UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

Various Topics in Type Ia Supernovae

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

MALIA T. JENKS
Norman, Oklahoma
2019

Various Topics in Type Ia Supernovae

A DISSERTATION APPROVED FOR THE
HOMER L. DODGE DEPARTMENT OF PHYSICS AND
ASTRONOMY

BY

Dr. Edward Baron, Chair

Dr. Steven Cavallo

Dr. Chung Kao

Dr. Xinyu Dai

Dr. John Wisniewski

Contents

List of Tables

List of Figures

Abstract

This work takes two approaches to increasing our understanding of type Ia supernovae. The first part uses observations to study their spectral evolution and diversity. The second part uses modeling to investigate a potential progenitor. We cannot yet explain all of the observed diversity of these objects that are and will continue to be important tools for understanding the universe. Our understanding of SN Ia is limited by not understanding the progenitor system or systems that produce the explosions and by an incomplete understanding of the physics of the explosion itself.

Through the use of 197 spectro-photometric time series spectra from the Nearby Supernova factory it is possible to measure the time evolution of spectral features in these objects. This unique data set allows us to look for new behaviors. We find two narrow wavelength regions that for some of the sample show a near constant flux over a period of several days. About half of the sample show this plateau behavior in the wavelength region from 6125 Å to 6200 Å around 20 days after maximum light, the rest of the sample show either no plateau or a slow linear decline in flux over the same time period. The groups that show either a plateau or linear decline can be separated from the objects not showing a plateau like behavior. The presence of this plateau feature are associated with a stronger secondary maximum in the I and R bands and higher absolute magnitudes. The

other region found to show this plateau behavior occurs in the wavelength region between 6800 Å and 7000 Å about 20 days after maximum light. This plateau is a feature of most of the supernova in the sample. Using a set of existing PHOENIX synthetic spectral time series, we looked for the same behavior found in the observational data. From this model data we argue that the plateaus found in this work are likely related to the ionization behavior of Fe II/Fe III over time.

One proposed progenitor path of SNe Ia is non-violently merging double white dwarf systems. There is some indication that the mass ratio of the merging system may result in different outcomes. This work takes the quasi-hydrostatic state output of a 3D spherical particle hydrodynamic (SPH) simulation of three different merging white dwarf systems and models the initial viscous evolution with FLASH, a hydrodynamic simulation code. The three merging systems start with varied mass ratios from 0.55 to 1. The FLASH code was run in 2.5D with a simplified magnetic field to supply viscosity for about 100 seconds with a grid spacing on the order of 10 km. This fine grid allows us to look for potential carbon ignition points in the relaxing merged system. We find that none of these mergers blow a substantial wind or show significant mass loss. We find heating on the interface between the lower mass outer region and the higher mass central star, as well as along the rotation axis in the central region. None of the heated regions reached carbon ignition conditions in the simulated time.

Chapter 1

Background

Type Ia supernovae (SNe Ia) have been instrumental in furthering our understanding of the universe. They are among the brightest transient objects in the observable universe. Additionally they are relatively homogeneous, and some of the variation can be normalized. This makes them useful as "standardizable candles", allowing them to be used to measure the cosmological constant and show that the expansion of the universe is accelerating (Riess et al., 1998; Perlmutter et al., 1999).

SNe Ia are important neucleosynthisis sites in the universe (Maoz et al., 2010). These explosions produce most of the iron group element enrichment to their host galaxies. The explosions also produce intermediate mass elements. The amount of nickel produced in the explosion is related to the intrinsic luminosity of the SN Ia.

Type Ia supernova have been studied for over 40 years, however there is still much that is not well understood about them. The type of system or systems that produce these explosions, their progenitors, is still debated. There is still unexplained diversity in the observed spectra of SNe Ia.

1.1 SN Ia Diversity and Standardization

The Phillips relation (Phillips, 1993; Phillips et al., 1999; Goldhaber et al., 2001) is a relationship between the maximum luminosity in the B band and the decline in magnitude over 15 days after the maximum. This relationship between the brightness and the rate of decline is generally understood to come from variation in the amount of radioactive nickel produced in the explosion. More radioactive nickel means more heating, causing a higher temperature. This causes variations in the opacity of the ejecta and results in different diffusion times. SN Ia with brighter B band maximum decline slower than SN Ia with dimmer B band maxima. The amount of nickel produced in the explosion is correlated with the maximum luminosity of the SN Ia. The correlation between the brightness (nickel mass) and the diffusion time leads to the Phillips relation (Khokhlov et al., 1993; Nugent et al., 1995; Kasen and Woosley, 2007). This light curve shape relation allows to use SNe Ia as standardizable candles. It does not however explain all of the observed diversity in SNe Ia.

SNe Ia also show diversity in their colors (Wang et al., 2005b; Cartier et al., 2011). This diversity has been correlated to the peak magnitude as well. There have been attempts to further standardize SNe Ia by use of their colors (Tripp, 1998; Tripp and Branch, 1999; Astier et al., 2006; Jha et al., 2007). The nature of this correlation is still debated (Kasen and Woosley, 2007; Folatelli et al., 2010). A small fraction of this diversity can be explained by dust, but most of it needs to be explained by other means. The diversity may come from supernova physics,

or the physics of the progenitor and its environment at the time of explosion, or a combination of these.

SALT2 (Guy et al., 2007) is a SN Ia standardization fitting made by using spectral and light curve templates for SN Ia. The primary components found in this fitting are the date of maximum light (x0), the stretch (x1), and color variation (c). The x1 parameter can be converted to match up to the Δ_{m15} from the Phillips relation. The color variation is given by $c = (B-V)_{max} - < B-V >$. The x0 parameter gives sets the 0 value for the phase of the SN Ia, and can be used instead of explosion time. The use of this standardization method removes much of the variability but not all of it.

SNe Ia brightness has also been found to be correlated to properties of its host galaxy (Sullivan et al., 2010; Childress et al., 2013a,b; Rigault et al., 2015). This may be related to the star formation history of the host galaxy, but it is still unknown how these host galaxy properties interact with the environment of the supernova to cause this diversity.

Branch et al. (Branch et al., 2005, 2006, 2007, 2008, 2009) captured some of the spectral diversity of SNe Ia by plotting the pseudo equivalent widths of the Si II λ6355 and λ5970 features against each other (for an updated version of this figure see, for example, Blondin et al., 2012; Baron, 2014). This diagram, shown in Panel (a) of Figure 2.5, referred to as a Branch diagram has been used to group SNe Ia into four classes: core normals (CN), cools (CL), shallow silicon (SS), and broad line (BL). Benetti et al. (2005) used a different method to group SNe Ia in to 3 classes that overlap with the Branch classifications. These classes

are Faint (overlapping with CL), High Velocity Gradient or "HVG" (overlapping with BL), and Low Velocity Gradient or "LVG" (overlapping with CN and SS).

Additionally other peculiar sub-classes of SNe Ia have been discovered. *2000cx* are rare, photometrically-peculiar events that do not follow the Phillips relation, showing a rise time typical of a SN Ia, but with an unusually slower decline and high photospheric temperature (Li et al., 2001; Candia et al., 2003; Silverman et al., 2013). *2001ay* is a BL-HVG event with an extremely slow decline rate but with an apparently modest ^{56}Ni yield of 0.6 solar masses (Krisciunas et al., 2011; Baron et al., 2012). *2002cx/Iax* are non-rare events that are spectroscopicly similar to normal SNe Ia, but have lower maximum-light velocities, low luminosities for their decline rates, yet generally hotter photospheres (Li et al., 2003; Phillips et al., 2007; Foley et al., 2013). *2002ic* are SNe Ia-like events with a strong CSM interaction (Hamuy et al., 2003; Chugai et al., 2004; Deng et al., 2004; Kotak et al., 2004; Wang et al., 2004; Benetti et al., 2006; Han and Podsiadlowski, 2006; Dilday et al., 2012). *2006bt* are SNe Ia with broad light curves like a hot, luminous event and lacking a prominent secondary maximum in the near-IR, but displaying spectra at maximum similar to those of low-luminosity SNe Ia (Foley et al., 2010; Maguire et al., 2011). Moreover, several SNe Ia (2003fg, 2006gz, 2007if, 2009dc) have been observed whose brightness and light curve shape have led them to be classified as super-Chandrasekhar explosions (Howell et al., 2006; Jeffery et al., 2006; Hicken et al., 2007; Ofek et al., 2007; Scalzo et al., 2010; Tanaka et al., 2010; Silverman et al., 2011). A super-Chandrasekhar explosion may be due to a double degenerate progenitor where the mass of the binary exceeds a Chandrasekhar

mass, or possibly due to supermassive white dwarfs due to rotational support (Yoon and Langer, 2004, 2005).

1.2 SNe Ia Progenitors

The nature of the progenitor system(s) of type Ia supernovae remain largely unknown (Hillebrandt et al., 2013). We believe that SNe Ia result from thermonuclear runaway of a white dwarf that is at or near the Chandrasekhar limit. For this to occur the white dwarf needs to gain mass, so it should be in a binary or higher order system. Proposed systems can be generally divided into two types; single degenerate, and double degenerate.

The first of these is known as the single degenerate scenario (Whelan and Iben, 1973). In this scenario the white dwarf gains mass from a non-degenerate companion. Most commonly through an accretion disk from the Roche lobe overflow from the companion: a red giant or a main sequence star.

The other type is known as the double degenerate scenario (Iben and Tutukov, 1984; Webbink, 1984). In this scenario the mass is gained through interaction with an other white dwarf. This can be a dynamic merger (Pakmor et al., 2012) of the white dwarfs or it can involve an accretion disk from the tidal disruption of the less massive white dwarf. This tidal disruption is possible because of the inverse mass-radius relationship of white dwarfs.

There are some other proposed progenitor scenarios that differ slightly from these two main types. One is similar to the single degenerate cases however if the white dwarf has a strong magnetic field it can accrete material onto the poles

by carrying it along field lines rather than through an accretion disk (Wheeler, 2012). Another that starts with the same type of system as the single degenerate case but then enters a common envelope phase and the white dwarf becomes the the core of the now combined star. Then the pressure of the outer layers causes the white dwarf to reach carbon burning conditions. The white dwarf then explodes inside the star leaving no remaining companion. This scenario is known as "core-degenerate" (Soker, 2013). Another proposed progenitor scenario for SNe Ia is collision of white dwarfs (Raskin et al., 2009; Rosswog et al., 2009; Hawley et al., 2012; Kushnir et al., 2013). In this scenario two white dwarfs collide nearly head on rather than spiral inward as is the usual double degenerate case. This can occur in very dense regions of stars or because of interactions with a third body in a system containing two white dwarfs. The shocks from the collision trigger the explosion so the total mass of the two white dwarfs can vary. This explosion would be asymmetrical.

The source of Type Ia supernova could be from any of these scenarios or a mix of them. With the current state of uncertainty as to the nature of the progenitor system(s) the understanding of the explosion, stellar evolution, and the sources of diversity are impeded. Also without understanding the progenitor system and its evolution it is impossible to correct systematics for use in precision cosmology.

1.2.1 Observational Constraints

Observations of SNe Ia have provided some constraints on the progenitor systems. However, these constraints have not led to a consensus yet. Some of these

constraints come from considering delay time distributions (Ruiter et al., 2009; Maoz et al., 2010; Maoz and Mannucci, 2012), looking at pre-explosion images (Liu et al., 2012; Nielsen et al., 2014), and looking for specific indicators of each scenario.

The delay time distribution considers the time to produce the supernova, from the time of formation of the progenitor system to the explosion. Each progenitor type has different associated delay times that can be found from the expected range of lifetimes for that type of system. Observed delay times come from looking at the properties of host galaxies for a large number of SN Ia to determine the likely age of the progenitor system. The single degenerate scenario has a difficult time producing some of the observed supernovae with very large delay times because the companion star to the white dwarf cannot have evolved into a degenerate star, and must be able to donate sufficient mass to the white dwarf. Longer delay times would indicate smaller companion stars. However the total number of supernovae predicted by either scenario or even the sum of both is still too few to match observed rates (Maoz and Mannucci, 2012).

It has been proposed that if the progenitor system of a nearby SN Ia is single degenerate with a red giant companion that the companion star might be observable in images taken before the explosion. So far these companions have not been found (Li et al., 2011). However, many of the upper limits on the pre-explosion luminosity of the system do not rule out all non-degenerate companions. These observations are sufficient to indicate that at least some SN Ia come from the double degenerate scenario or single degenerates with main

sequence companions.

The single degenerate scenario has a companion star that will be impacted by the explosion. This leads to the question of what effects the interaction of the expanding ejecta colliding with the companion will cause and if they can be observed. One prediction is a large UV flash a few days after explosion (Kasen, 2010). This flash has not been observed, but there is still debate about the size of this effect (Maeda et al., 2014).

The single degenerate scenario also involves accretion of hydrogen on to the white dwarf. The accretion must be at a rate that leads to steady burning of this material. If it builds up on the surface before burning it will lead to a nova; this doesn't increase the mass of the white dwarf (Livio and Truran, 1992). This steady burning on the surface should produce a soft X-ray source (Rappaport et al., 1992). We would expect to be able to see this in pre-explosion images and it has not been observed (Liu et al., 2012). If the companion star is a helium star the accreted material may be helium rather than hydrogen (Ruiter et al., 2014).

The single degenerate scenario leaves behind the companion star. It might be possible to find this star after the explosion for SNe Ia that are both recent and close enough (Ruiz-Lapuente, 1997). The surviving companion may show altered chemical abundances, higher surface temperatures and have large peculiar velocities.These stars are still hard to definitively identify. It may be possible to identify or rule out a surviving companion to Tycho's supernova. One candidate star Tycho G has been proposed as this surviving companion (Ruiz-Lapuente et al., 2004; González Hernández et al., 2009).

In the double degenerate case both stars are involved in the explosion so there is no companion for the ejecta to impact, unless the system was extremely nearby the white dwarfs would not be observable in pre-explosion images, and could not be found post-explosion. There are however other properties of double degenerate explosion scenarios that can be tested for in observations.

In the case of a dynamic merger the explosion may be highly non-spherical (Pakmor et al., 2012). The symmetry of the explosion can be studied with spectro-polarimetry. In looking at the spectro-polarimetry of many SNe Ia they all appear to be highly spherical (Wang et al., 1997, 2005a). This could be an indication that most SNe Ia are not coming from collisions of two inward spiraling white dwarfs, or other scenarios that would usually result in largely asymmetrical explosions.

There is still another possibility for the double degenerate case; the more massive white dwarf tidally disrupts the companion white dwarf, which forms an accretion disk. In this case the matter being accreted is primarily carbon and oxygen. This means that it will not undergo nuclear burning on the surface of the white dwarf. Thus accretion can occur at a much greater rate than in the single degenerate scenario. The explosion in this case is expected to be largely spherical.

In some observed SNe Ia it has been suggested that the explosion necessary to produce them must have come from a white dwarf with greater than the Chandrasekhar mass limit (Howell et al., 2006; Hicken et al., 2007; Scalzo et al., 2010; Tanaka et al., 2010; Silverman et al., 2011). Such an explosion would be very difficult to produce with the single degenerate scenario. In the double degenerate collision case it is possible to get a mass larger than the Chandrasekhar mass

because the explosion mass would be the total mass of both white dwarfs. In the tidal disruption case the explosion would still occur when the primary white dwarf reaches the mass limit. This mass limit can increase if the star is rotating (Yoon and Langer, 2004, 2005). The accretion could cause a spin up of the white dwarf. The greater accretion rate could cause this increase in rotation to be substantial (Piersanti et al., 2003). This may offer a way to increase the explosion mass of these systems.

In both scenarios a white dwarf gained mass. The elemental composition of the accreted matter is different in each case. In the single degenerate scenario the accreted matter is usually hydrogen, but could be helium. In the double degenerate cases the mass gained is usually predominately carbon, but could be helium if the other white dwarf is very low mass. It may be possible to detect signatures from this donated material in the outer layers of the explosion. Some early observations have shown potential high velocity carbon features (Parrent et al., 2011).

1.3 Models

Modeling is an important tool in improving our understanding of SNe Ia. This work uses two different models for that purpose `PHOENIX` (see, for example, Hauschildt and Baron, 1999), a radiative transfer code, and FLASH4.4, a multi-physics simulation code. We use existing `PHOENIX` models to investigate the physics that may cause features seen in observations. We use FLASH4.4 simulations to investigating the post merger evolution of a possible SNe Ia progenitor.

PHOENIX can be run in several different modes including; 1D to 3D, LTE or NLTE, light-curve mode, and time dependent mode. The model calculates the spectrum of the given hydrodynamic state. PHOENIX works by solving the full radiative equations for the system. In this work I used a set of 2D time-dependent LTE spectra for several variations of W7 created for a different purpose (Lexen, 2014).

FLASH4 is a modular multi-physics simulation code. It can be used for a wide variety of physical simulations (Calder et al., 2002). This versatility is gained through the use of many independent modules for the relevant physics. Thus when building and using the code each of the desired modules needs to be specified. These modules include the hydrodynamic solver, equation of state, magnetic field solver, and others. For some simulations it is possible to use particles, but most including those run for this work are solved on a grid. For the purposes of this study the simulations are run in cylindrical coordinates on a uniform grid.

Chapters 2 and 3 are written as stand alone papers. Chapter 2 is in preparation to be submitted for publication.

Chapter 2

Narrow Band Light Curve Plateaus

2.1 Introduction

Type Ia supernovae (SNe Ia) are thought to be the thermonuclear explosion of a C+O white dwarf whose mass increases in a binary system, by interaction with a close companion. However, the exact progenitor system is unknown, as are the details of the explosion itself. It is well understood empirically that there exists a sub-class of SNe Ia with a light curve shape luminosity relation, often called the Phillips relation (Phillips, 1993; Phillips et al., 1999; Goldhaber et al., 2001). It was through the use of these SNe Ia as standardizable candles that the dark energy was discovered (Riess et al., 1998; Perlmutter et al., 1999).

To zeroth order, the Phillips relation (Phillips, 1993; Phillips et al., 1999; Goldhaber et al., 2001) is understood as due to a variation in the total amount of radioactive nickel produced in the SN causing higher temperature, which leads to variations in the opacity, resulting in variations in the diffusion time. The correlation between the brightness (nickel mass) and the diffusion time leads to the Phillips relation (Khokhlov et al., 1993; Nugent et al., 1995; Kasen and Woosley, 2007). Yet, while the light curve shape relation allows to use SNe Ia as standard candles, it does not explain all of the observed diversity.

Other studies have made use of the color of the SNe Ia to standardize them further (Tripp, 1998; Tripp and Branch, 1999; Astier et al., 2006; Jha et al., 2007). While the nature of the color diversity correlated to peak magnitude is still debated (Kasen and Woosley, 2007; Folatelli et al., 2010), it appears more and more clear that a significant fraction of this behavior is not explained by dust extinction, and has to be accounted for by the supernova physics and/or the physics of the progenitor environment.

The state of the art of SNe Ia standardization manages to standardize them up to a final remaining dispersion in the Hubble diagram of $\sim 14\%$ (Scolnic et al., 2018). In the constant struggle to better understand SNe Ia, either to improve their use as cosmological candles or for their own sake, significant progress has been made. In particular, a correlation between SNe Ia brightness and their host properties, probably related to its star formation history, has been unveiled (Sullivan et al., 2010; Childress et al., 2013a,b; Rigault et al., 2015). While a link between the environment of the supernova and its brightness seems very plausible, the mechanism through which they interact remains unknown.

In addition, observations carried out since the 1980's have increasingly revealed a widespread diversity in SNe Ia spectra and light curves requiring a whole new understanding of the field. This spectral diversity is partially captured in the work of Branch et al. (Branch et al., 2005, 2006, 2007, 2008, 2009) who plotted pseudo equivalent widths of the Si II λ6355 and 5970 features against each other (for an updated version of this figure see, for example, Blondin et al., 2012; Baron, 2014) Branch et al. used this diagram to group SNe Ia into four classes: core normals

(CN), cools (CL), shallow silicon (SS), and broad line (BL). Using a different approach Benetti et al. (2005) arrived at similar classes: Faint (overlapping with CL), High Velocity Gradient or "HVG" (overlapping with BL), and Low Velocity Gradient or "LVG" (overlapping with CN and SS). Some of this variation has been ascribed to asymmetrical explosions (Maeda et al., 2010; Maund et al., 2010), and asymmetric distributions of both iron group elements (IGE, including radioactive nickel) as well as of intermediate mass elements, (IME), are possible. However, this seems unlikely to be the whole explanation since the 21st Century has seen the discovery of an uncomfortably large number of peculiar sub-classes of SNe Ia identified by their prototypes.

The Nearby Supernova factory has gathered SNe Ia spectral time series over more than a decade using an Integral Field Unit technology that produces a photometrically calibrated sample and efficient host galaxy subtraction. We take advantage of the unique character of this dataset to probe SNe Ia diversity through the direct analysis of the best quality subset of the data.

Our understanding of SNe Ia at this point is an unfinished puzzle. We use the Nearby Supernova factory (SNf) dataset in order to add a new piece. We began this project with the creation of spectral evolution movies, using a spline fit to interpolate between epochs. Since the SNf dataset is flux calibrated, the SNe Ia spectra flux scales epoch by epoch in a meaningful way, making spectral evolution movies a natural way to look at SNe Ia behavior. This proved particularly useful for the subset of SNe Ia with one spectrum every 3 days or more. Those movies almost immediately brought to light previously unnoticed behavior of SNe Ia: the

presence of plateaus at ~ 6200 and ~ 6900 Å lasting more than 10 days.

