



Universidad de Carabobo
Facultad Experimental de Ciencias y Tecnología
Departamento de Matemáticas Aplicadas



Soluciones autosimilares al colapso estelar relativista mediante Maple y GRTensorIII

*Trabajo de Grado presentado por el Profesor Víctor Enrique Medina
Franco como credencial de mérito para optar al título de Magister
Scientiarum en Matemáticas, con la tutoría del Dr. Nelson Falcón*

Valencia, 27 de Marzo de 2025

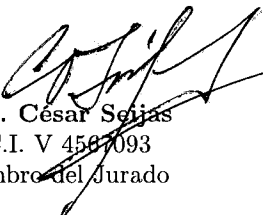


Universidad de Carabobo
Facultad Experimental de Ciencias y Tecnología
Dirección de Postgrado

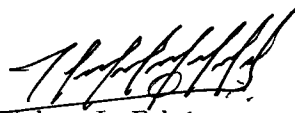


ACTA DE VEREDICTO

Hoy, 27 de marzo de 2025, a las 11:00 a.m., reunidos en el Departamento de Física de la Facultad Experimental de Ciencias y Tecnología de la Universidad de Carabobo, presentes los ciudadanos: Dr. Nelson L. Falcón V., Dr. Luis A. Rodríguez y Dr. César Seijas, de nacionalidad venezolana, mayores de edad, hábiles en derecho, identificados con las cédulas de identidad N° 6101724, 11678882 y 4567093 respectivamente, docentes/investigadores, de la Universidad de Carabobo, con domicilio en Venezuela, miembros del Jurado designado por el Consejo de Postgrado en sesión N° 02/2025 de fecha 27/02/2025 y notificados formalmente el 05/03/2025, para evaluar el Trabajo de Grado, presentado por el ciudadano Víctor Enrique Medina Franco, cédula de identidad No. V-7179304, titulado: **"Soluciones autosimilares al colapso estelar relativista mediante Maple y GRTensorIII"**, se levanta la presente *Acta de Veredicto* del Jurado para dejar constancia de los siguientes particulares: PRIMERO: El Presidente del Jurado ciudadano Dr. Nelson L. Falcón V, plenamente identificado, después de oír los planteamientos hechos por los miembros del Jurado sobre la evaluación del precitado Trabajo de Grado, ya identificado, y estando dentro del lapso legal establecido en el artículo 140 del Reglamento de los Estudios de Postgrado de la Universidad de Carabobo, se acuerda por unanimidad: **APROBAR**, el Trabajo de Grado, titulado: **"Soluciones autosimilares al colapso estelar relativista mediante Maple y GRTensorIII"**, presentado por el ciudadano Víctor Enrique Medina Franco, cédula de identidad No.V-7179304, SEGUNDO: Notifíquese al Consejo de Postgrado y de la Facultad Experimental de Ciencias y Tecnología de la Universidad de Carabobo, de la presente decisión. No habiendo más nada que tratar, se da por terminado a los 27 días del mes de marzo del año 2025. Es todo, se leyó y conformes firman.


Dr. César Seijas
C.I. V 4567093
Miembro del Jurado


Dr. Luis A. Rodríguez
C.I.V-11678882
Miembro del Jurado


Dr. Nelson L. Falcón
V. C.I. 6101724
Presidente del Jurado



Agradecimiento

A mi hija Daniela y a mi hijo Moisés porque son la esperanza y un compromiso de dejarle un mundo mejor y más justo.

A los que siempre me apoyaron, pero que en este momento de escribir esto, no me viene tanta gente a mi memoria.

Gracias...

Cuando una persona sufre de un delirio se llama locura.

Cuando muchas personas sufren de un delirio se llama religión

Robert M. Pirsig



Universidad de Carabobo
Facultad Experimental de Ciencias y Tecnología
Departamento de Matemáticas Aplicadas



RESUMEN

El estudio del colapso gravitacional de objetos compactos de la Teoría General de la Relatividad es importante para modelar las etapas finales de la evolución estelar que dan lugar a novas y Supernovas. La obtención de soluciones autosimilares involucra, aun en los casos mas simples, la resolución de un conjunto de ecuaciones diferenciales en derivadas parciales, no lineales y acopladas. En el presente trabajo se desarrolla un algoritmo en *Maple* y *GRTensorIII* que permite calcular los elementos métricos y determinar las variables físicas relevantes. La simetría esférica reduce el sistema de ecuaciones parciales a ecuaciones diferenciales ordinarias y con ello el tratamiento numérico, empleando para ello distintas ecuaciones de estado de Schwarzschild, Tolman V y Tolman VI, que incluye el caso ultrarelativista. Los resultados muestran que al incorporar las Identidades de Bianchi, se cierra univocamente el problema, permitiendo modelar la emisión del flujo de radiación sin suponer la incorporación de un pulso gaussiano. Se concluye que el colapso estelar de esferas autogravitantes, puede ser descrito convenientemente a través de las ecuaciones de campo y las leyes de conservación, cuya integración numérica es posible mediante el algoritmo presentado

Palabras clave : Relatividad General: Esferas radiantes, Colapso Gravitacional: Supernovas, Sistemas de Computo Algebraico: Maple , GRTensorIII



Universidad de Carabobo
Facultad Experimental de Ciencias y Tecnología
Departamento de Matemáticas Aplicadas



Abstract

The study of the gravitational collapse of compact objects of the General Theory of Relativity is important for modeling the final stages of stellar evolution that give rise to novae and supernovae. Obtaining self-similar solutions involves, even in the simplest cases, the resolution of a set of coupled non-linear partial differential equations. In this work, an algorithm is developed in *Maple* and *GRTensorIII* that allows calculating the metric elements and determining the relevant physical variables. Spherical symmetry reduces the system of partial equations to ordinary differential equations and thus the numerical treatment, using for this purpose different Schwarzschild, Tolman V and Tolman VI state equations, which includes the ultrarelativistic case. The results show that by incorporating the Bianchi Identities, the problem is uniquely closed, allowing the modeling of the emission of the radiation flux without assuming the incorporation of a Gaussian pulse. It is concluded that the stellar collapse of self-gravitating spheres can be conveniently described through the field equations and conservation laws, whose numerical integration is possible through the presented algorithm

Keywords : General Relativity: Radiating spheres, Gravitational Collapse: Supernovae, Algebraic Computing Systems: Maple , GRTensorIII

Índice general

1. Introducción	1
1.1. El Problema	1
1.2. Justificación	2
1.3. Objetivos	2
1.3.1. Objetivo General	2
1.3.2. Objetivos Específicos	2
1.4. Antecedentes	2
1.5. Metodología	4
1.6. Limitaciones	4
2. Marco Teórico	6
2.1. Ecuaciones Básicas	6
2.2. Ecuaciones de Superficie	11
2.3. Modelos	13
2.3.1. Modelo tipo Schwarzschild	13
2.3.2. Modelo Tolman VI	14
2.3.3. Modelo Tolman V	14
3. Maple y GRTensorIII	17
3.1. Maple	17
3.2. GRTensorIII	19
3.3. Maple+GRTensorIII	20
3.4. Estructura del Algoritmo	20
4. Fundamentos Computacionales	22
4.1. Qt Creator	23
4.2. Algunas características del programa	23
4.3. Métodos numéricos	25
4.3.1. Ecuaciones diferenciales ordinarias	25
4.3.1.1. Método de Runge-Kutta	26
4.3.1.2. Método de Bulirsch - Stoer	27
4.3.2. Implementación de métodos numéricos	28
4.3.2.1. edovar	28

4.3.2.2.	edoprocpi	29
4.3.2.3.	libedointegra	30
4.3.2.4.	edoplot	30
4.3.2.5.	einsvar	30
4.4.	Interfaz Gráfica al Usuario (GUI)	32
4.4.1.	Implementación de la Interfaz Gráfica	32
4.4.1.1.	edowininfo	32
4.4.1.2.	varfiswin	34
5.	Resultados y Gráficas	36
5.1.	Modelo tipo Schwarzschild	36
5.2.	Modelo tipo Tolman VI	36
5.3.	Modelo Tolman V	37
6.	Conclusiones	43
	Bibliografía	45

Índice de figuras

4.1. Interfaz Grafica al Usuario (GUI) y Cálculo numérico	22
4.2. Archivos y Dependencias del programa VARFIS	24
5.1. Radio A en función de la coordenada temporal normalizada $\frac{u}{m(0)}$ en el modelo tipo Schwarzschild.	37
5.2. Valores de presión normalizados $8\pi m(0)^2 \bar{P}$ en función de la variable temporal $\frac{u}{m(0)}$ en el modelo tipo Schwarzschild.	38
5.3. Radiación $8\pi m(0)^2 \varepsilon$ para el modelo de Schwarzschild.	39
5.4. Radio A de la distribución según la variable temporal $\frac{u}{m(0)}$ en un modelo tipo Tolman VI	39
5.5. Valores de la presión normalizada $8\pi m(0)^2 \bar{P}$ en función de la variable temporal $\frac{u}{m(0)}$ en el modelo tipo Tolman VI.	40
5.6. Radiación emitida $8\pi m(0)^2 \varepsilon$ en función de la variable temporal $\frac{u}{m(0)}$ en un modelo tipo Tolman VI.	40
5.7. Evolución del radio A , en función de la variación temporal normalizada $\frac{u}{m(0)}$ (Tolman V)	41
5.8. Presión normalizada $8\pi m(0)^2 \bar{P}$ en función de la variable temporal $\frac{u}{m(0)}$ para el modelo de Tolman V	41
5.9. Radiación $8\pi m(0)^2 \varepsilon$ para el modelo de Tolman V	42

Índice de Acrónimos

ASHMEDAI	Manipulador Algebraico escrito en Lisp por M. Levine
Back-End	Capa interna del desarrollo de Software
BSD	Distribución de Software de Berkeley
CADABRA	Manipulador Algebraico. Usa TEX como Interfaz Gráfica
CAS	Computer Algebra System
EDS	Ecuaciones Diferenciales de Superficie
FORM	Manipulador Algebraico escrito en C
Front-End	Muestra al usuario la entrada o salida de los procesos realizados
Github	Repositorio de Software
GNUPLOT	Programa gráfico
GRT	Teoría de la Relatividad General
GRTensorIII	Cálculos de componentes de tensores relatividad general.
GRTensorM	Versión de GRTensor para MATHEMATICA
GUI	Interfaz Gráfica de Usuario (Graphical User Interface)
HEP	Física de Altas Energías (High Energy Physics)
IDE	Ambiente Integrado de Desarrollo
MAPLE	Manipulador Algebraico desarrollado por B. Char
MATHEMATICA	Manipulador Algebraico escrito en LISP
MAXIMA	Manipulador Algebraico escrito en LISP
MDI	Interface de Multiples Documentos
Qt Creator	Ambiente de Trabajo
REDBERRY	Manipulador Algebraico basado en Java

REDUCE	Manipulador Algebraico escrito en LISP por A. Hearn
SAGE	Manipulador Algebraico basado en Python
SCHOONSHIP	Manipulador Algebraico escrito en Assembler por M. Veltman
SDK	Herramientas de Desarrollo de Software

Lista de Publicaciones

1. *Using Maple and GRTensorIII in relativistic spherical models.* V. Medina
2022/05 <https://arxiv.org/abs/2205.11985>
2. *A self-similar solution of a fluid with spherical distribution in general relativity.* V. Medina
2022/01 <https://ui.adsabs.harvard.edu/abs/2022InJPh..96..317M>
3. *Relativistic Model for Radiating Spherical Collapse.* V. Medina y N. Falcon (14)
2018/01 <https://progress-in-physics.com/2018/PP-52-11.PDF>

Capítulo 1

Introducción

La ecuación fundamental de la Teoría de la Relatividad General es la ecuación de campo gravitatorio $G_{ab} = \kappa T_{ab}$, que realaciona la geometría del espacio-tiempo con el tensor de energía impulso de la distribución de materia. La dificultad principal para su aplicación a casos específicos reside en que las ecuaciones de campo constituyen un sistema de ecuaciones diferenciales parciales no lineales; y en consecuencia, sólo se conocen algunas pocas soluciones exactas, relevantes para el estudio de su evolución con la materia. Algunas de éstas soluciones consideran la estructura del tensor energía impulso con la materia representada por polvo, fluido perfecto, nube de cuerdas cósmicas, viscosidad aparente, etc. Éstas condiciones reducen las variables en las ecuaciones diferenciales parciales y se transforman en ecuaciones diferenciales ordinarias con menos variables que, en algunos casos especiales, se pueden resolver más fácilmente ya sean analítica o numéricamente. En el contexto de la Teoría de la Relatividad General, estas simetrías han sido ampliamente estudiadas [1, 2, 3, 4, 5, 6, 7, 8, 9]. Una de estas simetrías que juega un papel importante en situaciones cosmológicas o en el colapso gravitacional, es la simetría de auto semejanza. La principal ventaja de esta simetría es que reduce el número de variables independientes al introducir una variable auto semejante y, por tanto, reduce las ecuaciones de campo. Esta variable es un conjunto adimensional de variables independientes construidas a partir de las coordenadas del espacio y el tiempo.

1.1. El Problema

Con el desarrollo de la computación moderna, los manipuladores algebraicos (CAS), se han convertido en una herramienta fundamental en la investigación en física teórica [10, 11, 12, 13, 14]. En particular, en el campo de la relatividad general, se pueden utilizar manipuladores algebraicos para realizar cálculos tensoriales engorrosos [15, 16, 17, 18, 19]. Desde una perspectiva del usuario, esto traslada la dificultad de los cálculos de geometría diferencial a la manipulación de ciertos objetos matemáticos mediante código. Por lo tanto, los usuarios no sólo deben comprender la relación entre objetos matemáticos abstractos, sino también definir e introducir estos objetos del modo más riguroso posible. Si bien existen librerías o paquetes de códigos como *Cadabra*© [20, 21], *SageMath*© [22, 23], o varios para *Mathematica*© [24, 25, 26] o *GRTensorIII*© para *Maple*©

[27, 28, 29, 30, 18] se han utilizado extensa y fructíferamente, aun se carece de un algoritmo general que permita resolver las ecuaciones obtenidas al modelar sistemas autogravitantes en el marco de la relatividad general.

1.2. Justificación

En el presente trabajo se obtiene un conjunto de herramientas necesarias para poder desarrollar un algoritmo para el tratamiento del colapso gravitatorio en el marco de la Teoría General de la Relatividad. Al utilizar el *GRTensorIII* se calculan gran cantidad de elementos métricos y determinar varias variables físicas que se pueden estudiar con ayuda de las ecuaciones de campo gravitacional. La simetría esférica reduce el sistema de ecuaciones parciales a ecuaciones diferenciales ordinarias y con ello el tratamiento numérico. El uso de las ecuaciones de conservación permite el uso de condiciones mas cercanas a la realidad que el usado en estudios anteriores en el tema. Finalmente, el desarrollo de una interfaz gráfica nos permitirá construir las bases para una futura extensión de este trabajo para la posibilidad de aprovechar técnicas de animación asistida por computadora y poder mostrar las variables físicas calculadas en una simulación gráfica.

1.3. Objetivos

En este apartado se plantean los objetivos de esta investigación, ordenados de acuerdo a una jerarquía de objetivos específicos puestos en función de alcanzar el objetivo principal.

1.3.1. Objetivo General

- Modelar el colapso estelar de esferas autogravitantes mediante las ecuaciones de campo de Einstein y sus ecuaciones de conservación.

1.3.2. Objetivos Específicos

- Describir el colapso y evolución de objetos compactos autogravitantes a partir de las ecuaciones de campo de Einstein y las identidades de Bianchi
- Desarrollar e implementar un conjunto de herramientas de software y estructuras de datos para simular los resultados teóricos de los modelos a estudiar.
- Implementar una Interfaz Gráfica al Usuario (GUI) tanto para la entrada de los datos de integración del sistema de ecuaciones diferenciales como para los métodos de integración en lenguaje C++ para los modelos de colapso estelar relativista.

1.4. Antecedentes

El primer programa en Relatividad General (GR) fue GEOM, escrito por J.G. Fletcher [10]. Su principal habilidad era calcular el tensor de Riemann de una métrica dada. En 1969, R. A. d’Inverno [12] desarrolló ALAM (para el manipulador algebraico Atlas Lisp) [13] y lo usó para calcular

los tensores de Riemann y Ricci de la métrica de Bondi. Según [31], Bondi y sus colaboradores tardaron 6 meses en completar los cálculos originales, mientras que el cálculo de ALAM tomó 4 minutos y resultó en el descubrimiento de 6 errores en el documento original. Desde entonces, se han desarrollado numerosos paquetes y se han llevado a cabo varias investigaciones utilizando software algebraico manipulativo y, en muchos casos, los cálculos eran tan extensos que habrían sido prohibitivamente costosos de completar sin la ayuda de una computadora [32]. Su principal ventaja es la capacidad de manejar una gran cantidad de cálculos algebraicos y esta particularidad ha permitido avances en campos de la física teórica como GR o Física de Altas Energías (High Energy Physics HEP).

Muchos de estos problemas han utilizado algún software gratuito, de código abierto y de uso general con énfasis en el cálculo tensorial para GR, como REDBERRY basado en Java [33, 34], SAGE [35] y SAGEMANIFOLDS escrito en Python, CADABRA [20, 21] desarrollado en C++ y Python. Otros, aplicados específicamente para HEP, han sido diseñados en base a algoritmos especiales [36, 37, 38] e implementados en programas como SCHOONSHIP [39, 40, 41], diseñado por M. Veltman, ASHMEDAI [11, 42, 43] por M. Levine, REDUCE [44, 45, 46, 47] por A. Hearn, MACSYMA (que se convertirá en MAXIMA en 1998) [48, 49, 50] por J. Moses desarrollado en el MIT, o FORM más reciente [51, 52, 53] por J. Vermaseren. También se usa software propietario como MATHEMATICA [54, 55, 56, 26] de S. Wolfram o MAPLE [27, 57, 30, 29, 58, 59, 60, 61, 17] de B. Char desarrollado en la Universidad de Waterloo. Los programas CAS mencionados anteriormente constituyen solo una parte muy pequeña de las aplicaciones disponibles, sistemas de propósito especial y generales que se pueden consultar en repositorios y listas, mantenidos y actualizados frecuentemente [62, 63, 64, 65, 66, 67]. De todas las aplicaciones para la manipulación algebraicas, mencionadas anteriormente, nos vamos a referir en este trabajo a *MAPLE*.

Maple es un CAS de propósito general, desarrollado inicialmente en la Universidad de Waterloo como resultado de las discusiones sobre el estado de la computación simbólica en la década de 1980. En ese momento, existían grandes sistemas como *ALTRAN* [68], *CAMAL* [69, 70, 71], *REDUCE* [72] y *MACSYMA* [48], basándose en la tecnología informática de los años 60, y decidieron diseñar un nuevo sistema desde cero, aprovechando los avances en ingeniería de software disponibles y las lecciones de experiencia. Sus características básicas de diseño (por ejemplo, 73 estructuras de datos elementales, entrada/salida, aritmética numérica y simplificación elemental) están codificadas en un lenguaje de bajo nivel para mayor eficiencia.

GRTensorIII es un paquete de álgebra computacional para realizar cálculos en el área general de geometría diferencial [18]. Su principal objetivo es el cálculo de componentes tensoriales en espaciotiempos curvos especificados en términos de una métrica o conjunto de vectores base. La biblioteca se basa en una serie de comandos especiales que comienzan con "gr" (por ejemplo, *grcalc*, *grdisplay*, *gralter*, *grdefine*, etc.) para manejar una serie de objetos (pre) geométricos. definido como el tensor métrico, tensor de Ricci y escalar, tensor de Einstein, símbolos de Christoffel, etc. Esta biblioteca de objetos se puede ampliar para definir nuevos tensores, o utilizar el formalismo de Newman-Penrose. Aunque originalmente se diseñó para su uso en el campo de la relatividad general, *GRTensor* es útil en muchos otros campos [74]. Hay una versión para *MATHEMATICA* llamada *GRTensorM*. El paquete *GRTensorIII* se desarrolló originalmente para *Maple V* y se puede ejecutar con versiones desde *Maple V* Release 3 hasta *Maple 13*. La versión *GRTensor III* se ejecuta a partir de *Maple 15*. Toda la documentación y el software se distribuyen de forma

gratuita para ayudar tanto a la investigación como a la enseñanza.

1.5. Metodología

Para este trabajo, se desarrollan tres módulos independientes (*Simbólico*, *Numérico*, y *de Visualización*), una biblioteca de modelos, que se irá enriqueciendo en la medida de su implementación, y un ambiente integrado de trabajo visual usando el software GNUPLOT:

- El módulo algebraico o *Simbólico* permite al usuario, mediante manipulación simbólica, construir las expresiones analítica de los elementos métricos y sus derivadas a partir de las variables efectivas desarrollados en el algoritmo de Medina y Falcón [75] así como el sistema de ecuaciones diferenciales ordinarias para funciones evaluadas en la superficie o Sistema de Ecuaciones de Superficie (SES).
- El módulo *Numérico* permitirá, integrar numéricamente el (SES), evaluar las variables hidrodinámicas: (densidad, presión, velocidad del material, densidad de flujo de radiación, etc.) dentro de la distribución a partir de las ecuaciones de Campo Gravitacional. En esta primera versión, se utiliza un archivo con las condiciones iniciales el tipo de modelo a integrar, permitiendo el tratamiento de modelos isótropos, anisótropos, cargados eléctricamente, etc. En una versión posterior, se tratará de simular varias capas con material con ecuaciones de estado distintas. Se validará las variables físicas calculadas indicando cuando éstas no resulten ser físicamente aceptables (densidades negativas, velocidades de propagación superlumínicas etc.) y creará un archivo de salida por variable física. De este archivo de salida se podrá obtener la evolución de las variables físicas en el tiempo de Bondi (y dependiendo de la estructura de la métrica, en tiempo de Schwarzschild) para poder comparar los resultados con otros códigos o con eventuales mediciones astrofísicas. Codificado en Lenguaje C .
- El módulo de *Visualización*, permitirá analizar la evaluación de las variables hidrodinámicas dentro de la distribución material [76, 77, 78]. Permitirá la visualización (usando el GNUPLOT) y facilitar el análisis de datos calculados en el módulo *Numérico* al presentar la evolución del modelo. Se utilizará C++ para su implementación.

1.6. Limitaciones

La simetría esférica es la principal limitación del trabajo, pero permite el tratamiento semi-numérico de las últimas etapas de la evolución estelar. Sin embargo, mucho de los resultados obtenidos coinciden con los resultados obtenidos en los estudios mas generales de la Relatividad Numérica, donde se utiliza principalmente la métrica de Bondi-Sachs para establecer las ecuaciones de campo gravitacional y técnicas para resolver el sistema de ecuaciones en derivadas parciales no lineales.

La estructura de ésta memoria de investigación es la siguiente: En el capítulo 2 se mostrarán las ecuaciones constitutivas de la Relatividad General, aplicadas al colapso de esferas autogravitantes. Seguidamente, se desarrolla una metodología para resolver este conjunto de ecuaciones, dadas

diversas ecuaciones de estado (Capítulo 3). En el capítulo 4 se describe los algoritmos empleado en la codificación de *Maple* y GRTensorIII para obtener el conjunto de ecuaciones diferenciales parciales; cuya integración numericas se muestran en el capítulo 5. Finalmente, las conclusiones se muestran en el capítulo 6.

Capítulo 2

Marco Teórico

En éste capítulo se presenta una breve descripción del método utilizado para obtener las soluciones no estáticas en esferas radiantes fluidas y autogravitantes, como modelo constitutivo para el colapso gravitacional usado en la astrofísica [75, 7].

2.1. Ecuaciones Básicas

Consideremos una distribución con simetría esférica autogravitante de materia estelar en colapso. Esta distribución de materia se modela como un fluido perfecto con densidad de energía $\hat{\rho}$ y presión $\hat{P}$, visto por un observador minkowskiano local que se mueve con el fluido. Existe un flujo de radiación no polarizada, isotrópica, medida por el mismo observador. Esta densidad de energía tiene componentes radial $\hat{\varepsilon}$ y angular $\hat{\sigma}$.

En coordenadas de radiación, la métrica toma la forma.

$$ds^2 = e^{2\beta} \frac{V}{r} du^2 + 2e^{2\beta} du dr - r^2 d\theta^2 - r^2 \sin^2 \theta d\phi^2, \quad (2.1)$$

donde u y r son coordenadas temporales y radiales respectivamente; β y V son funciones de u y r ; θ, ϕ son las coordenadas angulares habituales. En estas coordenadas las ecuaciones del campo gravitatorio son:

$$-8\pi T_{00} = -\frac{V_{,0} - 2V\beta_{,0}}{r^2} - \frac{V}{r^3} (e^{2\beta} - V_{,1} + 2\beta_{,1}V), \quad (2.2)$$

$$-8\pi T_{01} = -\frac{1}{r^2} (e^{2\beta} - V_{,1} + 2\beta_{,1}V), \quad (2.3)$$

$$-8\pi T_{11} = -\frac{4\beta_{,1}}{r}, \quad (2.4)$$

$$-8\pi T_2^2 = -8\pi T_3^3 = -e^{2\beta} \left\{ 2\beta_{01} - \frac{1}{2r^2} [rV_{,11} - 2\beta_{,1}V + 2r(\beta_{,11}V + \beta_{,1}V_{,1})] \right\}. \quad (2.5)$$

Obsérvese que se han utilizado los subíndices $_{,0}$ para la derivada de la variable de tipo temporal u y $_{,1}$ para la derivada de la variable radial r . El punto y coma $_{;\alpha}$ para la diferenciación covariante. Las expresiones que relacionan las coordenadas del sistema local de Minkowski con las coordenadas

radiativas de Bondi son:

$$\begin{aligned} dt &= \left(\frac{\partial t}{\partial u} \right) du + \left(\frac{\partial t}{\partial r} \right) dr \\ &= e^\beta \left(\frac{V}{r} \right)^{\frac{1}{2}} du + e^\beta \left(\frac{r}{V} \right)^{\frac{1}{2}} dr, \end{aligned} \quad (2.6)$$

$$dx = \left(\frac{\partial x}{\partial r} \right) dr = e^\beta \left(\frac{r}{V} \right)^{\frac{1}{2}} dr, \quad (2.7)$$

$$dy = \left(\frac{\partial y}{\partial \theta} \right) d\theta = r d\theta, \quad (2.8)$$

$$dz = \left(\frac{\partial z}{\partial \phi} \right) d\phi = r \cdot \sin \theta d\phi. \quad (2.9)$$

Considerando que el material dentro de la distribución consiste básicamente en un fluido perfecto, entonces se puede expresar el tensor de energía de impulso, como:

$$\hat{T}_{\alpha\beta} = (\hat{\rho} + \hat{P}) \cdot U_\alpha U_\beta - \hat{P} \cdot \eta_{\alpha\beta}, \quad (2.10)$$

donde $U_\alpha = (1, 0, 0, 0)$, representada la velocidad para un observador que se mueve en el mismo sistema local de Minkowski, con una velocidad radial $\omega = \frac{dx}{dt}$, luego la expresión del tensor en el sistema de Lorentz, se expresa:

$$\bar{T}_{\mu\nu} = \Lambda_\mu^\alpha(\omega) \Lambda_\nu^\beta(\omega) \hat{T}_{\alpha\beta}, \quad (2.11)$$

y la matriz de transformación de Lorentz está definida como

$$\Lambda_\mu^\alpha(\omega) = \begin{bmatrix} \frac{1}{\sqrt{1-\omega^2}} & -\frac{\omega}{\sqrt{1-\omega^2}} & 0 & 0 \\ -\frac{\omega}{\sqrt{1-\omega^2}} & \frac{1}{\sqrt{1-\omega^2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.12)$$

Reemplazando los valores de densidad, presión y radiación por

$$\bar{\rho} = \hat{\rho} + 3\hat{\sigma}, \quad \bar{P} = \hat{P} + \hat{\sigma}, \quad \bar{\varepsilon} = \hat{\varepsilon} \frac{1+\omega}{1-\omega}, \quad (2.13)$$

obtenemos que los componentes contravariantes no nulos del tensor de energía total del impulso (Materia + Radiación Polarizada + Radiación No polarizada) en el sistema de Lorentz son:

$$\bar{T}_{00} = \bar{\varepsilon} + \frac{\bar{\rho} + \omega^2 \bar{P}}{1 - \omega^2}, \quad (2.14)$$

$$\bar{T}_{01} = \bar{T}_{10} = -\bar{\varepsilon} - \frac{\omega}{1 - \omega^2} (\bar{\rho} + \bar{P}), \quad (2.15)$$

$$\bar{T}_{11} = \frac{\bar{P} + \omega^2 \bar{\rho}}{1 - \omega^2} + \bar{\varepsilon}, \quad (2.16)$$

$$\bar{T}_{22} = \bar{T}_{33} = \bar{P}, \quad (2.17)$$

utilizando (2.6) - (2.9) se obtienen los componentes contravariantes del tensor de energía-impulso

en coordenadas radiativas tales como:

$$T_{00} = e^{2\beta} \left(\frac{V}{r} \right) \left(\frac{\bar{\rho} + \omega^2 \bar{P}}{1 - \omega^2} + \varepsilon \right), \quad (2.18)$$

$$T_{01} = T_{10} = e^{2\beta} \left(\frac{\bar{\rho} - \omega \bar{P}}{1 + \omega} \right), \quad (2.19)$$

$$T_{11} = e^{2\beta} \left(\frac{r}{V} \right) \left(\frac{1 - \omega}{1 + \omega} \right) (\bar{\rho} + \bar{P}), \quad (2.20)$$

$$T_{22} = \frac{T_{33}}{\sin^2 \theta} = r^2 \bar{P}. \quad (2.21)$$

Además, se puede calcular la velocidad de la materia en coordenadas radiativas y se obtiene

$$\frac{\partial r}{\partial u} = \frac{V}{r} \cdot \frac{\omega}{1 - \omega}. \quad (2.22)$$

Al definir las variables efectivas como

$$\rho \equiv \frac{\bar{\rho} - \omega \bar{P}}{1 + \omega}, \quad P \equiv \frac{\bar{P} - \omega \bar{\rho}}{1 + \omega}, \quad \varepsilon \equiv \bar{\varepsilon}. \quad (2.23)$$

Para el caso límite $\omega = 0$ o en el origen $r = 0$ es necesario que $\rho = \bar{\rho}$ y $P = \bar{P}$. Reescribiendo los componentes $T_{\mu\nu}$ en función de las variables efectivas:

$$T_{00} = e^{2\beta} \left(\frac{V}{r} \right) \left[\frac{\omega(\rho + P)}{(1 - \omega)^2} + \rho + \varepsilon \right], \quad (2.24)$$

$$T_{01} = T_{10} = e^{2\beta} \rho, \quad (2.25)$$

$$T_{11} = e^{2\beta} \left(\frac{r}{V} \right) (\rho + P), \quad (2.26)$$

$$T_{22} = \frac{T_{33}}{\sin^2 \theta} = r^2 \bar{P}, \quad (2.27)$$

y las ecuaciones de campo (2.2 -2.5) son ahora

$$-\frac{V}{r^2} \left[\left(2\beta_{,0} - \frac{V_{,0}}{V} \right) - \frac{1}{r} (2V\beta_{,1} - V_{,1} + e^{2\beta}) \right] = 8\pi e^{2\beta} \left(\frac{V}{r} \right) \left[\varepsilon + \rho + \frac{\omega(\rho + P)}{(\omega - 1)^2} \right], \quad (2.28)$$

$$2V\beta_{,1} - V_{,1} + e^{2\beta} = 8\pi r^2 e^{2\beta} \rho, \quad (2.29)$$

$$\frac{4\beta_{,1}}{r} = 8\pi \frac{r}{V} e^{2\beta} (\rho + P), \quad (2.30)$$

$$-2\beta_{,01} + \frac{1}{2r^2} [rV_{,11} - 2\beta_{,1}V + 2r(V_{,1}\beta_{,1} + V\beta_{,11})] = 8\pi e^{2\beta} \bar{P}. \quad (2.31)$$

Se obtienen tres relaciones no triviales de ecuaciones de conservación ($T^\mu_{\nu;\mu} = 0$) (Identidades de

Bianchi):

$$T_{1;\mu}^{\mu} = \frac{e^{-2\beta}}{2\pi r} \beta_{,10} - \frac{\partial P}{\partial r} - \frac{1}{2} \left(2\beta_{,1} + \frac{V_{,1}}{V} - \frac{1}{r} \right) (\rho + P) - \frac{2(P - \bar{P})}{r} = 0, \quad (2.32)$$

$$T_{0;\mu}^{\mu} = \frac{V}{r} \left\{ \left(2\beta_{,1} + \frac{V_{,1}}{V} + \frac{1}{r} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \frac{\omega}{(1 - \omega)^2} \frac{\partial(\rho + P)}{\partial r} \right\} \\ + \frac{1}{2} \left(2\beta_{,0} - \frac{V_{,0}}{V} \right) (\rho + P) + \frac{\partial \rho}{\partial u} + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} + \frac{1 + \omega}{(1 - \omega)^3} (\rho + P) \frac{\partial \omega}{\partial r} \right] = 0, \quad (2.33)$$

$$e^{2\beta} T_{;\mu}^{\mu 1} = \frac{V}{r} \left\{ \left(2\beta_{,1} + \frac{V_{,1}}{V} + \frac{1}{r} \right) \left[\varepsilon + \frac{1 + \omega^2}{2(1 - \omega)^2} (\rho + P) \right] + \right. \\ \left. + \frac{\omega(\rho + P)_{,1}}{(1 - \omega)^2} + \frac{\partial P}{\partial r} + \frac{1}{r} [\rho + \bar{P} - (P - \bar{P})] \right\} + \\ - \frac{1}{2} \left(2\beta_{,0} - \frac{V_{,0}}{V} \right) (\rho + P) + \frac{\partial P}{\partial u} + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} + \frac{1 + \omega}{(1 - \omega)^3} (\rho + P) \frac{\partial \omega}{\partial r} \right] = 0. \quad (2.34)$$

Comparando las ecuaciones anteriores (2.32-2.34), se puede ver que dos de ellas son independientes, ya que existe una relación entre ellas:

$$e^{2\beta} T_{;\mu}^{\mu 1} - T_{0;\mu}^{\mu} = \left(\frac{V}{r} \right) T_{1;\mu}^{\mu} = 0. \quad (2.35)$$

y la traza de la ecuación de campo se escribe como:

$$8\pi [(\rho - \bar{P}) + (P - \bar{P})] = 4\beta_{,01} e^{-2\beta} - \frac{e^{-2\beta}}{r^2} [rV_{,11} - 2V\beta_{,1} + \\ + 2r(V_{,1}\beta_{,1} + V\beta_{,11})] - \frac{2e^{-2\beta}}{r^2} (V_1 - e^{2\beta}). \quad (2.36)$$

La traza de la ecuación de campo es importante porque se puede utilizar como alternativa a algunas de las ecuaciones de conservación. En particular, se puede demostrar que es equivalente a (2.32).

Al definir el aspecto de masa de Bondi como $V \equiv e^{2\beta} (r - 2m)$, y después de algo de álgebra, podemos escribir las ecuaciones del campo gravitacional como:

$$m_{,1} = 4\pi r^2 \rho, \quad (2.37)$$

$$\beta_{,1} = 2\pi r \frac{(\rho + P)}{1 - \frac{2m}{r}}, \quad (2.38)$$

$$m_{,0} = -4\pi r^2 e^{2\beta} \left(1 - \frac{2m}{r} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right], \quad (2.39)$$

$$8\pi \bar{P} = -2\beta_{,01} e^{-2\beta} + \left(1 - \frac{2m}{r} \right) \left(2\beta_{,11} + 4\beta_{,1}^2 - \frac{\beta_{,1}}{r} \right) + \frac{1}{r} [3\beta_{,1} (1 - 2m_{,1}) - m_{,11}] \quad (2.40)$$

Además, la versión correspondiente para la expresión de la traza de las ecuaciones de campo:

$$4\pi [(\bar{P} + P) + (\bar{P} - \rho)] = -2\beta_{,01}e^{-2\beta} + \left(1 - \frac{2m}{r}\right) \left(2\beta_{,11} + 4\beta_{,1}^2 + \frac{\beta_{,1}}{r}\right) + \frac{1}{r} [3\beta_{,1}(1 - 2m_{,1}) - m_{,11}] - \frac{2m_{,1}}{r^2}, \quad (2.41)$$

y tomando como ecuaciones de conservación independientes (2.32) y (2.33), tenemos

$$-\frac{e^{-2\beta}}{2\pi r}\beta_{,10} + \frac{\partial P}{\partial r} + \left(4\pi r^2 P + \frac{m}{r}\right) \frac{(\rho + P)}{r(1 - \frac{2m}{r})} + \frac{2}{r}(P - \bar{P}) = 0, \quad (2.42)$$

$$\begin{aligned} & \frac{e^{2\beta}}{r} \left[1 + \left(1 - \frac{2m}{r}\right) + 4\pi r^2 (P - \rho)\right] \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2}\right] + \\ & + \frac{V}{r} \frac{\omega}{(1 - \omega)^2} \frac{\partial}{\partial r} (\rho + P) + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} + \frac{(1 + \omega)(\rho + P)}{(1 - \omega)^3} \frac{\partial \omega}{\partial r}\right] + \frac{\partial \rho}{\partial u} = 0. \end{aligned} \quad (2.43)$$

En este punto es necesario destacar lo siguiente:

- Se puede demostrar que la expresión de la traza de la ecuación de campo (2.41), es equivalente a la ecuación (2.42), similar a la ecuación de equilibrio hidrostático de TOV, (ver, por ejemplo [79]).
- Respecto a la ecuación de conservación (2.43), también se puede obtener a partir de las ecuaciones de campo (2.37) y (2.39) y observando que las segundas derivadas mixtas conmutan; es decir, $m_{,01} = m_{,10}$.

Ahora, combinando (2.39) con (2.43)

$$\begin{aligned} \frac{2m_{,0}e^{-2\beta}}{4\pi r^3(1 - \frac{2m}{r})} \left[1 - \frac{m}{r} + 2\pi r^2 (P - \rho)\right] &= \frac{\omega}{(1 - \omega)^2} \left(1 - \frac{2m}{r}\right) (\rho + P)_{,1} + r \left[\frac{\partial \varepsilon}{\partial r} + \right. \\ & \left. \frac{(1 + \omega)(\rho + P)}{(1 - \omega)^3} \frac{\partial \omega}{\partial r}\right] + \frac{e^{2\beta}}{4\pi r^2} m_{,10}. \end{aligned} \quad (2.44)$$

Considerando que los perfiles de radiación de ε y el valor de ω varían poco, se puede escribir una expresión muy similar a la ecuación de Euler:

$$\begin{aligned} 2m_{,0} \left[1 - \frac{m}{r} + 2\pi r^2 (P - \rho)\right] &= \frac{\omega \cdot e^{2\beta}}{(1 - \omega)^2} \left[4\pi r^2 \left(1 - \frac{2m}{r}\right)^2 r (\rho + P)_{,1}\right] + \\ & + r \left(1 - \frac{2m}{r}\right) m_{,10}. \end{aligned} \quad (2.45)$$

Una expresión similar a la anterior, se puede encontrar en [80], aunque no fue utilizada en ningún otro trabajo posterior. La ecuación (2.45) permite relacionar el intercambio de masa con las otras magnitudes del sistema autogravitante y se omite en trabajos previos sobre la evolución de esferas de fluidos radiantes [81, 82, 83, 84, 85, 80]. Esta omisión impide completar el sistema de ecuaciones y motiva la inclusión de una función ad-hoc para la luminosidad [81, 84, 83, 86, 87, 88]. Con las

ecuaciones de campo (2.37)-(2.40) podemos calcular las expresiones de las variables físicas ω , $\bar{P}$, $\bar{\rho}$, si conocemos las expresiones $m(u, r)$ y $\beta(u, r)$ en cada capa del material en estudio. Es por esto que las EOS $P(u, r)$ $\rho(u, r)$ son de gran importancia para determinar el comportamiento de las variables físicas presentes en el campo de ecuaciones y su posterior evolución.

2.2. Ecuaciones de Superficie

La inexistencia de un observador privilegiado en el marco de la Relatividad General de la Relatividad hace necesario definir marcadores de Superficie , que son parámetros tales como el radio, la masa y la velocidad en la superficie.

Considerando las ecuaciones de campo (2.37) y (2.38) podemos realizar la integración y obtener los valores de m y β en función de la coordenada radial:

$$m = \int_0^r 4\pi r^2 \rho dr, \quad (2.46)$$

$$\beta = 2\pi \int_a^r \frac{\rho + P}{1 - \frac{2m}{r}} r \cdot dr. \quad (2.47)$$

Con estas dos ecuaciones, y conociendo la dependencia radial de P y ρ , se pueden obtener las expresiones de las funciones métricas m y β excepto su dependencia temporal que está relacionada con las condiciones de contorno. Aquí para determinar estas funciones, dependientes de la variable temporal u , se utiliza una modificación del método utilizado en [75] para obtener una solución no estática, que se puede resumir en:

1. Supóngase que las variables efectivas P y ρ (eq. 2.23) tienen la misma dependencia radial que en la solución estática, pero observando que ahora la condición de contorno $\bar{P}_a = 0$ ahora se expresa como

$$P_a = -\omega_a \rho_a. \quad (2.48)$$

El subíndice $_a$ indica que la cantidad se evalúa en el límite de la distribución.

2. Utilizando las expresiones de las variables efectivas y con (2.46) y (2.47) se calculan los valores de m y β excepto para tres variables de superficie que se ajustan mediante las condiciones de borde de distribución del material tanto para las ecuaciones de campo como para las ecuaciones de conservación.
3. Eq. (2.22) evaluada en $r = a$,
4. Eq. (2.45) evaluada en $r = a$,
5. En lugar de utilizar la ecuación de conservación (2.32) evaluada en a , utilizaremos la ecuación de la traza de las ecuaciones de campo, (2.41) evaluada en a ,
6. Seleccionando un conjunto de datos iniciales, para realizar la integración numérica de la EDO establecida en (2) se conocen en su totalidad las funciones m y β
7. Utilizando las ecuaciones (2.37) a (2.40), se calculan los valores de las variables físicas en cada sección o capa de la distribución del material.

Siguiendo el procedimiento establecido en (2) vamos a definir las variables de superficie así como las ecuaciones diferenciales que controlan su variación temporal.

