Universidad Nacional Autónoma de México
Posgrado en Astrofísica
Instituto de Astronomía

Angular momentum effects in the evolution of a relativistic jet through a massive progenitor using 3D numerical simulations

TESIS
Que para optar por el grado de:
Maestra en Ciencias (Astrofísica)

Presenta:
María Fernanda Clever Uribe

Tutor
Dr. Diego López Cámara Ramírez
Instituto de Astronomía, UNAM

Miembros del comité tutor
Dra. Gloria Delgado Inglada
Instituto de Astronomía, UNAM
Dr. Aldo Rodríguez Puebla
Instituto de Astronomía, UNAM

Ciudad Universitaria, CD.MX., Enero de 2021

Jury:

President: Dr. Fabio De Colle

Secretary: Dra. Rosa Leticia Becerra Godínez

Vocal: Dr. Emilio Tejeda Rodríguez

1st Substitute: Dr. José Alejandro Esquivel Salazar

2nd Substitute: Dr. Simone Dichiara

Supervisor:

Dr. Diego López Cámara Ramírez

Work presented on January 2021 to the jury:

President: Dr. Fabio De Colle
Secretary: Dr. Rosa Leticia Becerra Godínez
Vocal: Dr. Emilio Tejeda Rodríguez

Advisor:
Dr. Diego López Cámara Ramírez.

This work was developed at Instituto de Astronomía, Universidad Nacional Autónoma de México, under the tutoring of Dr. Diego López Cámara Ramírez. This work was funded by the projects LANCAD-UNAM-DGTIC-321 (DGTIC-UNAM) and PAPIIT AG 100820.

*The path I chose didn't destroy what I was, but rather, it made me unmasked the **plucky** being I needed to be.*

Dedication

A mis padres. Aunque he estado lejos de ustedes por más de cinco años, siempre están en mi mente y en mi corazón. Espero que en algún momento se sientan orgullosos de mí.

A mis tías, Leticia y Norma, y a mis abuelos, Nidelvia y Pedro; sin su apoyo no podría haber llegado tan lejos. Tienen todo mi cariño y admiración, ahora y siempre.

A Félix, quien se mantuvo a mi lado a lo largo de este extraño y complicado año, y por más de nueve años. Por ser mi compañero fiel, mi refugio y mi motivación. Asimismo, quiero agradecer a sus padres, Mónica y Félix, y su hermana, Andy, por su calidez y reconfortantes ánimos.

Al profesor Pedro Miramontes, ahora un gran amigo y guía para mi. Por sus consejos (y a veces regaños) que me impulsan a ser una mejor persona.

A mis amigos Juan, Paola, Tito, José Luis, Guille y Luis Fernando, quienes son parte importante de mi vida y una gran segunda familia. A Alejandra y Mayté, que siguen siendo mis grandes amigas incluso después de tantos años. A mis más recientes amigos, Milton, Jess, Iván, Aguayo, Josué, José y Eli; por sus consejos, bromas y tardes de pizza que compartieron conmigo. Gracias por su amistad y comprensión.

Acknowledgments

To Dr. Diego López Cámara Ramírez, who has been my supervisor for more than four years. For his guide and support.

To the members of the jury; Dr. Fabio de Colle, Dra. Rosa Leticia Becerra Godínez, Dr. Emilio Tejeda, Dr. José Alejandro Esquivel Salazar and Dr. Simone Dichiara.

Specially, I would like to thank my *alma mater*, Universidad Nacional Autónoma de México. For teaching me the values of study and hard work.

Also, I want to thank to all my teachers of Instituto de Astronomía and Instituto de Ciencias Nucleares; for their support and all the knowledge I have acquired.

To the funding provided by the proyect LANCAD-UNAM-DGTIC-321 in Miztli supercomputer of UNAM and the proyect PAPIIT AG 100820.

Contents

List of Figures

Resumen

Los destellos de rayos gamma largos son fenómenos de altas energías y transitorios relacionados con chorros (jets) colimados que se mueven a velocidades relativistas cuyo posible origen son estrellas masivas evolucionadas que poseen un mecanismo central (agujero negro) que acreta y colima el material generando un jet. En este modelo conocido como colapsar, el jet se ve sometido a la dinámica de su progenitor, lo cual puede alterar su propagación y su subsecuente interacción con el medio interestelar. Una de las principales características de estos progenitores es su rápida rotación, que puede modificar tanto la morfología del propio progenitor así como la del jet. En este trabajo se presentan una serie de modelos tridimensionales de progenitores con diferentes distribuciones de momento angular realizados a partir de simulaciones numéricas. Asimismo, mostramos un análisis del cambio en las morfologías de los progenitores a un tiempo posterior en el cual alcanzaron su deformación máxima. Tomando en cuenta éstos resultados, se seleccionaron dos modelos representativos como progenitores del jet relativista a ser comparados con un modelo sin rotación. Nuestras simulaciones exhiben una relación entre la distribución del momento angular del progenitor con ciertas propiedades del jet tales como el factor de Lorentz máximo que alcanza el jet, el tiempo que le lleva alcanzar la superficie del progenitor (*break out time*) y el ancho de su envolvente (o *cocoon*). Los resultados indican que al considerar un momento angular con una distribución angular, el jet se propaga más fácilmente, comparándolo con una distribución lineal, alcanzando factores de Lorentz mayores ($\Gamma_{\rm bo} \sim 16$) y conlleva a una disminución del *break out time* a comparación de un progenitor sin rotación. La rotación tiene otras implicaciones de gran importancia sobre la evolución del jet derivado del comportamiento de los parámetros antes mencionados. Uno de ellos es el tamaño del radio fotosférico, pues al incluir rotación, se alcanzan velocidades mayores en el jet, lo cual implica una diminución en la densidad, y con ello la disminución de la profundidad óptica a una distancia menor a diferencia de un modelo sin rotación (suponiendo una opacidad constante). Debido a que el momento angular afecta la fase pronta del jet, se discuten futuras implementaciones que complementarían el marco teórico de este modelo.

Abstract

Long gamma-ray bursts are transient high energy phenomena related to relativistic jets whose possible origin are massive evolved stars with a central mechanism (black hole), which accretes and collimates the material and produces a jet. In this model known as the collapsar, the jet is subject to the dynamics of its progenitor and this leads to changes in its propagation and its subsequent interaction with the interstellar medium. One of the main features about these progenitors is the rapid rotation, it can modify the structure of the progenitor itself as well as that of the jet. We present a series of three-dimensional progenitor models with different angular momentum distributions performed through numerical simulations. Likewise, we show an analysis on the alteration of progenitor's morphology at a later time (t=10 s). By considering the previous results, we selected two representative models to be the progenitors of the relativistic jet and they were compared against a non-rotating model. Our simulations show a relation between the angular momentum distribution and some of the jet's properties, such as de maximum Lorentz factor, the break out time, and the width of the cocoon. Also, this results indicate that an angular distribution of angular momentum allows an easier propagation of the jet, compared with a linear distribution, reaching higher Lorentz factor values ($\Gamma_{\rm bo} \sim 16$) and leading to a decrease of the break out time when comparing with a non-rotating progenitor. The inclusion of rotation has additional important implications derived from the previous parameters such as the photospheric length. As the velocities became higher in our models, the density dropped, which will lead to a decrease of the optical depth (assuming a constant opacity), compared with a non-rotating model. With this results, we concluded that the angular momentum affects the jet's prompt emission (and possibly its following stages), some improvements to the theoretical framework that could be implemented to the model are also discussed.

Motivation and Goals

Our motivation to perform this study comes from the desire to improve some of the previous studies concerning angular momentum by taking different angular momentum distributions and proposing long gamma-ray bursts rotating progenitors based on these angular momentum settings. By building a three-dimensional model of a rotating progenitor, we can approach a more realistic model and we can study the effects of the stellar rotation on the propagation of the jet.

Also, by comparing the final models against the non-rotating progenitor model, we expect to find some differences in the features of the jet such as the break out time (the time at which the jet reaches the surface of the progenitor), the maximum Lorentz factor attained by the material in the jet, the morphology of the stellar cocoon surrounding the jet, among others.

Finally, the main goals of this study were the following;

- Study the effects of rotation on the morphology of the progenitors.
- Find possible differences in the Lorentz factor, break out time and morphology of a relativistic jet drilling the rotating progenitor in comparison with a non-rotating model.
- Infer some implications for the jet evolution using the physical properties obtained from the models and relate them to possible observational data.

1 — Introduction

Throughout this chapter, we will introduce the astrophysical phenomena of long Gamma-Ray Bursts (lGRBs) within a general framework. Next, we will frame this phenomena within the so-called collapsar model, which is the main basis of our simulations and additionally, we will discuss some previous studies that have analyzed the angular momentum implementation in their models and we will summarize their results.

1.1 Gamma-Ray Bursts

Gamma-Ray Bursts are one of the most energetic events catalogued in transient astrophysics. Their own name has two important features that describe these phenomena: A sudden release of high energy photons ranging from keV to several MeV and the typical isotropic luminosity in γ-rays is about $10^{51} - 10^{53}$ erg s^{-1}.

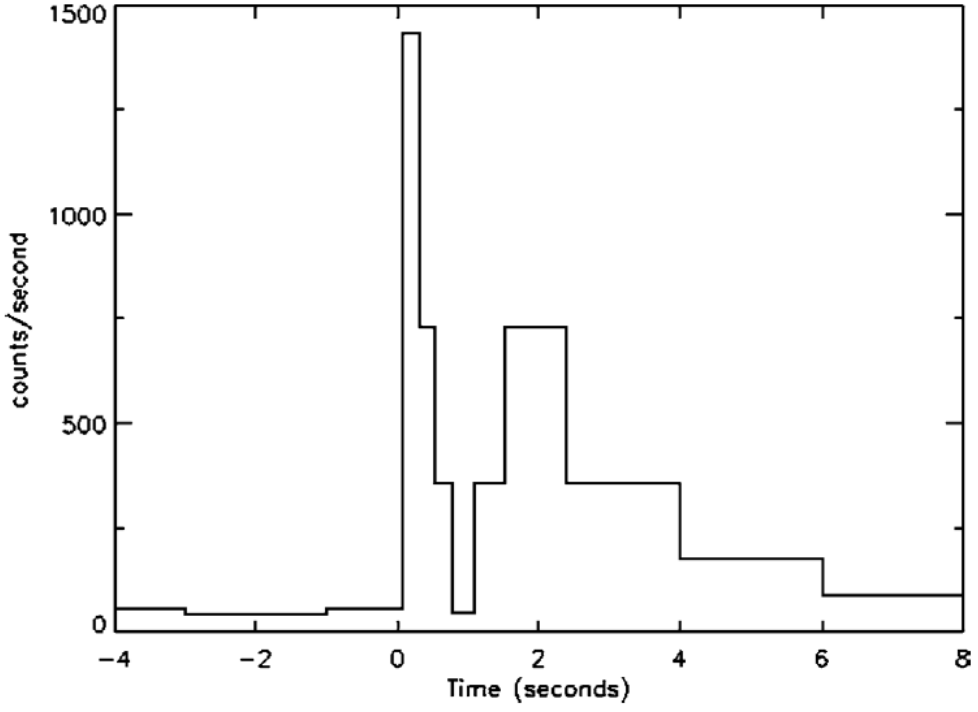

Figure 1.1: Light curve of the first GRB detected in 1967. (Image credit: Klebesadel et al., 1973)

The discovery was a serendipitous finding. It was on July 2nd, 1967, when the Vela military

satellites (specifically, the Vela IVa,b satellites) detected a GRB for the first time ever. Because of the need to monitor x-ray flashes after a nuclear blast, these satellites possessed x-ray and gamma-ray detectors. Despite the very poor temporal resolution, their results shown some important features about this event: A light curve with two asymmetric peak fluxes, one of them peaking around MeVs and a 10 s duration.

A few years later, in 1973, Klebesadel and his colleagues published the first light curve of this GRB, which was called 670702 (Figure 1.1). After this discovery, the GRB phenomenon became a growing field where the observational data was crucial in its understanding.

One of the most relevant missions was the Compton Gamma Ray Observatory spacecraft (CGRO)[1] which carried the instrument called BATSE. BATSE was meant to detect high energy transitory phenomena with gamma and X-ray emission and, during its lifetime, it detected around 2700 GRBs, averaging one per day. BATSE released valuable information about GBRs' extragalactic origins and distribution. For instance, Figure 1.2 exhibits a clearly isotropical sky distribution (in Galactic coordinates) of 2074 GRBs, together with their respective intensities (color bar). Also, the GRBs found by BATSE showed that their spectra are non-thermal and variability was evident. Finally, by studying the duration of the GRBs, two families of GRBs were identified (Kouveliotou et al., 1993); long-duration GRBs (lGRBs) and short-duration GRBs (sGRBs).

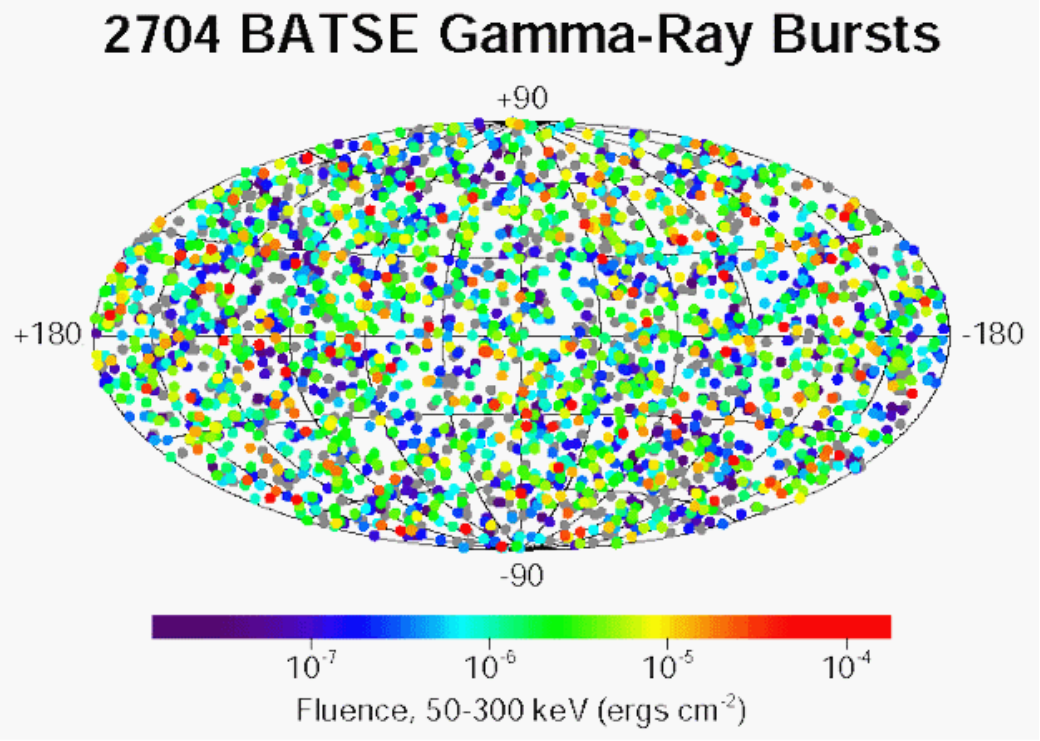

Figure 1.2: GRB map found by BATSE (CGRO). (Image credit: NASA)

By the end of the BATSE era, an improvement in observational data was necessary. This

[1] https://heasarc.gsfc.nasa.gov/docs/cgro/cgro/

need was fulfilled with the satellites BeppoSAX[2] and HETE[3] whose goal was to find GRB counterparts in different wavelengths such as the optical window or X-rays. Together, these satellites were capable of both, finding the first GRB *afterglows* and measuring its redshift (z=0.835) (Metzger et al., 1997) for the first time. Furthermore, the first evidence of the lGRBs' origin via the GRB-SN association was also provided by these satellites (Stanek et al., 2003; Hjorth et al., 2003).

The GRB exploration continued with the Swift observatory[4] that consisted of the Burst Alert Telescope (BAT), the X-Ray Telescope and the UV-Optical Telescope. These instruments covered a wide range of wavelengths which led to other discoveries concerning the GRBs' afterglows: Swift showed that short and long GRBs have different origins, and was able to identify host galaxies for some sGRBs. Additionally, it found the so-called X-ray flares and broadened the GRB's redshift range by contributing with the measurement of more bursts (Cusumano et al., 2006; Totani et al., 2006; Tanvir et al., 2009 among others).

The next GRB mission is the Fermi Gamma-Ray Space Telescope[5]. Fermi enable the broad-band spectral study of the prompt emission and the discovery of the different components in the spectrum, up to high energy (>10 GeV). It is mainly composed by the Large Area Telescope and the Gamma-ray Burst Monitor. Its main purpose is to study the GRB prompt emission spectra. Its detections of high energy photons has given us information about the physical mechanisms that produce the GRBs and these can further constrain the theoretical framework.

A new era is approaching with the launch of other missions such as SVOM (Space-based multi-band astronomical Variable Objects Monitor, http://www.svom.fr/en/), which is intended to increase the location accuracy of GRBs and enable follow-up with ground-based instruments to obtain substantial spectral data.

Now, we would like to introduce some concepts related with observable features of GRBs that might be useful for the following sections of this work. There are two essential regions which describe two phases of a GRB; these are the prompt emission and the afterglow. The prompt emission is defined as the emission of gamma-ray photons and is related with the duration of the burst. On the other hand, the afterglow of a GRB comes after this sudden release of energy. The ejecta emitted from the central engine of the GRB (Section 1.2) interacts with the surrounding medium and, eventually, decelerates. This interaction creates a forward shock (that propagates into the medium) and a reverse shock (that propagates towards the ejecta) (Section 2.2). Charged particles are accelerated in this shocks creating non-thermal emission (via synchrotron radiation). The ejecta slows down and results in a softer emission known as the afterglow.

With this description we can now characterize some other properties of the GRBs based on their observational features.

[2]https://heasarc.gsfc.nasa.gov/docs/sax/sax.html
[3]https://heasarc.gsfc.nasa.gov/docs/hete2/hete2.html
[4]https://www.nasa.gov/mission _pages/swift/main
[5]https://fermi.gsfc.nasa. gov/

As we mentioned before, there are two categories of GRBs distinguished by their duration (Figure 1.3); the duration is related to the parameter T_{90}, interval of time at which the detector collects from 5% to 95% of the total fluence. lGRBs have a T_{90} distribution that peaks around 20-30 s while the short-duration GRBs T_{90} peaks in 0.2-0.3 s (Zhang, 2018). Nevertheless, lGRBs possess other important features that distinguish them from sGRBs, some are described below.

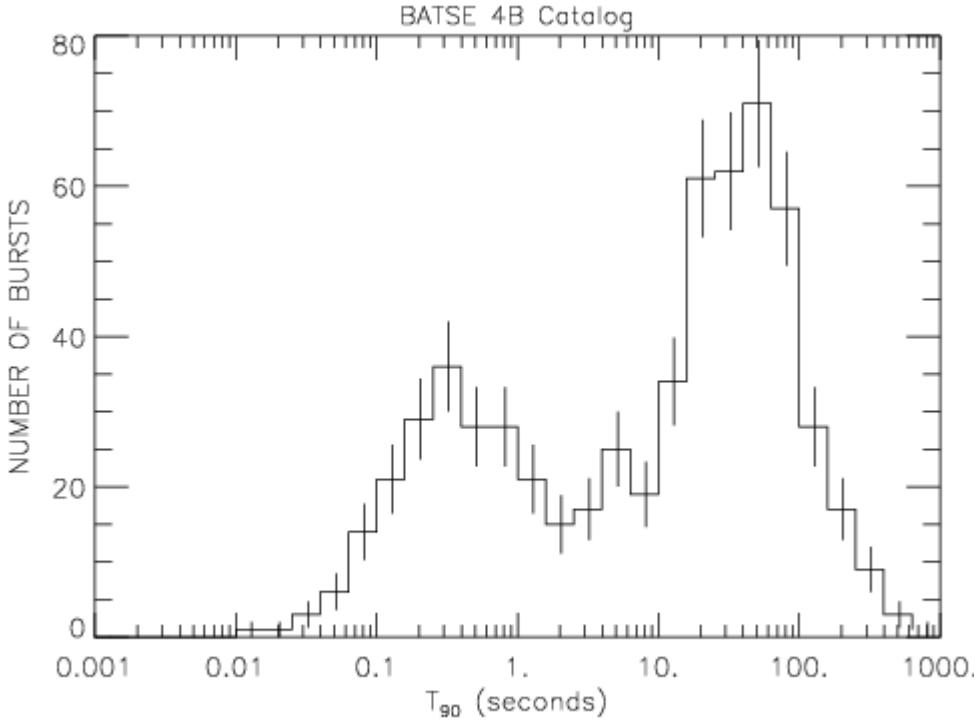

Figure 1.3: T_{90} distribution. (Image credit: NASA)

The prompt emission from a lGRB is softer than that of a sGRB by a few orders of magnitude. This property can be quantified by a hardness ratio (photon counts ratio between two energy bands). In addition to T_{90}, this can also lead to a bimodal separation between the two classes of GRBs, although the distinction is not as clear in this case (Horváth et al., 2006).

Another important feature is the type of host galaxies where lGRBs can be found. Many of them are located in irregular, star-forming galaxies, including some spiral galaxies (Fruchter et al., 2006), whereas sGRB are typically associated with all types of galaxies. Furthermore, it has been identified that lGRBs more likely occur in regions of low metallicity (Fruchter et al., 2006).

lGRB type is associated with supernovae Type Ibc (SN Ibc, no hydrogen or helium lines in their spectra) (an example is the SN 1998bw associated with GRB 980425, from Hjorth et al., 2003); these SNe have been associated to the final explosion of massive stars that, during their evolution, have lost their hydrogen envelope, which is a relevant ingredient for lGRBs' model: a core-collapsing massive star progenitor (Woosley and Bloom, 2006).

With this brief GRBs observational background, now, we are now ready to focus solely on the type of GRBs that concerns our study: lGRBs.

1.2 Long Gamma-Ray Burst Progenitors

In the previous section we mentioned that the main ingredient for an acceptable lGRB model is the presence of a massive star progenitor. On the other hand, there are many additional physical ingredients that must be taken into account in order to build a solid model. In the following lines we will discuss some of these features and the progenitors proposed for lGRBs.

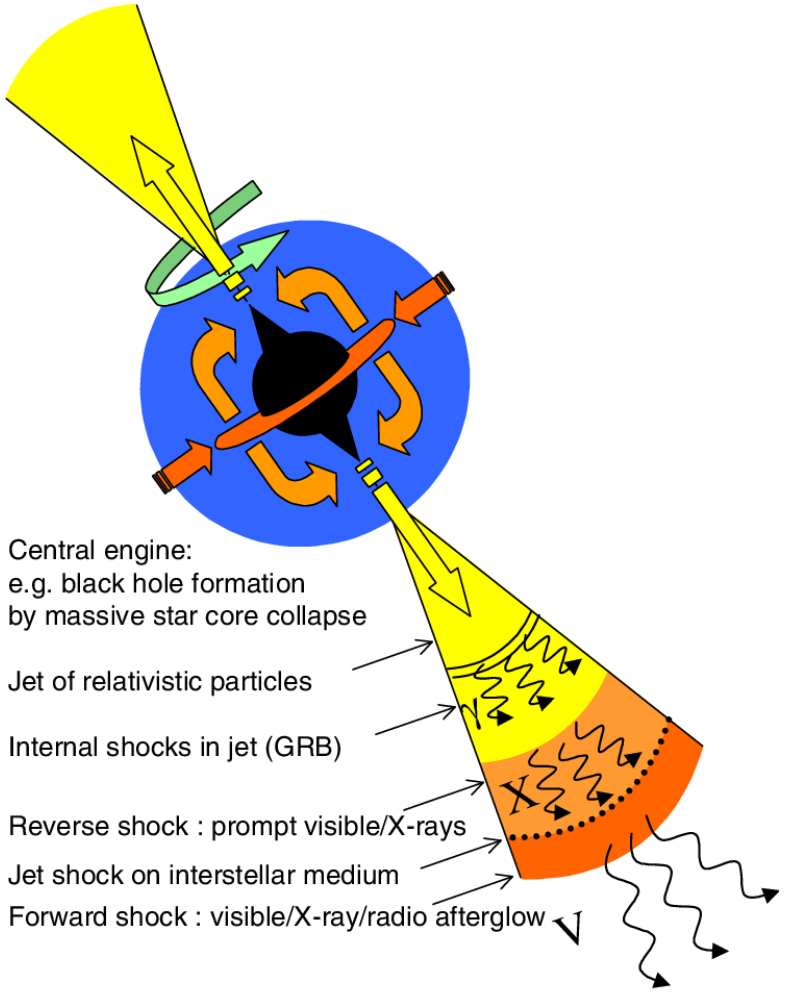

Figure 1.4: Collapsar progenitor scheme. (Image credit: Schanne et al., 2006)

One of the main candidates is a Wolf-Rayet (WR) star (Woosley, 1993), which is a rapidly rotating massive star with a large Zero-Age Main Sequence (ZAMS) mass ($\sim 20 M_{\odot}$). lGRBs are thought to be the result of the death of a massive star in which the star core collapses and produces a black hole (known as the central engine of the GRB) with a surrounding accretion disk (Woosley and Bloom, 2006). By means of the information inferred from the observation of these events, we are able to recognize the main characteristics of the progenitor star and one of them is the angular momentum. Angular momentum becomes then an important aspect to

study, since it must be present in the progenitor in order to create an accretion disk that will collect the surrounding material from the stellar envelope.

We need to point out that the star has lost its hydrogen envelope by this stage, which will facilitate the forming jet to break out from the star. Some authors have studied the effects of rotation rate and mass loss during the star's life and their link to metallicity (Woosley and Heger, 2006).

Because of the mass loss and the rotation rate play an important role in the initial mechanism of a lGRB, the WR stars are considered to be the most likely progenitor. In a nutshell, the currently preferred picture of lGRB progenitors goes as follows: A massive WR-star whose core rotates rapidly dies and eventually forms a black hole along with an accretion disk. An overview of this model, known as collapsar (Woosley, 1993), is displayed on Figure 1.4, where we can appreciate the central engine and its accretion disk, embedded within the massive envelope, and the relativistic jet breaking out from said envelope.

In terms of the collimation of the jet, there have been proposed some theoretical mechanisms to explain the formation of the stream towards the poles of the black hole. One of these mechanisms is the system introduced by Blandford & Znajek (1977) where a Kerr black hole, with a differentially rotating disk, holds a large magnetic field ($B \sim 10^{15}$ G) attached to its event horizon. Because of the previous conditions, in the Blandford-Znajek scheme appears the magnetorotational instability. This is also related to the disk viscosity (Shakura & Sunyaev, 1973); the friction due to viscosity leads to an energy loss and creates a turbulent environment in the disk.

Another important mechanism is the one proposed by Blandford & Payne (1982) where an inner black hole is also considered but instead, the magnetic field is attached to the differentially rotating disk and is smaller by several orders of magnitude ($B \sim 10^2 - 10^4$ G).

Besides the collapsar model, there have been several studies proposing alternative progenitors. Some of them incorporate combinations of binary systems that include black holes, neutron stars or white dwarfs (Ruffini et al., 2016, 2018). Some others study blue supergiants (Mészáros and Rees, 2001) and specific mergers, for instance, a helium star with a black hole (Fryer and Woosley, 1998). All of the above found interesting results which account for different lGRB properties; nevertheless, some of them were discarded because of the improvement of observational data, which provided new information about the lGRBs structure and physical parameters (energy, time-scales) which eventually lead to constrain its localization and, thus, the most likely progenitors. A full table with all possible progenitor candidates can be consulted in Nemiroff (1994). The collapsar model is the most accepted lGRBs progenitor because it gathers important observational features of this phenomena as we described before: SN Type Ic association, massive progenitor with low-metallicity that may be found in star-forming galaxies (short-lived stars) and the prompt emission duration.

Within the models introduced to study the lGRBs emissions, an important case to remark is the photospheric model (e.g., Mészáros et al., 2002; Daigne and Mochkovitch, 2002) proposed to study prompt phase. The basis of this model is related to the value of the photons optical depth, τ, which is much larger than unity bellow the photospheric radius, $R_{\rm ph}$, so there is no

energy loss due to photons.

Inside the region where $\tau \geq 1$, there can be several emission processes, such as synchrotron, Bremsstrahlung and Compton scattering but, since the medium is optically thick, the emission is thermalized and black body spectrum emerges. Beyond the photospheric radius, the observed spectrum may cease to be thermal (e.g., Rees & Mészáros 2005; Beloborodov 2010). In addition, the spectrum coming from the prompt phase is usually fitted by using a broken power law known as the Band function (Band et al. 1993).

From now on, we will consider the collapsar as the progenitor for our model and we will discuss further details in Chapter 4, where we explain the features of the pre-SN progenitor that we have chosen to perform our numerical simulations called the 16TI model (Woosley and Heger, 2006).

1.3 Previous Studies Concerning Angular Momentum and 3D Simulations

Angular momentum and the way it affects the evolution of the burst is the foremost point of this work. Several authors have done different approaches to study the progenitor's rotation via numerical simulations. In this section, we will summarize some works that constituted our project's starting point.

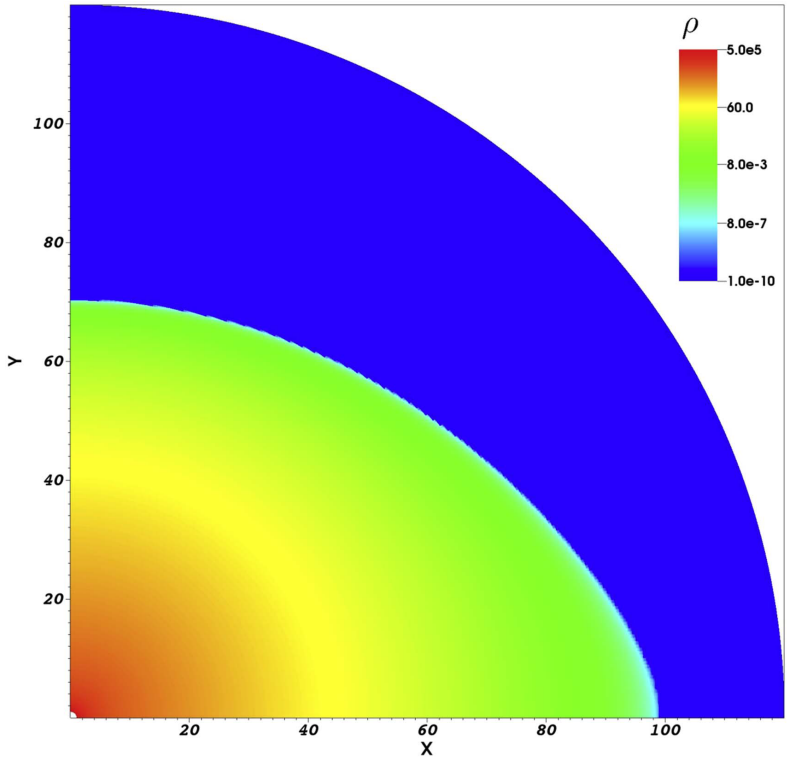

Figure 1.5: Density distribution (g cm^{-3}) of the rotating star found by Geng et al. (2016). The unit scale per dimension is 4×10^8 cm (Geng et al. (2016)).