The purpose of this paper is the presentation and analysis of those plateaus, defined as the existence of a stable flux in time over a duration of more than 10 days in wavelength-integrated regions of ~ 100 Å. In the first section of the paper we present the dataset we consider. In the second section, we describe precisely how we define a narrow light curve and a plateau, and the robust technique we used to measure them after their unveiling by our visual inspection of the aforementioned movies. In a third section we detail the behavior of the two long duration plateau we restrict ourselves to, and follow with a discussion of how they correlate with the usual SNe Ia population descriptors. In the fourth section we present a first tentative physical explanation of these plateaus, and their potential in probing SNe Ia models. Based on the physical motivations described in the fifth section, we then discuss the feasibility of the plateau as some sort of standard in SNe Ia and finally we conclude. Technical details are presented in A.

2.2 Nearby Supernova factory spectral time series

2.2.1 The Nearby Supernova factory

The data were obtained using the SuperNova Integral Field Spectrograph (SNIFS, Lantz et al., 2004). SNIFS is a fully integrated instrument optimized for automated observation of point sources on a structured background over the full ground-based optical window at moderate spectral resolution. It consists of a high-throughput wide-band pure-lenslet integral field spectrograph (IFS Bacon et al., 1995, 2001),

a multi-filter photometric channel to image the stars in the vicinity of the IFS field-of-view (FOV) to monitor atmospheric transmission during spectroscopic exposures, and an acquisition guiding channel. The IFS possesses a fully-filled 6.4" × 6.4" spectroscopic field of view subdivided into a grid of 15 × 15 spatial elements, a dual channel spectrograph covering 3200—5200 Å and 5100—10000 Å simultaneously with FWHM resolutions of 5.65 Å and 7.54 Å respectively, and an internal calibration unit (continuum and arc lamps). SNIFS is continuously mounted on the south bent Cassegrain port of the University of Hawaii 2.2 m telescope (UH 2.2m) on Mauna Kea. The telescope and instrument, under script control, are supervised remotely.

2.2.2 Data selection: 197 SNe Ia time series

Our first visual approach using movies meant that we started by using the highest quality SNe Ia available, in terms of signal to noise and phase coverage. Once we found that some SNe Ia displayed features with a stable flux over periods longer than 10 days, we turned to the whole dataset. Figure 2.1 shows the spectral time evolution of the well covered SN PTF10icb.

In order to compare the time evolution of SNe Ia, we need a meaningful phase determination. We choose SALT2 fits of synthetic photometry obtained by numerical integration of the observer frame spectra available in theoretical top hat filters designed to avoid the instrumental split between the spectrograph blue and red arms. We kept only the SNe Ia which met the following criteria:

- At least 5 epochs for each SN Ia among which:

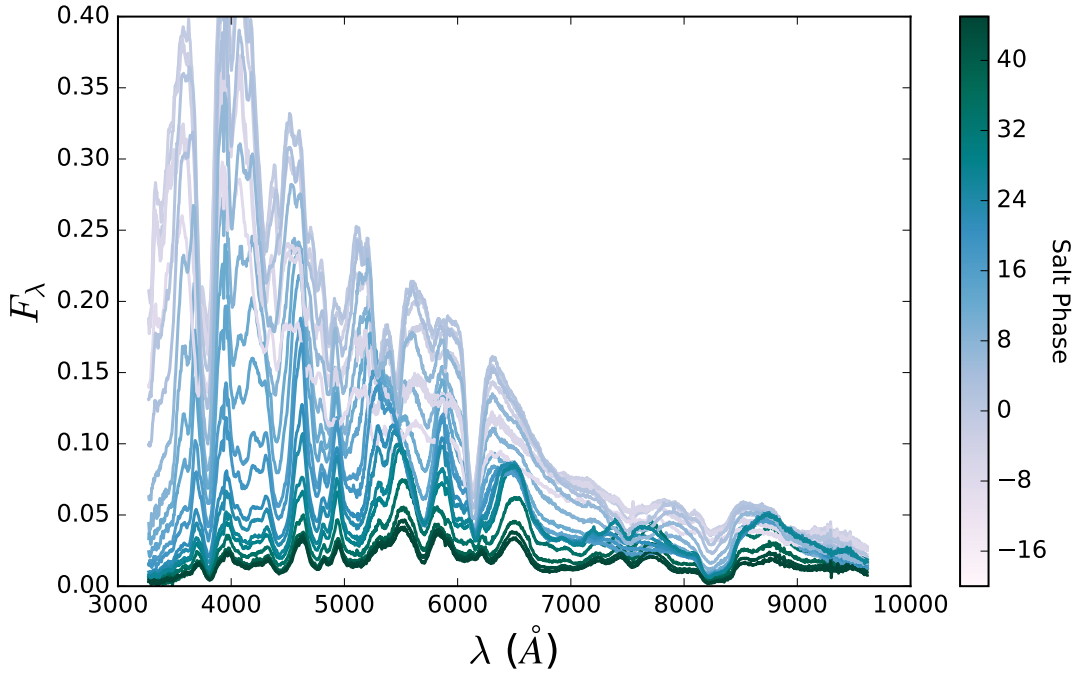


Figure 2.1 The time series of the spectral evolution of SN PTF10icb, representative of the phenomena seen in our movies.

  - 4 epochs at least between phase -10 and phase +35
  - at least one epoch between phase -10 and phase +7
  - at least one epoch between phase +7 and phase +20 days
- Less than 20% of outliers to the SALT2 fit at the 0.2 magnitude level
- Normal Median Absolute Deviation (nMAD) of the SALT2 fit residuals lower than 0.12

This criterion results in a sample of 197 SNe Ia from a total sample of 223 SNe.

We correct each spectrum from Milky Way extinction using the Schlegel et al. (1998) dust map and the Cardelli et al. (1989, p. CCM) extinction law.

2.3 Plateaus: Stable flux regions in notoriously evolving objects

2.3.1 First findings from visual inspection

SNe Ia are notoriously variable events, with broad spectral lines caused by elements expanding away from the explosion center at velocities of the order of 10000 km s^{-1}, and evolving noticeably with a characteristic timescale of 1–3 days. The presence of spectral regions staying stable over more than 10 days is striking, and could be windows to physical regions in the SNe Ia ejecta, where the line formation is simpler, and hopefully easier to comprehend.

Our visual inspection of the spectral evolution of the best sampled SNe Ia of the SNfactory dataset unveiled first a plateau redwards of the Si II trough at $\sim$ 6100 Å. This plateau happens more than 10 days after maximum light. At that time, the Si II contribution to the feature is becoming weak, and Fe II and Fe III lines dominate instead.

This, together with the Fe II/Fe III single element spectra presented in Bongard et al. (2008) that will be discussed below in more detail, helped us to notice the existence of a second plateau in the 6800 - 7000 Å wavelength region, where the same lines dominate.

The work of Bongard et al. (2008) discusses numerous other Fe II/Fe III regions, but we defer their examination to future work. A cursory examination suggests that these regions may exhibit short plateaus; however, none comes close to being as clearly visible as the plateau of the two regions we study here.

The visual inspection of most of the available spectral evolution movies showed no sign of a plateau bluewards of 5000 Å at post maximum light phases. The line formation in the near-UV/blue part of the spectrum is complex to model and understand due to the large number of iron group element lines involved. It is possible that because of the much larger optical depth resulting from line blanketing that the physical region probed by the blue spectrum is at significantly larger velocities than the region where the redder plateaus are formed. It is also possible that the significant role played by fluorescence in iron group element lines transferring flux from the blue to the red prevents the formation of a stable flux region in the blue. While understanding the difference in behavior between near-UV/blue and the red part of the spectral energy distribution is beyond the scope of this work, it is important to study in the future. us.

The complex analysis of pre-maximum light spectral formation is also outside of the scope of this work.

2.3.2 Plateau definition and detection

While the animated movies of spectral evolution are useful tools for visual inspection of the SNf dataset, they don't provide a clear way to quantify what is seen. The plateau behavior we isolated was localized in narrow wavelength regions, making the introduction of *narrow light curves* natural. Narrow light curves are defined as the time evolution of the numerical integration of a spectral time series over a wavelength range not extending further than one supernova feature (i.e. ~ 200 Å around 6000 Å), which is much less than the average width

of any broadband filters used in photometry.

The selection of the width of the narrow light curve is in part arbitrary, or at least ill defined: line blending in SNe Ia spectral formation makes the quest for the perfect wavelength range somewhat pointless. Instead, we concentrated on finding wavelength ranges that make the effect we study clearly visible, while being robust to small boundary changes. Once this goal was reached, we left aside the natural urge towards optimization and proceeded to deepen our description and understanding of the behavior of the narrow light curves instead.

The specific behavior we decided to track was the presence, or absence, of plateaus in narrow light curves. In this first paper, we will not attempt to be exhaustive in finding all the possible narrow light curve ranges at which plateaus might happen. Instead, we selected the two most visible ones in the movies, which implies that we selected the plateaus of longest duration. Some shorter plateaus might exist, but we defer their discussion to a more extensive analysis of the spectral evolution of SNe Ia that goes beyond the scope of this paper. The two wavelength ranges that we selected based on visual localization of the plateau are $\Delta w_{P6} = (6125\text{Å}, 6200\text{Å})$ and $\Delta w_{P7} = (6800\text{Å}, 7000\text{Å})$. As mentioned above, aside from making sure that wavelength changes of the boundaries of the order of a few tens of Å didn't change our results significantly, we didn't try to optimize those ranges.

We sorted the objects in to three categories based on the behaviors of these narrow light curves. A SN with a plateau will have a nearly constant flux in the narrow band for at least a few days, an object with a tilted plateau has a nearly

linear behavior, and an object without a plateau doesn't show either of these behaviors. We discuss the details of the algorithm and cuts in A.

In the rest of this section we describe the two major plateaus that we found, and their distribution in terms of SNe Ia demographics.

2.3.3 P7: the plateau in the 6800 – 7000 Å wavelength region

This plateau is found in the first peak redwards of the Si II 6100 Å trough, i.e. in the wavelength region 6800 – 7000 Å. We find 120 SNe Ia with a plateau, 42 with a tilted plateau, and 16 with no plateau for at total sample of 178 SNe Ia in total. The other SNe Ia of the total sample available where dropped due to insufficient sampling in the plateau region, as described in A.

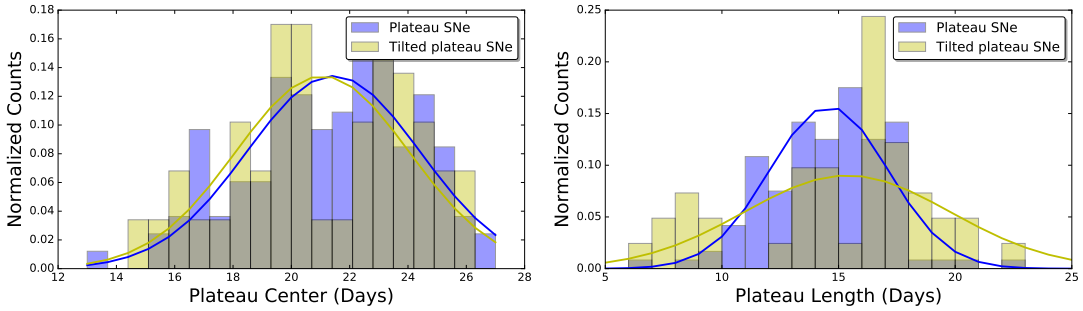
The plateau center of plateau SNe Ia is 21.4 days after maximum light with a $\sigma = 3.0$ days. The average plateau duration in this wavelength region is 14.6 days with a $\sigma = 2.5$ days.

The tilted plateau center found in this wavelength region is at phase 20.9 on average, with a $\sigma = 3.0$ days, for an average duration of 15.2 days with a $\sigma = 4.5$ days.

The behavior is illustrated in Figure 2.2.

2.3.4 P6 : the long duration plateau in the 6125-6200 Å narrow light curve

This plateau is found in the narrow light curve integrated in the wavelength region 6125 – 6200 Å. We find 90 SNe Ia with a plateau, and 50 with a tilted plateau, and



(a) P7 Centers

(b) P7 Lengths

Figure 2.2 Histograms of the plateau centers and lengths for the tilted and non-tilted 6800-7000 Å (P7) light curves.

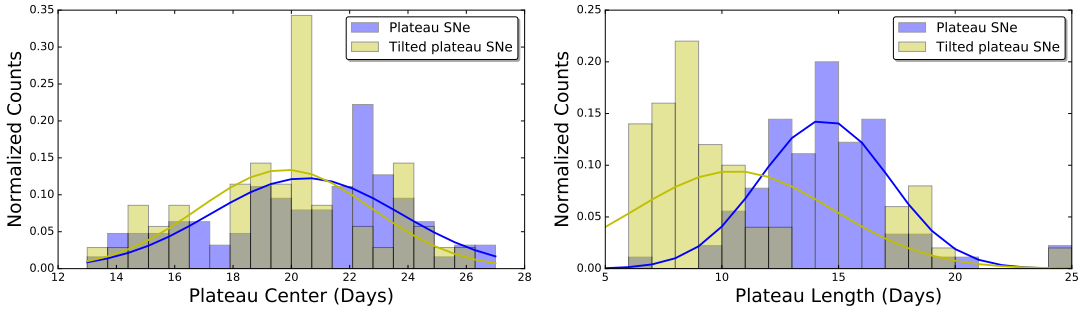
39 with no plateau for a sample of 179 SNe Ia in total. The rest of the SNe Ia of the sample have been dropped due to insufficient sampling in the plateau region, as discussed in A.

The average plateau center is at 20.5 days after maximum light, with a distribution width of 3.3 days. Note that the phase of the plateau center distribution isn't Gaussian.

The average plateau length is 14.4 days, and seems to be normally distributed with a $\sigma_{\text{length}} = 2.8$ days.

The tilted plateaus on the other hand have a much shorter average length of ~ 7.5 days, while still being centered on average around day 19.7 after maximum light. The center of the tilted plateau region being again distributed along a quite flat distribution of $\sigma = 3.$ days.

The behavior is illustrated in Figure 2.3.



(a) Centers (b) Lengths

Figure 2.3 Histograms of the plateau centers and lengths for the tilted and non-tilted 6125-6200 Å light curves

2.4 P6 and P7 demographics

We show in Figure 2.4 a set of pie diagrams describing the P6 and P7 plateau demographics. A little more than half of the SNe Ia sample displays a P6 plateau, and more than two thirds display a P7 plateau.

The distinction between plateau and tilted plateau is, in a large part, artificial. Both groups actually form a continuum distinguished by a cut based on the slope of the plateau. The choice of this cut is not based on physical grounds, but on signal to noise considerations: a plateau is defined as all the slopes not steep enough to be measured (see A for more details). With that in mind, if we group together the plateau and tilted plateau events, we see from the two upper pie diagrams of Figure 2.4 that P6 and even more P7 are a prominent feature of SNe Ia.

The pie diagram on the lower left of Figure 2.4 shows that the number of SNe Ia with both P6 and P7 plateau, or with both P6 and P7 tilted plateau is about one third of the total sample. While not negligible, this is significantly

less than the individual populations of P6 and P7 plateaus, indicating that the continuous evolution between plateau and tilted plateaus of P6 and P7 don't correlate strongly. The lack of strong correlation between the P6 and P7 slopes hints that they probe different physical regions of the SNe Ia ejecta. On the other hand, the fact that they happen at similar phases, for similarly long periods of time, is an improbable coincidence, therefore hinting that both P6 and P7 share a common physical cause.

Both points are corroborated by the lower right pie diagram of Figure 2.4. It shows that if we consider both plateau and tilted plateau SNe Ia in the same group, the number of events with both P6 and P7 amounts to two thirds of the population, and is twice as large as the number of events with only one of the two behaviors.

The two events that are absent from both the P6 and P7 plateau/tilted plateau are SNF20071015-000 and LSQ13aiz. An examination of the spectral evolution of these objects does not indicate anything particularly remarkable, thus we defer any discussion of the objects to future work.

2.4.1 P6 and P7 and historical spectral indicators at maximum light

Figure 2.5 shows the Branch diagram of the CfA sample (upper panel), the Branch diagram of the P6 sample (lower left) and the P7 sample (lower right). The clear difference in the CfA and SNf sample that there is a lack of the CL sub-class in the SNf (and maybe a smaller deficit in the SS and BL sub-classes) likely due, at least in part to the fact that CfA was a targeted search, whereas SNf is blind as

well as that the SNf followed up core normals for cosmology.

Interestingly, for the P6 sample, the no plateau class dominates the Cool sub-class. This is consistent with the P6 plateau correlation with $x1$ (see below). On the other hand, the Broad Line sub-class, has members from all plateau populations, making it more closely related to the Core Normal group than the Cool population from the plateau perspective. The shallow silicon sub-class is dominated by plateau supernovae for both P6 and P7. Again, for P6, this is consistent with the correlation with $x1$.

The few SNe Ia without P7 plateau, while too small in number to be able to provide definite constraints, tend to show that the P7 plateau is ubiquitous among all the Branch sub-classes. This difference between P6 and P7 is an indication that both plateaus, while very similar in behavior, are formed in different regions of the SNe Ia, with P7 being probably formed deeper, where the variation in IME has much less impact.

2.4.2 P6 and P7 versus $x1$ and c

We now turn here to the study of the plateaus demographics compared to the SALT2 color c and the SALT2 stretch factor $x1$ (Guy et al., 2007). These are well known standardization parameters that empirically describe the diversity of SNe Ia brightnesses and colors, at least for the Core Normal sub-class.

While the color parameter accounts for intrinsic SNe Ia variability, it also accounts for environmental reddening effects. On the other hand, the light curve width factor $x1$ is likely only related to intrinsic variability in SNe Ia.

Figure 2.6 shows the histograms of the different plateau categories versus the SALT2 light-curve shape parameter $x1$. It is clear that the plateau and tilted plateau supernovae, for the case of P6, are drawn from the population of slower declining supernovae. Moreover, the tilted plateau population displays an average $x1$ smaller than the plateau SNe Ia, showing that the three populations actually form a continuum along the $x1$ dimension. From a KS test, the P6 plateau and no plateau form two $x1$ distinct distributions with a q-value $q \equiv 1 - p$ smaller than 10^{-10}. P6 tilted plateau and no plateau form two distinct $x1$ populations with a q-value smaller than 10^{-5}. And P6 plateau and tilted plateau form a common $x1$ distribution with a q-value of order ~ 0.2.

The P7 plateau, on the other hand, shows no relation with $x1$. Yet, while all groups span the entire $x1$ distribution, the no plateau family might display a tighter dispersion around $x1 = 0$. Given the almost ubiquitous presence of P7 plateau in the SNe Ia of our sample, the no plateau group doesn't provide enough statistics to push this observation further.

Figure 2.7 shows the P6 and P7 slopes versus $x1$ with the expected behavior, plateaus have the smallest slopes, tilted plateaus the next smallest, and no plateaus the largest. The continuum nature of the plateau is clearly evident.

Figure 2.8 shows plateau length vs $x1$ for P6 and P7. In the P6 sample the range of both length and $x1$ is smaller for the plateaus than for the tilted plateaus and length of the tilted plateaus is smaller. There is indication for a correlation between $x1$ and plateau length. In fact, this correlation indicates that the duration times of the P6 plateaus are stretchable in the sense of the light curve shape

parameter introduced by Goldhaber et al. (2001) and further improved by Burns et al. (2014). This empirical result, lends support for the model variations that we discuss in section 2.5. In the P7 case there is no clear separation between plateaus and tilted plateaus.

Figure 2.9 shows that there is a correlation between P7, and to a lesser extent, P6 height with SALT2 color parameter, c, even though the relation between them is clearly not linear. There is a smaller non-linear correlation between the SALT2 $x1$ parameter and P6 height, while P7 shows no sign of such a correlation with $x1$. This is consistent with P6 being an intrinsic property of SNe Ia that probes mainly the IME, and thus behaves differently for the Branch Cool sub-class.

Figure 2.10 shows the plateau height and length versus the pEW Si 4000 (Folatelli, 2004). This figure reiterates what is already evident in Figure 2.8 — that for P6 there is a definite stretch-ability or correlation with $x1$, that is not evident in P7. That is, P6 and P7 are different from one another. The weak correlation that is seen between the intrinsic property pEW Si 4000 and both P7 length and height adds a potential lever arm in disentangling the intrinsic fraction of P7 height from its variability due to dust, assuming that we are correct that the primary variation of P7 height is due to dust properties.

While the simultaneous onset of the P6 and P7 plateaus indicate a common physical cause, the difference in the behavior of P6 and P7 with respect to $x1$, supports the claim that they probe different regions or physical conditions of the supernova ejecta.

We have examined the correlation of plateau length and height with other

spectroscopic indicators such as $\Re_{\rm Ca}$, $\Re_{\rm Si}$, and $\Re_{SiS}$ (Nugent et al., 1995; Bongard et al., 2006) and they only serve to reinforce two qualitative results: 1) P6 is correlated with $x1$ (since $\Re_{\rm Ca}$, $\Re_{\rm Si}$, and $\Re_{SiS}$ are correlated with $x1$), 2) P6 and P7 are qualitatively different, so the physics that they are probing is likely to be quantitatively different.

Figure 2.11 shows the histograms of P6 and P7 plateau, tilted plateau, and no plateau groups with respect to the SALT2 color parameter c. Not surprisingly, given the light curve shape results, P6 SNe Ia with plateau are bluer than no plateau supernovae. This is to be expected from the fact that slow declining supernovae are hotter (due to more nickel heating) than fast declining supernovae (Nugent et al., 1995). The plateau P7 on the other hand doesn't display a special color behavior for any of the three groups.

Figure 2.12 shows the centers versus $x1$. While P6 length showed that it followed stretch, the time of onset of the P6 plateau clearly depends on other physical causes. This again supports our assumption that P6 is probably created in the IME region, where different features dominate at different epochs depending on the details of the SNe Ia explosion, while its evolution is driven by the inner core of the supernova, which is more standard and for which the main diversity parameter is the supernova absolute brightness.