- Como se indica en (3), evaluando la expresión (2.22) en $r = a$, obtenemos una de las ecuaciones de superficie

$$\dot{a} = \frac{\partial a}{\partial u} = \left(1 - \frac{2m_a}{a}\right) \left(\frac{\omega_a}{1 - \omega_a}\right). \quad (2.49)$$

Normalizando, respecto de la masa inicial $m(0) = m(u = 0, r = a)$ de la distribución del material, las variables superficiales definidas como:

$$A \equiv \frac{a}{m(0)} \quad M \equiv \frac{m_a}{m(0)} \quad \Omega \equiv \frac{1}{1 - \omega_a}. \quad (2.50)$$

Distinguiendo con un círculo en la parte superior de la variable $(\dot{\Delta})$, la derivada con respecto a u y con un punto $(\dot{\Delta})$ la derivada con respecto a la variable de tiempo normalizada $\left(\frac{u}{m(0)}\right)$, podemos escribir la ecuación (2.49) de la siguiente manera

$$\dot{A} = \left(1 - \frac{2M}{A}\right) (\Omega - 1), \quad (2.51)$$

Esta ecuación es válida para todos los modelos.

- Continuando con (4), al evaluar la ecuación (2.45) en $r = a$, es necesario calcular la derivada de la densidad y la presión efectiva con respecto a r ; condición que la hace dependiente del modelo.

$$\begin{aligned} 2m_{,0}|_a \left\{1 - \frac{m}{r} + r\beta_{,1} \left(1 - \frac{2m}{r}\right) - m_{,1}\right\}_a - r \left(1 - \frac{2m}{r}\right) \frac{\partial}{\partial u} (4\pi r^2 \rho) \Big|_a &= \\ &= \Omega (\Omega - 1) \left[4\pi r^2 \left(1 - \frac{2m}{r}\right)^2 r \frac{\partial}{\partial r} (\rho + P)\right]_a. \end{aligned} \quad (2.52)$$

- Finalmente, con la condición establecida en (5), evaluamos la expresión de la traza de las ecuaciones de campo (2.41) en $r = a$ para obtener otra ecuación para las variables de superficie. Al igual que la ecuación anterior, también depende del modelo.

$$\begin{aligned} 4\pi (P - \rho)_a &= \frac{1}{a} [3\beta_{,1} (1 - 2m_{,1}) - m_{,11}]_a - \frac{2m_{,1}}{r^2} \Big|_a - 2\beta_{,01}|_a + \\ &+ \left(1 - \frac{2M}{A}\right) \left(2\beta_{,11} + 4\beta_{,1}^2 + \frac{\beta_{,1}}{r}\right) \Big|_a. \end{aligned} \quad (2.53)$$

En términos de variables de superficie, las ecuaciones (2.52) y (2.53) son similares y pueden escribirse como:

$$\Upsilon_M \dot{A} + \Xi_M \dot{M} + \Lambda_M \dot{\Omega} = \Delta_M, \quad (2.54)$$

$$\Upsilon_\Omega \dot{A} + \Xi_\Omega \dot{M} + \Lambda_\Omega \dot{\Omega} = \Delta_\Omega, \quad (2.55)$$

donde

$$\begin{aligned}\Upsilon_\xi &\equiv \Upsilon_\xi(A, M, \Omega) \quad , \quad \Xi_\xi \equiv \Xi_\xi(A, M, \Omega) \\ \Lambda_\xi &\equiv \Lambda_\xi(A, M, \Omega) \quad , \quad \Delta_\xi \equiv \Delta_\xi(A, M, \Omega) \quad \forall \xi \in \{M, \Omega\},\end{aligned}$$

son funciones de (A, M, Ω) . Como se indicó en la subsección 6, el conjunto de tres ecuaciones (2.49), (2.52) y (2.53) forman un sistema de Ecuaciones Diferenciales Ordinarias (EDO). Junto con las condiciones iniciales, se puede integrar numéricamente y determinar completamente m y β . En la siguiente sección, se muestra la aplicación de este algoritmo para diversas ecuaciones de estado constitutivas que relacionen las variables de Presión y densidad, en el colapso estelar de esferas autogravitantes.

2.3. Modelos

2.3.1. Modelo tipo Schwarzschild

Obtendremos como primer ejemplo de prueba, la conocida solución de densidad constante estática e interna, de Schwarzschild. Para ello, vamos a suponer que la densidad depende únicamente de la variable de tipo temporal, como se explica en [81, 89]

Ecuaciones de Estado: Podemos escribir la ecuación de estado para el modelo tipo Schwarzschild como

$$\rho = \frac{3m}{4\pi r^3} \quad (2.56)$$

$$P = \rho \left\{ \frac{1 - \frac{1}{g} \left[\frac{1 - \frac{2M}{A} \left(\frac{r}{a} \right)^2}{1 - \frac{2M}{A}} \right]^{\frac{1}{2}}}{\frac{1}{g} \left[\frac{1 - \frac{2M}{A} \left(\frac{r}{a} \right)^2}{1 - \frac{2M}{A}} \right]^{\frac{1}{2}} - 3} \right\} \quad (2.57)$$

Condición de Contorno : El valor de g se determina a partir de la condición de contorno ($\bar{P}_a = 0$) entonces la presión efectiva satisface la relación (2.48); y en consecuencia $g = \frac{1}{3-2\Omega}$.

Ecuaciones de Superficie: Evaluando las ecuaciones (2.22), en $r = a$, obtenemos (2.51) y con (2.42) y (2.43) en $r = a$ obtenemos entonces

$$\dot{A} = \left(1 - \frac{2M}{A}\right) (\Omega - 1), \quad (2.58)$$

$$\dot{M} = 3 \left(\frac{M}{A}\right)^2 \left(1 - \frac{2M}{A}\right) \frac{(\Omega - 1)(2\Omega - 3)}{\frac{3M}{A} - \Omega \left(1 + \frac{2M}{A}\right)} \quad (2.59)$$

$$\begin{aligned}\dot{\Omega} &= \frac{2\Omega(1 - \Omega)}{A} \left(1 - \frac{2M}{A}\right) - \frac{1}{A} \left(\frac{M}{A}\right) \left(\frac{3 - 2\Omega}{1 - \frac{2M}{A}}\right) \dot{A} + \\ &\quad - \frac{\Omega}{M} \frac{1}{\left(1 - \frac{2M}{A}\right)} \dot{M}\end{aligned} \quad (2.60)$$

usando (2.38) y (2.37) obtenemos, luego de la integración inmediata

$$m(r) = m(0) \cdot M \left(\frac{r}{a} \right)^3 \quad (2.61)$$

$$\beta = \frac{1}{2} \log \left[1 + \frac{3}{2\Omega} \left(\sqrt{\frac{1 - \frac{2M}{A}}{1 - \frac{2M}{A} \left(\frac{r}{a} \right)^2}} - 1 \right) \right]. \quad (2.62)$$

2.3.2. Modelo Tolman VI

Ecuaciones de Estado: Siguiendo [90] podemos asumir como solución estática

$$4\pi a^2 \rho = 3h \left(\frac{a}{r} \right)^2 \quad (2.63)$$

$$4\pi a^2 P = h \left(\frac{a}{r} \right)^2 \left[\frac{1 - 9 \cdot z \left(\frac{r}{a} \right)}{1 - z \left(\frac{r}{a} \right)} \right], \quad (2.64)$$

Condición de Contorno : Como antes, el valor de z se determina a partir de la condición de contorno ($\bar{P}_a = 0$) entonces $P_a = -\omega_a \rho_a$; y en consecuencia $z = \frac{4\Omega-3}{3(4\Omega-1)}$; y $h = \frac{m}{3r}$.

Ecuaciones de Superficie: Evaluando las ecuaciones (2.22), (2.42) y (2.43) at $r = a$ obtenemos

$$\dot{A} = \left(1 - \frac{2M}{A} \right) (\Omega - 1), \quad (2.65)$$

$$\dot{M} = -\frac{\left(1 - \frac{2M}{A} \right)^2 (16\Omega^2 + 3) \left(\frac{M}{A} \right)}{8 \left[\Omega \left(1 - \frac{2M}{A} \right) + \frac{M}{A} \right]} \dot{A} \quad (2.66)$$

$$\begin{aligned} \dot{\Omega} = & -\frac{1}{A} \left[\frac{M}{A} - \frac{(4\Omega-3)(4\Omega-1)}{8} \left(1 - \frac{2M}{A} \right) \right] \\ & + (4\Omega-1)(4\Omega-3) \frac{\dot{A}}{8A} + \frac{\Omega}{\left(1 - \frac{2M}{A} \right)} \frac{\dot{M}}{M}, \end{aligned} \quad (2.67)$$

y los valores correspondiente de β y m son

$$\beta = \frac{2M}{3A} \frac{1}{\left(1 - \frac{2M}{A} \right)} \left\{ \log \left(\frac{r}{a} \right) + 2 \log \left[\frac{3 - \left(\frac{4\Omega-3}{4\Omega-1} \right) \left(\frac{r}{a} \right)}{3 - \left(\frac{4\Omega-3}{4\Omega-1} \right)} \right] \right\} \quad (2.68)$$

$$m = m(0) M \left(\frac{r}{a} \right). \quad (2.69)$$

2.3.3. Modelo Tolman V

Se presenta explícitamente otro ejemplo que ilustra el método descrito en la sección 2.2. Es necesario establecer la Ecuación de Estado para utilizar las ecuaciones (2.46) y (2.47) y calcular la dependencia radial de $m(r)$ y $\beta(r)$.

Ecuaciones de Estado : El modelo se obtiene específicamente de la solución de Tolman V [90]

y tomando esta solución estática como punto de partida, podemos escribir

$$8\pi r^2 \rho = \frac{3}{7}h + \frac{10z}{3} \left(\frac{r}{a}\right)^{\frac{7}{3}}, \quad (2.70)$$

$$8\pi r^2 P = \frac{1}{7}h - 2z \left(\frac{r}{a}\right)^{\frac{7}{3}}. \quad (2.71)$$

Condición de Contorno : La condición de contorno, ($\bar{P}_a = 0$ o su equivalente $P_a = -\omega_a \rho_a$) nos permite fijar los valores de $z = \frac{1}{7}(4\Omega - 3) \cdot \left(\frac{2M}{3A}\right)$ y $h = -2(2\Omega - 5) \cdot \left(\frac{2M}{3A}\right)$, como función de las variables de superficie (A, M, Ω) . Las ecuaciones de estado para la densidad y la presión efectiva se pueden escribir

$$4\pi r^2 \rho = \left(\frac{2M}{3A}\right)(2\Omega - 5) + \frac{5}{3} \left(\frac{2m}{r}\right), \quad (2.72)$$

$$4\pi r^2 P = \left(\frac{2M}{3A}\right)(5 - 2\Omega) - \left(\frac{2m}{r}\right). \quad (2.73)$$

Ecuaciones de Superficie: Siguiendo el punto 2 de la sección 2.2, tenemos la ecuación de superficie para la variable A en (2.51) y en la evaluación de las ecuaciones (2.42) y (2.45) en $r = a$,

$$\dot{A} = \left(1 - \frac{2M}{A}\right)(\Omega - 1), \quad (2.74)$$

$$\dot{M} = -\frac{M}{6A} \left(1 - \frac{2M}{A}\right) \left[3(4\Omega - 3) - 2 \left(\frac{2M}{A}\right)(4\Omega - 1)\right], \quad (2.75)$$

$$\dot{\Omega} = \left[1 - \frac{2}{3}(2\Omega - 3)^2\right] \frac{\dot{A}}{2A} - \left(\frac{M}{A} + 2\Omega - 3\right) \frac{\dot{M}}{2M \left(1 - \frac{2M}{A}\right)}, \quad (2.76)$$

Cuando se realiza la integración numérica de (2.74), (2.75) y (2.76) con un conjunto de datos inicial, se tienen los valores de las funciones métricas β y m como

$$\beta = -\frac{1}{7 + 2(2\Omega - 5) \cdot \left(\frac{2M}{A}\right)} \cdot \left[2(2\Omega - 5) \left(\frac{2M}{3A}\right) \log\left(\frac{r}{a}\right) + \right. \\ \left. + \log\left(\frac{1 - \frac{2M}{A} \left\{1 + \frac{(4\Omega - 3)}{7} \left[\left(\frac{r}{a}\right)^{\frac{7}{3}} - 1\right]\right\}}{1 - \frac{2M}{A}}\right)\right], \quad (2.77)$$

$$m = m(0) \left(\frac{r}{a}\right) M \left\{1 + \frac{(4\Omega - 3)}{7} \left[\left(\frac{r}{a}\right)^{\frac{7}{3}} - 1\right]\right\}. \quad (2.78)$$

Integrando numéricamente las ecuaciones (2.74), (2.75) y (2.76), con un conjunto inicial de datos, las variables superficiales así determinadas pueden emplearse para calcular los valores de las variables físicas con el procedimiento establecido en (7) de la sección (2.2), en las diferentes capas de la distribución del material. En las gráficas de las variables superficiales (A, M, Ω) y de las variables físicas (presión, densidad, radiación, etc.) se observa un comportamiento exponencial (dispersión) u oscilaciones amortiguadas. La forma dispersiva o amortiguada del comportamiento temporal de las variables superficiales o físicas depende mucho del conjunto de valores iniciales utilizados en la simulación.

En el siguiente capítulo se describen los algoritmos utilizados para la solución numéricas.

Las soluciones numéricas para los tres casos de Ecuaciones de Estado previamente referidos, se mostrarán en el capítulo 5.

Capítulo 3

Maple y GRTensorIII

Este capítulo proporcionará una breve introducción de cómo se pueden utilizar *GRTensorIII* y su motor algebraico *Maple* para realizar cálculos tensoriales en Relatividad General. Hay una serie de programas que en diversos grados, pueden manipular expresiones tensoriales, incluyendo *Cadabra* [91, 21], *MathTensor* [92, 32], *Canon* [93], *Riemann* [57], *xAct* [94]. Se pretende mostrar cómo usar *GRTensorIII* para calcular el colapso gravitacional en Relatividad General.

3.1. Maple

Maple es un lenguaje y sistema para el cálculo matemático simbólico, desarrollado en la Universidad de Waterloo a partir de diciembre de 1980. El nombre **Maple** no es un acrónimo, sino que es simplemente una referencia a la identidad Canadiense. El tipo de cálculo proporcionado por Maple es conocido por varios otros nombres como *manipulación algebraica* o *álgebra computacional* [95, 27]. Se puede usar interactivamente como una calculadora matemática y los procedimientos computacionales se pueden escribir usando el lenguaje de programación Maple de alto nivel.

La motivación principal del desarrollo de Maple en los años 1980 era lo poco accesible de la computación simbólica. Y para tener acceso a un sistema poderoso como MACSYMA [48, 49, 50] era necesario tener una computadora central de alto desempeño, accesible solo a un reducido número de usuarios. En un entorno universitario, esto significaba que no era factible ofrecer computación simbólica a grandes grupos de usuarios potencialmente interesados. El desarrollo de los sistemas *muMATH* [96] y *PICOMATH* [97] mostró que se podría proporcionar una capacidad de cálculo simbólico importante en microcomputadoras de espacio de direcciones pequeñas y de bajo costo. Estos sistemas fueron los precursores y mostraban que era factible diseñar un sistema con una gama completa de capacidades para el cálculo simbólico, que no estuviese restringido por el pequeño espacio de direcciones de los primeros microcomputadores, ni inaccesible debido a la demanda de los recursos de hardware. Estaba bien claro que parecía posible diseñar un sistema modular *compacto* cuyas demandas de memoria crecerían correctamente con las necesidades del programa de aplicación. Otra de las preocupaciones es la *portabilidad*, ya que se vislumbraba la necesidad de instalar el programa a una gran variedad de sistemas informáticos que entrarían en el mercado en la década de los años ochenta. Se reconoció que la *accesibilidad* se ve muy afectada

por la calidad de la interfaz de usuario que proporciona un sistema. Por lo tanto, los principales objetivos de diseño del sistema Maple son: compacidad, con un conjunto poderoso de instrucciones para computación simbólica; portabilidad y una buena interfaz de usuario. En detalle, podemos establecer las principales características del sistema Maple en:

1. *Kernel ó Núcleo: (Compacidad)*: Originalmente el núcleo del sistema estaba escrito en un lenguaje similar a BCPL. Se utilizó un macroprocesador para generar código para varios lenguajes de implementación en la familia BCPL (en particular, C). En la actualidad toda la compilación se realiza en lenguaje C. El núcleo compilado del sistema Maple es relativamente compacto (aproximadamente 100K bytes, dependiendo del sistema operativo y del compilador). Este núcleo incluye el intérprete para el lenguaje Maple, la aritmética básica, el espacio para las tablas y matrices, las rutinas de impresión (incluida la visualización bidimensional), la simplificación básica y las funciones básicas (tales como el *coeff*, *degree*, *map*, y *divide*). Algunas funciones (como *expand*, *dif* (diferenciación) y *taylor*) tienen un *núcleo* en el kernel y carga automáticamente las rutinas externas de la biblioteca para las extensiones. Las operaciones matemáticas de nivel superior (como *gcd*, *int* (integrar) y *solve*, están completamente en la biblioteca y se cargan solo cuando se les necesitan.
2. *La biblioteca*: contiene la mayoría de las rutinas de Maple. Incluye funciones relacionadas con el cálculo, el álgebra lineal, estadísticas, gráficos y muchos otros temas. La biblioteca de Maple consiste en rutinas individuales y paquetes de rutinas. El sistema Maple accede e interpreta estas rutinas según sea necesario. Como resultado de esta modularidad, la computadora consume recursos proporcionales solo a las instalaciones que se utilizan, lo que mejora la eficiencia del sistema. Los usuarios pueden ver y modificar todas las rutinas de la biblioteca, que se implementan en el lenguaje de programación de alto nivel de Maple. Por lo tanto, es útil aprender el lenguaje de programación Maple para que pueda modificar el código Maple existente para producir rutinas personalizadas.
3. *Interfaz de usuario*: Maneja la entrada de expresiones y comandos matemáticos, la visualización de resultados y el control de las opciones de entorno de la hoja de trabajo de Maple.

Debido a la gran cantidad de compañías de software en crecimiento, en los años de su introducción y desarrollo, el sistema Maple fue diseñado para permitir la portabilidad a una variedad de diferentes sistemas operativos. Además incorpora un amplio conjunto de conocimientos matemáticos a través de una biblioteca de funciones. Las funciones de la biblioteca están codificadas en el lenguaje de programación Maple a nivel de usuario que fue diseñado para facilitar la expresión y la ejecución eficiente de operaciones matemáticas. Una consecuencia del diseño de Maple es la extensibilidad del usuario ya que las funciones definidas por el usuario tienen el mismo estado que las funciones de la biblioteca del sistema. Estos objetivos de diseño llevaron a varias características novedosas. Las estructuras de datos fundamentales de Maple son objetos etiquetados representados internamente como vectores dinámicos (matrices de longitud variable). Específicamente, cada instancia de una estructura de datos es un vector en el que el primer componente codifica la siguiente información: la longitud de la estructura, el tipo de objeto de datos (como suma, producto, conjunto, número racional, etc.), el estado de la simplificación y la recolección de basura. Los otros componentes del vector son los operandos del objeto de datos y suelen ser indicadores a estructuras de datos

similares. Hay un uso extensivo del hash para minimizar los requisitos de almacenamiento (asegurando que los objetos equivalentes se almacenan solo una vez) y para minimizar el tiempo de ejecución (al recordar el resultado de las operaciones críticas). Se utiliza un intérprete rápido para ejecutar los códigos de la biblioteca, lo que permite que el *núcleo* compilado del sistema incluya solo aquellas operaciones que se han identificado como cuellos de botella. Otra característica importante diseñada para hacer que Maple sea fácil de usar es que se han evitado las "banderas" globales en la medida de lo posible, mediante el diseño de algoritmos híbridos que incorporan la selección automática de algoritmos. Se puede encontrar más información técnica importante sobre el diseño del sistema Maple en Char et al. [95, 27]. Al inicio de su desarrollo y para comprobar su portabilidad, se instaló en una variedad de computadoras centrales (VAX VMS y Berkeley Unix, IBM VM/CMS, DEC20 TOPS-20) y microsistemas (MASSCOMP, Microvax, Cadmus, y otros microcomputadoras con sistemas operativos similares a Unix). Un usuario que invoca a Maple en cualquiera de estos sistemas provoca la carga de un módulo binario (aproximadamente 140 kilobytes) denominado *MathSoft Kernel Maple (MKM)* y la creación de algunos objetos iniciales. El núcleo contiene código para hacer aritmética de enteros, racionales, polinomios y series, el intérprete del lenguaje de programación Maple, las rutinas de entrada/salida y otros servicios esenciales del sistema. El código fuente para el núcleo está escrito en términos de macros que permiten reorientar a una variedad de dialectos de programación tipo C, compiladores, sistemas operativos, etc. La funcionalidad matemática de Maple (por ejemplo, el código de integración, el cálculo de series de Taylor, el álgebra lineal, la solución de ecuaciones algebraicas, etc.) reside en su directorio de bibliotecas¹. Cuando el usuario requiere una funcionalidad particular para un cálculo, los archivos apropiados se cargan automáticamente en la memoria principal para que el intérprete de Maple los ejecute.

Con el paso del tiempo y las mejoras en las versiones posteriores del sistema, se ha utilizado este núcleo Maple (*MathSoft Kernel Maple*) en programas como *MathCad* y *Matlab* como motor algebraico ó inicialmente en aplicaciones a la relatividad general como en XLOOPS [98, 99]. Otra aplicación interesante es el diseño y uso de *Distributed Maple*, un entorno para ejecutar en paralelo programas de álgebra computacional en multiprocesadores y clusters heterogéneos. El sistema inserta núcleos del sistema de álgebra computacional Maple como motores computacionales en una red y la coordinación de los clusters se ha implementado en el lenguaje de programación Java [59]. Sin embargo, en este trabajo nos limitaremos a aplicar y explorar la relación existente de Maple como motor algebraico para GRTensorIII.

3.2. GRTensorIII

GRTensorIII [15, 100] es un paquete de álgebra computacional para realizar cálculos en el área general de la geometría diferencial. Tiene alrededor de diecisiete mil líneas de código Maple y presenta una versión para Mathematica (GRTensorM). Su propósito es el cálculo de componentes tensoriales en espacio-tiempo curvo especificados en términos de una métrica o un conjunto de vectores de base. Se le han dedicado muchos esfuerzos para presentar una interfaz gráfica muy cuidada y robusta. Los índices contravariantes se presentan de manera adecuada como superíndi-

¹La suma total de los códigos fuente de la biblioteca, si se cargaron todos (un evento improbable), es aproximadamente 1.5 megabytes.

ces y los covariantes como subíndices. El paquete contiene una biblioteca de definiciones estándar con un gran número de aplicaciones y es fácilmente expandible al poder definir nuevos tensores y adicionalmente se pueden realizar cálculos en geometría de n dimensiones espacio-tiempo, siempre que sean de dimensión finita. Aunque originalmente fue diseñado para su uso en el campo de la relatividad general, GRTensorIII es útil en muchos otros campos [74]. Fue desarrollado originalmente para Maple V y se puede ejecutar con todas las versiones de Maple, desde Maple V Release 3 hasta Maple 16. La distribución del software y su documentación es gratuita a partir de la página web GRTensor, como ayuda para la enseñanza e investigación.

3.3. Maple+GRTensorIII

Aquí describiremos brevemente la estructura y las características principales de los procedimientos de Maple para el formalismo de obtención de componentes de las ecuaciones de campo de Einstein y las ecuaciones de conservación – Identidades de Bianchi –, como el utilizado en Medina-Falcon [75]. Para realizar este objetivo, necesitamos establecer el tensor métrico (métrica) del espacio tiempo a estudiar. Cuando el objetivo es el cálculo de componentes de objetos indexados, tensores o la definición de nuevos tensores, debemos en primer lugar, especificar la geometría del espacio. Una forma de especificar una geometría de espacio es usar la función *makeg()*. Esta función se puede usar para ingresar toda la información necesaria para especificar una métrica de coordenadas o una base (un conjunto de n vectores linealmente independientes relacionados por un producto interno definido por el usuario). Las métricas creadas se pueden guardar en archivos ASCII. Estos archivos se pueden cargar en GRTensorIII usando los comandos *qload()* o *grload()*. Este enfoque de cargar las métricas es el que se ha utilizado en ésta investigación. Aquí el objetivo es presentar los cálculos de, con énfasis en los métodos, paquetes y técnicas de álgebra computacional.

3.4. Estructura del Algoritmo

El algoritmo de obtención del cálculo de las ecuaciones de campo ($G_{\mu\nu} = \kappa T_{\mu\nu}$) y las ecuaciones de conservación ($T^{\mu\nu}_{;\nu} = 0$), se encuentra en el apéndice o en el repositorio Arxiv con la dirección Medina2022. Ahí se puede observar la contrucción del tensor energía - impulso y su expresión del cambio de coordenadas en la métrica de Bondi. Sin embargo, se puede cambiar la forma de éste calculo, simplemente cambiando la métrica que el usuario necesite. Finalmente la expresión para

nuestro interés en la métrica de Bondi, resulta en:

$$\begin{aligned} T_{0;\lambda}^{\lambda} &= \frac{V}{r^2} \left(1 + \frac{rV_{,1}}{V} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \\ &+ \frac{V}{r} \frac{\partial}{\partial r} \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \frac{\partial}{\partial u} \left(\rho + \frac{Q^2}{8\pi r^4} \right) = 0 \end{aligned} \quad (3.1)$$

$$\begin{aligned} T_{1;\lambda}^{\lambda} &= \frac{e^{-2\beta}}{2\pi r} \beta_{,10} - \frac{\partial}{\partial r} \left(P - \frac{Q^2}{8\pi r^4} \right) - \frac{1}{2} \left(2\beta_{,1} + \frac{V_{,1}}{V} - \frac{1}{r} \right) (\rho + P) + \\ &- \frac{2}{r} \left[\left(P - \frac{Q^2}{8\pi r^4} \right) - \left(P_t + \frac{Q^2}{8\pi r^4} \right) \right] = 0 \end{aligned} \quad (3.2)$$

$$T_{;\lambda}^{\lambda 0} = e^{-2\beta} T_{1;\lambda}^{\lambda} = 0 \quad (3.3)$$

$$\begin{aligned} e^{2\beta} T_{;\lambda}^{\lambda 1} &= \frac{V}{r^2} \left(1 + 4r\beta_{,1} + \frac{rV_{,1}}{V} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \frac{V}{2r} \left(2\beta_{,1} + \frac{V_{,1}}{V} - \frac{1}{r} \right) (\rho + P) + \\ &+ \frac{2V}{r^2} \left[\left(P - \frac{Q^2}{8\pi r^4} \right) - \left(P_t + \frac{Q^2}{8\pi r^4} \right) \right] + \\ &\frac{V}{r} \frac{\partial}{\partial r} \left(P - \frac{Q^2}{8\pi r^4} \right) + \frac{V}{r} \frac{\partial}{\partial r} \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] - \frac{\partial}{\partial u} \left(P - \frac{Q^2}{8\pi r^4} \right) = 0 \end{aligned} \quad (3.4)$$

dos de estas ecuaciones son independientes y se puede obtener la ecuación de conservación de Tolman Oppenheimer Volkoff (TOV) u otra ecuación independiente ó tomar las dos ecuaciones (3,1) y (3,4) que permiten obtener relaciones de las derivadas parciales $\frac{\partial \varepsilon}{\partial r}$ y $\frac{\partial \omega}{\partial r}$ o sus equivalentes $\dot{\varepsilon}$ y $\dot{\omega}$. Estas relaciones dan cuenta de la evolución temporal de tanto de la radiación como de la contracción del material. Ese estudio quedará pendiente en un próximo trabajo.

Las Ecuaciones (3.1-3.4) aplicadas a las ecuaciones de Estado de Schwarzschild, Tolman V y TolmanVI se resolverán numericamente en el capítulo 5.

Capítulo 4

Fundamentos Computacionales

En éste capítulo, trataremos de explicar algunas características del programa (**varfis**) utilizado para obtener algunas soluciones autosimilares para el colapso relativista, utilizando computación numérica y gráfica. Para lograr este objetivo, nos hemos enfocado en separar dos ramas o vertiente al desarrollar ésta herramienta computacional. En algunas implementaciones de la arquitectura de software, puede haber muchas capas entre el hardware y el usuario final. La parte frontal es una abstracción que simplifica el componente subyacente al proporcionar una interfaz fácil de usar (Front-End), mientras que la parte posterior generalmente se encarga del almacenamiento de datos y la lógica (Back-End). Es por ello que se diseñado dos capas principales para el desarrollo de ésta herramienta:

1. Implementación de los métodos numéricos utilizados realizada totalmente en lenguaje C (Back End) y
2. La interfaz gráfica del usuario (GUI) o Front-End, realizada en lenguaje C++.

En el desarrollo de software, el Front-End se refiere a la capa de presentación con la que interactúan los usuarios, mientras que el Back-End implica la gestión y el procesamiento de datos en segundo plano. En el modelo cliente-servidor, el cliente suele considerarse el Front-End, que se encarga de las tareas de cara al usuario, y el servidor es el Back-End, que gestiona los datos y la lógica. (Figura 4.1)

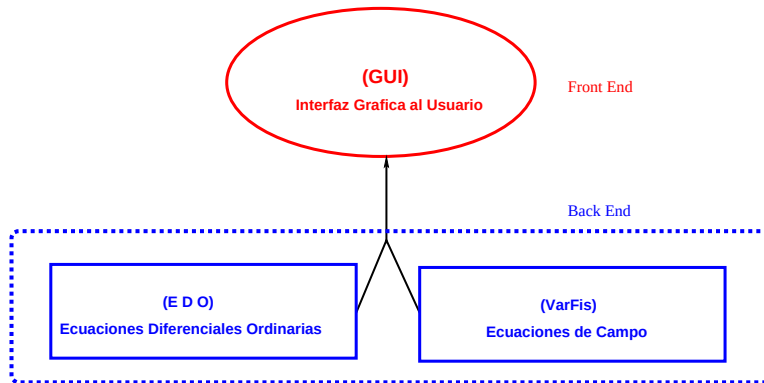


Figura 4.1: Interfaz Grafica al Usuario (GUI) y Cálculo numérico

Para desarrollar ambas vertientes del desarrollo del software, se ha seleccionado como ambiente o Entorno de Desarrollo Integrado (IDE -), Qt Creator. Existen otros ambientes de desarrollo (SDK), tales como Visual Studio (Microsoft), Eclipse (Java), KDevelop (Unix, Windows), NetBeans (Java, C++), Android Studio (Google, Android), IntelliJ IDEA (Java, Kotlin, Groovy), PyCharm (Python), GitHub (Microsoft, Open Source), Xcode (Apple), Atom (Similar Qt Creator), Code::Blocks (IDE multiplataforma para desarrollo en C y C++), Geany (Programación de Múltiples Lenguajes) etc., que se especializan en un lenguaje en particular o en compañías productoras de Software.

Algunas de las características principales tanto del software de desarrollo, (Qt Creator), tanto como de la Implementación de los métodos numéricos (Back End) y la Interfaz Gráfica al Usuario (Front End), se detallan en las secciones siguientes.

4.1. Qt Creator

Qt Creator es un entorno de desarrollo integrado (IDE) multiplataforma para desarrollar y depurar aplicaciones C++ utilizando el SDK de Qt GUI Application Development Framework, y está incluido en el SDK. El software tiene una licencia LGPL gratuita y se ejecuta en Windows, OS X, Linux y versiones de BSD. Viene con un editor de código, un depurador visual, un diseñador de formularios y una herramienta de diseño de GUI. El editor admite funciones como autocompletado y resaltado de sintaxis. Se puede obtener el software para las plataformas Windows, macOS y Linux.

El diseño del programa es consecuencia de separar los procedimientos y funciones de la Simulación Numérica y de las estructura de datos necesarias para la toma de datos del usuario y presentar los resultados de forma gráfica con ayuda del GNUPLOT. En la versión actual, los datos se pueden suministrar en la interfaz gráfica, sin embargo, se espera que en una próxima versión se pueda realizar esa toma de datos de nuevo a partir de la línea de comandos. Se utilizan varias estructuras de datos en su código para representar tanto la integración del sistema de Ecuaciones diferenciales Ordinarias, que es la base de la simulación numérica y las necesarias tanto para solicitar los datos como las ventanas que presentan la salida de los cálculos realizados. En las siguientes secciones se explicarán algunas de estas estructuras de datos.

4.2. Algunas características del programa

Un programa de computadora evoluciona con la madurez del programador¹ y del objetivo que apunta. La versión actual del programa *varfis* — por *V*ariables *F*ísicas —, consta de dos partes bien diferenciadas. La primera es resolver las ecuaciones de superficie, (EDS) que son un conjunto de ecuaciones diferenciales ordinarias que establecen la temporalidad de la solución obtenida y adicionalmente el cálculo de las variables físicas a partir de las Ecuaciones de Campo de Einstein (*Back End*). La segunda es el desarrollo de una interfaz gráfica al usuario (*Front End*).

La primera parte, se implementa en Lenguaje C y permite realizar todos los cálculos numéricos necesarios para alcanzar dicho objetivo. Las primeras versiones del programa utilizaban la línea de

¹O del equipo que lo desarrolla

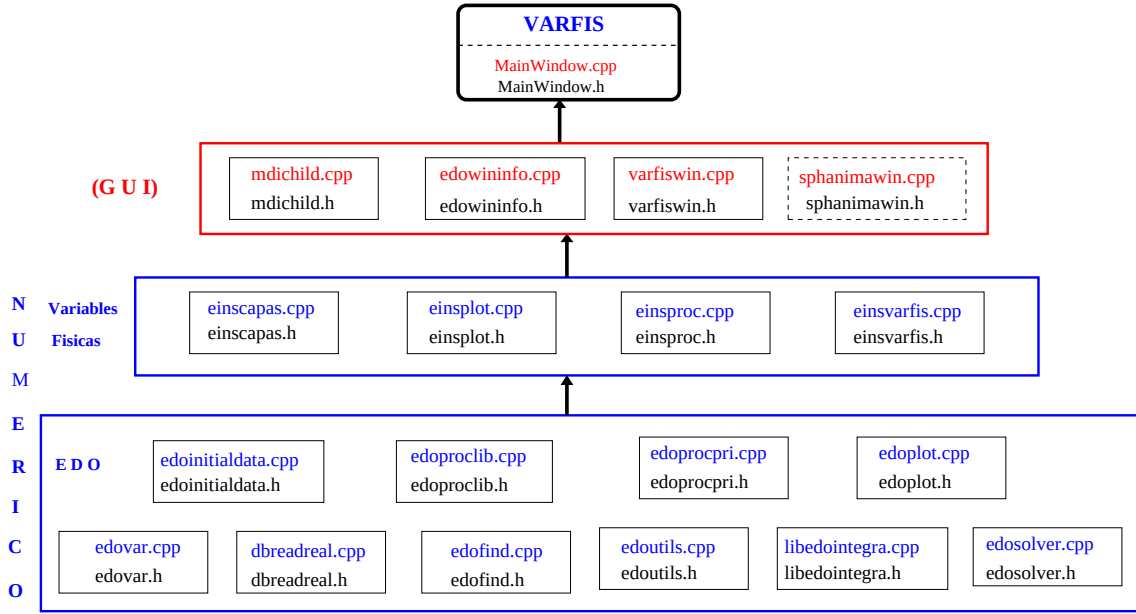


Figura 4.2: Archivos y Dependencias del programa VARFIS

comandos para ingresar todos los datos — Datos necesarios para resolver el sistema de ecuaciones diferenciales, detalles técnicos para el modulo de integración, el sistema de ecuaciones a resolver, los tipos diferentes de archivos para calcular las variables físicas, etc. — Todos estos detalles se ingresaban desde la línea de comandos. La información que se le entregaba al usuario era en parte establecida en un conjunto de archivos de datos y archivos de comandos para el GNU PLOT, que es el encargado de mostrar las gráficas x vs $F(x)$. Todos estos procesos de entrada y salida de datos, se hacían desde la consola y se utilizaba la biblioteca estandar de de entrada y salida de datos — `stdio.h` — (library). Sin embargo, después de la versión 11 de la implementación del compilador para C++, ésta librería fue descontinuada, junto con otras; principalmente por posibles fallos de seguridad Deprecated, Github. Se puede seguir usando, por ejemplo la librería `cstdio.h`, pero se decidió otra opción. En lugar de usar la consola como archivo estandar de salida, se seleccionó un enfoque diferente para comunicación con el usuario. Tomando como ejemplo los demonios del sistema operativo Linux, que no tienen acceso a la consola, se decidió como alternativa, utilizar el sistema de archivo y la facilidad de las librería `<fstream>` para las operaciones de entrada y salida. Este enfoque permite comunicarse con una interfaz gráfica al usuario de una manera mas fluida y realizar todas las operaciones de cálculo de variables físicas y cálculo numérico en lenguaje C y dedicar un esfuerzo en C++ para desarrollar una interfaz que pueda ser modular. Esto significa que se puede separar tanto los cálculos (numéricos, etc.) como la interfaz gráfica, en módulos separados y en principio permite una manera más cómoda de extender características nueva tanto para la simulación numérica (nuevos procedimientos de integración hamming, predictor-corrector, etc.) o ligar en un futuro, la interfaz gráfica al cálculo de las variables físicas a la animación. Para ésta versión del programa, se utilizará básicamente el programa GNU PLOT como Interfaz Gráfica.

1. los archivos de prototipos de las funciones y procedimientos (*.h);

2. los archivos que desarrollan el código en C (*.c) o (*.cpp) de los prototipos establecidos en los archivos de encabezado y después se enlazan en tiempo de compilación,
3. los archivos de objetos compartidos (*— shared objects —*, *.so). En nuestro programa, los archivos de biblioteca (library), van a corresponder a
 - a) los sistemas de ecuaciones,
 - b) a los cálculos de la métrica y
 - c) al calculo de las variables físicas.

Se compilan aparte y se enlazan en tiempo de ejecución. Esto permite tanto tener diversas ecuaciones de superficie, de acuerdo a las multiples ecuaciones de estado que se puedan estudiar, así como las métricas correspondientes y las variables físicas a calcular. En principio, permitiría contar con una única aplicación (“codigo”) que contenga todos los modelos calculados como una biblioteca de casos particulares y que permita el desarrollo de nuevos modelos (analíticos) con un mínimo de cálculo (algebraico y numérico). Adicionalmente, para potenciar la utilidad de esta herramienta es necesario que permita el desarrollo de los modelos más reales. En este sentido es necesario incorporar ecuaciones de estados correspondientes a materia nuclear ultradensa y ecuaciones de transporte de radiación en la cual las opacidades y emisividades jueguen un papel importante ([101, 102]).

Debido a la extensión del programa – 18 archivos de prototipos (.h) y 19 archivos de codigo *.cpp – vamos a elegir algunos de ellos. En la sección siguiente, vamos a comentar algunas de las estructuras de datos y su correspondiente código fuente.

4.3. Métodos numéricos

La base de la simulación numérica es un conjunto de archivos prototipos en lenguaje C (.h o headers) que permite la definición de las diferentes estructuras de datos para la integración de un sistema de ecuaciones diferenciales ordinarias (ODE). Los métodos numéricos que se exponen en este capítulo se estudian en diferentes secciones: En la sección 4.3.1 se establece algunos de los métodos numéricos para la integración del sistema de ecuaciones de la superficie (EDS). La sección 4.3.2.5 se establecen las algunas estructuras de datos necesarias para realizar el cálculo de las variables físicas, tomando en consideración las ecuaciones de campo de Einstein, para el sistema de coordenadas de Bondi.

4.3.1. Ecuaciones diferenciales ordinarias

Tal como se señaló en el capítulo anterior, el problema en estudio se reduce a la integración numérica de un conjunto de n ecuaciones diferenciales de primer orden para el conjunto de ecuaciones y_i , $i = 1, 2, \dots, n$ que tienen la forma en general:

$$\frac{dy_i}{dx} = f_i(x, y_1, \dots, y_n) \quad i = 1, \dots, n \quad (4.1)$$

donde las funciones f_i son conocidas. Tan importante como establecer el sistema de ecuaciones diferenciales también es establecer las condiciones de frontera. Las condiciones de contorno son condiciones algebraicas sobre los valores de las funciones en (4.1). En general, estas condiciones de contorno se pueden especificar o clasificar en dos grandes tipos:

- *Problema de valor inicial*: Los valores de las funciones y_i se establecen en algún valor inicial x_0 y se desea determinar el valor de las y_i en algún punto final x_f
- *Problema de frontera*: En este caso las condiciones de frontera se especifican en más de un punto. Típicamente algunas condiciones se establecen en el valor x_0 y el resto en x_f .

Aquí haremos mención solamente al problema con valor inicial. Existen muchas técnicas para realizar la integración del sistema de Ecuaciones Diferenciales Ordinarias, sin embargo, aquí utilizaremos dos procedimientos numéricos para resolver el problema de valor inicial para la integración de las ecuaciones de la superficie (EDS): método de *Runge-Kutta* y el método de *Bulirsch-Stoer* ó *extrapolación de Richardson*

4.3.1.1. Método de Runge-Kutta

Este método consiste en obtener la solución en varios pasos pequeños utilizando el método de Euler para luego combinar la información obtenida en una expansión en serie de Taylor para mejorar la precisión. La expresión para el método de Euler:

$$y_{n+1} = y_n + h \cdot f(x_n, y_n) \quad (4.2)$$

establece una solución desde x_n hasta $x_{n+1} \equiv x_n + h$. Existen muchas razones para no utilizar directamente este método de integración, y la más importante es que no es muy preciso al compararlos con otros métodos y no es muy estable[103]. A pesar de estos inconvenientes, tomemos en consideración la expresión (4.2) para tomar una estimación ó prueba en la mitad del intervalo, y usemos tanto el valor en x como en y , en el *punto medio*, para calcular una aproximación al valor real en todo el intervalo o paso. En ecuaciones esto se apreciaría como:

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1) \\ y_{n+1} &= y_n + k_2 + O(h^3) \end{aligned} \quad (4.3)$$

Con la expresión para la aproximación anterior, se dice que este método es un método de segundo orden². De hecho la ecuación (4.3) se denomina *Runge-Kutta de segundo orden* o *método del punto medio*.