Geng et al. (2016) performed 2.5D axisymmetric simulations (which are 2D simulations

where the rotation is followed) of a collapsing and rapidly rotating star with the emission of an intermittent jet of constant-luminosity. Here, they considered the 16TI model (Woosley & Heger, 2006) with an approximation on the rotation (Hachisu's Self-Consistent Field method[6]; Hachisu 1986a, 1986b) to obtain a 2D rotating star configuration. They focused on the inner region of the progenitor; their paper studied how the jet pierced and broke out from the star, so they assumed rigid rotation (Ω =5.3$\times^{-3}$ rad s^{-1}, which corresponds to an specific angular momentum $J_0 \sim 10^{18}$ cm^2 s^{-1}). In this work, the authors defined a critical rotation configuration $q = R_{\rm pol}/R_{\rm eq}$ (where $R_{\rm pol}$ is the polar radius and $R_{\rm eq}$, the equatorial radius) which corresponds to the maximum ratio reached at equilibrium between centrifugal force and gravity and it equals 0.7. In Figure 1.5, the initial condition implemented by Geng et al. (2016) is displayed. Here, the envelope of the progenitor is already modified into an oblate shape as a consequence of its angular momentum.

A revealing result found in Geng's study is that the envelope's oblate configuration is the main reason why the breakout time t_{bo} varies if the model does not include rotation. This result goes hand in hand with the objectives that were initially set in our project.

Nevertheless, a more complete study would involve different rotation distributions along the progenitor. Also, due to potentially high rotation rates, the star shape will be modified and different configurations can tell us a relation between the breakout time and progenitor's morphology.

To quantify the changes in the morphology of the star due to angular momentum, we considered the study developed by Maeder and Meynet (2012), where they stated a critical rotation value for the ratio between the polar radius and the equatorial radius ($R_{\rm pol}/R_{\rm eq}$). This critical rotation configuration is reached when the centrifugal force compensates central gravity. We take the value of this ratio ($\sim$ 0.7) as a reference for the maximum deformation of the progenitor, so we could inject the jet at this point of the evolution.

Also, López-Cámara et al. (2013) carried out relativistic hydrodynamic (RHD) simulations with adaptive mesh refinement (AMR) using 2D and 3D models to compare the jet and progenitor features, using 16TI pre-SN progenitor. Some common problems in 2D simulations will appear when the jet is launched and dense material accumulates in front of it. Figure 1.6 shows one of the main results from López-Cámara et al. (2013): the jet's evolution from a 3D low resolution model, which eventually breaks out from the star. In these simulations, Cartesian coordinates were used and the jet was solved with a resolution of, at least, Δ =2.5$\times 10^8$ cm. The injection radius of the jet was at $R_i = 10^9$ cm with an opening angle of θ_0=10$^{\rm o}$, a luminosity of L =5.33$\times 10^{50}$ erg s^{-1} and a Lorentz factor of $\Gamma_0 = 5$.

One of the results of López-Cámara et al. was that, even though the general properties of the jet will not be affected by dimensionality, the jet can break out from the star and propagate faster in 3D models than 2D ones because, unlike for the 2D models, the 3D jet can move around the vertical axis where there is less resistance.

[6]The Hachisu's Self-Consistent Field method is applied to generate equilibrium rotating initial configurations assuming a barotropic equation of state, self-gravitation via the Poisson equation and a rotation law (generally, rigid body rotation).

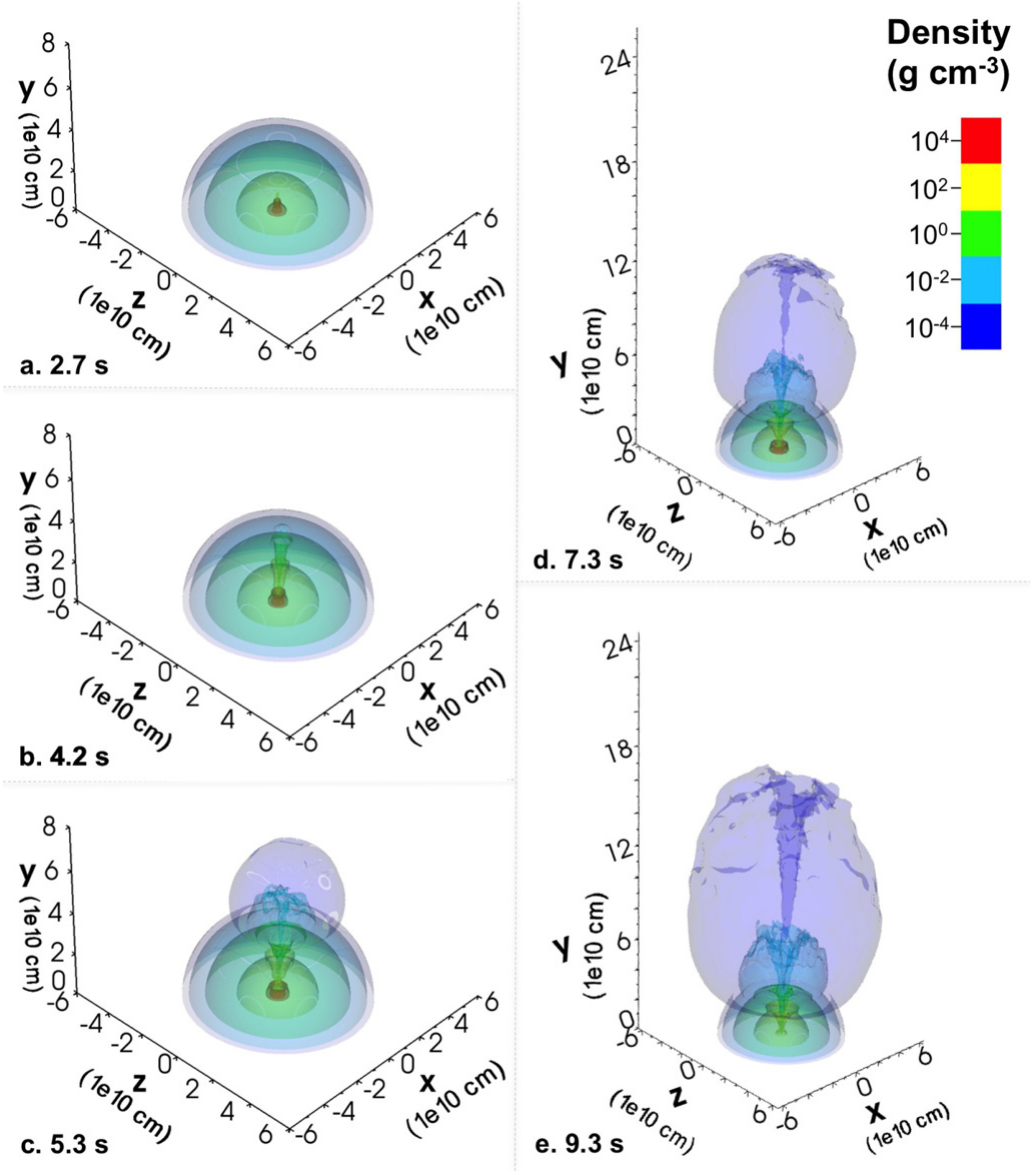

Figure 1.6: Density contours (g cm^{-3}) of a 3D jet model of low resolution images taken from López-Cámara et al. (2013). (Image credit: López-Cámara et al. (2013)).

Moreover, in another work made by López-Cámara et al. (2010), they studied angular momentum distribution by separating it into a radial and polar angle components $J = J(R)\Theta(\theta)$ with R the spherical radius and θ, the polar angle. They fixed $\Theta(\theta) = \sin^2\theta$ and varied the radial distribution (constant, linear and the one proposed in Woosley and Heger (2006) with a normalization factor).

Considering the settings applied in the previous studies, we came up with the following approach in order to improve the model of the progenitor with angular momentum: we recalled the angular momentum decomposition proposed by López-Cámara et al. (2010) which is $J = J_1(R)J_2(\theta)$ or, alternatively, $J = J_1(R)J_2(z)$. With this functional form, we now can propose different $J_2(\theta)$, distributions and study their effect on the morphology of the progenitor.

Also, we will implement the same angular velocity value found by Geng et al. (2016) so we could compare against their progenitor's initial condition where the jet is injected. Finally, we applied similar features for the values of the jet like López-Cámara et al. (2013), which are the most common jet conditions and it would help to establish control settings for the simulations.

On the whole, we will apply important ingredients, summarized from all these previous studies: 16TI model as the basis of the jet progenitor (which gathers important features associated with the collapsar model), 3D RHD simulations, AMR and a set of angular momentum distributions to study their effects in the progenitor and the jet's propagation.

2 — Theoretical Framework

In this chapter, we will discuss the basis of hydrodynamics, focused on a numerical relativistic implementation and how we introduced the angular momentum within the initial conditions.

Also, this chapter is devoted to explain the functioning of the code FLASH in its 4.3 version, which was used to perform our simulations.

2.1 Relativistic Hydrodynamics

Numerical simulations have been implemented over the years to study the behaviour of many astrophysical phenomena. The evolution of the jets coming from GRBs can be followed by applying the physics of hydrodynamics, assuming the jet can be described as a fluid (for instance that the mean free path of the particles is much smaller than the length of the spatial variations of the macroscopic variables).

Given that the jet is, basically, material coming from the center of a star which is moving rapidly through the stellar envelope with velocities close to the speed of light, we need to have a description of this phenomenon in the relativistic regime.

Recalling the non-relativistic Euler equations for a perfect fluid (disregarding viscous effects) in three dimensions (3D) in a fixed volume element, we have the following governing equations:

- **Mass conservation**

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{2.1}$$

  where ρ is the fluid density and $\mathbf{u}$=(v_x, v_y, v_z) is the velocity field.

- **Momentum conservation**

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \rho \mathbf{u}\mathbf{u} + \nabla P = f_{\text{ext}}, \tag{2.2}$$

  where P is the pressure and f_{ext} =(f_x, f_y, f_z) are the external forces per unit volume.

- **Energy conservation**

$$\frac{\partial E}{\partial t} + \nabla \cdot [\mathbf{u}(E + P)] = G - L + f_{\text{ext}} \cdot \mathbf{u}. \tag{2.3}$$

where E is the energy density and G,L represent the energy gain and loss respectively.

Note that the bold letters represent 3-dimensional vectors. These equations can be rewritten in vectorial form, considering a Cartesian coordinate system; we get:

$$\mathbb{U} = \begin{bmatrix} \rho \\ \rho v_{\mathrm{x}} \\ \rho v_{\mathrm{y}} \\ \rho v_{\mathrm{z}} \\ E \end{bmatrix}, \qquad \mathbb{F} = \begin{bmatrix} \rho v_{\mathrm{x}} \\ \rho v_{\mathrm{x}}^2 + P \\ \rho v_{\mathrm{x}} v_{\mathrm{y}} \\ \rho v_{\mathrm{x}} v_{\mathrm{z}} \\ v_{\mathrm{x}}(E+P) \end{bmatrix}, \qquad \mathbb{G} = \begin{bmatrix} \rho v_{\mathrm{y}} \\ \rho v_{\mathrm{y}} v_{\mathrm{x}} \\ \rho v_{\mathrm{y}}^2 + P \\ \rho v_{\mathrm{y}} v_{\mathrm{z}} \\ v_{\mathrm{y}}(E+P) \end{bmatrix}, \tag{2.4}$$

$$\mathbb{H} = \begin{bmatrix} \rho v_{\mathrm{z}} \\ \rho v_{\mathrm{z}} v_{\mathrm{x}} \\ \rho v_{\mathrm{z}} v_{\mathrm{y}} \\ \rho v_{\mathrm{z}}^2 + P \\ v_{\mathrm{z}}(E+P) \end{bmatrix}, \qquad \mathbb{S} = \begin{bmatrix} 0 \\ f_{\mathrm{x}} \\ f_{\mathrm{y}} \\ f_{\mathrm{z}} \\ G - L + f \cdot v_{\mathrm{x}} \end{bmatrix},$$

where the double letters represent a 5-dimensional vector. Simplifying into one expression, we obtain:

$$\frac{\partial \mathbb{U}}{\partial t} + \frac{\partial \mathbb{F}}{\partial x} + \frac{\partial \mathbb{G}}{\partial y} + \frac{\partial \mathbb{H}}{\partial z} = \mathbb{S}, \tag{2.5}$$

where the vector $\mathbb{U}$ represents the so called conserved variables of the system; $\mathbb{F}$, $\mathbb{G}$ and $\mathbb{H}$ are the fluxes in x, y and z directions and $\mathbb{S}$ gathers source terms.

For relativistic hydrodynamics (RHD), we have to change the conserved variables vector $U = U(\rho, \mathbf{u}, P)$ into a new vector of relativistic conserved variables $V = V(D, \mathbf{m}, \varepsilon)$:

$$D = \Gamma\rho \qquad \mathbf{m} = \rho h \Gamma^2 \mathbf{u} \qquad \varepsilon = \rho h \Gamma^2 - P, \tag{2.6}$$

where D is the fluid density, $\mathbf{m}$ is the momentum density, ε is the total energy density and Γ is the Lorentz factor given by $\Gamma = (1 - \mathbf{u}^2)^{-1/2}$. Natural units are considered in this equations ($c = 1$, where c is the speed of light). Neglecting the source terms $\mathbb{S}$ and using the functional form shown in (2.9), we obtain the RHD equations:

$$\frac{\partial}{\partial t} \begin{bmatrix} D \\ \mathbf{m} \\ \varepsilon \end{bmatrix} + \nabla \cdot \begin{bmatrix} D\mathbf{v} \\ \mathbf{m}\mathbf{v} + P\mathbf{I} \\ \mathbf{m} \end{bmatrix} = 0. \tag{2.7}$$

This description is very convenient to translate the RHD equations into an RHD code.

The equation of state (EOS) will close the system of conservation laws, since we have a set of 5 differential equations and 6 unknown variables (E, P, $\mathbf{u}$, ρ); the selection of the EOS must represent the behaviour of the macroscopic variables of our system.

Generally, to simplify the solution of the equations and to give a first approach for characterize the system, the EOS is taken as an ideal gas approximation in the Newtonian regime, where the pressure is written as,

$$P = (\gamma - 1)\rho\epsilon \tag{2.8}$$

where γ is the adiabatic index and ϵ is the specific internal energy. A more complex EOS would content several species of ideal gases with different values for γ. Also is common to take a polytropic equation of state with a general form of

$$P = \kappa\rho^{\gamma} \tag{2.9}$$

where κ is a constant which depends on the kind of regime we are working on and the pressure has a power-law dependence on density; the values adopted by γ can be either 4/3 or 5/3 for the relativistic and no-relativistic description, respectively.

Particularly, FLASH 4.3 (Section 2.4) supports only one ideal relativistic equation of state, with the following expression for the specific enthalpy:

$$h = 1 + \frac{\gamma}{\gamma - 1}\frac{P}{\rho}, \tag{2.10}$$

where P is the pressure, ρ is the density and γ is the specific heat ratio (do not confuse with Lorentz factor Γ). In our case, we applied a γ value of 4/3 since our gas belongs to a relativistic regime.

2.2 Relativistic Rankine-Hugoniot Jump Conditions

So far, we have introduced the hydrodynamic treatment for a perfect gas, which is the first approach to study a GRB as a fluid, and as said before, it is necessary to invoke the relativistic regime if bulk velocities are close to the speed of light (Zhang, 2018).

Using the equations described above, we can find a group of conditions that relates the fluid properties to shock variables and they are implemented in semi-analytic and numerical blast waves studies. These are called the shock jump conditions or Rankine-Hugoniot conditions. Throughout this section, we will obtain the form of jump conditions in a relativistic scheme.

Recalling the description made by Uhm (2011), we consider the scheme of a blast wave moving through an ambient medium (Figure 2.1). In the following expressions we will denote the pre-shock region with a (1) sub-index and the post-shock region with a (2) sub-index.

Writing the conservation equations in a simplified form in the shock frame, we obtain:

$$\gamma_2\beta_2\rho_2 = \gamma_1\beta_1\rho_1, \tag{2.11}$$

$$\gamma_2^2\beta_2(e_2 + p_2) = \gamma_1^2\beta_1\rho_1c^2, \tag{2.12}$$

$$\gamma_2^2\beta_2(e_2 + p_2) + p_2 = \gamma_1^2\beta_1\rho_1c^2, \tag{2.13}$$

where c is the speed of light, ρ is the rest-mass density, e the energy density, p the pressure, γ is the gas Lorentz factor for each region and $\beta = (1 - 1/\gamma^2)^{1/2}$. Assuming that the pre-shock region is cold ($p_1 = 0$, $e_1 = \rho_1c^2$) and writing the relativistic EOS as $p_2 = k_2(e_2 - \rho_2c^2)$ in terms the parameter k_2 whose value varies between 1/3 and 2/3, we can solve the system of Equations

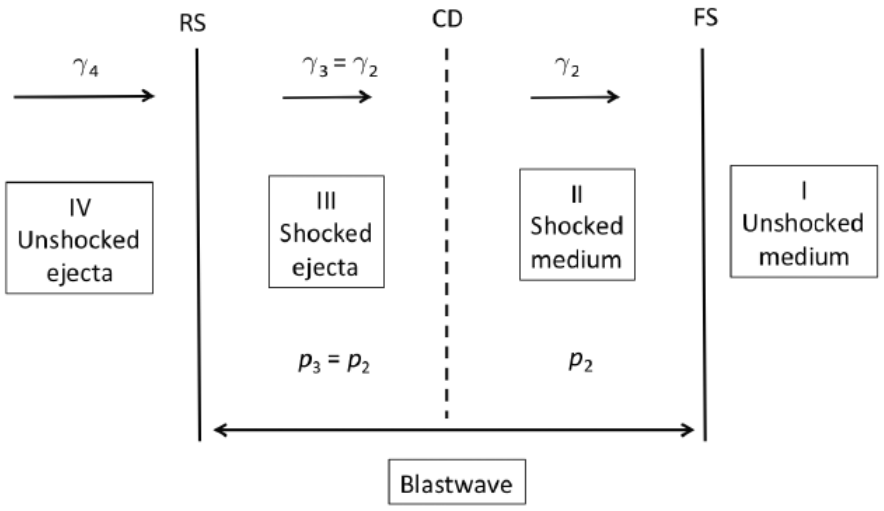

Figure 2.1: Representation of two fluids colliding. The region between the forward shock (FS) and reverse shock (RS) is called the blast wave. The shocked medium and shocked ejecta regions are separated by a contact discontinuity. (Image credit: Zhang, 2018)

(2.11)-(2.13).

Introducing the compression ratio $a = \rho_2/\rho_1$ and simplifying, we found the expressions for γ_2^2, γ_1^2 and $p_2/\rho_1 c^2$ in terms of k_2:

$$\gamma_2^2 = \frac{(a+1)}{a(1-k_2^2)+(1+k_2)^2}, \qquad \gamma_1^2 = \frac{(a+1)[ak_2-(1+k_2)]^2}{a(1-k_2^2)+(1+k_2)^2}, \tag{2.14}$$

$$p_2/\rho_1 c^2 = \frac{(ak_2)^2-(ak_2)(2+k_2)}{(1+k_2)}. \tag{2.15}$$

The next step is to define two more parameters. These are the relative velocity $\beta_{12} = (\beta_1-\beta_2)/(1-\beta_1\beta_2)$ and the relative Lorentz factor γ_{12},

$$\gamma_{12} = (1-\beta_{12}^2)^{-1/2} = (1-\beta_1\beta_2)\gamma_1\gamma_2 = \gamma_1\gamma_2 - [(\gamma_1^2-1)(\gamma_2^2-1)]^{1/2}, \tag{2.16}$$

and, by using Equation (2.14), we can write γ_{12} in terms of k_2,

$$\gamma_{12} = \frac{(ak_2-1)}{(1+k_2)}; \tag{2.17}$$

then, we obtain the equation for the compression ratio, as,

$$a = \frac{\rho_2}{\rho_1} = \frac{(1+k_2)\gamma_{12}+1}{k_2}. \tag{2.18}$$

Substituing Equation (2.18) in Equations (2.14) and (2.15), we have,

$$\gamma_1^2 = \frac{(\gamma_{12}+1)[(1+k_2)\gamma_{12}-k_2]^2}{(1-k_2^2)\gamma_{12}+(1+k_2^2)}, \tag{2.19}$$

$$\gamma_2^2 = \frac{(\gamma_{12}+1)}{(1-k_2^2)\gamma_{12}+(1+k_2^2)} \tag{2.20}$$

$$p_2 = (\gamma_{12} - 1)[(1 + k_2)\gamma_{12} + 1]\rho_1 c^2. \tag{2.21}$$

Finally, taking $k_2 = 1/3$ and the limit when $\gamma_{12} \to \infty$ (ultra-relativistic case), we get,

$$\gamma_1^2 = 2\gamma_{12}^2, \qquad \gamma_2^2 = \frac{9}{8} \tag{2.22}$$

$$p_2 = \frac{4}{3}\gamma_{12}^2 \rho_1 c^2, \qquad a = 4\gamma_{12}; \tag{2.23}$$

noting that $\gamma_{12} = \gamma_1/\sqrt{2}$ from Equation (2.22), we rewrite the expressions and we obtain the shock conditions,

$$\gamma_2^2 = \frac{3}{2\sqrt{2}}, \qquad a = \frac{4}{\sqrt{2}}\gamma_1, \tag{2.24}$$

$$p_2 = \frac{2}{3}\gamma_1^2 \rho_1 c^2. \tag{2.25}$$

where the only free parameter is the Lorentz factor for region 1, γ_{12}.

The description above has been built by assuming a blast wave with two components: the shocked regions between the forward shock (FS) and reverse shock (RS). However, we can work it out differently assuming that the blast moves with a common Lorentz factor Γ (based on a result from Kobayashi and Sari (2000), which shows that $\Gamma \approx$ constant in the blast wave). We can find similar expressions for the density of the ejecting material by imposing radial symmetry in the conservative equations.

Obtaining the initial profile of ejecta density $\rho_{\rm ej}$,

$$\rho_{\rm ej}(\tau, r) = \frac{L_{\rm ej}}{4\pi r^2 v_{\rm ej} \Gamma_{\rm ej}^2 c^2} \tag{2.26}$$

which depends on the ejection time τ and cylindrical radius r. If we consider an opening angle θ_0, we can rewrite $r = R_{\rm i} \tan\theta_0$, where R_i is the injection radius and θ_0 is the half-opening angle. Here, $v_{\rm ej} = c(1 - 1/\Gamma_{\rm ej}^2)^{1/2}$ is the velocity of the ejecta, $\Gamma_{\rm ej}$ represents the Lorentz factor of the ejecta and $L_{\rm ej}$ is the luminosity, which depends on the mass flow rate $\dot{M}(\tau)$,

$$L_{\rm ej} = \Gamma_{\rm ej} \dot{M} c^2. \tag{2.27}$$

All calculations are valid even if we assume a small opening angle θ_0, as long as $\Gamma_{\rm ej} \gg \theta^{-1}$. Finally, by assuming a pressure balance in the blast wave and $\Gamma_{\rm ej} \gg 1$ (relativistic approximation), one can find a relation between the instantaneous Lorentz factor Γ and $\Gamma_{\rm ej}$,

$$\Gamma = \Gamma_{\rm ej} \left[1 + 2\Gamma_{\rm ej} \left(\frac{\rho_{\rm l}}{\rho_{\rm ej}} \right)^{1/2} \right]^{-1/2} \tag{2.28}$$

where $\rho_{\rm l}$ is the ambient medium ahead the FS.

2.3 Specific Angular Momentum Implementation

In this section, we discuss briefly a mathematical procedure to include an angular momentum distribution in Cartesian coordinates.

First, we recall a simple expression for the specific angular momentum:

$$\mathbf{J} = \mathbf{R} \times \mathbf{v}, \tag{2.29}$$

where $\mathbf{J}$ is the angular momentum, $\mathbf{R}$ is the spherical radius and $\mathbf{v}$ is the velocity (Figure 3.2). If we assume that the vectors $\mathbf{R}$ and $\mathbf{v}$ satisfy that $\mathbf{R} \perp \mathbf{v}$, then we come across with a relation between $\mathbf{J}$ and the tangential velocity v_φ; that is:

$$\mathbf{R} \perp \mathbf{v} \qquad \Rightarrow \qquad J = Rv_\varphi \tag{2.30}$$

On the other hand, the tangential velocity can be written using the angular velocity Ω:

$$v_\varphi = \Omega \times \mathbf{r}, \tag{2.31}$$

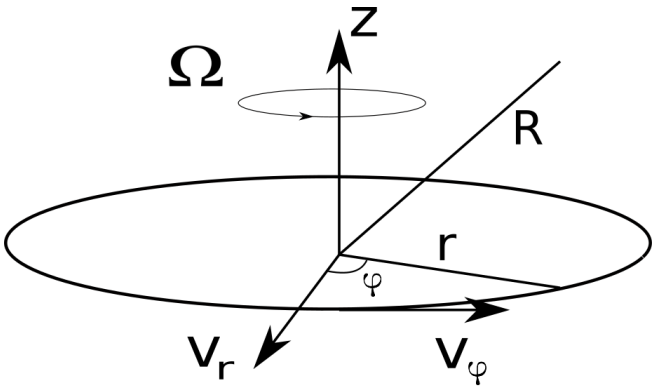

Figure 2.2: Angular velocity scheme.

with $\mathbf{r}$ the cylindrical radius. Again, assuming that the angular velocity is perpendicular to the cylindrical radius,

$$\Omega \perp \mathbf{r} \qquad \Rightarrow \qquad v_\varphi = \Omega r = \Omega\sqrt{R^2 - z^2}; \tag{2.32}$$

substituting equation (2.32) in equation (2.30), we obtain

$$J = \Omega R r = \Omega R\sqrt{R^2 - z^2}. \tag{2.33}$$

This expression allows us to write the specific angular momentum in terms of a constant angular velocity, which implies rigid rotation. If we simplify the expression (2.33) by setting $z = 0$, we obtain that

$$J(r, z = 0) = \Omega r^2 \tag{2.34}$$

so the specific angular momentum increases as the square of cylindrical radius r in the equator. Now, using equation (2.30), we can take the tangential velocity in terms of J, that is $v_\varphi = J/R$.

For Cartesian coordinates, we need to transform the tangential velocity into the Cartesian components of velocity v_x and v_y. A relation between the velocity components in polar and Cartesian coordinates can be found with a simple transformation:

$$\begin{cases} u = v_x = v_{\mathrm{r}} \cos\varphi - v_\varphi \sin\varphi \\ v = v_y = v_{\mathrm{r}} \sin\varphi + v_\varphi \cos\varphi, \end{cases} \tag{2.35}$$

where v_r is the radial velocity. In this case, this kind of motion is irrelevant within the GRB progenitor, so we set $v_r = 0$ and rewrite (2.35) using J as follows:

$$\begin{cases} v_{\mathrm{x}} = -J \sin\varphi / R \\ v_{\mathrm{y}} = J \cos\varphi / R. \end{cases} \tag{2.36}$$

By substituting equation (2.33) in (2.36), we obtain the expressions for the velocity components in terms of constant angular velocity. Using that $\sin\varphi = y/r$ and $\cos\varphi = x/r$, we have

$$\begin{cases} u = v_{\mathrm{x}} = -\Omega y \\ v = v_{\mathrm{y}} = \Omega x. \end{cases} \tag{2.37}$$

However, we also wanted to include the effects of the angular momentum distribution in the vertical direction. In Chapter 4 we will explain the angular momentum decomposition into $J(R, z) = J(R)J(z)$ applied in the stellar rotation.

3 — The Code: FLASH 4.3

FLASH (http://flash.uchicago.edu/site/flashcode/; Fryxell et al., 2000) is a publicly available modular code that, within its capabilities, solves hydrodynamic (HD), magnetohydrodynamics (MHD) and relativistic hydrodynamic (RHD) problems. Its architecture allows the user to apply different modules in each simulation such as adaptive mesh refinement (AMR), different Riemann solvers (HLL, HLLC, etc.), parallelization processes, etc. In this work, we used the 4.3 version of this code.

In this section, we will talk about these modules and their main features. Also, some verification tests are studied to assess the functioning of the code.

3.1 Setting Up a New Problem with the RHD Module

As mentioned before, one of FLASH capabilities is to solve RHD problems; this is the main implementation used to carry out our simulations. Here, we explain how to set up a new problem using the FLASH RHD module.

When FLASH is installed on a computer or cluster, it generates a set of different directories, each containing information about the executables and the simulation parameters. We are going to focus on the *source* directory.

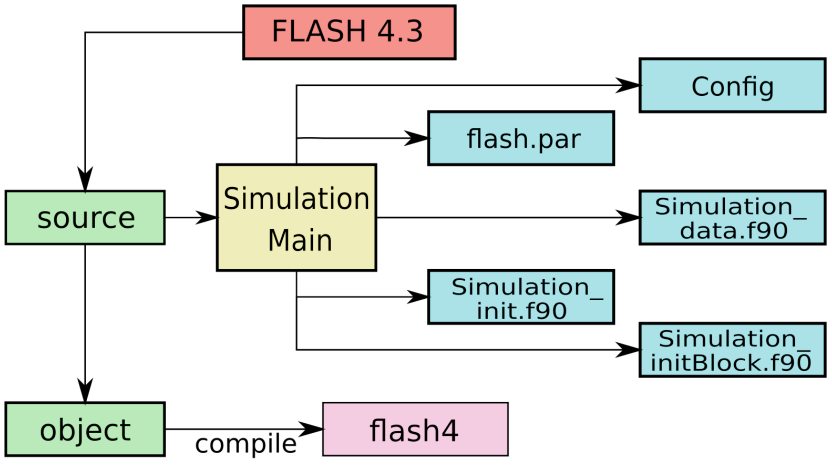

Figure 3.1: FLASH basic structure to configure a new simulation

The *source* directory contains modules that have modifiable scripts for both physical or numerical tools. The numerical tools consist of the grid parameters, drivers, file formats, among others. The physical tools establish the relevant physical problem within the code; for instance the inclusion of relativity, equation of state (EOS), self-gravity, physical constants, etc.

To initialize a new problem, we have to work with the source directory, in order to add the modified scripts with our specifications. After we gather these files, we need to place this information inside a directory called *Simulation Main*, which is located inside the source directory. Then, we can compile the code using the object directory, where the executable is generated. A flowchart of this sequence is shown in Figure 3.1.

The files required in the Simulation Main directory are listed below:

- **Config**

  In the *Config* file we specify the route of the physical and numerical tools to be applied in our simulations. This way, the executable becomes more efficient. An example of the Config file can be seen in Appendix A.

  Also, in this file we can add some values used to build up the initial condition for the problem. For example, in our case, we can define the density, pressure and velocity of the interstellar medium (ISM). If we do not specify these parameters in *flash.par* file, the code will take this values as *default* values for every simulation.

- **flash.par**

  The *flash.par* file contains the runtime and initial parameters for the model (e.g. jet specifications, ISM and stellar values for density, pressure and velocity). Also, here we must specify the gas ratio of specific heats γ, the boundary conditions (which can be *user defined*, depending on the type of problem), the checkpoint and plot files parameters, the Riemann solver, size of the computational volume, the plot variables, the Courant-Friedrichs-Lewy value (or CFL, which is a parameter used to find stable time steps for the simulation), the maximum time to run the simulation and the adaptive mesh refinement values and variables (if necessary). All variables are in the cgs system.

  This file can be modified without recompiling the code, this means that we can modify some of the initial values to run a new simulation with the same executable. See Appendix B for more details.

- **Simulation_data.f90**

  In *Simulation_data.f90* the local data for the simulation setup are stored. The code gets the values from the *flash.par* file needed to run the simulation. Also, here are included the headers used to call the constants for the simulation. In Appendix C, an example for this script is included.

- **Simulation_init.f90**

Simulation_init.f90 script initializes all the data specified previously in *Simulation_data.f90*. Another important feature for this script is to rename the variables defined in the *flash.par* file, if the user needs to. An example of this script can be found in Appendix D.