On the contrary, the P7 plateau center is clearly related to $x1$, while its height and length are not. This again supports the hypothesis that P7 is created in the deeper IGE dominated regions, whose specific features become apparent as the pseudo-photosphere recedes, with the rate of recession being directly proportional

to the supernova absolute magnitude and thus, $x1$. Moreover, the fact that P7 happens at a time dependent on the supernova absolute magnitude, $x1$, while its characteristics (height and length) appear independent of $x1$, supports the hypothesis that this almost ubiquitous property of SNe Ia is due to a powerfully standard physical behavior, like, for example, the transition between Fe II and Fe III as discussed below.

2.4.3 P6 light curve and spectra demographics

In the upper left of Figure 2.13, we display the average B light curve of the three P6 groups, using SALT2 phase to describe the time evolution. SALT2 phase is 0 at the time maximum light in the B-band. This plot clearly shows the light curve shape relation with P6 that was noted when comparing the demographics with their $x1$ distribution: P6 non-plateau SNe are fast decliners, P6 tilted plateau SNe are slower, and P6 plateau SNe are the slowest decliners.

This same behavior is seen in the V-band (upper right of Figure 2.13) which for these SNe, not far away from Branch normal, is a reasonably good stand-in for the bolometric lightcurve (see, e.g, Ganeshalingam et al. 2011, but also see Hayden et al. 2010). The V-band lightcurve peak of non-plateau SNe is 2–3 days earlier than that of SNe showing either kind of plateau.

The R and I-band lightcurves, on the lower left and lower right of Figure 2.13 show that the P6 plateau supernovae display a strong secondary maximum that is significantly delayed from that of the non plateau supernovae. We discuss this below in terms of the fluorescence of the Fe III to Fe II ionization transition.

We show in Figure 2.14 the average spectra of the three P6 groups, for different epochs. While the detailed analysis of the spectral differences and their intricate relationship with SNe Ia physics goes beyond the scope of this paper, there are still few salient points worth noting. For example, for pre-maximum light spectra, the P6 no-plateau group shows a clear difference from the tilted plateau and plateau group in the UV features. These features have been shown as one of the places where the remaining diversity of SNe Ia might be tractable.

The middle left panel of Figure 2.14 shows a clear color difference between P6 no-plateau and P6 tilted plateau and plateau SNe Ia at 15 – 16 days after maximum light, i.e. around the onset time for P6 and P7 plateaus. This color difference remains visible afterwards, until 35 – 40 days after maximum light, where the colors seem to again become similar for all groups. At that time, the P6 no-plateau SNe Ia are systematically fainter than the other ones, as expected from the fact that they are faster decliners, as discussed when the $x1$ parameter distributions were considered.

While this color difference is again out of the scope of this paper, the fact that it happens during the P6 and P7 plateau duration is noteworthy. In addition, it stresses the importance not only of SNe Ia color, but also of their color evolution, that can not be captured by only one parameter defined at maximum light.

2.4.4 P7 light curve and spectra demographics

Figure 2.15 shows the median light curves for the three P7 plateau, tilted plateau and no-plateau categories. None of the broad band light curves shown display

prominent differences between the three groups. Again, the small number of SNe in the no plateau category implies that we should exercise caution in drawing conclusions about the population variations of P7. In particular, the departure of the P7 no-plateau V and R bands (see Figure 2.15(b,c)) from the global behavior at about day +12 — +20, just at the minimum between the two maxima needs to be considered. While we note this behavior, we won't further analyze its implications here.

Figure 2.16 shows the median spectra of the P7 plateau, tilted plateau and no-plateau groups. The pre-maximum light spectra on the upper left show a strong difference between the three groups in the UV features. The tilted plateau median spectra also departs from the two groups, in particular from the plateau group. This is the only instance of a clear salient difference between P7 plateau and P7 tilted plateau behaviors. It is also noticeable that the P7 plateau SNe Ia are the only group that doesn't display pre-maximum high velocity Ca II.

Afterwards, P7 plateau and tilted plateau behavior match each other extremely well, making them indistinguishable for all practical purposes. And while the low statistics of no-plateau events again calls for caution, we do see a color difference between their color and that of the plateau/tilted plateau group in the 15 – 16 and 20 – 21 plots, which is the period during which both P6 and P7 plateau occur.

2.5 Discussion

2.5.1 PHOENIX Experiment

In this section we discuss the use of the radiative transfer code PHOENIX to probe the physics behind the P6 and P7 plateau behavior. Since no standard model for SNe Ia exists, detailed analysis of hydrodynamic and synthetic spectral modeling is beyond the scope of this paper. Therefore, we chose a pragmatic approach. We took advantage of a set of models that have been computed for a different purpose, and re-analyzed them in the light of this new plateau feature that we are presenting.

PHOENIX time dependent models

We used the results of a time series of LTE spectra of varied W7 explosion models (Nomoto et al., 1984), with `PHOENIX`. Using the models of Lexen (2014) where the explosion velocities were adjusted the from 0.5 to 1.5 times the original model, we treated the modeled time series in the same manner as the observations and produced narrow band light curves for each velocity. Since variations in velocity change the effective opacity, we expected that these models might address the physical variations that we seek to understand (Wang et al., 2012).

Figure 2.17 and Figure 2.18 show models with and without P6 plateaus, respectively. Figure 2.19 shows the evolution of the spectra around the wavelength region where the light curves are constructed. Clearly the Si II $\lambda\lambda$6355 feature is much stronger in the model with a plateau than in the one without, due to the

fact that the higher velocities lead to higher Doppler shifts and a washing out of the feature. While Figure 2.19 displays an extreme parameter variation, this behavior is not clearly seen in the observed spectra, leading us to be cautious that the model is indeed capturing all the underlying physics. However, since we have no first principles model of SNe Ia, this is the best we can do at present.

Additionally, strong Fe II features develop in the plateau model, while they are much weaker in the non plateau model. This difference in the spectra in the plateau region is significantly more pronounced in the models than it is in the observations. Yet it supports the hypothesis that the evolution of the Fe II/Fe III dominated pseudo-photosphere might be driving the P6 plateau behavior.

For the P7 region, the models do not produce plateaus, but rather produce light curves that are very similar to one another. While the light curves are not good representations of observations, the reasons are understandable due to the fact that the physics in the PHOENIX experiment was quite simplified. However, since the models do reproduce the qualitative structure of the observations (P6 plateaus appear and disappear, P7 light curves are quite uniform) we suggest that the underlying physics is being captured in these models. A more detailed analysis would need the extensive study of many models, which is beyond the scope of this paper.

It seems that the plateau formation in P6 is related to the distribution of iron group elements in the region where they intermingle with IME, whereas P7 is less sensitive to the distribution and is produced at lower velocities.

The question arises are the models capturing the physics that creates the

plateau, but just exaggerating it? We think that is likely to be the case, since although no model of SN Ia fully captures all of the observational complexity, the underlying physics is unlikely to be wildly different given that the models do capture the gross features of the the SN Ia phenomenon.

A plausible explanation of the plateau behavior: Fe II / Fe III fluorescence

Here we concentrate on the causes that explain the presence or absence of P6 plateaus in our PHOENIX simulations.

Examining the ionization structure of the models we find that P6 non-plateau supernovae show a clear transition from Fe III to Fe II (and Co III to Co II) with the Fe II/Fe III transition region moving steadily in velocity (Figure 2.18) with time. Whereas for the plateau supernova there is more of a caustic behavior, where the Fe II/Fe III transition hangs up at a specific velocity and Fe II never becomes the dominant ionization stage in the core. Figure 2.17 shows that hangup in velocity occurs at just about the right time as the plateau. Note that the fluorescence is strongest when there is a transition between the dominant ionization stage going from III $\leftrightarrow$ II and that is evident at about the right epoch in the model with the plateau, whereas in the model without the plateau there is a quick transition to Fe II as the dominant ionization stage.

Physically in the absence of radioactive decay and gamma ray escape the picture seems backwards, the faster expanding ejecta should have lower densities, which at equal temperatures would lead to a higher Fe III/Fe II ratio. If the cores expand close to adiabatically, $T^{3/2}/\rho$ should be approximately constant. Since in

homologous expansion $\rho \propto t^{-3}$ that implies that $T \propto \sqrt{t}$ and assuming the Saha Boltzmann law that would give a time dependence of Fe III/Fe II $\propto t^{2.75} f(\sqrt{t})$ where $f(T)$ is a weak function of T. The preceding assumes that the electron density exactly follows the density and adiabatic expansion, both of which are violated. Thus, the results we see must be determined by the electron density, which in turn is controlled by the rate of gamma-ray escape. The abundance of of Fe II increases with the decreasing abundance of electrons, exactly opposite to the Saha Boltzmann expectation.

So rather than assuming that the physical effect is due to variations in the kinetic energy of the explosions, it is more likely that rather our simple variation of parameter roughly captures the behavior in the variation of gamma-ray escape.

Given that the fluorescence associated with the Fe III/Fe II transition (Kasen, 2006; Jack et al., 2015) is related to the secondary maximum in the IR, we would expect to see variation in the secondary maxima of plateau versus non-plateau supernovae as we noted in the median light curve in the I-band, see Figure 2.13.

The variation between the spectral evolution in the plateau case and the non-plateau case is illustrated in Figure 2.19. In both cases we see the evolution of the Si II λ6355 line into features created by blends of Fe II. In the non-plateau case the Si II features are more washed out.

Note that this discussion, based on P6 simulated behavior is not only consistent with a P6 plateau created in the IME region where IGE start becoming important, but also with a P7 plateau formed much deeper where the IGE alone dominate.

In Figure 2.17 and Figure 2.18 the evolution of the ratio of Fe III/e^- for three

W7 models is shown. In the bottom row, the standard W7 model, which shows a marked plateau, has Fe III hang up in the photospheric region $10000 < v < 15000$ km/s during the plateau time ~ 25 days. The model with 1.3 times the velocity shows a similar behavior, but the inward evolution of the Fe III abundance is much faster with epoch. Similarly, the model with 0.7 times the standard W7 velocity shows a much slower evolution and the Fe III abundance only recedes in velocity at the latest times.

This hypothesis would tend to validate our prediction that the plateau/non-plateau parameterization is likely to be continuous rather than discrete.

From the physics discussion above, we think that, in particular, the P7 plateau is formed in a relatively narrow region that has to have quite a high emissivity. This is likely related to the fact that in order to reproduce the Phillips relation, the distribution of iron group elements can not be varied arbitrarily (Hoeflich, 2002; Kasen and Woosley, 2007). This seems to be confirmed by the PHOENIX experiments, where despite the rather drastic changes to the W7 model the P7 light curves are qualitatively unchanged.

Kasen (2006), finds that the Fe II/Fe III emissivity in the I-band peaks at a temperature of around 6000 K. While Kasen (2006) does not plot the R-band emissivity, we have seen from both the observations and models that the light curves in I and R exhibit similar behavior. Thus, we expect the presence/absence of the plateau is indicative of an iron region at roughly the same velocity that reaches and stays near to 6000 K during the plateau phase. In LTE approximation, this means that at about 22.5 days after blue maximum light, in the P7 narrow

band region, the flux is formed by a region of roughly constant total emissivity. While this is an approximation, it leads us to make the hypothesis that the P7 plateau flux should be very standard among SNe Ia. This is observationally supported by the absence of correlation between $x1$ and P7 height and the weak correlation between pEW Si 4000 and P7 height, leaving essentially dust, which is an extrinsic cause, as the like cause of its variation.

The unique spectrophotometric dataset of the Nearby Supernova factory allows us to intercompare the monochromatic flux of different supernova in novel ways. Here, we have examined the flux in two narrow wavelength ranges and found regions that remain constant for ten or more days. We find two regions, that we have denoted P6 and P7. P6 is near the defining Si II line and appears to have a continuum of behavior, that is there is a continuum of supernovae that have P6 plateaus to those without. The P7 region appears to be a consistent, almost ubiquitous property of SNe Ia. While we believe both plateaus are associated with iron fluorescence and strongly related to the Fe III/Fe II transition region, it is clear that they are probing different physical aspects of the ejecta.

2.5.2 P7 narrow band: a probe of total dust extinction?

P7 is apparently an intrinsic property of SNe Ia. The correlation of P7 with SALT2 color, c and its lack of correlation with $x1$ makes it tempting to assume that P7 plateau height could be an almost constant feature of SNe Ia, with the observed variation in P7 due to dust extinction. Since probing this hypothesis would need an additional detailed study of the SN Ia hosts population and NaID

features, we consider it beyond the scope of this paper.

It is difficult to consider a flux plateau with a duration of more than ten days in such a variable event as a SN Ia without wondering how standard the plateau flux is between SNe Ia.

SNe Ia have been considered as potential *standard candles* because their restframe blue magnitude shows a dispersion of $\sim 40\%$ (Phillips et al., 1999). Their adoption as good cosmological probes stems from the ability to make them correctable candles (Phillips, 1993; Phillips et al., 1999). To this end we calculated a Hubble Diagram using the P7 height ($\equiv$ m70) in place of the rest frame absolute magnitude in the B band. Initially, it seemed that m70 could be more standard than M_B since the rms residuals for uncorrected M_B were $\sigma = 0.47$, whereas those for m70 were $\sigma = 0.27$. However, m70 is significantly to the red of the B-band and we could just be seeing the reduced effect of dust in the redder bands (Krisciunas et al., 2004; Papadogiannakis et al., 2019b). Taking into acount the reduced effects of dust m70 has a dispersion very similar to that of uncorrected M_B. Therefore, m70 is in fact measuring something closely related to M_B, in spite of not showing a strong correlation with $x1$. This is consistent with our hypothesis that P7 is a very homogeneous intrinsic property of SNe Ia, providing a very homogeneous flux whose dispersion is essentially driven by other causes, for example, by dust.

This makes the P7 plateau an extremely powerful probe of SNe Ia intrinsic properties and makes it a potentially powerful tool for disentangling extinction by dust or other extrinsic causes. This is a subject for further development.

2.5.3 Connections to Other Work

Ashall et al. (2019) show that the velocity of the H-band break in the spectra serves as a measure of the position of $_{56}$Ni and hence correlates with the Branch sub-class. If our supposition that P6 forms in the IME region and P7 forms in the IGE region we should expect to see a correlation between P6, P7, and the H-band break velocity. Similarly, Papadogiannakis et al. (2019a) analyze the time of onset of the secondary maximum in the r band as well as the integrated flux under the secondary maximum, $\bar{\mathcal{F}}_{r2}$. They find the time of onset of the secondary maximum to be well correlated with the parameter s_{BV} (Burns et al., 2014), which is related to $x1$. They find that $\bar{\mathcal{F}}_{r2}$ is related to the total progenitor mass through the transparancy timescale and using the models of Goldstein and Kasen (2018) relate this to the total mass of the progenitor white dwarf. Their results are not inconsistent with our results, with $\bar{\mathcal{F}}_{r2}$ seeming to be more likely similar to P6. Indeed one can interpret our `PHOENIX` experiment as altering the time of transparency.

2.5.4 Summary

P6 is correlated with light curve shape parameters, but it also contains more information. Using a set of `PHOENIX` models that were constructed for an entirely different purpose, we are able to reproduce the behavior of the P6 plateau. We find that the time spent in the Fe III/Fe II recombination phase is correlated with the nature of the plateau (or lack thereof). These results are in general agreement

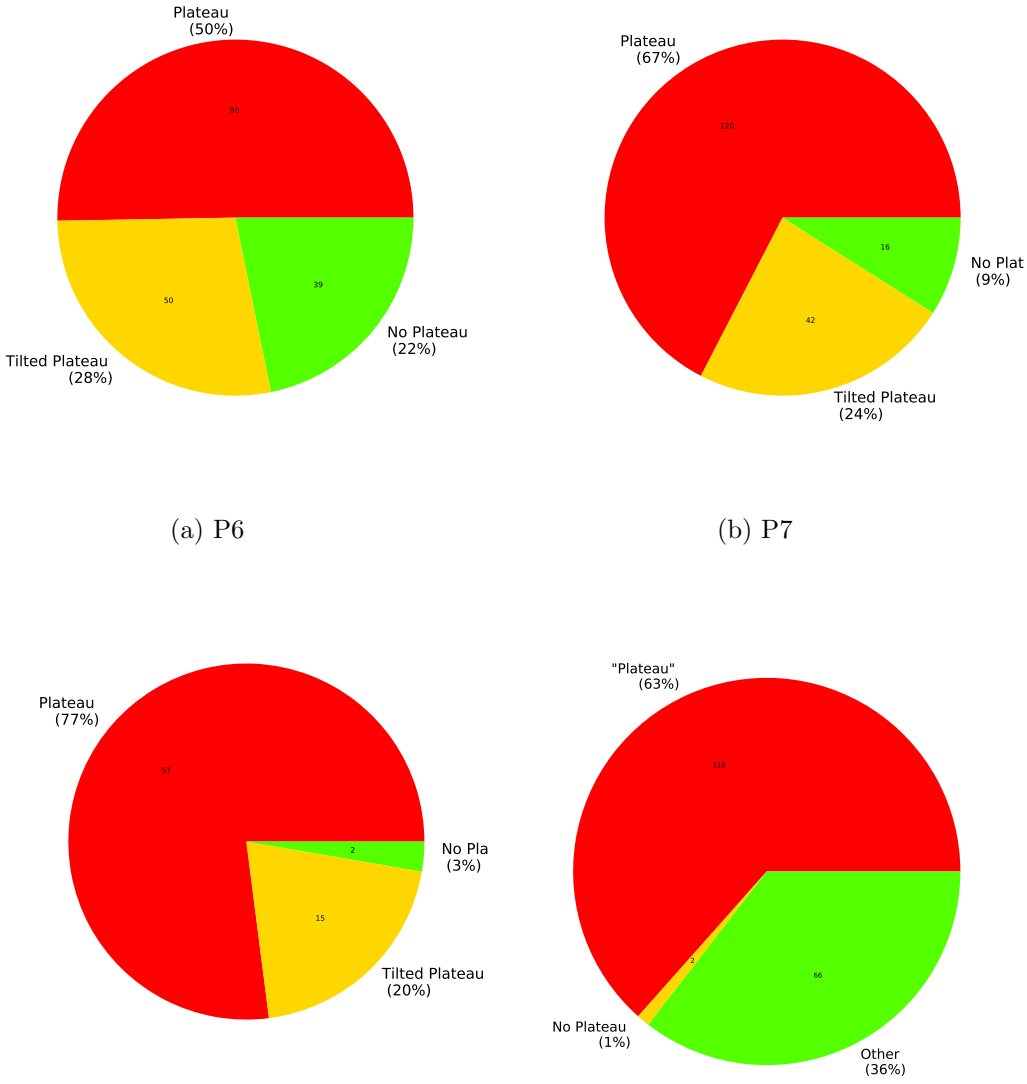
with those of Kasen (2006). We note that while our models were constructed solely to produce mathematical variations, and the variations are not physical, we have to interpret the model results with some caution. However, the underlying physics of recombination/fluorescence is quite generic to all possible models for the SNe Ia phenomenon. Given that we do not understand the underlying ejecta model this is sufficient for this work, but future work should examine the effects in a full NLTE context.

The P7 plateau, in addition to being a common feature of SNe Ia, displays a stunning flux homogeneity, with a flux dispersion over the full sample of $\sigma_{P7} = 0.27$. This, together with the absence of a correlation between P7 plateau flux and $x1$ and its very small correlation with pEW Si 4000, leads us to make the hypothesis that the main driver of P7 flux variation is dust. This makes P7 plateau potentially an extremely powerful handle to disentangle intrinsic SNe Ia variability from that due to dust.

The analysis of the remaining correlation of P7 height with the intrinsic pEW Si 4000, as well as the study of SNe Ia host population and Na ID features presence in their spectra is an exciting avenue that our reporting of this new SNe Ia feature clearly points at.

2.6 Acknowledgements

This work was done in collaboration with Sebastion Bongard. The Nearby Supernova factory supplied the observational data and SALT2 fitting used in this work.



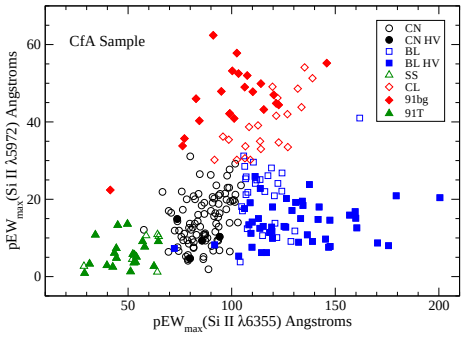
(a) P6

(b) P7

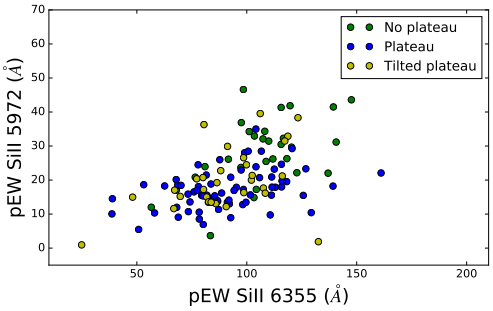
(c) Both

(d) Both Intersections

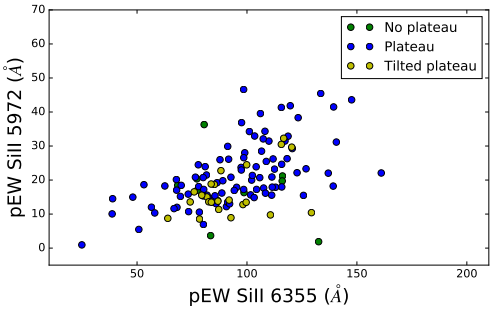
Figure 2.4 **Panel (a):** The break out of SNe Ia between the groups that display a P6 plateau, that display a P6 tilted plateau, and the group that display no plateau. **Panel (b):** Same as panel (a) for the P7 plateau. **Panel (c):** In red the group of SNe Ia displaying both P6 and P7 plateau. The green group represents SNe Ia displaying both P6 and P7 tilted plateau. And the yellow group represents SNe Ia that show no P6 nor P7 plateau. **Panel (d):** In red the group displaying P6 plateau or tilted plateau *and* P7 plateau or tilted plateau. The yellow group represents SNe Ia without P6 plateau or tilted plateau *and* P7 plateau or tilted plateau. Finally, the green group represents the SNe Ia that display one plateau or tilted plateau and not the other.