Podemos utilizar el mismo razonamiento para obtener una mejor aproximación en la solución y acotar el término del error, al elegir la correcta combinación de los términos y coeficientes. Esta es la idea central método de Runge-Kutta. De todos los métodos de este estilo, utilizados para la integración de sistemas de ecuaciones diferenciales, uno de los más populares es la fórmula de

²Un método se denomina de n-esimo orden si el término del error es $O(h^{n+1})$.

Runge Kutta de cuarto orden:

$$\begin{aligned}
 k_1 &= hf(x_n, y_n) \\
 k_2 &= hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1) \\
 k_3 &= hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2) \\
 k_4 &= hf(x_n + h, y_n + k_3) \\
 y_{n+1} &= y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + O(h^5)
 \end{aligned} \tag{4.4}$$

La fórmula de Runge-Kutta de cuarto orden necesita de cuatro evaluaciones de la expresión $f(x, y)$ por cada paso de integración. Si la comparamos con la expresión del método del punto medio, se necesita dos evaluaciones más por paso, pero no podemos concluir que siempre es más preciso. No necesariamente mayor orden significa mayor precisión.

En la implementación de esta rutina en el programa se utilizó una versión en Lenguaje C del método de quinto orden de Cash-Karp de Runge-Kutta. La expresión para este método de quinto orden y la estimación de error se expresa en la ecuación

$$\begin{aligned}
 k_1 &= hf(x_n, y_n) \\
 k_2 &= hf(x_n + a_2h, y_n + b_{21}k_1) \\
 \dots &\quad \dots \\
 k_6 &= hf(x_n + a_6h, y_n + b_{61}k_1 + \dots + b_{65}k_5) \\
 y_{n+1} &= y_n + c_1k_1 + c_2k_2 + c_3k_3 + c_4k_4 + c_5k_5 + O(h^6)
 \end{aligned} \tag{4.5}$$

donde los coeficientes a_k y c_k se pueden encontrar en la referencia ([103]) pág. 716 y siguientes.

4.3.1.2. Método de Bulirsch - Stoer

En este método se usa la idea de extrapolar un resultado obtenido hacia el valor que se obtendría si el paso hubiese sido menor. En particular la extrapolación hacia un paso nulo sería el objetivo ideal. La implementación de esta idea fué desarrollada en lenguaje Algol por primera vez por Bulirsch y Stoer [104] y por esto muchos de los métodos de extrapolación reciben este nombre.

Existen tres razones importantes para el desarrollo e implementación del método de Bulirsch-Stoer para la integración de un sistema de ecuaciones diferenciales. La primera es el método de extrapolación de Richardson, el cual retomaremos en el método de integración de Romberg. La idea es considerar la respuesta final de un cálculo numérico como si fuese una función analítica de un parámetro ajustable como h . A esta función analítica se le evalúa para varios valores del parámetro ajustable h , que no necesariamente tiene que ser un valor pequeño, para no poner en riesgo el cálculo por overflow o cualquier otra forma de error en la implementación. Cuando se conozca lo suficiente de la función, podemos *encontrar* una expresión para ella y evaluarla para el punto $h = 0$.

La segunda idea de este proceso es como *encontrar* la función analítica que fije los valores encontrados. Bulirsch y Stoer en su implementación original utilizaron la fortaleza de la interpolación numérica por medio de funciones racionales. Esta aproximación con funciones racionales, tiene la ventaja de ser muy estable aún si varios términos de potencias de h tienen magnitudes similares. En otras palabras, h puede ser tan grande como se desee y por esto la noción del orden de método no tiene una gran trascendencia.

La tercera idea es que se utilice como función analítica una función estrictamente par, permitiendo que la aproximación se mantenga en términos de potencias de h^2 en lugar de h . Al concatenar todas estas ideas se obtiene el Método de *Bulirsch-Stoer* [104]. En la referencia ([103, 105]) se encuentra una explicación con detalles técnicos muy buena donde se incluye una versión en Fortran de la subrutina de integración, y en ([104]) se encuentra los fundamentos matemáticos y el artículo original de Bulirsch y Stoer donde se propone el método de extrapolación racional, así como una versión en Algol del proceso de integración de las ecuaciones diferenciales ordinarias.

Por supuesto que existen muchos más métodos de Integración de Ecuaciones Diferenciales Ordinarias, pero en esta versión del programa usaremos principalmente éstos dos. Sin embargo, se espera extender los métodos de integración en versiones posteriores ya que el enlace del tipo de método a utilizar se hace en tiempo de corrida al utilizar librerías (ó bibliotecas) de objeto compartido (lib*.so - *shared objects* -) y no en tiempo de compilación o librerías estáticas. Se espera tener en la próxima versión del programa, poder elegir del menú para la selección algún otro método de integración, como el método de Hamming o una versión del Predictor-Corrector.

4.3.2. Implementación de métodos numéricos

Como se puede observar en la figura 4.2 el conjunto de archivos que se encuentran en la base piramidal, es la base de los métodos numéricos de integración. Sin embargo, se debe notar que esta figura no incluye los archivos de librería o biblioteca (*lib*.so*) que definen el sistema de ecuaciones de superficie (SES), la integración de los elementos de la métrica, o el cálculo de las variables de superficie. Se muestran los archivos base del programa. A continuación se comentan algunas características de éstos archivos que nos establecen las bases numéricas y los archivos necesarios para determinar las variables físicas, vía las ecuaciones de campo de Einstein

4.3.2.1. edovar

Para la implementación de las rutinas de integración del sistema de ecuaciones diferenciales se utiliza varias estructuras de datos definida en el archivo **edovar.h**:

TValue: Para mantener los datos iniciales y los detalles técnicos de la integración,

TSysOfEq: El sistema de Ecuaciones a integrar,

TSysIntegracion : Tipo para los métodos de integración y los datos iniciales.

En el listado (4.1) se puede apreciar todo el detalle de la implementación.

Listado 4.1: edovar.h

```
#ifndef EDOVAR_H
#define EDOVAR_H

/*Numero maximo de ecuaciones*/

#define maxnueq          10

/* Apuntador y tipo al arreglo de numeros reales */
typedef double TVector[maxnueq + 1];
/* Apuntador y tipo de Sistema de ecuaciones*/
```

```

typedef void (*TSysOfEq)(double x, double *y, double *dy);

/* Apuntador y tipo para prototipo de los mtodos de integracin*/
typedef int (*TSysIntegracion)
(TSysOfEq Equations, int n, double *x, double *paso, double *hminimo,
 double *eps, double *y, double *ya, int Error);

/* Apuntador y tipo para los valores iniciales para la integracion del */
/* sistema de ecuaciones diferenciales (EDO)*/

typedef struct TValue {
    /* Nombre del archivo de datos*/
    char *FileOfData;
    /* Nmero total de ecuaciones en el sistema*/
    int Ecuaciones;
    /* Valores inicial, final, paso inicial, espaciado */
    /* entre puntos consecutivos y tolerancia exigida al */
    /* driver de integracion */
    double Inicial, Final, Paso, Espaciado, Tolerancia;
    /* Tipo de Error : 0→ Error relativo,*/
    /*                      1→ Error promedio,*/
    /*                      2→ Error absoluto*/
    /* Salvar los datos de la integracin: 0→ Salvar 1→ No salvar*/
    /* Derivada: 0→ No se guardan los datos de la derivada */
    /*                      1→ Se guardan los datos de la derivada */
    int TipoError, Salvar, Derivada;
    /* Apuntador a los valores iniciales del sistema de ecuaciones*/
    double *PAtYinicial;
} TValue;
/*zot → tolerancia al zero:*/
/*    el compilador considera que dos numeros reales que */
/*    difieren en esta cantidad son considerados iguales */

/*zotup → solo es un numero muy grande */

/* procedimiento para buscar los valores de tolerancia zero */

void edovar_init(double *zot, double *zotup);
//
#endif // EDOVAR_H

```

4.3.2.2. edoprocpri

Estos archivos `edoprocpri.h` y `edoprocpri.c` tienen los *procedimientos principales* para realizar la integración del sistema de ecuaciones diferenciales ordinarias que dan cuenta de la evolución temporal. Permiten enlazar (linker) el sistema de ecuaciones ligado al modelo a estudiar, en tiempo

de ejecución ya que utilizan bibliotecas de enlace dinámico (`shared object .so`) donde se encuentra el sistema de ecuaciones y la selección de los método de integración.

Listado 4.2: edoprocpri.h

```
#ifndef EDOPROCPRI_H
#define EDOPROCPRI_H
#include <fstream>
char *createnamelogfile(char *namefile);
double desub(std::fstream &f, char **argvlst);
#endif // EDOPROCPRI_H
```

4.3.2.3. libedointegra

Este es uno de los archivos principales que controla el proceso de integración de un sistema de ecuaciones diferenciales. Permite el ajuste automático del paso de integración. Carga tanto el método de integración `TSysIntegracion`, como el sistema de ecuaciones con el tipo `TSysOfEq` que se enlaza en tiempo de ejecución.

Listado 4.3: libedointegra.h

```
#ifndef LIBEDOINTEGRA_H
#define LIBEDOINTEGRA_H
#include <fstream>
#include "edovar.h"
int EdoInt(std::fstream &f, TSysIntegracion Metodo, TSysOfEq Equations,
          int n, double *x, double *xfinal, double *h, double *sp,
          double *eps, double *y, int error, int deriv, double tiny);
#endif // LIBEDOINTEGRA_H
```

4.3.2.4. edoplot

Una vez realizada con éxito la integración numérica se construyen los archivos de comandos necesarios (edoplot) para utilizar el GNUPLOT y realizar los gráficos, de cada variable de superficie obtenida.

Listado 4.4: edoplot.h

```
#ifndef EDOPLOT_H
#define EDOPLOT_H
int namesvarplot(int n, char *filedata, char **namesofvar);
char *setplotvars(int n, char *filedata, char **outdataname);
#endif // EDOPLOT_H
```

4.3.2.5. einsvar

Este otro modulo numérico tiene la finalidad de ser prototipo para las estructuras de datos que permiten establecer el cálculo de las variables físicas a partir de las ecuaciones de campo. Las mas importantes en éste modulo:

TMetrica: Establece la métrica del sistema de coordenadas elegido (Bondi en nuestro caso)

TSetVarfis: Calcula los valores de las variables físicas en cada sección o capa de la distribución de materia.

Estos dos tipos se enlazan en tiempo de ejecución

Listado 4.5: einsvar.h

```
/* Header for module einstvar, generated by p2c 1.21alpha-07.Dec.93 */
#ifndef EINSVAR_H
#define EINSVAR_H
//
/* Unidad que establece los tipos basicos para otras unidades */
/* que calculan las variables fisicas (varfis) e integracion */
/* de funciones de una variable (para calcular) ï¿œ (beta) y m */
/* definidas en la metrica en coordenadas radiativas tal como */
/* se establece por ejemplo en Medina et al (1986) Evolution */
/* of radiating charged spheres in General Relativity, Journal */
/* Canadian of Physics o los paper de Bondi (1960) o Herrera et */
/* al (1980)*/
//
#include "edovar.h"
#define MaxVarfis 15
#define CapasMaximo 25
/* Apunta al tipo que define el arreglo para las variables */
/* fisicas */
typedef double TVarfis[MaxVarfis + 1];
/* Apunta al tipo que define el arreglo donde se guardan las */
/* capas donde se desea calcular las variable fisicas */
typedef double TCapasMax[CapasMaximo + 1];
/* Apunta al tipo que define la metrica del modelo a simular */
typedef long (*TMetrica)
(TSysOfEq Equations, double capa, double x, double *y, double *TVarfis);
/* Apunta al tipo que calcula las variables fisicas modelo a simular */
typedef long (*TSetVarfis)
(double capa, double *y, double *MyBeta, double *Salida);
/* Apunta al tipo que almacena los nombres de las variables */
/* fisicas que se calculan */
typedef char *TNamesVarfis[MaxVarfis + 1];
//
/* Apuntador de los Maximo en el radio, Datos al modelo, MÃ©trica
y al sistema de ecuaciones (Ecuaciones de Superficie —SSE— ) */

typedef struct TInfoVarfis
{
    void *lpData;
    void *lpMetric;
    void *lpSystem;
```

```

    void    *lpVarfis;
} TInfoVarfis;

double setvarfiscalc(std::fstream &f, char **argvlst);
#endif /*EINSVAR_H*/
/* End. */

```

4.4. Interfaz Gráfica al Usuario (GUI)

4.4.1. Implementación de la Interfaz Gráfica

En la implementación de la simulación visual, se utiliza una cantidad relativamente pequeña de recursos de hardware y de software. No es necesaria una estación de trabajo para la simulación ya que la interfaz gráfica del usuario suministrada en el sistema operativo es suficiente, así como las funciones API³ que tiene el software para utilizarla. Ya se expuso todo lo que en ésta versión del programa le concierne a los métodos numéricos de integración y de resolución de ecuaciones diferenciales ordinarias que se utilizan en el programa.

La animación es una alternativa a la representación gráfica tradicional es x vs $F(x)$, como en el caso que se ha utilizado aquí. Mostrar una figura que represente el modelo físico bajo estudio y simular la evolución de éste mediante técnicas de animación. La animación por color se adapta perfectamente para la representación de la evolución de una variable en configuraciones unidimensionales. Logra el efecto de movimiento de los objetos a través de cambios cromáticos al modificar la definición de los colores secuencialmente en la tabla de colores. Esto es muy similar a los anuncios luminosos que encendiendo y apagando bombillos cambian la posición de la figura, y repitiendo esos cambios en forma cíclica, ésta cobra dinamismo. Al representar la evolución de configuraciones esféricas, este esquema refleja, adicionalmente, los posibles rebotes de la superficie de la esfera al mostrar la disminución o expansión de su radio. Es de hacer notar que el desarrollo de éstas técnicas de animación requieren un trabajo extra, que muy posiblemente se muestre en una versión posterior del programa.

En lo que compete al resto del capítulo vamos a resaltar algunos detalles de los archivos utilizados para la creación del programa. En casi todos los programas en Lenguaje C, y este no es la excepción, se pueden tener tres tipos de archivos bien diferenciados:

4.4.1.1. edowininfo

Para esta sección vamos a establecer algunas de las estructuras de datos básicas para la presentación y selección de los datos en una ventana. El marco principal para es un estructura de datos que hace de sosten o base para las demás ventanas. Esta estructura es una MDI (Interface de Multiples Documentos – Multiple Document Interface, por sus siglas en inglés) y el ella creamos las ventanas hijas con la estructura de datos que se define en `edowininfo`. Los detalles de ésta estructura se aprecia en 4.6.

³Por el acrónimo del inglés Application Programming Interface **API**

Listado 4.6: edowininfo.h

```

#ifndef EDOWININFO_H
#define EDOWININFO_H
#include <QtWidgets>
#include <QMenu>
#include <QMessageBox>
#include <QMouseEvent>
#include <QPaintEvent>
#include <QPainter>
#include <QTextDocument>
class QPoint;

class edowininfo : public QWidget
{
    Q_OBJECT
public:
    edowininfo(QString file = nullptr, QString tokenini = nullptr,
               QWidget *parent = nullptr);
    ~edowininfo();
    void mousePressEvent(QMouseEvent *event);
    void paintEvent(QPaintEvent *event);
    void setlogfileName();
    void setupdataInfo(char *file, char *tokenini);
    int setdataainfolst(const QStringList &infotoupdate);
    double setnameproceqn(char *valuenam, char *tokenini);
    double difeqnsolve();

signals:
    void findNext(const QString &);
private slots:
    void setvalue(const QString &value);
    void menuSelection(QAction* action);

private:
    QPoint lastPoint;
    int totproc = 0;
    char *datanamesproc = NULL;
    char **eleccion = NULL;
    char **dataainfolst = NULL;
    double elapsedtime = 0;
    QString eqn_sel = nullptr;
    QString logfilestring;
};

void edoWinName(const QStringList &lstupdate, QString &winName);
#endif // EDOWININFO_H

```

El archivo *edowininfo.cpp* implementa los métodos, funciones y procedimientos para lograr la selección de los diversos parámetros de los datos de usuario. Este archivo es algo extenso y no lo presentaremos en un listado. Sin embargo, podemos destacar algunas de las diferentes características, como la selección del menu flotante como en `void mousePressEvent(QMouseEvent *event)`, que responde al evento de pulsación del boton derecho en el ratón, para escoger, ya sea:

1. el método de integración a utilizar,
2. la ecuación de superficie a integrar ó
3. el procedimiento para el cálculo de las variables físicas

Esto se logra al cambiar los diferentes archivos de objetos compartidos (lib*.so) compilados y enlazados en tiempo de ejecución. Una de las cosas necesarias es que los procedimientos cargan el objeto y se debe colocar en el mismo directorio el archivo fuente que define el objeto compartido. Esto es necesario ya que el programa lee el archivo fuente para leer los nombres de ya sea el método de integración, la ecuación de superficie a integrar ó el cálculo de las variables físicas. Esto es responsabilidad primordialmente de la función `setCollectOptions` que se encuentra en el archivo *edoproclib.cpp*, que es un archivo fuente en lenguaje C.

4.4.1.2. varfiswin

Esta clase define los parámetros y características de la ventana que muestra los datos de salida de los cálculos realizados y la manera de establecer las peticiones para los nombres de las ecuaciones a integrar como los métodos utilizados para calcular la métrica y poder calcular las variables físicas a determinar en cada capa de la distribución de material. Muestra al usuario, al pulsar el boton derecho del ratón, un menú para elegir todas las ecuaciones posibles a integrar, y todas las opciones posibles.

Listado 4.7: varfiswin.h

```
#ifndef VARFISWIN_H
#define VARFISWIN_H
#include <QtWidgets>
#include <QMenu>
#include <QMessageBox>
#include <QMouseEvent>
#include <QPaintEvent>
#include <QPainter>
#include <QTextDocument>
//
//#include
class QPoint;

class varfiswin : public QWidget
{
    Q_OBJECT
public:
    varfiswin(QString file = nullptr, QString tokenini = nullptr, QWidget *parent = nullptr);
    ~varfiswin();
    void mousePressEvent(QMouseEvent *event);
```

```

    void paintEvent(QPaintEvent *event);
    void setupdataInfo(char *file , char *tokenini);
    void setlogfileName();
    void nameWinvarfis(const QStringList &lstupdate , QString &winName);
    int setuplstvalue(const QStringList &varfislstupdate);
    double updatevalue(char *valuenam, char *tokenini);
    double varfisCalc();

signals:
    void findNext(const QString &);
private slots:
    void setvalue(const QString &value);
    void menuSelection(QAction* action);

private:
    QPoint lastPoint;
    int totproc          = 0;
    char *datanamesproc  = NULL;
    char **eleccion      = NULL;
    char **vrfiscmdline  = NULL;
    double elapsedtime   = 0;
    QString eqn_sel      = nullptr;
    QString logfilestring;
};

void varfisWinName(const QStringList &lstupdate , QString & winName);
#endif // VARFISWIN_H

```

Todas los objetos que se muestran aquí se encuentran en cada una de las unidades del Modulo Visual de la figura 4.2. En ésta versión del programa, aun está por desarrollar el modulo de animación.

En el próximo capítulo se mostraran algunos resultados de las simulaciones realizadas así como las correspondientes gráficas, usando el GNU PLOT, para mostrar el comportamiento de las variables de Superficie, como de las variables físicas. Se espera tener listo el modulo de animación tanto en el ambiente Qt y en X-Windows, para una próxima versión.

Capítulo 5

Resultados y Gráficas

Se presentan los resultados de la integración numérica del algoritmo establecido en los capítulos anteriores, para un conjunto de datos iniciales y diferentes tipo de ecuaciones de estado, ya discutidas. La ventaja de éste algoritmo es la posibilidad de utilizar diferentes tipos de ecuación de estado, puesto que usualmente solo se ha empleado la ecuación de estado ultrarelativista $P = \kappa\rho$. El modelo de Tolman V, aquí utilizado, es el que mas se asemeja a la ecuación de estado ultrarelativista y es el que presenta mejor coincidencia en su comportamiento oscilatorio amortiguado al obtenido por otros autores.

5.1. Modelo tipo Schwarzschild

La figura 5.1, muestra la evolución del radio A . Se puede observar que $\Omega = 1$, representa una condición de equilibrio estático, $\Omega > 1$ representa la expansión; $\Omega < 1$, el colapso. En ambos casos el sistema vuelve al equilibrio muy rápidamente. Para hacer alguna comparación, tomamos los datos iniciales muy cercanos a los elegidos en la referencia [81]. No utilizamos el valor para $\Omega = 1$, ya que en esta aproximación el sistema tiene comportamiento estático. Los valores iniciales que se tomaron de las variables de superficie para ésta simulación se tomaron $A = 5$; $M = 1,0$; $\Omega = 0,833$. La línea a trozos representan los resultados obtenidos en el presente trabajo y la línea continua es la solución en Herrera et al.[81] con los mismos datos iniciales.

Las figuras 5.2 Presión y 5.3 Radiación, representan los perfiles de variables físicas para el modelo de Schwarzschild en las capas $\frac{r}{a} = 0,0, 0,4, 1,0$ función del tiempo. Obtenemos variaciones monótonas en las magnitudes físicas, como consecuencia de la no suposición de un pulso Gaussiano. En particular se observa en la figura 5.3, todas las capas emiten radiación pero decrecen hasta alcanzar un estado de equilibrio para $\Omega = 1$, a diferencia de la figura 7 en Herrera et al [81].

5.2. Modelo tipo Tolman VI

La figura 5.4 muestra la variación temporal del radio de la esfera radiante para diferentes valores de Ω . Con valores iniciales, $A = 6,66667$; $M = 1$. $\Omega = 0,87$ (rebote) y $\Omega = 0,85$ (contracción) y el valor crítico $\Omega_0 = 0,860723$. En las siguientes figuras 5.5,5.6 se se muestran las variaciones de

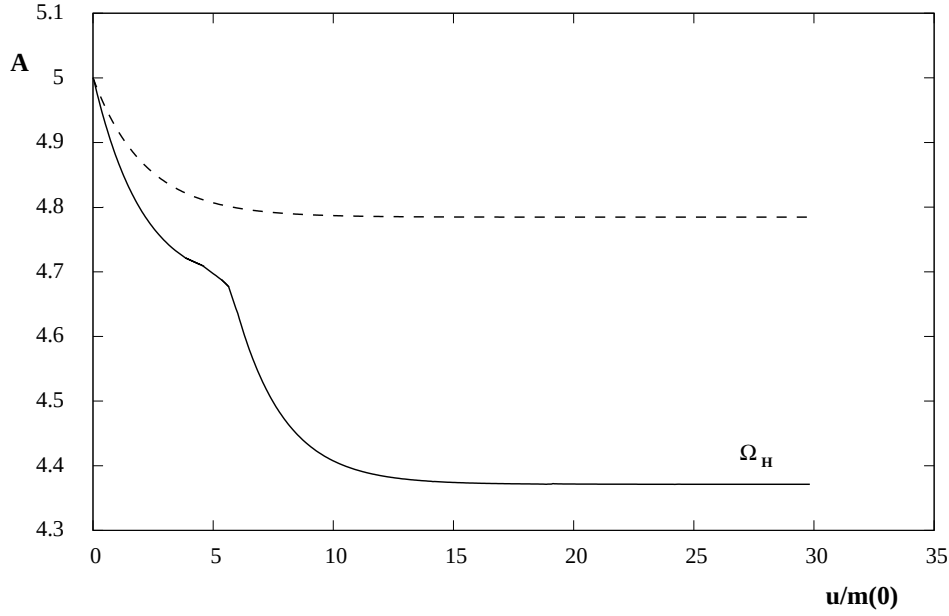


Figura 5.1: Radio A en función de la coordenada temporal normalizada $\frac{u}{m(0)}$ en el modelo tipo Schwarzschild.

presión y radiación de algunas capas interiores con los valores $\frac{r}{a} = 0, 0,4, 0,6, 0,8$ y $1,0$. para las capas en contracción ($\Omega = 0,87$). Observamos en el ejemplo de la figura 5.6 que todas las capas absorbieron energía durante el colapso inicial y luego generaron la emisión como un pulso radiativo, antes de regresar a la configuración de equilibrio.

5.3. Modelo Tolman V

En la figura (5.7) se puede observar, al mantener fijos los valores de $M = 1,0$ y $\Omega = 0,87$, que para valores del radio inicial de la distribución $A = 3,1$, hay un crecimiento exponencial en la evolución temporal y se presentan oscilaciones amortiguadas para los valores de $A = 3,8$ y $A = 8,0$. Cabe señalar que en trabajos previos como Patiño-Rago [84] o Krori [106] donde se utilizó la ecuación de estado de Tolman V como solución estática, no se reportó un comportamiento autosimilar debido a que la integración del sistema de ecuaciones superficiales se limitó a un intervalo corto de la variable temporal. Nótese que para valores pequeños del radio A se produce una expansión continua (explosión). Sin embargo, para esferas menos compactas, las oscilaciones amortiguadas tienden a alcanzar un equilibrio estático. Por este motivo, se realiza la simulación del valor $A = 8,0$ en la figura (5.7) para demostrar que con datos iniciales similares a los utilizados en [84, 106], se obtiene el comportamiento oscilatorio amortiguado. La evolución temporal del radio A está de acuerdo con Choptuik [107] y Gundlach [108, 109]. Este comportamiento oscilatorio amortiguado se extiende a las diferentes variables físicas que se calcularon y por tanto este comportamiento se repite para la presión normalizada (Figura 5.8). A partir de los datos obtenidos en el ajuste, se concluye que los valores del pseudoperíodo entre la variable del radio de la distribución y la

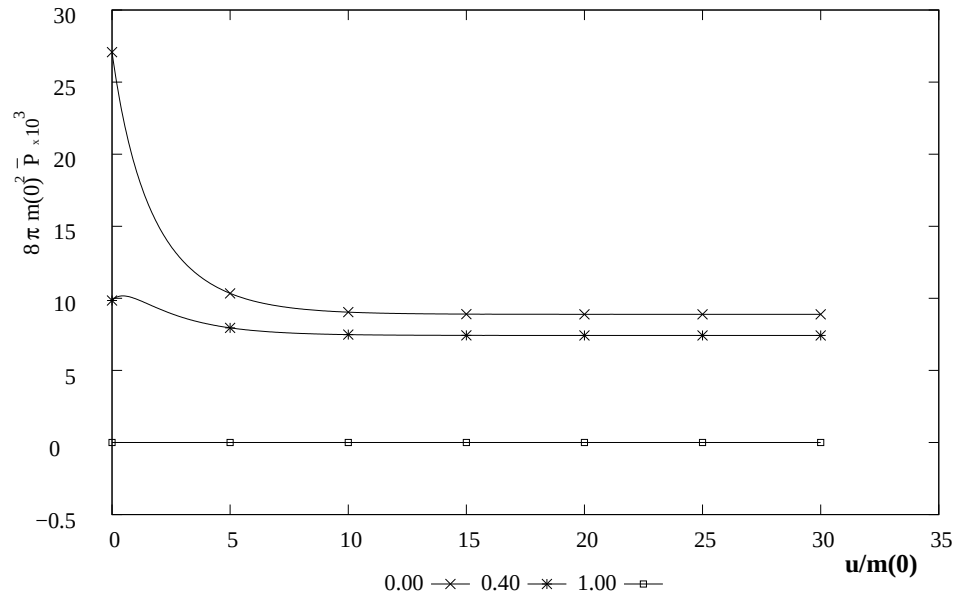


Figura 5.2: Valores de presión normalizados $8\pi m(0)^2 \bar{P}$ en función de la variable temporal $\frac{u}{m(0)}$ en el modelo tipo Schwarzschild.

densidad normalizada son muy similares. Esta ecuación de estado, empleado el método descrito ha sido utilizado por Medina [7] para estudiar las oscilaciones del colapso gravitacional en el marco de la GRT

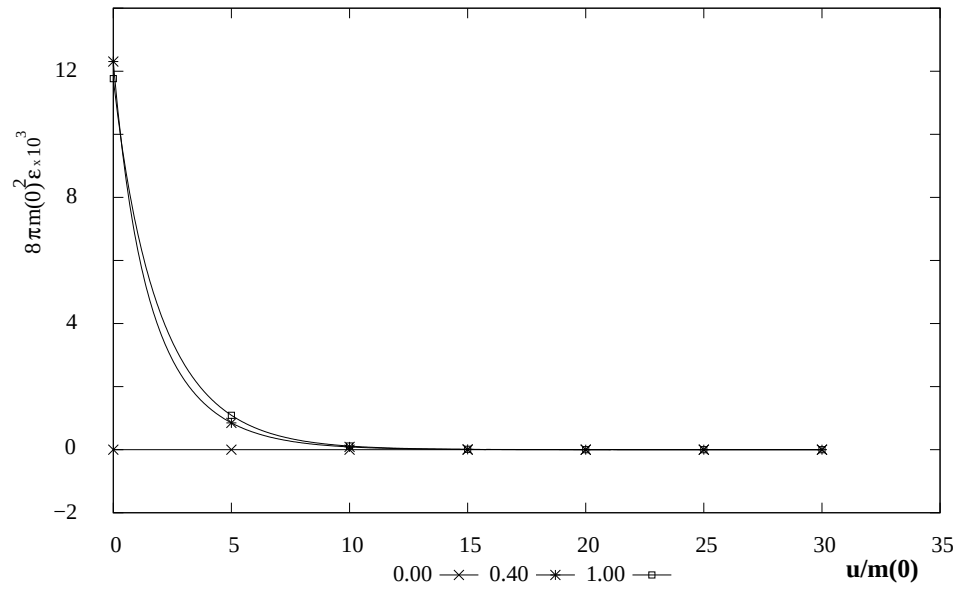


Figura 5.3: Radiación $8\pi m(0)^2 \epsilon$ para el modelo de Schwarzschild.

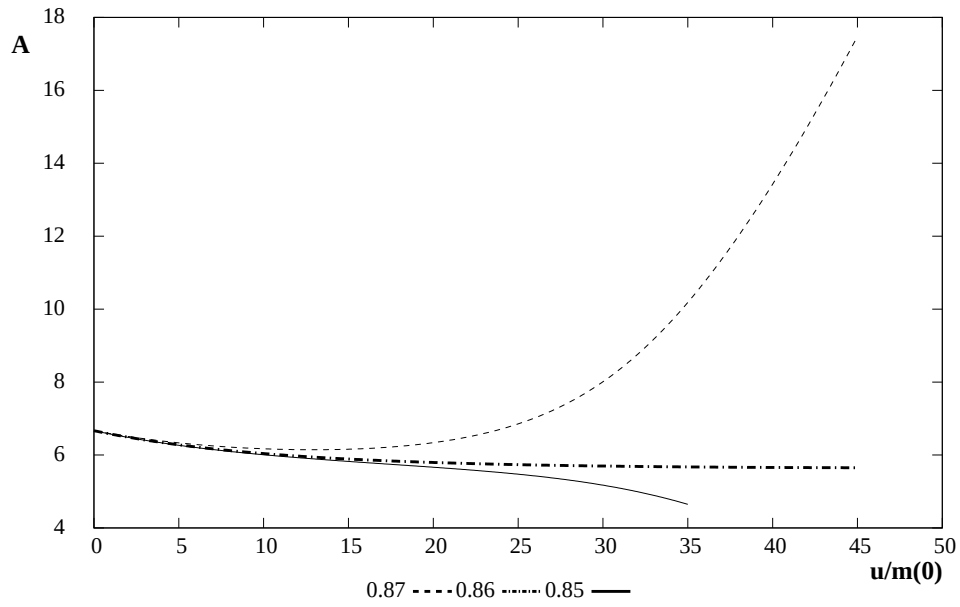


Figura 5.4: Radio A de la distribución según la variable temporal $\frac{u}{m(0)}$ en un modelo tipo Tolman VI

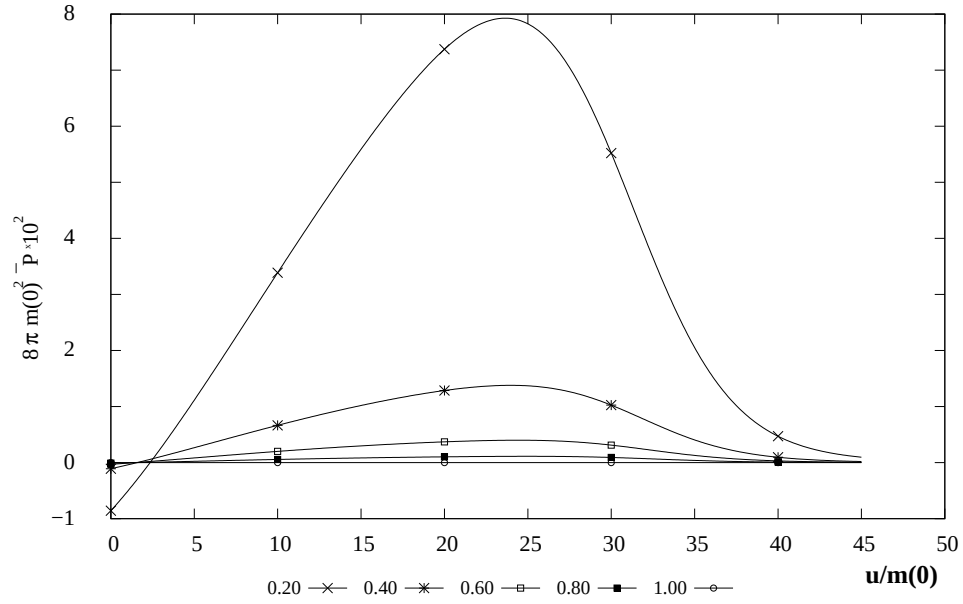


Figura 5.5: Valores de la presión normalizada $8\pi m(0)^2 \bar{P}$ en función de la variable temporal $\frac{u}{m(0)}$ en el modelo tipo Tolman VI.

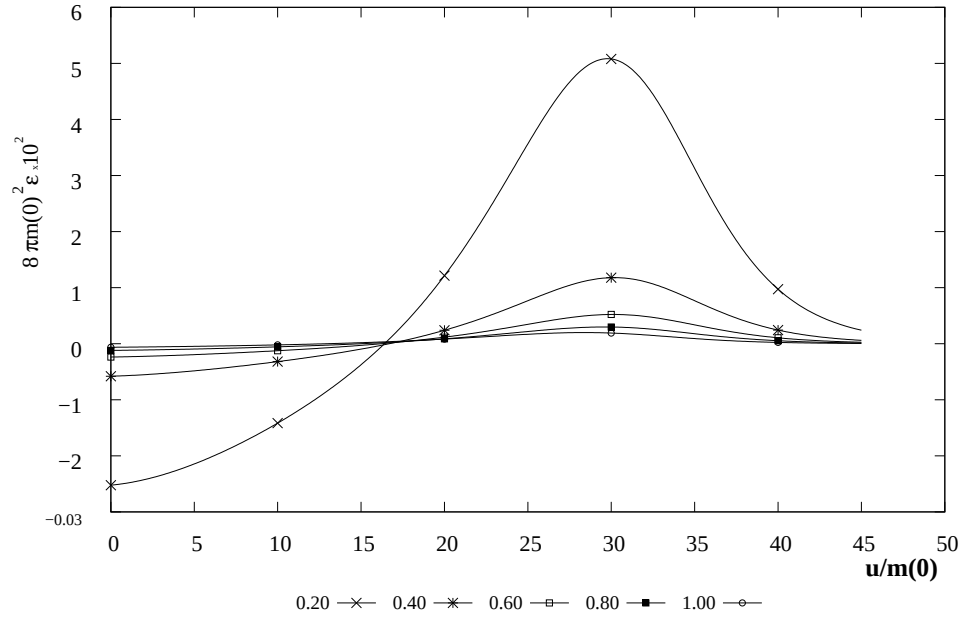


Figura 5.6: Radiación emitida $8\pi m(0)^2 \varepsilon$ en función de la variable temporal $\frac{u}{m(0)}$ en un modelo tipo Tolman VI.

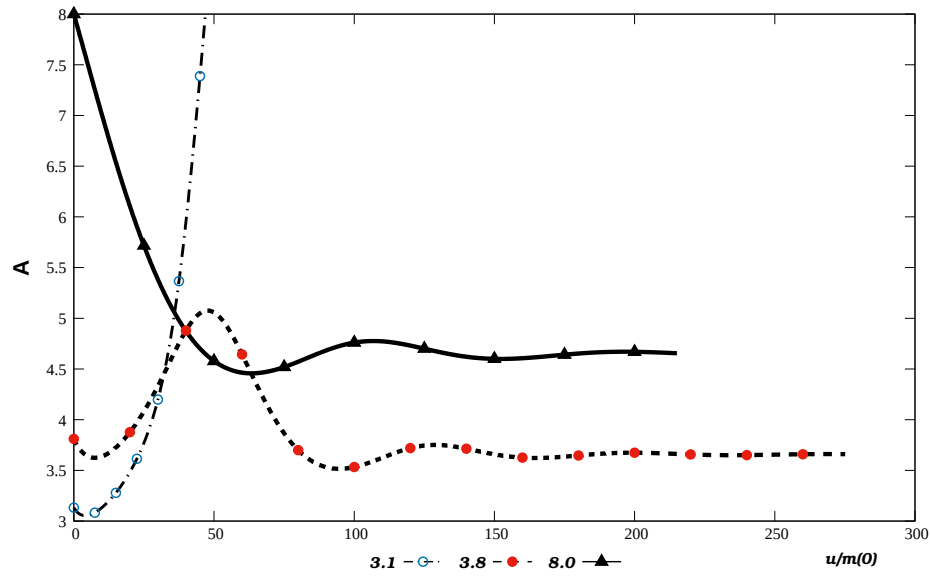


Figura 5.7: Evolución del radio A , en función de la variación temporal normalizada $\frac{u}{m(0)}$ (Tolman V)

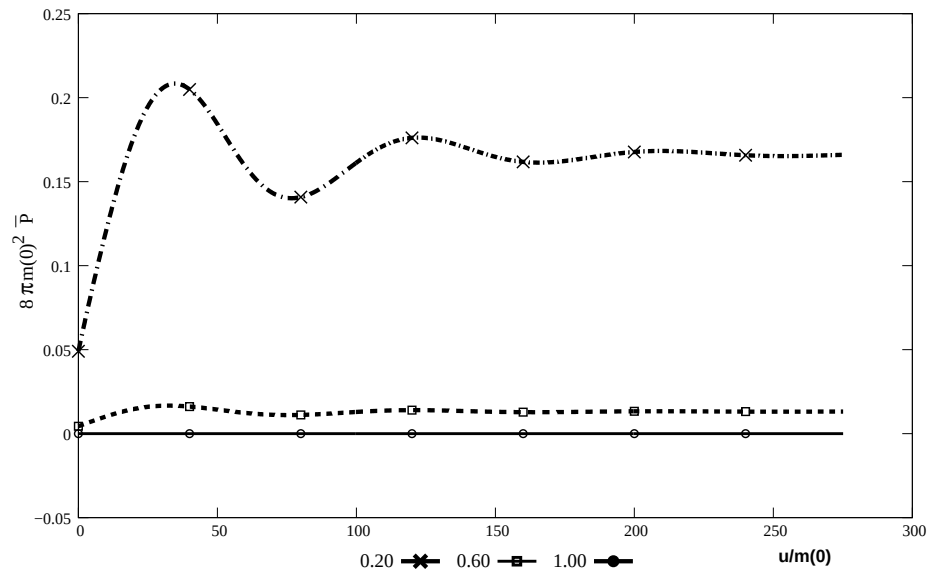


Figura 5.8: Presión normalizada $8\pi m(0)^2 \bar{P}$ en función de la variable temporal $\frac{u}{m(0)}$ para el modelo de Tolman V

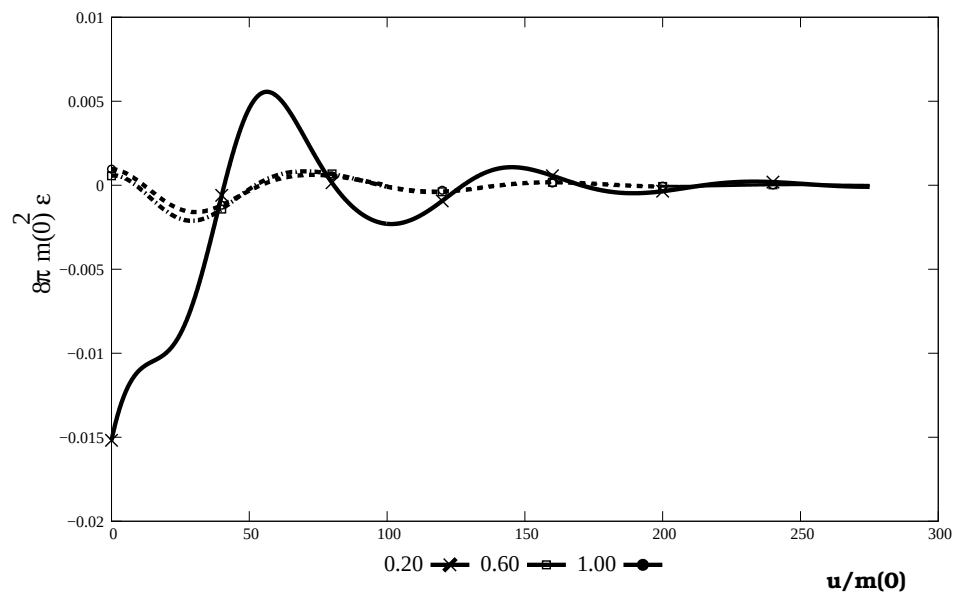


Figura 5.9: Radiación $8\pi m(0)^2 \varepsilon$ para el modelo de Tolman V

Capítulo 6

Conclusiones

Describimos una serie de procedimientos informáticos utilizados en Relatividad General apoyándonos en las facilidades de una plataforma integrada como el paquete Maple y GRTensorIII. Estos procedimientos de álgebra computacional, computación numérica y gráfica permitieron Modelar el colapso estelar de esferas autogravitantes mediante las ecuaciones de campo de Einstein y sus ecuaciones de conservación. Además su implementación permite facilitar el cálculo algebraico en otras aplicaciones con fines de investigación como de docencia avanzada en Matemáticas.

Se realizaron simulaciones numéricas para estudiar el comportamiento de un fluido perfecto con simetría esférica utilizando el esquema previamente establecido en [75, 7]. Este algoritmo se basa en que las soluciones de las ecuaciones de campo de Einstein tienen la característica de autosemejanza y permiten transformar las ecuaciones en derivadas parciales (PDE) en otras ecuaciones diferenciales ordinarias equivalentes. Adicionalmente, es necesario, junto con las ecuaciones de conservación, establecer una Ecuación de Estado (EOS) para completar la simulación del Colapso Gravitacional.