- **Simulacion_initBlock.f90**

  Simulacion_initBlock.f90 allows us to create the initial condition for the simulation with the data stored in the previous files. The size of each coordinate is defined and it iterates all over the domain to assign the parameter values to the domain cells. See Appendix E for more details.

In our case, two additional scripts were implemented. The first one is used to apply specific boundary conditions, so we can enable the jet injection. This script is *Grid_bcApplyToRegionSpecialized.f90* and it is applied when the problem requires an specific type of condition in certain domain cells (in other words, when there is not a general condition such as *outflow, periodic* or *reflect*). The details about the boundary conditions are explained in Chapter 4.

The second additional script is called *Grid_markRefineDerefine.f90*. The purpose of this file is to set up an adaptive mesh which will refine in the zones specified by the user; this way, we can apply the refinement where needed and reduce it in unnecessary zones. In our case, we needed the highest refinement on the jet's evolution path, along the z axis. The specifications of this implementation are discussed in Chapter 4.

3.2 Adaptive Mesh Refinement

Adaptive mesh refinement is a technique applied in simulations to improve the solution at certain grid regions where a variable gradient is important. Given an initial number of grid divisions (or blocks), the code is able to refine each one of them hierarchically so that we get a better resolution where needed. The number of refinement levels is defined by the user.

In FLASH 4.3, the AMR grid is implemented with the PARAMESH library (MacNeice et al. 1999), which is the default package in this code. The data structure of this kind of AMR consists of an arrangement of *blocks of cells* that are indexed, this means each cell has a block identifier with its processor number and local block number.

The grid defined in this scheme has a set of these blocks that are related with each other with a tree data structure, where the blocks at the root have the largest cells and their children have smaller cells.

In the setup file *flash.par*, we can choose the variables with respect to which to refine (i.e density, pressure, velocities and/or energy) and also the refinement levels.

In Figure 3.2, we show an example of an AMR mesh applied to the RHD Riemann problem in 2D. Here, we have a comparison of a fixed mesh with a low refinement value ($\Delta = 0.12$ cm resolution), a higher one ($\triangle = 0.02$ cm resolution) and the AMR mesh. It is noticeable that the AMR is applying the highest refinement along the diagonal, where the oblique shocks appear

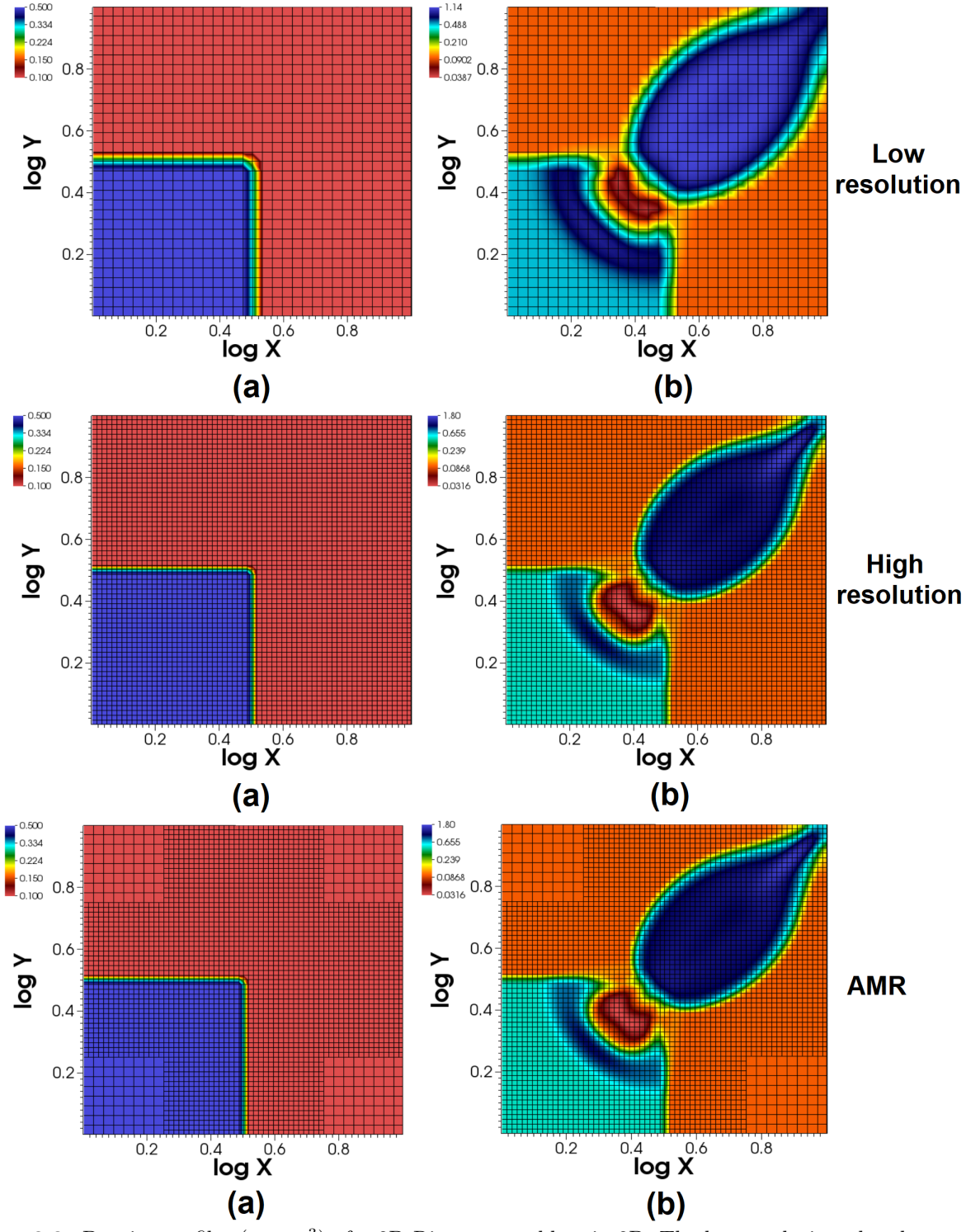

Figure 3.2: Density profiles (g cm^{-3}) of a 2D Riemann problem in 2D. The low resolution plots have a mesh with 32×32 blocks ($\triangle = 0.12$ cm resolution), the high resolution plots consist of a mesh with 64×64 blocks ($\triangle = 0.02$ cm resolution) and the AMR mesh has these values as the lowest and highest resolution limits, respectively. The plots show (a) the initial configuration at $t = 0$ and (b) the evolution at $t = 0.4$.

after the evolution of the initial conditions (see Section 3.2). The limits taken in the AMR example are the values from the fixed meshes mentioned previously.

As we can see, one of the main advantages of the AMR is the ability to obtain accurate results, comparable with those obtained from a high-resolution mesh, in a more effective and faster manner. In this example, the fixed mesh spent about 50% more execution time than the AMR mesh.

3.3 Parallelization

Parallelization is an important feature we have to take into consideration when working with every computational task. Generally, the running time of common serial processing simulations is high because of aspects such as high resolution, number of iterations, etc. This is why, a well-programmed parallelization method can be used in order to take advantage of modern computers and, thus, to reduce the running time.

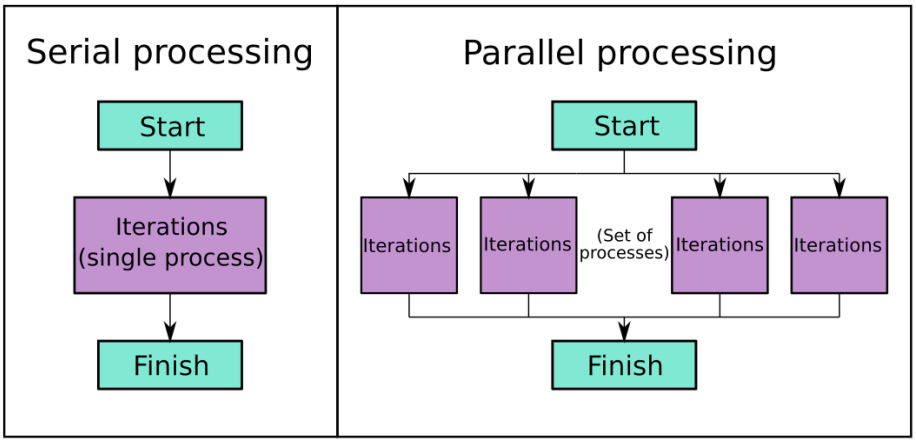

Figure 3.3: Differences between serial and parallel processing.

FLASH 4.3 implements different ways to apply parallelization using Message-Passing Interface (MPI), OpenMP or a hybrid application between these libraries (MPI+OpenMP). For this task, we consider MPI. In the following lines, we will show further information about this implementation.

MPI is a library which can be used in different programming languages (C, FORTRAN, C++) and is intended to distribute the computational workload using all available processors via message-passing (as its name implies).

The bases of message-passing are the *point-to-point* communications. That is to say, using this feature, a group of processes classified by rank are able to establish communication with one another. They can exchange information by sending and receiving operations; the messages

that contain the operations are distinguishable because they have a *tag*. So, using the *rank* and the *tag*, the processes are able to handle the data properly.

If now we need to share information with every process, it is necessary to write a code capable of managing optimally all the information received and sent by all processes. This is the application of MPI libraries, to control effectively communications.

In our case, we worked with Miztli supercomputer to perform the simulations. The parallelization was implemented with a special script where we have to specify the necessary number of processors. An example can be found in Appendix H.

3.4 Miztli Supercomputer

We ran all simulations on Miztli, which is the UNAM supercomputer. It is capable of executing 118 trillions arithmetic operations per second (118, TFlop/s). Some of its specifications are the following:

- 8 440 processing cores (contained in 664 Intel E5-2670v1 processors, 116 Intel E52660v3 processors, 116 Intel E5-2683v4 processors and 8 Intel E7 8860v4 processors).
- 16 NVIDIA M2090 video cards.
- RAM total memory of 47,232 GB.
- Massive storage of 216 TB.

Also, Miztli works with a queuing system for users; depending on the resources assigned to each user, they can send their works to one of the queues.

For our simulations, a set of 32-64 processors were used (depending on the model). Depending on the maximum level of refinement, the run time varied between 1 to 9 hours approximately (visually that the highest refinement level implies longer execution times). This running times are reachable if the adequate configuration is implemented, for example, by reducing the high refinement zones with the adaptive mesh and a suitable Courant number. Many tests were performed to find the adequate parameters.

3.5 Verification tests

Some tests to assess the proper functioning of the FLASH code are shown in this section. We select these tests based on their repeatability and the possibility of apply a rotated system on them. First, we will show the results from the extrapolation of a 1D blast wave problem into a 3D domain and we will compare the outcome against with the original results from the 1D test. Next, a 3D treatment and comparison of a 2D Riemann test is discussed. Finally, we will present a 2D jet-based test which will also be compared with previous results and a 3D treatment of this jet test will be described.

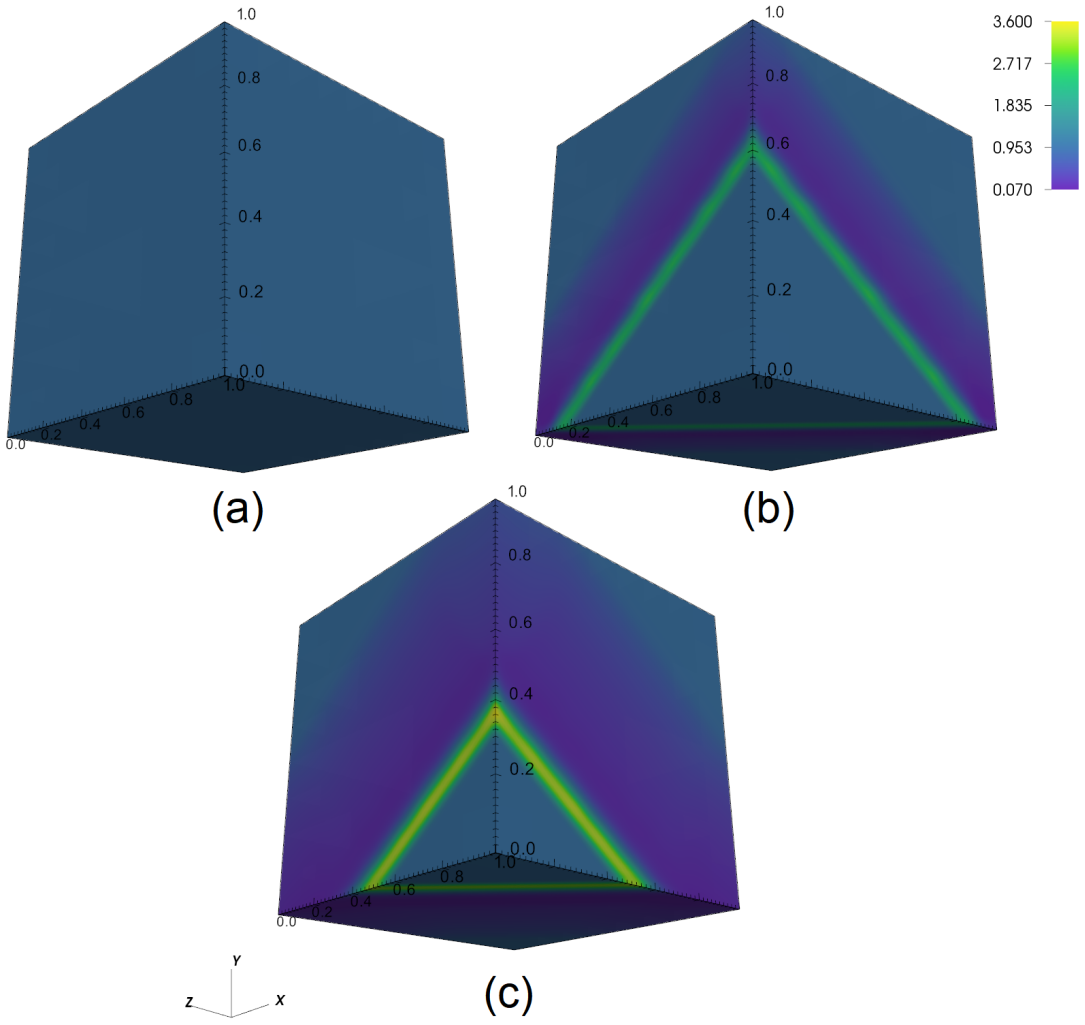

Figure 3.4: Density profiles (g cm^{-3}) for the 3D blast wave test for (a) t=0, (b) t=0.176 and (c) t=0.35.

3.5.1 3D Blast Wave

The initial setup for this cartesian 1D shock tube establishes a discontinuity at $x = 0.5$ as follows:

$$(\rho, v_{\rm x}, P) = \begin{cases} (1, 0, 10^3), & x < 0.5 \\ (1, 0, 10^{-2}), & x > 0.5 \end{cases} \tag{3.1}$$

Results based on this test can be found at Del Zanna and Bucciantini (2002), Mignone and Bodo (2005), etc. This test is appropriate to assess the solvers for RHD problems, as the blast reaches a Lorentz factor of $\Gamma \sim 6$. We expect with this test to follow the shock evolution, as created by the pressure gradient.

Now, since we want to extend this problem into a three dimensional test, we apply a 45° rotation around the y-axis and, then, another 45° rotation with respect to the z-axis so that the discontinuity is oblique to the grill. The runtime parameters used for the simulation are a CFL=0.2 and it was performed over a grid with 64 cells per axis. Figure 3.4 shows the evolution of the shock with density maps for t=0, t=0.176 and t=0.35. The rotation in the

initial condition sets a tetrahedron in the 3D domain, density and velocities have equal values in each zone as we expressed in Equation 3.1.

To compare our results with the known 1D profiles, we took, first of all, a slice in the y=0 plane as seen in Figure 3.5. After that, we project again along the diagonal and obtain the 1D curves for each variable.

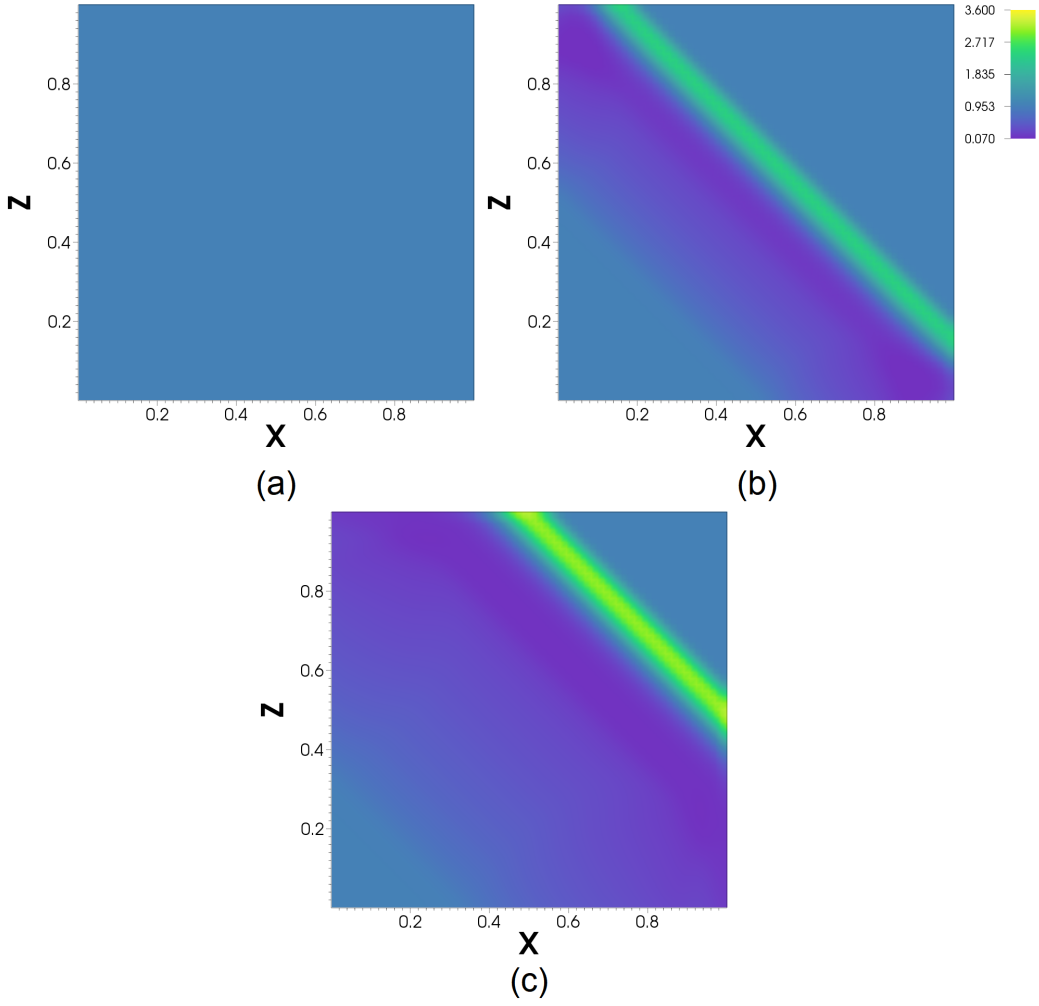

Figure 3.5: Density profiles (g cm^{-3}) for the blast wave in y=0 plane for (a) t=0, (b) t=0.176 and (c) t=0.35.

In Figure 3.6, we show the 1D graphs obtained from the 3D results at t=0.35. The 1D results from Del Zanna and Bucciantini (2002) are displayed in Figure 3.7.

Now, we compare the results from Figure 3.6 and Figure 3.7. For each case, different resolution and CFL values were applied. Furthermore, the geometry of the 3D setup certainly affects the resulting curves. However, the behaviour of each variable is quite similar comparing with Del Zanna and Bucciantini's results.

3.5.2 Rotated RHD Riemann Test

Moving on to the next test, we took the RHD Riemann 2D test and we make an a similar setup as in the previous case of the rotated 1D blast wave. With this test, we were able to study the

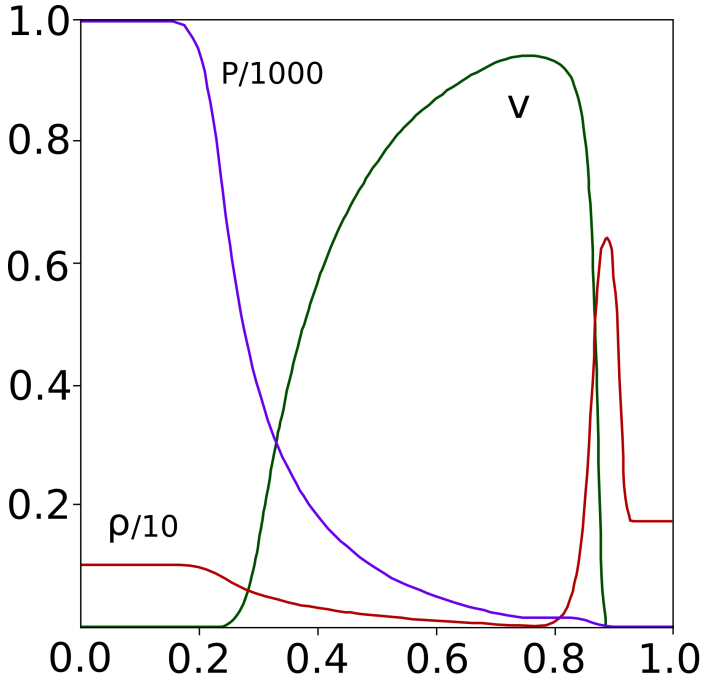

Figure 3.6: 1D profiles of pressure, velocity and density at t=0.35.

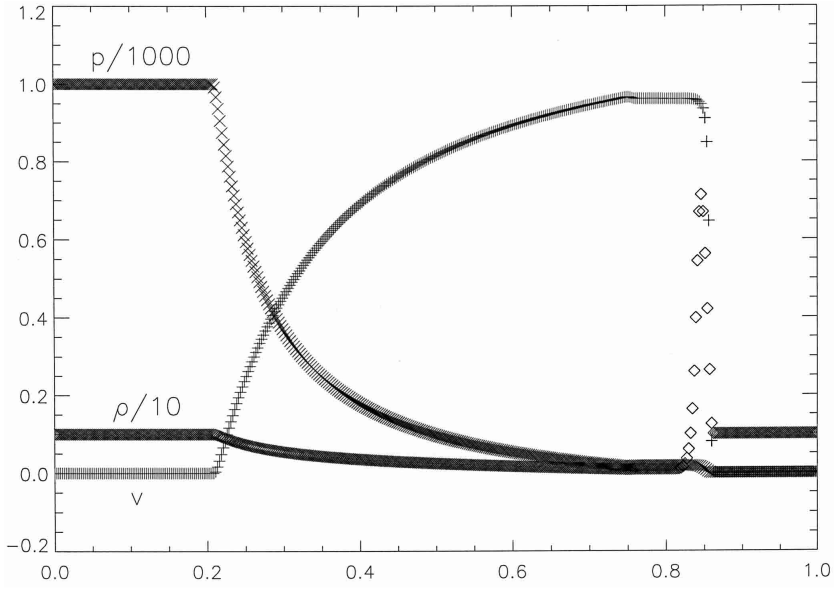

Figure 3.7: Results from Del Zanna and Bucciantini (2002) for the 1D relativistic blast wave at t=0.35. P,ρ and v are scaled with different factors. (Image credit: Del Zanna and Bucciantini (2002).

interactions of curved shocks and contact discontinuities. Specifically, we run a Riemann 2D test simulation with the following initial conditions:

$$(\rho, v_{\rm x}, v_{\rm y}, P) = \begin{cases} (0.1, 0, 0, 0.01), & x, y > 0, \\ (0.1, 0.99, 0, 1), & x < 0 < y, \\ (0.5, 0, 0, 0.1), & x, y < 0, \\ (0.1, 0, 0.99, 1), & y < 0 < x, \end{cases} \tag{3.2}$$

using Cartesian coordinates (Del Zanna and Bucciantini, 2002). The integration was carried out up to t=0.4 with a CFL=0.4. The computational domain was set with 125 equidistant cells per dimension, giving a mesh of 125×125 cells.

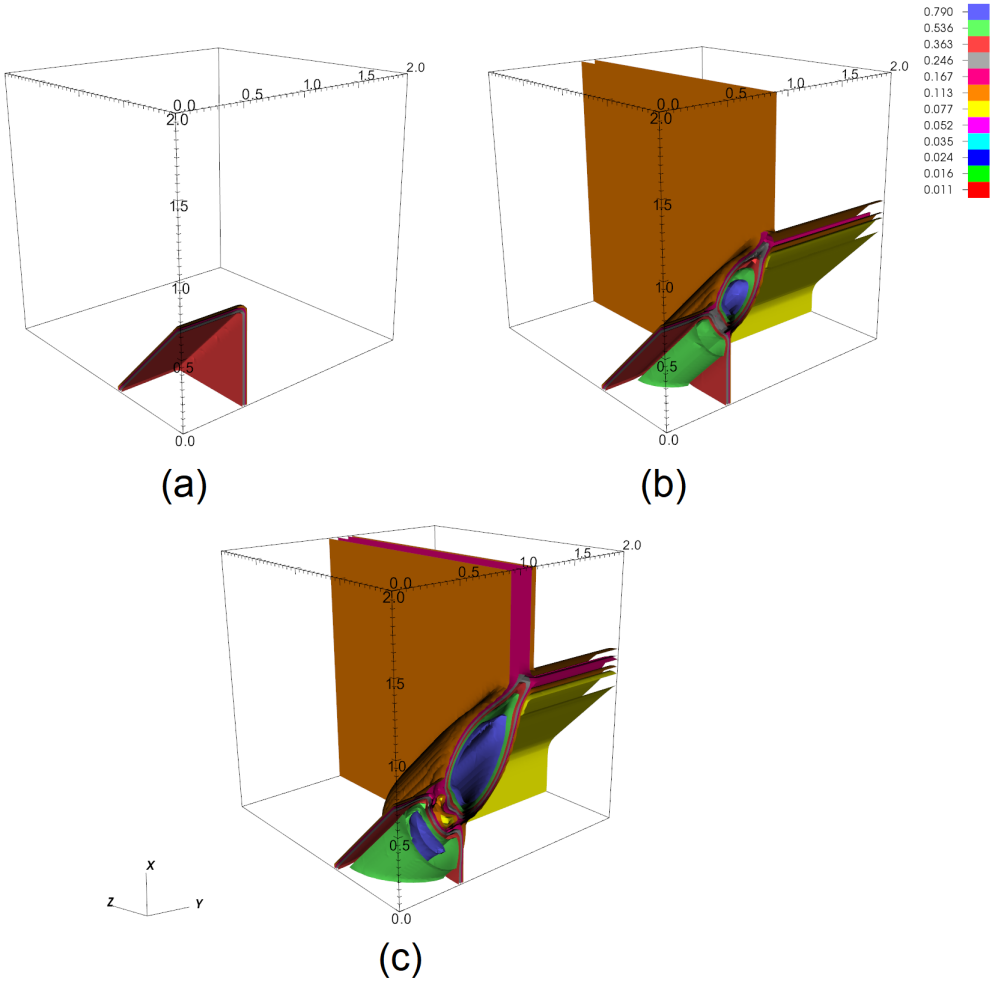

Figure 3.8: Density isosurfaces (g cm^{-3}) in a 3D grid using an extended region for (a) t=0, (b) t=0.2 and (c) t=0.4.

The 3D setup involved the rotation of this plane around the y-axis by a 45° angle. The maximum integration time was also t=0.4 and the simulation had a Courant number CFL=0.4 with a grid of 64 cells per dimension. Figure 3.8 shows the evolution of the density iso-surfaces in a 3D map.

We can compare first our result in Figure 3.9 (which is a 2D cut of the results in Figure 3.8 onto the y=0 plane) for t=0.4 with Del Zanna and Bucciantini's plot in Figure 3.10. The main difference between these tests are the lines that appear in the lower left quadrant. These are due to the type of solver (method applied to computed numerical fluxes across a discontinuity) used in the integration (Del Zanna and Bucciantini, 2002). Del Zanna and Bucciantini report their results using a LLF solver and they recommend to use a Roe-type solver to avoid the spurious waves that lead to these errors, so we did as suggested.

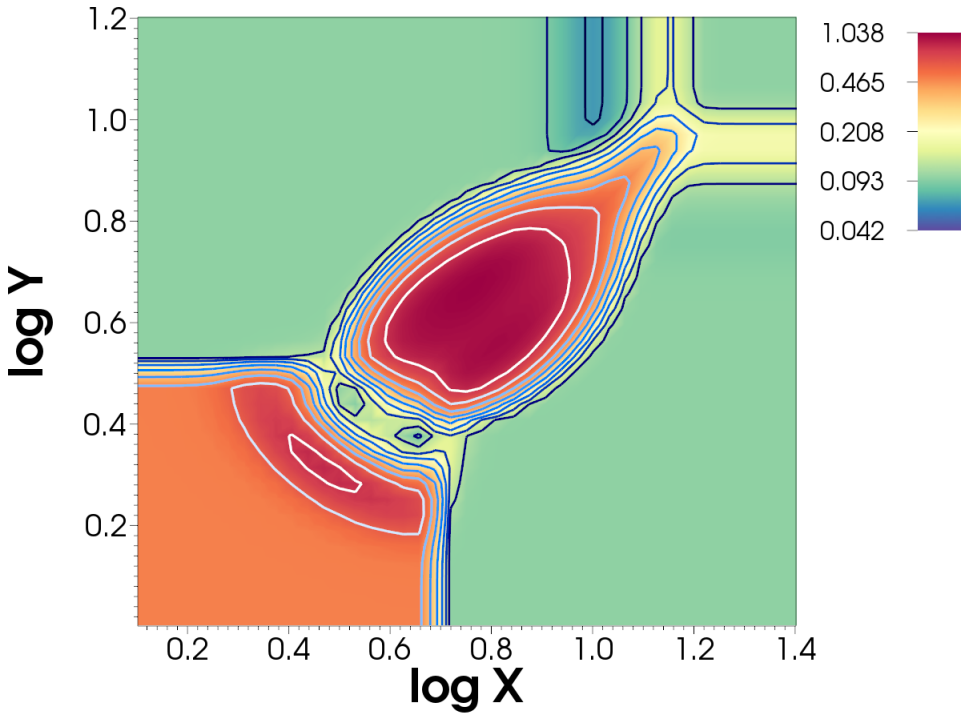

Figure 3.9: Density profile (g cm^{-3}) in the y=0 plane.

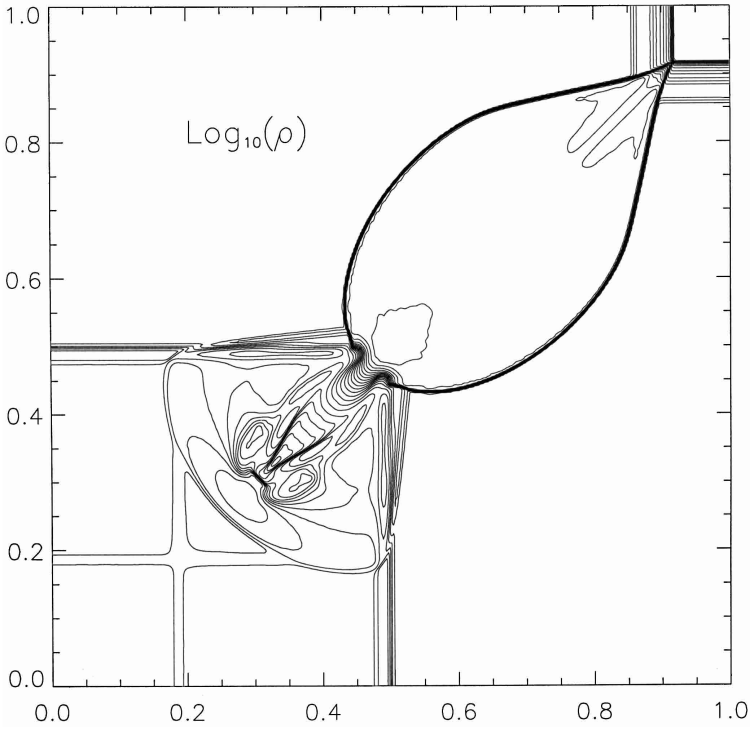

Figure 3.10: Density iso-contours (g cm^{-3}) in logarithmic scale at t=0.4 from Del Zanna and Bucciantini (2002). (Image credit: Del Zanna and Bucciantini (2002).