(a) CfA Sample

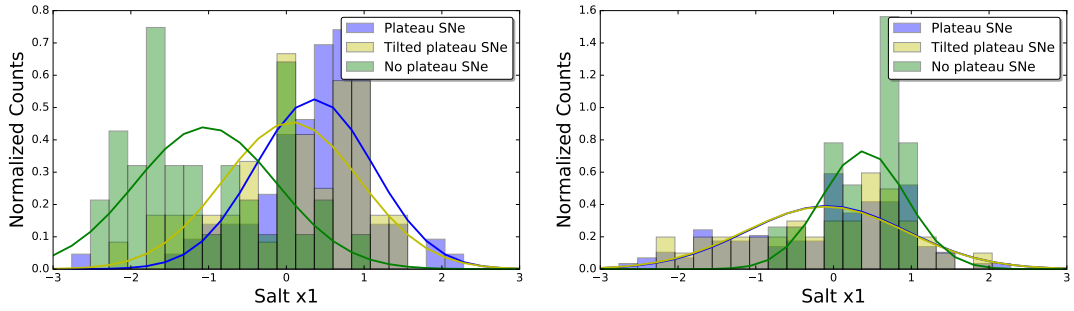


(b) P6



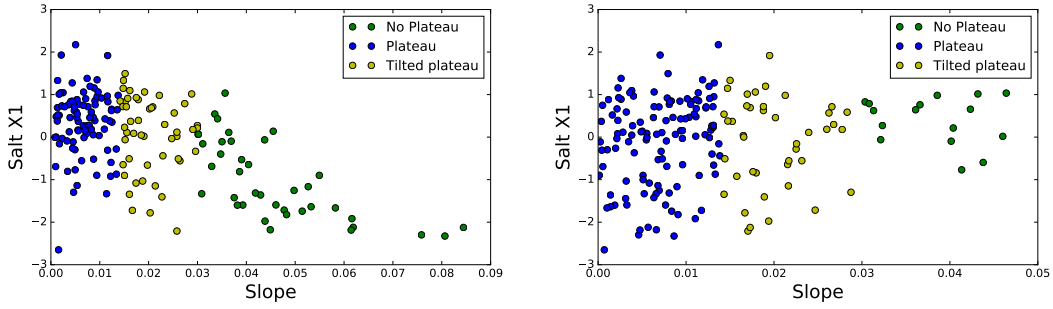
(c) P7

Figure 2.5 Panel (a): Branch diagram for CfA sample. Panel (b): Branch diagram for the P6 sample Panel (c): the same diagram for the P7 sample. The box shows the region of the core normals in the P6 and P7 samples.



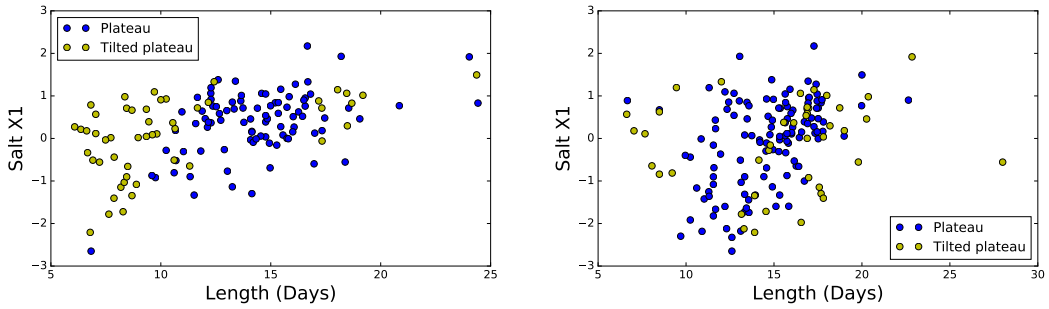
(a) P6 (b) P7

Figure 2.6 Histogram of the plateau, tilted plateau and no plateau populations for P6 in panel (a) and P7 in panel (b), dispersed with respect to SALT2 $x1$ light curve width parameter. The average values and standard deviation of $x1$ for the three P6 groups are: no plateau, $\bar{x} = -0.8$, $\sigma = 0.9$; plateau, $\bar{x} = 0.4$, $\sigma = 0.7$; titled plateau, $\bar{x} = 0.05$, $\sigma = 0.9$. The average values and standard deviations for the P7 groups are: no plateau, $\bar{x} = 0.4\sigma = 0.5$; plateau, $\bar{x} = -0.08$, $\sigma = 1.$; titled plateau, $\bar{x} = -0.1$, $\sigma = 1$.



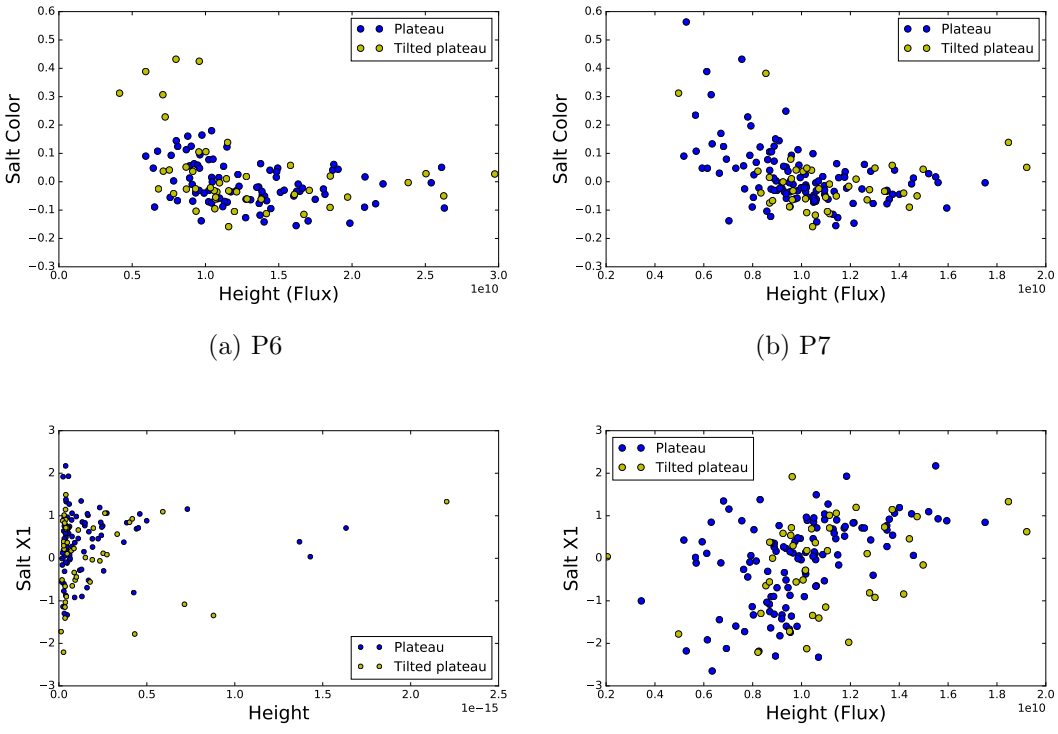
(a) P6 (b) P7

Figure 2.7 Panel (a): Slope versus $x1$ for P6. Panel (b): Slope versus $x1$ for P7.



(a) P6 (b) P7

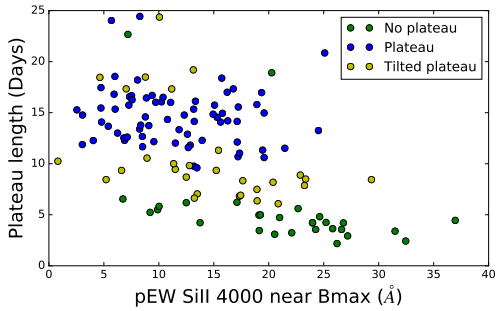
Figure 2.8 Panel (a): Length versus $x1$ for P6. Panel (b): Length versus $x1$ for P7.



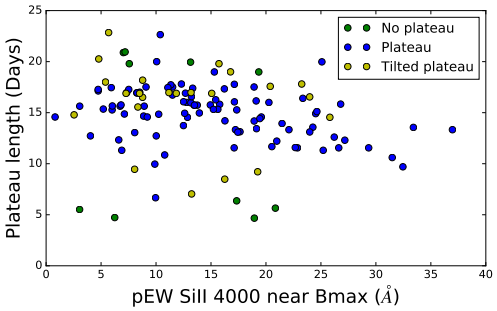
(a) P6 (b) P7

(c) P6 (d) P7

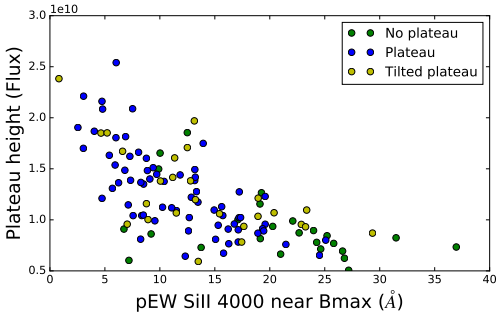
Figure 2.9 Panel (a): Height versus c for P6. Panel (b): Height versus c for P7. Panel (c): Height versus $x1$ for P6. Panel (d): Height versus $x1$ for P7.



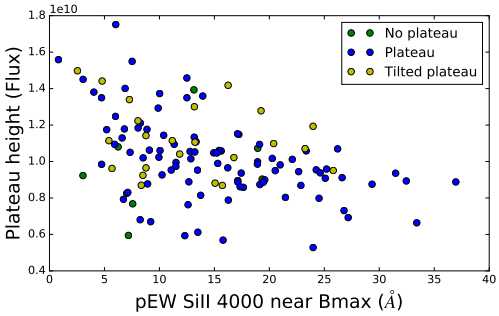
(a) P6



(b) P7

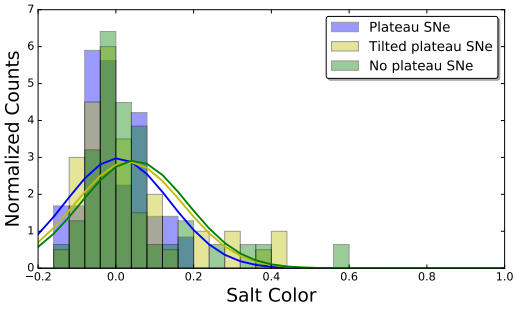


(c) P6

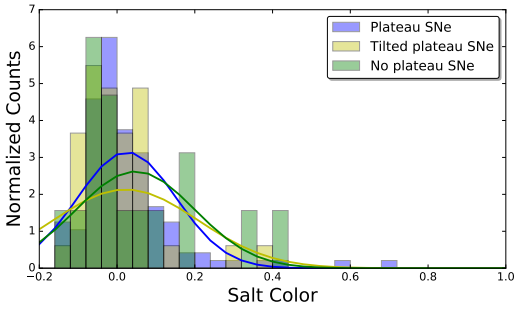


(d) P7

Figure 2.10 Panel (a): pEW Si 4000 versus length for P6. Panel (b): pEW Si 4000 versus length for P7. Panel (c): pEW Si 4000 versus height for P6. Panel (d): pEW Si 4000 versus height for P7.

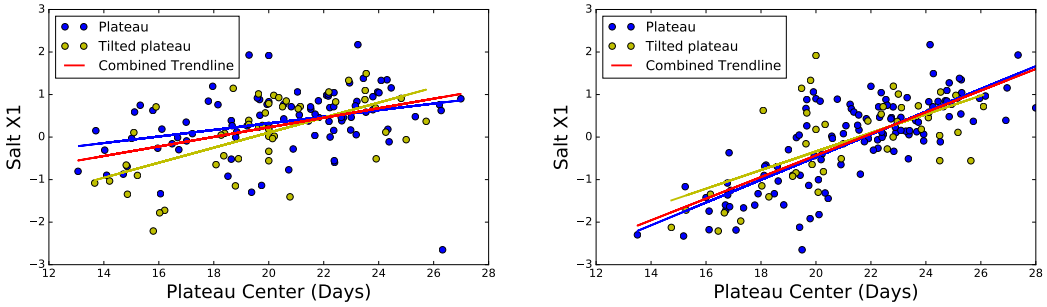


(a) P6



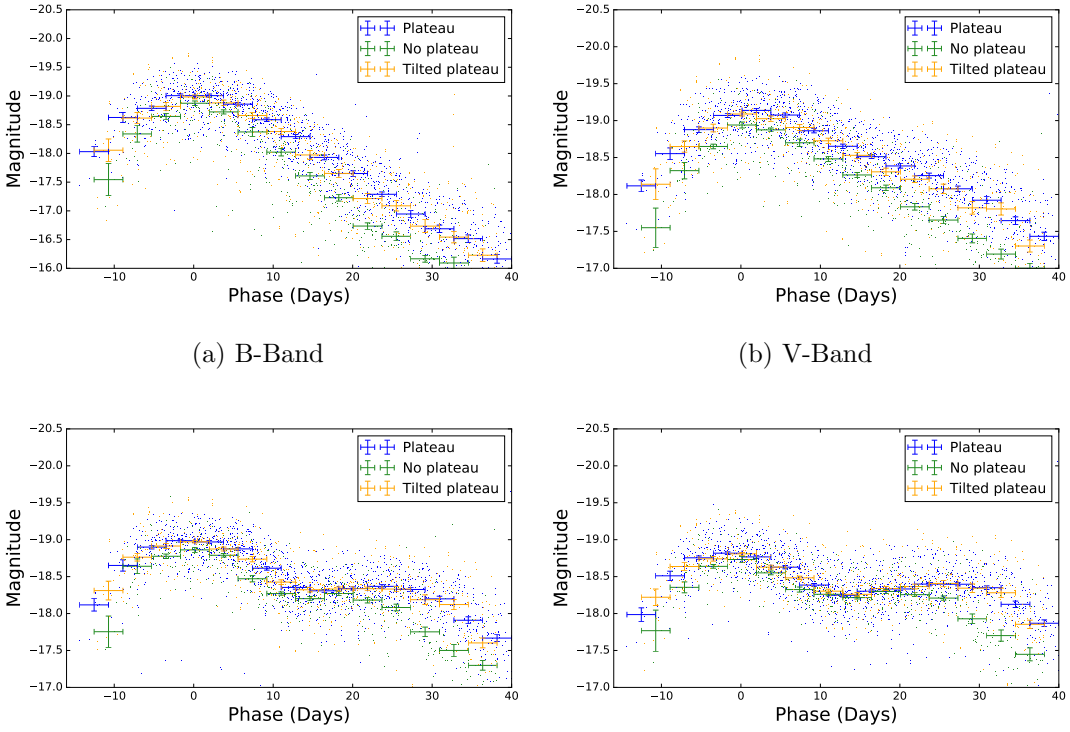
(b) P7

Figure 2.11 Histogram of P6 (Panel a) and P7 (Panel b) populations dispersed along the SALT2 color parameter distribution. The color average values and dispersions for P6 three groups are: no plateau, $\bar{x} = 0.04$, $\sigma = 0.1$; plateau, $\bar{x} = -0.01$, $\sigma = 0.07$; titled plateau, $\bar{x} = -0.08$, $\sigma = 0.1$. The average values and standard dispersions for P7 are: no plateau, $\bar{x} = 0.05$, $\sigma = 0.1$; plateau, $\bar{x} = 0.02$, $\sigma = 0.1$; titled plateau, $\bar{x} = 0.02$, $\sigma = 0.2$



(a) P6 (b) P7

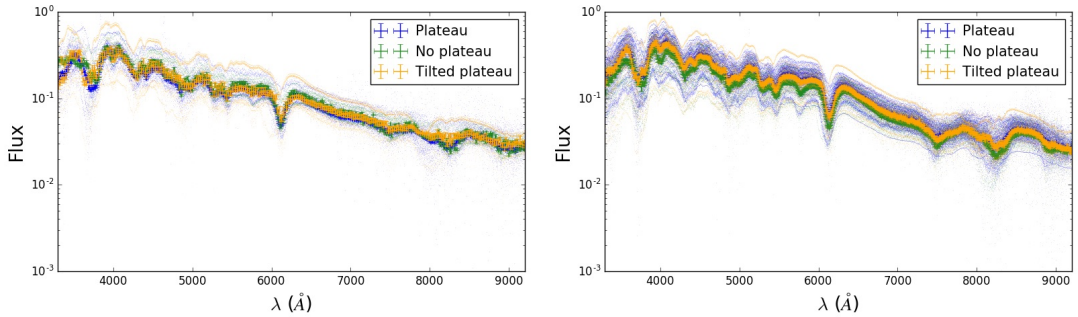
Figure 2.12 Panel (a): Centers versus $x1$ for P6. The correlation coefficients (Pearson) are $r = 0.3, 0.6, 0.4$, for plateau, tilted, and combined, respectively. Panel (b): Centers versus $x1$ for P7. The correlation coefficients (Pearson) are $r = 0.8, 0.6, 0.6$, for plateau, tilted, and combined, respectively.



(a) B-Band (b) V-Band

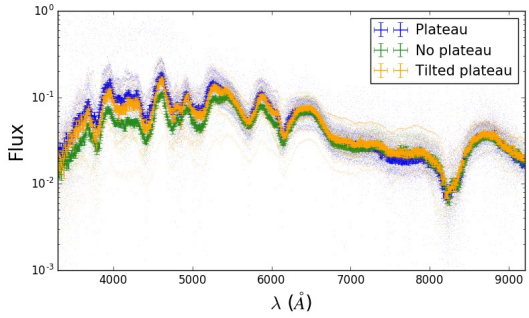
(c) R-Band (d) I-Band

Figure 2.13 SNF Median Lightcurves for P6 plateau, tilted plateau and no plateau groups.

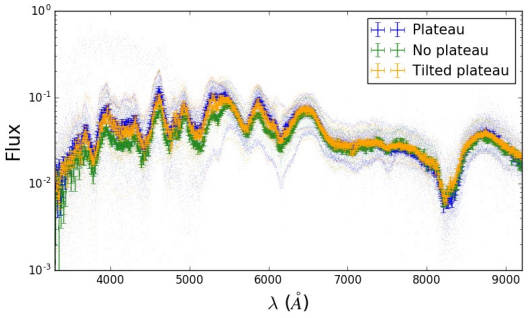


(a) -6 – -5 bin

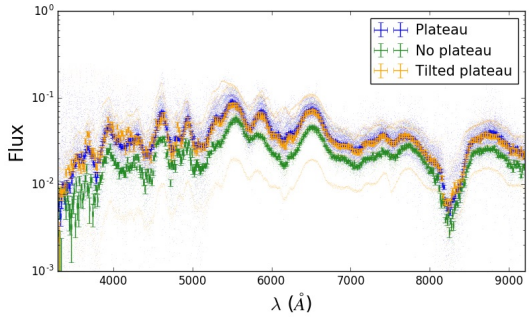
(b) -1 – +1 bin



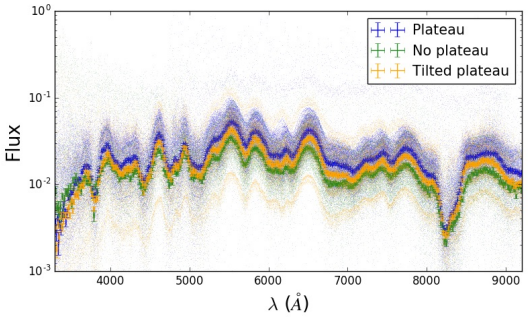
(c) +15 – +16 bin



(d) +20 – +21 bin

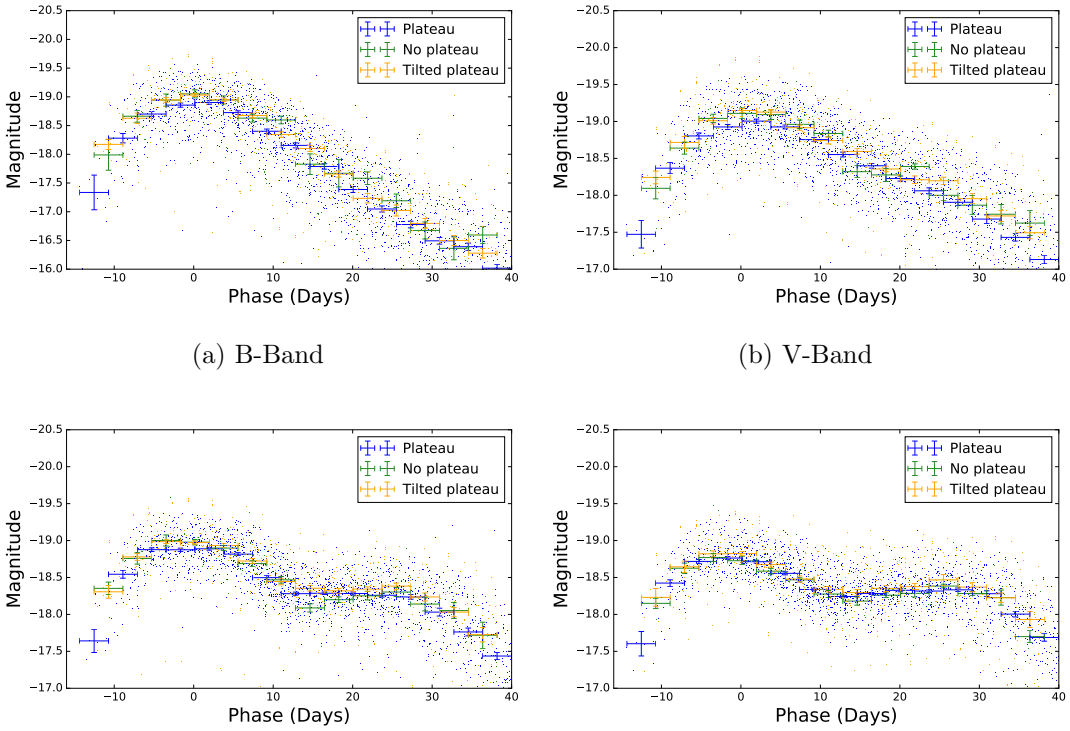


(e) +27 – +28 bin



(f) +35 – +40 bin

Figure 2.14 Median spectra for the three P6 no plateau, tilted plateau and plateau categories.



