApJ*, 764, 151
Hairer, E., Nørsett, S., & Wanner, G. 2008, Solving Ordinary Differential Equations I: Nonstiff Problems (Heidelberg: Springer)
Hartle, J. B., & Thorne, K. S. 1968, *ApJ*, 153, 807
Henon, M. 1982, *PhyD*, 5, 412
Hessels, J. W. T., Ransom, S. M., Stairs, I. H., et al. 2006, *Sci*, 311, 1901
Hunter, J. D. 2007, *CSE*, 9, 90
Ishihara, H., Takahashi, M., & Tomimatsu, A. 1988, *PhRvD*, 38, 472
Israel, W., & Khan, K. A. 1964, *NCim*, 33, 331
Jaroszynski, M., Wambsganss, J., & Paczynski, B. 1992, *ApJL*, 396, L65
Karas, V., Vokrouhlický, D., & Polnarev, A. G. 1992, *MNRAS*, 259, 569
Katz, J. I. 1997, *ApJ*, 478, 527
Kelly, B. J., Baker, J. G., Etienne, Z. B., Giacomazzo, B., & Schnittman, J. 2017, *PhRvD*, 96, 123003
Kerr, R. P. 1963, *PhRvL*, 11, 237
Lamb, F. K., Boutloukos, S., Van Wassenhove, S., et al. 2009a, *ApJ*, 706, 417
Lamb, F. K., Boutloukos, S., Van Wassenhove, S., et al. 2009b, *ApJL*, 705, L36
Lee, J. 2006, Riemannian Manifolds: An Introduction to Curvature, Graduate Texts in Mathematics (New York: Springer)

- Lee, J. 2013, *Introduction to Smooth Manifolds*, Graduate Texts in Mathematics (New York: Springer)
- Lehto, H. J., & Valtonen, M. J. 1996, *ApJ*, **460**, 207
- Liang, E. P. T., & Price, R. H. 1977, *ApJ*, **218**, 247
- Lo, K. H., Miller, M. C., Bhattacharyya, S., & Lamb, F. K. 2013, *ApJ*, **776**, 19
- Lo, K. H., Miller, M. C., Bhattacharyya, S., & Lamb, F. K. 2018, *ApJ*, **854**, 187
- Ludlam, R. M., Miller, J. M., Cackett, E. M., et al. 2016, *ApJ*, **824**, 37
- Luminet, J.-P. 1979, *A&A*, **75**, 228
- Lyubarskii, Y. E., & Syunyaev, R. A. 1982, *SvAL*, **8**, 330
- Majumdar, S. D. 1947, *PhRv*, **72**, 390
- Mihalas, D., & Mihalas, B. W. 1984, *Foundations of Radiation Hydrodynamics* (New York: Oxford Univ. Press)
- Miller, M. C., & Lamb, F. K. 1998, *ApJL*, **499**, L37
- Miller, M. C., & Lamb, F. K. 2015, *ApJ*, **808**, 31
- Mohr, P. J., Newell, D. B., & Taylor, B. N. 2016, *RvMP*, **88**, 035009
- Morsink, S. M., Leahy, D. A., Cadeau, C., & Braga, J. 2007, *ApJ*, **663**, 1244
- Mościbrodzka, M., & Gammie, C. F. 2018, *MNRAS*, **475**, 43
- Mundim, B. C., Nakano, H., Yunes, N., et al. 2014, *PhRvD*, **89**, 084008
- Nättilä, J., & Pihajoki, P. 2018, *A&A*, **615**, A50
- Noble, S. C., Gammie, C. F., McKinney, J. C., & Del Zanna, L. 2006, *ApJ*, **641**, 626
- Novikov, I. D., & Thorne, K. S. 1973, in *Black Holes (Les Astres Occlus)*, ed. C. Dewitt & B. S. Dewitt (New York: Gordon and Breach), 343
- O'Neill, B. 1983, *Semi-Riemannian Geometry With Applications to Relativity*, Pure and Applied Mathematics (Sand Diego, CA: Academic)
- Özel, F., & Freire, P. 2016, *ARA&A*, **54**, 401
- Page, D. N., & Thorne, K. S. 1974, *ApJ*, **191**, 499
- Papapetrou, A. 1945, *PRIAA*, **51**, 191, <https://www.jstor.org/stable/20488481>
- Pechenick, K. R., Ftaclas, C., & Cohen, J. M. 1983, *ApJ*, **274**, 846
- Penrose, R., & Rindler, W. 1987, *Spinors and Space-Time*, Vol. 1, Two-Spinor Calculus and Relativistic Fields (Cambridge: Cambridge Univ. Press)
- Pihajoki, P. 2015, *CeMDA*, **121**, 211
- Pihajoki, P. 2016, *MNRAS*, **457**, 1145
- Pihajoki, P., Mannerkoski, M., Nättilä, J., & Johansson, P. H. 2018, *Arcmancer*, v0.2, Zenodo, doi:10.5281/zenodo.1264666.