It is noticeable that, across the domain, the structure conserves the curved shock fronts in the lower left quadrant and the initial discontinuities in the upper right quadrant.

3.5.3 Jet-based Test

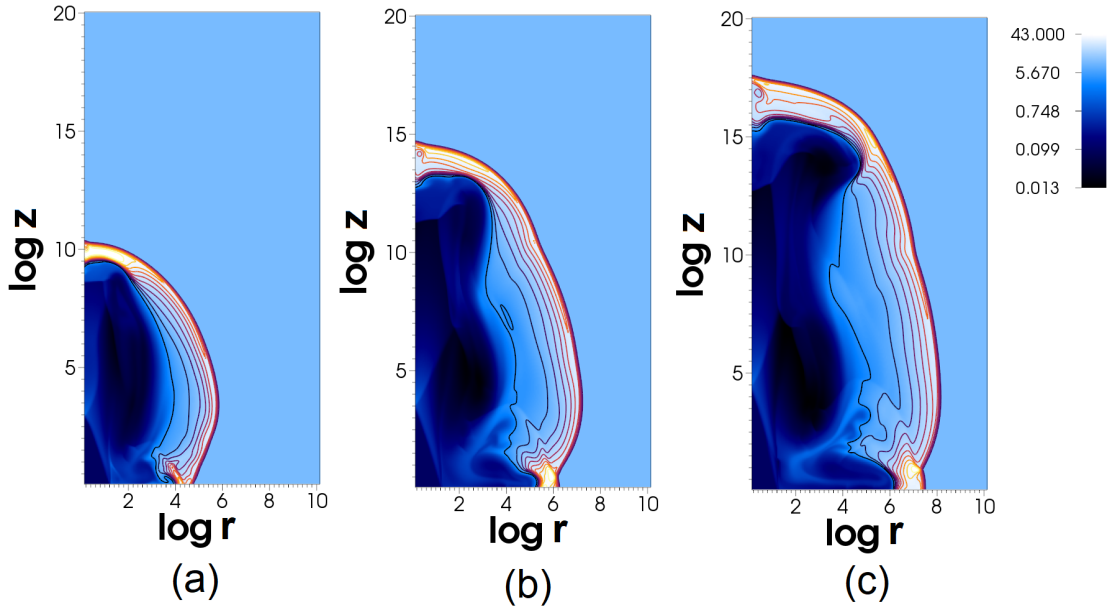

Figure 3.11: Jet evolution using density shades and density isocontours in logarithmic scale (g cm^{-3}) for (a) t=20, (b) t=30 and (c) t=40 in 2D cylindrical coordinates.

A relativistic jet was added as astrophysical test in this section so we can study specifically an outflow interacting with with its surrounding medium. The initial conditions were taken from Del Zanna and Bucciantini (2002). They proposed an axisymmetric jet in 2D cylindrical coordinates:

$$(\rho, v_z, v_r, P) = \begin{cases} (0.1, 0.99, 0, 0.01), & r \leq 1, z \leq 1, \\ (10, 0, 0, 0.01), & \text{otherwise.} \end{cases} \tag{3.3}$$

We reproduced this test with the following runtime parameters: A CFL=0.3 and a resolution of 512 cells along the z direction and 256 cells along the cylindrical radius r direction, the jet was injected within 26 cells; also, we applied the HLL solver. The material was injected with $\Gamma = 7.089$. In Figure 3.11, the jet evolution is displayed using isodensity contours at three instants of time: t=20, 30 and 40.

In Figure 3.12, the same test is shown from Del Zanna and Bucciantini (2002). In their study, the jet's launching region corresponds to 20 computational cells (that implies 400 cells in the z direction and 160 cells in the radial direction) and they employed the same solver, but instead, a Courant number of CFL=0.25 is used. The material is injected similarly with $\Gamma \sim 7.1$. The plots in Figure 3.12 follow the jet evolution for the same times as in Figure 3.11.

In both cases, the jet goes forward consistently and in Del Zanna and Bucciantini's test, its trajectory matches our results at each time step. An important feature to notice is the low turbulence found in our density maps; these vortices, due to Kelvin-Helmoltz instabilities, is expected to appear by increasing the refinement level in our simulations. Overall, a qualitatively similar jet morphology is obtained in this test.

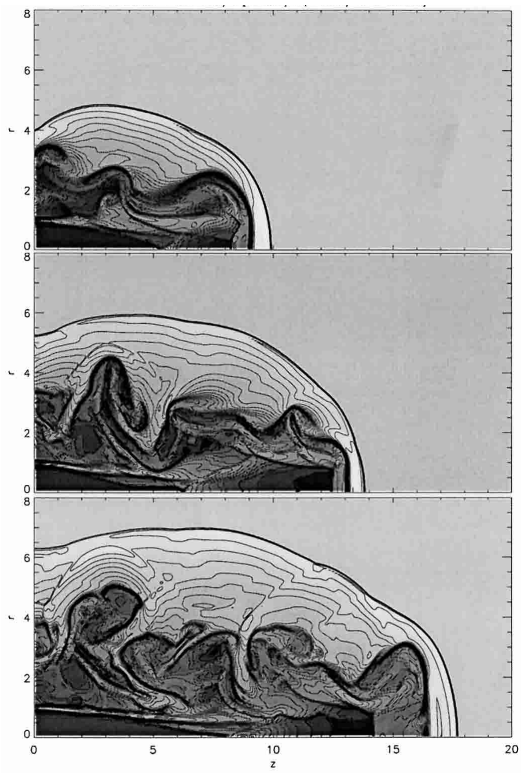

Figure 3.12: From top to bottom, jet evolution for t=20, t=30 and t=40 with logarithmic density contours and shades. (Image credit: Del Zanna and Bucciantini (2002)).

We extended this study to a three-dimensional test so we can compare the evolution of a jet with similar properties as Del Zanna and Bucciantini's test. The initial conditions applied are as follows:

$$(\rho, v_{\rm z}, v_{\rm r}, v_\theta, P) = \begin{cases} (0.1, 0.99, 0, 0, 0.01), & r \leq 1, z \leq 1, \\ (10, 0, 0, 0, 0.01), & \text{otherwise}, \end{cases} \tag{3.4}$$

in such a way that the jet is still injected inside a radius $r = 1$ with $v_z = 0.99$; we established a Cartesian coordinate system for this test, so $r = \sqrt{x^2 + y^2}$. The domain consists of from 250 cells in the z direction and 260 cells in each of the other two directions. The Courant number is set to CFL=0.3 and the same solver was used as in the previous tests. In Figure 3.13, the 3D density contours are shown for the same times as in the 2D tests and also, in Figure 3.14, a 2D cut of the 3D results is displayed.

An important difference between the 3D test and 2D test is the path travelled by the jet for the same time span. In the 3D simulation, the outflow can travel a longer distance because there are not an imposition of symmetry like in 2D simulations (López-Cámara et al., 2013). However, the evolution of the inflow's density and Lorentz factor were similar for both tests (2D and 3D).

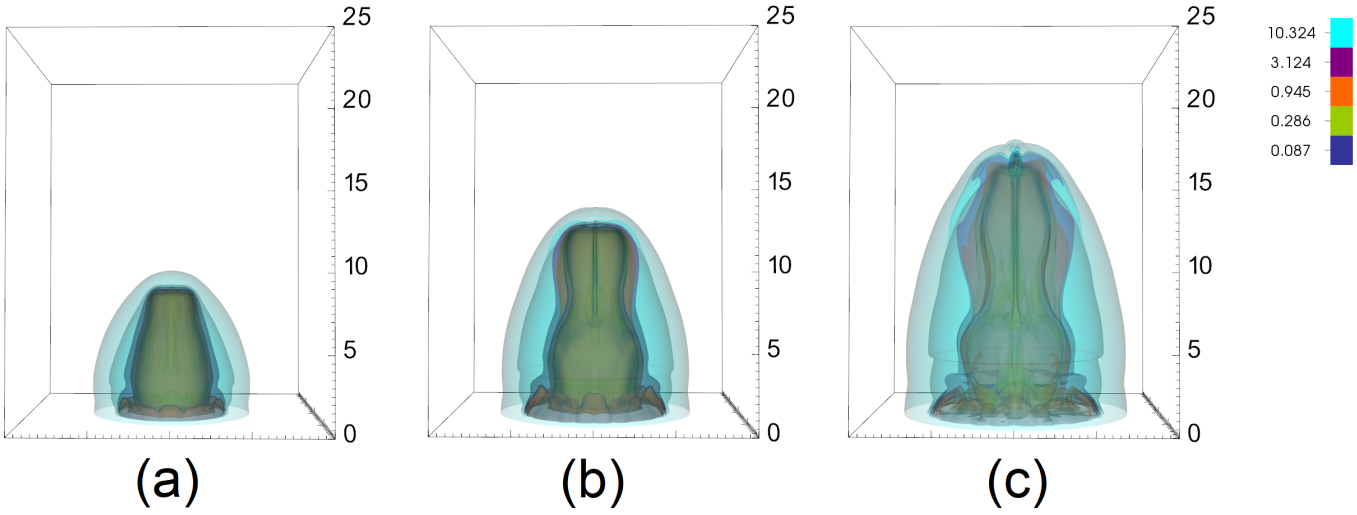

Figure 3.13: Density contours in a 3D extended domain in logarithmic scale (g cm^{-3}) for (a) t=20, (b) t=30 and (c) t=40 in Cartesian coordinates.

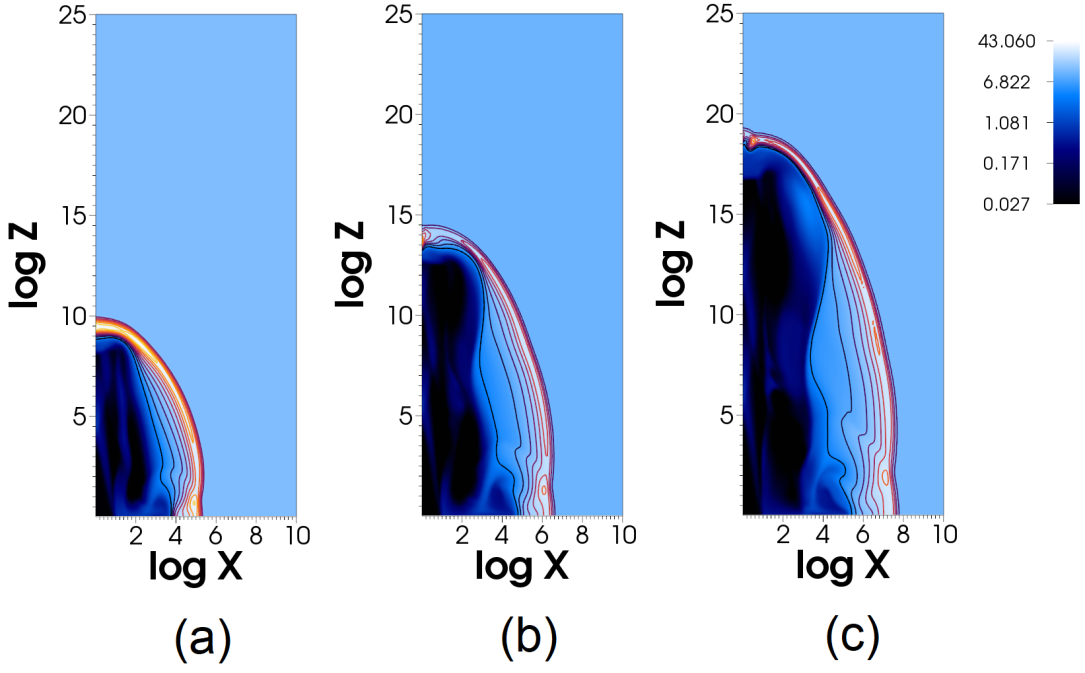

Figure 3.14: Density shades and contours in logarithmic scale (g cm^{-3}) for (a) t=20, (b) t=30 and (c) t=40 in the xz mid plane.

4 — Numerical Setup

In this chapter we will discuss the numerical setup used to perform our simulations, which involves the initial and boundary conditions.

We will introduce the progenitor density and pressure profile from the 16TI model of Woosley & Heger (2006) and the incorporation of different angular momentum distributions.

4.1 Initial Condition

In order to accomplish our main goal, which is to study the evolution of a jet emitted from its 3D rotating progenitor core, we have to build an initial condition considering the progenitor's density and pressure initial profiles and, in this case, its initial rotation profile.

These conditions can be taken from Woosley and Heger (2006) where they show several pre-SN progenitors with different features such as metallicity, initial mass and mass loss rate, rotational velocity and magnetic field. It is important to say that all these profiles are built in 1D and this entails several approximations to compute the model properties.

In Table 4.1 are some model examples taken from the mentioned paper. In the first column the model's name tag is listed, the second column shows the metallicity in solar units, in the third column it is pointed out if the model incorporates a magnetic field, in the fourth column we can find the final mass for each model, and in the following column the initial rotational velocity of the progenitor is shown, in the last column the Kerr parameter is also listed (which is related to the spin of a Kerr black hole).

Model	Metallicity [$Z_\odot$]	B-field	M_{final} [$M_\odot$]	v_{rot} [kms^{-1}]	a_{BH}
16TI	1/100	yes	13.95	390	0.44
12TH	1/100	yes	9.23	380	0.17
12OE	1/10	no	11.83	245	2.2
16OF	1/10	no	8.97	325	1.1
12SI	1	yes	6.95	405	0.03

Table 4.1: Progenitor examples from Woosley & Heger (2006).

We can identify the properties of the models directly from their name tags: The first number indicates the ZAMS, the first letter identifies the metallicity (which is T for 1% $Z_\odot$, O for 10% $Z_\odot$ and S for $Z_\odot$) and the last letter includes the features for mass loss rate and magnetic torques.

These stellar evolution models were built using the implicit hydrodynamics code KEPLER, which solves the conservation equations shown in Section 2.1 by assuming spherical symmetry (Weaver et al. 1978). Nevertheless, this code does not consider rotation nor magnetic fields because of this assumption. In order to account for the effects of rotation, Woosley and Heger use an approximate scheme for angular momentum transport which treats it as a diffusion process (Heger et al. 2000).

Also, they included magnetic torques using a dynamo model (Spruit 2002). The basis of the formalism are the assumption of magnetic fields within stratified layers of the star, which have differential rotation, and a magnetic instability in the toroidal field (Tayler instability). This process transports effectively angular momentum.

For our simulations, we use the 16TI model as initial condition because it is a model that has been widely studied and it is known to be a likely progenitor for lGRBs. In the following section we describe some other features and profiles from the 16TI model.

4.1.1 Woosley & Heger 16TI Progenitor

As mentioned before, the initial condition for our simulations was set up using the pre-SN progenitor 16TI model proposed by Woosley & Heger (2006). This model assumes a Wolf-Rayet star that initially has 16 $M_\odot$ and a 1% $Z_\odot$ metallicity. The final mass of this pre-SN star is found to be 13.95 $M_\odot$ with a radius of 4.1×10^{10} $R_\odot$. This model also includes the final state of elements within the star, which are distributed as shells, going from Silicon, Sulfur, Calcium and Argon at inner radii ($10^8 - 10^9$ cm) to Oxygen and Magnesium at external radii ($10^9 - 4\times10^{10}$ cm); this progenitor has lost completely its Hydrogen envelope through winds during its evolution. However, the detailed chemical composition of the progenitor is neglected for the purposes of this study; we assume a composition of a progenitor made of a single monatomic gas with an adiabatic index value of γ =5/3.

The star assumed for this model rotated at approximately 400 km s^{-1} at the equator in the main sequence stage and had low metallicity (1% $Z_\odot$) and then became a Wolf-Rayet (WR). Also, the iron core has a Kerr parameter $a_{\rm BH}$ of 0.44 at 3 $M_\odot$, which means it would form a disk to carry extra angular momentum (if a progenitor reaches $0.3 \lesssim a$ is also a good candidate; the angular momentum increases outwards) (Woosley & Heger, 2006). This feature makes this model a good candidate to be a collapsar .

This model has 1D density and pressure profiles, which means that we have to map these profiles into a 3D region for our simulations. The 1D profiles can be seen in Figure 4.1. Also, we consider this pre-SN progenitor to be immersed in an ISM with fixed constant density and pressure ($\rho_{\rm ism} = 10^{-13}$ g cm^{-3}, $P_{\rm ism} = 10^{-16}$ dyn).

After an interpolation, we obtain 3D profiles as shown in Figure 4.2. The simulations were performed for the top half of the pre-SN progenitor. Thus, the computational domain was set as $(12\times10^{10})\times(12\times10^{10})\times(6\times10^{10})$ cm assuming reflection symmetry across the xy-plane.

The 16TI model also provides a distribution of angular momentum as a function of the spherical radius $J(R)$. The third panel in Figure 4.1 displays the angular momentum distribution

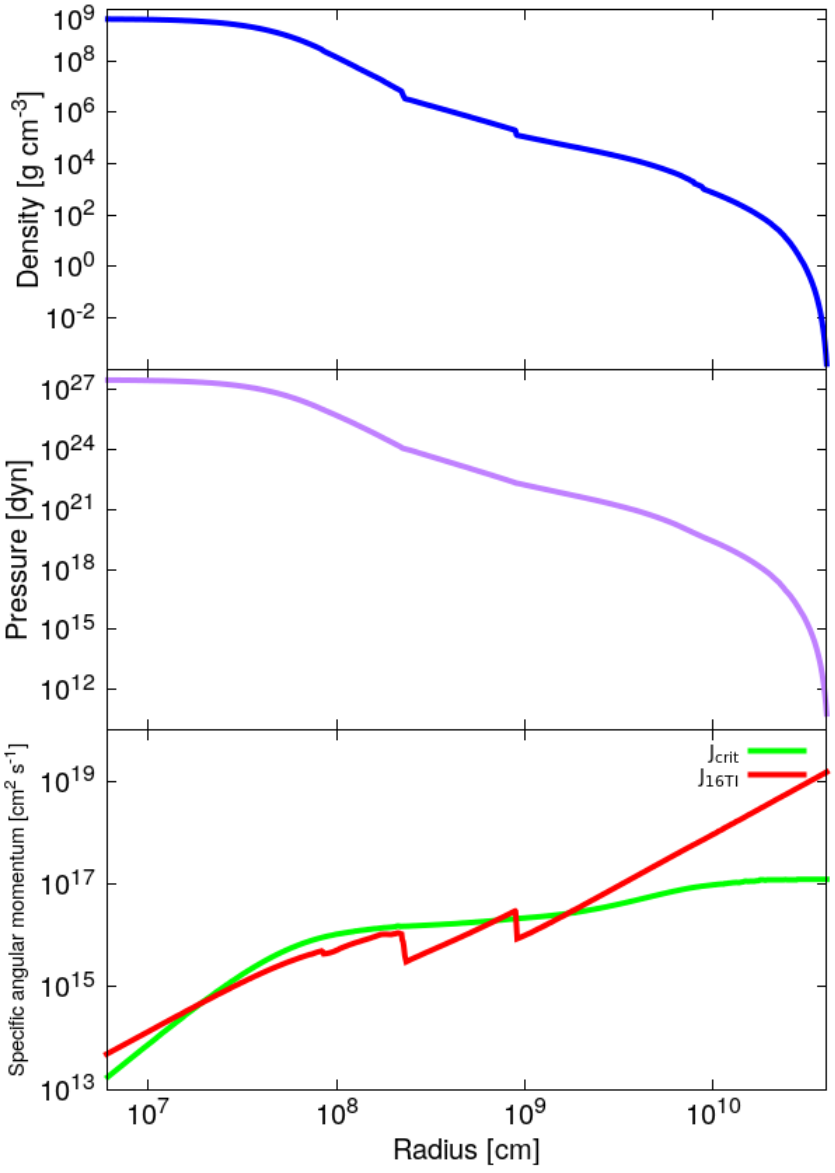

Figure 4.1: 16TI density (top panel), pressure (middle panel) and specific angular momentum (bottom panel) profile.

versus the radius; here, we can notice the different layers where the curve has jumps. Also, the green line establishes a limit of the specific angular momentum called the critical specific angular momentum, $J_{\rm crit}$; this parameter is defined mathematically as $J_{\rm crit} = 2r_{\rm g}c$, where $r_{\rm g}$ is the gravitational radius. This parameter determines the threshold values (depending on the black hole mass) at which the formation of a centrifugally support disk is feasible (López-Cámara et al., 2010).

Following the collapsar model, is it assumed that a black hole is formed from the star iron core due to gravitational collapse. This nucleus contains 3 $M_\odot$ of iron, so we will get a 3 $M_\odot$ black hole with a massive surrounding envelope. The rotation can stop the material from falling into the black hole if its angular momentum is higher than this $J_{\rm crit}$ limit so an accretion disk can be formed. For the 16TI model, above 10^8 cm, $J_{\rm crit}$ has values up to $\sim 10^{17}$ cm^2 s^{-1}.

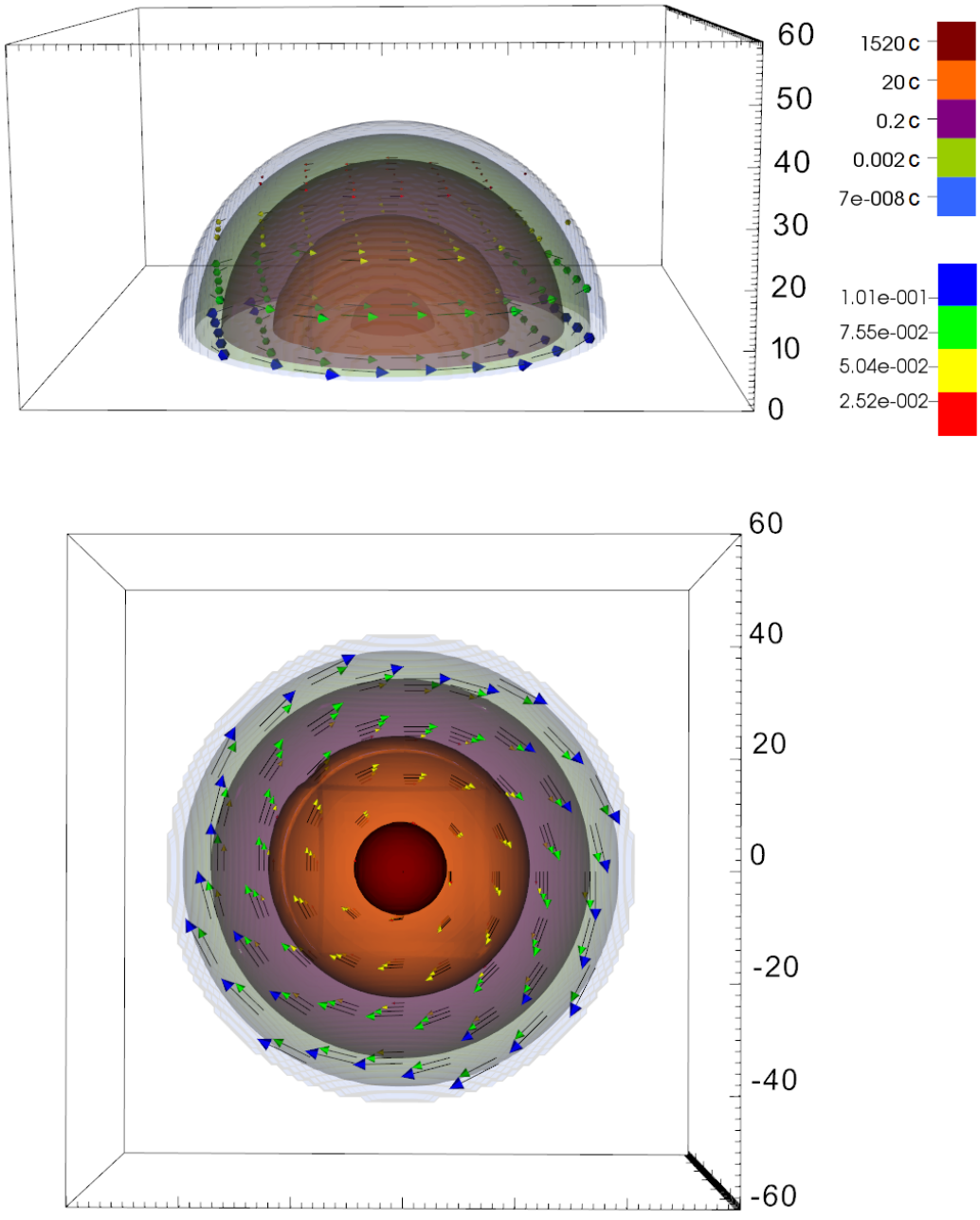

Figure 4.2: Density contours (g cm^{-3}) of 16TI model mapped into a 3D domain. The scale unit for each spatial axis is 10^9 cm. An example of the velocity field due to angular momentum implementation is also shown.

Modelling the discontinuities represents a challenge for the code and, in some cases, impossible to solve them because the jump in these distribution of angular momentum is too steep. For this reason, we consider a modification to the original angular momentum distribution by using a smooth functional dependence with a qualitatively similar behaviour. In addition, we also consider a configuration that involves the vertical coordinate (along the z-axis), which means a $J(z)$ distribution. In the next section we will explain in detail the angular momentum

decomposition proposed for the progenitor motion.

4.1.2 Angular Momentum Decomposition

Let us recall the set of equations (2.36), where we concluded that by assuming a specific angular momentum distribution, the velocity components in Cartesian coordinates can be established. From the Woosley & Heger (2006), one can approximate a $J(R)$ distribution that increases with the radius but it was important to take into account the dependence on the polar angle θ. Thus, we propose a decomposition into the radial and polar components of the angular momentum distribution:

$$J = J(R, \theta) = J_1(R)J_2(\theta), \tag{4.1}$$

and we can express a similar decomposition using cylindrical coordinates:

$$J = J(r, z) = J_1(r)J_2(z). \tag{4.2}$$

A study concerning this kind of decomposition was performed by López-Cámara et al. (2010) where they fixed a form for $J_2(\theta)$ ($\Theta(\theta) = \sin^2\theta$) and varied the functional form of $J_1(R)$ to explore the accretion rates in the disk. In our case, we propose different distributions for both, $J_2(z), J_2(\theta)$ and the angular velocity. The functional form of these distributions are shown in Table 4.2.

Ω=constant	$J(z), J(\theta)$
Ω_0 =1.25×10^{-3} rad s^{-1}	$1 - z/(5 \times 10^{10}\text{cm})$
Ω_1 =3.12×10^{-3} rad s^{-1}	$1 - \cos\theta$
Ω_2 =5.3×10^{-3} rad s^{-1}	$1 - \cos^2\theta$

Table 4.2: Specific angular momentum distributions for radius dependency ($J(r)$) and polar angle dependency ($J(z), J(\theta)$).

In Figure 4.3, we show the 16TI radial angular momentum distribution and the approximations proposed with different angular velocity values in 1D profiles, which concerns the first set of models studied in Section 5.1. It is important to remark that our approximations reach the threshold value of J_{crit} for radius higher than 10^9, which is a valid assumption since we are launching the jet in a region nearby this radius 2×10^9, so the rotation effects involve the envelope where the jet is propagating. The angular velocity for the J_{crit} value at 2×10^9 is $\sim 2.5 \times 10^{-3}$rad s^{-1}.

Next, the proposed 1D vertical angular momentum distribution profiles are plotted, which are related with the progenitor models discussed in Section 5.2. In Figure 4.4, the $J(\theta)$ distributions are shown and, in Figure 4.5, the $J(z)$ distribution is also plotted. All of these distributions where considered following the hypothesis that the rotation decreases towards the poles of the progenitor. Here is important to say that the vertical angle, θ is measured from zero (the z direction) to $\pi/2$ (the equatorial plane), so the specific angular momentum has a maximum at $\pi/2$.

For the angular momentum with dependency on the angle, we take the cosine function with two power laws (in first and second grade) because the behaviour of the two functions is different; the second-grade function has a smoother increase than the first-grade one. This difference might be noticed in the morphology of the rotating progenitors.

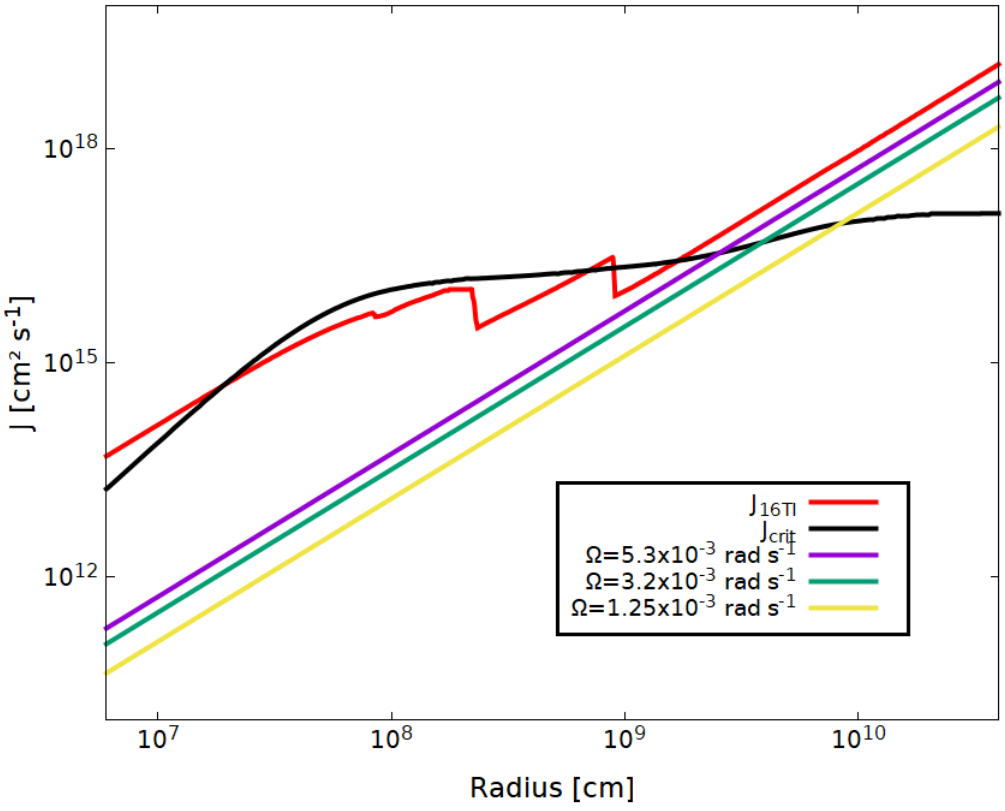

Figure 4.3: 1D specific angular momentum profiles for the 16TI model (red), the model with Ω_2 =5.3×10^{-3} rad s^{-1} (vilet), the model with Ω_2 =3.2×10^{-3} rad s^{-1} (green), the model with Ω_2 =1.25×10^{-3} rad s^{-1} (yellow) and the threshold established by the $J_{\rm crit}$ value (black).