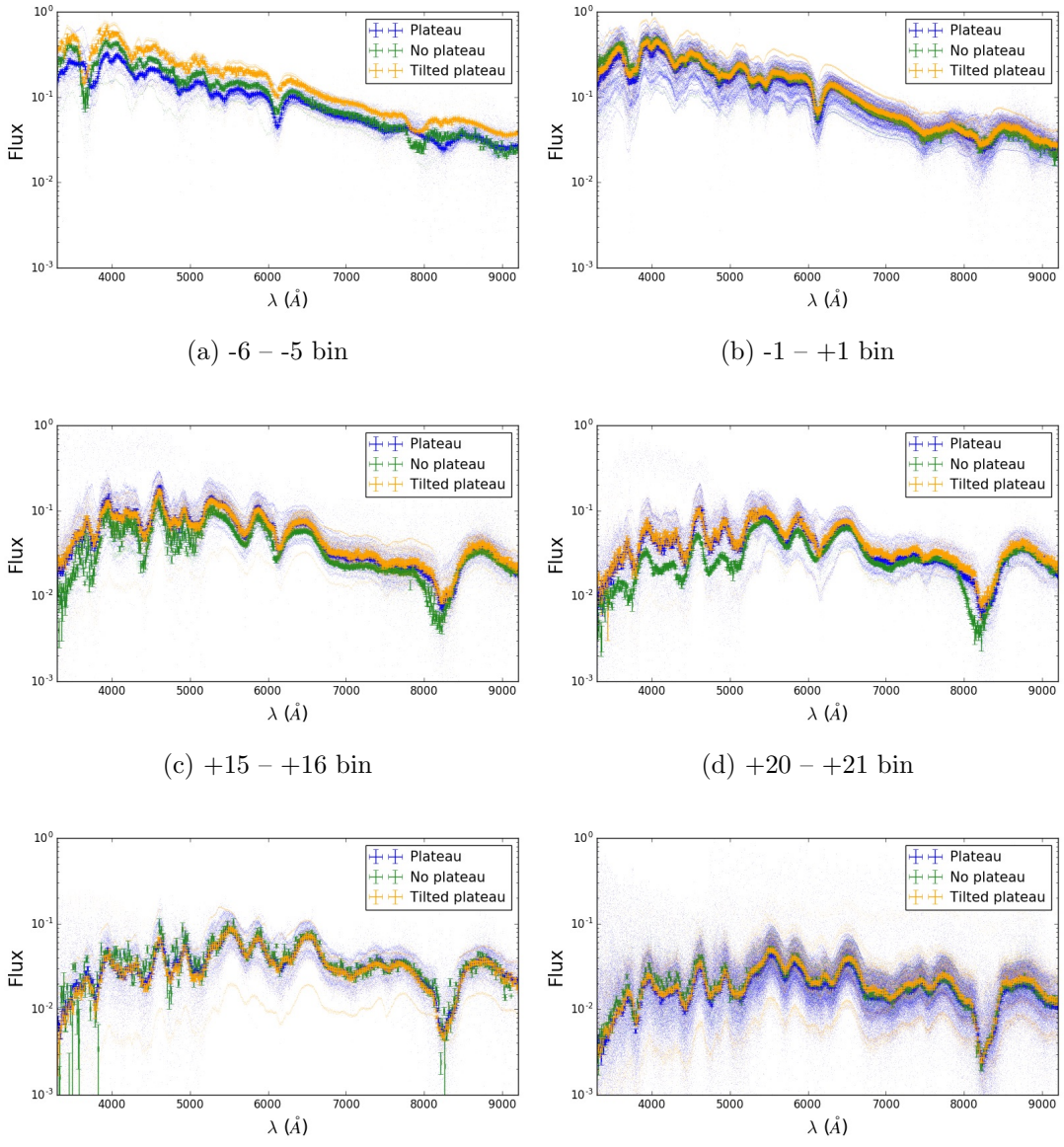
(a) B-Band

(b) V-Band

(c) R-Band

(d) I-Band

Figure 2.15 Broad band median lightcurves for three P7 plateau, tilted plateau and no-plateau categories.



(a) -6 – -5 bin

(b) -1 – +1 bin

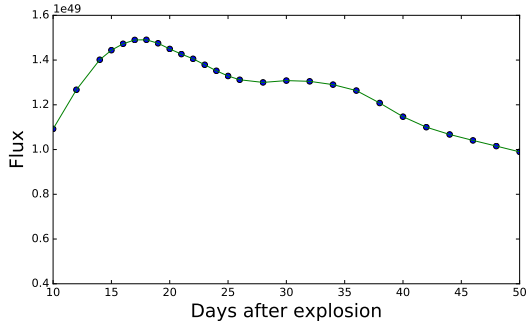
(c) +15 – +16 bin

(d) +20 – +21 bin

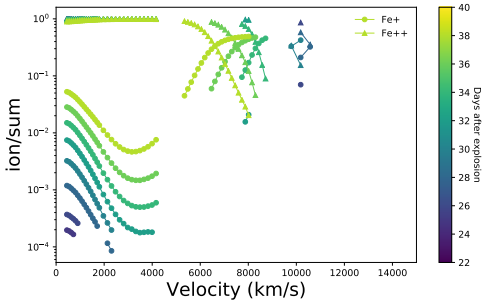
(e) +27 – +28 bin

(f) +35 – +40 bin

Figure 2.16 Median spectra for three P7 plateau, tilted plateau and no-plateau categories.

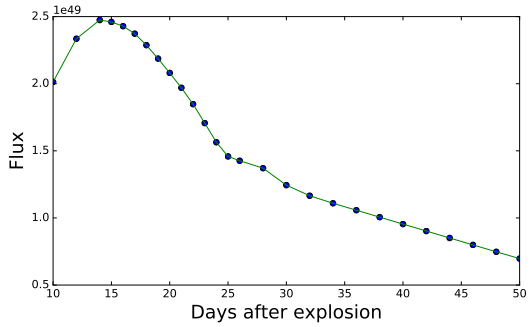


(a) Light Curve

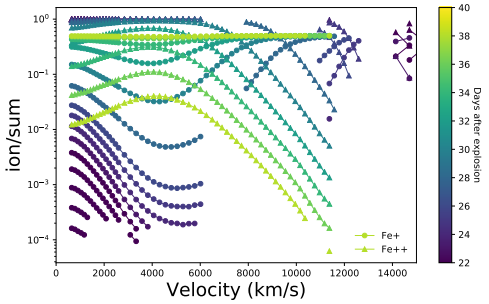


(b) Ion Ratios

Figure 2.17 The narrow lightcurve (P6) and iron ion ratios [$Fe^{+}/(Fe^{+}+Fe^{++}+e^{-})$ and similarly for Fe^{++}] of Model W7 with all velocities decreased by a factor of 0.90. This model shows a clear plateau.

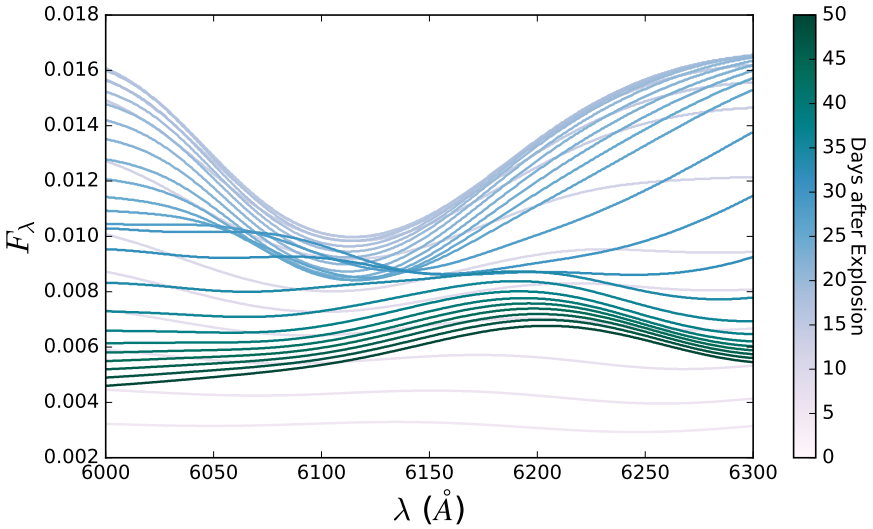


(a) Light Curve

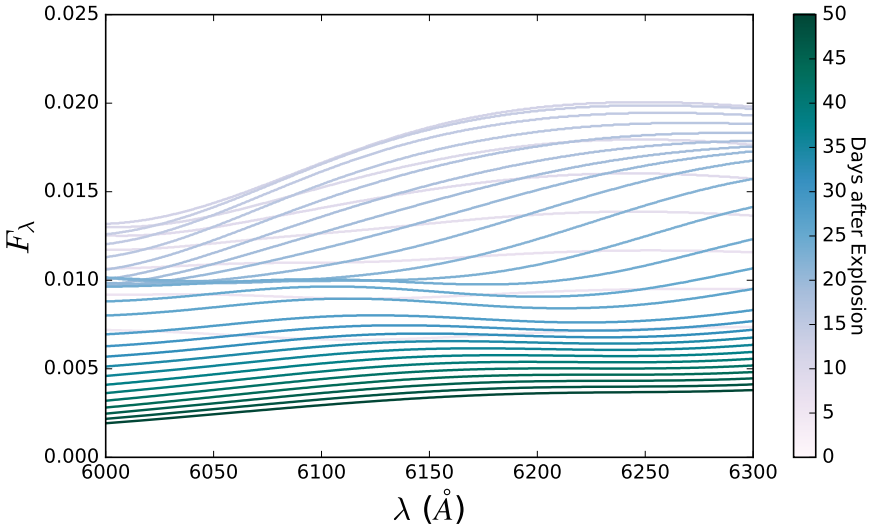


(b) Ion Ratios

Figure 2.18 The narrow lightcurve (P6) and iron ion ratios (same is in Figure 2.17) of Model W7 with all velocities increased by a factor of 1.30. This model shows scant evidence for a plateau.



(a)



(b)

Figure 2.19 Panel (a): The evolution of the spectra of Model W7 in the wavelengths around the narrow bin. Panel (b) The evolution of the spectra of Model W7 with all velocities increased by a factor of 1.5 in the wavelengths around the narrow bin.

Chapter 3

Post Merger Viscous Evolution of Chandrasekhar Mass Binaries of Differing Mass Ratios in FLASH

3.1 Introduction

Type Ia supernova (SN Ia) have been used and will continue to be used in studying the expansion of the universe because of their usefulness as standardizable candles (Riess et al., 1998; Perlmutter et al., 1999). As important as these supernovae are in the study of dark energy there are still several things we don't know about them. Type Ia supernovae are thermonuclear explosions of white dwarfs. We know that these explosions are a result of interactions in binary systems, but the exact nature of the progenitor system(s) remains unknown (Hillebrandt et al., 2013). There have been several proposed progenitor scenarios. All of these have been able to explain some of the observed properties, but not all of them. As such, there is not a consensus on which system or systems are the most likely. It is possible that type Ia supernova have several different types progenitors.

One of the oldest proposed progenitor models is the single degenerate scenario (Whelan and Iben, 1973). In this scenario the supernova occurs due to mass transfer between a white dwarf and a non-degenerate companion star. The mass transfer causes the mass of the white dwarf to hit the Chandrasekhar mass. Once

this occurs the white dwarf is no longer stable and carbon burning is ignited. How the burning occurs and causes the explosion leads to a few variations of this progenitor model, but the explosion unbinds the star and produces iron group elements, intermediate mass elements, and possibly some unburned carbon and oxygen.

Another group of proposed progenitor models are double degenerate scenarios (Iben and Tutukov, 1984; Webbink, 1984). A system of two or more white dwarfs is the progenitor system in all of these scenarios. In this case the mass of the explosion doesn't have to be the Chandrasekhar mass. The explosion in these cases can be caused by a violent merger of the white dwarfs or may involve accretion of tidally disrupted material. As in the other scenarios the explosion is powered by burning the carbon and oxygen. This scenario also unbinds the star and produces the range of observed elements from iron group down to carbon.

It is important for progenitor scenarios to be able to reproduce the observed properties of SNe Ia. Some observational constraints are observed rates, the amount and location of elements in the ejecta, the energy of the explosion, the observed light curve shapes, observed spectral properties, polarization, etc. The models of each progenitor are not perfect and none of the scenarios have been able to explain and match all of the observations. SNe Ia may have more than one type of progenitor, meaning a progenitor model may only explain a portion of the observations.

There are several different models of non-violent mergers between two white dwarfs. The oldest models have the secondary form an accretion disk around the

primary much like the single degenerate picture without a donor star. In this case the material accreted is already carbon and oxygen so it will not undergo any nuclear burning. Piersanti et al. (2003) show this will produce a wind and can spin up the primary star. In this picture the primary white dwarf will explode once it has accreted enough material to reach the Chandrasekhar mass.

Yoon et al. (2007) and Shen et al. (2012) find that the disrupted white dwarf will form a hot envelope as well as a thick disk around the primary white dwarf. Very little mass is lost in this model. The envelope will deposit material onto the primary and gain mass from the thick disk. This model can have hot spots. They find that the temperature in these spots may become hot enough to ignite carbon burning. If the white dwarf burns to an oxygen neon composition it will collapse into a neutron star rather than explode. They find a narrow range of mass combinations that could lead to a SN Ia.

Schwab et al. (2016) find a secular merger results in a hot partially thermally supported envelope around a cold-core composed of the primary white dwarf. In this model the temperature at the interface between the core and envelope is hot enough for self-sustaining carbon burning. Shen et al. (2012) further find that the burning near the interface will proceed inward and burn all of the star. This will lead to an accretion induced collapse. So they conclude that these models will not produce SN Ia.

Zhu et al. (2013) looked at the differences between mergers of similar and dissimilar masses. For models where the mass ratio of the initial system is significantly different from one they find the cold core with warm envelope and

thick disk picture already discussed. For models where the mass ratio is near one the picture is different. In this case they find that both stars are disrupted and there is significant mixing of the stars. These models have a hot core with a thick disk. The hottest points in this model are near the center. The core of this model has rapid rotation.

Previous modeling work on non-violent mergers has focused on systems where the mass ratio is less than 0.7. These studies have found these mergers are more likely to collapse into a neutron star than to produce a SN Ia (Shen et al., 2012; Schwab et al., 2016). This is because carbon burning initiated on the surface of the merged white dwarf. Zhu et al. (2013) looked at a range of mass ratios to compare the amount of mixing between the material of the original white dwarfs. This study found that in the case of equal mass or nearly equal mass merging white dwarfs the primary was more disrupted in the merger. This led to deeper mixing of material from the secondary and could move the carbon ignition point further into the merged white dwarf.

This work will focus on the viscous evolution of merging white dwarfs in the time just after a non-violent merger. The goal is to compare the differences in the evolution of systems with the same total mass but different mass ratios. In each case the total mass of the system is the Chandrasekhar mass. The models are stopped if conditions for carbon burning are met.

Starting from 3D SPH models that have been evolved through the dynamical evolution of the merger to a state of quasi-hydrostatic equilibrium. We use FLASH4.4 (Calder et al., 2002) to model the viscous evolution of the white dwarf

Table 3.1. SPH models uses as the starting point for FLASH simulations

Primary Mass $M_\odot$	Secondary Mass $M_\odot$	Simulation Time s
0.7	0.7	1400
0.9	0.5	300

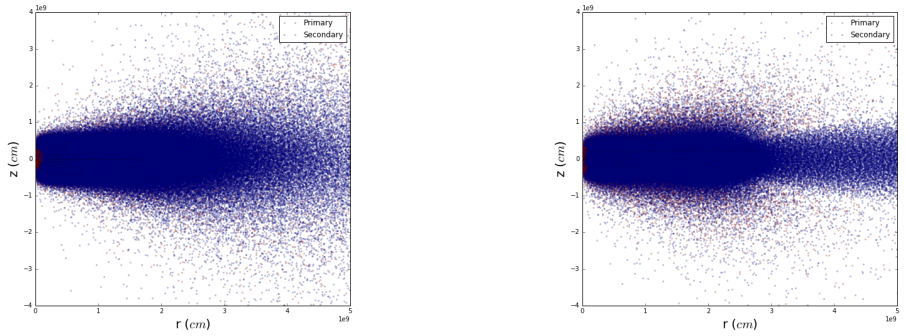
merger. FLASH is a modular multi-physics simulation code, useful for modeling many different physical systems. We run a 2.5D FLASH4.4 model, by assuming symmetry around the z-axis but maintaining the angular velocities, until carbon burning conditions are met or a simulation time of 120 seconds is reached.

First we discuss the SPH simulations used to initialize our models. Then we explain how we converted these 3D particle models into a 2.5D mesh grid needed for the FLASH4.4 simulations. Next we discuss the viscosity treatment in FLASH and our initially assumed magnetic field. Then we discuss the results of our models.

3.2 SPH models

This work focuses on the viscous evolution of merging white dwarfs. The initial state of each of our models comes from SPH models of Sato et al. (2015, 2016). These models were run with 716800 particles in 3D. We used two of these models with different mass ratios: 0.7 and 0.7 solar masses (77), and 0.9 and 0.5 solar masses (95). These models where run until a state of quasi-static hydrodynamic equilibrium was reached. Information on the merger simulation for each of these models are given in Table 3.1.

It is clear from Figure 3.1 that significant mixing of material from each of the white dwarfs occurs during the merger in all cases not just the similar mass case as Zhu et al. (2013) found. The models with a greater mass difference reached a quasi-static state faster than the model with a mass ratio of 1.



(a) 0.7 + 0.7 (b) 0.9 + 0.5

Figure 3.1 Distribution of Particles at the end of the SPH run. The particles are moved from 3D to 2D by setting the ϕ coordinate to 0. The color of each particle shows if it originally belonged to the primary (red) or secondary (blue).

3.2.1 Converting SPH to grid

To run the viscous evolution in FLASH we needed to convert the 3D particle data from the Sato models into a 2.5D mesh grid. First we converted all of the particles from Cartesian coordinates to cylindrical coordinates. We assume circular symmetry around the z-axis and moved all the particles to the r-z plane. We kept the angular velocities of the particles for viscosity calculations.

The modules used for this FLASH4.4 simulation require a uniform grid to use cylindrical coordinates. We chose a mesh grid smaller than the domain of the SPH code to have a fine enough grid spacing. This cut excludes a very small

portion of the mass of the merger. We used a mesh grid of 8192x8192 points over a range from 0 to 50,000 km in r and -40,000 to 40,000 km in z. This gives a grid spacing on the order of 10 km in each direction.

The particles were mapped onto the grid. Each grid point considered all particles within two smoothing lengths. The value at each point is found by taking a weighted trapezoidal integral of the values of the particles. Further discussion and the code used are included in Appx B.

3.3 Modeling Viscous evolution

We initialized the FLASH model with a magnetic field to provide viscosity. For simplicity we start the model with a dipole field outside the dense core of the merged white dwarfs and a constant field inside as shown in Figure 3.2. We set the transition from constant field strength to a dipole at 10^9 cm. The constant internal field strength and the strength at this interface were set to 10^6 Gauss.

The viscosity in the system is found using the Spitzer viscosity (Spitzer, 1962). This viscosity is valid for fully ionized gases in a magnetic field. It is reasonable to assume that all of the material in the simulation is ionized. This viscosity is due to the magnetic field and is proportional to $T^{2.5}$.