- Porth, O., Olivares, H., Mizuno, Y., et al. 2017, *ComAC*, **4**, 1
- Potekhin, A. Y. 2014, *PhyU*, **57**, 735
- Poutanen, J., & Gierliński, M. 2003, *MNRAS*, **343**, 1301
- Psaltis, D., & Johannsen, T. 2012, *ApJ*, **745**, 1
- Psaltis, D., & Özel, F. 2014, *ApJ*, **792**, 87
- Rauch, K. P., & Blandford, R. D. 1991, *ApJL*, **381**, L39
- Rauch, K. P., & Blandford, R. D. 1994, *ApJ*, **421**, 46
- Riffert, H., & Meszaros, P. 1988, *ApJ*, **325**, 207
- Sadiq, J., Zlochower, Y., & Nakano, H. 2018, *PhRvD*, **97**, 084007
- Savitzky, A., & Golay, M. J. E. 1964, *AnaCh*, **36**, 1627
- Schnittman, J. D., & Krolik, J. H. 2009, *ApJ*, **701**, 1175
- Schnittman, J. D., & Krolik, J. H. 2010, *ApJ*, **712**, 908
- Shakura, N. I., & Sunyaev, R. A. 1973, *A&A*, **24**, 337
- Shcherbakov, R. V., & Huang, L. 2011, *MNRAS*, **410**, 1052
- Shcherbakov, R. V., & McKinney, J. C. 2013, *ApJL*, **774**, L22
- Sillanpää, A., Haarala, S., Valtonen, M. J., Sundelius, B., & Byrd, G. G. 1988, *ApJ*, **325**, 628
- Stark, R. F., & Connors, P. A. 1977, *Natur*, **266**, 429
- Steiner, A. W., Lattimer, J. M., & Brown, E. F. 2016, *EPJA*, **52**, 18
- Stergioulas, N. 2003, *LRR*, **6**, 3
- Stevens, A. L., Fiege, J. D., Leahy, D. A., & Morsink, S. M. 2016, *ApJ*, **833**, 244
- Strohmayer, T. E. 1992, *ApJ*, **388**, 138
- Suleimanov, V., Poutanen, J., & Werner, K. 2011, *A&A*, **527**, A139
- Treu, T. 2010, *ARA&A*, **48**, 87
- Urbanec, M., Miller, J. C., & Stuchlík, Z. 2013, *MNRAS*, **433**, 1903
- Valtonen, M. J., Lehto, H. J., Nilsson, K., et al. 2008, *Natur*, **452**, 851
- van der Walt, S., Colbert, S. C., & Varoquaux, G. 2011, *CSE*, **13**, 22
- Viergutz, S. U. 1993, *A&A*, **272**, 355
- Viironen, K., & Poutanen, J. 2004, *A&A*, **426**, 985
- Villata, M., Raiteri, C. M., Sillanpää, A., & Takalo, L. O. 1998, *MNRAS*, **293**, L13
- Vincent, F. H., Bejger, M., Rozanska, A., et al. 2018, *ApJ*, **855**, 116
- Vincent, F. H., Gourgoulhon, E., Herdeiro, C., & Radu, E. 2016, *PhRvD*, **94**, 084045
- Vincent, F. H., Paumard, T., Gourgoulhon, E., & Perrin, G. 2011, *CQGra*, **28**, 225011
- Weinberg, D. H., Mortonson, M. J., Eisenstein, D. J., et al. 2013, *PhR*, **530**, 87
- Weinberg, N., Miller, M. C., & Lamb, D. Q. 2001, *ApJ*, **546**, 1098
- Zhang, S. N., Feroci, M., Santangelo, A., et al. 2016, *Proc. SPIE*, **9905**, 99051Q