Adicionalmente se ha mostrado la generalización de las simulaciones para tres ecuaciones de Estado diferentes: una ecuación con densidad constante (Modelo de Schwarzschild), otra referida a materia nuclear (Tolman VI) y la versión paramétrica del caso ultrarelativista o solución V de Tolman, mientras que hasta hoy, solo se habían obtenido las simulaciones numéricas para el caso de fluido ultrarelativista. . Una vez realizada la integración numérica de las variables de superficie, se pueden calcular algunas variables físicas (radio, presión, densidad, radiación, etc.) mediante las ecuaciones de campo, en varias capas de la distribución esférica del fluido. Esto permite describir la evolución temporal de cada variable física en Novas y Supernovas.

Cada una de las soluciones obtenidas presentan características especiales. Por ejemplo, La solución de Schwarzschild de densidad constante, no manifiesta flujo de radiación apreciable (no radía). La solución VI de Tolman (materia nuclear) nos muestra que la condición del pulso de radiación gaussiano no es necesaria ya que se puede obtener el pulso de radiación utilizando las ecuaciones de conservación y que, dependiendo de las condiciones iniciales, se puede obtener una solución asintóticamente estable. Sin embargo, la solución V de Tolman con una selección adecuada de las condiciones iniciales, describe oscilaciones amortiguadas que ya se han predicho en otros escenarios de Relatividad Numérica [[108, 110].

Se espera en un futuro extender el presente trabajo y relacionar las condiciones iniciales de las

variables de superficie (A, M, Ω) y los valores que fijan los parámetros del movimiento oscilatorio (pseudoperiodo T , la fase φ y el tiempo de relajación τ) con la ventaja de poder utilizar una amplia gama de EOS. Ello permitiría relacionar los valores iniciales con los parámetros de la oscilación y con otros modelos de aplicación campo de la Relatividad Numérica [111, 110, 112].

Bibliografía

- [1] M. E. Cahill and A. H. Taub. Spherically symmetric similarity solutions of the einstein field equations for a perfect fluid. *Communications in Mathematical Physics*, 21(1):1–40, March 1971.
- [2] Douglas M. Eardley. Self-similar spacetimes: Geometry and dynamics. *Communications in Mathematical Physics*, 37(4):287–309, December 1974.
- [3] P. S. Wesson. An exact solution to einstein’s equations with a stiff equation of state. *Journal of Mathematical Physics*, 19:2283–2284, January 1978.
- [4] C. B. Collins and J. M. Lang. A class of self-similar perfect-fluid spacetimes, and a generalisation. *Classical and Quantum Gravity*, 4(1):61–78, January 1987.
- [5] B. J. Carr and A. A. Coley. Topical review: Self-similarity in general relativity. *Classical and Quantum Gravity*, 16(7):R31–R71, jul 1999.
- [6] G. S. Hall. *Symmetries and curvature structure in general relativity*, volume 46. 2004.
- [7] Víctor Medina. A self-similar solution of a fluid with spherical distribution in general relativity. *Indian Journal of Physics*, 96(1):317–328, jan 2022.
- [8] Jorge L. deLyra, Rodrigo de A. Orselli, and C. E. I. Carneiro. Exact solution of the einstein field equations for a spherical shell of fluid matter. *General Relativity and Gravitation*, 55(5):68, May 2023.
- [9] Ragab Mohamed Gad and Awatif Al-Jedani. Self-similar solutions of a bianchi type-iii model with a perfect fluid and cosmic string cloud in riemannian geometry. *Symmetry*, 15(9), 2023.
- [10] John G. Fletcher, Reg Clemens, Richard Matzner, Kip S. Thorne, and Barbara A. Zimmerman. Computer programs for calculating general-relativistic curvature tensors. *Astrophysical Journal, Letters*, 148:L91, May 1967.
- [11] M. J. Levine. Dirac matrix and tensor algebra on a digital computer. pages 454–455, 1967.
- [12] R. A. d’Inverno. Alam - atlas lisp algebraic manipulator. *The Computer Journal*, 12(2):124–127, 01 1969.
- [13] Raymon A. D’Inverno. Algebraic computing in general relativity. *General Relativity and Gravitation*, 6(6):567–593, dec 1975.

- [14] R.A. D’Inverno. Computer Algebra: from the Visible to the Invisible. *General Relativity and Gravitation*, 38:1095–1108, jun 2006.
- [15] P. Musgrave, D. Pollney, and K. Lake. Grtensorii: a computer algebra system for general relativity. In S.P. Braham, J.D. Gegenberg, and R.J. McKellar, editors, *Proceedings of the 6th Canadian Conference on General Relativity and Relativistic Astrophysics*, page 313, 1997.
- [16] Davies G. pertutils.mpl by george davies written for grtensorii. PertUtils, 1998. <http://grtensor.phy.queensu.ca/NewDemo/Blackholes/Utils/pertutils.mpl>.
- [17] C.A. Sporea and D.N. Vulcanov. Using maple + grtensorii in teaching basics of general relativity and cosmology. *Rom. Rep. Phys.*, (68):29–40, 2016.
- [18] Lake K. Musgrave P., Pollney D. Grtensoriii package for maple. GRTensorIII, 2021. <https://github.com/grtensor/grtensor>.
- [19] Víctor Medina. Using maple and grtensoriii in relativistic spherical models. *arXiv e-prints*, page arXiv:2205.11985, may 2022.
- [20] Kasper Peeters. Cadabra: a field-theory motivated symbolic computer algebra system. *Computer Physics Communications*, 176(8):550 – 558, 2007.
- [21] L. Brewin. A brief introduction to Cadabra: A Tool for Tensor Computations in General Relativity. *Computer Physics Communications*, 181:489–498, mar 2010.
- [22] Éricourgoulhon and Marco Mancini. Symbolic tensor calculus on manifolds: a SageMath implementation. *Les cours du CIRM*, 6(1):1–54, 2018.
- [23] Gabriel Pascu. On some foundational aspects of teaching differential geometry and general relativity with sagemath. In *American Institute of Physics Conference Series*, volume 2843 of *American Institute of Physics Conference Series*, page 050010. AIP, jul 2023.
- [24] Review by: Jenny Baglivo. Computer algebra systems: Maple and mathematica. *The American Statistician*, 49, 02 1995.
- [25] A.V. Korolkova, D.S. Kulyabov, and L.A. Sevastyanov. Tensor computations in computer algebra systems. *arXiv e-prints*, feb 2014.
- [26] Stephen Wolfram. A class of models with the potential to represent fundamental physics. *arXiv e-prints*, page arXiv:2004.08210, April 2020.
- [27] Bruce W. Char, Keith O. Geddes, W. Morven Gentleman, and Gaston H. Gonnet. The design of maple: A compact, portable and powerful computer algebra system. In *In EUROCAL ’83, Proceedings of the International Symposium on Symbolic and Algebraic Computation*, EUROCAL ’83, pages 101–115, London, UK, UK, apr 1983. Springer-Verlag.
- [28] Joseph L. Zachary. Introduction to scientific programming: Computational problem solving using maple and c. *American Journal of Physics*, 65, 1997.

- [29] V.L. Kalashnikov. Introduction to Relativistic Astrophysics and Cosmology through Maple. *arXiv General Relativity and Quantum Cosmology e-prints*, mar 2001.
- [30] Dumitru N. Vulcanov and Gabriela Ciobanu. Maple procedures in teaching the canonical formalism of general relativity. *arXiv e-prints*, page physics/0010053, oct 2000.
- [31] Éric Gourgoulhon and Marco Mancini. Symbolic tensor calculus on manifolds: a SageMath implementation. *Les cours du CIRM*, 6(1), 2018. talk:1.
- [32] M.A.H. MacCallum. Computer algebra in gravity research. *Living Rev Relativ*, 21:6, aug 2018.
- [33] D.A. Bolotin and S.V. Poslavsky. Introduction to Redberry: a Computer Algebra System designed for tensor manipulation. *arXiv e-prints*, feb 2013.
- [34] S. Poslavsky and D. Bolotin. Redberry: a Computer Algebra System designed for tensor manipulation. In *Journal of Physics Conference Series*, volume 608 of *Journal of Physics Conference Series*, page 012060, may 2015.
- [35] Zimmermann P.; Casamayou A.; Cohen N.; Connan G. *Computational Mathematics with SageMath // Polynomials*, volume 10.1137/1.9781611975468, chapter Chapter 7, pages 125–151. SIAM, 2018.
- [36] V.P. Gerdt. Analytical calculations in high energy physics by computer. *Computer Physics Communications*, 20:85–90, sep 1980.
- [37] V.P. Gerdt, O.V. Tarasov, and D.V. Shirkov. Reviews of topical problems: Analytic calculations on digital computers for applications in physics and mathematics. *Soviet Physics Uspekhi*, 23:59–77, jan 1980.
- [38] Friedrich Jegerlehner (eds.) Friedrich Jegerlehner (auth.). *The Anomalous Magnetic Moment of the Muon*, volume 274 of *Springer Tracts in Modern Physics 274*. Springer-Verlag Berlin Heidelberg, 2 edition, 2017.
- [39] H. Strubbe. Development of the schoonschip program. *Computer Physics Communications*, 18:1–5, sep 1979.
- [40] Martinus J. G. Veltman and David N. Williams. Schoonschip '91. *The University of Michigan Ann Arbor, MI 48109-1120*, 8, 06 1993.
- [41] Stefan Weinzierl. Computer algebra in particle physics, 2002.
- [42] M.J. Levine and R. Roskies. Hyperspherical approach to quantum electrodynamics: sixth-order magnetic moment. *Physical Review D*, 9(2):421–429, January 1974.
- [43] M. J. Levine and R. Roskies. Ashmedai and a large algebraic problem. In *Proceedings of the Third ACM Symposium on Symbolic and Algebraic Computation*, SYMSAC '76, pages 359–364, New York, NY, USA, 1976. ACM.

- [44] A. C. Hearn. Computation of algebraic properties of elementary particle reactions using a digital computer. *Communications of the ACM*, 9(8):573–577, aug 1966.
- [45] Anthony C. Hearn. Applications of symbol manipulation in theoretical physics. *Commun. ACM*, 14(8):511–516, August 1971.
- [46] W.S. Brown and A.C. Hearn. Applications of symbolic algebraic computation. *Computer Physics Communications*, 17(1):207 – 215, 1979.
- [47] A.G. Grozin. *Using REDUCE in High Energy Physics*. Cambridge University Press, 2005, sep 2005.
- [48] V.E. Lewis, Massachusetts Institute of Technology, Massachusetts Institute of Technology. Laboratory for Computer Science, United States. Army Research Office, and United States. Office of Naval Research. *Proceedings of the 1979 MACSYMA Users' Conference: Washington, D.C., June 20-22, 1979*. MIT Laboratory for Computer Science, 1979.
- [49] Richard Pavelle and Paul S. Wang. Macsyma from f to g. *Journal of Symbolic Computation*, 1(1):69 – 100, 1985.
- [50] Joel Moses. Macsyma: A personal history. *Journal of Symbolic Computation*, 47(2):123 – 130, 2012.
- [51] Matthias Steinhauser, Takahiro Ueda, and Jos A.M. Vermaseren. Parallel versions of form and more. *Nuclear and Particle Physics Proceedings*, 261-262:45–57, apr 2015.
- [52] B. Ruijl, T. Ueda, and J. Vermaseren. Form version 4.2. *arXiv e-prints*, July 2017.
- [53] J. Vermaseren. New features of form and future plans. In *13th International Symposium on Radiative Corrections; Applications of Quantum Field Theory to Phenomenology. 25-29 September, 2017 . St. Gilgen, Austria. Online at https://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=290, id.22, page 22, January 2017.*
- [54] Stephen Wolfram. *Some Topics in Theoretical High-Energy Physics*. PhD thesis, CALIFORNIA INSTITUTE OF TECHNOLOGY., jan 1980.
- [55] Stephen Wolfram. Undecidability and intractability in theoretical physics. *prl*, 54(8):735–738, February 1985.
- [56] Stephen Wolfram. The practical and theoretical future of computation in physics. In *APS March Meeting Abstracts*, volume 2019 of *APS Meeting Abstracts*, page F22.001, January 2019.
- [57] Portugal R. and Sautú S.L. Applications of maple to general relativity. *Computer Physics Communications*, 105, 1997.
- [58] D.N. Vulcanov. Gravity, torsion, dirac field and computer algebra using maple and reduce. *arXiv General Relativity and Quantum Cosmology e-prints*, September 2002.

- [59] Wolfgang Schreiner, Christian Mittermaier, and Karoly Bosa. Distributed maple: parallel computer algebra in networked environments. *Journal of Symbolic Computation*, 35(3):305 – 347, 2003.
- [60] D.N. Vulcanov and V.D. Vulcanov. Maple libraries for cosmology. *eprint arXiv:cs/0409006*, sep 2004.
- [61] Frank Y. Wang. *Physics with MAPLE: The Computer Algebra Resource for Mathematical Methods in Physics*. Physics Textbook. Wiley-VCH, 1 edition, may 2006.
- [62] SWMATH. Information service for mathematical software. SWMATH, 2021. <https://www.swmath.org/>.
- [63] SIGSAM Bulletin. Special interest group on symbolic & algebraic manipulation. Dictionary Of Computing, 2021. <http://www.sigsam.org/Resources/Software.html>.
- [64] Maxima A Computer Algebra System. Other free computer algebra systems. MAXIMA, 2021. <https://maxima.sourceforge.io/compalg.html>.
- [65] RISC Research Institute for Symbolic Computation. Research institute for symbolic computation. Research Institute for Symbolic Computation, 2021. <https://risc.jku.at/software/>.
- [66] FOLDOC. Free on line dictionary of computing. Dictionary Of Computing, 2021. <https://foldoc.org/symbolic+mathematics>.
- [67] Computer Algebra Netherlands. Foundation can (computer algebra netherlands). Computer Algebra Netherlands, 2021. <https://computeralgebra.nl/>.
- [68] W. S. Brown. Proceedings of the first acm symposium on symbolic and algebraic manipulation. In SYSMAC '66, editor, *A language and system for symbolic algebra on a digital computer ALTRAN*, number 1 in SAAC, pages 501–540, 1966.
- [69] J. P. Fitch and H. I. Cohen. Using camal for algebraic computations in general relativity. *General Relativity and Gravitation*, 11(6):411–418, dec 1979.
- [70] S. J. Campbell and J. Wainwright. Algebraic computing and the newman-penrose formalism in general relativity. *General Relativity and Gravitation*, 8(12):987–1001, dec 1977.
- [71] J. Wainwright. Algebraic computing in general relativity. In *General Relativity and Gravitation 1977*, page 19, aug 1977.
- [72] A. C. Hearn. The reduce program for computer algebra. In A. Visconti, editor, *3th Colloquium on Advanced Computing Methods in Theoretical Physics*, pages AV/1–AV/19, Mar-seilles, jun 1973. Centre de Physique Théorique CNRS. CNRS technical report 73/P.550.
- [73] Michael Monagan and Roman Pearce. Poly: A new polynomial data structure for maple 17. *ACM Commun. Comput. Algebra*, 46(3/4):164 – 167, jan 2013.

- [74] P. Musgrave and K. Lake. Engineering applications of grtensorii 1:elasticity-finite deformations. *GRTensorII*, 1(1):22, aug 1998.
- [75] V. Medina and N. Falcon . Relativistic model for radiating spherical collapse. *Progress in Physics*, 14(1):46–53, jan 2018.
- [76] Noemi Castiñeyra. Trabajo de ascenso asistente: Técnicas de animación asistida por computadora., 1992.
- [77] F. Aguirre, H. Henández, L. A. Núñez, and et al. ESFERAS: An Algebraic, Numerical and Visual Environment for Radiating Spheres in General Relativity. In P. S. Letelier and W. A. Rodrigues, Jr., editors, *Gravitation: the Spacetime Structure, SILARG VIII*, page 582, 1994.
- [78] Noemi Castiñeyra. Trabajo de ascenso agregado: Interfaces y algoritmos para aplicaciones de visualización científica., 1996.
- [79] J. D. Bekenstein. Hydrostatic Equilibrium and Gravitational Collapse of Relativistic Charged Fluid Balls. *Phys. Rev. D*, 4:2185–2190, oct 1971.
- [80] A. Patiño and H. Rago. Self Gravitating Charged Fluid in a Single Coordinate System. *Astrophysics and Space Science*, 257:213–221, 1997.
- [81] L. Herrera, J. Jiménez, and G. J. Ruggeri. Evolution of radiating fluid spheres in general relativity. *Phys. Rev. D*, 22:2305–2316, nov 1980.
- [82] M. Cosenza, L. Herrera, M. Esculpi, and L. Witten. Evolution of radiating anisotropic spheres in general relativity. *Phys. Rev. D*, 25:2527–2535, may 1982.
- [83] V. Medina, L. Núñez, H. Rago, and A. Patiño. Evolution of radiating charged spheres in general relativity. *Canadian Journal of Physics*, 66:981–986, nov 1988.
- [84] A. Patiño and H. Rago. A Sphere Contraction in General Relativity . *Lett. Nuovo Cimento*, 38:321–328, oct 1983.
- [85] W. Barreto and A. da Silva. Gravitational collapse of a charged and radiating fluid ball in the diffusion limit. *General Relativity and Gravitation*, 28:735–747, jun 1996.
- [86] A. di Prisco, N. Falcón, L. Herrera, M. Esculpi, and N. O. Santos. Pre-relaxation Processes in a Radiating Relativistic Sphere. *General Relativity and Gravitation*, 29:1391–1405, nov 1997.
- [87] W. Barreto, B. Rodríguez, and H. Martínez. Radiating Fluid Spheres in the Effective Variables Approximation. *Astrophysics and Space Science*, 282:581–593, nov 2002.
- [88] L. Herrera, W. Barreto, A. di Prisco, and N. O. Santos. Relativistic gravitational collapse in noncomoving coordinates: The post-quasistatic approximation. *Phys. Rev. D*, 65(10):104004, may 2002.
- [89] F. Aguirre, L. A. Nunez, and T. Soldovieri. Variable Eddington Factor and Radiating Slowly Rotating Bodies in General Relativity. *ArXiv General Relativity and Quantum Cosmology e-prints*, mar 2005.

- [90] R. C. Tolman. Static Solutions of Einstein's Field Equations for Spheres of Fluid. *Physical Review*, 55:364–373, feb 1939.
- [91] Kasper Peeters. Introducing cadabra: A symbolic computer algebra system for field theory problems. 1 2007.
- [92] C. Heinicke and F.W. Hehl. Computer algebra in gravity. *arXiv General Relativity and Quantum Cosmology e-prints*, may 2001.
- [93] L.R.U Manssur and R Portugal. The canon package: a fast kernel for tensor manipulators. *Computer Physics Communications*, 157(2):173 – 180, 2004.
- [94] J. Martin-Garcia. The xact package. <http://www.xact.es/>, mar 2004.
- [95] Gaston H. Gonnet Bruce Char, Keith O. Geddes. The design of maple: A compact, portable, and powerful computer algebra system. <http://www.maplesoft.com>, ene 1982.
- [96] Albert D. Rich and David R. Stoutemyer. *Capabilities of the mumath-79 computer algebra system for the intel-8080 microprocessor*, volume 10.1007/3-540-09519-5. 1979.
- [97] David R. Stoutemyer. Picomath-80-8482. *ACM SIGSAM Bulletin*, 14, 08 1980.
- [98] Do Hoang Son. Feynman loop integrals and their automatic computer-aided evaluation. *Johannes Gutenberg-Universitat Mainz*, 1(1):124, may 2003.
- [99] Bauer Christian. Use of computer algebra in the calculation of feynman diagrams. *Johannes Gutenberg-Universitat Mainz*, 1(1):156, nov 2004.
- [100] P. Musgrave, D. Pollney, and K. Lake. Grtensor ii. <http://grtensor.phy.queensu.ca>, feb 1998.
- [101] D. Mihalas and B. W. Mihalas. *Foundations of radiation hydrodynamics*. 1984.
- [102] Dimitri Mihalas, Barbara Weibel Mihalas, Albert Fu, and W. David Arnett. Foundations of radiation hydrodynamics. *Physics Today*, 39(6):90, jan 1986.
- [103] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, USA, second edition, 1992.
- [104] Stoer J. and Bulirsch R. Numerical treatment of ordinary differential equations by extrapolation methods. *Numerische Mathematik*, 8:1–13, 1966.
- [105] W.H. Press. *Numerical Recipes in FORTRAN: The Art of Scientific Computing*. Number v. 1-2 in Fortran numerical recipes. Cambridge University Press, 1992.
- [106] K. D. Krori, P. Borgohain, and R. Sarma. Nonstatic radiating spheres in general relativity. *Phys. Rev. D*, 31:734–741, feb 1985.
- [107] M. W. Choptuik. Universality and scaling in gravitational collapse of a massless scalar field. *Physical Review Letters*, 70:9–12, jan 1993.

- [108] Carsten Gundlach. Critical gravitational collapse with angular momentum: From critical exponents to scaling functions. *Physical Review D*, 65(6):064019, Mar 2002.
- [109] Thomas W. Baumgarte, Carsten Gundlach, and David Hilditch. Critical phenomena in the gravitational collapse of electromagnetic waves. *arXiv e-prints*, page arXiv:1909.00850, Sep 2019.
- [110] J. Celestino and T. W. Baumgarte. Critical collapse of ultrarelativistic fluids: Damping or growth of aspherical deformations. *Phys. Rev. D*, 98(2):024053, jul 2018.
- [111] Carsten Gundlach and Thomas W. Baumgarte. Critical gravitational collapse with angular momentum. ii. soft equations of state. *Physical Review D*, 97(6):064006, Mar 2018.
- [112] Carsten Gundlach, Thomas W. Baumgarte, and David Hilditch. Critical phenomena in gravitational collapse with two competing massless matter fields. *arXiv e-prints*, page arXiv:1908.05971, Aug 2019.

Publicaciones

Using Maple and GRTensorIII in relativistic spherical models

*Medina V.**

Abstract

This article presents some aspects and experience in the use of algebraic manipulation software applied to general relativity. Some years ago certain results were reported using computer algebra platforms, but the growing popularity of graphical platforms such as Maple allows us to approach the problem of the simplifications of many expressions from another point of view. Some simple algebraic programming procedures are presented (in Maple with the GRTensorIII package) to obtain and study material distributions with spherical symmetry and to search for exact solutions of the Einstein field equations. The purpose is to show how useful a computer algebra system can be. All calculations were performed using the GRTensorIII computer algebra package, which runs on Maple 2017, along with several Maple routines that we have used specifically for the simplification of many of the algebraic expressions that are very common in this type of problem.

Keywords: Computer algebra, General relativity , Gravitation theory , Algorithms, Maple , GRTensorIII

PACS: 95.30.Sf, 04.20.- q, 04.40.-b

*Departamento de Física - Facultad de Ingeniería - Universidad de Carabobo. Valencia, Venezuela: vmedina@uc.edu.ve

1 Introduction

Computer algebra systems (CAS) have a wide variety of applications in fields that require time-consuming, difficult-to-perform, and error-prone calculations when done manually. They are used more and more frequently and especially when it is necessary to complete a calculation in pages after pages for many hours or perhaps several days. As early as the late 1950s and early 1960s, various programs appeared [1, 2, 3, 4] aimed at demonstrate that in the scientific field it is possible to go beyond the purely numerical area and use them to carry out a symbolic calculation. For reasons such as these, various members of the computer algebra family of languages were created, and in fact many were created to perform specific calculations of great complexity in fields such as electronic optics [5], celestial mechanics [6], quantum electrodynamics [7, 8, 9, 10, 11, 12] or general relativity [13, 14, 15, 16]. Computer algebra for general relativity (GR) has a long history, beginning almost as early as computer algebra itself in the 1960s.

The first GR program was GEOM, written by J.G. Fletcher [17]. Its main ability was to compute the Riemann tensor of a given metric. In 1969, R. A. d’Inverno [18] developed ALAM (for Atlas Lisp Algebraic Manipulator) [19] and used it to compute the Riemann and Ricci tensors of the Bondi metric. According to [20], Bondi and his collaborators took 6 months to complete the original calculations, while the ALAM calculation took 4 minutes and resulted in the discovery of 6 errors in the original paper. Since then, numerous packages have been developed and various investigations have been carried out using manipulative algebraic software (CAS) and in many cases the calculations were of such length that they would have been prohibitively expensive to complete without the aid of a computer [21]. Its main advantage is the ability to handle a large number of algebraic calculations and this particularity has allowed advances in fields of theoretical physics such as GR or High Energy Physics (HEP).

Many of these problems have used some free, open source, general purpose software with an emphasis on tensor calculus for GR, such as Java-based REDBERRY [22, 23], SAGE [24] and SAGEMANIFOLDS written in Python, CADABRA[25, 26] developed in C++ and Python. Others, applied specifically for HEP, have been designed based on special algorithms [27, 28, 12] and implemented in programs like SCHOONSHIP [29, 30], designed by M. Veltman, ASHMEDAI [31, 32, 33] by M. Levine, REDUCE [7, 8, 9, 11] by A. Hearn, MACSYMA (to become MAXIMA in 1998) [34, 35, 36] by J. Moses developed at MIT, or more recent FORM [37, 38, 39] by J. Vermaseren. Proprietary software such as MATHEMATICA [40, 41, 42, 43] by S. Wolfram or MAPLE [44, 45, 46, 47, 48, 49, 50, 51, 52] from B. Char developed at the University of Waterloo. The CAS programs mentioned above constitute only a very small part of the available applications, special purpose and general systems that can be consulted in repositories and lists, maintained and frequently updated [53, 54, 55, 56, 57, 58]. Of all the CAS applications mentioned above, we are going to refer in this work to **MAPLE**.

Maple is a general purpose CAS, initially developed at the University of Waterloo as a result of discussions

on the state of symbolic computing in the 1980s. At the time, large systems such as ALTRAN [59], CAMAL [13, 14, 15], REDUCE [60] and MACSYMA [34], based on the computer technology of the 1960s, they decided to design a new system from scratch, taking advantage of the advances in software engineering available and the lessons of experience. Its basic design features (for example, 61 elementary data structures, input/output, number arithmetic, and elementary simplification) are encoded in a low-level language for efficiency. An important property is that most of the algebraic facilities of the system are implemented using the high-level user language. The basic system, or kernel, is compact and efficient enough to be practical to use for a shared environment or on personal computers with very little main memory. Library functions are loaded into the system as needed, adding features such as polynomial factorization, equation solving, indefinite integration, and matrix manipulation to the system. The modularity of this design allows users to demand available computing resources in proportion to their actual use. It has specialized libraries for elementary and special mathematical functions, it offers support for symbolic and numerical computation with exact results, it can handle a wide set of equation systems, including Diophantine equations, ordinary differential equations (ODE), partial differential equations (PDE), Differential Algebraic Equations (DAE), Delay Differential Algebraic Equations (DDE) and recurrence relations. Initially the kernel of the system was written in macros that could be translated by a locally developed macroprocessor (called Margay) into versions of the kernel in the C programming language for various operating systems; currently only C is used. The GUI was first released with version 5 (Maple V) and continued to be numbered until Maple 18 and then changed to a yearly label. Due to the low demand for main memory to run the kernel and the modular design for many of the possible user applications in a high-level language, it has become one of the main symbolic algebra systems used by many researchers and engineering corporations. worldwide [62, 63]. These applications or packages can be programmed by the user and many can be found on the web 64 as *GRTensorII* or its update *GRTensor III* . (in short *GRTensor*).

GRTensor is a computer algebra package for performing calculations in the general area of differential geometry 65. Its main objective is the computation of tensor components in curved spacetimes specified in terms of a metric or set of basis vectors. The library relies on a series of special commands starting with "gr" (for example, *grcalc*, *grdisplay*, *gralter*, *grdefine*, etc.) to deal with a series of (pre) geometric objects. defined as the metric tensor, Ricci tensor and scalar, Einstein tensor, Christoffel symbols, etc. This library of objects can be extended to define new tensors, or use the Newman-Penrose formalism. Although originally designed for use in the field of general relativity, *GRTensor* is useful in many other fields 66 There is a version for **MATHEMATICA** called *GRTensorM*. The *GRTensorII* package was originally developed for *Maple V* and can be run with versions from *Maple V Release 3* to *Maple 13*. The *GRTensor III* version runs as of *Maple 15*. All documentation and software are distributed free of charge to help both for research and teaching.

The final objective of this article is to present the calculations, especially in Sec. 2 that have been carried out with an emphasis on the methods, packages and techniques of computational algebra that we use in a spreadsheet developed for the *GRTensorIII* package running on the *Maple 17* platform. Initially, it was written in *GRTensorII* running on top of *Maple 13* and when upgrading *GRTensor* to version III, it became necessary to upgrade the *Maple* version as well. The spreadsheet follows the algorithm described in 67, 68 starting from a spherically symmetric perfect fluid distribution of density $\hat{\rho}$, radial pressure $\hat{P}$, tangential pressure $\hat{P}_t$, a flux of unpolarized radiation moving in the radial direction with density $\hat{\varepsilon}$ and $3\hat{\mu}$ the isotropic radiation of the energy density, in a local Minkowskian system. By performing the corresponding coordinate transformations, to a radiative (Bondi) coordinate system, we can construct an Impulse Energy Tensor that must satisfy both the Einstein field equations ($G_{ab} = \kappa T_{ab}$) as the conservation equations ($T_{;b}^{ab} = 0$). When calculating these equations, it is necessary to simplify them and for this, some 69, procedures and functions have been used, as in 70. In Sec. 3 some comments on the calculations made are presented and in the last section the conclusions and the possibility of extending this procedure to other astrophysical scenarios are detailed.

2 Algorithm structure

As indicated in the previous section, the created spreadsheet basically follows the procedure used in [71] and for the charged case in [72] with the modifications made in 67, 68. In the original sheet, the commands are grouped by sections and each of the sections has a name that suggests the calculation to be carried out. In this build for the article, commands (in red) and output (in blue) from the Maple interpreter will be reproduced. All the commands to perform the calculations of the sheet are present in this work, however, not all the outputs (in blue) were placed and the most extensive ones were edited.

2.1 Login

To start the Maple session we loaded the *GRTensorIII* libraries, the expression simplification routines were taken from Davies [70] and the directory from where the metrics will be loaded. We start with a reset statement to ensure that you have a new session of the sheet,

```
>restart;
>grtw();
```

```

"
GRTensorIII v2.2 Oct 1, 2018"
" Copyright 2018, Peter Musgrave, Denis Pollney, Kayll Lake"
" Latest version is at http : //github.com/grtensor/grtensor"
" For help ?grtensor"
" Support/contact grtensor3@gmail.com"
module() ...end module()

```

(1)

```
>with(grtensor): groptions():
```

The expression simplification routines were taken from Davies [70]

```
>read '/home/victor/relatividad/maple/loadmpl/pertutils.mpl';
```

2.2 Minkowskian local flat space

It is assumed that the region of the space to be considered is composed of energy density anisotropic material $\hat{\rho}$, radial pressure $\hat{P}$, tangential pressure $\hat{P}_t$, radiation isotropic energy density $3\hat{\mu}$ and non-polarized radiation of energy density $\hat{\xi}$ propagating in a radial direction. We are going to establish the metric of this region of space.

```
>qload(minkowski);
Calculated ds for minkowski (0.001000 sec.)
```

```

Default spacetime = minkowski
For the minkowski spacetime :
Coordinates
x(up)
xα = [t x y z]
Line element
ds2 = dt2 - dx2 - dy2 - dz2
The Minkowski metric - plana-

```

(2)

The covariant components of the metric tensor of the local Minkowski system are calculated and displayed. You can use two **GRTensorIII** instructions: *grcalc()* to calculate and *grdisplay()* to show the result. However, *grcalcd()*,

allows you to do the calculation and shows the result, combining the two previous instructions.

```
>grcalcd(g(dn,dn));
```

Let's set the values for the contravariant unit vectors in the Minkowski reference system. We will use them to calculate the components of the tensor in the local system

```
>grdef('u0{^a }:=kdelta{$t ^a} '): grdef('u1{^a }:=kdelta{$x ^a} '):  
grdef('u2{^a }:=kdelta{$y ^a} '): grdef('u3{^a }:=kdelta{$z ^a} '):
```

And to establish the Stress-energy tensor of this region, let's start with: Energy density anisotropic material ρ , Radial pressure P and Tangential pressure P_t

```
>grdef('TM{^a b}:= (rho_M + P_t)*u0{^a }*u0{^b }- P_t *g{^a b} + (P_M - P_t)*u1{^a }*u1{^b } '):  
>grcalcd(TM(dn, dn));
```

Isotropic radiation of energy density 3μ

```
>grdef('Tp{^a b}:= 3*u0{^a }*u0{^b } +u1{^a }*u1{^b }+u2{^a }*u2{^b }+u3{^a }*u3{^b } '):
```

The component corresponding to Polarized radiation

```
>grcalcd(Tp(dn, dn));
```

Unpolarized radiation of energy density ξ propagating in the radial direction

```
> grdef(' v{a }:= [1, -1, 0, 0 ] ');  
> grdef('Tnp{a b}:= xi *v{a }*v{b } ');  
> grcalcd(Tnp(dn, dn));
```

Bringing together the three parts of the Stress-Energy tensor, we get the stress-energy tensor Matter + radiation:

```
> grdef('T0{a b}:=TM{a b}+mu*Tp{a b}+Tnp{a b} ');  
> grcalcd(T0(dn, dn));
```

```
Calculated T0(dn,dn) for minkowski (0.003000 sec.)
```

'CPUTime' = 0.005

For the minkowski spacetime :

T0(dn, dn)

T0(dn, dn)

$$T0_{ab} = \begin{bmatrix} 3\mu + \rho_M + \xi & -\xi & 0 & 0 \\ -\xi & P_M + \mu + \xi & 0 & 0 \\ 0 & 0 & P_t + \mu & 0 \\ 0 & 0 & 0 & P_t + \mu \end{bmatrix} \quad (3)$$

Redefining the radiation and matter variables of the stress-energy tensor:

```
> grmap(T0(dn, dn), subs, rho_M = rho[0]-3*mu, 'x');
> grmap(T0(dn, dn), subs, P_M = P[0]-mu, 'x');
> grmap(T0(dn, dn), subs, P_t = P[t]-mu, 'x');
> grdisplay(T0(dn,dn));
```

For the minkowski spacetime :

$$T0(dn, dn)$$

$$T0(dn, dn)$$

$$T0_{ab} = \begin{bmatrix} \rho_0 + \xi & -\xi & 0 & 0 \\ -\xi & P_0 + \xi & 0 & 0 \\ 0 & 0 & P_t & 0 \\ 0 & 0 & 0 & P_t \end{bmatrix} \quad (4)$$

2.3 Lorentz transformation

Let's suppose that you have an observer moving in relation to the local system Minkowskian, with a radial velocity ω . The components of the Stress-Energy tensor in this new system of Lorentz, will be given by the relationship:

$$\bar{T}_{\mu\nu} = \Lambda_{\mu}^{\alpha} \Lambda_{\nu}^{\beta} \hat{T}_{\alpha\beta}$$

where Λ_{μ}^{α} is the transformation matrix

```
> grdef('umsqrto := 1/sqrt(1-omega^2) ');
grdef('Lambda{^a b}:= umsqrto *kdelta{^a $t}*kdelta{$t b}
-omega* umsqrto *kdelta{^a $x}* kdelta{$t b} -omega* umsqrto *kdelta{^a $t}* kdelta{$x b}
+ umsqrto *kdelta{^a $x}* kdelta{$x b}+ kdelta{^a $y}* kdelta{$y b}+
kdelta{^a $z}* kdelta{$z b}');
```

```
> grcalcd(Lambda(up, dn));
```

CPU Time = 0.011

For the minkowski spacetime :

Lambda(up, dn)

$\Lambda(up, dn)$

$$\Lambda_{\mu}^{\alpha} = \begin{bmatrix} \frac{1}{\sqrt{1-\omega^2}} & -\frac{\omega}{\sqrt{1-\omega^2}} & 0 & 0 \\ -\frac{\omega}{\sqrt{1-\omega^2}} & \frac{1}{\sqrt{1-\omega^2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5)$$

Let's carry out the transformation operation now. To do it, we will establish this transformation as:

```
> grdef('T1{a b}:=Lambda{^c a}*Lambda{^d b}*T0{c d}');
```

```
> grcalcd(T1(dn, dn));
```

CPU Time = 0.012

For the minkowski spacetime :

T1 (dn, dn)

$\Lambda(up, dn)$

$$T1_{ab} = \begin{bmatrix} -\frac{\rho_0+2\omega\xi+\omega^2 P_0+\omega^2\xi}{-1+\omega^2} & \frac{\omega_0\rho_0+2\omega\xi+\xi+\omega P_0}{-1+\omega^2} & 0 & 0 \\ \frac{\omega_0\rho_0+2\omega\xi+\xi+\omega P_0}{-1+\omega^2} & -\frac{\omega^2\rho_0+\omega^2\xi+2\omega\xi+P_0+\xi}{-1+\omega^2} & 0 & 0 \\ 0 & 0 & P_t & 0 \\ 0 & 0 & 0 & P_t \end{bmatrix} \quad (6)$$

We obtain the expression of the Stress-energy tensor in the local system, with radial velocity ω .

2.4 Bondi Radiative Coordinate System

As the study we are doing, is related to radiation, then it is logical to assume that we must use a coordinate system according to the theme. Therefore, we are going to use Bondi's radiative coordinate system as in [73].

```
>qload(bondi);
```

```
Calculated ds for bondi (0.001000 sec.)
```


Default space time = bondi

For the bondi spacetime :

Coordinates

$x(up)$

$x^\alpha = [u, r, \theta, \phi]$

Line element

$$ds^2 = \frac{V(u,r) \cdot e^{2\beta(u,r)}}{r} du^2 + 2e^{2\beta(u,r)} dr \cdot du - r^2 d\theta^2 - r^2 \sin^2 \theta d\phi^2$$

The Bondi metric (Proc. Roy. Soc. A 269 21)

(7)

Changing the expression of the Stress-energy tensor of the local Minkowskiana metric to its structure in the Bondi radiation coordinate system.

```
>grdef('Vsr:=sqrt(V(u,r)/r)');
>grdef('Mu{^a b}:= exp(beta(u,r))*Vsr*kdelta{^a $u}*kdelta{$u b}+
exp(beta(u,r))/Vsr*kdelta{^a $u}*kdelta{$r b}+ exp(beta(u,r))/
Vsr*kdelta{^a $r}*kdelta{$r b}+r*kdelta{^a $theta}*kdelta{$theta b}+
r*sin(theta)*kdelta{^a $phi}*kdelta{$phi b}');
```

Defining and showing the matrix of the Minkowski local system transformation to the Lorentz system

```
>grcalcd(Mu(up, dn));
Calculated Vsr for bondi (0.002000 sec.)
Calculated grtensor:-kdelta(dn,dn) for bondi (0.002000 sec.)
Calculated grtensor:-kdelta(up,dn) for bondi (0.002000 sec.)
Calculated Mu(up,dn) for bondi (0.011000 sec.)
```

CPU Time = 0.021

For the bondi spacetime :

$Mu(up, dn)$

$M(up, dn)$

$$M^a{}_b = \begin{bmatrix} e^{2\beta(u,r)} \sqrt{\frac{V(u,r)}{r}} & \frac{e^{2\beta(u,r)}}{\sqrt{\frac{V(u,r)}{r}}} & 0 & 0 \\ 0 & \frac{e^{2\beta(u,r)}}{\sqrt{\frac{V(u,r)}{r}}} & 0 & 0 \\ 0 & 0 & r & 0 \\ 0 & 0 & 0 & r \sin \theta \end{bmatrix} \quad (8)$$

This matrix of transformation, allows us to obtain the expression of the stress-energy tensor of the local system of Minkowski to the system of radiative coordinates with

$$\bar{T}_{ab} = M^\alpha{}_a M^\beta{}_b \hat{T}_{\alpha\beta} \quad (9)$$

where $M^a{}_b$ is the transformation matrix. As we are performing an operation between two expressions with different metrics, we must specify the space - or metric - corresponding to each term of the multiplication

```
> grdef('TB<2>{a b}:=Mu{^c a} *Mu{^d b}*T1<1>{c d}');
```

by defining what is the scope of the definition of each term

```
> grcalcd(1 = minkowski, 2 = bondi, TB(dn, dn));
```

CPU Time = 0.202

For the bondi spacetime :

TB (dn, dn)

TB (dn, dn)

$$TB_{ab} = \begin{bmatrix} -\frac{e^{2\beta(u,r)} V(u,r) (\rho_0 + 2\omega\xi + \omega^2 P_0 + \omega^2 \xi)}{r(-1+\omega^2)} & -\frac{(\omega P_0 - \rho_0)}{\omega+1} e^{2\beta(u,r)} & 0 & 0 \\ -\frac{(\omega P_0 - \rho_0)}{\omega+1} e^{2\beta(u,r)} & -\frac{(\omega \rho_0 + \omega P_0 - \rho_0 - P_0)r}{V(u,r)(\omega+1)} & 0 & 0 \\ 0 & 0 & r^2 P_t & 0 \\ 0 & 0 & 0 & r^2 \sin^2 \theta P_t \end{bmatrix} \quad (10)$$

2.5 Effective Variables

We are going to define the effective variables in a similar way to the case of [71, 67]. However, here we will denote it as ρ_1 and P_1 , without a bar at the top:

```
> grmap(TB(dn, dn), subs, rho[0] = (rho[1]+omega*P[1])/(1-omega), 'x');
> grmap(TB(dn, dn), subs, P[0] = (P[1]+omega*rho[1])/(1-omega), 'x');
```

Changing the term a bit for radiation

```
> grmap(TB(dn, dn), subs, xi = (1-omega)*epsilon/(1+omega), 'x');
> gralter(TB(dn, dn), expand, factor);
> grdisplay(TB(dn, dn));
```

For the bondi spacetime :

$TB(dn, dn)$

$TB(dn, dn)$

$$TB_{ab} = \begin{bmatrix} \frac{e^{2\beta(u,r)} V(u,r) (\omega^2 \varepsilon + \omega^2 \rho_1 + \omega P_1 - 2\varepsilon \omega - \omega \rho_1 + \varepsilon + \rho_1)}{r(-1+\omega)^2} & \rho_1 e^{2\beta(u,r)} & 0 & 0 \\ \rho_1 e^{2\beta(u,r)} & \frac{r e^{2\beta(u,r)} (\rho_1 + P_1)}{V(u,r)} & 0 & 0 \\ 0 & 0 & r^2 P_t & 0 \\ 0 & 0 & 0 & r^2 \sin^2 \theta P_t \end{bmatrix} \quad (11)$$

Using the routines defined in [70] we can obtain an alternate expression for TB_{uu} :

```
> TBdndnuu := kfactor(hcollect(grcomponent(TB(dn, dn), [u, u]), {P[1], rho[1], V(u, r)},
{r, omega, beta(u, r)}), V(u, r)*exp(2*beta(u, r))/r);
```

$$TBdndnuu = \frac{V(u, r)}{r} e^{2\beta(u, r)} \left(\frac{\omega P_1}{(-1 + \omega)^2} + \frac{(-\omega + \omega^2 + 1) \rho_1}{(-1 + \omega)^2} + \varepsilon \right) \quad (12)$$

or equivalently

$$TBdndnuu = \frac{V(u, r)}{r} e^{2\beta(u, r)} \left[\frac{\omega (\rho_1 + P_1)}{(1 - \omega)^2} + \rho_1 + \varepsilon \right]$$

2.6 Electromagnetic Component

If we are interested in the possibility that the material subject to the study presents an electric charge, it is necessary to include it in the tensor and therefore establish the expression for the Faraday tensor

$$T_{\alpha\beta} = \frac{1}{4\pi} \left[F_{\alpha\mu} F_{\beta}^{\mu} + \frac{1}{4} g_{\alpha\beta} (F_{\mu\nu} F^{\mu\nu}) \right] \quad (13)$$

where F^{ab} satisfies the equations

$$(\sqrt{g} F^{\alpha\beta})_{,\beta} = 4\pi \sqrt{g} J^{\alpha} \quad F_{[\alpha\beta;\gamma]} = 0$$

Assuming spherical symmetry, the only non-zero component is F and we have F^{01}

$$\begin{aligned} (e^{2\beta} r^2 F^{01})_{,1} &= 4\pi e^{2\beta} r^2 J^0 \\ (e^{2\beta} r^2 F^{10})_{,0} &= 4\pi e^{2\beta} r^2 J^1 \end{aligned}$$

and when integrating the first:

$$(e^{2\beta} r^2 F^{01}) = \int_0^r (4\pi e^{2\beta} r^2 J^0) dr \equiv Q(u, r)$$

Because of this, we can write

$$F^{01} = \frac{e^{-2\beta}}{r^2} \cdot Q(u, r)$$

where $Q(u, r)$, plays the role of electric charge and the electromagnetic component of the stress tensor is

```
> grdef('F{^a ^b}:= F01*(kdelta{$u ^a}*kdelta{$r ^b} -kdelta{$r ^a}*kdelta{$u ^b}))');
> grdef('Faraday2:=F{^a ^b}*F{a b} ');
> grdef('Tem{a b}:= (F{a c}* F{^c b}+(1/4)*g{a b}*F{c d}*F{^c ^d })/(4*Pi) ');
> grcalcd(Faraday2);
> grcalcd(F(up, up));
```

CPU Time = 0.