3.3.1 FLASH code setup

FLASH is a parameterized code meaning that we need to decide which physics schemes to use in the model. We used the MHD Staggered Mesh solver for the hydrodynamics. We used a Helmholtz equation of state because of the degenerate

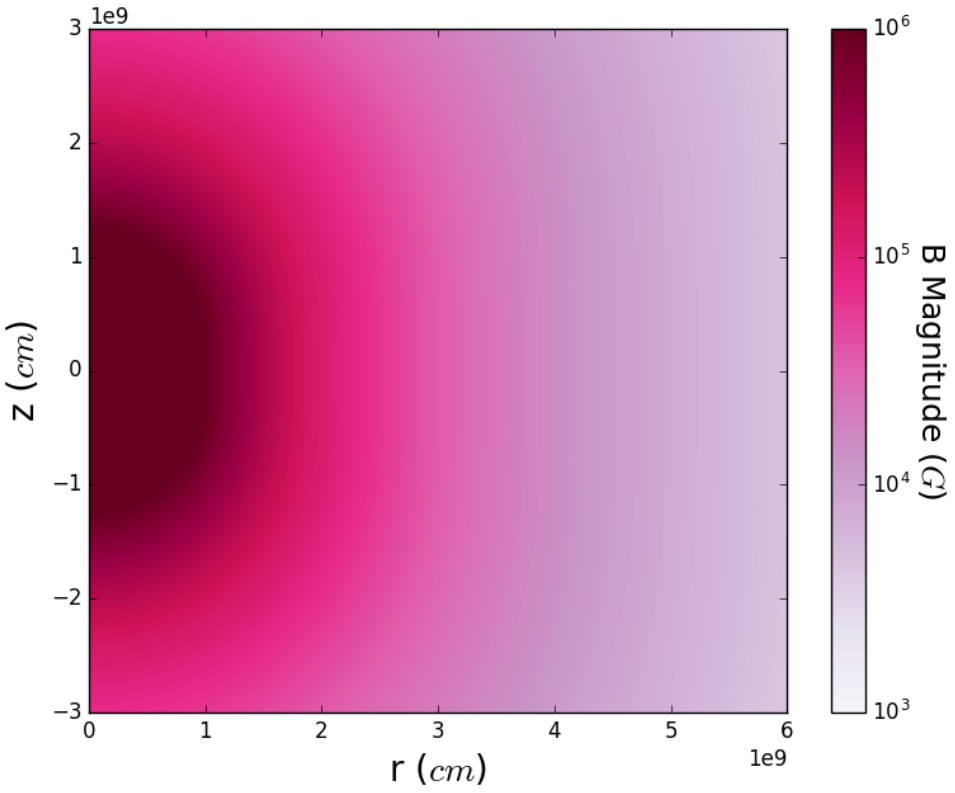
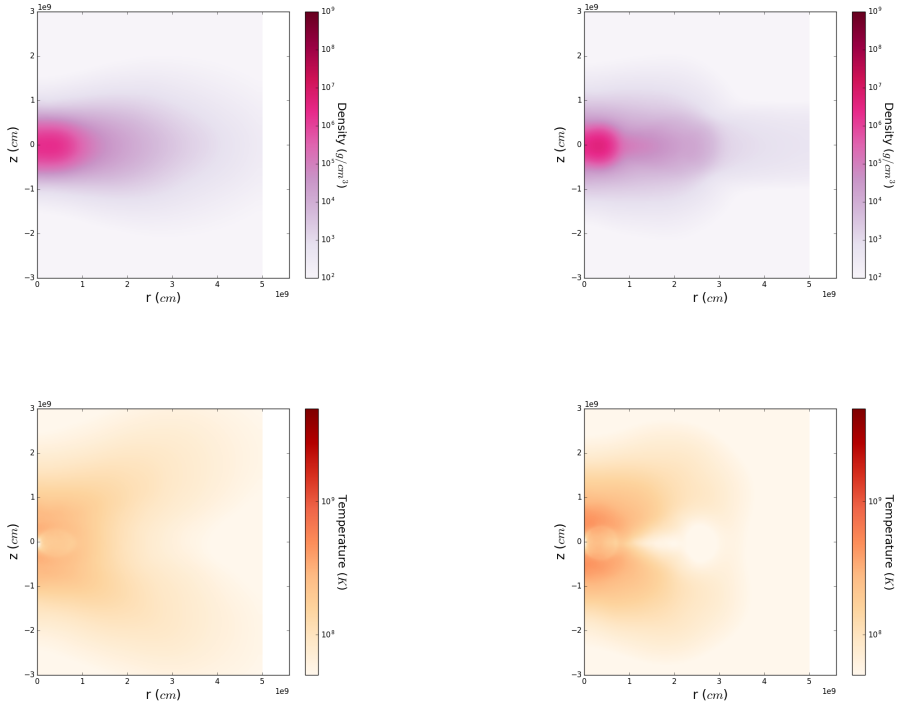


Figure 3.2 Dipole magnetic field with constant B in the center used to initialize models

nature of the material in the merger. We use the Poisson gravity module. We set the composition of the model to be 40% carbon and 60% oxygen which requires the inclusion of the nuclear burning unit; however, we do not allow nuclear burning to occur in the simulation.

Figure 3.3 show the initial density and temperature structures of each model. All of these models appear to have a cold-core surrounded by a hot envelope and thick disk. At the start of our simulations no model has any region with conditions necessary for Carbon burning.



(a) 0.7 + 0.7 (b) 0.9 + 0.5

Figure 3.3 The initial density and temperature structure from the FLASH simulations. The top row shows the density for each model, the bottom shows the temperature.

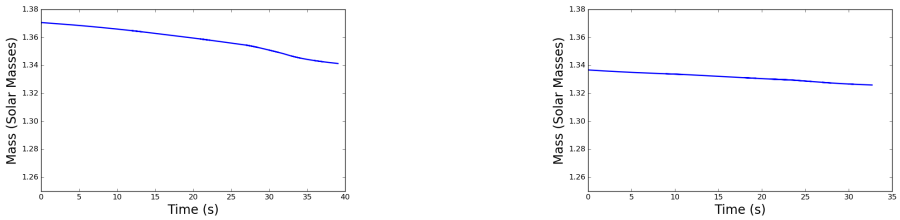
As the simulations progressed it was necessary to restrict the time-steps of the model. The Courant condition (CFL) is a time-step limiter that prevents

information from traveling more than one grid step each time step. The Courant number is given by Equation 3.1; where u is the velocity, Δt is the time-step, and Δx is the grid spacing. The CFL condition given to FLASH is the maximum value for the Courant number in the simulation. Thus lowering the CFL condition will reduce the time-steps used in the simulation because the grid spacing in unchanged. We reduced the CFL condindion given to the simulation as needed to maintain stability. We also monitored the temperature and density across the model to stop the simulation if conditions for carbon burning were met.

$$C = u\frac{\Delta t}{\Delta x} \tag{3.1}$$

3.3.2 Model outputs

The simulations stores some global information about the state of the model at each step. One of the parameters saved is the total mass of the system, Figure 3.4 shows that only a small amount of mass leaves the simulation region in the time we are modeling. From this it is clear that the none of the simulations are blowing a significant wind.



(a) 0.7 + 0.7 (b) 0.9 + 0.5

Figure 3.4 These plots show the change in the total mass of each simulation in Solar Masses over the simulation time in seconds.

As the simulations ran a snapshot of several physical parameters was saved every second. I have combined these to be able to view the evolution of the merged system.

Figure 3.5 shows a snapshot of the model at a simulation time of 30 seconds. In both the equal mass model, and the model with a large initial mass difference the material outside of the core is changing from a disk like structure to more of an envelope. In the 95 model, the core expands. In both models heating mostly occurs in the region around the core, with some heating of the core along the rotation axis. The 95 model reaches higher temperatures in this time than the 77 model. Both models still show a small cold central region.

3.4 Discussion

In each of the simulations the highest temperatures do not occur in the central regions of the merged white dwarf where the density is the highest. In the first 40 seconds of the viscous evolution we modeled, nuclear burning temperatures for carbon were not reached.

The evolution of of both the density and temperature structure of the simulation from the merger of two equal mass white dwarfs is slower than the cases with unequal mass progenitors. This is also seen in the longer simulation time needed by the SPH code to reach a quasi-stable state.

The early behavior of the models show heating on the region between the less dense envelope and dense central star. If temperatures in these hot spots become hot enough to ignite carbon burning then it is likely these models would burn

inward and lead to a collapse rather than a supernova. The models to this point also show potential for hot spots to occur along the rotation axis in the dense interior of the star. An ignition from an interior point may lead to an explosion.

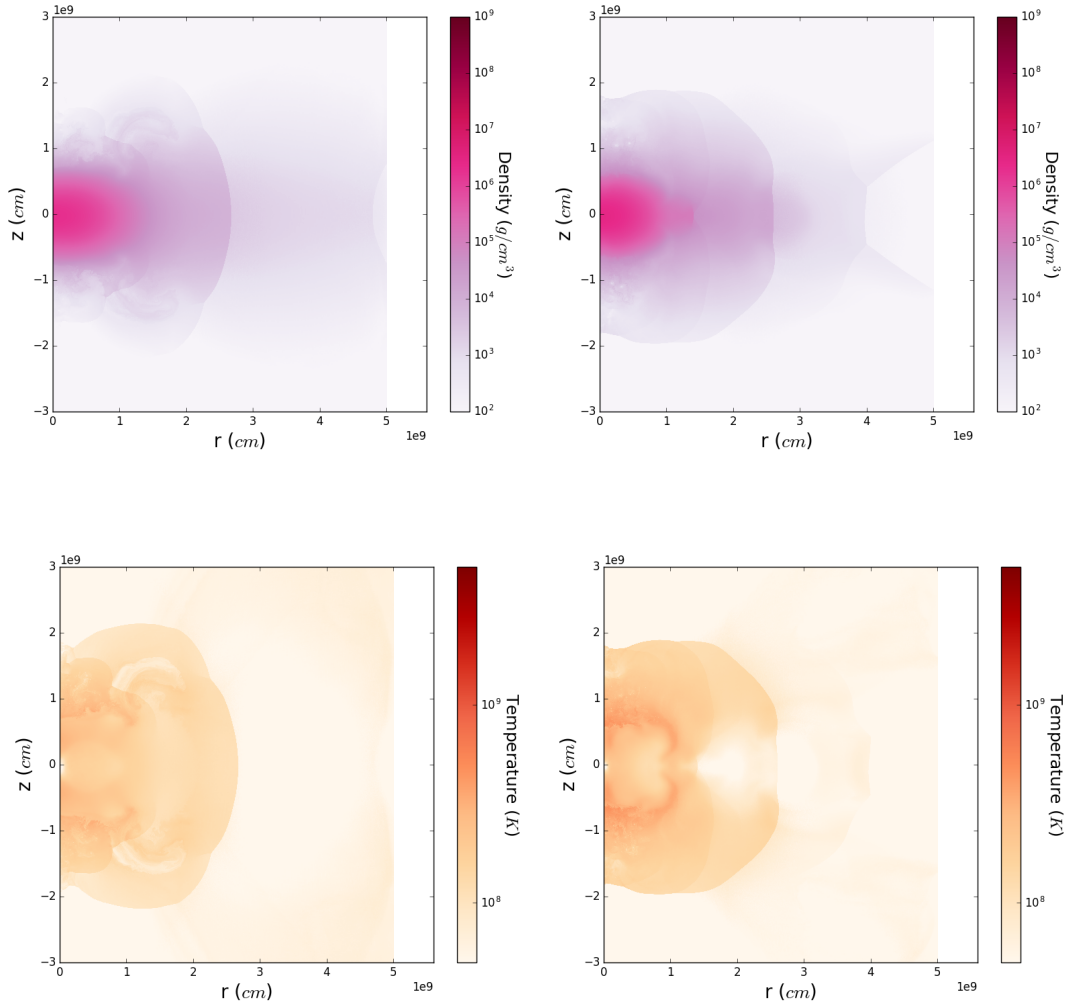
This slower evolution of the equal mass merger may allow more time for the hotter outer regions of the merged system to heat the colder inner core. If this is true it maybe that nuclear burning conditions are more likely to occur in the inner portions of an equal mass merger than in a merger of unequal mass. That situation would lead to condition more favorable for producing a supernova rather than a collapse into a neutron star.

3.5 Future Work

In this work none of the models evolved to a point where nuclear-burning initiated, and have not settled enough to expect the white-dwarf to collapse into a neutron star. Thus, further modeling of the viscous and thermal evolution of the merged systems evolution is needed. If nuclear burning conditions are reached in further modeling that alone is not sufficient to determine if a SN Ia might occur. The nuclear burning and any resulting explosion would also need to be modeled as well. The spectra produced by these would then need to be compared to observed SNe Ia to determine if any of these systems is a possible channel to produce SNe Ia.

3.6 Acknowledgments

The software used in this work was in part developed by the DOE NNSA-ASC OASCR Flash Centerat the University of Chicago. This work was done in collaboration with Keiichi Maeda of Kyoto University. This work was funded in part by the NSF EAPSI summer fellowship 2016. Support for this fellowship was supplied by both NSF and JSPS.



(a) 0.7 + 0.7

(b) 0.9 + 0.5

Figure 3.5 Snapshots of two of the models 30 seconds into the FLASH model run. The top row shows the density structure of each model, and the bottom shows the temperature structure.

Chapter 4

Summary

In seeking to fill gaps in our understanding SNe Ia it is useful to approach the problem in many ways. Observations can be used to find phenomenological features in both the light curves and spectra. Observations can help to group objects that may have different physical processes occurring. Modeling is greatly helpful in identifying the physics involved in producing SNe Ia. Modeling is necessary in finding the progenitors of these explosions because they are too faint to be observed. A greater understanding can be reached when both modeling and observational techniques are used together.

This work uses various approaches to improve our understanding of Type Ia supernovae. Greater understanding of SNe Ia diversity and the systems that lead to these explosions increases their usefulness as tools in understanding the universe. Using the Nearby Supernova factory data set we find two spectral regions with a substantial period of nearly stable flux. This is particularly noteworthy because of the rapidly changing nature of these objects. Through the use of existing `PHOENIX` models we find that this behavior may be linked to iron ionization.

A thorough search for other plateau like features in the SNf data set may find several other regions showing interesting behavior. These features may be related to features other than iron. Time series spectra from more modern explosion

models will allow for a deeper look at the physics causing the observed plateau behavior we found. This better understanding could increase the usefulness of these features in studying SNe Ia.

Hydrodynamic modeling of possible SNe Ia progenitors is a first step in determining if the scenario can actually result in a SN Ia like explosion. In this work we take 3D models of three cases of non-violently merging white dwarfs and model the next part of their evolution. None of the merged systems show significant mass loss. The models show heating between the less dense outer region and the more dense inner regions, and along the rotation axis. None of the models reached temperatures high enough to ignite carbon burning in the simulated time.

Further modeling of the viscous and thermal evolution of each of these hydrodynamic models will be needed to find if the result of each merger is a SN Ia or an accretion induced collapse to a neutron star. The total mass of each of these systems is above the Chandrasekhar limit and significant mass loss is not occuring it is not possible for the system to remain stable as a white dwarf. The region where the hot spots begin nuclear burning is likely to be very important in determining this outcome. Such simulations will need to run with nuclear burning. In the case of an explosion it will be important to test that the spectra produced match with observed SNe Ia.

Bibliography

Ashall, C. et al. (2019). "Carnegie Supernova Project-II: Using Near-Infrared Spectroscopy to determine the location of the outer ^{56}Ni in Type Ia Supernovae". In: *ApJ* submitted, astro-ph/1902.10088.

Astier, P. et al. (2006). "The Supernova Legacy Survey: Measurement of Ω_M, Ω_Λ, and w from the First Year Data Set". In: 447, p. 31.

Bacon, R. et al. (1995). "3D spectrography at high spatial resolution. I. Concept and realization of the integral field spectrograph TIGER." In: *A&AS* 113, p. 347.

Bacon, R. et al. (2001). "The SAURON project - I. The panoramic integral-field spectrograph". In: *MNRAS* 326, pp. 23–35.

Baron, E. (2014). "SNe Ia: Can Chandrasekhar Mass Explosions Reproduce the Observed Zoo?" In: *Nuclear Physics A* 928, pp. 319–330.

Baron, E. et al. (2012). "A Physical Model for SN 2001ay, a normal, bright, extremely slowly declining Type Ia supernova". In: 753, p. 105.

Benetti, S. et al. (2005). "The Diversity of Type Ia Supernovae: Evidence for Systematics?" In: 623, pp. 1011–1016.

Benetti, S. et al. (2006). "Supernova 2002ic: The Collapse of a Stripped-Envelope, Massive Star in a Dense Medium?" In: 653, pp. L129–L132.

Blondin, S. et al. (2012). "The Spectroscopic Diversity of Type Ia Supernovae". In: *AJ* 143, p. 126.

Bongard, S. et al. (2006). "Type Ia Supernova Spectral Line Ratios as Luminosity Indicators". In: 647, p. 480.

— (2008). "Multi-layered Spectral Formation in SNe Ia Around Maximum Light". In: 687, p. 456.

Branch, D. et al. (2007). "Comparative Direct Analysis of Type Ia Supernova Spectra. III. Premaximum". In: *PASP* 119, pp. 709–721.

Branch, D., L. C. Dang, and E. Baron (2009). "Comparative Direct Analysis of Type Ia Supernova Spectra. V. Insights from a Larger Sample and Quantitative Subclassification". In: *PASP* 121, pp. 238–247.

Branch, D. et al. (2005). "Comparative Direct Analysis of Type Ia Supernova Spectra. I. SN 1994D". In: *PASP* 117, pp. 545–552.

Branch, D. et al. (2006). "Comparative Direct Analysis of Type Ia Supernova Spectra. II. Maximum Light". In: *PASP* 118, pp. 560–571.

Branch, D. et al. (2008). "Comparative Direct Analysis of Type Ia Supernova Spectra. IV. Postmaximum". In: *PASP* 120, p. 135.

Burns, C. R. et al. (2014). "The Carnegie Supernova Project: Intrinsic Colors of Type Ia Supernovae". In: *ApJ* 789, 32, p. 32.

Calder, A. C. et al. (2002). "On Validating an Astrophysical Simulation Code". In: *ApJS* 143, pp. 201–229.

Candia, P. et al. (2003). "Optical and Infrared Photometry of the Unusual Type Ia Supernova 2000cx". In: *PASP* 115, pp. 277–294.

Cardelli, J. A., G. C. Clayton, and J. S. Mathis (1989). "The relationship between infrared, optical, and ultraviolet extinction". In: *ApJ* 345, pp. 245–256.

Cartier, R. et al. (2011). "A study of the color diversity around maximum light in Type Ia supernovae". In: *MNRAS* in press, astro-ph/1110.0480.

Childress, M. et al. (2013a). "Host Galaxies of Type Ia Supernovae from the Nearby Supernova Factory". In: *ApJ* 770, 107, p. 107.

— (2013b). "Host Galaxy Properties and Hubble Residuals of Type Ia Supernovae from the Nearby Supernova Factory". In: *ApJ* 770, 108, p. 108.

Chugai, N. N., R. A. Chevalier, and P. Lundqvist (2004). "Circumstellar Interaction of the Type Ia sUpernova 2002ic". In: *MNRAS* 355, pp. 627–637.

Deng, J. et al. (2004). "Subaru Spectroscopy of the Interacting Type Ia Supernova SN 2002ic: Evidence of a Hydrogen-rich, Asymmetric Circumstellar Medium". In: 605, pp. L37–L40.

Dilday, B. et al. (2012). "PTF 11kx: A Type Ia Supernova with a Symbiotic Nova Progenitor". In: *Science* 337, p. 942.

Folatelli, G. (2004). "Spectral homogeneity of type Ia supernovae". In: *New A Rev.* 48, pp. 623–628.

Folatelli, G. et al. (2010). "The Carnegie Supernova Project: Analysis of the First Sample of Low-Redshift Type-Ia Supernovae". In: *AJ* 139, pp. 120–144.

Foley, R. J. et al. (2010). "SN 2006bt: A Perplexing, Troublesome, and Possibly Misleading Type Ia Supernova". In: *ApJ* 708, pp. 1748–1759.

Foley, Ryan J. et al. (2013). "Type Iax Supernovae: A New Class of Stellar Explosion". In: *ApJ* 767, p. 57.

Ganeshalingam, M., W. Li, and A. V. Filippenko (2011). "The rise-time distribution of nearby Type Ia supernovae". In: *MNRAS* 416, pp. 2607–2622.

Goldhaber, G. et al. (2001). "Timescale Stretch Parameterization of Type Ia Supernova B-band Light Curves". In: 558, p. 359.

Goldstein, D. A. and D. Kasen (2018). "Evidence for Sub-Chandrasekhar Mass Type Ia Supernovae from an Extensive Survey of Radiative Transfer Models". In: *ApJ* 852, L33, p. L33.

González Hernández, J. I. et al. (2009). "The Chemical Abundances of Tycho G in Supernova Remnant 1572". In: *ApJ* 691, pp. 1–15.

Guy, J. et al. (2007). "SALT2: using distant supernovae to improve the use of type Ia supernovae as distance indicators". In: *A&A* 466, pp. 11–21.

Hamuy, M. et al. (2003). "An Asymptotic-Giant-Branch Star in the Progenitor System of a Type Ia Supernova". In: *Nature* 424, pp. 651–654.

Han, Z. and P. Podsiadlowski (2006). "A single-degenerate model for the progenitor of the Type Ia supernova 2002ic". In: 368, p. 1095.

Hauschildt, P. H. and E. Baron (1999). "Numerical Solution of the Expanding Stellar Atmosphere Problem". In: *J. Comp. Applied Math.* 109, p. 41.

Hawley, W. P., T. Athanassiadou, and F. X. Timmes (2012). "Zero Impact Parameter White Dwarf Collisions in FLASH". In: *ApJ* 759, 39, p. 39.

Hayden, B. T. et al. (2010). "The Rise and Fall of Type Ia Supernova Light Curves in the SDSS-II Supernova Survey". In: *ApJ* 712, pp. 350–366.

Hicken, M. et al. (2007). "The Luminous and Carbon-rich Supernova 2006gz: A Double Degenerate Merger?" In: 669, p. L17.

Hillebrandt, W. et al. (2013). "Towards an understanding of Type Ia supernovae from a synthesis of theory and observations". In: *Frontiers of Physics* 8, pp. 116–143.

Hoeflich, P. (2002). "Gamma-rays as probes for the multi-dimensionality of Type Ia supernovae". In: *New Astronomy Review* 46, pp. 475–480.

Howell, A. et al. (2006). "The type Ia supernova SNLS-03D3bb from a super-Chandrasekhar-mass white dwarf star". In: *Nature* 443, p. 308.

Iben, I. and A. Tutukov (1984). "Supernovae of type I as end products of the evolution of binaries with components of moderate initial mass (M not greater than about 9 solar masses)". In: 54, p. 335.

Jack, D., E. Baron, and P. H. Hauschildt (2015). "Identification of the feature that causes the I-band secondary maximum of a Type Ia supernova". In: *MNRAS* 449, pp. 3581–3586.

Jeffery, D. J., D. Branch, and E. Baron (2006). "On SN 2003fg: The Probable Super-Chandrasekhar-Mass SN Ia". In: *unpublished, ArXiv Astrophysics e-prints* astro-ph/0609804.

Jha, S., A. G. Riess, and R. P. Kirshner (2007). "Improved Distances to Type Ia Supernovae with Multicolor Light-Curve Shapes: MLCS2k2". In: 659, pp. 122–148.