Finally, it is important to underline that gravitational force due to the central engine were not considered for the initial condition. To support this assumption, we computed the free-fall time at the iron core radius (innermost boundary for the jet injection) and we compared it against the break out time of jet (time at which the jet reaches the surface of the envelope). If $t_{\rm ff} < t_{\rm bo}$, the jet will propagate before the gravitational effects, due to the central engine, affect its evolution.

Taking the iron core radius $R_{\rm Fe} \sim 10^8$ cm and the average density at this shell $\bar{\rho}$ ∼6.65×10^7 g cm^{-3} (Woosley & Heger, 2006), we have

$$t_{\rm ff} \sim (G\bar{\rho})^{-1/2} \sim 0.5{\rm s} \tag{4.3}$$

and by considering the typical break out time for lGRBs, $t_{\rm bo}$ ∼10 s, we obtain

$$t_{\rm ff} < t_{\rm bo} \tag{4.4}$$

so the gravitational effects can thus be ignored.

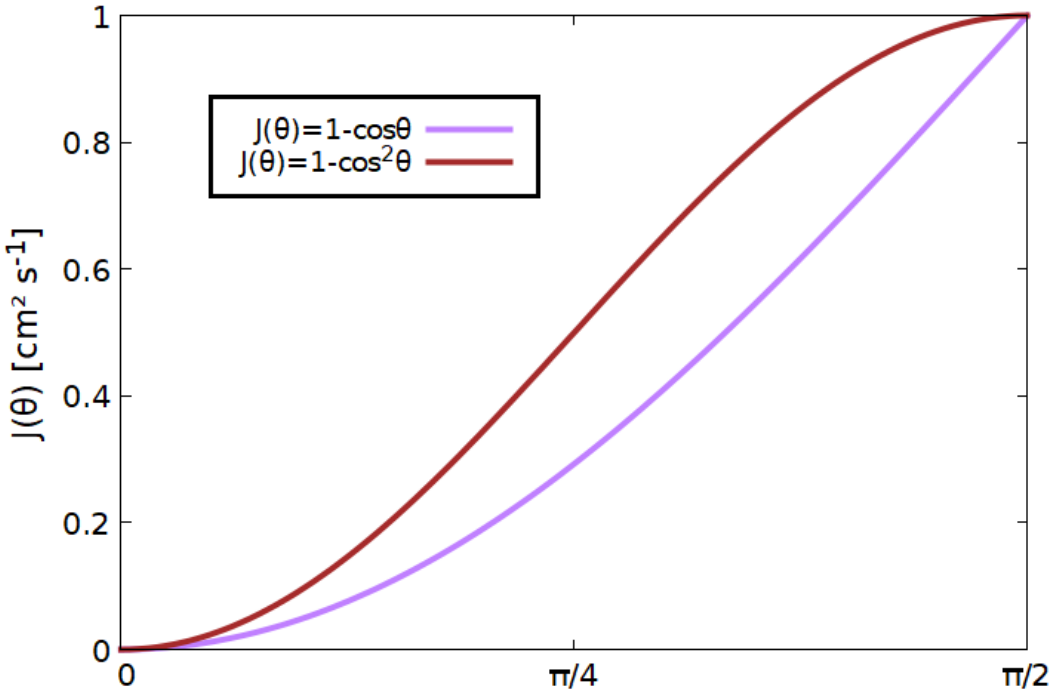

Figure 4.4: 1D vertical angular momentum distributions with angular dependency. The two models are $J(\theta) = 1 - \cos\theta$ (purple) and $J(\theta) = 1 - \cos^2\theta$ (cherry).

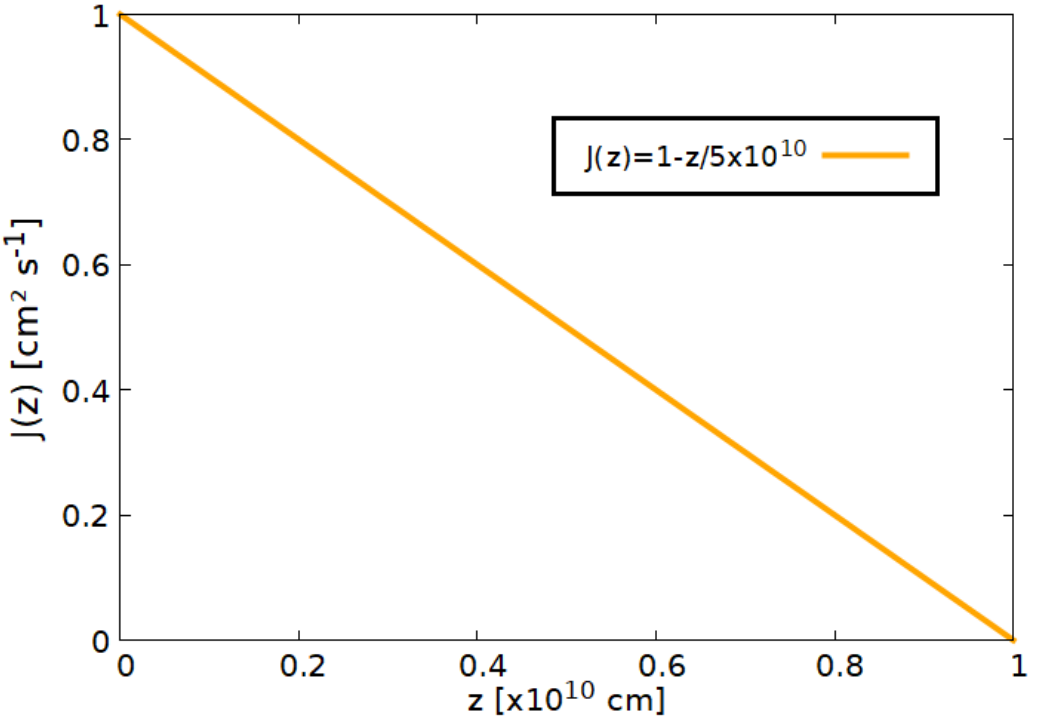

Figure 4.5: 1D vertical angular momentum distributions with dependency on the z coordinate.

4.2 Boundary Conditions

The injection of the relativistic jet was imposed as a boundary condition by considering a constant inflow from a small, central region of the progenitor. The jet was launched at the plane $z_{\rm min} = 2 \times 10^9$ cm, and a half-opening angle of $\theta_0 = 5°$ (Geng et al., 2016; López-Cámara et al., 2016). In the following lines we describe the parameters applied for the jet injection, which are the Lorentz factor, density and luminosity.

Here we explain the 3D implementation of the jet's opening angle and the criterion used to assign the injection cells. A scheme of the 3D boundary conditions implementation is shown in Figure 4.6 for the initial injection time ($t = 0$s) and a posterior time.

First of all, we have to apply the velocity conditions of the jet, for this, we impose

$$\begin{cases} v_r = v_{\rm r,0} \left(\dfrac{r}{R_{\rm i}} \right), & r < R_i \\ v_z = v_{\rm z,0} \end{cases} , \tag{4.5}$$

where $v_{\rm r}$ is the radial velocity, $v_{\rm z}$ the vertical velocity, r the cylindrical radius, R_i is the radius within the jet is injected and $v_{\rm r,0}, v_{\rm z,0}$ constants in each case. The distribution applied in the radial component encodes the effect of the opening angle of the opening angle. Also, we consider the relations between the velocities

$$\begin{cases} \tan\theta = \dfrac{v_{r,0}}{v_{z,0}} \\ v_{r,0} = v_{z,0} \tan\theta \end{cases} . \tag{4.6}$$

Since we were working in Cartesian coordinates, we have to consider the following transformation:

$$\begin{cases} v_{\rm x} = v_{\rm r} \cos\varphi - v_\varphi \sin\varphi \\ v_{\rm y} = v_{\rm r} \sin\varphi + v_\varphi \cos\varphi \end{cases} , \tag{4.7}$$

where $v_{\rm x}$ and $v_{\rm y}$ are the velocities along the x and y-axis, respectively. Setting v_φ=0 and substituting the values in Equation (4.5), we obtain

$$\begin{cases} v_{\rm x} = v_{\rm r,0} \left(\dfrac{r}{R_{\rm i}} \right) \cos\varphi \\ v_{\rm y} = v_{\rm r,0} \left(\dfrac{r}{R_{\rm i}} \right) \sin\varphi \\ v_{\rm z} = v_{\rm z,0} \end{cases} , \tag{4.8}$$

Finally, the cells assigned for the jet launching are selected with the following criterion

$$\tan\theta_0 = \frac{r}{z_{\rm min}} \tag{4.9}$$

where $z_{\rm min}$ is the lower boundary in z-direction. With equation (4.13) we impose that the cells enclosed in the cylindrical radius $r \leq z_{\rm min} \tan\theta_0$ are launching the jet.

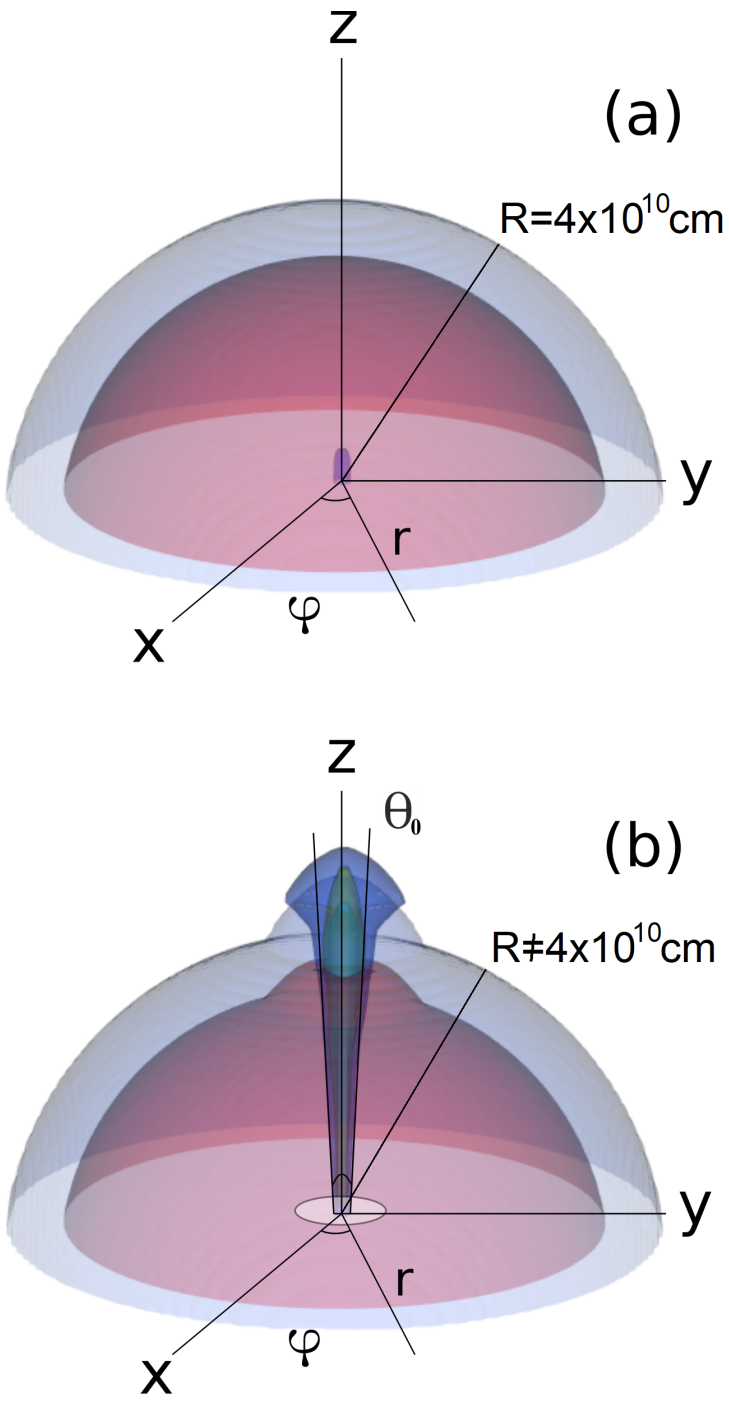

Figure 4.6: 3D scheme of boundary conditions at (a) $t = 0$ s (jet's injection) and (b) $t \gg 0$ s (high evolved stage). In (b), we state that the radius of the progenitor differs from the initial radius, $R = 4 \times 10^{10}$cm, due to the angular momentum.

The simulation domain is the top half of the pre-SN progenitor, whose boundaries are $x_{\rm max} = -x_{\rm min} = 2 \times 10^{10}$ cm, $y_{\rm max} = -y_{\rm min} = 2 \times 10^{10}$ cm, $z_{\rm min} = 2 \times 10^{9}$ cm and $z_{\rm max} = 6 \times 10^{10}$

cm. Reflective conditions were set at the equatorial plane $z = z_{\min}$ and all the other boundaries are set with outflow conditions.

On the other hand, to avoid the effects of symmetry along the jet's propagation path (since we make an interpolation from a 1D profile, we want to build a 3D model that has no perfect symmetry), we impose small perturbations in the jet's velocity (v_x and v_y components). This kind of perturbation varies the ejection area of the jet as a function of a given amplitude, η, and a frequency, ω.

Recalling the expressions for velocity components obtained in equation (4.8), we introduce the perturbation as

$$\begin{cases} u = v_x = v_{r,0}\left(\dfrac{r}{R_i}\right)\cos\varphi(1+\eta\cos\omega t) \\ v = v_y = v_{r,0}\left(\dfrac{r}{R_i}\right)\sin\varphi(1+\eta\cos\omega t) \\ w = v_z = v_{z,0} \end{cases} . \tag{4.10}$$

For our simulations, we chose a perturbation with the values $\eta = 2$ and $T = 0.01$ s, which imply that $\omega = 628.31$ rad s^{-1}. These values are applied to all out models (rotating and non-rotating cases).

It is important for this implementation to choose a sufficiently small perturbation, this means, that it should not affect the upper limit for the jet's velocity ($v/c < 1$, where c is the speed of light). By taking $\eta = 2$, $v_{r,0}$=0.083c and $v_{z,0}$=0.95c (velocity conditions for θ_0=5°), from Equation (4.10) we can obtain the maximum velocity reached by the jet,

$$v = \sqrt{v_r^2 + v_z^2} = 0.9820c, \tag{4.11}$$

where $v_r^2 = v_x^2 + v_y^2$; this velocity leads to a Lorentz factor of $\Gamma_0 = 5.3$, if we set the jet's density as $\rho_j = 0.3117$ g cm^3, the luminosity can be computed by applying the Rankine-Hugoniot conditions from Section 2. Using equation (2.26), we get a luminosity of $L_{\rm jet} \sim 9 \times 10^{49}$ erg s^{-1}, which is a typical jet luminosity.

These parameters are specified on the input file *Grid_bcApplyToRegionSpecialized.f90*, mentioned in Chapter 2, and an example of which can be consulted in Appendix F.

4.2.1 Refinement Along the Jet Direction

We use an adaptive grid with rectangular-shaped pixels for the jet injection. The purpose of applying this mesh is to impose the highest refinement level on the jet axis as well as on the boundary between the progenitor and the ISM.

The lower limits on the refinement are $\Delta x = \Delta y$ =5.2×10^8 cm and Δz=9.5×10^8 cm (which is a minimum refinement level of 1 in the FLASH parameters) and the highest limits on the refinement are $\Delta x = \Delta y$ =1.52×10^8 cm and Δz=2.4×10^8 cm (which is a maximum refinement level of 6). The jet was injected within the highest refinement region with 28 cells. In Figure

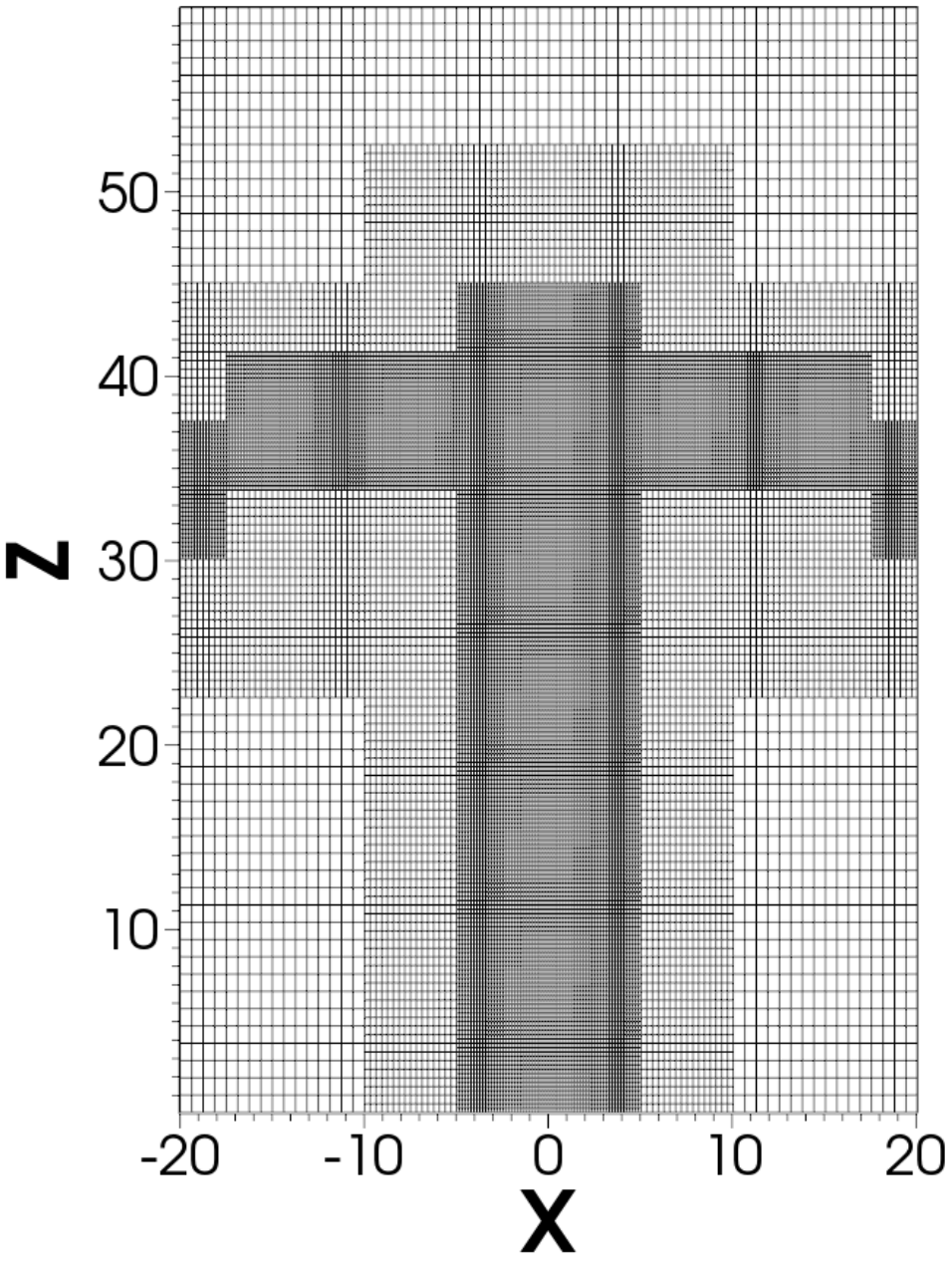

Figure 4.7: AMR configuration fixed for the jet's injection in a 2D map. The scale unit for each spatial axis is 10^9 cm.

4.7, we show the mesh distribution across the domain projected onto a 2D map in the xz-plane. This configuration is kept fixed throughout the simulation.

As we mentioned in Chapter 3, to set this mesh we need an additional file to specify the limits on each refinement level. An example of this file can be consulted in Appendix G.

5 — Results and Discussion

In this chapter, we will show the results of our simulations. First, a comparison between the progenitors with different angular velocity (Ω) values using $J(z) = \text{const}$,, is illustrated. Also, a comparison now varying the vertical dependence of the angular momentum distribution $J(z)$ using a selected value of Ω from the previous study is presented.

Moreover, the ratio between the equatorial and polar radii of the progenitor is analysed to quantify the changes in the morphology due to the angular momentum.

Finally, two models are chosen as the jet's progenitors. The breakout time ($t_{\rm bo}$), breakout Lorentz factor ($\Gamma_{\rm bo}$), cocoon radius $r_{\rm bo}$ and final angular momentum distribution are obtained to study the effects of the jet breaking through the progenitor.

5.1 Progenitors with Different Ω Values

In this first subsection of the results we will compare the changes in the morphology of the progenitor, which was affected by the angular momentum imposed with $J(z)$ =const and different normalization values for $J(R)$ (which is related to distinct Ω values).

For the first setups, we take three different values of angular velocity Ω:1.25×10^{-3} rad s^{-1}, 3.12×10^{-3} rad s^{-1} and 5.3×10^{-3} rad s^{-1}, so we have three initial models. These quantities correspond to the following angular momentum values for $R = 4 \times 10^{10}$cm: J= 2×10^{18} cm^2 s^{-1}, J$\sim$ 5×10^{18} cm^2 s^{-1} and J= 8.48×10^{18} cm^2 s^{-1}, respectively. All of these models are above the $J_{\rm crit}$ value for the 16TI model from the study by Woosley and Heger (2006) described in Section 4.1.1 (Figure 4.1), which is about 10^{17} cm s^{-1}.

For all of the models, we maintain the azimuthal angular momentum distribution $J(z)$ as a constant to highlight the variations on angular velocity. In Table 5.1, the label for each model is shown: the first half of the label refers to the distribution applied for J(z) and, the second half denotes the angular velocity, which is L for the lowest value (1.25×10^{-3} rad s^{-1}), M for the medium (3.12×10^{-3} rad s^{-1}) and H for the highest value (5.3×10^{-3} rad s^{-1}). We will refer to the models by these labels in the following sections.

In Figure 5.1, we show the velocity field inside the progenitor considering $J(z) = 1$ cm^2 s^{-1} with an Ω value at the initial time t=0 s. The velocity reaches its minimum value (zero) at the polar axis, according to the derived equations (2.37) and (4.2); the maximum velocity, 0.216 c, is reached at the external radius (4×10^{10} cm) in the equatorial plane, is to be expected since the angular momentum increases with cylindrical radius. Also, in Figure 5.2, the initial condition

Model	$J(z)$ (cm^2 s^{-1})	Ω (rad s^{-1})
J(z)cnt-ΩL	1	1.25×10^{-3}
J(z)cnt-ΩM	1	3.12×10^{-3}
J(z)cnt-ΩH	1	5.3×10^{-3}

Table 5.1: Specific angular momentum models by label.

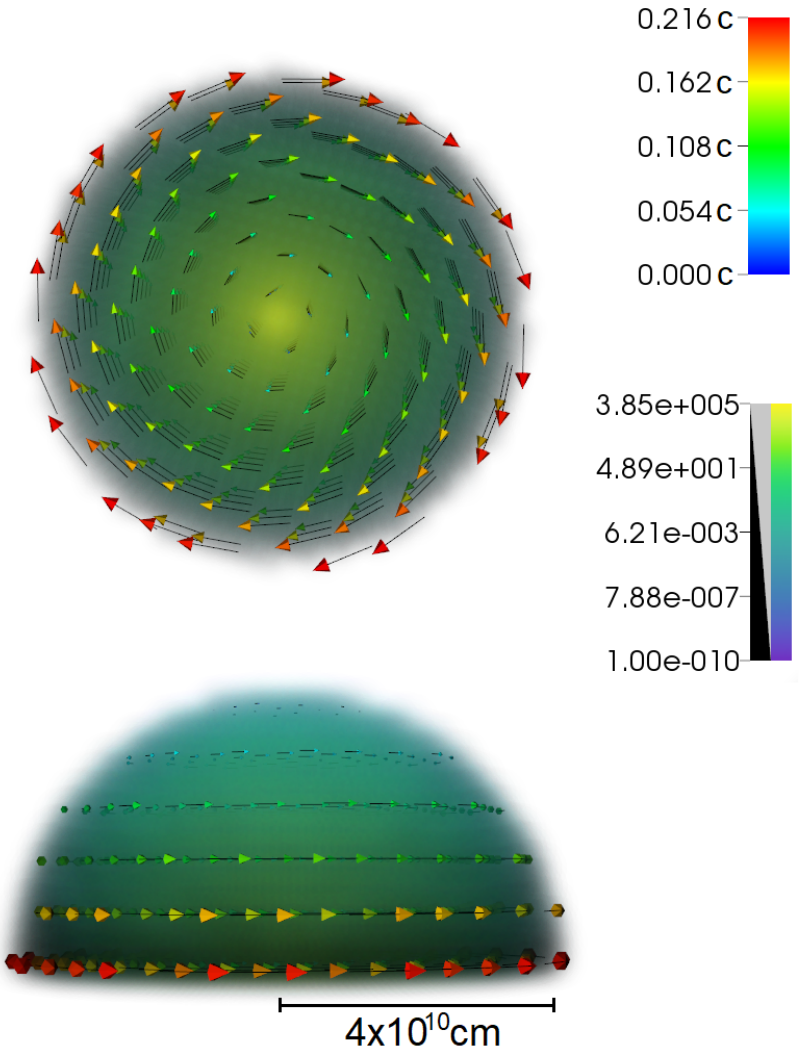

Figure 5.1: Velocity field inside the progenitor J(z)cnt-ΩH. The arrow length and its color show the magnitude of the vector. We can clearly observe that the velocity increases towards external radii. The top color palette corresponds to the velocity magnitude while the bottom one displays the density values (g cm^{-3}).

for the density profile is displayed with density contours onto the xz-plane (y=0). The initial density profile, taken from Woosley and Heger's (2006), was considered to be the same for each model.

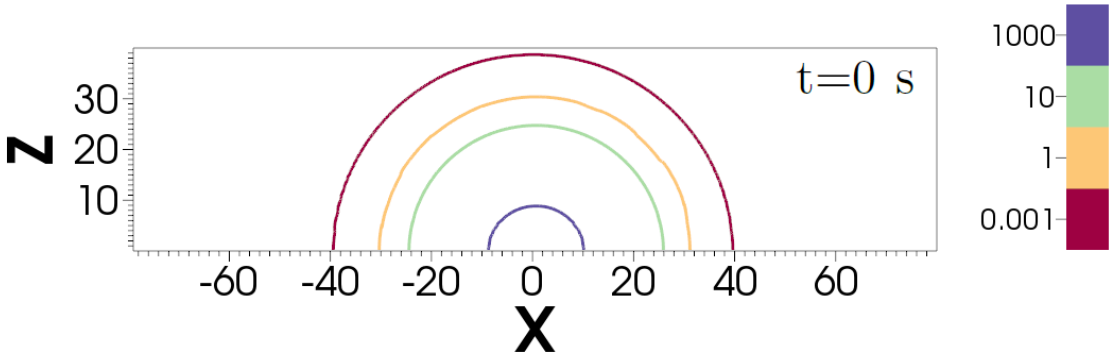

Figure 5.2: Density contours (g cm^{-3}) for t=0 s in the xz-mid plane (y=0). The scale unit for each spatial axis is 10^9 cm.

We let the simulation run for 10 s in each model to enable the progenitor to reach the extreme ratio of the polar to equatorial radii (Maeder & Meynet, 2012). In Figure 5.3 we show the evolution for the J(z)cnt-ΩM model at t=0 s, t=5 s and t=10 s with the same density contours as in Figure 5.2. With this representation, it is noticeable that the equatorial radius $R_{\rm eq}$ suffers an elongation due to rotation causing all the progenitor's shells to expand along this direction.

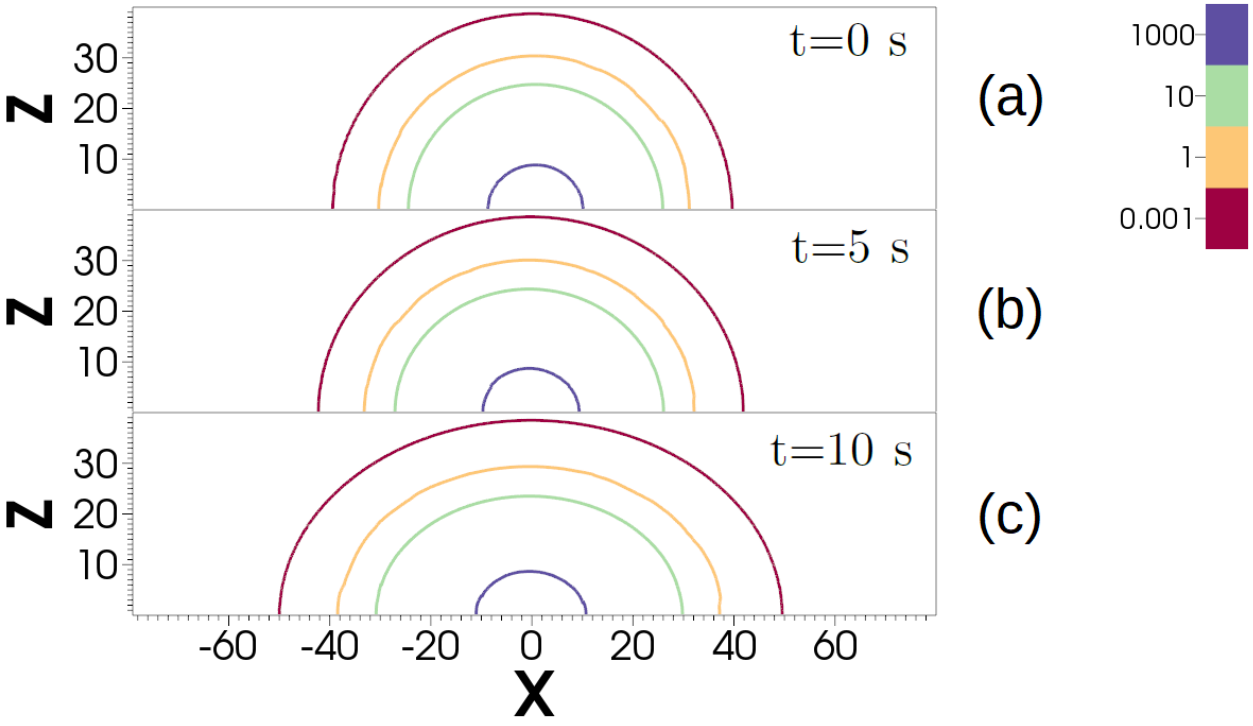

Figure 5.3: Density contours (g cm^{-3}) in the xz-plane for (a) t=0 s, (b) t=5 s and (c) t=10 s of the model J(z)cnt-ΩM. The scale unit for each spatial axis is 10^9 cm.

Thus, when considering only the radial angular momentum settings, the progenitor suffers an oblate deformation and it also affects, not only the external radii of the progenitor, but the

inner regions where the jet begins its propagation.

Next, in Figure 5.4 we show the final 2D density distribution for the three models at t=10 s. This plot reveals the effects of the different values of angular velocity, the higher the value in the velocity, the greater the widening of $R_{\rm eq}$.