For the bondi spacetime :

$F(up, up)$

$F(up, up)$

$$F^{ab} = \begin{bmatrix} 0 & F01 & 0 & 0 \\ -F01 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (14)$$

In terms of F^{01} , the electromagnetic component of the stress tensor is

```
> grcalcd(Tem(dn, dn));
Calculated g(up,up) for bondi (0.004000 sec.)
Calculated F(up,dn) for bondi (0.004000 sec.)
Calculated Tem(dn,dn) for bondi (0.000000 sec.)
```

CPU Time = 0.011

For the bondi spacetime :

Tem(dn, dn)

Tem(dn, dn)

$$Tem_{ab} = \begin{bmatrix} \frac{1}{8} \frac{V(u,r) (e^{2\beta(u,r)})^3 F01^2}{\pi r} & \frac{1}{8} \frac{(e^{2\beta(u,r)})^3 F01^2}{\pi} & 0 & 0 \\ \frac{1}{8} \frac{(e^{2\beta(u,r)})^3 F01^2}{\pi} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{8} \frac{r^2 (e^{2\beta(u,r)})^3 F01^2}{\pi} & 0 \\ 0 & 0 & 0 & \frac{1}{8} \frac{r^2 \sin(\theta)^2 (e^{2\beta(u,r)})^3 F01^2}{\pi} \end{bmatrix}, \quad (15)$$

The electromagnetic component of the stress tensor is expressed in terms of the electric charge as:

```
> grmap(Tem(dn, dn), subs, F01 = exp(-2*beta(u, r))*Q(u, r)/r^2, 'x');
> grmap(Faraday2, subs, F01 = exp(-2*beta(u, r))*Q(u, r)/r^2, 'x');
> gralter(Tem(dn, dn), expand, factor);
Component simplification of a GRTensorIII object:
Applying routine expand to object Tem(dn,dn)
Applying routine factor to object Tem(dn,dn)
```

CPUTime = 0.007

```
> grdisplay(Tem(dn, dn));
```

For the bondi spacetime :

$Tem(dn, dn)$

$Tem(dn, dn)$

$$Tem_{ab} = \begin{bmatrix} \frac{1}{8} \frac{V(u,r)(e^{2\beta(u,r)})^2 Q^2(u,r)}{\pi r^5} & \frac{1}{8} \frac{(e^{2\beta(u,r)})^2 Q^2(u,r)}{\pi r^4} & 0 & 0 \\ \frac{1}{8} \frac{(e^{2\beta(u,r)})^2 Q^2(u,r)}{\pi r^4} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{8} \frac{Q^2(u,r)}{\pi r^2} & 0 \\ 0 & 0 & 0 & \frac{1}{8} \frac{\sin(\theta)^2 Q^2(u,r)}{\pi r^2} \end{bmatrix}. \quad (16)$$

2.7 Stress tensor

The stress tensor total that includes matter + radiation + electric charge is:

```
> grdef('T{a b} := TB{a b}+Tem{a b}');
```

If we want to study neutral or uncharged cases, we can turn OFF the contribution of the electric charge

```
> #grmap(T(dn, dn), subs, Q(u, r) = 0, 'x')
```

Calculating the covariant components

```
> grcalc(T(dn, dn));
> gralter(T(dn, dn), expand, factor);
> grdisplay(T(dn, dn));
```

For the bondi spacetime :

$T(dn, dn)$

$T(dn, dn)$

$$\begin{aligned}
T_{uu} &= \frac{1}{8} \frac{1}{(-1+\omega)^2 r^5 \pi} \left(V(u, r) \cdot (e^{\beta(u, r)})^2 (8r^4 \pi \varepsilon \omega^2 + 8r^4 \pi \omega^2 \rho_1 + 8r^4 \pi \omega P_1 - 16r^4 \pi \varepsilon \omega \right. \\
&\quad \left. - 8r^4 \pi \omega \rho_1 + 8r^4 \pi \varepsilon + 8r^4 \pi \rho_1 + (Q(u, r))^2 - 2(Q(u, r))^2 \omega + \omega^2 (Q(u, r))^2 \right) \\
T_{ru} &= \frac{1}{8} \frac{(e^{\beta(u, r)})^2 (8r^4 \pi \rho_1 + Q^2(u, r))}{r^4 \pi} \\
T_{ur} &= \frac{1}{8} \frac{(e^{\beta(u, r)})^2 (8r^4 \pi \rho_1 + Q^2(u, r))}{r^4 \pi} \\
T_{rr} &= \frac{r(e^{\beta(u, r)})^2 (P_1 + \rho_1)}{V(u, r)} \\
T_{\theta\theta} &= \frac{1}{8} \frac{8r^4 P_t \pi + Q(u, r)^2}{\pi r^2} \\
T_{\phi\phi} &= \frac{1}{8} \frac{\sin^2 \theta (8r^4 P_t \pi + Q(u, r)^2)}{\pi r^2}
\end{aligned} \tag{17}$$

As an example of the use of simplification routines, we can factor the time covariant component

```

>tempTdndnuu := kfactor(hcollect(grcomponent(T(dn, dn), [u, u]), {P[1], rho[1], V(u, r)},
{r, omega, beta(u, r)}), V(u, r)*exp(2*beta(u, r))/r);

```

$$tempTdndnuu := \frac{V(u, r) e^{2\beta(u, r)} \left(\frac{\omega P_1}{(-1+\omega)^2} + \frac{(-\omega+\omega^2+1)\rho_1}{(-1+\omega)^2} + \frac{1}{8} \left(\frac{8r^4 \pi \varepsilon + Q(u, r)^2}{r^4 \pi} \right) \right)}{r} \tag{18}$$

The contravariant radial component can be simplified with the help of the previously loaded factoring functions

```

> tempTupuprr := kfactor(hcollect(hcollect(grcomponent(T(up, up), [r, r]), {epsilon,
Q(u, r), V(u, r), exp(2*beta(u, r))}, {epsilon, P[1], rho[1], Q(u, r)}), {epsilon,
P[1], rho[1], Q(u, r)}, {V(u, r), beta(u, r)}), exp(-2*beta(u, r))*V(u, r/r)

```

$$tempTupuprr = \frac{V(u, r) e^{-2\beta(u, r)} \left(\varepsilon + \frac{(-\omega+\omega^2+1)P_1}{(-1+\omega)^2} - \frac{1}{8} \frac{Q(u, r)^2}{r^4 \pi} + \frac{\omega \rho_1}{(-1+\omega)^2} \right)}{r} \tag{19}$$

that we can simplify as

$$tempTupuprr = \frac{V(u, r) e^{-2\beta(u, r)}}{r} \left[\frac{\omega(\rho_1 + P_1)}{(1-\omega)^2} + \varepsilon + \left(P_1 - \frac{Q(u, r)^2}{8\pi r^4} \right) \right]$$

2.8 Temporary and radial dependency of the Stress Tensor

To calculate the conservation equations, it is necessary to establish that the density $\rho = \rho(u, r)$, pressure $P = P(u, r)$, radiation $\varepsilon = \varepsilon(u, r)$ and tangential pressure $P_t = P_t(u, r)$ depend on that of the temporal and radial coordinates.

```

> grmap(T(up, dn), subs, rho[1] = rho(u, r), 'x');
> grmap(T(up, dn), subs, P[1] = P(u, r), 'x');
> grmap(T(up, dn), subs, P[t] = P[t](u, r), 'x');
> grmap(T(up, dn), subs, epsilon = epsilon(u, r), 'x');
> grmap(T(up, dn), subs, omega = omega(u, r), 'x');
> gralter(T(up, dn), simplify);
> grdisplay(T(up, dn));

```

For the bondi spacetime :

$T(up, dn)$

$T(up, dn)$

$$\begin{aligned}
T_b^a = & \left[\begin{array}{cccc} \frac{1}{8} \left(\frac{8r^4 \pi \rho(u,r) + Q(u,r)^2}{r^4 \pi} \right), & r \left(\frac{P(u,r) + \rho(u,r)}{V(u,r)} \right), & 0, & 0 \\ \frac{V(u,r)}{r} \frac{(\varepsilon(u,r)\omega(u,r)^2 + \omega(u,r)P(u,r) - 2\varepsilon(u,r)\omega(u,r) + \omega(u,r)\rho(u,r) + \varepsilon(u,r))}{(-1 + \omega(u,r))^2}, & \frac{-Q(u,r)^2 + 8r^4 \pi P(u,r)}{r^4 \pi} & 0, & 0 \\ 0, & 0, & -\frac{1}{8} \left(\frac{8r^4 \pi P_t(u,r) + Q(u,r)^2}{r^4 \pi} \right), & 0 \\ 0, & 0, & 0, & -\frac{1}{8} \left(\frac{8r^4 \pi P_t(u,r) + Q(u,r)^2}{r^4 \pi} \right) \end{array} \right] \quad (20)
\end{aligned}$$

2.9 Conservation equations (TOV)

These equations are important since the Einstein field equations are composed of two structures: One is the geometry of the G_{ab} system and the other is basically the energy T_{ab} . From the expressions that we have taken the geometry is linear in the second derivative and non-linear in the first. However, the impulse energy tensor that we have used is linear. Presumably, this non-linearity, of the geometrical part must have a non-linear version in some expression of the impulse energy tensor. It is therefore necessary to calculate the components of the conservation equation since it is possible that they contain the *non-linear* expression necessary to correct our choice of the structure of the Stress Tensor.

```

> grdef('cero{a } := [0, 0, 0, 0 ]');
> grdef('TOV{b } := T{^a b;a} = cero{b}');

```

Calculating the covariant components of the conservation equation:


```

> grcalc(TOV(dn));
> gralter(TOV(dn), simplify);
> grdisplay(TOV(dn));

```

For the bondi spacetime :

$TOV(dn)$

$TOV(dn)$

$$TOV_u = \left(\frac{1}{4r^4 V(u,r)} \frac{1}{(-1 + \omega(u,r))^3} \left(8 \left(\omega(u,r)^2 \varepsilon(u,r) + (P(u,r) + \rho(u,r) - 2 \cdot \varepsilon(u,r)) \omega(u,r) + \varepsilon(u,r) \right) \cdot \right. \right. \\ \left. \left. V(u,r)^2 r^3 \pi (-1 + \omega(u,r)) + \dots = 0 \right. \right. \quad (21)$$

$$TOV_r = \left(\frac{1}{4r^4 V(u,r)^2} \pi \left(4r^5 \pi V(u,r) \left(\frac{\partial}{\partial u} \rho(u,r) \right) + 4r^5 \pi V(u,r) \left(\frac{\partial}{\partial u} \rho(u,r) \right) - \dots \dots \right. \right. \\ \left. \left. - 4r^5 \pi P(u,r) r^4 \pi \left(\frac{\partial}{\partial u} V(u,r) \right) - 4r^5 \pi \rho(u,r) + \dots = 0 \right. \right. \quad (22)$$

We can use the simplification routines loaded at the beginning of the sheet to compare the terms observed in the conservation equations. The following simplifications allow us to establish that some of the identities are not really independent:

```

>TOVn[0] := hcollect(simplify(exp(2*beta(u, r))*lhs(grcomponent(TOV(up), [u]))),
{P(u, r), rho(u, r), diff(P(u, r), r), diff(Q(u, r), r), P[t](u, r)},
{diff(V(u, r), r), diff(V(u, r), u), diff(beta(u, r), r), diff(beta(u, r), u)});

```

$$TOV_{n_0} = \left(-\frac{1}{2} \frac{\frac{\partial}{\partial r} V(u,r)}{V(u,r)} - \frac{r \left(\frac{\partial}{\partial u} V(u,r) \right)}{V(u,r)^2} - \left(\frac{\partial}{\partial r} \beta(u,r) \right) + \frac{2r \frac{\partial}{\partial u} \beta(u,r)}{V(u,r)} - \frac{3}{2r} \right) P(u,r) + \\ - \frac{\partial}{\partial r} P(u,r) + \frac{1}{4} \frac{Q(u,r) \frac{\partial}{\partial r} Q(u,r)}{\pi r^4} + \left(-\frac{1}{2} \frac{\frac{\partial}{\partial r} V(u,r)}{V(u,r)} - \frac{r \left(\frac{\partial}{\partial u} V(u,r) \right)}{V(u,r)^2} + \right. \\ \left. - \left(\frac{\partial}{\partial r} \beta(u,r) \right) + \frac{2r \frac{\partial}{\partial u} \beta(u,r)}{V(u,r)} + \frac{1}{2r} \right) \rho(u,r) + \frac{2P_t(u,r)}{r} + \frac{r \left(\frac{\partial}{\partial u} P(u,r) + \frac{\partial}{\partial u} \rho(u,r) \right)}{V(u,r)} \quad (23)$$

```

>TOVn[1] := hcollect(hcollect(simplify(r*exp(2*beta(u, r))*lhs(grcomponent(TOV(up),
[r]))/V(u, r)), {V(u, r), beta(u, r), diff(V(u, r), r), diff(V(u, r), u), diff(

```

```

beta(u, r), r), diff(beta(u, r), u)}, {diff(epsilon(u, r), r), diff(epsilon(u, r),
u), diff(omega(u, r), r), diff(omega(u, r), u), epsilon(u, r), omega(u, r))},
{P(u, r), Q(u, r), rho(u, r), diff(P(u, r), r), diff(Q(u, r), r),
diff(Q(u, r), u), diff(epsilon(u, r), r), epsilon(u, r), P[t](u, r)}, {V(u, r),
beta(u, r), diff(V(u, r), r), diff(V(u, r), u), diff(beta(u, r), r), diff(beta
(u, r), u)});

```

$$\begin{aligned}
TOVn_1 := & \left(\frac{((\omega(u, r))^2 + 1) \frac{\partial}{\partial r} \beta(u, r)}{(-1 + \omega(u, r))^2} + \right. \\
& + \frac{1}{2} \frac{-2r \frac{\partial}{\partial r} \omega(u, r) + 3(\omega(u, r))^3 - 3 + 7\omega(u, r) - 7(\omega(u, r))^2 - \dots}{r(-1 + \omega(u, r))^3} \\
& - \frac{1}{2} \frac{(\frac{\partial}{\partial r} V(u, r)) (\omega(u, r))^2 + 2r (\frac{\partial}{\partial u} \beta(u, r)) (\omega(u, r))^2 - \dots}{V(u, r) (-1 + \omega(u, r))^2} + \\
& \left. \frac{-\frac{\partial}{\partial r} V(u, r) + 2r \frac{\partial}{\partial u} \beta(u, r)}{V(u, r) (-1 + \omega(u, r))^2} + \frac{1}{2} \frac{r \frac{\partial}{\partial u} V(u, r)}{(V(u, r))^2} \right) P(u, r) + \\
& + \left(-\frac{1}{4} \frac{\frac{\partial}{\partial r} Q(u, r)}{\pi r^4} + \frac{1}{4} \frac{\frac{\partial}{\partial u} Q(u, r)}{V(u, r) \pi r^3} \right) Q(u, r) \dots
\end{aligned} \tag{24}$$

```

>TOVn[2] := hcollect(simplify(r*lhs(grcomponent(TOV(dn), [u]))/V(u, r)), {P(u, r), rho(u, r),
diff(P(u, r), r), diff(Q(u, r), u), diff(V(u, r), r), diff(V(u, r), u), diff(rho(u, r), r),
diff(rho(u, r), u), diff(beta(u, r), r), diff(beta(u, r), u), diff(epsilon(u, r), r),
diff(omega(u, r), r), epsilon(u, r), omega(u, r)}, {P(u, r), Q(u, r), epsilon(u, r),
omega(u, r)});

```

$$\begin{aligned}
TOV_{n_2} = & \left(\frac{\left(\frac{\partial}{\partial r} V(u, r) \right) \omega(u, r)}{V(u, r) (-1 + \omega(u, r))^2} - \frac{1}{2} \frac{r \frac{\partial}{\partial u} V(u, r)}{(V(u, r))^2} + 2 \frac{\left(\frac{\partial}{\partial r} \beta(u, r) \right) \omega(u, r)}{(-1 + \omega(u, r))^2} + \frac{r \frac{\partial}{\partial u} \beta(u, r)}{V(u, r)} \right. \\
& - \frac{(1 + \omega(u, r)) \frac{\partial}{\partial r} \omega(u, r)}{(-1 + \omega(u, r))^3} + \frac{\omega(u, r)}{r (-1 + \omega(u, r))^2} \Big) P(u, r) + \left(\frac{\left(\frac{\partial}{\partial r} V(u, r) \right) \omega(u, r)}{V(u, r) (-1 + \omega(u, r))^2} - \frac{1}{2} \frac{r \frac{\partial}{\partial u} V(u, r)}{(V(u, r))^2} \right. \\
& + 2 \frac{\left(\frac{\partial}{\partial r} \beta(u, r) \right) \omega(u, r)}{(-1 + \omega(u, r))^2} + \frac{r \frac{\partial}{\partial u} \beta(u, r)}{V(u, r)} - \frac{(1 + \omega(u, r)) \frac{\partial}{\partial r} \omega(u, r)}{(-1 + \omega(u, r))^3} + \frac{\omega(u, r)}{r (-1 + \omega(u, r))^2} \Big) \rho(u, r) + \\
& + \frac{\left(\frac{\partial}{\partial r} P(u, r) \right) \omega(u, r)}{(-1 + \omega(u, r))^2} + \frac{1}{2} \frac{Q(u, r) \frac{\partial}{\partial u} Q(u, r)}{r^3 \pi V(u, r)} + \frac{\left(\frac{\partial}{\partial r} V(u, r) \right) \epsilon(u, r)}{V(u, r)} + \frac{\omega(u, r) \frac{\partial}{\partial r} \rho(u, r)}{(-1 + \omega(u, r))^2} + \frac{r \frac{\partial}{\partial u} \rho(u, r)}{V(u, r)} \\
& + 2 \left(\frac{\partial}{\partial r} \beta(u, r) \right) + \epsilon(u, r) + \frac{\partial}{\partial r} \epsilon(u, r)
\end{aligned} \tag{25}$$

```

>TOVn[3] := hcollect(simplify(lhs(grcomponent(TOV(dn), [r]))), {P(u, r), rho(u, r),
diff(P(u, r), r), diff(Q(u, r), r), P[t](u, r)}, {diff(V(u, r), r), diff(V(u, r), u),
diff(beta(u, r), r), diff(beta(u, r), u)});

```

$$\begin{aligned}
TOV_{n_3} = & \left(-\frac{1}{2} \frac{\frac{\partial}{\partial r} V(u, r)}{V(u, r)} - \frac{r \frac{\partial}{\partial u} V(u, r)}{(V(u, r))^2} - \frac{\partial}{\partial r} \beta(u, r) + 2 \frac{r \frac{\partial}{\partial u} \beta(u, r)}{V(u, r)} - \frac{3}{2r} \right) P(u, r) + \left(-\frac{1}{2} \frac{\frac{\partial}{\partial r} V(u, r)}{V(u, r)} \right. \\
& - \frac{r \frac{\partial}{\partial u} V(u, r)}{(V(u, r))^2} - \frac{\partial}{\partial r} \beta(u, r) + 2 \frac{r \frac{\partial}{\partial u} \beta(u, r)}{V(u, r)} + \frac{1}{2r} \Big) \rho(u, r) - \frac{\partial}{\partial r} P(u, r) + \frac{1}{4} \frac{Q(u, r) \frac{\partial}{\partial r} Q(u, r)}{\pi r^4} \\
& + \frac{2P_t(u, r)}{r} + \frac{r \left(\frac{\partial}{\partial u} P(u, r) + \frac{\partial}{\partial u} \rho(u, r) \right)}{V(u, r)}
\end{aligned} \tag{26}$$

By comparing term by term, a certain regularity can be observed, and we can verify that the two equations are the same:

```

> temp03 := collect(simplify(TOVn[0]-TOVn[3]), r);

```

$$temp03 := 0 \tag{27}$$

With the previous expression we can conclude that

$$e^{2\beta} T^a_{;a} = T^a_{1;a} = 0$$

and the same conservation equation is obtained. We have in this way 3 independent conservation equations.

```

> temp21 := hcollect(hcollect(hcollect(simplify(TOVn[2]-TOVn[1]), {V(u, r), beta(u, r)},
{diff(P(u, r), u), diff(V(u, r), r)}), {P(u, r), rho(u, r), diff(P(u, r), r), diff(Q(u, r), u)},
{diff(rho(u, r), u), diff(beta(u, r), r), diff(beta(u, r), u)}), {P(u, r), rho(u, r),
P[t](u, r)}, {Q(u, r), diff(Q(u, r), r)});

```

$$\begin{aligned}
temp21 := & \left(-\frac{1}{2} \frac{\frac{\partial}{\partial r} V(u, r)}{V(u, r)} - \frac{r \frac{\partial}{\partial u} V(u, r)}{(V(u, r))^2} - \frac{\partial}{\partial r} \beta(u, r) + 2 \frac{r \frac{\partial}{\partial u} \beta(u, r)}{V(u, r)} - \frac{3}{2r} \right) P(u, r) + \left(-\frac{1}{2} \frac{\frac{\partial}{\partial r} V(u, r)}{V(u, r)} \right. \\
& - \frac{r \frac{\partial}{\partial u} V(u, r)}{(V(u, r))^2} - \frac{\partial}{\partial r} \beta(u, r) + 2 \frac{r \frac{\partial}{\partial u} \beta(u, r)}{V(u, r)} + \frac{1}{2r} \Big) \rho(u, r) - \frac{\partial}{\partial r} P(u, r) + \frac{1}{4} \frac{Q(u, r) \frac{\partial}{\partial r} Q(u, r)}{\pi r^4} \\
& + \frac{2P_t(u, r)}{r} + \frac{r \left(\frac{\partial}{\partial u} P(u, r) + \frac{\partial}{\partial u} \rho(u, r) \right)}{V(u, r)}
\end{aligned} \tag{28}$$

```

> temp21m0 := collect(simplify(temp21-TOVn[0]), r);

```

$$temp21m0 := 0 \tag{29}$$

The difference of the two equations brings us back to the conservation equation of (TOV: Tolman Oppenheimer Volkoff). However, taken separately they are not proporcionales to $TOV[0] = TOV[3]$.

2.10 Einstein field equations

Taking as input the covariant components of the Einstein field equations:

```

>grdef('Eins{a b} := G{a b} = 8*Pi*T{a b}');
>grcalc(Eins(dn, dn));
> gralter(Eins(dn, dn), simplify, factor, radsimp);
> grdisplay(Eins(dn, dn));

```

For the bondi spacetime :

Eins(dn, dn)

Eins(dn, dn)

$$\begin{aligned}
Eins_{uu} &= \left(\frac{r \frac{\partial}{\partial u} V(u, r) + 2 (V(u, r))^2 \frac{\partial}{\partial r} \beta(u, r) - 2 r V(u, r) \frac{\partial}{\partial u} \beta(u, r) - V(u, r) \frac{\partial}{\partial r} V(u, r) + V(u, r) e^{2\beta(u, r)}}{r^3} \right) = \\
&= \frac{1}{(-1 + \omega(u, r))^2 r^5} \left(V(u, r) e^{2\beta(u, r)} \left(8 r^4 \pi \epsilon(u, r) (\omega(u, r))^2 + 8 r^4 \pi (\omega(u, r))^2 \rho(u, r) + \right. \right. \\
&\quad + 8 r^4 \pi \omega(u, r) P(u, r) - 16 r^4 \pi \epsilon(u, r) \omega(u, r) - 8 r^4 \pi \omega(u, r) \rho(u, r) + 8 r^4 \pi \epsilon(u, r) + \\
&\quad \left. \left. + 8 r^4 \pi \rho(u, r) + (Q(u, r))^2 - 2 (Q(u, r))^2 \omega(u, r) + (Q(u, r))^2 (\omega(u, r))^2 \right) \right) \\
Eins_{ru} &= \left(\frac{2 V(u, r) \frac{\partial}{\partial r} \beta(u, r) - \frac{\partial}{\partial r} V(u, r) + e^{2\beta(u, r)}}{r^2} = \frac{e^{2\beta(u, r)} \left(8 r^4 \pi \rho(u, r) + (Q(u, r))^2 \right)}{r^4} \right) \\
Eins_{ur} &= \left(\frac{2 V(u, r) \frac{\partial}{\partial r} \beta(u, r) - \frac{\partial}{\partial r} V(u, r) + e^{2\beta(u, r)}}{r^2} = \frac{e^{2\beta(u, r)} \left(8 r^4 \pi \rho(u, r) + (Q(u, r))^2 \right)}{r^4} \right) \\
Eins_{rr} &= \left(4 \frac{\frac{\partial}{\partial r} \beta(u, r)}{r} = 8 \frac{\pi e^{2\beta(u, r)} (P(u, r) + \rho(u, r))}{V(u, r)} \right) \\
Eins_{\theta\theta} &= \left(\frac{1}{2} \left(-4 \left(\frac{\partial^2}{\partial u \partial r} \beta(u, r) \right) r^2 + \left(\frac{\partial^2}{\partial r^2} V(u, r) \right) r + 2 \left(\frac{\partial}{\partial r} V(u, r) \right) \left(\frac{\partial}{\partial r} \beta(u, r) \right) r + \right. \right. \\
&\quad \left. \left. + 2 V(u, r) \left(\frac{\partial^2}{\partial r^2} \beta(u, r) \right) r - 2 V(u, r) \frac{\partial}{\partial r} \beta(u, r) \right) e^{-2\beta(u, r)} = \frac{8 r^4 P_t(u, r) \pi + (Q(u, r))^2}{r^2} \right) \\
Eins_{\phi\phi} &= \left(\frac{\sin(\theta)^2}{2} \left(-4 \left(\frac{\partial^2}{\partial u \partial r} \beta(u, r) \right) r^2 + \left(\frac{\partial^2}{\partial r^2} V(u, r) \right) r + 2 \left(\frac{\partial}{\partial r} V(u, r) \right) \left(\frac{\partial}{\partial r} \beta(u, r) \right) r + \right. \right. \\
&\quad \left. \left. + 2 V(u, r) \left(\frac{\partial^2}{\partial r^2} \beta(u, r) \right) r - 2 V(u, r) \frac{\partial}{\partial r} \beta(u, r) \right) e^{-2\beta(u, r)} = \right. \\
&\quad \left. \frac{\sin(\theta)^2 \left(8 r^4 P_t(u, r) \pi + (Q(u, r))^2 \right)}{r^2} \right) \tag{30}
\end{aligned}$$

Using the simplification routines for the uu component of the Einstein tensor : $G_{00} = 8\pi T_{00}$

```

>lhsEuu := hcollect(lhs(grcomponent(Eins(dn, dn), [u, u])), {r, diff(V(u, r), u),
diff(beta(u, r), u)}, {V(u, r), diff(V(u, r), r), diff(beta(u, r), r)})

```

$$\begin{aligned}
lhsEuu &= \frac{-2 V(u, r) \frac{\partial}{\partial u} \beta(u, r) + \frac{\partial}{\partial u} V(u, r)}{r^2} + \\
&\quad + \frac{V(u, r) e^{2\beta(u, r)} + 2 (V(u, r))^2 \frac{\partial}{\partial r} \beta(u, r) - V(u, r) \frac{\partial}{\partial r} V(u, r)}{r^3} \tag{31}
\end{aligned}$$

And for the right side of the same component of the Einstein field equation:

```

> rhsEuu := kfactor(hcollect(hcollect(rhs(grcomponent(Eins(dn, dn), [u, u])),

```

```
{Q(u, r), V(u, r), exp(2*beta(u, r)), epsilon(u, r)}, {P(u, r), Q(u, r), rho(u, r),
epsilon(u, r))}, {P(u, r), Q(u, r), rho(u, r), epsilon(u, r)}, {V(u, r), beta(u, r)}},
8*Pi*exp(2*beta(u, r))*V(u, r)/r);
```

$$rhsEu_{uu} := \frac{8\pi e^{2\beta(u,r)} V(u,r) \left(\frac{\omega(u,r)P(u,r)}{(-1+\omega(u,r))^2} + \frac{1}{8} \frac{(Q(u,r))^2}{r^4\pi} + \frac{((\omega(u,r))^2 - \omega(u,r) + 1)\rho(u,r)}{(-1+\omega(u,r))^2} + \epsilon(u,r) \right)}{r} \quad (32)$$

Let's see now its Mixed components:

```
> grcalc(Eins(up, dn));
> gralter(Eins(up, dn), expand, factor);
> grdisplay(Eins(up, dn));
```

For the bondi spacetime :

Eins(up, dn)

Eins(up, dn)

$$\begin{aligned}
Eins_u^u &= \left(\frac{2V(u,r) \frac{\partial}{\partial r} \beta(u,r) - \frac{\partial}{\partial r} V(u,r) + (e^{\beta(u,r)})^2}{(e^{\beta(u,r)})^2 r^2} = \frac{8r^4 \pi \rho(u,r) + (Q(u,r))^2}{r^4} \right) \\
Eins_r^u &= \left(4 \frac{\frac{\partial}{\partial r} \beta(u,r)}{(e^{\beta(u,r)})^2 r} = 8 \frac{\pi r (P(u,r) + \rho(u,r))}{V(u,r)} \right) \\
Eins_u^r &= \left(-\frac{2V(u,r) \frac{\partial}{\partial u} \beta(u,r) - \frac{\partial}{\partial u} V(u,r)}{(e^{\beta(u,r)})^2 r^2} = \right. \\
&\quad \left. 8 \frac{\pi V(u,r) (\omega(u,r) P(u,r) + \epsilon(u,r) (\omega(u,r))^2 - 2\epsilon(u,r) \omega(u,r) + \omega(u,r) \rho(u,r) + \epsilon(u,r))}{r(-1 + \omega(u,r))^2} \right) \\
Eins_\theta^\theta &= \left(\frac{1}{2} \frac{1}{(e^{\beta(u,r)})^2 r^2} \left(4 \left(\frac{\partial^2}{\partial u \partial r} \beta(u,r) \right) r^2 - \left(\frac{\partial^2}{\partial r^2} V(u,r) \right) r - 2 \left(\frac{\partial}{\partial r} V(u,r) \right) \left(\frac{\partial}{\partial r} \beta(u,r) \right) r + \right. \right. \\
&\quad \left. \left. - 2V(u,r) \left(\frac{\partial^2}{\partial r^2} \beta(u,r) \right) r + 2V(u,r) \left(\frac{\partial}{\partial r} \beta(u,r) \right) \right) = -\frac{8r^4 P_t(u,r) \pi + (Q(u,r))^2}{r^4} \right) \\
Eins_\phi^\phi &= \left(\frac{1}{2} \frac{1}{(e^{\beta(u,r)})^2 r^2} \left(4 \left(\frac{\partial^2}{\partial u \partial r} \beta(u,r) \right) r^2 - \left(\frac{\partial^2}{\partial r^2} V(u,r) \right) r - 2 \left(\frac{\partial}{\partial r} V(u,r) \right) \left(\frac{\partial}{\partial r} \beta(u,r) \right) r + \right. \right. \\
&\quad \left. \left. - 2V(u,r) \left(\frac{\partial^2}{\partial r^2} \beta(u,r) \right) r + 2V(u,r) \left(\frac{\partial}{\partial r} \beta(u,r) \right) \right) = -\frac{8r^4 P_t(u,r) \pi + (Q(u,r))^2}{r^4} \right)
\end{aligned}$$

Grouping terms for the right side of the EqnEins $Eins(up, dn)[r, u]$ component has

```

> rhsEru := kfactor(hcollect(rhs(grcomponent(Eins(up, dn), [r, u])), {P(u, r),
epsilon(u, r), omega(u, r)}, {V(u, r)}), 8*V(u, r)*Pi/r);

```

$$rhsEru = \frac{8\pi V(u,r)}{r} \left(\frac{\omega(u,r) P(u,r)}{(-1 + \omega(u,r))^2} + \epsilon(u,r) + \frac{\omega(u,r) \rho(u,r)}{(-1 + \omega(u,r))^2} \right) \quad (33)$$

An interesting expression is the one associated with the field equation ($G_{11} = 8\pi T_{11}$), which includes the pressure and is very similar to the one obtained previously ($G_1^1 = 8\pi T_1^1$), which relates density to the functions of the metric

```

> lhsErr := hcollect(lhs(grcomponent(Eins(up, dn), [r, r])), {P(u, r),
epsilon(u, r), omega(u, r)}, {V(u, r)});

```

$$lhsErr := \frac{-2V(u,r) \frac{\partial}{\partial r} \beta(u,r) - \frac{\partial}{\partial r} V(u,r) + (e^{\beta(u,r)})^2}{(e^{\beta(u,r)})^2 r^2} \quad (34)$$

```
> rhsErr := hcollect(rhs(grcomponent(Eins(up, dn), [r, r])), {P(u, r)}, {V(u, r)});
```

$$rhsErr := -8\pi P(u, r) + \frac{(Q(u, r))^2}{r^4} \quad (35)$$

Both relate the metrical elements -- $V(u, r)$ and $\beta(u, r)$ -- with the physical quantities -- $P(u, r)$ and $\rho(u, r)$ -- of pressure and density.

2.11 Trace

Let's calculate the trace of G

```
> grdef('TG := g{^a ^b}*G{a b} ');
> grcalcd(TG);
```

'CPU Time' = 0.005

For the bondi spacetime :

TG

$$\begin{aligned} TG := & \frac{1}{e^{2\beta(u,r)}r^2} \left(2V(u,r) \frac{\partial}{\partial r} \beta(u,r) - 2 \frac{\partial}{\partial r} V(u,r) + 2e^{2\beta(u,r)} + 4 \left(\frac{\partial^2}{\partial u \partial r} \beta(u,r) \right) r^2 \right. \\ & \left. - \left(\frac{\partial^2}{\partial r^2} V(u,r) \right) r - 2 \left(\frac{\partial}{\partial r} V(u,r) \right) \left(\frac{\partial}{\partial r} \beta(u,r) \right) r - 2V(u,r) \left(\frac{\partial^2}{\partial r^2} \beta(u,r) \right) r \right) \end{aligned} \quad (36)$$

Very similar to the equation the left side of the angular component, of the field equations $G_2^2 = 8\pi T_2^2$. Now the trace of the Stress tensor:

```
> grdef('TT := g{^a ^b}*T{a b} ');
> grcalcd(TT);
```

'CPU Time' = 0.004

For the bondi spacetime :

TT

$$TT = \rho(u, r) - P(u, r) - 2P_t(u, r) \quad (37)$$

The contraction of the Field equation $Traza = g^{\mu\nu} G_{\mu\nu} = 8\pi g^{\mu\nu} T_{\mu\nu}$


```
> grdef('TrazaTE := g{^a ^b}*G{a b} = 8*Pi* g{^a ^b}*T{a b}');
> grcalcd(TrazaTE);
```

$$\begin{aligned}
TrazaTE &= \frac{1}{e^{2\beta(u,r)}r^2} \left(2V(u,r) \frac{\partial}{\partial r} \beta(u,r) - 2 \frac{\partial}{\partial r} V(u,r) + 2e^{2\beta(u,r)} + 4 \left(\frac{\partial^2}{\partial u \partial r} \beta(u,r) \right) r^2 \right. \\
&\quad \left. - \left(\frac{\partial^2}{\partial r^2} V(u,r) \right) r - 2 \left(\frac{\partial}{\partial r} V(u,r) \right) \left(\frac{\partial}{\partial r} \beta(u,r) \right) r - 2V(u,r) \left(\frac{\partial^2}{\partial r^2} \beta(u,r) \right) r \right) \\
&= -8\pi (-\rho(u,r) + P(u,r) + 2P_t(u,r))
\end{aligned} \tag{38}$$

This equation of the trace is important since it can be shown that it is equivalent to the equation ($T^{\mu 0}_{;\mu} = 0$), the Tolman-Oppenheimer-Volkoff equation. When comparing the left side of the trace equation with the left side of the angular component $G^2_2 = 8\pi T^2_2$, of the equations from field, we get

```
> difG22TG := simplify((1/2)*grcomponent(TG)-lhs(grcomponent(Eins(up, dn), [theta, theta])))
```

$$difG22TG := \frac{e^{-2\beta(u,r)} \left(-\frac{\partial}{\partial r} V(u,r) + e^{2\beta(u,r)} \right)}{r^2} \tag{39}$$

3 Results and Comments

This section is dedicated to presenting some results obtained when executing the established and described commands (*MAPLE17 + GRTensorIII*) of the previous section. We can summarize that the spreadsheet allows us to calculate the different tensor terms for a radiative coordinate system from the Einstein Field equations and their Conservation equations. Throughout the spreadsheet development process, the importance of the expression to determine the components of the Stress tensor from the Minkowski metric to its connection with the Bondi radiative coordinate metric of the subsection (2.4) should be taken into account (9). You can see that it is necessary to specify the scope of each component of the metric (9), to make the calculation of the entire sheet flow without interruption ([74]). This procedure has been used to perform the calculations for ([67, 68]). We prove that this calculation process fully coincides with the result obtained manually. In these articles the equations allow us to carry out the study and behavior of the physical variables calculated in the articles cited above. For example, the Conservation equations ($T^{\mu}_{\nu;\mu} = 0$) for the charged case are:

$$\begin{aligned}
T_{0;\lambda}^\lambda &= \frac{V}{r^2} \left(1 + \frac{rV_{,1}}{V} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \\
&\quad + \frac{V}{r} \frac{\partial}{\partial r} \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \frac{\partial}{\partial u} \left(\rho + \frac{Q^2}{8\pi r^4} \right) = 0
\end{aligned} \tag{40}$$

$$\begin{aligned}
T_{1;\lambda}^\lambda &= \frac{e^{-2\beta}}{2\pi r} \beta_{,10} - \frac{\partial}{\partial r} \left(P - \frac{Q^2}{8\pi r^4} \right) - \frac{1}{2} \left(2\beta_{,1} + \frac{V_{,1}}{V} - \frac{1}{r} \right) (\rho + P) + \\
&\quad - \frac{2}{r} \left[\left(P - \frac{Q^2}{8\pi r^4} \right) - \left(P_t + \frac{Q^2}{8\pi r^4} \right) \right] = 0
\end{aligned} \tag{41}$$

$$T_{;\lambda}^{\lambda 0} = e^{-2\beta} T_{1;\lambda}^\lambda = 0 \tag{42}$$

$$\begin{aligned}
e^{2\beta} T_{;\lambda}^{\lambda 1} &= \frac{V}{r^2} \left(1 + 4r\beta_{,1} + \frac{rV_{,1}}{V} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \frac{V}{2r} \left(2\beta_{,1} + \frac{V_{,1}}{V} - \frac{1}{r} \right) (\rho + P) + \\
&\quad + \frac{2V}{r^2} \left[\left(P - \frac{Q^2}{8\pi r^4} \right) - \left(P_t + \frac{Q^2}{8\pi r^4} \right) \right] + \\
&\quad + \frac{V}{r} \frac{\partial}{\partial r} \left(P - \frac{Q^2}{8\pi r^4} \right) + \frac{V}{r} \frac{\partial}{\partial r} \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] - \frac{\partial}{\partial u} \left(P - \frac{Q^2}{8\pi r^4} \right) = 0
\end{aligned} \tag{43}$$

As can be seen in (2.9) and the concordance of these results with those obtained manually, it can allow us to trust that this calculation to be carried out in another coordinate system should have a similar performance. Of course, it is necessary to point out that it is still pending to use the complete equation (31) shown in the article ([68]). This comparative study between the results obtained so far and solving the complete equation with the terms $\frac{\partial \varepsilon}{\partial r}$ and $\frac{\partial \omega}{\partial r}$ will be carried out in a future work to publish.