Kasen, D. (2006). "Secondary Maximum in the Near-Infrared Light Curves of Type Ia Supernovae". In: 649, pp. 939–953.

— (2010). "Seeing the Collision of a Supernova with Its Companion Star". In: *ApJ* 708, pp. 1025–1031.

Kasen, D. and S. E. Woosley (2007). "On the Origin of the Type Ia Supernova Width-Luminosity Relation". In: 656, pp. 661–665.

Khokhlov, A., E. Müller, and P. Hoeflich (1993). "Light curves of Type Ia supernova models with different explosion mechanisms". In: 270, p. 223.

Kotak, R. et al. (2004). "On the nature of the circumstellar medium of the remarkable Type Ia/IIn supernova SN 2002ic". In: 354, pp. L13–L17.

Krisciunas, K. et al. (2011). "The Most Slowly Declining Type Ia Supernova 2001ay". In: *AJ* 142, p. 74.

Krisciunas, K., M. Phillips, and N. Suntzeff (2004). "Hubble Diagrams of Type Ia Supernovae in the Near Infrared". In: 602, p. L81.

Kushnir, D. et al. (2013). "Head-on Collisions of White Dwarfs in Triple Systems Could Explain Type Ia Supernovae". In: *ApJ* 778, L37, p. L37.

Lantz, B. et al. (2004). "SNIFS: a wideband integral field spectrograph with microlens arrays". In: *Optical Design and Engineering*. Ed. by L. Mazuray, P. J. Rogers, and R. Wartmann. Vol. 5249. Proc. SPIE, pp. 146–155.

Lexen, Ernst (2014). "Parameter sensitivity of synthetic spectra and light curves of Type Ia supernovae". PhD thesis. University of Hamburg.

Li, W. et al. (2001). "The Unique Type Ia Supernova 2000cx in NGC 524". In: 113, p. 1178.

— (2011). "Exclusion of a luminous red giant as a companion star to the progenitor of supernova SN 2011fe". In: *Nature* 480, pp. 348–350.

Li, W. et al. (2003). “SN 2002cx: The Most Peculiar Known Type Ia Supernova”. In: *PASP* 115, pp. 453–473.

Liu, J. et al. (2012). “On the Nature of the Progenitor of the Type Ia SN2011fe in M101”. In: *ApJ* 749, p. 141.

Livio, M. and J. Truran (1992). In: 389, p. 695.

Maeda, K., M. Kutsuna, and T. Shigeyama (2014). “Signatures of a Companion Star in Type Ia Supernovae”. In: *ApJ* 794, 37, p. 37.

Maeda, K. et al. (2010). “An asymmetric explosion as the origin of spectral evolution diversity in type Ia supernovae”. In: *Nature* 466, p. 82.

Maguire, K. et al. (2011). “PTF10ops - a subluminous, normal-width light curve Type Ia supernova in the middle of nowhere”. In: *MNRAS* 418, pp. 747–758.

Maoz, D. and F. Mannucci (2012). “Type-Ia Supernova Rates and the Progenitor Problem: A Review”. In: *PASA* 29, pp. 447–465.

Maoz, D., K. Sharon, and A. Gal-Yam (2010). “The Supernova Delay Time Distribution in Galaxy Clusters and Implications for Type-Ia Progenitors and Metal Enrichment”. In: *ApJ* 722, pp. 1879–1894.

Maund, J. R. et al. (2010). “The Unification of Asymmetry Signatures of Type Ia Supernovae”. In: 725, p. L167.

Nielsen, M. T. B. et al. (2014). “Upper limits on the luminosity of the progenitor of Type Ia supernova SN 2014J”. In: *MNRAS* 442, pp. 3400–3406.

Nomoto, K., F.-K. Thielemann, and K. Yokoi (1984). “Accreting White Dwarf Models of Type I Supernovae. III - Carbon deflagration supernovae”. In: 286, p. 644.

Nugent, P. et al. (1995). “Evidence for a Spectroscopic Sequence Among SNe Ia”. In: 455, p. L147.

Ofek, E. O. et al. (2007). “SN 2006gy: An Extremely Luminous Supernova in the Galaxy NGC 1260”. In: *ApJ* 659, pp. L13–L16.

Pakmor, R. et al. (2012). “Normal Type Ia Supernovae from Violent Mergers of White Dwarf Binaries”. In: *ApJ* 747, L10, p. L10.

Papadogiannakis, S. et al. (2019a). “Characterising the secondary maximum in the r-band for Type Ia Supernovae: Diagnostic for the ejecta mass”. In: *MNRAS* submitted astro-ph/1812.01438.

Papadogiannakis, S. et al. (2019b). “R-band light-curve properties of Type Ia supernovae from the (intermediate) Palomar Transient Factory”. In: *MNRAS* 483, pp. 5045–5076.

Parrent, J. T. et al. (2011). “A Study of Carbon Features in Type Ia Supernova Spectra”. In: *ApJ* 732, 30, p. 30.

Perlmutter, S. et al. (1999). “Measurements of Omega and Lambda from 42 High-Redshift Supernovae”. In: 517, p. 565.

Phillips, M. M. (1993). "The Absolute Magnitudes of Type Ia Supernovae". In: 413, p. L105.

Phillips, M. M. et al. (2007). "The Peculiar SN 2005hk: Do Some Type Ia Supernovae Explode as Deflagrations?" In: *PASP* 119, p. 360.

Phillips, M. M. et al. (1999). "The Reddening-Free Decline Rate Versus Luminosity Relationship for Type IA Supernovae". In: 118, pp. 1766–1776.

Piersanti, L. et al. (2003). "Carbon-Oxygen White Dwarfs Accreting CO-rich Matter. I. A Comparison between Rotating and Nonrotating Models". In: *ApJ* 583, pp. 885–901.

Rappaport, S., R. Di Stefano, and J. D. Smith (1992). In: 426, p. 692.

Raskin, C. et al. (2009). "On Type Ia supernovae from the collisions of two white dwarfs". In: *MNRAS* 399, pp. L156–L159.

Riess, A. et al. (1998). "Observational Evidence from Supernovae for an Accelerating Universe and a Cosmological Constant". In: 116, p. 1009.

Rigault, M. et al. (2015). "Confirmation of a Star Formation Bias in Type Ia Supernova Distances and its Effect on the Measurement of the Hubble Constant". In: *ApJ* 802, 20, p. 20.

Rosswog, S. et al. (2009). "Collisions of White Dwarfs as a New Progenitor Channel for Type Ia Supernovae". In: *ApJ* 705, pp. L128–L132.

Ruiter, A. J., K. Belczynski, and C. Fryer (2009). "Rates and Delay Times of Type Ia Supernovae". In: *ApJ* 699, pp. 2026–2036.

Ruiter, A. J. et al. (2014). "The effect of helium accretion efficiency on rates of Type Ia supernovae: double detonations in accreting binaries". In: *MNRAS* 440, pp. L101–L105.

Ruiz-Lapuente, P. (1997). "The quest for a supernova companion." In: *Science* 276, pp. 1813–1814.

Ruiz-Lapuente, P. et al. (2004). "The binary progenitor of Tycho Brahe's 1572 supernova". In: *Nature* 431, pp. 1069–1072.

Sato, Y. et al. (2015). "A Systematic Study of Carbon-Oxygen White Dwarf Mergers: Mass Combinations for Type Ia Supernovae". In: *ApJ* 807, 105, p. 105.

— (2016). "The Critical Mass Ratio of Double White Dwarf Binaries for Violent Merger-induced Type Ia Supernova Explosions". In: *ApJ* 821, 67, p. 67.

Scalzo, R. A. et al. (2010). "Nearby Supernova Factory Observations of SN 2007if: First Total Mass Measurement of a Super-Chandrasekhar-Mass Progenitor". In: 713, pp. 1073–1094.

Schlegel, D., D. Finkbeiner, and M. Davis (1998). "Maps of Dust Infrared Emission for Use in Estimation of Reddening and Cosmic Microwave Background Radiation Foregrounds". In: 500, p. 525.

Schwab, J., E. Quataert, and D. Kasen (2016). "The Evolution and Fate of Super-Chandrasekhar Mass White Dwarf Merger Remnants". In: *MNRAS* submitted, astro-ph/1606.02300.

Scolnic, D. M. et al. (2018). "The Complete Light-curve Sample of Spectroscopically Confirmed SNe Ia from Pan-STARRS1 and Cosmological Constraints from the Combined Pantheon Sample". In: *ApJ* 859, 101, p. 101.

Shen, K. J. et al. (2012). "The Long-Term Evolution of Double White Dwarf Mergers". In: 748, p. 35.

Silverman, J. M. et al. (2013). "SN 2000cx and SN 2013bh: extremely rare, nearly twin Type Ia supernovae". In: *MNRAS* 436, pp. 1225–1237.

Silverman, J. M. et al. (2011). "Fourteen months of observations of the possible super-Chandrasekhar mass Type Ia Supernova 2009dc". In: *MNRAS* 410, pp. 585–611.

Soker, N. (2013). "The Core-Degenerate Scenario for Type Ia Supernovae". In: *IAU Symposium*. Ed. by R. Di Stefano, M. Orio, and M. Moe. Vol. 281. IAU Symposium, pp. 72–75.

Spitzer, L. (1962). *Physics of Fully Ionized Gases*.

Sullivan, M. et al. (2010). "The dependence of Type Ia Supernovae luminosities on their host galaxies". In: p. 755.

Tanaka, M. et al. (2010). "Spectropolarimetry of Extremely Luminous Type Ia Supernova 2009dc: Nearly Spherical Explosion of Super-Chandrasekhar Mass White Dwarf". In: 714, pp. 1209–1216.

Tripp, R. (1998). "A Two-Parameter Luminosity Correction for Type Ia Supernovae". In: 331, pp. 815–820.

Tripp, R. and D. Branch (1999). "Determination of the Hubble Constant Using a Two Parameter Luminosity Correction for Type Ia Supernovae". In: 525, p. 209.

Wang, L., J. C. Wheeler, and P. Hoeflich (1997). "Polarimetry of the Type Ia Supernova SN 1996X". In: 476, p. L27.

Wang, L. et al. (2004). "On The Hydrogen Emission from the Type Ia Supernova 2002ic". In: 604, p. 53.

Wang, L. et al. (2005a). "Pre-Maximum Spectropolarimetry of the Type Ia SN 2004dt". In: in press, astro-ph/0409593.

Wang, X. et al. (2012). "Evidence for Type Ia Supernova Diversity from Ultraviolet Observations with the Hubble Space Telescope". In: *ApJ* 749, p. 126.

Wang, X. et al. (2005b). "A Novel Color Parameter as a Luminosity Calibrator for Type Ia Supernovae". In: in press, astro-ph/0501565.

Webbink, R. F. (1984). "Double white dwarfs as progenitors of R Coronae Borealis stars and Type I supernovae". In: 277, p. 355.

Wheeler, J. C. (2012). "White Dwarf/M Dwarf Binaries as Single Degenerate Progenitors of Type Ia Supernovae". In: 758, p. 123.

Whelan, J. and I. J. Iben (1973). "Binaries and Supernovae of Type I". In: 186, pp. 1007–1014.

Yoon, S.-C. and N. Langer (2004). "Presupernova Evolution of Accreting White Dwarfs with Rotation". In: 419, pp. 623–644.

— (2005). "On the Evolution of Rapidly Rotating Massive White Dwarfs Towards Supernovae or Collapses". In: 435, pp. 967–985.

Yoon, S.-C., P. Podsiadlowski, and S. Rosswog (2007). "Remnant evolution after a carbon-oxygen white dwarf merger". In: *MNRAS* 380, pp. 933–948.

Zhu, C. et al. (2013). "A Parameter-space Study of Carbon-Oxygen White Dwarf Mergers". In: *ApJ* 767, 164, p. 164.

Appendix A

Plateau Detection Algorithm and cuts

The visual inspection of our data shows that the plateau we are looking for happen around a SALT2 phase of $\sim 22.$ days after maximum light. We define a first phase window between SALT2 phase of 2 days after maximum light and SALT2 phase of 50 days after maximum light in which we will look for a time window with flat derivative.

We describe here the plateau detection algorithm on the specific example of the P6 plateau detection in the $6125 - 6200$ Å region.

The characteristic time of spectral evolution in SNe Ia is about 2 days around maximum light and tends to increase at later phases. A realistic and yet agnostic model of narrow light curves is a 4^{th} degree spline fit with equally spaced knots.

Since the data we are using can still present occasional glitches, we use the spline fitting to iteratively remove all the 2.5σ outliers, always making sure that at least four data points are left in the time window considered.

Once the spline fit converged, we proceed to the plateau detection. The plateau is defined as a region centered on phase p_c, enclosed between phase $p_{-10\%}$ and phase $p_{+10\%}$. Those phases are defined as the phases where the spline value $b(p)$ reaches $\pm 10\%$ of the spline value at phase p_c. We thus have $b(p_{+10\%}) = 1.1b(p_c)$ and $b(p_{-10\%}) = 0.9b(p_c)$.

We start the detection at $p_c = 22.3$ days, and iterate as follows:

1. Find the edges of the region enclosing $b(p_c) \pm 10\%$

2. Find the middle of this region, and call this the new center, updating the value of p_c

Since the SNe Ia narrow Light Curves in this time region are monotonic and smooth, this algorithm converges to a center and edges that comply both with $b(p_{\pm 10\%}) = b(p_c) \pm 0.1 b(p_c)$ and with $\frac{\Delta p}{2} = p_{+10\%} - p_c = p_c - p_{-10\%}$, where Δp is the width of the interval found.

Taylor expanding:

$$b(p_{+10\%}) = b(p_c) + (p_{+10\%} - p_c)\frac{db}{dx}(p_c) + \frac{1}{2}(p_{+10\%} - p_c)^2 \frac{d^2b}{dx^2}(p_c) + \dots \quad \text{(A.1)}$$

We can write:

$$\begin{aligned}(p_{+10\%} - p_c)\frac{db}{dx}(p_c) + \frac{1}{2}(p_{+10\%} - p_c)^2 \frac{d^2b}{dx^2}(p_c) &\simeq -(p_{-10\%} - p_c)\frac{db}{dx}(p_c) \\ &- \frac{1}{2}(p_{-10\%} - p_c)^2 \frac{d^2b}{dx^2}(p_c)\end{aligned} \quad \text{(A.2)}$$

From which it directly follows that:

$$\begin{aligned}\frac{\Delta p}{2}\frac{db}{dx}(p_c) + \frac{1}{2}(\frac{\Delta p}{2})^2 \frac{d^2b}{dx^2}(p_c) &\simeq \frac{\Delta p}{2}\frac{db}{dx}(p_c) \\ &- \frac{1}{2}(\frac{\Delta p}{2})^2 \frac{d^2b}{dx^2}(p_c)\end{aligned} \quad \text{(A.3)}$$

And finally:

$$(\frac{\Delta p}{2})^2 \frac{d^2 b}{dx^2}(p_c) = 0 \tag{A.4}$$

Therefore, our algorithm converges to a point where the second derivative is zero, and thus to the center of a region with constant derivative.

We thus have 3 possible behaviors, that we recognize as the result of our calculations:

1. Plateau SNe. They correspond to a fractional change (derivative of the spline at the center of the bin/value of the spline at the center of the bin) < 0.015.

   For those the derivative is constant, with a constant equal to 0.

2. Tilted plateau SNe: They correspond to a fractional change between 0.015 and ~ 0.03. They region of null curvature extends over several days.

3. No plateau SNe: they correspond to fractional changes larger than 0.03 and here we just find the spot where the derivative is minimal.

In practice it appears that these 3 groups form a continuum.

For robustness, we systematically throw away SNe Ia that end up with one of the edges of the plateau region right on the last data point available.

Figure A.1 and Figure A.2 show the way that the method works for the case of a plateau and no plateau, respectively. We see that the plateau center does indeed converge towards the point corresponding to a null second derivative. For the no plateau case, we see that the algorithm still remains stable. The cases

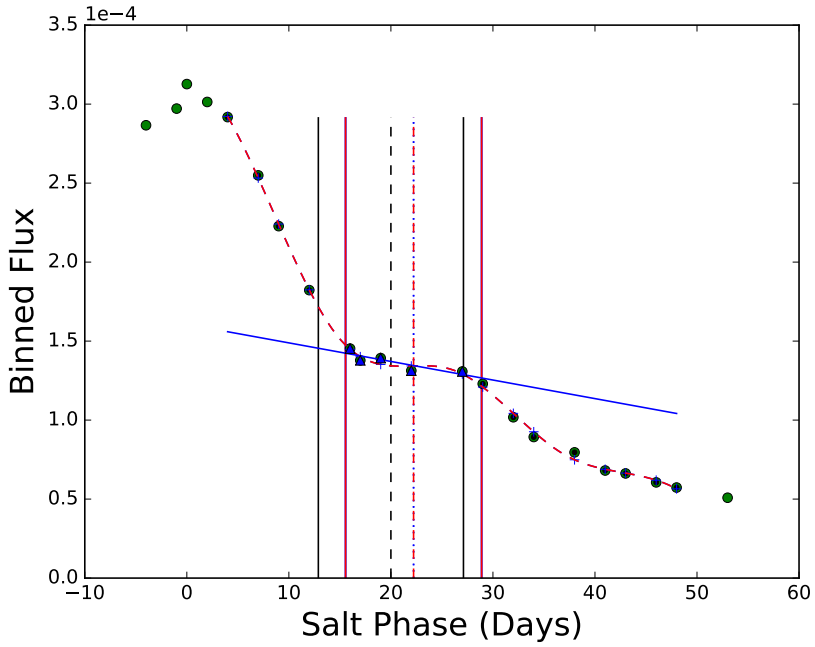


Figure A.1 The narrow band light curve of PFT09dnl, an example SN with a clear plateau. Green points: data not considered in the fit, Black points: included in initial fit, blue triangles (final fit), black solid lines: initial plateau region edges, red solid lines: final plateau region edges, blue solid lines: intermediate plateau region edges, black dashed line: initial plateau center, red dash-dot line: final plateau center, blue dotted line: intermediate plateau centers.

where the plateau region slides down and doesn't converge are very rare, and end up removed from the sample of SNe Ia considered because the latest plateau edge coincides with the last data point.

The choice of phase 22.3 days as the starting point of the algorithm was made by picking up the average value of all the plateau regions found after a first exploration. This choice simply ensures that the results of our algorithm are completely reproducible numerically. Differences between different versions of the analysis are also more tractable. Yet, we checked that using different starting points leads in the end to an average value of the plateau center phases of 22.3 days.

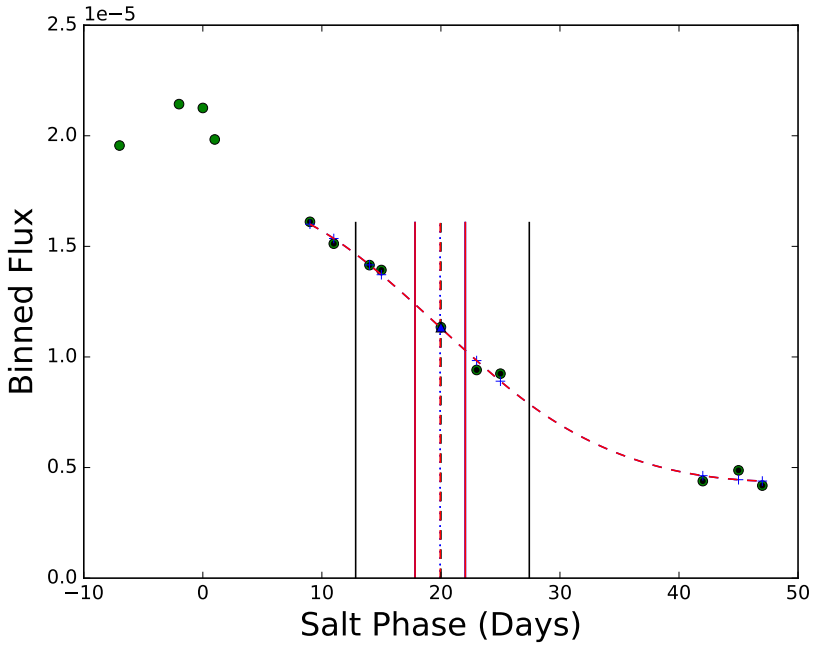


Figure A.2 The narrow band light curve of PTF11bnx, an example SN without a clear plateau. The point and line codes are the same as in Figure A.1.

Appendix B

3D SPH Particles to 2.5D Grid

This appendix has the code used to convert the 3D particle outputs into a FLASH readable uniform grid. The code is written in FORTRAN and parrallized with mpi. It reads in an input file containing the information from the SPH model in cgs units. Then it creates the uniform grid. Then at each grid point it finds all particles within two smoothing lengths and performs a weighted integral to find values for the density, temperature, and velocities at that point.

I would like to thank Robert Fisher and Suoquing Ji for their help figuring out how to convert the particle data from an SPH simulation into a grid format for use in FLASH. This code was created from example code shared with me by Suoquing Ji.