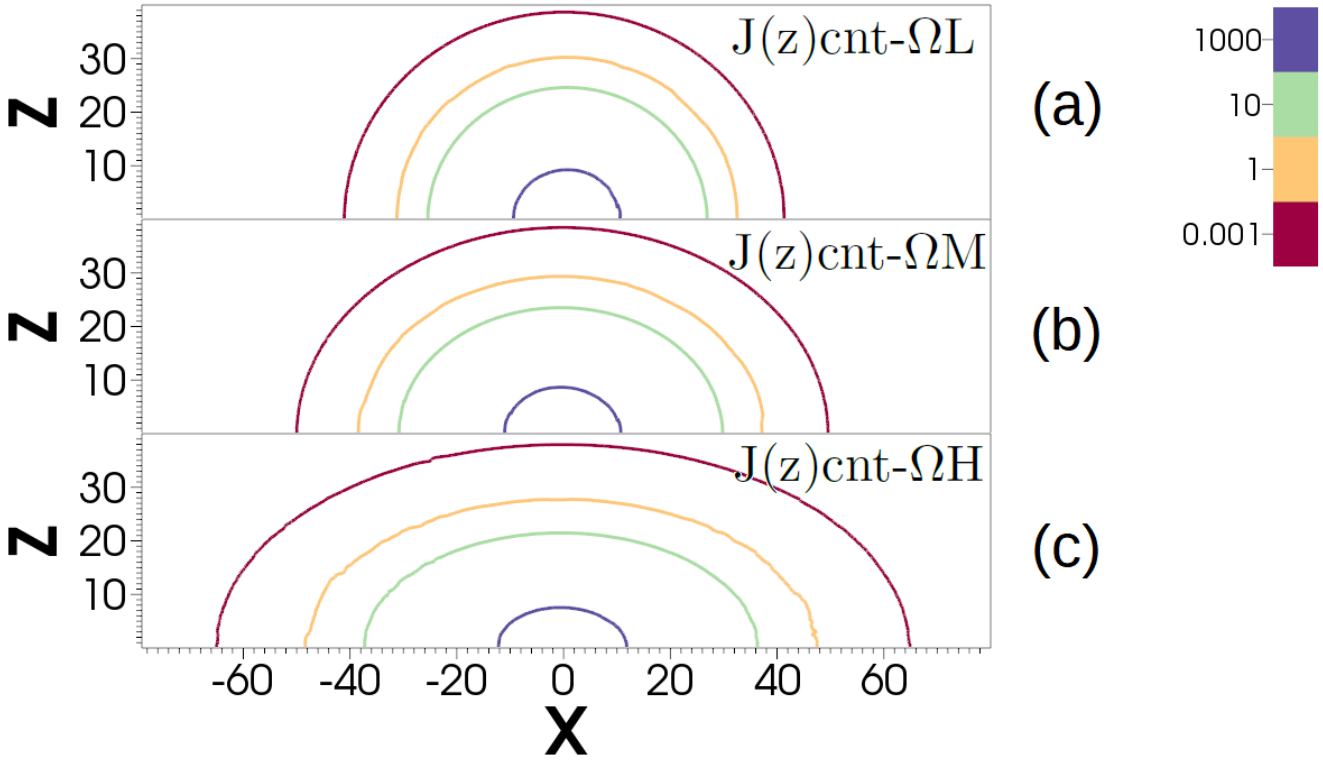

Figure 5.4: Density contours (g cm^{-3}) projected onto xz-plane for the models (a) J(z)cnt-ΩL, (b) J(z)cnt-ΩM, (c) J(z)cnt-ΩH at t=10 s. The scale unit for each spatial axis is 10^9 cm.

To quantify the deformation in the morphology of the three models, we take the ratio between the polar radius $R_{\rm pol}$ and equatorial radius $R_{\rm eq}$ at five times: t=0 s, t=2 s, t=5 s, t=8 s and t=10 s using the external density contour of 1000 g cm^3. The graph of the $R_{\rm pol}/R_{\rm eq}$ ratio is shown in Figure 5.5. Here, we can see that the ratio values become different from early times comparing the three models. For the low angular velocity model, J(z)cnt-ΩL, the ratio decreased to 0.95 at the maximum time considered, the middle J(z)cnt-ΩM model reached a value just below 0.8 and the most notable decrement achieved corresponds to the J(z)cnt-ΩH model with a value of 0.6. Note that, even though the angular velocity values have the same order of magnitude, their variation affects this ratio considerably. The percentage increase in $R_{\rm eq}$ for each model at t=10 s is: 4.75% for J(z)cnt-ΩL model, 26.52% for J(z)cnt-ΩM model and 64.87% for J(z)cnt-ΩH model.

With these results, we have to remark that the behaviour of this ratio is mainly due to the changes in $R_{\rm eq}$; the contraction of $R_{\rm pol}$ in these models is less than 2%. We can observe the evolution of $R_{\rm pol}$ percentage change in Figure 5.6. Here, we can see that the high and medium angular velocity models experimented a similar contraction of $R_{\rm pol}$ up until t=2 s. After that, J(z)cnt-ΩH model increases faster than the J(z)cnt-ΩM model. For the lower angular velocity model, there are not large variations of $R_{\rm pol}$. This is an important result to consider since the

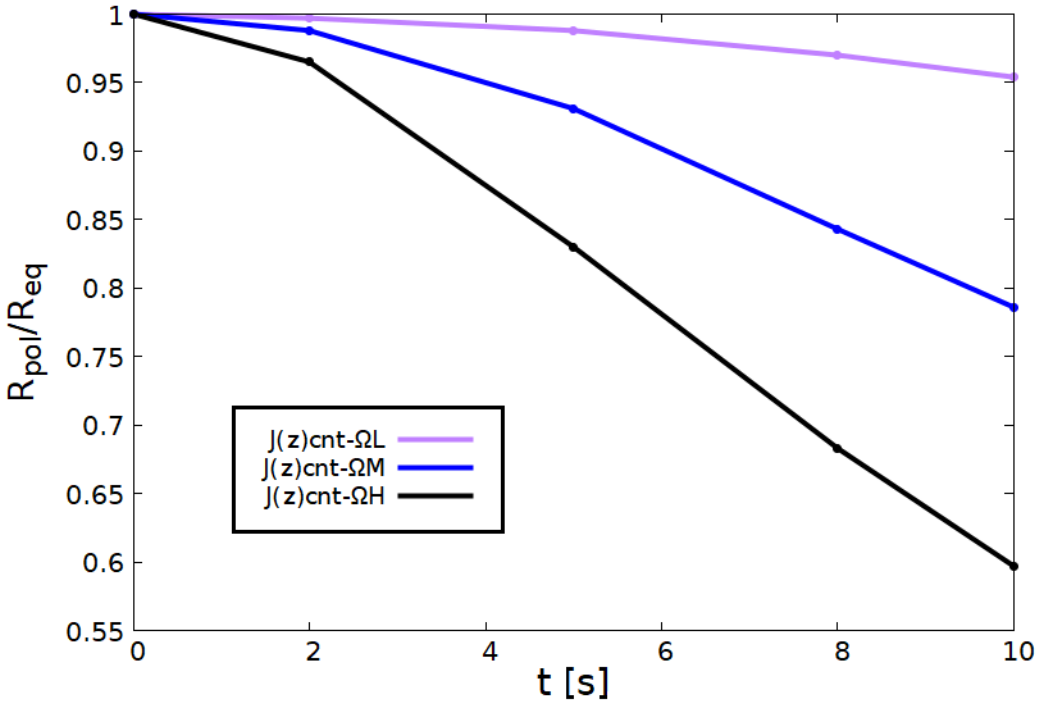

Figure 5.5: Time evolution of the ratio R_{pol}/R_{eq} for the models J(z)cnt-ΩL (purple), J(z)cnt-ΩM (blue) and J(z)cnt-ΩH (black).

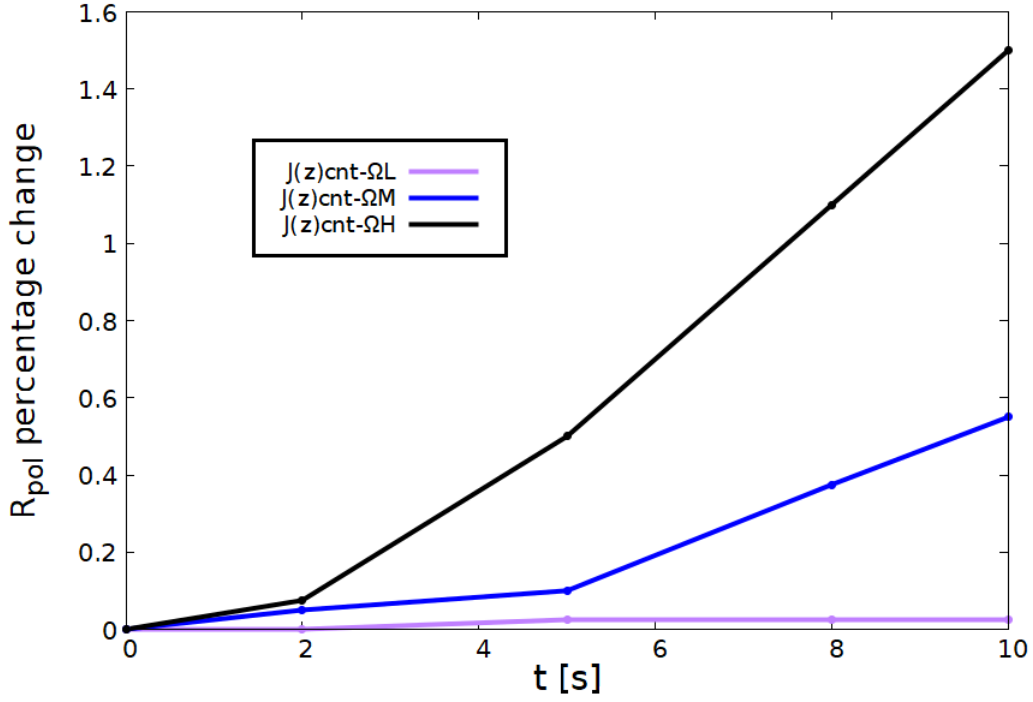

Figure 5.6: Percentage change of the polar radius $R_{\rm pol}$ compared with the initial radius of $4{\times}10^{10}$ cm for the models J(z)cnt-ΩL (purple), J(z)cnt-ΩM (blue) and J(z)cnt-ΩH (black).

jet evolution path is along the polar radius; then, although the changes in $R_{\rm eq}$ are important, there are not significant variations in $R_{\rm pol}$, so this could not affect the jet break out time.

From the study performed by Maeder & Meynet (2012), they stated the ratio value $R_{\rm pol}/R_{\rm eq}$ ∼0.7 to be the critical rotation configuration (when the centrifugal force compensates central gravity). This value was also reported by Geng et al. (2016) by applying the angular velocity Ω =5.3×10^{-3} rad s^{-1}. We obtained $R_{\rm pol}/R_{\rm eq}$=0.7 ratio with the same angular velocity configuration at t∼ 8 s. A comparison of these results is displayed in Table 5.2.

The observed behaviour coincides with the results found by Maeder & Meynet (2012) since we have critical rotation along the envelope, as they reported from previous structure models that the polar radius decreases by a few percent even with extreme rotation (high angular velocity values).

Author	$R_{\rm pol}/R_{\rm eq}$
Maeder & Meynet (2012)	0.666
Geng et al. (2016)	0.70
Our result	0.70

Table 5.2: Comparison of $R_{\rm pol}/R_{\rm eq}$ values between our results and previous studies.

Because our purpose is to study the geometry of the progenitor by taking different $J(z)$ distributions, we choose the intermediate value Ω =3.12×10^{-3} rad s^{-1}, since this is not an extreme limit on the angular velocity range but made an oblate morphology for the progenitor and is slightly above the $J_{\rm crit}$ value from the 16TI model.

5.2 Progenitors with Different Vertical Distributions

As mentioned before, we select the value Ω =3.12×10^{-3} rad s^{-1} (which remains constant) to develop three new models, now taking into account the $J(z)$ distributions. In Table 5.3, we show list the three models considered.

Model	Ω (rad s^{-1})	$J(z), J(\theta)$
J(z)lnr-ΩM	3.12×10^{-3}	$J(z) = 1 - z/5 \times 10^{10}$
J(θ)cos-ΩM	3.12×10^{-3}	$J(\theta) = 1 - cos\theta$
J(θ)cos$_2$-ΩM	3.12×10^{-3}	$J(\theta) = 1 - cos^2\theta$

Table 5.3: Specific angular momentum models by nomenclature.

We choose these $J(z)$ distributions in order to observe the effects of different functions in the final morphology of the progenitor: the cosine squared has a steeper drop for angular velocity along the polar axis in comparison with the cosine; similarly, the cosine distribution shows a smoother change in contrast to the linear distribution.

In Figure 5.7 we show an example of the velocity field due to the J(z)lnr-ΩM distribution at t=0 s. The maximum velocity differs from the previous models in Section 5.1; here, the

maximum velocity was 0.127 c, which is the 60% of the velocity found in Figure 5.1. In this plot we can notice how the rotation changes its magnitude along the z axis. In the equatorial plane the velocity reaches a maximum and it diminishes towards the poles. For the three models, this behaviour is preserved.

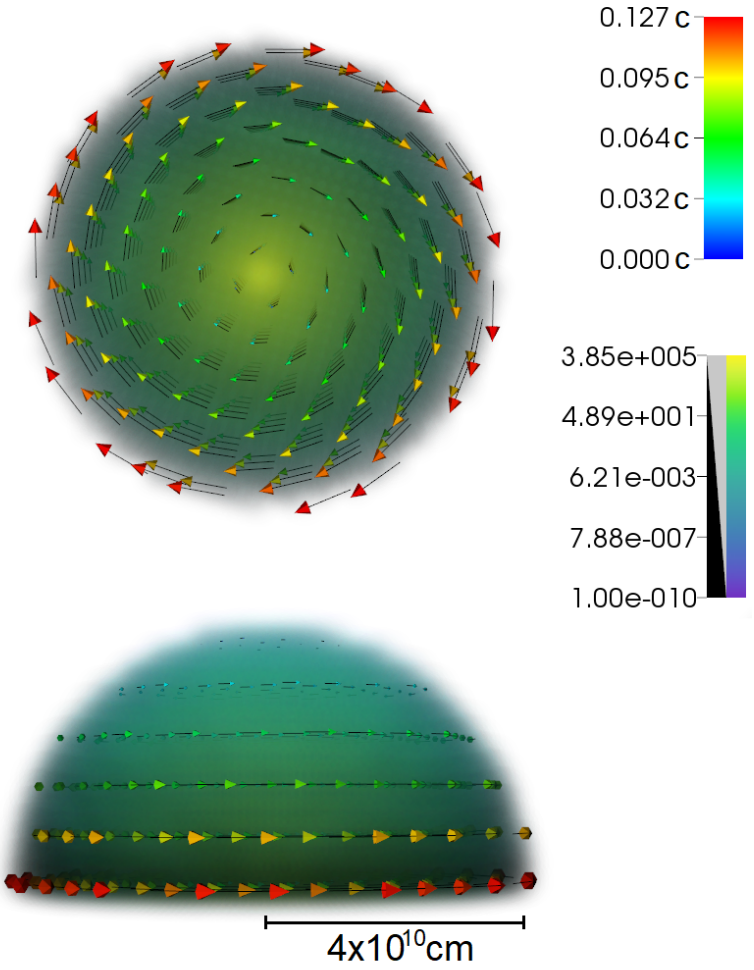

Figure 5.7: Velocity field inside the progenitor J(z)lnr-ΩM. The magnitude of the velocity field increases from the poles towards the equatorial plane. The top color palette corresponds to the velocity magnitude while the bottom one displays the density values (g cm^{-3}).

As in the previous section, we let the simulation ran for 10 s in each model starting from the same initial condition as we show in Figure 5.2. Now, in Figure 5.8 we used the same values for density contours as in Figures 5.3 and 5.4 to show the evolution at three time steps (t=0, 5 s and 10 s) of one of our models: the J(θ)$\cos_2$-ΩM model. The evolution of this model has some differences compared with the one shown in Figure 5.3; the morphology in the J(z)cnt-ΩM model remains quite spherical and, in the J(θ)$\cos_2$-ΩM model, the rotation pushes the material along the poles causing a notorious deformation in this region.

In this case, the shape of the progenitor might be analyzed as comprising two components by two components: an oblate structure along equatorial plane and a prolate one along the z

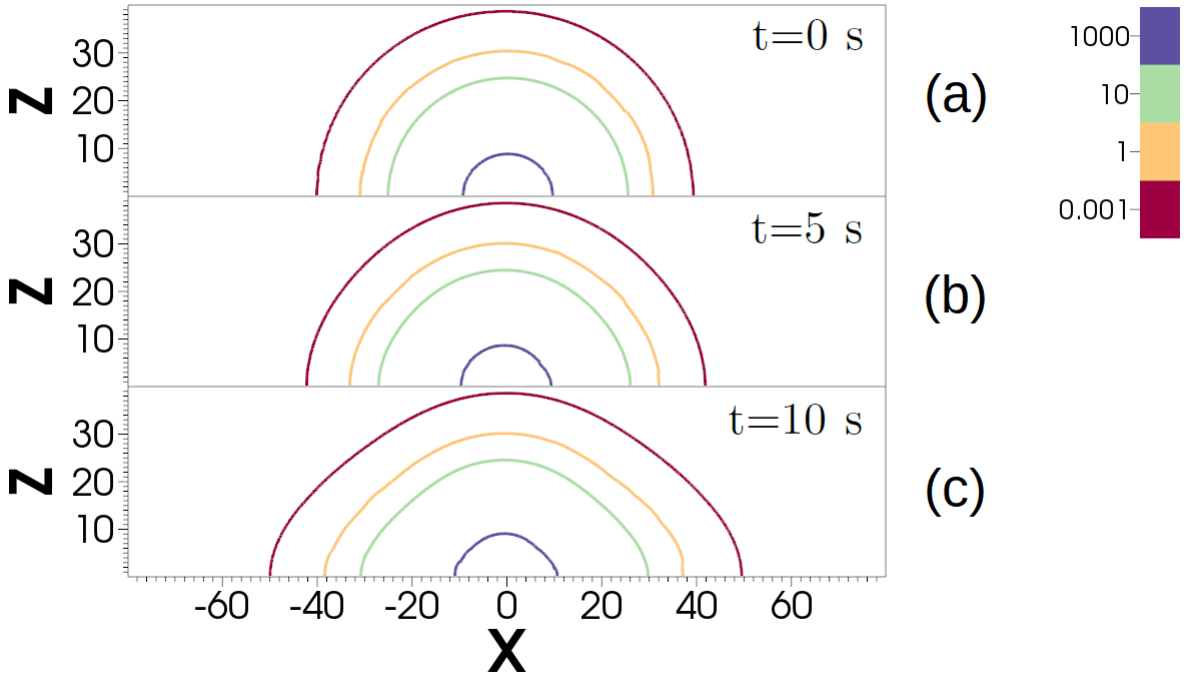

Figure 5.8: Density contours (g cm^{-3}) projected onto xz-plane for (a) t=0 s, (b) t=5 s and (c) t=10 s of the model J(z)cos$_2$-ΩM. The scale unit for each spatial axis is 10^9 cm.

axis that might be the effects of both functional forms in the angular momentum distribution: the radial component and the angular component. In Figure 5.9, a sketch of the decomposition for the J(θ)cos$_2$-ΩM model morphology is shown at t=10 s. Along the equatorial plane, we have an oblate ellipsoid (green surface) and, along the z-axis, a prolate ellipsoid is displayed (blue surface); both of them form a first approximation for the overall morphology (red surface).

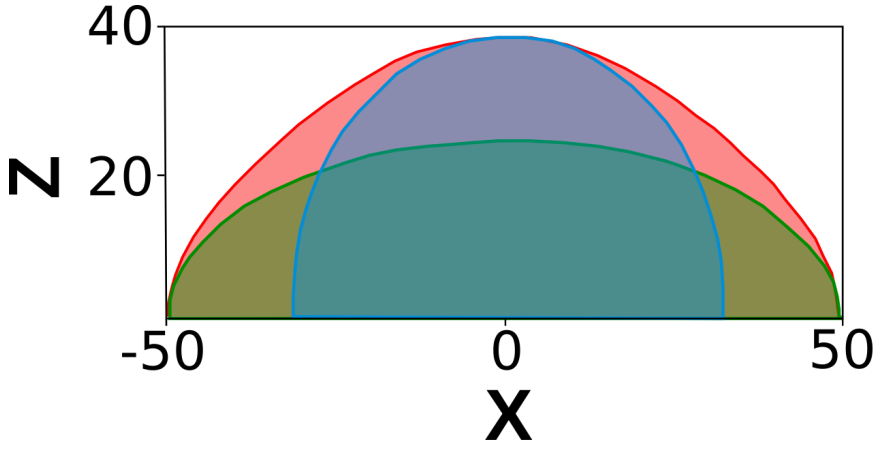

Figure 5.9: 2D composition for the J(θ)cos$_2$-ΩM model. The green shade represents the oblate component and the blue shade shows the prolate component. The red shade is the actual final morphology for this model at t=10 s.

We write the implicit equation for the 3D ellipsoid and find the values of the semi-axes of

the two structures (oblate and prolate) that compose the J(θ)cos$_2$-ΩM model, then, we compare them with the semi-axes values of the J(z)cnt-ΩM model from the previous section, which had the same angular velocity value (3.12×10^{-3} rad s^{-1}) but no major contribution for the $J(z)$ distribution. The general equation is,

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=1. \tag{5.1}$$

where, a is the semi-axis in the x direction, b is the semi-axis in y direction and c, the semi-axis in z direction. In Table 5.4, we show the values of these semi-axes for the prolate and oblate structures of J(θ)cos$_2$-ΩM model (first two lines) and the values for J(z)cnt-ΩM model (third line). The a and b semi-axes have the same value for the J(z)cnt-ΩM model and the oblate substructure of J(θ)cos$_2$-ΩM model, so the main deformation takes place along the z-axis, where the prolate component becomes more important.

Model	a ($\times10^{10}$cm)	b ($\times10^{10}$cm)	c ($\times10^{10}$cm)
J(θ)cos$_2$-ΩM (prolate)	3.2	3.2	4.005
J(θ)cos$_2$-ΩM (oblate)	5.06	5.06	2.8
J(z)cnt-ΩM	5.06	5.06	3.97

Table 5.4: Semi-axes values for the two components of J(θ)cos$_2$-ΩM model and the J(z)cnt-ΩM model.

For the comparison, we select the components in z-axis (from the prolate spheroid) and x-axis (from the oblate spheroid) of the J(θ)cos$_2$-ΩM model approximation. Taking the values for the semi-axes along x direction and z direction (there is symmetry for x- and y-axes) from both models and comparing them we get,

$$A=\frac{a_{cnt}}{a_{cos}}=1, \qquad C=\frac{c_{cnt}}{c_{cos}}=0.991; \tag{5.2}$$

where a_{cnt} and a_{cos} are the semi-axes in x direction for the J(z)cnt-ΩM and J(θ)cos$_2$-ΩM models respectively, c_{cnt} and c_{cos} are the semi-axes in z direction for the same models. When decomposing the geometry of the J(θ)cos$_2$-ΩM model, we notice that the principal modification of the morphology due to two types of rotation appears in the poles of the progenitor, so it may affect the jet propagation.

Summarizing the $J(z)$ distribution models, in Figure 5.10 are shown density contours for each model at t=10 s. Here we have to highlight the different shapes reached at the final evolution for the three models. When we take a linear distribution along the z axis (first panel in Figure 5.10), the density distribution is not far from the results obtained in Section 5.1 for a constant $J(z)$ distribution; in fact, they are quite similar and the oblate shape is preserved. Nevertheless, the next two models do not present this kind of shape.

If we take a look at the J(θ)cos-ΩM model, the structure is mainly modified along z axis, so a prolate shape is formed. As we mentioned before, in the J(θ)cos$_2$-ΩM model we find a similar behaviour in the z-axis, however, the oblate distribution in this model made a softer boundary in xy plane unlike J(θ)cos-ΩM model. This is an important feature to notice because, in the

case of the model from panel (b) has a not physically correct behaviour in the equatorial plane, so we can neglect this model.

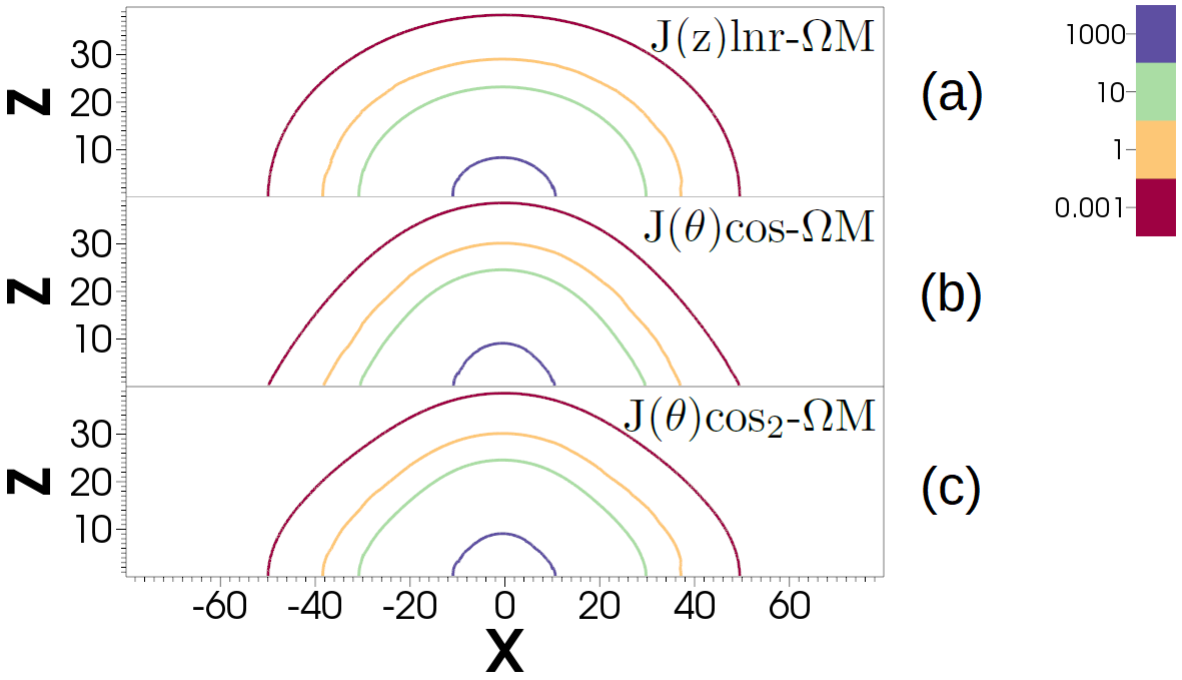

Figure 5.10: Density contours (g cm^{-3}) projected into xz-plane for the models (a) J(z)lnr-ΩM, (b) J(θ)cos-ΩM, (c) J(θ)cos$_2$-ΩM at t=10 s. The scale unit for each spatial axis is 10^9 cm.

In Figure 5.11, time evolution of the ratio $R_{\rm pol}/R_{\rm eq}$ is also plotted. The changes in this ratio is not as noticeable as in the models with different Ω values; for the vertical distributions this ratio had a very similar behaviour.

Finally, in Figure 5.12, we show the percentage change for $R_{\rm pol}$ as in the previous section. Here, the change for the J(θ)cos-ΩM and J(θ)cos$_2$-ΩM models, in contrast to the previous cases, adopted negative values because $R_{\rm pol}$ expands due to the contribution of the $J(\theta)$ distribution. These distributions for the angular momentum (cosines) allow the material to accumulate along the z direction, and, as a consequence, the polar radius increases. Contrary to these results, in the linear model we find a contraction of $R_{\rm pol}$, so the poles contracted. From this behaviour we expect different values for the jet's break out time since the polar radius increases for J(θ)cos$_2$-ΩM and J(θ)cos-ΩM models in comparison with the J(z)lnr-ΩM model.

Considering this analysis, we select two final models to apply the jet injection at $R_i = 2\times10^9$ cm. These models are J(z)lnr-ΩM and J(θ)cos$_2$-ΩM. Because of their final shapes and rotation profiles, we expect to have variations in the jet break out properties in this cases. The results involving the jet evolution through these progenitors are discussed in the next section.

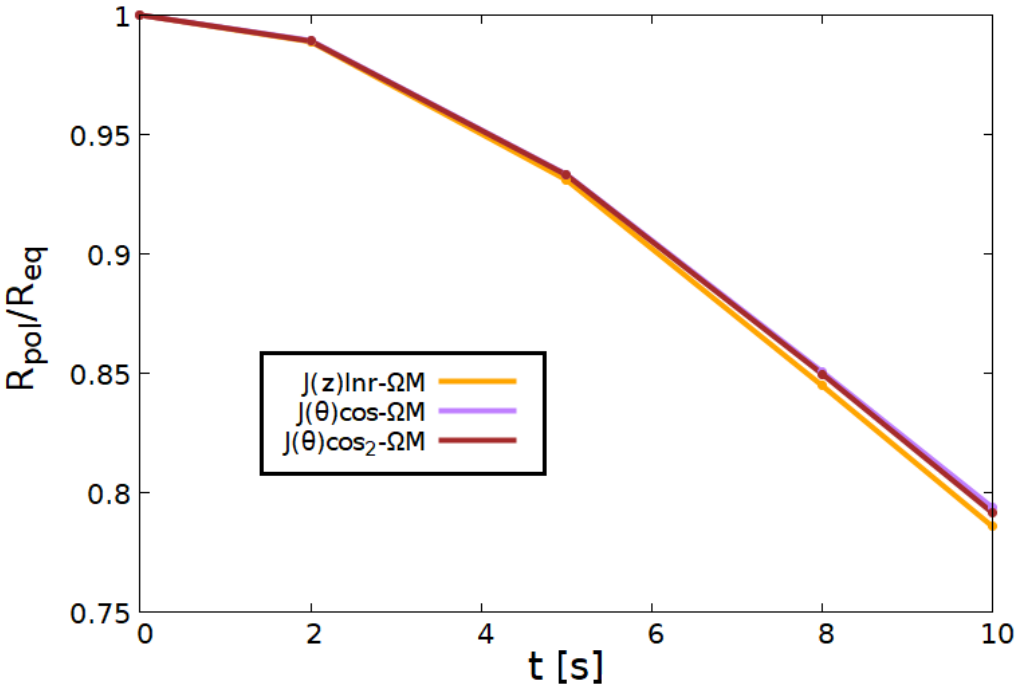

Figure 5.11: Time evolution of the ratio $R_{\rm pol}/R_{\rm eq}$ for the models J(z)lnr-ΩM (orange), J(θ)cos-ΩM (purple) and J(θ)cos$_2$-ΩM (red).

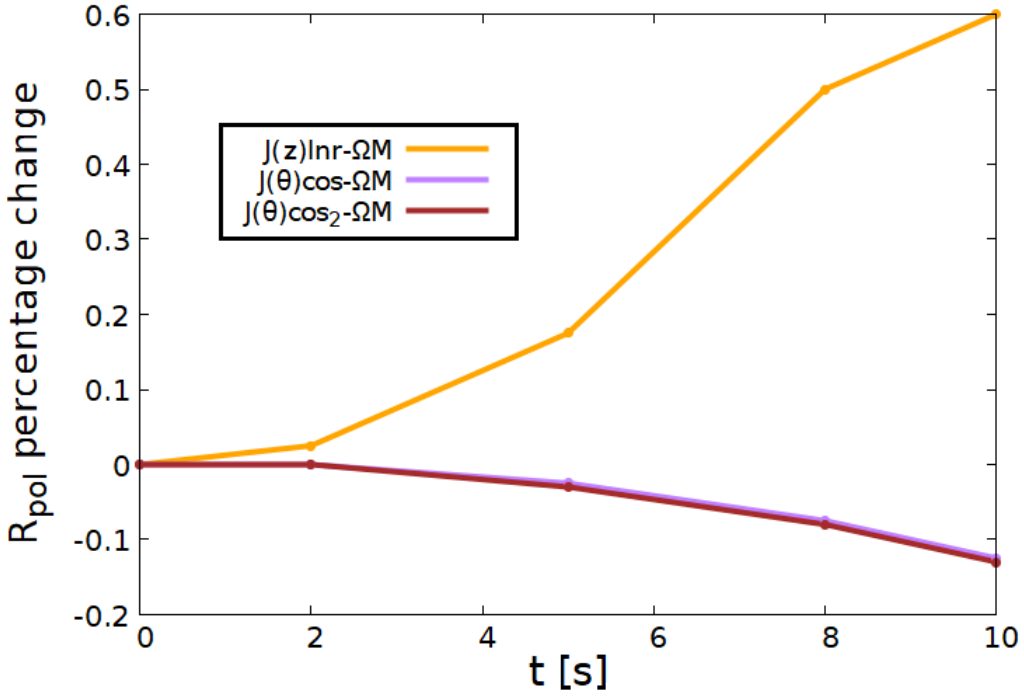

Figure 5.12: Percentage change of the polar radius $R_{\rm pol}$ compared to the initial radius of $4{\times}10^{10}$ cm for the models J(z)lnr-ΩM (orange), J(θ)cos-ΩM (purple) and J(θ)cos$_2$-ΩM (red). The negative values indicate an expansion from the initial radius.