4 Conclusions

In this article we describe a series of computer procedures used in GR relying on the facilities of an integrated platform such as the *Maple* package and *GRTensorIII*. These procedures of computational algebra, numerical and graphic computation can facilitate algebraic calculation both for research purposes and for teaching GR at different levels.

References

- [1] J. F. Nolan. Analytical differentiation on a digital computer. mathesis, Massachussettes Institute of Tecnology, may 1953.
- [2] H. G. Kahrimanian. Analytical differentiation by a digital computer. Master's thesis, Temple University, May 1953.

- [3] John McCarthy. Recursive functions symbolic expressions and their computation by machine, part i. *Communications of the ACM*, 3, 1960. doi: 10.1145/367177.367199. URL <http://gen.lib.rus.ec/scimag/index.php?s=10.1145/367177.367199>.
- [4] E. Bond, M. Auslander, S. Grisoff, R. Kenney, M. Myszewski, J. Sammet, R. Tobey, and S. Zilles. Formac an experimental formula manipulation compiler. In *Proceedings of the 1964 19th ACM National Conference*, ACM '64, pages 112.101–112.1019, New York, NY, USA, 1964. ACM. doi: 10.1145/800257.808916. URL <http://doi.acm.org/10.1145/800257.808916>.
- [5] Hawkes P.W. and Kasper E. Chapter 34 - numerical calculation of trajectories, paraxial properties and aberrations. In Hawkes P.W. and Kasper E., editors, *Principles of Electron Optics (Second Edition)*, volume 1 of -, pages 585–626. Elsevier, second edition edition, oct 2017. ISBN 978-0-08-102256-6. doi: <https://doi.org/10.1016/B978-0-08-102256-6.00034-1>. URL <https://www.sciencedirect.com/science/article/pii/B9780081022566000341>.
- [6] J. Laskar and M. Gastineau. Trip: General computer algebra system for celestial mechanics, oct 2012.
- [7] A. C. Hearn. Computation of algebraic properties of elementary particle reactions using a digital computer. *Communications of the ACM*, 9(8):573–577, aug 1966. ISSN 0001-0782. doi: 10.1145/365758.365766. URL <http://doi.acm.org/10.1145/365758.365766>.
- [8] Anthony C. Hearn. Applications of symbol manipulation in theoretical physics. *Commun. ACM*, 14(8):511–516, August 1971. ISSN 0001-0782. doi: 10.1145/362637.362641. URL <http://doi.acm.org/10.1145/362637.362641>.
- [9] W.S. Brown and A.C. Hearn. Applications of symbolic algebraic computation. *Computer Physics Communications*, 17(1):207 – 215, 1979. ISSN 0010-4655. doi: [https://doi.org/10.1016/0010-4655\(79\)90082-1](https://doi.org/10.1016/0010-4655(79)90082-1). URL <http://www.sciencedirect.com/science/article/pii/0010465579900821>.
- [10] S. Laporta and E. Remiddi. The analytic value of (g_e-2) at three loops in QED. *Nuclear Physics B Proceedings Supplements*, Vol.51, 51:142–147, December 1996. doi: 10.1016/S0920-5632(96)90018-3.
- [11] A.G. Grozin. *Using REDUCE in High Energy Physics*. Cambridge University Press, 2005, sep 2005.
- [12] Friedrich Jegerlehner (eds.) Friedrich Jegerlehner (auth.). *The Anomalous Magnetic Moment of the Muon*, volume 274 of *Springer Tracts in Modern Physics* 274. Springer-Verlag Berlin Heidelberg, 2 edition, 2017. ISBN 9783319635750 9783319635774.

- [13] J. P. Fitch and H. I. Cohen. Using camal for algebraic computations in general relativity. *General Relativity and Gravitation*, 11(6):411–418, dec 1979. doi: 10.1007/BF00759304.
- [14] S. J. Campbell and J. Wainwright. Algebraic computing and the newman-penrose formalism in general relativity. *General Relativity and Gravitation*, 8(12):987–1001, dec 1977. doi: 10.1007/BF00759742.
- [15] J. Wainwright. Algebraic computing in general relativity. In *General Relativity and Gravitation 1977*, page 19, aug 1977.
- [16] Andrzej Krasinski and Marek Perkowski. Ortocartan a new computer program for analytic calculations in general relativity. *General Relativity and Gravitation*, 13(1):67–77, jan 1981. doi: 10.1007/BF00766299.
- [17] John G. Fletcher, Reg Clemens, Richard Matzner, Kip S. Thorne, and Barbara A. Zimmerman. Computer programs for calculating general-relativistic curvature tensors. *Astrophysical Journal, Letters*, 148:L91, May 1967. doi: 10.1086/180022.
- [18] R. A. d’Inverno. Alam - atlas lisp algebraic manipulator. *The Computer Journal*, 12(2):124–127, 01 1969. ISSN 0010-4620. doi: 10.1093/comjnl/12.2.124. URL <https://doi.org/10.1093/comjnl/12.2.124>.
- [19] Raymon A. D’Inverno. Algebraic computing in general relativity. *General Relativity and Gravitation*, 6(6): 567–593, dec 1975. doi: 10.1007/BF00761964.
- [20] Éricourgoulhon and Marco Mancini. Symbolic tensor calculus on manifolds: a SageMath implementation. *Les cours du CIRM*, 6(1):1–54, 2018. doi: 10.5802/ccirm.26. URL <https://doi.org/10.5802%2Fccirm.26>.
- [21] M.A.H. MacCallum. Computer algebra in gravity research. *Living Rev Relativ*, 21:6, aug 2018. doi: 10.1007/s41114-018-0015-6.
- [22] D.A. Bolotin and S.V. Poslavsky. Introduction to Redberry: a Computer Algebra System designed for tensor manipulation. *arXiv e-prints*, feb 2013.
- [23] S. Poslavsky and D. Bolotin. Redberry: a Computer Algebra System designed for tensor manipulation. In *Journal of Physics Conference Series*, volume 608 of *Journal of Physics Conference Series*, page 012060, may 2015. doi: 10.1088/1742-6596/608/1/012060.
- [24] Zimmermann P.; Casamayou A.; Cohen N.; Connan G. *Computational Mathematics with Sage-Math — Polynomials*, volume 10.1137/1.9781611975468, chapter Chapter 7, pages 125–151. SIAM, 2018. ISBN 978-1-61197-545-1, 978-1-61197-546-8. doi: 10.1137/1.9781611975468.ch7. URL <http://gen.lib.rus.ec/scimag/index.php?s=10.1137/1.9781611975468.ch7>.

- [25] Kasper Peeters. Cadabra: a field-theory motivated symbolic computer algebra system. *Computer Physics Communications*, 176(8):550 – 558, 2007. ISSN 0010-4655. doi: <https://doi.org/10.1016/j.cpc.2007.01.003>. URL <http://www.sciencedirect.com/science/article/pii/S0010465507000318>.
- [26] L. Brewin. A brief introduction to Cadabra: A Tool for Tensor Computations in General Relativity. *Computer Physics Communications*, 181:489–498, mar 2010. doi: 10.1016/j.cpc.2009.10.020.
- [27] V.P. Gerdt. Analytical calculations in high energy physics by computer. *Computer Physics Communications*, 20:85–90, sep 1980. doi: 10.1016/0010-4655(80)90113-7.
- [28] V.P. Gerdt, O.V. Tarasov, and D.V. Shirkov. Reviews of topical problems: Analytic calculations on digital computers for applications in physics and mathematics. *Soviet Physics Uspekhi*, 23:59–77, jan 1980. doi: 10.1070/PU1980v023n01ABEH004898.
- [29] H. Strubbe. Development of the schoonschip program. *Computer Physics Communications*, 18:1–5, sep 1979. doi: 10.1016/0010-4655(79)90019-5.
- [30] Martinus J. G. Veltman and David N. Williams. Schoonschip '91. *The University of Michigan Ann Arbor, MI 48109-1120*, 8, 06 1993. URL <http://arxiv.org/abs/hep-ph/9306228v2>.
- [31] M. J. Levine. Dirac matrix and tensor algebra on a digital computer. pages 454–455, 1967.
- [32] M.J. Levine and R. Roskies. Hyperspherical approach to quantum electrodynamics: sixth-order magnetic moment. *Physical Review D*, 9(2):421–429, January 1974. doi: 10.1103/PhysRevD.9.421.
- [33] M. J. Levine and R. Roskies. Ashmedai and a large algebraic problem. In *Proceedings of the Third ACM Symposium on Symbolic and Algebraic Computation*, SYMSAC '76, pages 359–364, New York, NY, USA, 1976. ACM. doi: 10.1145/800205.806357. URL <http://doi.acm.org/10.1145/800205.806357>.
- [34] V.E. Lewis, Massachusetts Institute of Technology, Massachusetts Institute of Technology. Laboratory for Computer Science, United States. Army Research Office, and United States. Office of Naval Research. *Proceedings of the 1979 MACSYMA Users' Conference: Washington, D.C., June 20-22, 1979*. MIT Laboratory for Computer Science, 1979. URL <https://books.google.co.ve/books?id=NNY-AAAAIAAJ>.
- [35] Richard Pavelle and Paul S. Wang. Macsyma from f to g. *Journal of Symbolic Computation*, 1(1):69 – 100, 1985. ISSN 0747-7171. doi: [https://doi.org/10.1016/S0747-7171\(85\)80030-4](https://doi.org/10.1016/S0747-7171(85)80030-4). URL <http://www.sciencedirect.com/science/article/pii/S0747717185800304>.

- [36] Joel Moses. Macsyma: A personal history. *Journal of Symbolic Computation*, 47(2): 123 – 130, 2012. ISSN 0747-7171. doi: <https://doi.org/10.1016/j.jsc.2010.08.018>. URL <http://www.sciencedirect.com/science/article/pii/S0747717110001483>.
- [37] Matthias Steinhauser, Takahiro Ueda, and Jos A.M. Vermaseren. Parallel versions of form and more. *Nuclear and Particle Physics Proceedings*, 261-262:45–57, apr 2015. doi: 10.1016/j.nuclphysbps.2015.03.006.
- [38] B. Ruijl, T. Ueda, and J. Vermaseren. Form version 4.2. *arXiv e-prints*, July 2017.
- [39] J. Vermaseren. New features of form and future plans. In *13th International Symposium on Radiative Corrections; Applications of Quantum Field Theory to Phenomenology. 25-29 September, 2017 . St. Gilgen, Austria. Online at jA href="https://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=290">https://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=290">https://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=290*, id.22, page 22, January 2017.
- [40] Stephen Wolfram. *Some Topics in Theoretical High-Energy Physics*. PhD thesis, CALIFORNIA INSTITUTE OF TECHNOLOGY., jan 1980.
- [41] Stephen Wolfram. Undecidability and intractability in theoretical physics. *prl*, 54(8):735–738, February 1985. doi: 10.1103/PhysRevLett.54.735.
- [42] Stephen Wolfram. The practical and theoretical future of computation in physics. In *APS March Meeting Abstracts*, volume 2019 of *APS Meeting Abstracts*, page F22.001, January 2019.
- [43] Stephen Wolfram. A class of models with the potential to represent fundamental physics. *arXiv e-prints*, art. arXiv:2004.08210, April 2020.
- [44] Bruce W. Char, Keith O. Geddes, W. Morven Gentleman, and Gaston H. Gonnet. The design of maple: A compact, portable and powerful computer algebra system. In *In EUROCAL '83, Proceedings of the International Symposium on Symbolic and Algebraic Computation*, EUROCAL '83, pages 101–115, London, UK, UK, apr 1983. Springer-Verlag. ISBN 3-540-12868-9. URL <http://dl.acm.org/citation.cfm?id=646657.700400>.
- [45] Portugal R. and Sautú S.L. Applications of maple to general relativity. *Computer Physics Communications*, 105, 1997. URL <http://www.sciencedirect.com/science/article/pii/S0010465597000787>.
- [46] Dumitru N. Vulcanov and Gabriela Ciobanu. Maple procedures in teaching the canonical formalism of general relativity. *arXiv e-prints*, art. physics/0010053, oct 2000.
- [47] V.L. Kalashnikov. Introduction to Relativistic Astrophysics and Cosmology through Maple. *arXiv General Relativity and Quantum Cosmology e-prints*, mar 2001.

- [48] D.N. Vulcanov. Gravity, torsion, dirac field and computer algebra using maple and reduce. *arXiv General Relativity and Quantum Cosmology e-prints*, September 2002.
- [49] Wolfgang Schreiner, Christian Mittermaier, and Karoly Bosa. Distributed maple: parallel computer algebra in networked environments. *Journal of Symbolic Computation*, 35(3):305 – 347, 2003. ISSN 0747-7171. doi: [https://doi.org/10.1016/S0747-7171\(02\)00137-2](https://doi.org/10.1016/S0747-7171(02)00137-2). URL <http://www.sciencedirect.com/science/article/pii/S0747717102001372>.
- [50] D.N. Vulcanov and V.D. Vulcanov. Maple libraries for cosmology. *eprint arXiv:cs/0409006*, sep 2004.
- [51] Frank Y. Wang. *Physics with MAPLE: The Computer Algebra Resource for Mathematical Methods in Physics*. Physics Textbook. Wiley-VCH, 1 edition, may 2006. ISBN 3-527-40640-9,9783527406401. URL <http://gen.lib.rus.ec/book/index.php?md5=1BFCDB89E13646633CC45CE9922112B6>.
- [52] C.A. Sporea and D.N. Vulcanov. Using maple + grtensorii in teaching basics of general relativity and cosmology. *Rom. Rep. Phys.*, (68):29–40, 2016.
- [53] SWMATH. Information service for mathematical software. SWMATH, 2021. <https://www.swmath.org/>.
- [54] SIGSAM Bulletin. Special interest group on symbolic & algebraic manipulation. Dictionary Of Computing, 2021. <http://www.sigsam.org/Resources/Software.html>.
- [55] Maxima A Computer Algebra System. Other free computer algebra systems. MAXIMA, 2021. <https://maxima.sourceforge.io/compalg.html>.
- [56] RISC Research Institute for Symbolic Computation. Research institute for symbolic computation. Research Institute for Symbolic Computation, 2021. <https://risc.jku.at/software/>.
- [57] FOLDOC. Free on line dictionary of computing. Dictionary Of Computing, 2021. <https://foldoc.org/symbolic+mathematics>.
- [58] Computer Algebra Netherlands. Foundation can (computer algebra netherlands). Computer Algebra Netherlands, 2021. <https://computeralgebra.nl/>.
- [59] W. S. Brown. Proceedings of the first acm symposium on symbolic and algebraic manipulation. In SYSMAC '66, editor, *A language and system for symbolic algebra on a digital computer ALTRAN*, number 1 in SAAC, pages 501–540, 1966. doi: 10.1145/800005.807955.
- [60] A. C. Hearn. The reduce program for computer algebra. In A. Visconti, editor, *3th Colloquium on Advanced Computing Methods in Theoretical Physics*, pages

- AV/1–AV/19, Marseilles, jun 1973. Centre de Physique Théorique CNRS. URL https://inis.iaea.org/collection/NCLCollectionStore/_Public/07/229/7229583.pdf?r=1&r=1. CNRS technical report 73/P.550.
- [61] Michael Monagan and Roman Pearce. Poly: A new polynomial data structure for maple 17. *ACM Commun. Comput. Algebra*, 46(3/4):164 – 167, jan 2013. ISSN 1932-2240. doi: 10.1145/2429135.2429173. URL <https://doi.org/10.1145/2429135.2429173>.
- [62] Char Bruce W., Geddes Keith O., Gonnet Gaston H., Leong Benton, Monagan Michael B., and Watt Stephen M. *Maple V Language Reference Manual*. Springer US, 1 edition, 1991. ISBN 978-0-387-94124-0, 978-1-4615-7386-9. URL <http://gen.lib.rus.ec/book/index.php?md5=33976bde4f045b1ce2e97f0abe593d0e>.
- [63] Labahn George Geddes Keith O., Czapor Stephen R. *Algorithms for Computer Algebra*. Springer, 1992 edition, 1992. ISBN 0792392590, 9780792392590. URL <http://gen.lib.rus.ec/book/index.php?md5=8dc5ec01fc92cfffdddeeb422cc6149edb>.
- [64] Maplesoft. Index of descriptions for packages of library functions. Maple, 7 2021. <https://www.maplesoft.com/support/help/Maple/view.aspx?path=index/package>.
- [65] Lake K. Musgrave P., Pollney D. Grtensoriii package for maple. GRTensorIII, 2021. <https://github.com/grtensor/grtensor>.
- [66] P. Musgrave and K. Lake. Engineering applications of grtensorii 1:elasticity-finite deformations. *GRTensorII*, 1(1):22, aug 1998.
- [67] V. Medina and N. Falcon . Relativistic model for radiating spherical collapse. *Progress in Physics*, 14(1): 46–53, jan 2018.
- [68] Víctor Medina. A self-similar solution of a fluid with spherical distribution in general relativity. *Indian Journal of Physics*, 96(1):317–328, jan 2022. doi: 10.1007/s12648-020-01959-1.
- [69] Davies G. pertutils.mpl by george davies written for grtensorii. PertUtils, 1998. <http://grtensor.phy.queensu.ca/NewDemo/Blackholes/Utils/pertutils.mpl>.
- [70] G. Davies. Second-order black hole perturbations: A computer algebra approach, i - the schwarzschild space-time. *arXiv General Relativity and Quantum Cosmology e-prints*, October 1998.
- [71] L. Herrera, J. Jiménez, and G. J. Ruggeri. Evolution of radiating fluid spheres in general relativity. *Phys. Rev. D*, 22:2305–2316, nov 1980. doi: 10.1103/PhysRevD.22.2305.

-
- [72] V. Medina, L. Núñez, H. Rago, and A. Patiño. Evolution of radiating charged spheres in general relativity. *Canadian Journal of Physics*, 66:981–986, nov 1988. doi: 10.1139/p88-158. URL <http://gen.lib.rus.ec/scimag/index.php?s=10.1139/p88-158>.
- [73] H. Bondi. The Contraction of Gravitating Spheres. *Proceedings of the Royal Society of London Series A*, 281: 39–48, aug 1964. doi: 10.1098/rspa.1964.0167.
- [74] P. Musgrave and K. Lake. Junctions and thin shells in general relativity using computer algebra: I. the darmois - israel formalism. *Classical and Quantum Gravity*, 13:1885–1899, July 1996. doi: 10.1088/0264-9381/13/7/018.

A self-similar solution of a fluid with spherical distribution in general relativity

Víctor Medina

Indian Journal of Physics

ISSN 0973-1458

Indian J Phys

DOI 10.1007/s12648-020-01959-1



A self-similar solution of a fluid with spherical distribution in general relativity

V Medina* 

Departamento de Física, Facultad de Ingeniería, Universidad de Carabobo, Valencia, Venezuela

Received: 22 December 2019 / Accepted: 29 September 2020

Abstract: The purpose of this article is to use an algorithm to obtain a non-static solution to the Einstein–Euler equations of a spherically symmetric matter distribution of a perfect fluid. The equations obtained make up a nonlinear system of partial derivative equations (PDE), which in the vast majority of cases is quite difficult to solve, either analytically or numerically. In this context, the spherical symmetry of space-time is very useful since it allows to establish the self-similarity of the solutions. This assumption reduces the PDEs to a set of ordinary differential equations (ODE). Therefore, the self-similarity hypothesis is very powerful in the search for non-static analytical or numerical solutions. The ODEs thus established allow defining a set of variables adjusted to the boundary conditions of the spherical distribution of matter, together with an equation of state (EOS). In this work, Tolman’s V solution has been selected as EOS. Once the numerical integration has been carried out, the variables established in the ODE and some other physical variables determined in the algorithm (density, pressure, radiation, etc.) can present damped oscillations (rebounds), depending on the initial values. This behavior is similar to that found near phase transitions in condensed matter physics, but now the mass distribution plays the role of an order parameter. This result has been obtained in other simulations of numerical relativity, where the PDE obtained from the gravitational field equations are integrated and the expression of an ultrarelativistic fluid ($P = \kappa \rho$). It is important to note that there is the possibility of carrying out simulations with other EOS (with electrical charge, anisotropy, etc.) with this algorithm. Many of the calculations to obtain the field equations, such as the conservation equations, were performed using the *GRTensorIII* computational algebra package, running on *Maple 2017*; as well as some Maple routines that have been used for these types of problems.

Keywords: Relativistic fluids; Radiative gravitational collapse; Bianchi’s identities; Tolman V; Maple; GRTensorIII; Numerical relativity

1. Introduction

One of the most interesting problems in relativistic astrophysics and the theory of gravitation is to provide a description of the evolution of the gravitational collapse of a star of various solar masses. These compact objects have proven to be excellent tools for testing the fundamental properties of gravity and matter under extreme conditions. As a consequence, in the study of the final stages of stellar evolution, general relativity is of great importance [1, 2]. However, the gravitational field equations are nonlinear partial derived equations, which present some difficulties in obtaining their solution. Some exact analytical solutions [3]

and approximate analytical solutions [4] are known. As a result of the difficulty of obtaining analytical solutions, many efforts are made to obtain numerical solutions with the help of computers to model relativistic astrophysical objects. Within this field of numerical relativity, many efforts are made, among which we can include the work of [5–16] and many others. Although it is necessary to highlight the work of Choptuik [17].

Choptuik (see [18–21]) describes the phenomenon of critical gravitational collapse as follows: Although objects with very little gravity will not form any black holes and very strong gravitational objects can evolve into high mass black holes, then a whole family of objects can be “labeled” with a p parameter by interpolating the states between weak gravitation and strong gravitation. Here, p is a parameter that describes some aspect of the initial energy distribution of the scalar field such that for $p > p^*$ black

*Corresponding author, E-mail: vmedina@uc.edu.ve

holes are formed during evolution, while for $p < p^*$ all the scalar field is scattered to infinity. Hence, p^* denotes the black hole formation threshold for the particular family of initial data under consideration. The solution that leads to the limit $p \rightarrow p^*$ is called the *critical solution*. The result is that near the threshold p^* , the mass M can be described by the relationship:

$$M \propto (p - p^*)^\gamma, \quad (1)$$

where γ is a universal constant that does not depend on which family of scalar field initial data parameters is chosen [22]. This recalls the scale found near phase transitions in condensed matter physics, but now with the mass of the black hole playing the role of the order parameter. Furthermore, the critical solution (the result of the evolution of the initial data $p = p^*$) can have either continuous self-similarity (CSS) or discrete self-similarity (DSS), or none at all. In the case of CSS, the solutions are expressed in terms of some dimensionless combination of the time coordinate t and the radial coordinate r . In the simplest situation, the solution is invariant under the transformation $r \rightarrow ar$, $t \rightarrow at$ for any constant a . Geometrically, this corresponds to the existence of a homothetic Killing vector and is sometimes called *first* type similarity [23, 24]. A solution behaves with a DSS, if it has to be repeated after a certain time, the solution is repeated, with the global scale of reduced space. In other words, the scalar field collapses, but most bounces and radiates leaving a smaller amount of scalar field than it collapses on a smaller time scale, and most bounces and radiates leaving an even smaller amount of scalar field, etc. This entire infinite sequence of collapses takes a finite total amount of time, leaving a singularity [21]. When obtaining self-similar solutions for Einstein's field equations, there are at least two interesting features. The first is that it reduces the procedure to obtain nonlinear PDE solutions to ODE [25, 26]. In principle, this makes the solutions easier to study mathematically and the existence of an invariance that allows such a reduction [27]. Second, self-similar solutions play an important role in describing the asymptotic properties of more general models [28, 29].

Of particular importance is the discovery of critical phenomena in the collapse of a radiant fluid by [30]. A radiant fluid is a special case of an ultra-relativistic fluid, that is, a fluid whose EOS is

$$P = \kappa \cdot \rho, \quad (2)$$

where P is the pressure, ρ the energy density, and κ is a dimensionless constant that, for a radiant fluid, takes the value $\kappa = \frac{1}{3}$. For ultra-relativistic fluids, the critical solution is CSS, unlike DSS found for massless scalar wave. The critical exponent γ for a radiant fluid was determined analytically by [31] who considered several different

values of κ . The study of rebounds or spherical symmetry disturbances was considered by [22]. He showed that these modes have a damped oscillatory behavior described by

$$S \propto e^{\lambda T} \cos(\omega T + \phi), \quad (3)$$

where the damping coefficient λ and the angular frequency ω depend on the constant κ of the ultra-relativistic EOS (2) and that, for these modes, λ is negative if $\kappa \leq 0.49$, resulting in a damping of these modes, but that λ becomes positive for $\kappa \geq 0.49$, in which case the mode grows [15].

In almost all numerical simulations of gravitational collapse, the material under study is considered to be a fluid perfect ultra-relativistic and therefore, the expression (2) is taken as its EOS. However, this consideration is not mandatory [32] and of many of the EOS that can be used to study stellar material, we can highlight the EOS of Tolman's solution V, which in its nonparametric version is expressed as

$$P = \frac{1}{3} \left\{ 1 - \left(\frac{14}{3} \right) \frac{1}{(4\pi r^2 \rho)} \left[\frac{1}{4\pi r^2 (3\rho + 5P)} \right]^{\frac{1}{6}} \right\} \rho, \quad (4)$$

and can be considered a generalization of (2) or the constant MIT Bag [33]. Of all the analytical solutions described by [34], the least used, from the point of view of astrophysics, is solution V, although it was used in the article [35] and [36].

This article uses the self-similarity property of the solutions of the EEE (Einstein field equations and the conservation equations) [37, 38] with the aim of obtaining a system of ODE and uses, instead of the EOS (2), the EOS of the solution V of Tolman (4). For this, in Sect. 2, the field equations and the Bianchi identities are written, which are the basic theoretical framework of the study for the non-static solution. In Sect. 3, the algorithm is constructed with the EEE and the surface variables are defined, as well as the ODE that control their temporal evolution. Then, in Sect. 4 the solution V of Tolman is used as a static solution to apply the algorithm. However, to obtain one of the surface equations, the trace of the field equation is used. It can be shown, in this approach, that the trace of the field equation is equivalent to the Tolman–Oppenheimer–Volkoff (TOV) equation. Results and some comments are presented in Sect. 5. In the last section, the final observations are shown.

2. The basic equations

Here, we offer a brief general description of the method used to obtain a non-static solution for fluid radiant

spheres. For more details, we can consult the algorithm used in [27].

Consider a distribution with self-gravitating spherical symmetry of collapsing matter. This distribution of material will be modeled as a perfect fluid with energy density $\hat{\rho}$, pressure $\hat{P}$, seen by a local Minkowskian observer moving with the fluid. In addition, there is a non-polarized radiation flow that moves in the radial direction with a density $\hat{\varepsilon}$ and $3\hat{\sigma}$ is the isotropic radiation of the energy density measured by the same observer.

In radiation coordinates, the metric takes the form.

$$ds^2 = e^{2\beta} \frac{V}{r} du^2 + 2e^{2\beta} du dr - r^2 d\theta^2 - r^2 \sin^2 \theta d\phi^2, \quad (5)$$

where u and r are time-like and radial-like coordinates, respectively; β and V are functions of u and r ; θ, ϕ are the usual angles coordinates. In this coordinates, the gravitational field equations are:

$$\begin{aligned} -8\pi T_{00} &= -\frac{V_{,0} - 2V\beta_{,0}}{r^2} - \frac{V}{r^3} (e^{2\beta} - V_{,1} + 2\beta_{,1}V), \\ -8\pi T_{01} &= -\frac{1}{r^2} (e^{2\beta} - V_{,1} + 2\beta_{,1}V), \\ -8\pi T_{11} &= -\frac{4\beta_{,1}}{r}, \\ -8\pi T_2^2 &= -8\pi T_3^3 = -e^{2\beta} \left\{ 2\beta_{01} - \frac{1}{2r^2} [rV_{,11} - 2\beta_{,1}V \right. \\ &\quad \left. + 2r(\beta_{,11}V + \beta_{,1}V_{,1})] \right\}. \end{aligned}$$

Notice that subindices $_{,0}$ have been used for the derivative of the time-type variable u and $_{,1}$ for the derivative of the radial variable r . The semicolon $_{;\alpha}$ for covariant differentiation. The expressions that relate the coordinates of the local Minkowskian system with Bondi's radiative coordinates are:

$$\begin{aligned} dt &= \left(\frac{\partial t}{\partial u} \right) du + \left(\frac{\partial t}{\partial r} \right) dr \\ &= e^\beta \left(\frac{V}{r} \right)^{\frac{1}{2}} du + e^\beta \left(\frac{r}{V} \right)^{\frac{1}{2}} dr, \end{aligned} \quad (6)$$

$$dx = \left(\frac{\partial x}{\partial r} \right) dr = e^\beta \left(\frac{r}{V} \right)^{\frac{1}{2}} dr, \quad (7)$$

$$dy = \left(\frac{\partial y}{\partial \theta} \right) d\theta = r d\theta, \quad (8)$$

$$dz = \left(\frac{\partial z}{\partial \phi} \right) d\phi = r \cdot \sin \theta d\phi. \quad (9)$$

Considering that the material within the distribution of matter basically consists of a perfect fluid, then we can write the impulse energy tensor, corresponding to the matter in the distribution as:

$$\hat{T}_{\alpha\beta} = (\hat{\rho} + \hat{P}) \cdot U_\alpha U_\beta - \hat{P} \cdot \eta_{\alpha\beta}, \quad (10)$$

where $U_\alpha = (1, 0, 0, 0)$. For an observer moving in the same local Minkowski system, with a radial velocity $\omega = \frac{dx}{dt}$, the expression of the tensor in the Lorentz system, we can write:

$$\bar{T}_{\mu\nu} = A_\mu^\alpha(\omega) A_\nu^\beta(\omega) \hat{T}_{\alpha\beta}, \quad (11)$$

where the Lorentz transformation matrix is defined as

$$A_\mu^\alpha(\omega) = \begin{bmatrix} \frac{1}{\sqrt{1-\omega^2}} & -\frac{\omega}{\sqrt{1-\omega^2}} & 0 & 0 \\ -\frac{\omega}{\sqrt{1-\omega^2}} & \frac{1}{\sqrt{1-\omega^2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (12)$$

By replacing the values of density, pressure and radiation with

$$\bar{\rho} = \hat{\rho} + 3\hat{\sigma}, \quad \bar{P} = \hat{P} + \hat{\sigma}, \quad \bar{\varepsilon} = \hat{\varepsilon} \frac{1+\omega}{1-\omega},$$

we obtain that the non-null contravariant components of the total impulse energy tensor (Matter + Polarized Radiation + Nopolarized Radiation) in the Lorentz system are:

$$\begin{aligned} \bar{T}_{00} &= \bar{\varepsilon} + \frac{\bar{\rho} + \omega^2 \bar{P}}{1 - \omega^2}, \\ \bar{T}_{01} = \bar{T}_{10} &= -\bar{\varepsilon} - \frac{\omega}{1 - \omega^2} (\bar{\rho} + \bar{P}), \\ \bar{T}_{11} &= \frac{\bar{P} + \omega^2 \bar{\rho}}{1 - \omega^2} + \bar{\varepsilon}, \\ \bar{T}_{22} = \bar{T}_{33} &= \bar{P}, \end{aligned}$$

using (6)–(9) the contravariant components of the energy-impulse tensor are obtained in radiative coordinates such as:

$$\begin{aligned} T_{00} &= e^{2\beta} \left(\frac{V}{r} \right) \left(\frac{\bar{\rho} + \omega^2 \bar{P}}{1 - \omega^2} + \bar{\varepsilon} \right), \\ T_{01} = T_{10} &= e^{2\beta} \left(\frac{\bar{\rho} - \omega \bar{P}}{1 + \omega} \right), \\ T_{11} &= e^{2\beta} \left(\frac{r}{V} \right) \left(\frac{1 - \omega}{1 + \omega} \right) (\bar{\rho} + \bar{P}), \\ T_{22} &= \frac{T_{33}}{\sin^2 \theta} = r^2 \bar{P}. \end{aligned}$$

In addition, the velocity of matter in radiative coordinates can be calculated and you get

$$\frac{\partial r}{\partial u} = \frac{V}{r} \cdot \frac{\omega}{1 - \omega}. \quad (12)$$

When defining the effective variables as

$$\rho \equiv \frac{\bar{\rho} - \omega \bar{P}}{1 + \omega}, \quad P \equiv \frac{\bar{P} - \omega \bar{\rho}}{1 + \omega}, \quad \varepsilon \equiv \bar{\varepsilon}. \quad (14)$$

For the limit case $\omega = 0$ or in the origin $r = 0$, it is necessary that $\rho = \bar{\rho}$ and $P = \bar{P}$. Rewriting the $T_{\mu\nu}$ components based on the effective variables:

$$T_{00} = e^{2\beta} \left(\frac{V}{r} \right) \left[\frac{\omega(\rho + P)}{(1 - \omega)^2} + \rho + \varepsilon \right],$$

$$T_{01} = T_{10} = e^{2\beta} \rho,$$

$$T_{11} = e^{2\beta} \left(\frac{r}{V} \right) (\rho + P),$$

$$T_{22} = \frac{T_{33}}{\sin^2 \theta} = r^2 \bar{P},$$

and the field equations are

$$\begin{aligned} & -\frac{V}{r^2} \left[\left(2\beta_{,0} - \frac{V_{,0}}{V} \right) - \frac{1}{r} (2V\beta_{,1} - V_{,1} + e^{2\beta}) \right] \\ & = 8\pi e^{2\beta} \left(\frac{V}{r} \right) \left[\varepsilon + \rho + \frac{\omega(\rho + P)}{(\omega - 1)^2} \right], \end{aligned} \quad (15)$$

$$2V\beta_{,1} - V_{,1} + e^{2\beta} = 8\pi r^2 e^{2\beta} \rho, \quad (16)$$

$$\frac{4\beta_{,1}}{r} = 8\pi \frac{r}{V} e^{2\beta} (\rho + P), \quad (17)$$

$$\begin{aligned} & -2\beta_{,01} + \frac{1}{2r^2} [rV_{,11} - 2\beta_{,1}V + 2r(V_{,1}\beta_{,1} \\ & + V\beta_{,11})] = 8\pi e^{2\beta} \bar{P}. \end{aligned} \quad (18)$$

Three non-trivial relationships of conservation equations ($T_{\nu;\mu}^\mu = 0$) are obtained (Bianchi identities):

$$\begin{aligned} T_{1;\mu}^\mu &= \frac{e^{-2\beta}}{2\pi r} \beta_{,10} - \frac{\partial P}{\partial r} \\ & - \frac{1}{2} \left(2\beta_{,1} + \frac{V_{,1}}{V} - \frac{1}{r} \right) (\rho + P) - \frac{2(P - \bar{P})}{r} = 0, \end{aligned} \quad (19)$$

$$\begin{aligned} T_{0;\mu}^\mu &= \frac{V}{r} \left\{ \left(2\beta_{,1} + \frac{V_{,1}}{V} + \frac{1}{r} \right) \right. \\ & \times \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \frac{\omega}{(1 - \omega)^2} \frac{\partial(\rho + P)}{\partial r} \Big\} \\ & + \frac{1}{2} \left(2\beta_{,0} - \frac{V_{,0}}{V} \right) (\rho + P) + \frac{\partial \rho}{\partial u} + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} \right. \\ & \left. + \frac{1 + \omega}{(1 - \omega)^3} (\rho + P) \frac{\partial \omega}{\partial r} \right] = 0, \end{aligned} \quad (20)$$

$$\begin{aligned} e^{2\beta} T_{;\mu}^{\mu 1} &= \frac{V}{r} \left\{ \left(2\beta_{,1} + \frac{V_{,1}}{V} + \frac{1}{r} \right) \right. \\ & \times \left[\varepsilon + \frac{1 + \omega^2}{2(1 - \omega)^2} (\rho + P) \right] \\ & + \frac{\omega(\rho + P)_{,1}}{(1 - \omega)^2} + \frac{\partial P}{\partial r} + \frac{1}{r} [\rho + \bar{P} - (P - \bar{P})] \Big\} \\ & - \frac{1}{2} \left(2\beta_{,0} - \frac{V_{,0}}{V} \right) (\rho + P) + \frac{\partial P}{\partial u} + \frac{V}{r} \\ & \times \left[\frac{\partial \varepsilon}{\partial r} + \frac{1 + \omega}{(1 - \omega)^3} (\rho + P) \frac{\partial \omega}{\partial r} \right] = 0. \end{aligned} \quad (21)$$

Comparing the previous Eqs. (19)–(21), it can be seen that two of them are independent, since there is a relationship between them:

$$e^{2\beta} T_{;\mu}^{\mu 1} - T_{0;\mu}^\mu = \left(\frac{V}{r} \right) T_{1;\mu}^\mu = 0. \quad (22)$$

and the trace of the field equation is written as:

$$\begin{aligned} & 8\pi[(\rho - \bar{P}) + (P - \bar{P})] \\ & = 4\beta_{,01} e^{-2\beta} - \frac{e^{-2\beta}}{r^2} [rV_{,11} - 2V\beta_{,1} + \\ & + 2r(V_{,1}\beta_{,1} + V\beta_{,11})] - \frac{2e^{-2\beta}}{r^2} (V_{,1} - e^{2\beta}). \end{aligned} \quad (23)$$

The trace of the field equation is important since it can be used as an alternative to some of the conservation equations. In particular, it can be shown that it is equivalent to (19).

By defining the Bondi mass aspect as, $V \equiv e^{2\beta}(r - 2m)$, and after some algebra, we can write the gravitational field equations as:

$$m_{,1} = 4\pi r^2 \rho, \quad (24)$$

$$\beta_{,1} = 2\pi r \frac{(\rho + P)}{1 - \frac{2m}{r}}, \quad (25)$$

$$m_{,0} = -4\pi r^2 e^{2\beta} \left(1 - \frac{2m}{r} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right], \quad (26)$$

$$\begin{aligned} 8\pi \bar{P} &= -2\beta_{,01} e^{-2\beta} + \left(1 - \frac{2m}{r} \right) \\ & \times \left(2\beta_{,11} + 4\beta_{,1}^2 - \frac{\beta_{,1}}{r} \right) \\ & + \frac{1}{r} [3\beta_{,1}(1 - 2m_{,1}) - m_{,11}], \end{aligned} \quad (27)$$

also, the corresponding version for the expression of the trace of the field equations:

$$\begin{aligned}
 & 4\pi[(\bar{P} + P) + (\bar{P} - \rho)] \\
 &= -2\beta_{,01}e^{-2\beta} + \left(1 - \frac{2m}{r}\right) \left(2\beta_{,11} + 4\beta_{,1}^2 + \frac{\beta_{,1}}{r}\right) \\
 &+ \frac{1}{r} [3\beta_{,1}(1 - 2m_{,1}) - m_{,11}] - \frac{2m_{,1}}{r^2},
 \end{aligned} \tag{28}$$

and taking as independent conservation equations (19) and (20), we have

$$\begin{aligned}
 & -\frac{e^{-2\beta}}{2\pi r} \beta_{,10} + \frac{\partial P}{\partial r} + \left(4\pi r^2 P + \frac{m}{r}\right) \\
 & \times \frac{(\rho + P)}{r(1 - \frac{2m}{r})} + \frac{2}{r}(P - \bar{P}) = 0,
 \end{aligned} \tag{29}$$

$$\begin{aligned}
 & \frac{e^{2\beta}}{r} \left[1 + \left(1 - \frac{2m}{r}\right) + 4\pi r^2(P - \rho) \right] \\
 & \times \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] \\
 & + \frac{V}{r} \frac{\omega}{(1 - \omega)^2} \frac{\partial}{\partial r}(\rho + P) \\
 & + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} + \frac{(1 + \omega)(\rho + P)}{(1 - \omega)^3} \frac{\partial \omega}{\partial r} \right] + \frac{\partial \rho}{\partial u} = 0.
 \end{aligned} \tag{30}$$

At this point, it is necessary to emphasize the following:

- It can be demonstrated that the expression of the trace of the field equation (28) is equivalent to Eq. (29), similar to the hydrostatic equilibrium equation of TOV, (see, for example [39]).
- Regarding the conservation equation (30), it can also be obtained from the field equations (24) and (26) and observing that the second mixed derivatives commute; i.e., $m_{,01} = m_{,10}$.

Now, combining (26) with (30)

$$\begin{aligned}
 & \frac{2m_{,0}e^{-2\beta}}{4\pi r^3(1 - \frac{2m}{r})} \left[1 - \frac{m}{r} + 2\pi r^2(P - \rho) \right] \\
 &= \frac{\omega}{(1 - \omega)^2} \left(1 - \frac{2m}{r} \right) (\rho + P)_{,1} + r \left[\frac{\partial \varepsilon}{\partial r} \right. \\
 & \left. + \frac{(1 + \omega)(\rho + P)}{(1 - \omega)^3} \frac{\partial \omega}{\partial r} \right] + \frac{e^{2\beta}}{4\pi r^2} m_{,10}.
 \end{aligned} \tag{31}$$

Considering that the radiation profiles ε and the value of ω vary little, one can write an expression very similar to the Euler equation:

$$\begin{aligned}
 & 2m_{,0} \left[1 - \frac{m}{r} + 2\pi r^2(P - \rho) \right] \\
 &= \frac{\omega \cdot e^{2\beta}}{(1 - \omega)^2} \left[4\pi r^2 \left(1 - \frac{2m}{r} \right)^2 r(\rho + P)_{,1} \right] \\
 &+ r \left(1 - \frac{2m}{r} \right) m_{,10}.
 \end{aligned} \tag{32}$$

An expression similar to the previous one can be found in [40], although it was not used in any other subsequent work. Eq. (32) allows us to relate the mass exchange with the other quantities of the self-gravitating system and is omitted in previous works on the evolution of radiant fluid spheres [35, 40–44]. This omission prevents the completion of the system of equations and motivates the inclusion of an ad hoc function for luminosity [35, 41, 43, 45–47]. With the field equations (24)–(27), we can calculate the expressions of the physical variables $\omega, \bar{P}, \bar{\rho}$, if we know the expressions $m(u, r)$ and $\beta(u, r)$ in each layer of the material under study. This is why, the EOS $P(u, r)$ $\rho(u, r)$ are of great importance in determining the behavior of the physical variables present in the equations field and its subsequent evolution.

3. Building the surface equations

Considering the field equations (24) and (25), we can perform the integration and obtain the values of m and β as a function of the radial coordinate:

$$m = \int_0^r 4\pi r^2 \rho \, dr, \tag{33}$$

$$\beta = 2\pi \int_a^r \frac{\rho + P}{1 - \frac{2m}{r}} r \cdot dr. \tag{34}$$

With these two equations, and by knowing the radial dependence of P and ρ , the expressions of the metric functions m and β can be obtained except for their temporal dependence which is related to the contour conditions. Here to determine these functions, dependent on the time variable u , a modification of the method used in [27] is used to obtain a non-static solution, which can be summarized in:

1. Suppose that the effective variables P and ρ (Eq. 14) have the same radial dependence as in the static solution, but noting that now the boundary condition $\bar{P}_a = 0$ now expresses as

$$P_a = -\omega_a \rho_a. \tag{35}$$

The subscript a indicates that the quantity is evaluated at the boundary of the distribution.

2. Using the expressions of the effective variables and with (33) and (34), the values of m and β are calculated except for three surface variables that are adjusted by means of the material distribution boundary conditions for both the field equations and the conservation equations.

- (a) Equation (13) evaluated at $r = a$,
- (b) Equation (32) evaluated at $r = a$,
- (c) Instead of using the conservation equation (19) evaluated in a , we will use the equation of the trace of the field equations, (28) evaluated in a ,

3. By selecting a set of initial data, to perform the numerical integration of the ODE established in (2) the functions m and β are fully known,

4. Using Eqs. (24) to (27), the values of the physical variables in each section or layer of the material distribution are calculated.

Following the procedure established in (2), we are going to define the surface variables as well as the differential equations that control their temporal variation.

- As indicated in (2a), evaluating the expression (13) in $r = a$, we obtain one of the surface equations

$$\dot{a} = \frac{\partial a}{\partial u} = \left(1 - \frac{2m_a}{a}\right) \left(\frac{\omega_a}{1 - \omega_a}\right). \quad (36)$$

Normalizing, with respect to the initial mass $m(0) = m(u=0, r=a)$ of the material distribution, the surface variables defined as:

$$A \equiv \frac{a}{m(0)} \quad M \equiv \frac{m_a}{m(0)} \quad \Omega \equiv \frac{1}{1 - \omega_a}. \quad (37)$$

By distinguishing with a circle at the top of the variable $\left(\overset{\circ}{A}\right)$, the derivation with respect to u and with a point $\left(\dot{A}\right)$ the derivative with respect to the normalized time variable $\left(\frac{u}{m(0)}\right)$, we can write the equation (36) as follows

$$\dot{A} = \left(1 - \frac{2M}{A}\right)(\Omega - 1), \quad (38)$$

This equation is valid for all models.

- Continuing with (2b), when evaluating the equation (32) in $r = a$, it is necessary to calculate the derivative of the density and the effective pressure with respect to r ; condition that makes it dependent on the model.

$$\begin{aligned} 2m_{,0}|_a & \left\{ 1 - \frac{m}{r} + r\beta_{,1} \left(1 - \frac{2m}{r} \right) - m_{,1} \right\}_a \\ & - r \left(1 - \frac{2m}{r} \right) \frac{\partial}{\partial u} (4\pi r^2 \rho) \Big|_a \\ & = \Omega(\Omega - 1) \left[4\pi r^2 \left(1 - \frac{2m}{r} \right)^2 r \frac{\partial}{\partial r} (\rho + P) \right]_a. \end{aligned} \quad (39)$$

- Finally, with the condition set in (2c), we evaluate the expression of the trace of the field equations (28) in $r = a$ to obtain another equation for the surface variables. Like the above equation, it is also dependent on the model.

$$\begin{aligned} 4\pi(P - \rho)_a &= \frac{1}{a} [3\beta_{,1}(1 - 2m_{,1}) - m_{,11}]_a \\ & - \frac{2m_{,1}}{r^2} \Big|_a - 2\beta_{,01} \Big|_a + \\ & + \left(1 - \frac{2M}{A} \right) \left(2\beta_{,11} + 4\beta_{,1}^2 + \frac{\beta_{,1}}{r} \right) \Big|_a. \end{aligned} \quad (40)$$

In terms of surface variables, Eqs. (39) and (40) are similar and can be written as:

$$\Upsilon_M \dot{A} + \Xi_M \dot{M} + \Lambda_M \dot{\Omega} = \Delta_M, \quad (41)$$

$$\Upsilon_\Omega \dot{A} + \Xi_\Omega \dot{M} + \Lambda_\Omega \dot{\Omega} = \Delta_\Omega, \quad (42)$$

where

$$\begin{aligned} \Upsilon_\xi &\equiv \Upsilon_\xi(A, M, \Omega), \quad \Xi_\xi \equiv \Xi_\xi(A, M, \Omega) \\ \Lambda_\xi &\equiv \Lambda_\xi(A, M, \Omega), \quad \Delta_\xi \equiv \Delta_\xi(A, M, \Omega) \quad \forall \xi \in \{M, \Omega\}, \end{aligned}$$

are functions of (A, M, Ω) . As stated in Sect. 3, the set of three equations (36), (39) and (40) make up a system of (ODE), which with an initial data set can be numerically integrated and completely determined m and β . In the following Sect. 4, the application of this algorithm for a matter distribution with a Tolman V (EOS) is shown.

4. The Tolman V-like model

In this section, we explicitly present an example that illustrates the method described in Sect. 3. It is necessary to establish the EOS to use Eqs. (33) and (34) and calculate the radial dependence of $m(r)$ and $\beta(r)$.

Equations of state The model is obtained specifically from Tolman's solution V [34] and taking this static solution as a starting point, we can write

$$8\pi r^2 \rho = \frac{3}{7}h + \frac{10z}{3} \left(\frac{r}{a}\right)^{\frac{7}{3}}, \quad (43)$$

$$8\pi r^2 P = \frac{1}{7}h - 2z \left(\frac{r}{a}\right)^{\frac{7}{3}}. \quad (44)$$

The boundary condition The boundary condition ($\bar{P}_a = 0$ or its equivalent $P_a = -\omega_a \rho_a$) allows us to set the values of $z = \frac{1}{7}(4\Omega - 3) \cdot \left(\frac{2M}{A}\right)$ and $h = -2(2\Omega - 5) \cdot \left(\frac{2M}{3A}\right)$, as a function of the surface variables (A, M, Ω) . The equations of state for density and effective pressure can be written

$$4\pi r^2 \rho = \left(\frac{2M}{3A}\right)(2\Omega - 5) + \frac{5}{3} \left(\frac{2m}{r}\right),$$

$$4\pi r^2 P = \left(\frac{2M}{3A}\right)(5 - 2\Omega) - \left(\frac{2m}{r}\right).$$

Surface equations Following the point 2 of Sect. 3, we have the surface equation for the variable A in (38) and in evaluating the equations (29) and (32) at $r = a$,

$$\dot{A} = \left(1 - \frac{2M}{A}\right)(\Omega - 1), \quad (45)$$

$$\dot{M} = -\frac{M}{6A} \left(1 - \frac{2M}{A}\right) \left[3(4\Omega - 3) - 2\left(\frac{2M}{A}\right)(4\Omega - 1)\right], \quad (46)$$

$$\dot{\Omega} = \left[1 - \frac{2}{3}(2\Omega - 3)^2\right] \times \frac{\dot{A}}{2A} - \left(\frac{M}{A} + 2\Omega - 3\right) \frac{\dot{M}}{2M(1 - \frac{2M}{A})}, \quad (47)$$

when you perform numerical integration of (45), (46) and (47), with an initial data set, you have the values of the metric functions β and m as

$$\beta = -\frac{1}{7 + 2(2\Omega - 5) \cdot \left(\frac{2M}{A}\right)} \cdot \left[2(2\Omega - 5) \left(\frac{2M}{3A}\right) \log\left(\frac{r}{a}\right) + \log\left(\frac{1 - \frac{2M}{A} \left\{1 + \frac{(4\Omega - 3)}{7} \left[\left(\frac{r}{a}\right)^{\frac{7}{3}} - 1\right]\right\}}{1 - \frac{2M}{A}}\right)\right],$$

$$m = m(0) \left(\frac{r}{a}\right) M \left\{1 + \frac{(4\Omega - 3)}{7} \left[\left(\frac{r}{a}\right)^{\frac{7}{3}} - 1\right]\right\}.$$

Integrating numerically Eqs. (45), (46) and (47), with an initial data set, the surface variables thus determined can be used to calculate the values of the physical variables with the procedure established in (4) of Sect. (3), in the different layers of the material distribution. In the graphs of the

surface variables (A, M, Ω) and the physical variables (pressure, density, radiation, etc.), an exponential behavior (dispersion) or damped oscillations is observed. The dispersive or damped form of the temporal behavior of the surface or physical variables depends a lot on the set of initial values used in the simulation.

5. Results and comments

In Fig. 1, it can be observed, by keeping fixed the values of $M = 1.0$ and $\Omega = 0.87$, that for value of the initial radius of the distribution $A = 3.1$, there is an exponential growth in the temporary evolution and damped oscillations are presented for the values of $A = 3.8$ and $A = 8.0$. This analogous damped oscillatory or exponential growth behavior has been described, for example, in [15, 20] using the EOS of the ultrarelativistic fluid (2), and established by Eq. (3). Here, it has been used as EOS, Tolman V and the algorithm set out in the Sect. 3. Consequently, we can show an oscillatory behavior described by

$$s(u) = s_0 + s_M e^{-\left(\frac{u}{T}\right)} \cos\left[2\pi\left(\frac{u}{T}\right) + \varphi\right], \quad (48)$$

where $s(u)$ and s_0 and s_M correspond to the values of the physical variable studied, its average value and maximum value. The values of $u \rightarrow \frac{u}{m(0)}$, $\tau \rightarrow \frac{\tau}{m(0)}$ and $T \rightarrow \frac{T}{m(0)}$ are normalized with respect to the initial mass.

Fig. 2 compares the curve obtained by numerical integration with the adjustment coefficients for Eq. (48) where they have been used as the initial data set ($A = 5.5, M = 1.0, \Omega = 0.87$). Table 1 shows the values obtained in the curve fitting and the values of the pseudoperiod and the growth rate (decrease, depending on the case) can be seen. It should be noted that in previous works such as [35] or [36] where Tolman V EOS was used as a static solution, no self-similar behavior was reported perhaps because the integration of the system of surface equations was limited to a short interval of the temporal variable. For this reason, the simulation of the value $A = 8.0$ in Fig. 1 is performed to show that with initial data similar to those used in [35, 36], the damped oscillatory behavior is obtained.

As expected, this damped oscillatory behavior extends to the different physical variables that were calculated and therefore this behavior is repeated for all the normalized variables; density (Fig. 3), pressure (Fig. 4) and radiation (Fig. 5), where they were used as initial values of the surface variables a ($A = 5; M = 1.0; \Omega = 0.87$). Fig. 6 highlights the small difference between the curve obtained by numerical integration and the adjustment of the evolution of the normalized density at the edge of the distribution of material to be considered. From the data obtained in

Fig. 1 Evolution of the radius A , depending on the temporal variation normalized $\frac{u}{m(0)}$ for three different values $A = 3.1, 3.8$ and $A = 8.0$. The initial values of $M = 1$ and $\Omega = 0.87$ remain constant in the three graphs. Note that for small values of the radius A a continuous expansion (explosion) occurs. However, for less compact spheres, ($A = 3.8, 8.0$), damped oscillations tend to reach a static equilibrium. The temporal evolution of the radio A is in accordance with Choptuik [17] and Gundlach [20]

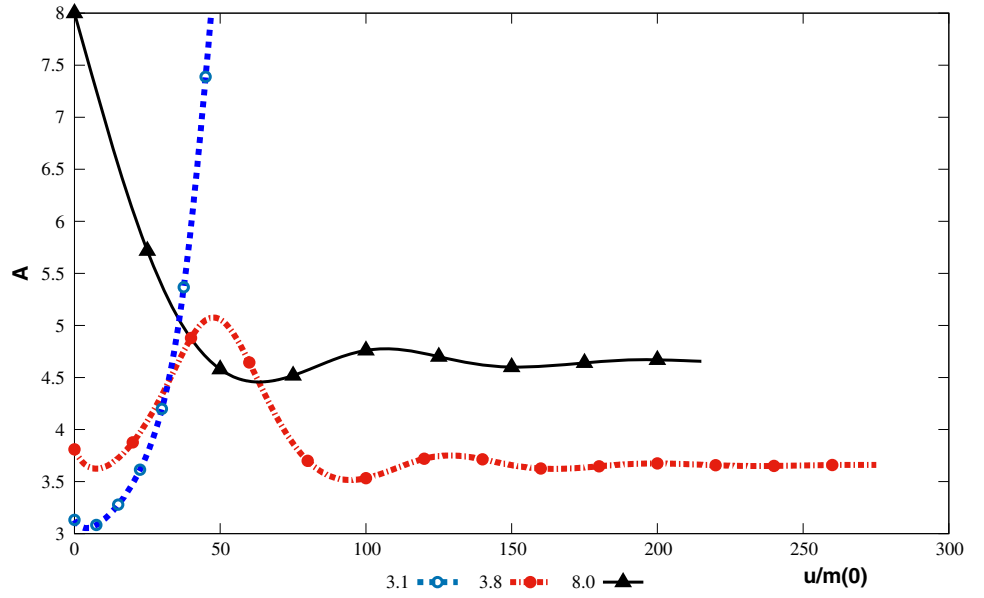


Fig. 2 The radius A as a function of the normalized time like coordinate $\frac{u}{m(0)}$ in the Tolman V model. Initial values for the surface variables $A = 5.5; M = 1.0; \Omega = 0.87$. The solid line is the numerical result, while the dashed line is a fit based on (48)

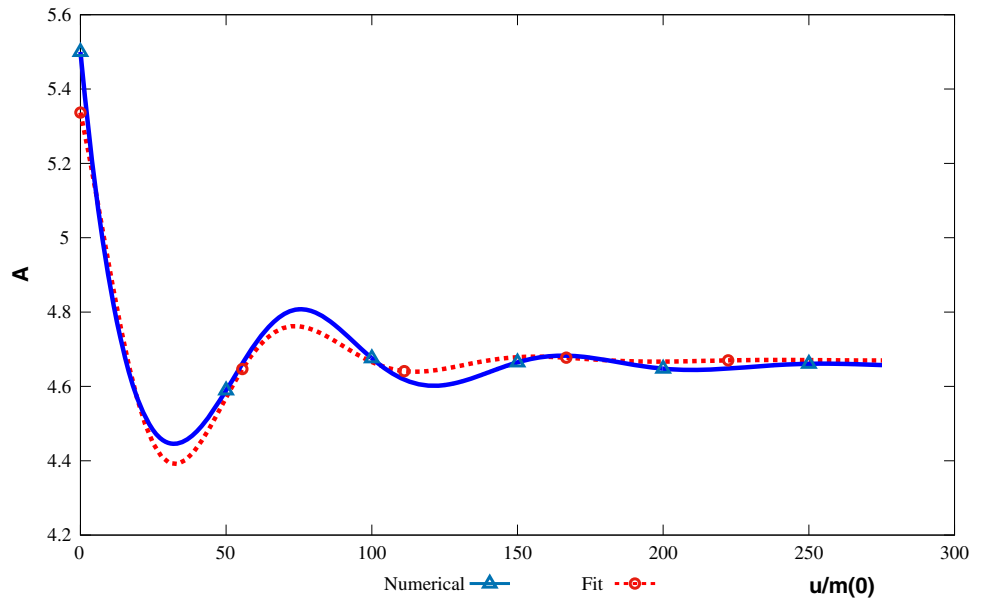


Table 1 Values corresponding to Eq. (48) that fix the data obtained for the normalized radius $A\left(\frac{u}{m(0)}\right)$ from the initial values $A(0) = 5.5, M(0) = 1.0, \Omega(0) = 0.87$

Final set of parameters	Asymptotic standard error	
$A_0 = 4.67003$	± 0.0005901	(0.01264)%
$\frac{\tau}{m(0)} = 37.311$	± 0.3354	(0.899)%
$\varphi = -0.336433$	± 0.007923	(2.355)%
$\frac{T}{m(0)} = 82.5767$	± 0.3025	(0.3664)%
$A_m = 0.706386$	± 0.005374	(0.3664)%

the adjustment (see Table 2), it is concluded that the pseudoperiod values between the variable of the distribution radius and the normalized density are very similar.

In relation to Figs. 7 and 8, the value of $\kappa = \frac{\bar{p}}{\bar{\rho}}$ is defined for different layers of the material and its temporal evolution. We have that this value is bounded in $\kappa \approx 0.35 \leq 0.49$, for the case of Fig. 7 that corresponds to the solution V of Tolman. Its behavior is similar to that observed for the case of the radiant fluid studied by [16] since for large values of density (or pressure), the second term on the right side of (4) becomes very small, coinciding with the limit with the expression for the radiant fluid. With respect to Fig.8, which corresponds to the

Fig. 3 Density $8\pi m(0)^2 \bar{\rho}$ in function of the temporal variable $\frac{u}{m(0)}$ for the model of Tolman V, with initial values of the surface variables $A = 5$; $M = 1.0$; $\Omega = 0.87$, for $\frac{r}{a} = 0.20, 0.6$, and 1.00

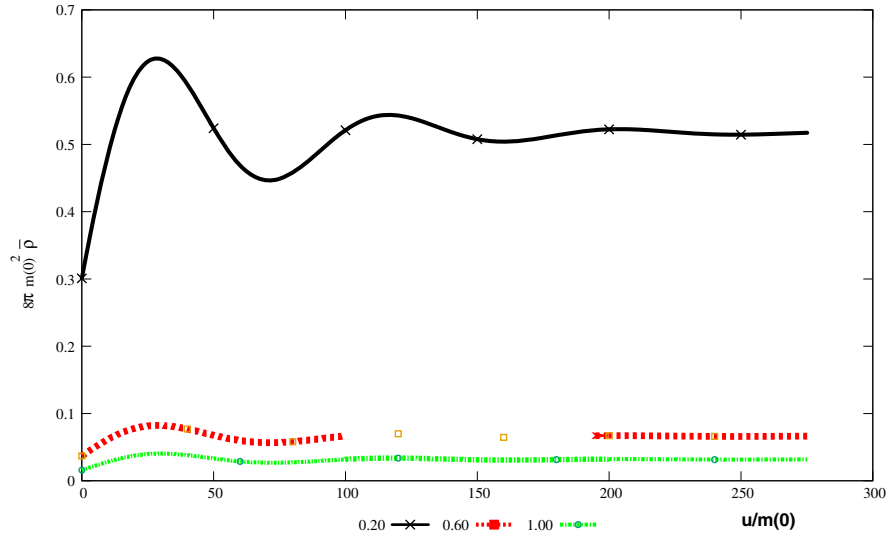
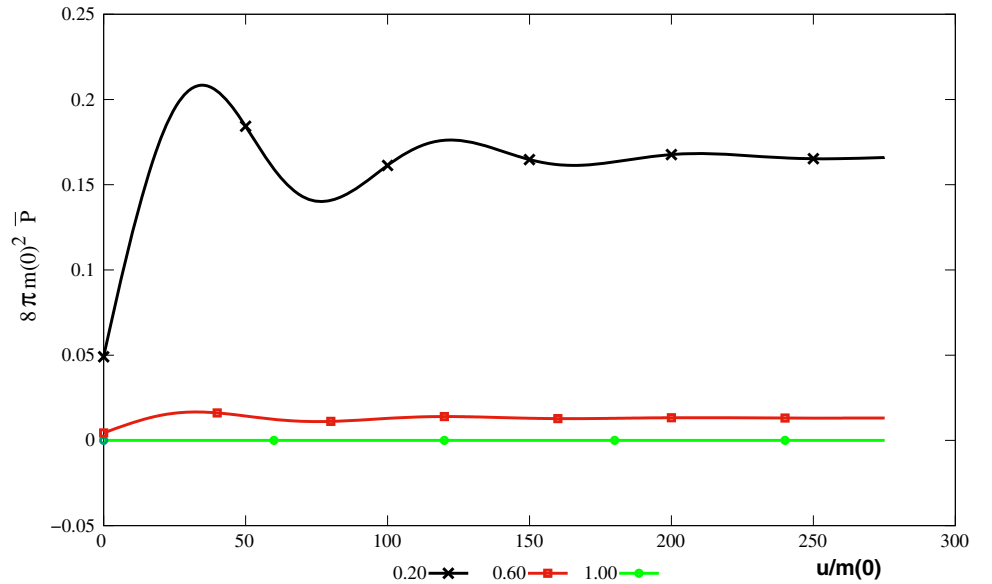


Fig. 4 Normalized pressure values $8\pi m(0)^2 \bar{P}$ in function of the time variable $\frac{u}{m(0)}$ for the model of Tolman V, with surface variables $A = 5$; $M = 1.0$; $\Omega = 0.87$, for $\frac{r}{a} = 0.20, 0.6, 0$ and 1.00



solution Tolman VI (taken from [27]) shows that there is a non-constant relationship between the pressure $\bar{P}$ and the density $\bar{\rho}$. Therefore, κ is not a constant value and will have a different behavior to the radiant fluid (2) or to the limit case of the solution V of Tolman.

If we observe some of the numerical solutions obtained for this model (Tolman V), we have that several initial values conform to the behavior established by Eq. (48). However, not all initial values present such a good fit. It is possible that this anomaly is due to the fact that we have taken a simplification of Eq. (30) which leads us to (32). Therefore, it is possible that a new approach, that is, considering all the terms of (30), allows us to find a solution similar to the one shown in [15, 16, 20, 48], where Choptuik spacetime disturbances can be described by functions of the form

$$s(u) \propto e^{\lambda u} \cos(\omega_p u + \varphi) \cdot g(u), \quad (49)$$

where λ determines the rate of growth of the disturbance or decay. The function $g(u)$ is a function with the period Δ of the discretely self-similar background solution.

6. Conclusions

In this article, numerical simulations were performed to study the behavior of a perfect fluid with spherical symmetry using the scheme previously established in [27]. This algorithm is based on the fact that the solutions of the Einstein field equations have the characteristic of self-similarity and allow the PDE equations to be transformed into another equivalent ODE. Additionally, it is necessary,

Fig. 5 Radiation profiles $8\pi m(0)^2 \varepsilon$ for the model of Tolman V, for initials values ($A = 5.0, M = 1.0, \Omega = 0.87$). All layers emit radiation but decrease until reaching a state of equilibrium for $\Omega = 1$. The layers to be plotted $\frac{r}{a} = 0.20, 0.6$, and 1.00

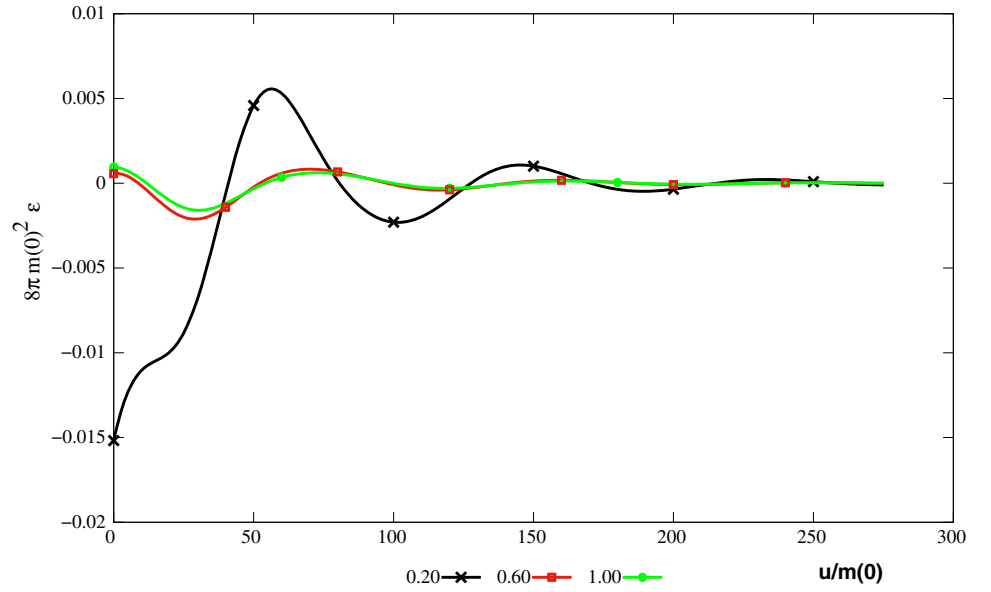
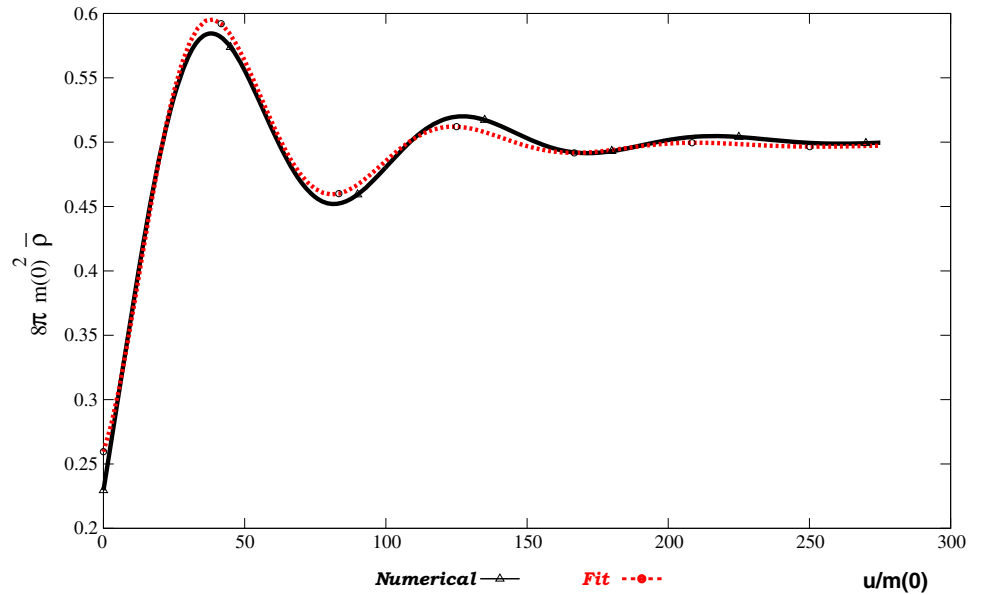


Fig. 6 Density $8\pi m(0)^2 \bar{\rho}$ as function of the time-like variable $\frac{u}{m(0)}$ in a TolmannV-like model. They were used as initial values of the surface variables $A = 5.5; M = 1.0; \Omega = 0.87$. The outermost layer was used for the numerical value of the integration with $\frac{r}{a} = 1$. Table 2 shows the different values of the parameters of the equation (48) (Fit dashed line) that adjust the numerical values (Numerical solid line)



together with the conservation equations (EEE), to establish an EOS to complete the simulation. In almost all the numerical simulations that make up the PDE obtained from EEE, the EOS used is that of the ultra-relativistic fluid (2). In this work, the parametric version of Tolman's V solution has been used as EOS; (43)–(44). Once the numerical integration of the surface variables (37) has been carried out, some physical variables (radius, pressure, density, radiation, etc.) can be calculated using the field equations (24)–(27), in several layers of the spherical distribution of the fluid. This allows a set of graphs to be made that show the temporal evolution of each physical variable. Figs. 1, 2, 3, 4, 5 and 6 show this behavior of some physical variables, and it is observed that their evolution is

well described by oscillations that are dampened or increased exponentially, according to the equation (48), in very good qualitative agreement with the results of [15, 20]. It is difficult to make a comparison between the numerical data obtained between the two numerical simulations since the expression for the Tolman solution equation V does not present a constant value for what can be assimilated as a value of κ . To compare these results, it is necessary to perform the same numerical simulation, but instead of using Tolman V; we must use a parametric expression of the EOS for the radiant fluid (2). Despite this difficulty, in Fig. 7 it can be seen that the relationship $\frac{\bar{p}}{\bar{\rho}}$ remains limited throughout its evolution and there is a behavior contrary to what is seen in Fig. 8 for the Tolman

Fig. 7 Ratio $\kappa = \frac{\bar{p}}{\bar{\rho}}$ as function of the time-like variable $\frac{u}{m(0)}$ in a Tolman V-like model. ($A = 5.8, M = 1.0, \Omega = 0.87$) with $\frac{\tau}{a} = 0.2, 0.6$, and 1.00

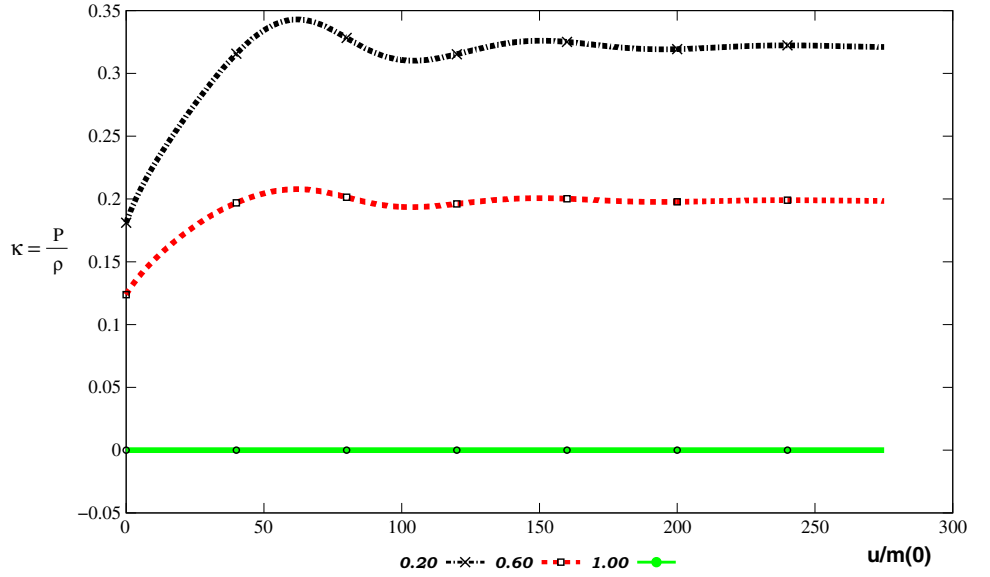


Fig. 8 Ratio $\kappa = \frac{\bar{p}}{\bar{\rho}}$ as function of the time-like variable $\frac{u}{m(0)}$ in a Tolman VI-like model with the initial value ($A = 6.0, M = 1.0, \Omega = 0.87$). For the layers in contraction $\frac{\tau}{a} = 0.2, 0.6$, and 1.00

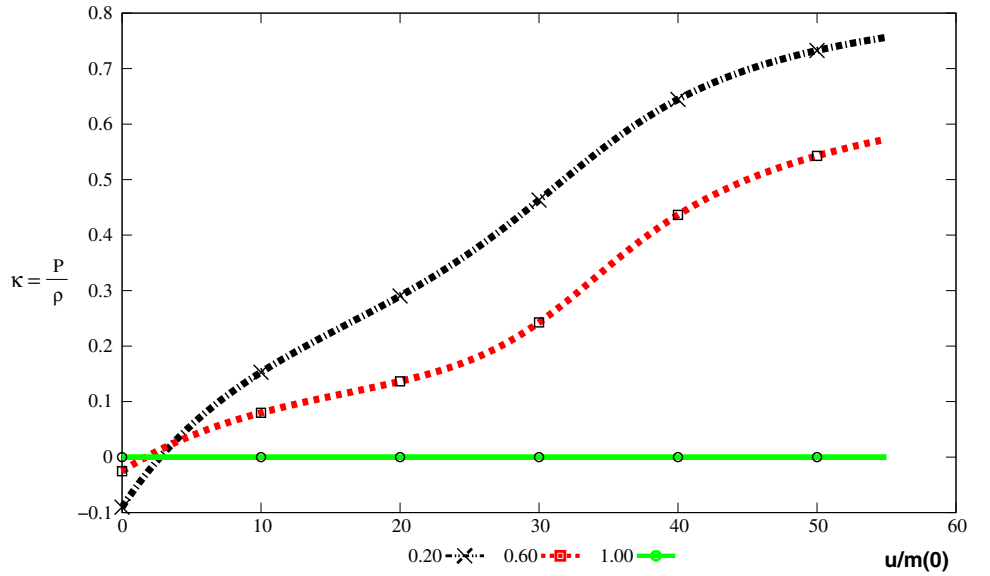


Table 2 Values corresponding to Eq. (48) that set the data obtained for the normalized density $8\pi \cdot m(0)^2 \bar{\rho}$ from the initial values $A = 5.5, M = 1.0, \Omega = 0.87$

Final set of parameters	Asymptotic standard error	
$\rho_0 = 0.497388$	± 0.0001204	(0.02421)%
$\frac{\tau}{m(0)} = 45.2002$	± 0.2073	(0.4585)%
$\varphi = -6.33699$	± 0.004595	(0.07251)%
$\frac{T}{m(0)} = 85.7544$	± 0.1495	(0.1744)%
$\rho_m = -0.238208$	± 0.000818	(0.3434)%

VI EOS case of [27]. With the possibility of changing the EOS, we can extend the present study and perhaps find a relationship between the initial values of the surface

variables (A, M, Ω) and the values of the pseudoperiod T , the phase φ and the relaxation time τ in Eq. (48). Furthermore, it is necessary to include all the terms of Eq. (30) to verify if this condition is sufficient to completely determine the function $g(u)$ and the period Δ of the expression (49). It is hoped that this latest result can be shown in a future publication.

References

- [1] P M Pizzochero arXiv e-prints [arXiv:1001.1272](https://arxiv.org/abs/1001.1272) (2010)
- [2] G Baym, T Hatsuda, T Kojo, P D Powell, Y Song and T Takatsuka *Rep. Prog. Phys.* **81** 056902 (2018)

- [3] J Bicak *ArXiv General Relativity and Quantum Cosmology e-prints* (2006)
- [4] L Blanchet *Living Rev. Relativ.* **17** 2 (2014)
- [5] T Nakamura, K Oohara, and Y Kojima *Prog. Theor. Phys. Suppl.* **90** 1 (1987)
- [6] R Gomez, J Winicour, and R Isaacson *J. Comput. Phys.* **98** 11 (1992)
- [7] R Gómez and J Winicour *J. Math. Phys.* **33** 1445 (1992)
- [8] R Gómez and J Winicour *Phys. Rev. D* **45** 2776 (1992)
- [9] T W Baumgarte and S L Shapiro *Phys. Rev. D* **59** 024007 (1998)
- [10] P Grandclément and J Novak *Living Rev. Relativ.* **12** 1 (2009)
- [11] T W Baumgarte, P J Montero, I Cordero-Carrión and E Müller *Phys. Rev. D* **87** 044026 (2013)
- [12] T Baumgarte in *APS Meeting Abstracts* (2014)
- [13] T W Baumgarte, P J Montero and E Müller *Phys. Rev. D* **91** 064035 (2015)
- [14] T W Baumgarte and P J Montero *Phys. Rev. D* **92** 124065 (2015)
- [15] J Celestino and T W Baumgarte *Phys. Rev. D* **98** 024053 (2018)
- [16] T W Baumgarte *ArXiv e-prints* (2018)
- [17] M W Choptuik, *Phys. Rev. Lett.* **70** 9 (1993)
- [18] C Gundlach *arXiv e-prints* [arXiv:gr-qc/9712084](https://arxiv.org/abs/gr-qc/9712084) (1997)
- [19] A Wang *Braz. J. Phys.* **31** 188 (2001)
- [20] C Gundlach *Phys. Rev. D* **65** 084021 (2002)
- [21] D Garfinkle *Rep. Prog. Phys.* **80** 016901 (2017)
- [22] C Gundlach *Phys. Rev. D* **65**, 064019 (2002)
- [23] A Koutras and J E F Skea, *Comput. Phys. Commun.* **115** 350 (1998)
- [24] B J Carr and A A Coley, *Class. Quantum Gravity* **16** R31 (1999)
- [25] B J Carr and A Koutras *Astrophys. J.* **405** 34 (1993)
- [26] B J Carr and A A Coley *Gen. Relativ. Gravit.* **37** 2165 (2005)
- [27] V Medina and N Falcon, *Prog. Phys.* **14** 46 (2018)
- [28] G I Barenblatt and Y B Zel'dovich *Annu. Rev. Fluid Mech.* **4** 285 (1972)
- [29] K Tomita *Prog. Theor. Phys.* **66** 2025 (1981)
- [30] C R Evans and J S Coleman *Phys. Rev. Lett.* **72** 1782 (1994)
- [31] T Koike, T Hara and S Adachi *Phys. Rev. Lett.* **74** 5170 (1995)
- [32] S M Wagh and K S Govinder *Gen. Relativ. Gravit.* **38** 1253 (2006)
- [33] S Banerjee *Pramana* **91** 27 (2018)
- [34] R C Tolman *Phys. Rev.* **55** 364 (1939)
- [35] A Patiño and H Rago *Lett. Nuovo Cimento* **38** 321 (1983)
- [36] K D Krori, P Borgohain and R Sarma *Phys. Rev. D* **31** 734 (1985).
- [37] J P de Leon *Gen. Relativ. Gravit.* **25** 865 (1993)
- [38] L Andersson and A Y Bertscher *Annales Henri Poincaré* **20** 813 (2019)
- [39] J D Bekenstein, *Phys. Rev. D* **4** 2185 (1971).
- [40] A Patiño and H Rago *Astrophys. Space Sci.* **257** 213 (1997)
- [41] L Herrera, J Jiménez, and G J Ruggeri *Phys. Rev. D* **22** 2305 (1980)
- [42] M Cosenza, L Herrera, M Esculpi and L Witten *Phys. Rev. D* **25** 2527 (1982)
- [43] V Medina, L Núñez, H Rago and A Patiño, *Can. J. Phys.* **66** 981 (1988)
- [44] W Barreto and A da Silva, *Gen. Relativ. Gravit.* **28** 735 (1996)
- [45] A di Prisco, N Falcón, L Herrera, M Esculpi and N O Santos, *Gen. Relativ. Gravit.* **29** 1391 (1997)
- [46] W Barreto, B Rodríguez and H Martínez *Astrophys. Space Sci.* **282** 581 (2002)
- [47] L Herrera, W Barreto, A di Prisco and N O Santos *Phys. Rev. D* **65** 104004 (2002)
- [48] J M Martín-García and C Gundlach *Phys. Rev. D* **59** 064031 (1999)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Relativistic Model for Radiating Spherical Collapse

Victor Medina¹ and Nelson Falcon²

¹Departamento de Física - Facultad de Ingeniería — Universidad de Carabobo. Valencia, Venezuela
E-mail: vmedina@uc.edu.ve

²Departamento de Física — Facultad Experimental de Ciencias y Tecnología - Universidad de Carabobo. Valencia, Venezuela
E-mail: nelsonfalconv@gmail.com

The relativistic models for radiating spherical collapse is important for to explain the emission process on very high energy in Supernova burst and Quasars. A general method is reviewed, to obtain models which describe non static radiating spheres, without having to make any hypothesis about the emission of radiation during the collapse. It is concluded that the field equations together with the conservation laws (Bianchi's Identity) form a complete set of integrable equations that do not require additional the emissivation hypothesis of a Gaussian pulse on at an arbitrary instant to trigger the collapse. The emissivation hypothesis of a Gaussian pulse is not only unnecessary, but also leads to qualitatively and quantitatively different solutions. Calculations were performed using the computer algebra package *GRTensorII*, running on *Maple 13*, along with several Maple routines that we have used specifically for this type of problems. The Schwarzschild and Tolman VI models are shown as examples where it's emphasizes the importance of using conservation equations properly, for describe the collapse for the self-gravitating sphere.

1 Introduction

The last phases of stellar evolution of massive stars are dominated by the contribution of stellar radiation due to changes of the inner or outer distribution of matter, in the gravitational potential of the radiating fluid spheres and, therefore, general relativity provides a description of the collapse of the compact objects (Neutron Stars, Black Holes). This description can be extended to explain the radiation process of very high energy in astrophysical scenarios, such as Supernova bursts and Quasars. A number of studies have been reported describing a gravitational collapse: Oppenheimer and Snyder [1], Tolman [2] and furthermore the study of the collapsing radiating fluid [3–6].

This scheme has recently been used for various scenarios of relativistic hydrodynamics. We can highlight some examples: charged fluids [7–9], isotropic [10] or anisotropic fluid [11, 12], shock waves [13, 14], in free space [15, 16] or diffusion process [17, 18]. It is necessary to contrast its quantitative results with other calculation schemes. Barreto et al. [19] have extended the semi-numerical scheme to the Schwarzschild coordinates, simulating some scenarios of the gravitational collapse.

Herrera and collaborators [6, 19–21] developed a general algorithm for modeling self gravitating spheres out of equilibrium, beginning from the known static solutions of Einstein's equations. This method divides the space-time in two spatial regions. The outer region is described by the Vaidya solution and the space-time metric in the interior is obtained by solving the Einstein field equations. Further, proper boundary conditions are imposed in order to guarantee a smooth matching of the solutions in the surface of the junction. This semi numerical technique has been used extensively to study high

energy in astrophysical scenarios [19, 21–27].

However in these numerical simulations a Gaussian pulse is introduced *ad hoc* to represent the emission of radiation that initiates the disequilibrium during the collapse of the radiating fluid ball [6, 20, 21, 23–25, 28]. These assumptions could be unnecessary and generate spurious solutions, since this loss of mass is prescribed by one of the conservation equations when applying the Bianchi Identity [29, 30]. Parts of the calculation of the Bianchi identities that were performed in this work were possible and verified using the *GRTensorII* package.

The purpose of this paper is to show the general method to obtain models which describe radiating non-static spheres without having to make any hypothesis about the emission of radiation during the collapse. This paper follows as much as possible the notation and physical description prescribed by Herrera et al. [6]. For this, the field equations and conservation laws are shown in Section 2; then section 3 establishes the procedure for the static solutions and obtaining the surface equations. The models Schwarzschild-like and Tolman VI-like are discussed in section 4 and 5 respectively, and in the last section are shown the concluding remarks.