5.3 Jet Injection in the J(z)lnr-ΩM and J(θ)cos$_2$-ΩM Models

In this final section, we present an analysis of the jet's properties using the two selected progenitor's models, J(z)lnr-ΩM and J(θ)cos$_2$-ΩM. We also show a comparison between these models with a control test with a non-rotating progenitor.

To perform the simulations, we let the rotating progenitors evolve for 10 s until is reached the deformation related to the extreme ratio of the polar radius to the equatorial radii (Maeder & Meynet, 2012) that we studied in the previous section, and then we imposed the injection of the jet. We stop the simulations once the jet reaches the progenitor's surface (break out). We expect the jet to break through the envelope in the two models considered here.

In Figure 5.13, the results from the simulation with the J(z)lnr-ΩM model are plotted for two different times: t=2 s and t=3 s using Lorentz factor contours. The velocity field for this distribution inside the progenitor is also shown as a vector field. From this figure, we can observe the distribution for the jet's Lorentz factor and its overall structure in 3D. In the upper panel, which corresponds to t=2 s, the jet has travelled up to $\sim 1.7\times10^{10}$ cm and at this point is has reached a maximum Lorentz factor of $\Gamma = 7.11$, for the lower panel at t=3 s, the break out of the jet is attained, so it crosses successfully the envelope of the progenitor. The maximum Lorentz factor in this case is $\Gamma = 17.4$.

Next, we compare the density profiles for the J(z)lnr-ΩM and J(θ)cos$_2$-ΩM models with the non-rotating model in two different time steps, including the break out time for both models. The comparison with the J(θ)cos$_2$-ΩM model is shown in Figure 5.14; in Figure 5.15 we show the comparison with the J(z)lnr-ΩM model at the same times. Both figures are obtained by taking a slice of the 3D density plot (similar to Figure 5.13) comprising the xz-plane (y=0).

From these plots, we would like to highlight some important features. Firstly, the differences between the distance travelled by the jet in the first two panels of Figure 5.15. As we can notice, when a progenitor with a linear angular momentum distribution is chosen, the structure of the progenitor is modified because of its elongation which causes the stellar matter to rearranged and to drop along the polar region so it allows the jet to propagate along z direction more easily (see Figure 5.3 and 5.4). However, as we can see in the case of the J(θ)cos$_2$-ΩM model, the $R_{\rm pol}$ has expanded as compared to the in comparison with the linear model, nonetheless, the angular momentum implementation also allowed the jet from this model to travel more distance because of the same reasoning as in the case of linear angular momentum distribution; the material in the poles decrease, so the jet must travel a lower distance to reach the surface of the progenitor.

Another consequence of these changes in morphology is the decrease of the breakout time when considering the rotation. For the J(θ)cos$_2$-ΩM model, the jet reaches the break out time at $t_{\rm bo}$=3.2 s, for the J(z)lnr-ΩM, the break out time occurs at $t_{\rm bo}$=3.0 s and for the non-rotating model, $t_{\rm bo}$=3.6 s. It is particularly interesting that, although in both rotating models the value of $t_{\rm bo}$ decrease, in the J(z)lnr-ΩM model, we obtain the lowest value for the $t_{\rm bo}$. This is due to the differences in the internal structure (density and pressure) of each progenitor caused by the rotation implementation discussed before.

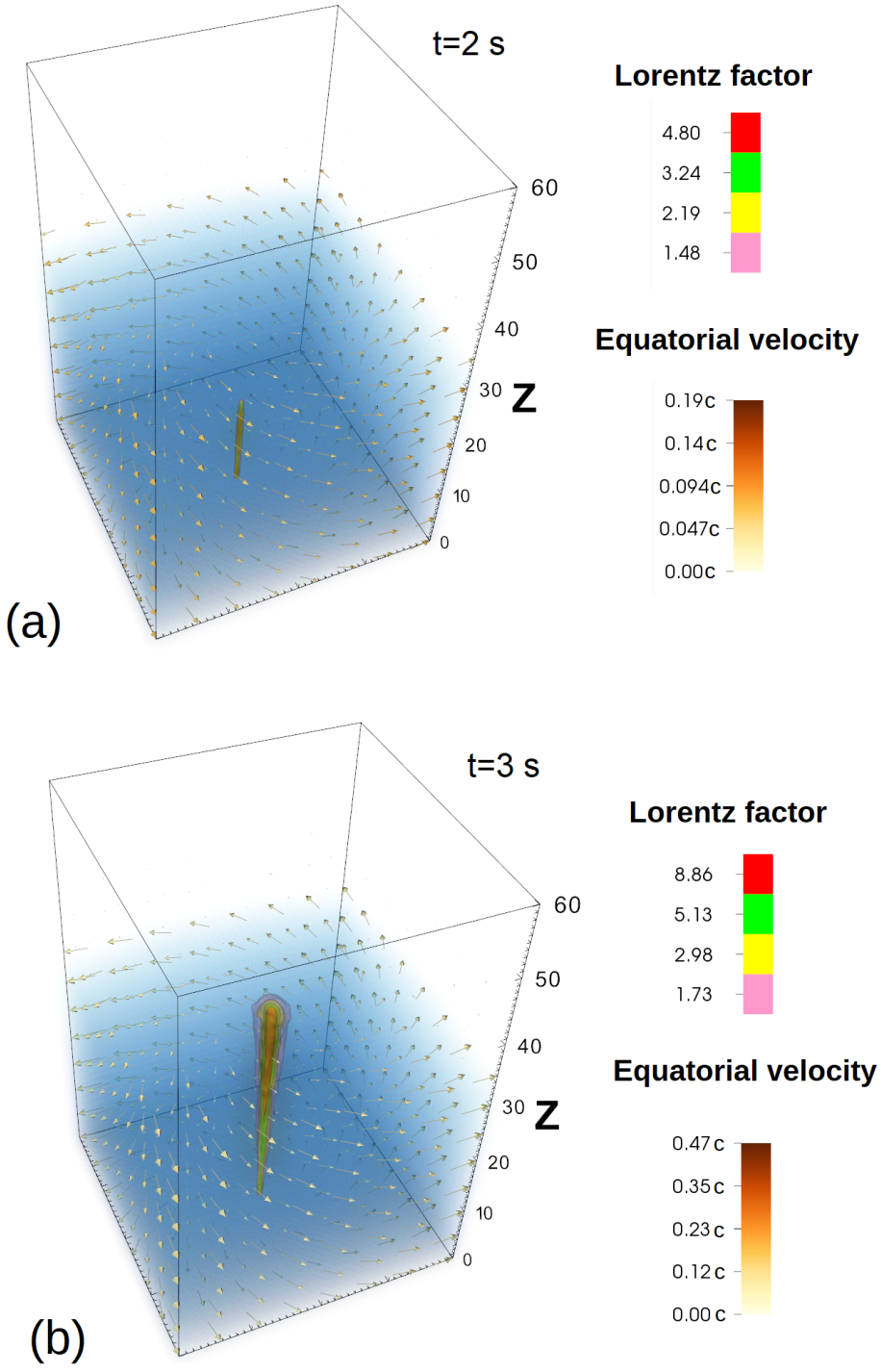

Figure 5.13: Evolution of the relativistic jet with J(z)lnr-ΩM model. In this plot, we are showing the density volume of the progenitor, the velocity field and the jet's Lorentz factor contours at (a)t=2.0 s and (b)t=3.0 s from the jet injection. The scale unit for each spatial axis is 10^9 cm.

As we observed in the previous section, after 10 s, the J(θ)cos$_2$-ΩM progenitor reaches a morphology were we can find two contributions: an oblate one caused by the constant angular

velocity and a prolate one, due to the angular momentum distribution along the z-axis. This last contribution made the material to accumulate along the jet axis (towards the poles of the progenitor) and, as a consequence, the break out time is larger than in the J(z)lnr-ΩM case by ∼6.66%. Nevertheless, the differences of $t_{\rm bo}$ between these models are not large.

Another important feature that we can see from these figures is the shocked jet structure. From Figure 5.14, we can note a difference in the shocked jet morphology between the two last panels, 1.(b) and 2.(b). In the non-rotating progenitor, the shocked jet density remains almost constant along its evolution, unlike the jet from J(θ)cos$_2$-ΩM model, where some turbulence is shown under z=2.5×10^{10} cm, plotted in the green coloured region with $\rho \sim 10^{-3} - 10^{-4}$ g cm^{-3}. This feature is also apparent in the J(z)lnr-ΩM model from Figure 5.15, 1.(b) panel, below z=2.5×10^{10} cm in the same density ranges.

By looking at Figure 5.15, we can notice a difference in the cocoon radius at the break out time. In the non-rotating progenitor, the cocoon radius is smaller compared to the cocoon from J(z)lnr-ΩM model. This is not so obvious in the J(θ)cos$_2$-ΩM model but is also caused again by the angular momentum distribution; the rearrangement of the material due to the rotation allows the cocoon to spread in the perpendicular direction of its propagation. We will recall this result later to make further analysis.

The next analysis concerns the behaviour of the Lorentz factor during the jet's evolution and how it is related to the break out time $t_{\rm bo}$. For the former, we show the Lorentz factor and density contours in Figure 5.16, comparing the J(θ)cos$_2$-ΩM model with the non-rotating case, and in Figure 5.17, for the J(z)lnr-ΩM model.

In both figures we show the minimum and maximum values for the Lorentz factor at the bottom of the color bar. From Figure 5.16, we observe that the jet's Lorentz factor for the rotating model (which is the J(θ)cos$_2$-ΩM) is higher than the value obtained for the non-rotating model by ∼ 11.2%. Considering the previous discussion, we can associate this results with the morphology changes caused by the angular momentum. Because the internal structure of the progenitor is restructured due to rotation, the density rearranges and allows the jet to propagate more easily, reaching a higher Lorentz factor and breaking through the progenitor in less time. A similar reasoning explains the Lorentz factor values shown in Figure 5.17, where in fact, the Lorentz factor for the J(z)lnr-ΩM model is even higher than in the J(θ)cos$_2$-ΩM case.

Since the rotation setup seems to have an impact on the evolution of the jet, if a progenitor has a certain type of angular momentum distribution, this can be an important factor on the break out time and, consequently on, the duration of the prompt phase.

In Table 5.5, we present a table with a summary of the jet features at break out from each model. In the first column the models are listed, in the second column we show the break out times ($t_{\rm bo}$), in the third column, the maximum Lorentz factor reached at break out time is listed ($\Gamma_{\rm bo,max}$) and finally, in the fourth column, we show the cocoon width at z=1.5×10^{10} cm ($W_{C,1.5e10}$). The width of the cocoon was measured using a jump in density to distinguish between the inner low-density region of the jet and the stellar material that forms the cocoon.

Going into a deeper analysis of the cocoon's development throughout the simulation, we

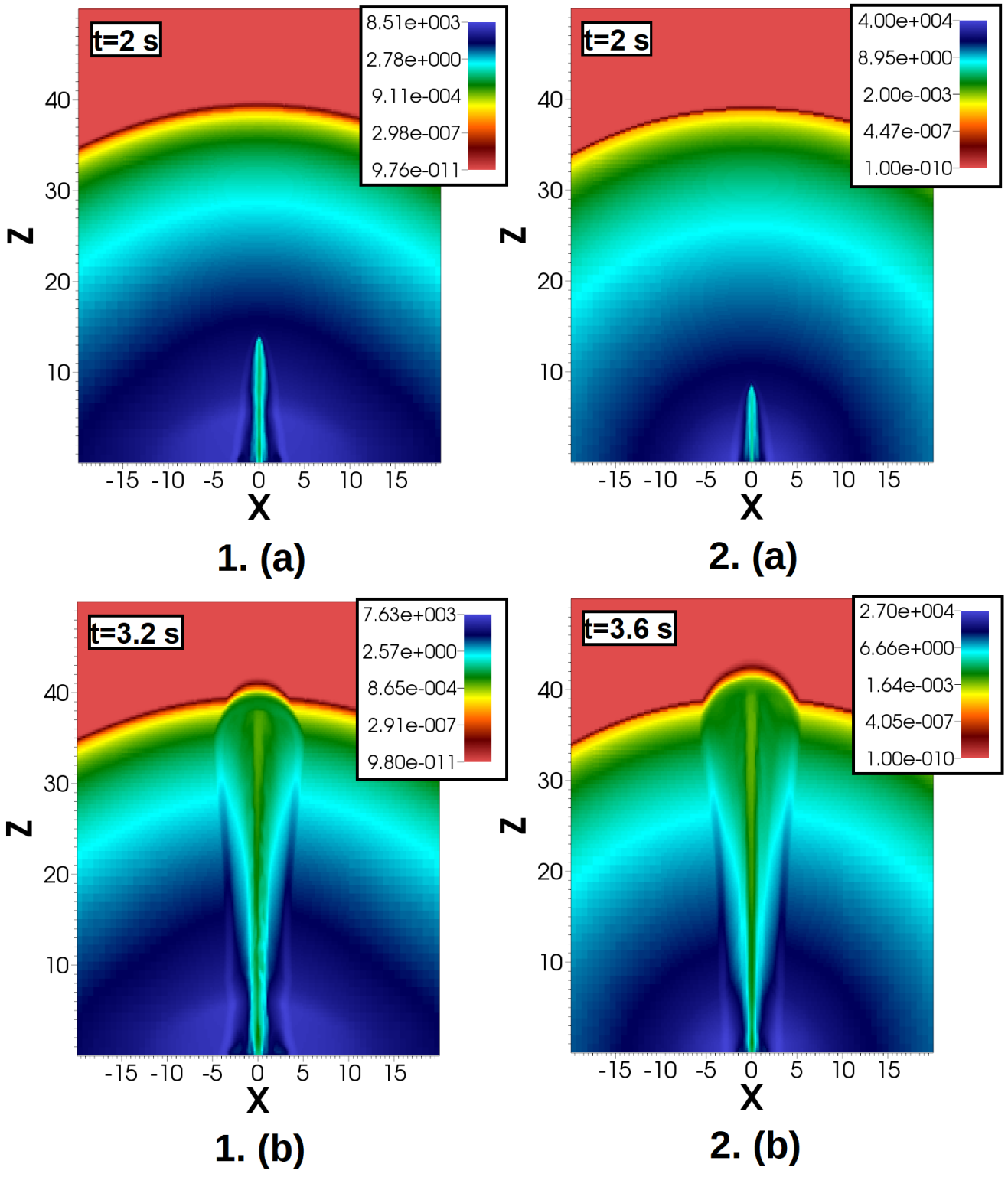

Figure 5.14: Comparison between density profiles (g cm^{-3}) of (1)J(θ)cos$_2$-ΩM model and (2)non-rotating model. Here, we are plotting the jet evolution at (a) t=2 s and (b) t=$t_{\rm bo}$, the break out time for each model. The break out time for the first model is $t_{\rm bo}$=3.2 s and for the second model is $t_{\rm bo}$=3.6 s. The scale unit for each spatial axis is 10^9 cm.

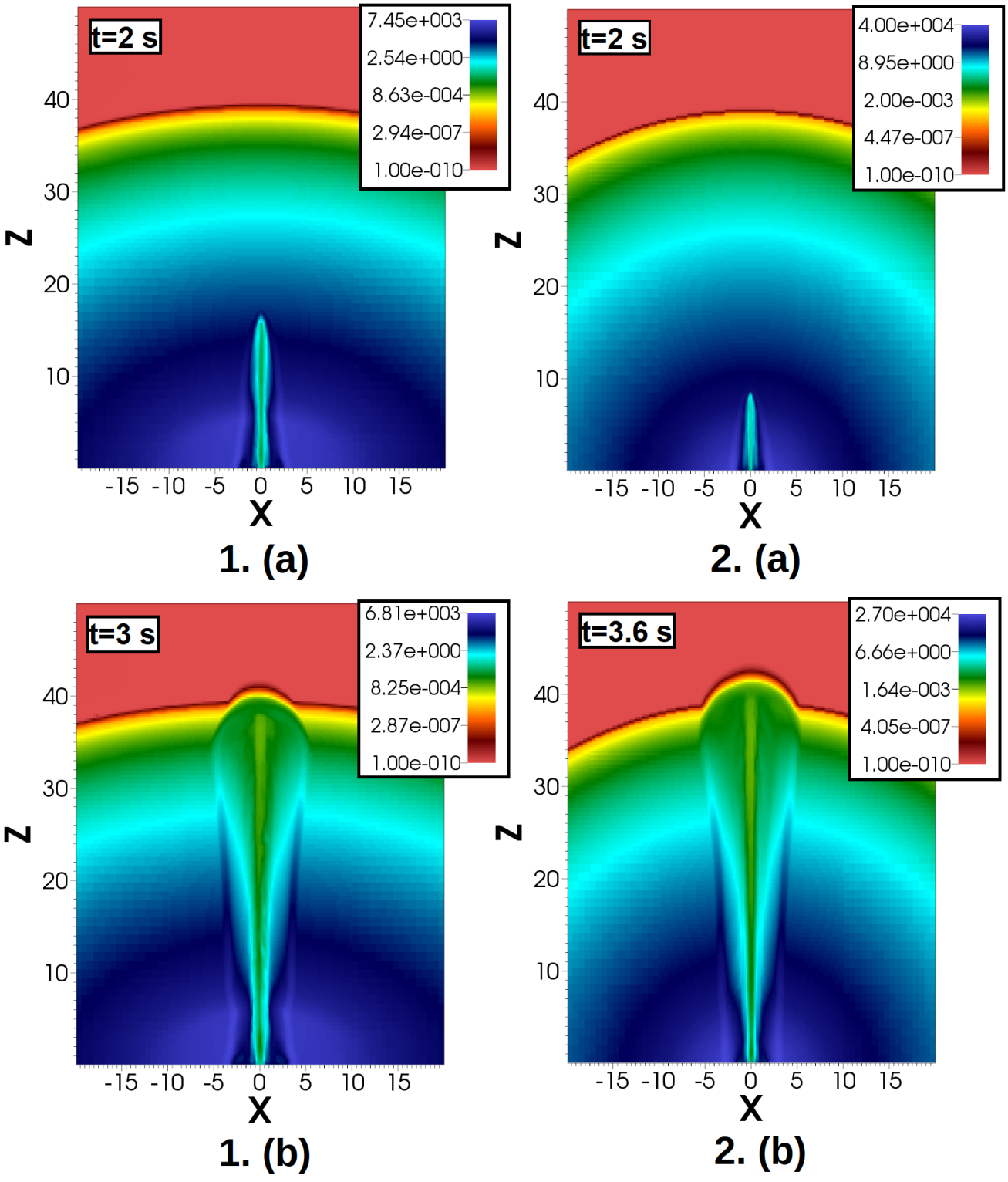

Figure 5.15: Comparison between density profiles (g cm^{-3}) of (1) J(z)lnr-ΩM model and (2) non-rotating model. Here, we are plotting the jet evolution at (a) t=2 s and (b) t=$t_{\rm bo}$, the break out time. The break out time for the first model is $t_{\rm bo}$=3.0 s and for the second model is $t_{\rm bo}$=3.6 s. The scale unit for each spatial axis is 10^9 cm.

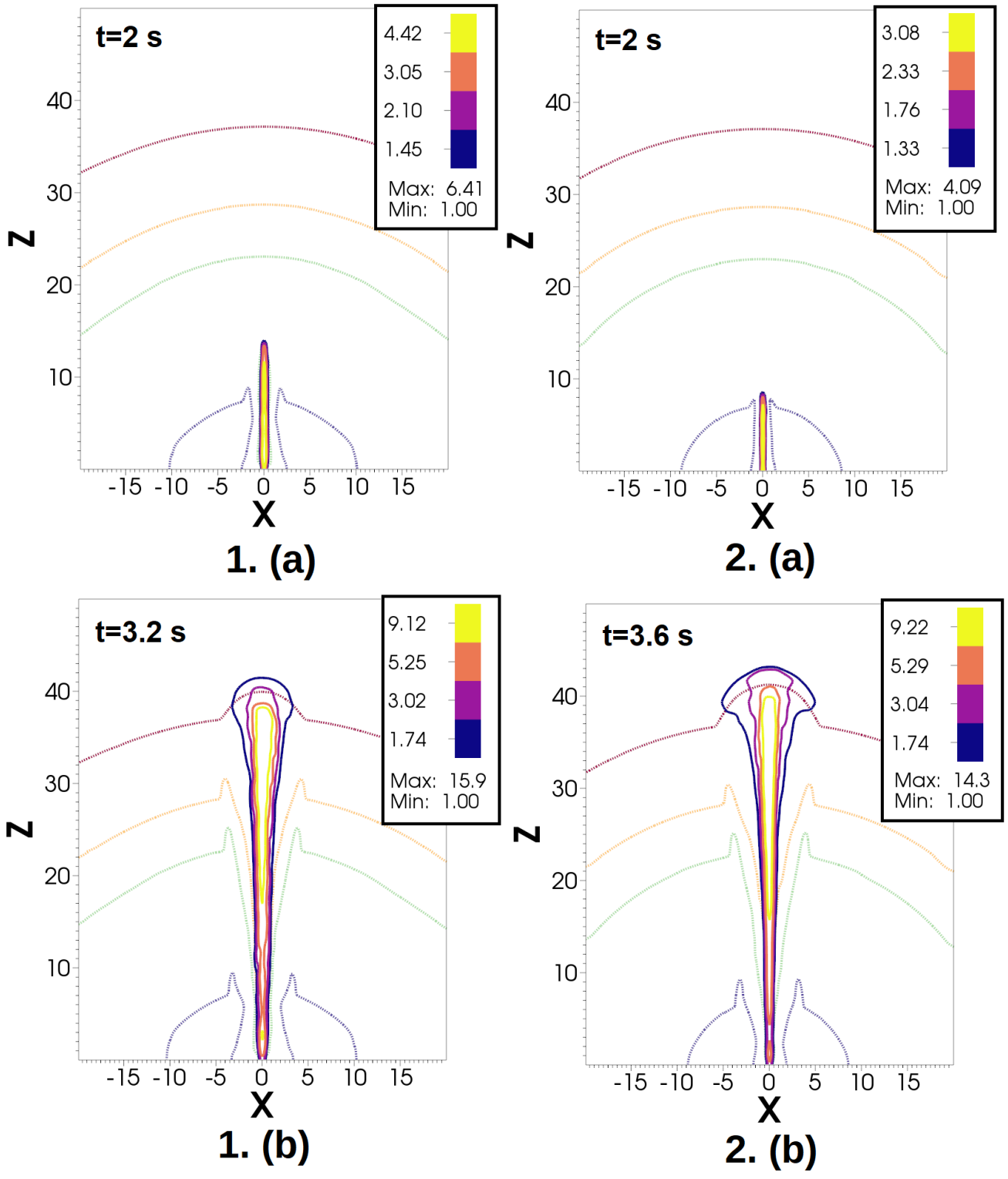

Figure 5.16: Lorentz factor (solid lines) and density (dotted lines) contours for (1) J(θ)cos$_2$-ΩM model and (2) non-rotating model. We are showing the results at the same times as in the previous figures; (a) t=2 s and (b) the break out time. The scale unit for each spatial axis is 10^9 cm.

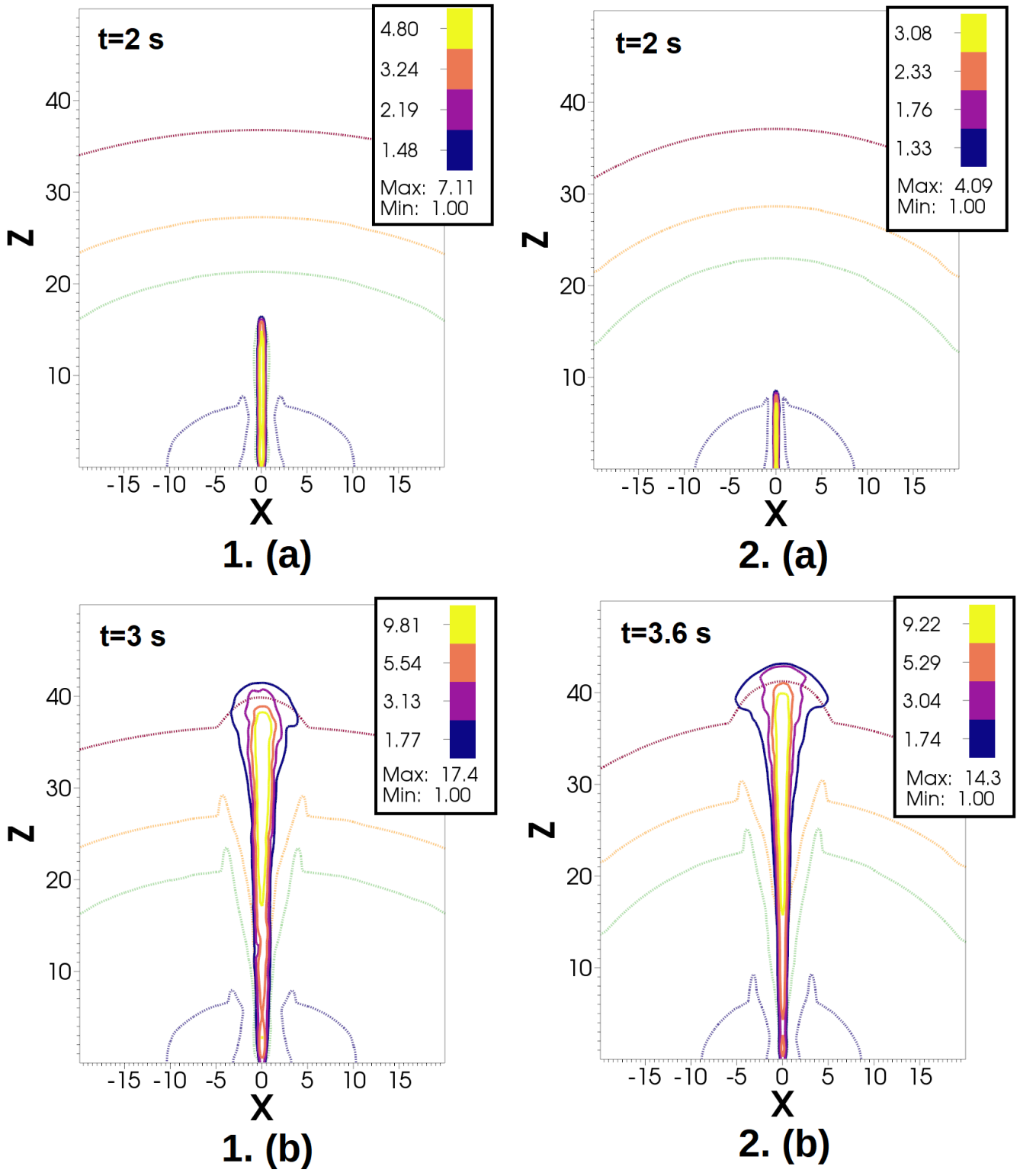

Figure 5.17: Lorentz factor (solid lines) and density (dotted lines) contours for (1) J(z)lnr-ΩM model and (2) non-rotating model. We are showing the results at the same times as in the previous figures; (a) t=2 s and (b) t=$t_{\rm bo}$, the break out time. The scale unit for each spatial axis is 10^9 cm.

Model	t_{bo} (s)	$\Gamma_{bo,max}$	$W_{C,1.5e10}$ (cm)
J(z)lnr-ΩM	3.0	17.4	3.2×10^9
J(θ)cos$_2$-ΩM	3.2	15.9	2.4×10^9
non-rotating	3.6	14.3	2.8×10^9

Table 5.5: Features from each model (first column) at the jet's break out: in the second column, we show the break out time (t_{bo}); in the third column, the maximum Lorenz factor reached at t_{bo} (Γ_{bo}) is listed and in the fourth column we show the width of the cocoon .

perform now a study on the cocoon's width which is measured at half of the total length of the jet, $W_{C,FWHM}$, (similarly as the Full Width at Half Maximum applied in the analysis of some functions) and by examining the jump in pressure to distinguish it from the rest of the progenitor's structure. In Figure 5.18, the temporal evolution of $W_{C,FWHM}$ is plotted for each model from $t = 0$ up to their break out times. We point out the behaviour of the two rotating models up until t=1.5 s, where both of them have an equal cocoon's width (above 2×10^9 cm), after this time, the J(z)lnr-ΩM model width expands faster than in the J(θ)cos$_2$-ΩM model. The non-rotating model had smaller width values than the two rotating models at all times (before t=1.5 s, it never exceeded 1.5×10^9 cm).

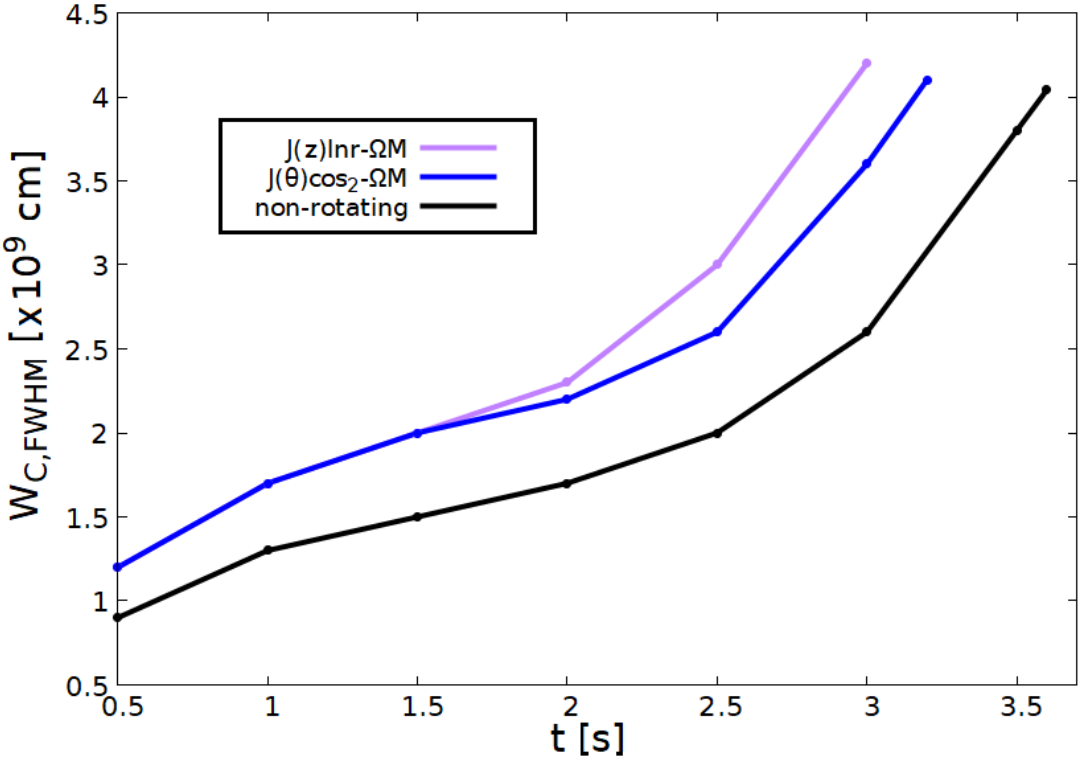

Figure 5.18: Cocoon width at half of the total length of the jet ($W_{C,FWHM}$) at different times for the three models. The final points in the curves show the value of $W_{C,FWHM}$ at the breakout time.

The cocoon spreads differently on the three models depending on the angular momentum profile. With the distribution set for J(θ)cos$_2$-ΩM, the jet and, consequently, its cocoon are confined to a small region compared to the J(z)lnr-ΩM model. By assuming the J(z)lnr-ΩM

model, this leads to a wider opening of the cocoon. Nevertheless, the spreading of the cocoon becomes important when angular momentum is considered, regardless of what kind of distribution is considered.

Next, we show the temporal evolution of $\Gamma_{\max}$ for the three models. For the three models in Figure 5.19. In this figure, we can observe that at early times (before t=1.5 s), the maximum values reached by both rotating models are very similar (between Γ_{max}=4 and Γ_{max}=6) and, as the jet evolves, the J(z)lnr-ΩM model reaches higher values. A possible explanation for this behaviour maybe be given by recalling the previous discussion about the link between the morphology and the angular momentum for these models. The values of $\Gamma_{\max}$ in the non-rotating model stay below the corresponding values of the rotating models ones (until t=1.5 s, these values did not exceeded Γ_{max}=4).

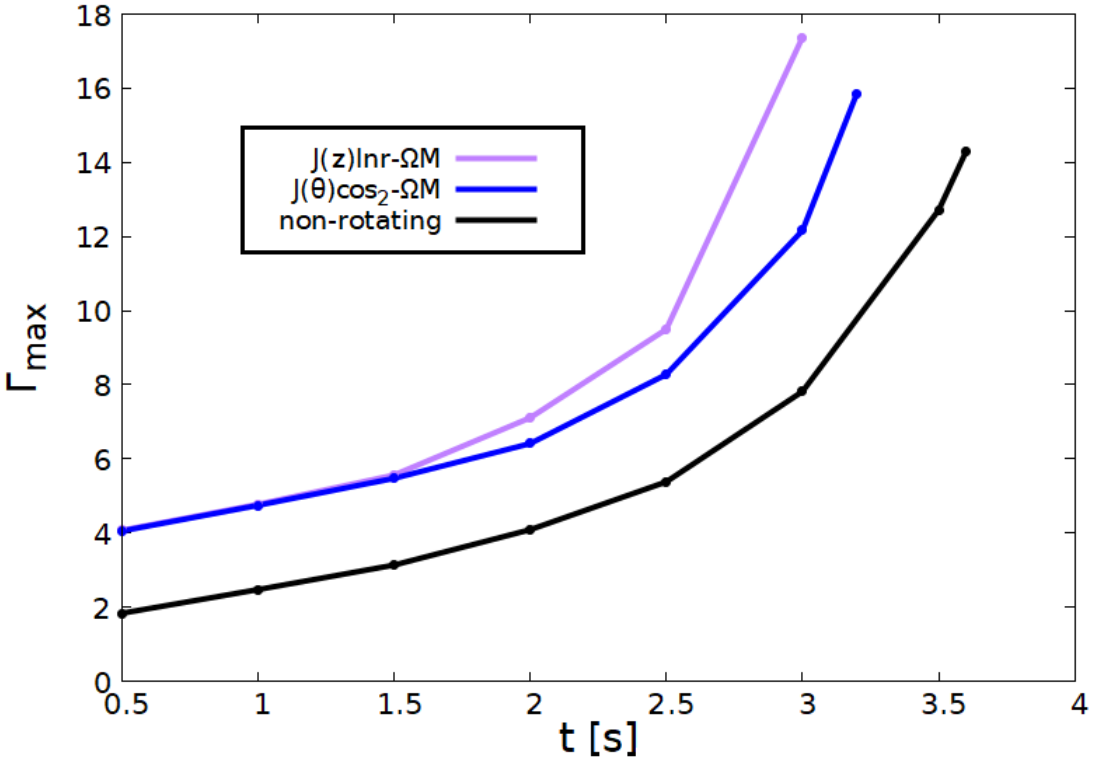

Figure 5.19: Maximum Lorentz factor at different times for the three models. The final points in the curves show the value of $\Gamma_{\max}$ at the break out.

Considering the maximum Lorentz factor evolution for the three models, we can conclude that the break out times shown in Table 5.5 are consistent with the properties of each model. When the jet reaches higher Lorentz factor values, it implies a lower break out time since the flux has higher velocities and this is an important factor derived from the rotating settings.

We can use the previous results and carry the analysis further by following the prompt emission physics. First of all, we can examine some properties from the simulations results such as the velocities reached by the jet and its density.

From linear momentum conservation, if we have a system in which $p = \rho v$ =constant (momentum per volume unit) and the velocity increases (in this case, we can derive the velocity from the Lorentz factor) then, the density must drop to conserve the momentum. From our simulations, we can observe that this is actually obtained at the jet's head (see Figure 5.14 and Figure 5.15, at z$\sim 3.7\times10^{10}$ cm); in the rotating progenitors where the Lorentz factor is higher, the density has lower values ($\sim 10^{-4}$ g cm^{-3}) than the non-rotating model ($\sim 10^{-3}$ g cm^{-3}).

Furthermore, the density distribution provides substantial information about the photospheric radius, $R_{\rm ph}$. The prompt emission could be measured using the photospheric radius (when the gamma photons are able to propagate since the medium is optically thin) and the synchrotron emission (due to particle acceleration via magnetic fields). In this case, we will reduce the analysis to the photospheric models (e.g. Mészáros et al., 2002; Daigne and Mochkovitch, 2002).

The optical depth τ has a dependency on the extinction coefficient α [cm^{-1}], which can be rewritten in terms of the mass density ρ [g cm^{-3}] and the opacity κ [cm^2 g^{-1}], giving us

$$\tau = \int_{z_0}^{z} \rho\kappa dz' \tag{5.3}$$

where z is the depth of the medium. From this expression, we can see that the optical depth is proportional to the density and inversely proportional to the velocity (by replacing $\rho = p/v$, assuming momentum conservation) at the break out if we consider a constant opacity. In the rotating models, the density drops as an consequence of the momentum conservation and this directly affects the optical depth. If the optical depth is reduced because of the density, this might imply that the jet reaches the photospheric radius earlier for a rotating progenitor than for a non-rotating one. This result might be relevant in the lGRBs framework, the evidence found in the simulations suggests that the rotation might affect the prompt emission time scales as a result from the jet's structure formed inside a rotating progenitor.

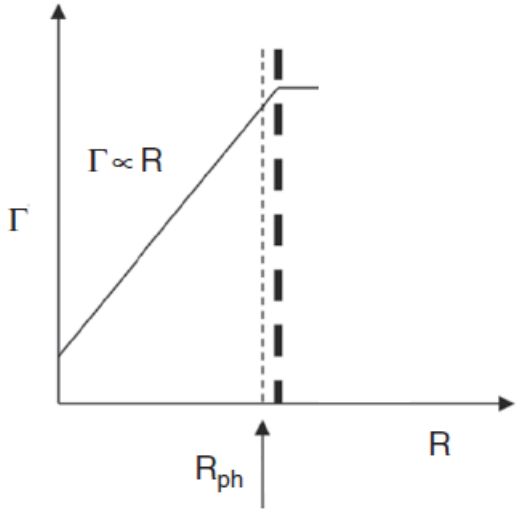

Figure 5.20: Lorentz factor (Γ) evolution during the first stage (acceleration phase) of the fireball model (Zhang, 2018).

If we assume the Lorentz factor evolution from the so called fireball model at the accelerating

phase(studied in detail by Mészáros et al., 1993b; Pirán et al., 1993 and Kobayashi et al., 1999), we can write an expression for the Lorentz factor and the photospheric radius with some approximations.

The first phase of this model, called the acceleration phase is delimited by the photospheric radius, $R_{\rm ph}$, which traces the region where the photons within the fireball are optically thin (or $\tau_{\rm ph} = 1$). Here, the outflow is ejected from the central engine and accelerates (the fireball expands), in such a way that the Lorentz factor scales as $\Gamma \sim R$, with R the radius of the explosion (Figure 5.20).

Next, at the end of the accelerating phase, we have the coasting radius, $R_{\rm c}$, when the maximum Lorentz factor is reached and the front shock enters a constant-velocity stage, since the swept material accumulates on the shock front and slows down its propagation. Depending on the regime, the shell reaches the coasting phase at the photosphere (the maximum Lorentz factor is attained) or, in the opposite case, the photospheric radius is lower than the coasting radius (many photons escape before the shell accelerates to its maximum Lorentz factor).

In our case, we are focused in the region associated with the first photons which are able to escape the fireball and which consist of gamma and hard X-rays (and then produce the prompt phase). This phase is delimited by $\Gamma \sim$constant. If we take that $R_{\rm ph} \sim R_{\rm c}$, then we can write the energy at this location as (assuming an spherical explosion)

$$E \sim \Gamma M_{\rm a} c^2 \sim \Gamma R_{\rm ph}^3 \rho_a c^2, \tag{5.4}$$

where E is the energy of the explosion, $M_{\rm a}$ the accumulated mass, $\rho_{\rm a}$ the medium density and Γ, the bulk Lorentz factor. Then, we get

$$R_{\rm ph} \sim \left(\frac{E}{\rho_{\rm a} \Gamma c^2} \right)^{1/3}. \tag{5.5}$$

From this equation we can see that, when the Lorentz factor increases, the photospheric (or coasting radius) will decrease. Nevertheless, we can improve this calculation by taking an specific regime for the photospheric radius (Mészáros and Rees, 2000b): when the fireball already reaches the coasting phase at the photosphere and when the fireball does not reach the maximum Lorentz factor.

Finally, considering the jet's variability would lead to a study of the internal shocks. These ingredients would contribute to build a more complete model and they might give some other implications about the jet's morphology and propagation in rotating progenitors.

6 — Conclusions

In this work, we presented the results from 3D RHD numerical simulations applied to the study of angular momentum distributions in pre-SN progenitors and the evolution of a relativistic and collimated jet piercing through the envelope in the context of lGRBs.

The main goals of this project were to construct a set of lGRB rotating progenitors were we could study the effects of different angular momentum distributions; and, the second objective was to inject a relativistic jet to follow its evolution until its break out from the progenitor were we could obtain information about the jet's Lorentz factor, the break out time and its cocoon. We used the FLASH code to perform our simulations. The numerical setup consisted in relativistic hydrodynamics in three dimensions and, to improve the computational performance and to decrease the runtime, we applied MPI paralellization and AMR.

We proposed three models with constant angular velocities, related to rigid body rotation, and three additional models with different azimuthal angular momentum distributions. From these models, we analysed the deformations due to the angular momentum (the expansion of the equatorial radius $R_{\rm eq}$ and the shrinking of the polar radius $R_{\rm pol}$) by taking the ratio $R_{\rm pol}/R_{\rm eq}$ and following the percentage change of $R_{\rm pol}$ with respect to the initial radius, $R_0 = 4 \times 10^{10}$ cm.

From this study of the progenitors, we selected two representative models that could affect the propagation of the jet. Specifically, we took the progenitor with linear angular momentum distribution (J(z)lnr-ΩM) and the one with angular momentum distribution following a cosine squared law (J(θ)cos$_2$-ΩM) with an intermediate angular velocity value. These models had different polar radius behaviour as we mention before but also, their final morphology after 10s was more physically plausible (softer edges and symmetry with respect to the xy-plane).

The jet injected in the rotating progenitors revealed some differences in its features, compared with the jet injected in the non-rotating model. First of all, we observed a decrease of the break out time in rotating progenitors. Additionally, the rotating models reached higher Lorentz factor values than the non-rotating one. The behaviour of both parameters is consistent; the higher the Lorentz factor, the lower the break out time. The discussion lead to consider that, due to the rotation settings, the density of the progenitor rearranges (the inner regions expand) and the density decreases along the polar axis, so the jet finds less resistance in its propagation to the surface of the progenitor.

From these results, we found some important implications concerning the prompt emission physics. The velocities (or Lorentz factors) and densities obtained from the simulations are consistent with linear momentum conservation. In addition, a discussion about the relation between the optical depth and the density was examined; since the density varies proportionally

to the optical depth, the latter would drop as the density does if we assume a constant opacity; so in lower density regions, we will have a low optical depth and thus, the photospheric radius $R_{\rm ph}$ will be reached.

Another important result that arises from the values obtained for the Lorentz factor is related to the fireball model during the accelerating phase (Mészáros et al., 1993b; Pirán et al., 1993 and Kobayashi et al., 1999). By writing the energy of the explosion in terms of the bulk Lorentz factor and the photospheric radius, we can find an expression for the latter. We found that the photospheric radius $R_{\rm ph} \propto \Gamma^{-1/3}$; therefore, when we get higher values for the Lorentz factor, the photospheric radius will be lower. This is a consistent implication if we consider the previous analysis concerning the optical depth.

To conclude, the most outstanding inference is the behaviour of the photospheric radius due to the variables of the jet achieved due to the rotating progenitor. This summarizes the significance of the angular momentum in the lGRBs prompt emission scenario and also this results give way to extend the study to the observable parameters beyond this phase (alterations in the light curves due to time scales and readjustments in the physical lengths).

Bibliography

[1] Band, D., Matteson, J., Ford, L., et al. 1993. ApJ, 413, 281–292.

[2] Beloborodov, A. M. 2010. MNRAS, 407(Sept.), 1033–1047.

[3] Blandford R.D., Payne D.G., 1982, MNRAS, 199, 883. doi:10.1093/mnras/199.4.883.

[4] Blandford R.D., Znajek R.L., 1977, MNRAS, 179, 433. doi:10.1093/mnras/179.3.433.

[5] Bromberg, O., Nakar, E., Piran, T., y Sari, R. 2011, ApJ, 740, 100.

[6] Cusumano, G., Mangano, V., Chincarini, G., et al. 2006. Nature, 440, 164.

[7] Daigne, F., and Mochkovitch, R. 2002. MNRAS, 336, 1271–1280.

[8] Del Zanna L., Bucciantini N., A&A 390 (3) 1177-1186 (2002).

[9] Esquivel, A., y Rodríguez, A., *Hidrodinámica de flujos reactivos*, Taller de Formación Estelar, ICN-UNAM, 2016.

[10] Flash Center for Computational Science, *FLASH User's Guide, Version 4.3*, 2015.

[11] Fruchter, A. S., Levan, A. J., Strolger, L., et al. 2006. Nature, 441(May), 463–468.

[12] Fryer, C. L., and Woosley, S. E. 1998. ApJ, 502, L9–L12.

[13] Fryxell, B., Olson, K., Ricker, P., Timmes, F. X., Zingale, M., Lamb, D. Q., MacNeice, P., Rosner, R., Truran, J. W., Tufo, H. (2000), *FLASH: An Adaptive Mesh Hydrodynamics Code for Modeling Astrophysical Thermonuclear Flashes*. The Astrophysical Journal Supplement Series. doi:10.1086/317361.

[14] Fryxell, B., Olson, K., Ricker, P., et al. 2010, Astrophysics Source Code Library, ascl:1010.082.

[15] Geng J.-J., Zhang B., Kuiper R., 2016, ApJ, 833, 116.

[16] Hachisu I., 1986, ApJS, 62, 461.

[17] Hayakawa T., Maeda K., 2018, ApJ, 854, 43.

[18] Heger A., Langer N., Woosley S.E., 2000, ApJ, 528, 368.

[19] Hjorth, J., Sollerman, J., Møller, P., et al. 2003. Nature, 423, 847–850.

[20] Horváth, I., Balázs, L. G., Bagoly, Z., Ryde, F., and Mészáros, A. 2006. A&A, 447, 23–30.

[21] Klebesadel, R. (2012). *The discovery of the gamma-ray burst phenomenon.* Cambridge: Cambridge University Press. doi:10.1017/CBO9780511980336.002.

[22] Kobayashi, S., Piran, T., and Sari, R. 1999. ApJ, 513, 669–678.

[23] Kouveliotou, C., Meegan, C. A., Fishman, G. J., et al. 1993. ApJ, 413, L101–L104.

[24] Maeder, A., & Meynet, G. 2012, RvMP, 84, 25.

[25] Maeder A., Meynet G., 2015, IAUS, 307, 9.

[26] Mészáros, P., and Rees, M. J. 1993b. ApJ, 405, 278–284.

[27] Mészáros, P., and Rees, M. J. 2000b. ApJ, 530, 292–298.

[28] Mészáros, P., and Rees, M. J. 2001. Collapsar jets, bubbles, and Fe lines. ApJ, 556(July), L37–L40.

[29] Mészáros, P., Ramirez-Ruiz, E., Rees, M. J., and Zhang, B. 2002. ApJ, 578(Oct.), 812–817.

[30] Metzger, M. R., Djorgovski, S. G., Kulkarni, S. R., et al. 1997. Nature, 387(June), 878–880.

[31] Mignone A., Bodo G., 2005, MNRAS, 364, 126.

[32] Miztli supercomputer official site: *http://www.super.unam.mx/index.php/home/acerca-de*

[33] Nemiroff R.J., 1994, ComAp, 17, 189.

[34] Levan A., Crowther P., de Grijs R., Langer N., Xu D., Yoon S.-C., 2016, SSRv, 202, 33.

[35] López-Cámara D., Lee W.H., Ramirez-Ruiz E., 2010, ApJ, 716, 1308.

[36] López-Cámara D., Morsony B., Begelman M., Lazzati, D. (2012). The Astrophysical Journal. 767. 10.1088/0004-637x/767/1/19.

[37] López-Cámara D., Morsony B.J., Begelman M.C., Lazzati D., 2013, ApJ, 767, 19.

[38] López-Cámara, D., Morsony, B. J., & Lazzati, D. 2014, MNRAS, 442, 2202.

[39] López-Cámara, D., Lazzati, D., & Morsony, B. J. 2016, ApJ, 826, 180.

[40] Piran, T., Shemi, A., and Narayan, R. 1993. MNRAS, 263, 861–867.

[41] Piran T., 1999, PhR, 314, 575.

[42] Rees, M. J., and Mészáros, P. 2005. ApJ, 628, 847–852.

[43] Ruffini, R., Rueda, J. A., Muccino, M., et al. 2016. ApJ, 832, 136.

[44] Ruffini, R., Rodriguez, J., Muccino, M., et al. 2018. ApJ, 859, 30.

[45] Shakura N.I., Sunyaev R.A., 1973, A&A, 500, 33.

[46] Stanek, K. Z., Matheson, T., Garnavich, P. M., et al. 2003. ApJ, 591, L17–L20.

[47] Tanvir, N. R., Fox, D. B., Levan, A. J., et al. 2009. Nature, 461, 1254–1257.

[48] Totani, T., Kawai, N., Kosugi, G., et al. 2006. PASJ, 58, 485–498.

[49] Uhm, Z. Lucas (2011).The Astrophysical Journal. 733. 10.1088/0004-637x/733/2/86.

[50] Weaver T.A., Zimmerman G.B., Woosley S.E., 1978, ApJ, 225, 1021.

[51] Woosley, S. E. 1993. ApJ, 405, 273–277.

[52] Woosley, S. E., and Bloom, J. S. 2006. ARA&A, 44, 507–556.

[53] Woosley, S.E., Heger, A. 2006, ApJ, 637, 914.

[54] Zhang, W., Woosley, S.E., MacFadyen, A. I. 2003, ApJ, 586, 356.

[55] Zhang, B. (2018). The Physics of Gamma-Ray Bursts. Cambridge: Cambridge University Press. doi:10.1017/9781139226530.

[56] Zhang, B.-B., Zhang, B., Sun, H., et al. 2018a. Nature Communications, 9, 447.

A — Config Script Example

```
REQUIRES Grid/GridBoundaryConditions
REQUIRES Driver
REQUIRES physics/Hydro/HydroMain/split/RHD
REQUIRES physics/Eos/EosMain/Gamma/RHD


PARAMETER sim_rhoAmbient            REAL     1.e-10
PARAMETER sim_pAmbient              REAL     1.e-13
PARAMETER sim_vxAmbient             REAL     0.0
PARAMETER sim_vyAmbient             REAL     0.0
PARAMETER sim_vzAmbient             REAL     0.0


PARAMETER theta                     REAL     5.0
PARAMETER x_o                       REAL     -3.0
PARAMETER y_o                       REAL     -3.0
PARAMETER v_ro                      REAL     0.0853
PARAMETER v_zo                      REAL     0.9760
```

Figure A.1: Example of Config file. In the firsts lines we get the paths about the physical information required for our simulation. Next, we set some important parameters and their values.

B — flash.par Script Example

```
## ISM values (in cgs)

sim_rhoAmbient  = 1.e-10
sim_pAmbient    = 1.e-13
sim_vxAmbient   = 0.0
sim_vyAmbient   = 0.0
sim_vzAmbient   = 0.0

## Jet values (in cgs and degrees)

sim_rhoJet      = 0.1039
sim_pJet        = 2.0535
z_m             = 2.0
theta           = 5.0
x_o             = -0.62
y_o             = -0.62
v_ro            = 0.08539
v_zo            = 0.97606

# Size of computational volume
xmin = -20.0
xmax = 20.0
ymin = -20.0
ymax = 20.0
zmin = 0.0
zmax = 60.0

#               Gas ratio of specific heats

gamma           = 1.333333333
```

Figure B.1: First section of flash.par file. Here, we configure some physical parameters (density, pressure, velocities for the jet an the ISM) and we establish the computational domain.

```
#  Grid dimensionality and geometry

geometry = "cartesian"

#  Boundary conditions

xl_boundary_type = "outflow"
xr_boundary_type = "outflow"

yl_boundary_type = "outflow"
yr_boundary_type = "outflow"

zl_boundary_type = "user"
zr_boundary_type = "outflow"


cfl             = 0.02

#       checkpoint file output parameters
checkpointFileIntervalTime =  10
#checkpointFileIntervalTime = 0.01
checkpointFileIntervalStep = 0
checkpointFileNumber = 0

#       plotfile output parameters
plotfileIntervalTime = 2.564843216
plotfileIntervalStep = 0
plotfileNumber = 0

nend            = 100000
nrefs           = 200000
```

Figure B.2: Second section of the flash.par file. We set the type of geometry applied in the simulation, the boundary conditions and the number of output files obtained from the simulation.

```
tmax = 512.96864

plot_var_1 = "dens"
plot_var_2 = "pres"
plot_var_3 = "velx"
plot_var_4 = "vely"
plot_var_5 = "velz"

#  Adaptive Grid refinement parameters

nblockx         = 1
nblocky         = 1
nblockz         = 1

lrefine_max  = 6
lrefine_min  = 1
refine_var_1 = "dens"
refine_var_2 = "pres"
refine_var_3 = "velx"
refine_var_4 = "vely"
refine_var_5 = "velz"


use_avisc       = .false.
cvisc           = 0.1

#    RIEMANN SOLVERS:
RiemannSolver   = "HLLC"
entropy         = .false.
```

Figure B.3: Third section of the flash.par file. We establish the maximum integration time, the variables saved in each output file, the type of Riemann solver, artificial viscosity and the AMR configuration (maximum and minimum limits and the refined variables).

C — Simulation_data.f90 Script Example

```
module Simulation_data

  implicit none

#include "constants.h"

  integer, save :: sim_meshMe
  real, save :: sim_rhoAmbient, sim_pAmbient, sim_vxAmbient, sim_vyAmbient
  real, save :: sim_vzAmbient, sim_rhoJet, sim_pJet, theta, x_o, y_o, v_ro, v_zo
  logical, save :: sim_gCell


end module Simulation_data
```

Figure C.1: Example of Simulation_data.f90 script. Here is necessary to write all the parameters involved in the initial condition.

D — Simulation_init.f90 Script Example

```
subroutine Simulation_init()

  use Simulation_data
  use Driver_interface, ONLY : Driver_getMype, Driver_abortFlash
  use RuntimeParameters_interface, ONLY : RuntimeParameters_get
  use Logfile_interface,           ONLY : Logfile_stamp

  implicit none

#include "constants.h"
#include "Flash.h"

  call Driver_getMype(MESH_COMM, sim_meshMe)

  call RuntimeParameters_get('sim_rhoAmbient', sim_rhoAmbient)
  call RuntimeParameters_get('sim_pAmbient', sim_pAmbient)
  call RuntimeParameters_get('sim_vxAmbient', sim_vxAmbient)
  call RuntimeParameters_get('sim_vyAmbient', sim_vyAmbient)
  call RuntimeParameters_get('sim_vzAmbient', sim_vzAmbient)
  call RuntimeParameters_get('sim_rhoJet', sim_rhoJet)
  call RuntimeParameters_get('sim_pJet', sim_pJet)
  call RuntimeParameters_get('theta', theta)
  call RuntimeParameters_get('x_o', x_o)
  call RuntimeParameters_get('y_o', y_o)
  call RuntimeParameters_get('v_ro', v_ro)
  call RuntimeParameters_get('v_zo', v_zo)


  sim_gCell = .true.

end subroutine Simulation_init
```

Figure D.1: Example of Simulation_init.f90 script. As we mention in Chapter 2, one of this script features is to change the parameters name if necessary. Also, it initializes the given data.

E — Simulation_initBlock.f90 Script Example

```
            do kk = 0, (sim_nSubZones-1)*K3D
               zz    = zCoord(k) + (kk*sim_inSubzml-.5)*dzz
               zDist = (zz - sim_zCenter) * K3D

               do jj = 0, (sim_nSubZones-1)*K2D
                  yy    = yCoord(j) + (jj*sim_inSubzml-.5)*dyy
                  yDist = (yy - sim_yCenter) * K2D

                  do ii = 0, (sim_nSubZones-1)
                     xx    = xCoord(i) + (ii*sim_inSubzml-.5)*dxx
                     xDist = xx - sim_xCenter

                     dist    = sqrt( xDist**2 + yDist**2 + (zDist+z_m)**2 )
                     distp   = sqrt( xDist**2 + yDist**2 )
                       rho = sim_rhoAmbient
                       p   = sim_pAmbient
                       vx  = sim_vxAmbient
                       vy  = sim_vyAmbient
                       vz  = sim_vzAmbient


                     if ( dist .le. 40.768541817556633 ) then

                       ind=1
                       do while (r_s(ind) .lt. dist*1.e9)
                          ind=ind+1
                       enddo

                       rho = rho_s(ind-1)*(1-(dist*1.e9-r_s(ind-1))/(r_s(ind)-r_s(ind-1)))+rho_s(ind)*(1-
(r_s(ind)-dist*1.e9)/(r_s(ind)-r_s(ind-1)))*(1+0.001*rand(seed))*(1-0.001*(distp/dist))
                       p = pres_s(ind-1)*(1-(dist*1.e9-r_s(ind-1))/(r_s(ind)-r_s(ind-1)))+pres_s(ind)*(1-
(r_s(ind)-dist*1.e9)/(r_s(ind)-r_s(ind-1)))*(1+0.001*rand(seed))*(1-0.001*(distp/dist))
                       p = p/9.e20

                       vx  = -(3.12e-3)*yDist*(1-(zDist/dist)**2)
                       vy  = (3.12e-3)*xDist*(1-(zDist/dist)**2)
                       vz  = 0.0


                     endif

                     if ( dist .ge. 40.768541817556633 ) then
                       rho  = sim_rhoambient
                       p = sim_pambient
                       vx  = sim_vxAmbient
                       vy  = sim_vyAmbient
                       vz  = sim_vzAmbient
                     end if

                  enddo
               enddo
            enddo
```

Figure E.1: A section of Simulation_initBlock.f90 script. The purpose of this file is to set the initial condition all over the computational domain.

F — Grid_bcApplyToRegionSpecialized.f90 Script Example

```
          if(face==LOW) then ! apply BC on the bottom boundary at z=zmin
             select case (bcType)
             case(USER_DEFINED)
                ! User BC at z=zmin
                t = 2*guard+1
                if (gridDataStruct==CENTER) then
                   do i = 1,guard
                      do j= 1,je
                         do k= 1,ke

                         if ( dr_simTime .gt. dr_tmax/2.0 ) then

                         radius = sqrt(((xCoord(j)-x_o)**2)+((yCoord(k)-y_o)**2))
                         r_j = z_m*tan((theta*3.1416)/180.)
                         frec = (2.*3.1416)/0.256   ! la variabilidad se da cada 0.01 s

                          if ( (radius .le. r_j).and.(radius .gt. 1e-10) ) then


                            if (ivar == DENS_VAR) regionData(i,j,k,ivar) = sim_rhoJet
                            if (ivar == VELX_VAR) regionData(i,j,k,ivar) = v_ro*(radius/r_j)*((xCoo
rd(j)-x_o)/radius)*(1+2.0*rand(seed)*cos(frec*dr_simTime))
                            if (ivar == VELY_VAR) regionData(i,j,k,ivar) = v_ro*(radius/r_j)*((yCoo
rd(k)-y_o)/radius)*(1+2.0*rand()*cos(frec*dr_simTime))
                            if (ivar == VELZ_VAR) regionData(i,j,k,ivar) = v_zo
                            if (ivar == PRES_VAR) regionData(i,j,k,ivar) = sim_pJet

                          else ! Apply reflecting BC
                            sign = 1.0
                            if (ivar == VELZ_VAR) then
                               sign = -1.0
                            endif
                            regionData(i,j,k,ivar)= regionData(t-i,j,k,ivar)*sign
                          endif

                         else

                          sign = 1.0
                          if (ivar == VELZ_VAR) then
                             sign = -1.0
                          endif
                            regionData(i,j,k,ivar)=regionData(t-i,j,k,ivar)*sign

                         endif
                         end do
                      end do
                   end do
```

Figure F.1: A section of Simulation_initBlock.f90 script. The user boundary conditions for the simulation are specified in this file. For our setup, we set the jet's injection for some cells in xy-plane (z=min), otherwise, we applied reflective conditions.

G — Grid_markRefineDerefine.f90 Script Example

```
do l = 1,gr_numRefineVars
   iref = gr_refine_var(l)
   ref_cut = gr_refine_cutoff(l)
   deref_cut = gr_derefine_cutoff(l)
   ref_filter = gr_refine_filter(l)
   call gr_markRefineDerefine(iref,ref_cut,deref_cut,ref_filter)
end do

#ifdef SPECIAL_REFINEMENT
!! Call for the specialized refinement
specsSize=7
!! Coordinate information ----------------------------------
!! define a range of coordinates of the rectangle in x-direction

specs(1) =  -5.0
specs(2) =  5.0

!! define a range of coordinates of the rectangle in y-direction
specs(3) =  -5.0
specs(4) =  5.0

!! define a range of coordinates of the rectangle in z-direction
specs(5) =  0.0
specs(6) =  45.0
!! End of coordinate information ------------------------------

!! Decide wheather or not we refine only blocks completely
specs(7) = 1.0

!! Bring all qualifying blocks to this level of refinement
lref = lrefine_max

call Grid_markRefineSpecialized (RECTANGLE,specsSize,specs,lref)
#endif
```

Figure G.1: Example of Simulation_initBlock.f90 script. To establish a fixed AMR along the jet axis, we specified the ranges in each coordinate for the refinement to be applied. In this case, the volume for the refinement is parallelepiped since we were working with Cartesian coordinates.

H — Miztli parallelization Script Example

```
#!/bin/bash
#BSUB -q q_hpc
#BSUB -oo jet.out
#BSUB -e jet.err
#BSUB -n 16
mpirun ./flash4
```

Figure H.1: Example of the parallelization script. From top to bottom lines, we need to specify the queue where we have the permissions to run the code, a script where the outcome information will be written (about the MPI), another script where we can found the runtime errors, the number of processors occupied and the executable name.