2 The Field equations and conservation relationships

Let us consider a non static radiating spheres. The metric takes the form [4]

$$ds^2 = e^{2\beta} \frac{V}{r} du^2 + 2e^{2\beta} du dr - r^2 d\theta^2 - r^2 \sin^2 \theta d\phi^2, \quad (1)$$

where u and r are time like and radial-like coordinates respectively; β and V are functions of u and r ; θ, ϕ are the usual angle coordinates. In these coordinates the gravitational field equations are:

$$\begin{aligned}
-8\pi T_{00} &= -\frac{V_{,0} - 2V\beta_{,0}}{r^2} - \frac{V}{r^3} (e^{2\beta} - V_{,1} + 2\beta_{,1}V) \\
-8\pi T_{01} &= -\frac{1}{r^2} (e^{2\beta} - V_{,1} + 2\beta_{,1}V) \\
-8\pi T_{11} &= -\frac{4\beta_{,1}}{r} \\
-8\pi T_2^2 = -8\pi T_3^3 &= -e^{2\beta} \left(2\beta_{01} - \frac{1}{2r^2} [rV_{,11} + \right. \\
&\quad \left. - 2\beta_{,1}V + 2r(\beta_{,11}V + \beta_{,1}V_{,1})] \right).
\end{aligned}$$

As usual, note that we used the subscript $_{,0}$ and $_{,1}$ for the derivative for u and r , respectively; and the semicolon (;) for covariant differentiation. Then transformation relations between local Minkowskian and radiative coordinates are:

$$\begin{aligned}
dt &= \left(\frac{\partial t}{\partial u} \right) du + \left(\frac{\partial t}{\partial r} \right) dr \\
&= e^\beta \left(\frac{V}{r} \right)^{\frac{1}{2}} du + e^\beta \left(\frac{r}{V} \right)^{\frac{1}{2}} dr \quad (2)
\end{aligned}$$

$$dx = \left(\frac{\partial x}{\partial r} \right) dr = e^\beta \left(\frac{r}{V} \right)^{\frac{1}{2}} dr \quad (3)$$

$$dy = \left(\frac{\partial y}{\partial \theta} \right) d\theta = r d\theta \quad (4)$$

$$dz = \left(\frac{\partial z}{\partial \phi} \right) d\phi = r \cdot \sin \theta d\phi. \quad (5)$$

We assumed the stellar material as perfect fluid, with energy density $\hat{\rho}$, radial pressure $\hat{P}$, without heat conduction neither viscosity, then

$$\hat{T}_{\alpha\beta} = (\hat{\rho} + \hat{P}) \cdot U_\alpha U_\beta - \hat{P} \cdot \eta_{\alpha\beta}, \quad (6)$$

where $U_\alpha = (1, 0, 0, 0)$, $3\hat{\sigma}$ is the isotropic radiation of the energy density, and $\hat{\varepsilon}$ no-polarized component of the energy density in radial direction. Now consider an observer in local Minkowskian system with radial velocity ω , in the Lorentzian system we can write:

$$\bar{T}_{\mu\nu} = \Lambda_\mu^\alpha \Lambda_\nu^\beta \hat{T}_{\alpha\beta}, \quad (7)$$

where the Lorentz matrix is

$$\Lambda_\mu^\alpha = \begin{bmatrix} \frac{1}{\sqrt{1-\omega^2}} & -\frac{\omega}{\sqrt{1-\omega^2}} & 0 & 0 \\ -\frac{\omega}{\sqrt{1-\omega^2}} & \frac{1}{\sqrt{1-\omega^2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (8)$$

We define

$$\bar{\rho} = \hat{\rho} + 3\hat{\sigma}, \quad \bar{P} = \hat{P} + \hat{\sigma}, \quad \bar{\varepsilon} = \hat{\varepsilon} \frac{1+\omega}{1-\omega}.$$

Note also that from (2-3) the velocity of matter in the radiative coordinates is given by

$$\frac{dr}{du} = \frac{V}{r} \cdot \frac{\omega}{1-\omega}, \quad (9)$$

so forth the energy-impulse tensor in the Lorentz system is

$$\begin{aligned}
\bar{T}_{00} &= \bar{\varepsilon} + \frac{\bar{\rho} + \omega^2 \bar{P}}{1-\omega^2} \\
\bar{T}_{01} = \bar{T}_{10} &= -\bar{\varepsilon} - \frac{\omega}{1-\omega^2} (\bar{\rho} + \bar{P}) \\
\bar{T}_{11} &= \frac{\bar{P} + \omega^2 \bar{\rho}}{1-\omega^2} + \bar{\varepsilon} \\
\bar{T}_{22} = \bar{T}_{33} &= \bar{P}.
\end{aligned}$$

Using (2) - (5) we obtain the energy-impulse tensor in radiative coordinates as:

$$\begin{aligned}
T_{00} &= e^{2\beta} \left(\frac{V}{r} \right) \left(\frac{\bar{\rho} + \omega^2 \bar{P}}{1-\omega^2} + \bar{\varepsilon} \right) \\
T_{01} = T_{10} &= e^{2\beta} \left(\frac{\bar{\rho} - \omega \bar{P}}{1+\omega} \right) \\
T_{11} &= e^{2\beta} \left(\frac{r}{V} \right) \left(\frac{1-\omega}{1+\omega} \right) (\bar{\rho} + \bar{P}) \\
T_{22} = \frac{T_{33}}{\sin^2 \theta} &= r^2 \bar{P}.
\end{aligned}$$

Remember that a bar indicates that the quantity is measured in the Lorentzian system, and the effective variables are written without bar. Now

$$\rho \equiv \frac{\bar{\rho} - \omega \bar{P}}{1+\omega}, \quad P \equiv \frac{\bar{P} - \omega \bar{\rho}}{1+\omega}, \quad \varepsilon \equiv \bar{\varepsilon}. \quad (10)$$

It can be seen at once that $\rho = \bar{\rho}$ and $P = \bar{P}$ in $r = 0$, also, in the static case $\omega = 0$. As before then:

$$\begin{aligned}
T_{00} &= e^{2\beta} \left(\frac{V}{r} \right) \left[\frac{\omega(\rho + P)}{(1-\omega)^2} + \rho + \varepsilon \right] \\
T_{01} = T_{10} &= e^{2\beta} \rho \\
T_{11} &= e^{2\beta} \left(\frac{r}{V} \right) (\rho + P) \\
T_{22} = \frac{T_{33}}{\sin^2 \theta} &= r^2 \bar{P},
\end{aligned}$$

thus the field equations are:

$$\begin{aligned}
-\frac{V}{r^2} \left[\left(2\beta_{,0} - \frac{V_{,0}}{V} \right) - \frac{1}{r} (2V\beta_{,1} - V_{,1} + e^{2\beta}) \right] &= \\
= 8\pi e^{2\beta} \left(\frac{V}{r} \right) \left[\varepsilon + \rho + \frac{\omega(\rho + P)}{(\omega - 1)^2} \right] & \quad (11)
\end{aligned}$$

$$2V\beta_{,1} - V_{,1} + e^{2\beta} = 8\pi r^2 e^{2\beta} \rho \quad (12)$$

$$\frac{4\beta_{,1}}{r} = 8\pi \frac{r}{V} e^{2\beta} (\rho + P) \quad (13)$$

$$\begin{aligned} -2\beta_{,01} + \frac{1}{2r^2} [rV_{,11} - 2\beta_{,1}V + 2r(V_{,1}\beta_{,1} + V\beta_{,11})] = \\ = 8\pi e^{2\beta} \bar{P}. \end{aligned} \quad (14)$$

Using the conservation equations (Bianchi Identity) $T^\mu_{\nu;\mu} = 0$, we obtain only three no-trivial relations:

$$\begin{aligned} T^\mu_{1;\mu} = \frac{e^{-2\beta}}{2\pi r} \beta_{,10} - \frac{\partial P}{\partial r} + \\ - \frac{1}{2} \left(2\beta_{,1} + \frac{V_{,1}}{V} - \frac{1}{r} \right) (\rho + P) - \frac{2(P - \bar{P})}{r} = 0, \end{aligned} \quad (15)$$

$$\begin{aligned} T^\lambda_{0;\lambda} = \frac{V}{r} \left\{ \left(2\beta_{,1} + \frac{V_{,1}}{V} + \frac{1}{r} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \right. \\ \left. + \frac{\omega}{(1 - \omega)^2} \frac{\partial(\rho + P)}{\partial r} \right\} + \frac{1}{2} \left(2\beta_{,0} - \frac{V_{,0}}{V} \right) (\rho + P) + \\ + \frac{\partial \rho}{\partial u} + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} + \frac{1 + \omega}{(1 - \omega)^3} (\rho + P) \frac{\partial \omega}{\partial r} \right] = 0, \end{aligned} \quad (16)$$

$$\begin{aligned} e^{2\beta} T^\lambda_{;\lambda} = \frac{V}{r} \left\{ \left(2\beta_{,1} + \frac{V_{,1}}{V} + \frac{1}{r} \right) \left[\varepsilon + \frac{1 + \omega^2}{2(1 - \omega)^2} (\rho + P) \right] \right. \\ \left. + \frac{\omega(\rho + P)_{,1}}{(1 - \omega)^2} + \frac{\partial P}{\partial r} + \frac{1}{r} [\rho + \bar{P} - (P - \bar{P})] \right\} + \\ - \frac{1}{2} \left(2\beta_{,0} - \frac{V_{,0}}{V} \right) (\rho + P) + \frac{\partial P}{\partial u} + \\ + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} + \frac{1 + \omega}{(1 - \omega)^3} (\rho + P) \frac{\partial \omega}{\partial r} \right] = 0. \end{aligned} \quad (17)$$

It is remarkable that only two Bianchi equations (15-17) are independent, then

$$e^{2\beta} T^\mu_{;\mu} - T^\mu_{0;\mu} = \left(\frac{V}{r} \right) T^\mu_{1;\mu} = 0. \quad (18)$$

If we use the Bondi mass aspect $V \equiv e^{2\beta} (r - 2m)$, after some elementary algebra, the equation system becomes equivalent to:

$$m_{,1} = 4\pi r^2 \rho \quad (19)$$

$$\beta_{,1} = 2\pi r \frac{(\rho + P)}{1 - \frac{2m}{r}} \quad (20)$$

$$m_{,0} = -4\pi r^2 e^{2\beta} \left(1 - \frac{2m}{r} \right) \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] \quad (21)$$

$$\begin{aligned} 8\pi \bar{P} = -2\beta_{,01} e^{-2\beta} + \left(1 - \frac{2m}{r} \right) \left(2\beta_{,11} + 4\beta_{,1}^2 - \frac{\beta_{,1}}{r} \right) + \\ + \frac{1}{r} [3\beta_{,1} (1 - 2m_{,1}) - m_{,11}]. \end{aligned} \quad (22)$$

Also, for two independent Bianchi equations (15) and (16), we obtain:

$$\begin{aligned} -\frac{e^{-2\beta}}{2\pi r} \beta_{,10} + \frac{\partial P}{\partial r} + \\ + \left(4\pi r^2 P + \frac{m}{r} \right) \frac{(\rho + P)}{r \left(1 - \frac{2m}{r} \right)} + \frac{2}{r} (P - \bar{P}) = 0, \end{aligned} \quad (23)$$

$$\begin{aligned} \frac{e^{2\beta}}{r} \left[1 + \left(1 - \frac{2m}{r} \right) + 4\pi r^2 (P - \rho) \right] \left[\varepsilon + \frac{\omega(\rho + P)}{(1 - \omega)^2} \right] + \\ + \frac{V}{r} \frac{\omega}{(1 - \omega)^2} \frac{\partial}{\partial r} (\rho + P) + \\ + \frac{V}{r} \left[\frac{\partial \varepsilon}{\partial r} + \frac{(1 + \omega)(\rho + P)}{(1 - \omega)^3} \frac{\partial \omega}{\partial r} \right] + \frac{\partial \rho}{\partial u} = 0. \end{aligned} \quad (24)$$

The expression (23) is the generalization of the Tolman - Oppenheimer - Volkoff (TOV) equation of hydrostatic equilibrium (see, for example [31]). It can be shown that the conservation equation (24) can also be obtained from the field equations (19) and (21), remembering that the second mixed derivatives commute, that is, $m_{,01} = m_{,10}$. Now, combining (21) with (24) we obtain:

$$\begin{aligned} \frac{2m_{,0} e^{-2\beta}}{4\pi r^3 \left(1 - \frac{2m}{r} \right)} \left[1 - \frac{m}{r} + 2\pi r^2 (P - \rho) \right] = \\ \frac{\omega}{(1 - \omega)^2} \left(1 - \frac{2m}{r} \right) (\rho + P)_{,1} + \\ + \frac{\partial \varepsilon}{\partial r} + \frac{(1 + \omega)(\rho + P)}{(1 - \omega)^3} \frac{\partial \omega}{\partial r} + \frac{e^{-2\beta}}{4\pi r^2} m_{,10}. \end{aligned}$$

If we assume that radiation profiles ε and the variable ω , vary little, so we can write an expression very similar to the Euler equation

$$\begin{aligned} 2m_{,0} \left[1 - \frac{m}{r} + 2\pi r^2 (P - \rho) \right] = \\ \frac{\omega \cdot e^{2\beta}}{(1 - \omega)^2} \left[4\pi r^2 \left(1 - \frac{2m}{r} \right)^2 r (\rho + P)_{,1} \right] + \\ + r \left(1 - \frac{2m}{r} \right) m_{,10}. \end{aligned} \quad (25)$$

Equation (25) is omitted in previous works on the evolution of radiating fluid sphere [6, 20, 21, 23, 28]. This omission prevents the closing of the system of equations, and motivates the spurious inclusion of a luminosity Gaussian pulse [21, 23–25, 28]. Equation (25) allows us relate the mass exchange with the time like and radial derivatives of the effective variables, and together with eq. (21), the radiation flux ε

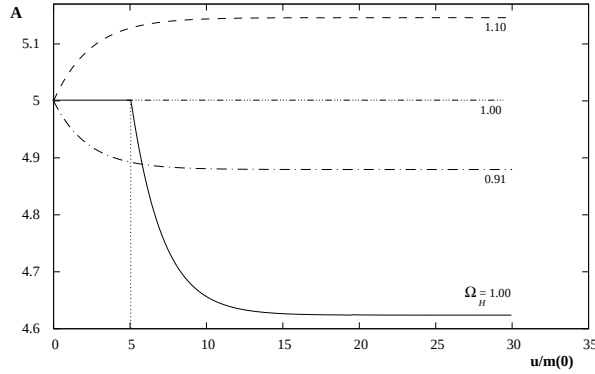


Fig. 1: The radius A as a function of the normalized time-like coordinate $\frac{u}{m(0)}$ for the initial values $A = 5.0$; $M = 1.0$ in the model Schwarzschild-like. Dashed line: $\Omega = 1$ static equilibrium, $\Omega = 1.1$ expansion; $\Omega = 0.91$ collapse. Solid line: solutions according to Herrera et al. [6].

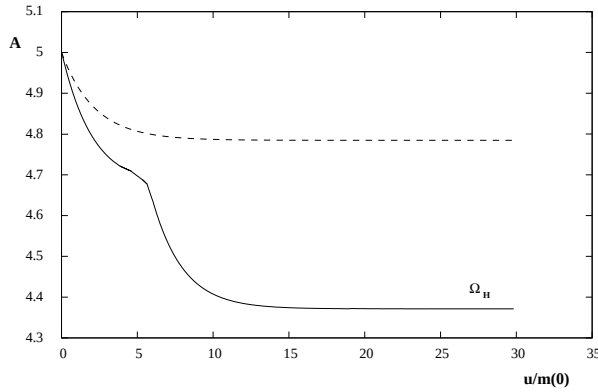


Fig. 2: The radius A as a function of the normalized time like coordinate $\frac{u}{m(0)}$ in the Schwarzschild-like model. Initial values for the surface variables $A = 5$; $M = 1.0$; $\Omega = 0.833$. Dashed line: calculations present. Solid line: solutions according to Herrera et al. [6].

is calculated. With the field equations (19) to (22) we can calculate the expressions of the physical variables ω , $\bar{P}$, $\bar{\rho}$, if we know the expressions $m(u, r)$ and $\beta(u, r)$ in each layer of the material under study. As a consequence, the state equations $P(u, r)$, $\rho(u, r)$ play an important role in determining the behavior of the physical variables present in the field equations and establishing their posterior evolution.

3 The models and surface equations

From the field equations (19) and (20) we can see

$$m = \int_0^r 4\pi r^2 \rho dr, \quad (26)$$

$$\beta = 2\pi \int_a^r \frac{\rho + P}{1 - \frac{2m}{r}} r \cdot dr. \quad (27)$$

These expressions for m and β are very similar to those obtained in the static case. This suggests a procedure to obtain dynamic solutions, following the same method of Herrera et al. [6], starting from a static solution:

1. Select a static solution of the gravitational field equations for a perfect fluid with spherical symmetry that explicitly shows its radial dependence

$$\rho_{static} = \rho(r) \quad P_{static} = P(r),$$

2. Suppose that the effective variables P and ρ (eq. 10) have the same radial dependence as in the static solution, but taking into account that now the edge condition $\bar{P}_a = 0$ is now expressed as

$$P_a = -\omega_a \rho_a. \quad (28)$$

Note that the subscript Δ_a indicates that the quantity Δ is evaluated at the edge of the distribution.

3. With this radial dependence for the effective variables, and together with (26) and (27), the values of m and β are calculable, except for three unknown functions (surface variables) that we are going to determine:

- (a) Equation (9) evaluated at $r = a$.
- (b) Equation (25) evaluated at $r = a$.
- (c) Equation (15) evaluated at $r = a$, or equation (22) evaluated at $r = a$.

4. Integrating numerically the ordinary differential equations obtained in (3), for a set of initial data, we completely determine the functions m and β .

5. With the field equations (19) to (22) we can calculate the expressions of the physical variables for the model considered.

As outlined in the previous methods (subsection 3), it is necessary to establish the surface variables and the equations that control its evolution.

- As mentioned in (subsection 3a), one of the surface equations is (9) evaluated at $r = a$, which takes the form

$$\dot{a} = \frac{da}{du} = \dot{A} = F(\Omega - 1), \quad (29)$$

where here it is very convenient to standardize the variables in terms of the initial mass $m(0) = m(u = 0, r = a)$ and define as surface variables:

$$A \equiv \frac{a}{m(0)} \quad M \equiv \frac{m_a}{m(0)} \quad \Omega \equiv \frac{1}{1 - \omega_a}, \quad (30)$$

as well as the variable

$$F = \left[e^{2\beta} \left(1 - \frac{2m}{r} \right) \right]_{r=a} = \left(\frac{V}{r} \right)_a, \quad (31)$$

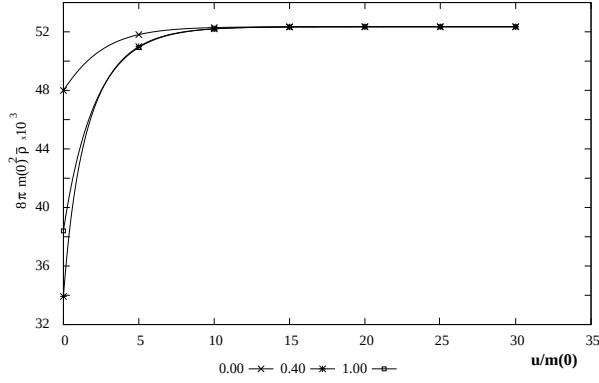


Fig. 3: Density $8\pi m(0)^2 \bar{\rho}$ in function of the temporal variable $\frac{u}{m(0)}$ for the model of Schwarzschild, for $\frac{\xi}{a} = 0.00, 0.40$ and 1.00 .

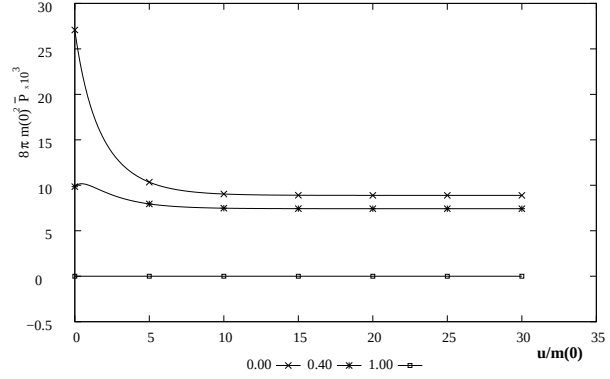


Fig. 4: Normalized pressure values $8\pi m(0)^2 \bar{P}$ in function of the time variable $\frac{u}{m(0)}$ for the model of Schwarzschild, for $\frac{\xi}{a} = 0.4$ and 1.00

Instead of using as surface variable F (31) – the gravitational potential at the surface –, as used in the references [6, 20, 21, 28]; we will use mass M , that is, the equation for radial evolution is

$$\dot{A} = \left(1 - \frac{2M}{A}\right)(\Omega - 1), \quad (32)$$

This equation is valid for all models.

- The second equation (25) is dependent on the model and it becomes necessary to calculate the first derivatives of the effective density and pressure, as can be seen.

$$\begin{aligned} 2m_{,0}|_a \left\{ 1 - \frac{m}{r} + r\beta_{,1} \left(1 - \frac{2m}{r} \right) - m_{,1} \right\}_a + \\ - r \left(1 - \frac{2m}{r} \right) \frac{\partial}{\partial u} (4\pi r^2 \rho) \Big|_a = \quad (33) \\ = \Omega(\Omega - 1) \left[4\pi r^2 \left(1 - \frac{2m}{r} \right)^2 r \frac{\partial}{\partial r} (\rho + P) \right]_a \end{aligned}$$

- The last equation (23) is the Tolman - Oppenheimer - Volkoff conservation equation evaluated at $r = a$, which we can write

$$\begin{aligned} \beta_{,10}|_a = 2\pi r \left(\frac{\partial P}{\partial r} \right) \Big|_a + \\ + \left[\frac{4\pi r^2 (\rho + P)}{2r^2 \left(1 - \frac{2m}{r} \right)} \right] \left(4\pi r^2 P + \frac{m}{r} \right)_a + \quad (34) \\ + 4\pi (P - P_t)|_a \equiv G. \end{aligned}$$

Both equations (33) and (34) have a similar structure, in terms of the surface variables:

$$\Upsilon_M \dot{A} + \Xi_M \dot{M} + \Lambda_M \dot{\Omega} = \Delta_M \quad (35)$$

$$\Upsilon_\Omega \dot{A} + \Xi_\Omega \dot{M} + \Lambda_\Omega \dot{\Omega} = \Delta_\Omega, \quad (36)$$

where

$$\Upsilon_\xi \equiv \Upsilon_\xi(A, M, \Omega), \quad \Xi_\xi \equiv \Xi_\xi(A, M, \Omega),$$

$$\Lambda_\xi \equiv \Lambda_\xi(A, M, \Omega), \quad \Delta_\xi \equiv \Delta_\xi(A, M, \Omega), \quad \forall \xi \in \{M, \Omega\}$$

are functions of (A, M, Ω) . These three equations (32), (33) and (34) allow us to establish a system of three ordinary differential equations for the surface variables; which together with the initials data set, determine m and β , as set forth in subsection 4. Below are two examples for the interior distribution Schwarzschild-like and Tolman VI-like in section 4 and 5, respectively.

4 The Schwarzschild-like model

We will get as the first test example Schwarzschild's well-known internal and static constant density solution. For this, we are going to assume that the density depends only on the time-type variable, as explained in [6, 32] we can write the state equation for the Schwarzschild type model as

$$\rho = \frac{3m}{4\pi r^3} \quad (37)$$

$$P = \rho \left\{ \frac{1 - \frac{1}{g} \left[\frac{1 - \frac{2M}{A} \left(\frac{r}{a} \right)^2}{1 - \frac{2M}{A}} \right]^{\frac{1}{2}}}{\frac{1}{g} \left[\frac{1 - \frac{2M}{A} \left(\frac{r}{a} \right)^2}{1 - \frac{2M}{A}} \right]^{\frac{1}{2}} - 3} \right\}, \quad (38)$$

where the value of g is determinated from the boundary condition ($\bar{P}_a = 0$) then the effective pressure satisfies the relationship (28); and consequently $g = \frac{1}{3-2\Omega}$. Evaluating equation (9), for $r = a$, we get (32) and with (23) and (24) for

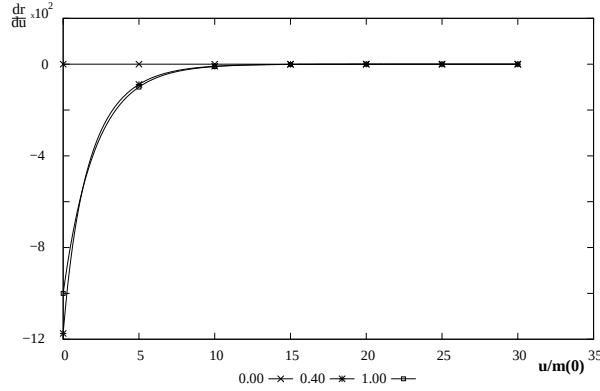


Fig. 5: Temporal velocity evolution in coordinates radiative $\frac{dr}{du}$, for the model of Schwarzschild, for $\frac{r}{a} = 0.00, 0.40$ and 1.00 .

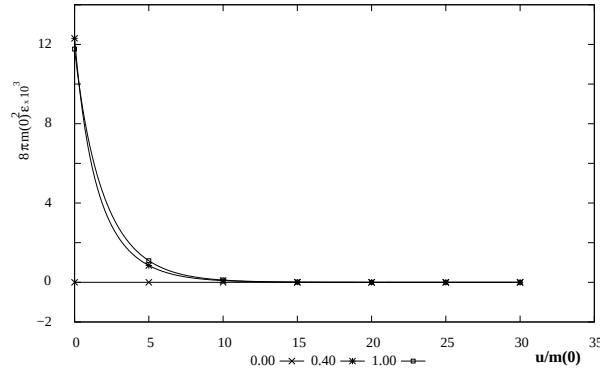


Fig. 6: Radiation profiles $8\pi m(0)^2 \epsilon$, for the model of Schwarzschild. All layers emit radiation but decrease until reaching a state of equilibrium for $\Omega = 1$.

$r = a$ we obtain then

$$\begin{aligned}\dot{M} &= 3 \left(\frac{M}{A} \right)^2 \left(1 - \frac{2M}{A} \right) \frac{(\Omega - 1)(2\Omega - 3)}{\frac{3M}{A} - \Omega \left(1 + \frac{2M}{A} \right)} \\ \dot{\Omega} &= \frac{2\Omega(1 - \Omega)}{A} \left(1 - \frac{2M}{A} \right) - \frac{1}{A} \left(\frac{M}{A} \right) \left(\frac{3 - 2\Omega}{1 - \frac{2M}{A}} \right) \dot{A} + \\ &\quad - \frac{\Omega}{M} \frac{1}{\left(1 - \frac{2M}{A} \right)} \dot{M},\end{aligned}$$

and from (20) and (19) we obtain, then after the immediate integration

$$\begin{aligned}m(r) &= m(0) \cdot M \left(\frac{r}{a} \right)^3 \\ \beta &= \frac{1}{2} \log \left[1 + \frac{3}{2\Omega} \left(\sqrt{\frac{1 - \frac{2M}{A}}{1 - \frac{2M}{A} \left(\frac{r}{a} \right)^2}} - 1 \right) \right].\end{aligned}$$

Figures 1 and 2 show the evolution of the radius A . Notice that $\Omega = 1$ represents a condition of static equilibrium,

$\Omega > 1$ represents expansion, $\Omega < 1$ the collapse. In both cases the system returns to equilibrium very quickly. In order to make some comparison, we took the initial data very close to those chosen in the reference [6]. We did not use the value for $\Omega = 1$, since with this approximation the system does not have static behavior. The figures 3, 4, 5, 6 represent the profiles of physical variables versus the time like coordinates for different pieces of material and for initials data. We obtain monotonous variations in the physical quantities, as a consequence of the non-assumption of the Gaussian pulse. In particular it is shown in figure 6, how all the layers emit monotonously, unlike the figure 7 in Herrera et al. [6]

5 The Tolman VI-like model

Following [2] we can assume as static solution

$$\begin{aligned}4\pi a^2 \rho &= 3h \left(\frac{a}{r} \right)^2 \\ 4\pi a^2 P &= h \left(\frac{a}{r} \right)^2 \left[\frac{1 - 9 \cdot z \left(\frac{r}{a} \right)}{1 - z \left(\frac{r}{a} \right)} \right],\end{aligned}$$

as before the value of z is determinated from the boundary condition ($\bar{P}_a = 0$) then $P_a = -\omega_a \rho_a$; and consequently $z = \frac{4\Omega - 3}{3(4\Omega - 1)}$; and $h = \frac{m}{3r}$. Evaluating the equations (9), (23) and (24) at $r = a$ we obtain

$$\begin{aligned}\dot{M} &= - \frac{\left(1 - \frac{2M}{A} \right)^2 (16\Omega^2 + 3) \left(\frac{M}{A} \right) \dot{A}}{8 \left[\Omega \left(1 - \frac{2M}{A} \right) + \frac{M}{A} \right]} \\ \dot{\Omega} &= - \frac{1}{A} \left[\frac{M}{A} - \frac{(4\Omega - 3)(4\Omega - 1)}{8} \left(1 - \frac{2M}{A} \right) \right] \\ &\quad + (4\Omega - 1)(4\Omega - 3) \frac{\dot{A}}{8A} + \frac{\Omega}{\left(1 - \frac{2M}{A} \right)} \frac{\dot{M}}{M},\end{aligned}$$

and the corresponding values of β and m are

$$\begin{aligned}\beta &= \frac{2M}{3A} \frac{1}{\left(1 - \frac{2M}{A} \right)} \left\{ \log \left(\frac{r}{a} \right) + 2 \log \left[\frac{3 - \left(\frac{4\Omega - 3}{4\Omega - 1} \right) \left(\frac{r}{a} \right)}{3 - \left(\frac{4\Omega - 3}{4\Omega - 1} \right)} \right] \right\} \\ m &= m(0) M \left(\frac{r}{a} \right).\end{aligned}$$

Figure 7 shows the temporal variation of the radius of a radiant sphere, for different values of Ω . There is a critical value Ω_0 for $A = 6.66$ and $M = 1$ and slight increase in Ω causes a permanent expansion or the contraction rises to the critical value. In the following figures 8,9,10 we show the variations of pressure density and radiation of some interior layers in case of surface expansion (explosion-like). We noted in the example in the figure 10 that all layers absorbed energy during the initial collapse, and then a radiative pulse is emitted, before returning to the equilibrium configuration.

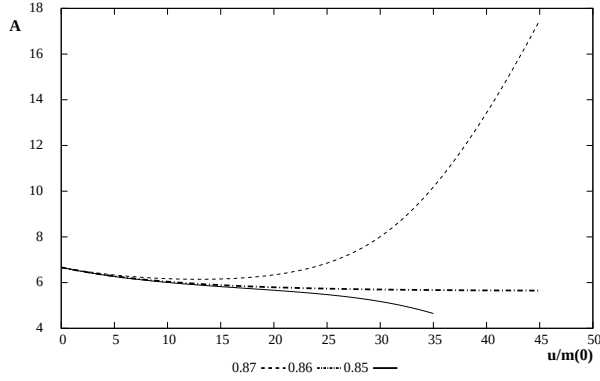


Fig. 7: Radio A of the distribution according to the variable temporary $\frac{u}{m(0)}$ in a TolmannVI-like model. With initials values, $A = 6.66667$; $M = 1$. $\Omega = 0.87$ (rebound) and $\Omega = 0.85$ (contraction) and the critical value $\Omega = 0.860723$

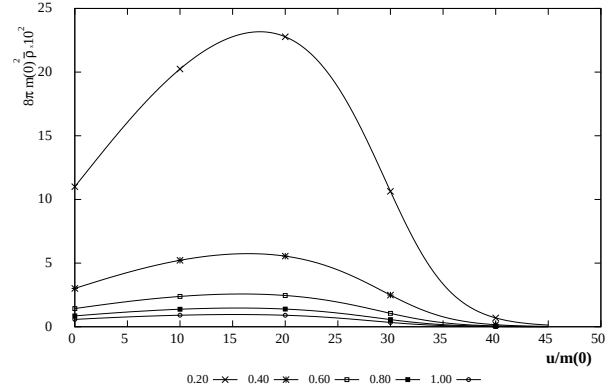


Fig. 9: Density $8\pi m(0)^2 \bar{\rho}$ as function of the time-like variable $\frac{u}{m(0)}$ in a TolmannVI-like model. For the layers in contraction ($\Omega = 0.87$) with $\frac{z}{a} = 0.2, 0.4, 0.6, 0.8$ and 1 .

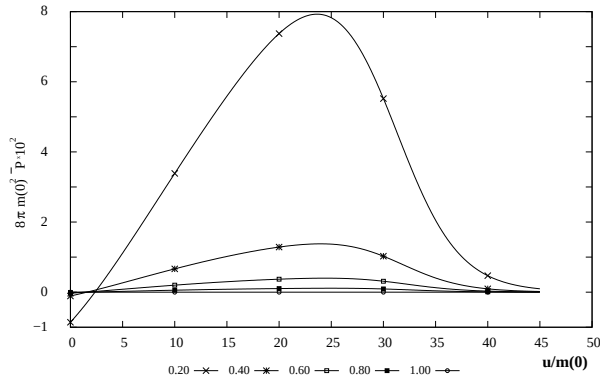


Fig. 8: Values of the normalized pressure $8\pi m(0)^2 \bar{P}$ as function of the time-like variable $\frac{u}{m(0)}$ in a TolmannVI-like model. For the layers in contraction ($\Omega = 0.87$) with $\frac{z}{a} = 0, 0.4, 0.6, 0.8$ and 1 .

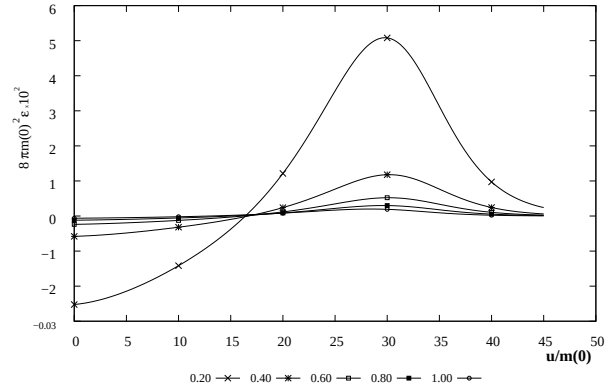


Fig. 10: Radiation profiles emitted $8\pi m(0)^2 \epsilon$ as function of the time-like variable $\frac{u}{m(0)}$ in a TolmannVI-like model. For the layers in contraction ($\Omega = 0.87$) with $\frac{z}{a} = 0.2, 0.4, 0.6, 0.8$ and 1 .

6 Conclusions

We have reviewed the relativistic description of the collapse of self-gravitating radiant spheres, following the usual procedure [6, 19–21, 23, 25, 28, 33] and find that it is an effective method for such a purpose, since the field equations together with the conservation laws (Bianchi's Identity) form a complete set of integrable equations that do not require an additional hypothesis about the emission of radiated energy. That is, the emission hypothesis of a Gaussian pulse at an arbitrary instant to trigger the collapse; it is not only unnecessary, but also leads to qualitatively and quantitatively different solutions, as we have shown in figures 1-9. We emphasize the importance of using conservation equations properly, as was done in Section 2; We formally reobtain the generalized TOV equation of the hydrostatic equilibrium (equation 23) and a relativistic version of the Euler equation for the self-gravitating sphere (equation 25).

We have seen that the Schwarzschild-like description is an

ideal case that does not represent the phenomenology of the high energy events observed in the stellar collapse of massive stars such as supernovas and quasars. The measurable magnitudes of density, pressure and emission evolve smoothly, returning to the equilibrium condition very rapidly (Figures 3-6). On the other hand, the Tolman VI description involves two possible qualitatively different scenarios, such as the implosion or the explosion of the outer layers of the self-gravitating sphere, depending on the initial values of the mass, radius and velocity observables, as we have shown in figure 7.

We have shown that, in the case of contraction, the density and pressure variables similarly evolve (Figures 8 and 9) as might be expected if a polytrope state equation is used. In addition, Figures 8 and 9, show a dependence of the evolution of such magnitudes according to the radius of the considered layer, with much higher values of density and pressure in the innermost layers, in agreement with the description of the stellar collapse of massive stars.

Finally figure 10 shows that during the collapse of the

self-gravitating radiating spheres a pulse of radiation emission is generated before reaching equilibrium again; which arises *naturally* from the complete solution of the evolution equations, and maybe is important to explain the emission process in very high energy in Supernova bursts and Quasars.

Submitted on January 12, 2018

References

- Oppenheimer J. R. and Snyder H. On Continued Gravitational Contraction. *Physical Review*, 1939, v. 56, 455–459.
- Tolman R. C. Static Solutions of Einstein's Field Equations for Spheres of Fluid. *Physical Review*, 1939, v. 55, 364–373.
- Vaidya P. C. Nonstatic Solutions of Einstein's Field Equations for Spheres of Fluids Radiating Energy. *Physical Review*, 1951, v. 83, 10–17.
- Bondi H. The Contraction of Gravitating Spheres. *Proceedings of the Royal Society of London Series A*, 1964, v. 281, 39–48.
- Bayin, S. Ş Radiating fluid spheres in general relativity. *Phys. Rev. D*, 1979, v. 19, 2838–2846.
- Herrera L., Jiménez J., and Ruggeri G. J. Evolution of radiating fluid spheres in general relativity. *Phys. Rev. D*, 1980, v. 22, 2305–2316.
- Patino A. and Rago H. The Effect of Electric Charge on the Evolution of Radiant Spheres in General Relativity. *Astrophysics and Space Science*, 1996, v. 241, 237–247.
- Sah A. and Chandra P. Class of Charged Fluid Balls in General Relativity. *International Journal of Astronomy and Astrophysics*, 2016, v. 6, 494–511.
- Pant N., Pradhan N. and Bansal R. K. Relativistic model of anisotropic charged fluid sphere in general relativity. *Astrophysics and Space Science*, 2016, v. 361, 41.
- Harko T. and Mak M. K. Exact power series solutions of the structure equations of the general relativistic isotropic fluid stars with linear barotropic and polytropic equations of state. *Astrophysics and Space Science*, 2016, v. 361, 283.
- Maurya S. K. and Maharaj S. D. Anisotropic fluid spheres of embedding class one using Karmarkar condition. *ArXiv e-prints*, 2017.
- Singh K. N., Bhar P., and Pant N. A new solution of embedding class I representing anisotropic fluid sphere in general relativity. *International Journal of Modern Physics D*, 2016, v. 25, 1650099.
- Herrera L. and Falcón N. Heat waves and thermohaline instability in a fluid. *Physics Letters A*, 1995, v. 201, 33–37.
- Herrera L. and Santos N. O. Cylindrical collapse and gravitational waves. *Classical and Quantum Gravity*, 2015, v. 22, 2407–2413.
- Van den Bergh N. and Slobodeanu R. Shear-free perfect fluids with a barotropic equation of state in general relativity: the present status. *Classical and Quantum Gravity*, 2016, v. 33(8), 085008.
- Herrera L., Denmat G. L. and Santos N. O. Dynamical instability and the expansion-free condition. *General Relativity and Gravitation*, 2012, v. 44, 1143–1162.
- Aguirre F., Hernandez H., and Nunez L. A. Radiation hydrodynamics and radiating spheres in general relativity. *Astrophysics and Space Science*, 1994, v. 219, 153–170.
- Chan R., Herrera L., Pacheco J. A. F. and Santos N. O. Diffusion processes in the collapse of a radiating spherical body. *Astrophysical Journal*, 1991, v. 382, 255–260.
- Herrera L. and Barreto W. Relativistic Gravitational Collapse in Co-moving Coordinates: the Post-Quasistatic Approximation. *International Journal of Modern Physics D*, 2011, v. 20, 1265–1288.
- Cosenza M., Herrera L., Esculpi M. and Witten L. Evolution of radiating anisotropic spheres in general relativity. *Phys. Rev. D*, 1982, v. 25, 2527–2535.
- Medina V., Nunez L., Rago H. and Patino A. Evolution of radiating charged spheres in general relativity. *Canadian Journal of Physics*, 1988, v. 66, 981–986.
- Bonnor W. B., de Oliveira A. K. G., and Santos N. O. Radiating spherical collapse. *Phys. Rep.*, 1989, v. 181, 269–326.
- di Prisco A., Falcón N., Herrera L., Esculpi M. and Santos N. O. Pre-relaxation Processes in a Radiating Relativistic Sphere. *General Relativity and Gravitation*, 1997, v. 29, 1391–1405.
- Herrera L., Barreto W., di Prisco A., and Santos N. O. Relativistic gravitational collapse in noncomoving coordinates: The post-quasistatic approximation. *Phys. Rev. D*, 2002, v. 65(10), 104004.
- Barreto W., Rodriguez B., and Martinez H. Radiating Fluid Spheres in the Effective Variables Approximation. *Astrophysics and Space Science*, 2002, v. 282, 581–593.
- Pant N., Mehta R. N., and Tewari B. C. Relativistic model of radiating massive fluid sphere. *Astrophysics and Space Science*, 2010, v. 327, 279–283.
- Tewari B. C. Relativistic collapsing radiating stars. *Astrophysics and Space Science*, 2012, v. 342, 73–77.
- Patiño A. and Rago H. A Sphere Contraction in General Relativity. *Lett. Nuovo Cimento*, 1983, v. 38, 321–328.
- Davies G. Second-Order Black Hole Perturbations: A Computer Algebra Approach, I — The Schwarzschild Spacetime. *ArXiv General Relativity and Quantum Cosmology e-prints*, 1998.
- Neary N., Ishak M., and Lake K. Tolman type VII solution, trapped null orbits, and w-modes. *Phys. Rev. D*, 2001, v. 64(8), 084001.
- Bekenstein J. D. Hydrostatic Equilibrium and Gravitational Collapse of Relativistic Charged Fluid Balls. *Phys. Rev. D*, 1971, v. 4, 2185–2190.
- Aguirre F., Nunez L. A., and Soldovieri T. Variable Eddington Factor and Radiating Slowly Rotating Bodies in General Relativity. *ArXiv General Relativity and Quantum Cosmology e-prints*, 2005.
- Barreto W., Rodríguez B., Rosales L., and Serrano O. Self-similar and charged radiating spheres: an anisotropic approach. *General Relativity and Gravitation*, 2007, v. 39, 537–538.