B.1 Main Code

```
PROGRAM cyl_main

use mpi

USE cyl_consts

IMPLICIT NONE

real*8, dimension(Nparticles)         :: r0,z0,dens0,temp0,vr0,vph0,vz0,smoothing_length0

integer, dimension(Nparticles)        :: number0

real*8, dimension(Mgrid_r+1)  :: rn

real*8, dimension(Mgrid_r)    :: rm

real*8, dimension(Mgrid_z+1)  :: zn

real*8, dimension(Mgrid_z)    :: zm


integer       :: i,j,k

logical       :: ug
```

```
real*8, dimension(1501,1002)  :: W0
real*8                :: dens,temp,vr,vph,vz  !quantities for each cell
real*8                :: dist1,dist2

real*8, dimension(Mgrid_r,Mgrid_z)        :: vr_dens

integer     :: ierr, rank, nproc, num
character(len=12) :: fname

      open(99, file='logfile.log')

      write(99,'(A,i7,A)')'INFO: Calculation start for ',Nparticles,' in cylindrical coordinates.'
      write(99,'(A,i4,A,i4,A)')'      Size: ',Mgrid_r,' x ',Mgrid_z,' cells'
      write(99,'(A,i9,A,i9,A)')'            r from 0 to ',redge,' km; z from (-) to (+) ',zedge,' km'
```

```
call cyl_particles(r0,z0,dens0,temp0,vr0,vph0,vz0,smoothing_length0,number0)

write(*,*) "particle data read"


call cyl_coords_ug(rn,zn,rm,zm)

write(*,*) "coordinate set up"


call cyl_kernel(W0)

write(*,*) "kernel function calculated"

!! calculates integral of kernel function from phi=0 to phi=2*pi


call mpi_init(ierr)


call mpi_comm_rank(MPI_COMM_WORLD,rank,ierr)

call mpi_comm_size(MPI_COMM_WORLD,nproc,ierr)
```

```
      num = 10 + rank
      write(fname,'("output",I2,".dat")')rank
      open(num,file = fname)

      i = Mgrid_r/nproc
      do j=(i*rank + 1), i*(rank+1)
         do k=1,Mgrid_z
              call cyl_density(rm(j),zm(k),dens,temp,vr,vph,vz,r0,z0,dens0,temp0,vr0,vph0,vz0,
                               smoothing_length0,number0,W0)
              write(num,2001)rm(j),zm(k),dens,temp,vr,vph,vz
2001          format(7e18.8)
         end do
      end do
```

```
        close(num)

        call mpi_finalize(ierr)

        write(99,'(A)')'INFO: Output files output_.dat'

        close(99)

END PROGRAM cyl_main
```


B.2 Constants File

```
MODULE cyl_consts

        IMPLICIT NONE

        integer, parameter      :: Nparticles=716800  !Number of particles - number of lines in input file


! Grid parameters: Mgrid_r, Mgrid_z: Number of gridpoints in r,z-direction


        integer, parameter      :: Mgrid_r = 8192

        integer, parameter      :: Mgrid_z = 8192


! Grid parameters: Domain sizes: redge, zedge: total grid size


        integer, parameter      :: redge=50000 !unit: km

        integer, parameter      :: zedge=40000
```

```
        integer, parameter      :: Mmonte=1000


!PARTICLE MASS
        real*8, parameter :: particle_mass=3.8850E+27 !unit: grams
        real*8, parameter :: pi=3.1415926535897
end module cyl_consts
```


B.3 Read in Particles

```
subroutine cyl_particles(r,z,dens,temp,vr,vph,vz,smoothing_length,number)
 USE cyl_consts
 implicit none
 real*8, intent(out), dimension(Nparticles)     :: r,z,dens,temp,vr,vph,vz,smoothing_length
```

```
integer, intent(out), dimension(Nparticles)    :: number


integer        :: i,j,l
real*8, dimension(Nparticles)  :: phi
real*8 :: x,y
real*8 :: vx,vy,mass,frac,dist
real*8 :: Ntyp,Nsigma


       open(11,file='grid_77_cyl.dat')
       open(99,file='log.log')

       read(11,*)      !number of particles
       read(11,*)      !dimension
```

```
        read(11,*)      !timestep


        i=0
        do j=1,(Nparticles)
                i=i+1
                read(11,*)x,y,z(i),dens(i),temp(i),vx,vy,vz(i),smoothing_length(i),mass
!1001           format(10E12.4)


                vz(i) = vz(i)
                r(i)=sqrt(x**2+y**2)
                if(y.lt.0) then
                        phi(i)=acos(x/r(i))
                else
                        phi(i)=2*pi-acos(x/r(i))
```

```
        end if


        if(r(i).eq.0) then
                vr(i)=0
                vph(i)=0
        else
                vr(i)=vx*x/r(i)+vy*y/r(i)
                vph(i)=-vx*y/r(i)+vy*x/r(i)
        end if


end do


return
```

```
end subroutine cyl_particles
```

B.4 Uniform Grid

```
subroutine cyl_coords_ug(rn,zn,rm,zm)
  USE cyl_consts
  IMPLICIT NONE
  real*8, intent(out), dimension(Mgrid_r+1)     :: rn
  real*8, intent(out), dimension(Mgrid_z+1)     :: zn
  real*8, intent(out), dimension(Mgrid_r)       :: rm
  real*8, intent(out), dimension(Mgrid_z)       :: zm !rm(i), zm(j) are centers of cell (i,j)
  integer, dimension(Mgrid_r+1) :: ri
  integer, dimension(Mgrid_z+1) :: zi
```

```
  real*8                :: drlin,dzlin
  integer       :: i,j,k,l,m,nr,nz,Mlin_r,Mlin_z


        open(99,file='log_coord.log')


        drlin=redge/Mgrid_r     !r-resolution [km]
        dzlin=2*zedge/Mgrid_z   !z-resolution [km]

!Setting up the grid in r-direction
        do i=1,(Mgrid_r+1)
                ri(i)=(i-1)*drlin
        end do


!Setting up the grid in z-direction
```

```
        zi(Mgrid_z/2+1)=0


        do i=1,(Mgrid_z/2)
                zi(Mgrid_z/2+1+i)=i*dzlin
                zi(Mgrid_z/2+1-i)=-i*dzlin
        end do


! (ri,zi) [km] --> (rn,zn) [cm]
        do i=1,(Mgrid_r+1)
                rn(i)=real(ri(i))*1.0e5
        end do
        do i=1,(Mgrid_z+1)
                zn(i)=real(zi(i))*1.0e5
        end do
```

```
! (xn,yn,zn): edges of gridcell --> (xm,ym,zm): center of gridcell
        do i=1,Mgrid_r
                rm(i)=(rn(i+1)+rn(i))/2
        end do

        do i=1,Mgrid_z
                zm(i)=(zn(i+1)+zn(i))/2
        end do

        write(99,'(A,2e12.4)')'Grid 0 = ',rm(1),zm(1)
        write(99,'(A,2e12.4)')'Grid last = ',rm(8192),zm(8192)
        write(99,'(A)')'INFO: Uniform Grid ready, starting to calculate quantities
                        for grid (cyl_coords finished)'
```

```
end subroutine cyl_coords_ug
```

B.5 Kernel

```
subroutine cyl_kernel(W)
!! Output is integral over 0->2pi, divided by 2pi
!!        not normalized!

  USE cyl_consts
  IMPLICIT NONE

  real*8, intent(out), dimension(1501,1002)     :: W
  integer       :: g,i,j,k,l,M1,M2
```

```
real*8        :: h,nu2D,mu,x,nu,phi1,phi2,qromb
real*8        :: integral
external      :: function1,function2

      open(99,file='logfile.log')

      do i=1,1501
              nu2D=real(i-1)/1000 !nu2D=sqrt((r-ri)**2+(z-zi)**2)/h  0 <= nu2D <= 2
              ! other kernel function: 0 <= nu <= 1.5
              ! function1 for   0 <= nu <= 0.5
              ! function2 for 0.5 <= nu <= 1.5
              do j=1,1002
                      if(j.eq.1) then
                              x=0
```

```
                if(nu2D.gt.0.5) then
                        phi1=0
                        phi2=pi
                else
                        phi1=pi
                        phi2=pi
                end if
        else if(nu2D.gt.0.5) then
                phi1=0
                x=exp(-10+0.01655*(j-2))
                if(((2.25-nu2D*nu2D)/x).ge.2) then
                        phi2=pi
                else
                        phi2=acos(1-(2.25-nu2D*nu2D)/x)
```

```
        end if
    else
        x=exp(-10+0.01655*(j-2))
        if(((2.25-nu2D*nu2D)/x).ge.2) then
            phi2=pi
            if(((0.25-nu2D*nu2D)/x).ge.2) then
                phi1=pi
            else
                phi1=acos(1-(0.25-nu2D*nu2D)/x)
            end if
        else
            phi1=acos(1-(0.25-nu2D*nu2D)/x)
            phi2=acos(1-(2.25-nu2D*nu2D)/x)
        end if
```

```
                        end if


                        integral=0.
                        integral=qromb(function2,phi1,phi2,nu2D,x)


                        if(.not.((integral.le.0).or.(integral.ge.0)))
                           write(99,'(A,3i7,f8.3)')'--ERROR-- (cyl_kernel) integral=NaN',i,j,l
                        W(i,j)=integral
                    end do
                end do


end subroutine cyl_kernel
```

B.6 Calculating Grid Values

```
subroutine cyl_density(r,z,dens,temp,vr,vph,vz,r0,z0,dens0,temp0,vr0,vph0,vz0,smoothing_length0,number0,W)
 USE cyl_consts
 IMPLICIT NONE

 real*8, intent(in)                              :: r,z
 real*8, intent(in), dimension(Nparticles)       :: r0,z0,dens0,temp0,vr0,vph0,vz0,smoothing_length0
 integer, intent(in), dimension(Nparticles)      :: number0
 real*8, intent(in), dimension(1501,1002)        :: W
 real*8, intent(out)                             :: dens,temp,vr,vph,vz

 real*8                                  :: dist2D,length
 real*8                                  :: rtemp
 integer                          :: i,j,k
```

```
real*8 :: W_anal,myW,help


      open(99,file='log_dens.log')
      dens=0
      temp=0


      vr=0
      vph=0
      vz=0


      do i=1,Nparticles
              length=2*smoothing_length0(i)
              if(length.ne.0) then
                      dist2D=((r-r0(i))*(r-r0(i))+(z-z0(i))*(z-z0(i)))/(length*length)
```

```
if(dist2D.gt.2.25) then
        CYCLE
else
        rtemp=2*r*r0(i)/(length*length)
        call analytic_integration(1.d0*sqrt(dist2D),rtemp,W_anal)
        myW=W_anal
        myW=myW/pi
        help = particle_mass*myW/(4./3*pi*length*length*length)
        dens=dens+help
        temp=temp+temp0(i)*help/dens0(i)
        vr=vr+vr0(i)*help/dens0(i)
        vph=vph+vph0(i)*help/dens0(i)
        vz=vz+vz0(i)*help/dens0(i)
end if
```

```
                else

                        write(99,'(A,i7)')'--ERROR-- smoothing_length = 0 for particle # ',i

                end if

        end do

end subroutine cyl_density


subroutine analytic_integration(nu2D,x,W_anal)

        use cyl_consts, ONLY : pi

        implicit none

        real*8,intent(in) :: nu2D,x

        real*8,intent(out) :: W_anal


        real*8 :: sp1,sp2,sp3

        real*8 :: phi1,phi2,phi3,phi4,W1,W2,qromb
```

```
external      :: function2


phi1=-1.

phi2=-1.

phi3=-1.

phi4=-1.

W_anal=0.d0


if(x==0) then

 if(nu2D>0.5) then

  phi1=0.

  phi2=0.

  if(nu2D<=1.5) then

   phi3=0.
```

```
    phi4=pi
   else
    phi3=0.
    phi4=0.
   end if
  else
   phi1=0.
   phi2=pi
   phi3=0.
   phi4=0.
  end if
 else !if(nu2D>0.5) then
  sp1=1+nu2D*nu2D/x
  sp2=1-(0.25-nu2D*nu2D)/x
```

```
sp3=1-(2.25-nu2D*nu2D)/x
if(sp2>1. .or. sp1<-1) then
 phi1=0.
 phi2=0.
else if(sp2>=-1. .and. sp1<=1.) then
 phi1=acos(sp1)
 phi2=acos(sp2)
else if(sp2>=-1. .and. sp1>1.) then
 phi1=0.
 phi2=acos(sp2)
else if(sp2<-1 .and. sp1<=1.) then
 phi1=acos(sp1)
 phi2=pi
else if(sp1>=1. .and. sp2<=-1.) then
```

```
 phi1=0.
 phi2=pi
end if
if(sp3>1. .or. sp2<-1.) then
 phi3=0.
 phi4=0.
else if(sp3>=-1. .and. sp2<=1.) then
 phi3=acos(sp2)
 phi4=acos(sp3)
else if(sp2<=1 .and. sp3<-1.) then
 phi3=acos(sp2)
 phi4=pi
else if(sp2>1. .and. sp3>=-1.) then
 phi3=0.
```

```
  phi4=acos(sp3)
 else if(sp2>=1. .and. sp3<=-1.) then
  phi3=0.
  phi4=pi
 end if
end if

if(min(phi1,phi2,phi3,phi4)==-1.) then
 write(*,*) "Error in assigning phi"
 write(*,*) phi1,phi2,phi3,phi4
 write(*,*) nu2D,x
 write(*,*) sp1,sp2,sp3
end if
```

```
        call fun1_anal(nu2D,x,phi1,phi2,W1)

        W_anal=W1+qromb(function2,phi3,phi4,nu2D,x)

end subroutine

subroutine fun1_anal(nu2D,x,phi1,phi2,W1)
        use cyl_consts, ONLY : pi
        implicit none
        real*8,intent(in) :: nu2D,x,phi1,phi2
        real*8,intent(out) :: W1
        W1=(-(8.*phi2*nu2D*nu2D)/3.+2*phi2-(8.*phi2*x)/3.+(8.*x*sin(phi2))/3.)&
           -(-(8.*phi1*nu2D*nu2D)/3.+2*phi1-(8.*phi1*x)/3.+(8.*x*sin(phi1))/3.)
end subroutine
```

```
FUNCTION function2(phi,nu2D,x)
  real*8, dimension(:), intent(in)      :: phi
  real*8, intent(in)                    :: nu2D,x
  real*8, dimension(size(phi))          :: mu,function2
        mu=sqrt(nu2D*nu2D+x*(1-cos(phi)))
        function2=4./3*(3./2-mu)*(3./2-mu)
        ! Ji: we use incomplete form
        !function2=-4.0*mu
END FUNCTION function2
```

Appendix C

FLASH Parameters

FLASH 4.4 is a modular code so a configuration file is necessary to specify the physics models needed in a particular model. These modules are supplied to the code in a Config file. That file also contains other parameters and the data file needed to initialized the grid. The Config file used for these simulations can be found in C.1.

The code also requires a file with model specific parameters. This file is called flash.par and the initial file can be found in C.2. This file specifies if the run of the simulation is a restart or not. The values in this file can be changed between restarts of the model. The values pertaining to restarts and plot files are frequently changed to continue a run. In this file the cfl condition needed to be adjusted to account for the changing sound speeds in the model.

To initialize the grid for our models we also needed to have code to initialize the grid. These codes can be found in C.4 and C.5. In these codes the grid values are set to match those supplied in the data file and the magnetic field is set at each grid point.

The setup for this simulation was then run with flags for a uniform grid, cylindrical coordinates, and appropriate values of nxb and nyb for the number of nodes used.

I would like to thank Robert Fisher and Suoqing Ji for help and sample code for initializing and running FLASH.

C.1 Config

```
# congifuration file for post merger viscous evolution

REQUIRES Driver

REQUIRES Grid


REQUIRES physics/Hydro/HydroMain/unsplit/MHD_StaggeredMesh

REQUIRES physics/Eos/EosMain/Helmholtz

REQUIRES physics/Gravity/GravityMain/Poisson


#for species to pass to helm eos

REQUIRES physics/sourceTerms/Burn/BurnMain/nuclearBurn

REQUIRES Simulation/SimulationComposition

REQUIRES physics/materialProperties/Viscosity/ViscosityMain/Spitzer
```

```
PARAMETER      rstar              REAL     2.1269e8

DATAFILES wd_merger.dat


PARAMETER xc12            REAL    0.4 [0.0 to 1.0]

PARAMETER xo16            REAL    0.6 [0.0 to 1.0]


PARAMETER mass_thres REAL 1e26

MASS_SCALAR jrey

MASS_SCALAR jmag
```


C.2 flash.par

```
# FLASH file for a WD merger viscous evolution in 2.5D cylindrical

#
```

```
# Example setup call
#  ./setup Merger -2d +cylindrical -nxb=512 -nyb=512 -auto +usm +ug
#


##################
# model parameters
##################
```



```
########################
# Viscosity information
########################

useViscosity = .true.
```

```
viscSuppressFactor = 1.0


########################
# physical domain
########################


geometry = "cylindrical"

xmin =  0.0
xmax =  50000.e5
ymin =  -40000.e5
ymax =   40000.e5
```

```
quadrant = .false.

octant = .false.


nblockx = 4

nblocky = 4


# boundary conditions for the whole star

xl_boundary_type = "AXISYMMETRIC"

xr_boundary_type = "diode"


yl_boundary_type = "diode"

yr_boundary_type = "diode"
```

```
grav_boundary_type = "isolated"


#########

# physics

#########



mpole_lmax = 10

mpole_dumpMoments       = .false.


# eos

eos_coulombMult = 0.e0


#####
```

```
# I/O
#####

# names of files
basenm = "merger_"

# for starting a new run
restart = .false.
# for a restart
#restart                  = .true.

checkpointFileNumber    = 0  #0 if new or restart number
plotFileNumber = 0  #0 or the number of the last plot +1
```

```
# dump checkpoint files every trstrt seconds

checkpointFileIntervalTime = 3

# dump plot files every tplot seconds

plotFileIntervalTime = 1


# go for nend steps or tmax seconds, whichever comes first

nend = 999999999

tmax = 2.e4

wall_clock_checkpoint   = 857000000.


# variables for plotting

plot_var_1 = "dens"

plot_var_2 = "temp"

plot_var_3 = "pres"
```

```
plot_var_4 = "velx"

plot_var_5 = "vely"

plot_var_6 = "velz"

plot_var_7 = "gpot"

plot_var_8 = "eint"

plot_var_9 = "magx"

plot_var_10 = "magy"

plot_var_11             = "magz"

plot_var_12             = "magp"

plot_var_13             = "gamc"


corners = .false.


##########
```

```
# timestep

##########


# CFL limit

cfl = 0.30


# initial and minimum

dtinit = 1.e-7

dtmin = 1.e-8

dtmax = 1.e-1

tstep_change_factor     = 1.1e0



#############
```

```
# miscellaneous
#############

# floor values

small = 1.e-100
smalle = 1.e-100
smallt = 1.e7
smallu = 1.e-10
smlrho = 2.e-2
smallp = 1.e15
smallx = 1.e-100

# misc (may be the same as defaults)
```

```
use_steepening    = .false.
convertToConsvdForMeshCalls    = .false.
use_cma_advection       = .false.


memory_stat_freq         = 1


### set the composition
xc12 = 0.4
xo16 = 0.6


order                    = 3
use_flattening           = .true.
use_cma_flattening       = .true.
shockDetect              = .false.
```

```
slopeLimiter            = minmod
RiemannSolver           = Roe
conserveAngMom          = .true.
mass_thres              = 4.e20

iprocs                  = 16
jprocs                  = 16

eintSwitch              = 1.e50

use_avisc               = .true.

UnitSystem              = "CGS"
killdivb                = .true.
```

```
prolMethod                = balsara_prol
```

C.3 Simulation_data.F90

```
!!****if* source/Simulation/SimulationMain/Merger
!!
!! NAME
!!
!!  Simulation_data
!!
!!
!! SYNOPSIS
!!
!!  use Simulation_data
```

```
!!
!! DESCRIPTION
!!
!!  Data module for WD merger viscous evolution
!!
!! run time parameters:
!!
!!
!!***

Module Simulation_data
#include "Flash.h"
#include "Eos.h"
```

```
real, save  :: vr0, vphi0, vz0

real, save  :: sim_smallx, sim_xc12, sim_xo16

real, save  :: sim_rstar

real, save    :: sim_cosPhi, sim_sinPhi, sim_cosTheta, sim_sinTheta

real, save    :: sim_drMin

real, save    :: sim_sdimI

integer, save    :: sim_npoints

real, save :: sim_cFrac, sim_deltaeNse

!real, save :: sim_densU, sim_presU, sim_tempU, sim_eintU

!real, save :: sim_densB, sim_presB, sim_tempB, sim_eintB

real, save :: sim_qbarU, sim_qbarNse, sim_qbarB

real, save :: sim_v0, sim_p0, sim_p1, sim_p2

real, save :: sim_refineDDens, sim_refineDtVel, sim_refineXtVel, sim_refineDphi1

real, save :: sim_refineXphi1, sim_refineSphi1, sim_refineXenuc
```

```
real, save :: sim_refineDensMin, sim_refineDensMax, sim_refineUniDens

real, save :: sim_refineUniDx, sim_refineUniRadius, sim_refineMaxRadius

real, save :: sim_refine_inner_dens_min, sim_refine_inner_dens_dx

real, save :: sim_GCDRefineAngle, sim_GCDRefineMaxRadius

real, save :: sim_GCDFocusTime, sim_GCDFocusAngle, sim_GCDFocusMaxRadius

real, save :: sim_GCDFocusMinRadius, sim_GCDFocusDx

logical, save :: sim_refine_allphi

integer, save :: sim_lrefineMin, sim_lrefineMax


real, save :: sim_smallU

integer, save :: sim_gridGeom

logical, save :: sim_restart


real,dimension(EOS_NUM), save :: sim_eosData, sim_eosDataU, sim_eosDataB, sim_eosDataNse
```

```
  integer,parameter                          :: SIM_NMAX=8192
  real,dimension(SIM_NMAX), save             :: sim_2dR, sim_2dZ
  real,dimension(SIM_NMAX,SIM_NMAX), save :: sim_2dDens, sim_2dTemp, sim_2dvr, sim_2dvphi,
                                                sim_2dvz


  real,save :: mass_thres
  logical,save :: sim_killdivb


  end Module Simulation_data
```


C.4 Simulation_init.F90

```
!!****if* source/Simulation/SimulationMain/Merger
!!
```

```
!! NAME
!!
!!  Simulation_init
!!
!!
!! SYNOPSIS
!!
!!  Simulation_init(integer(IN) :: myPE)
!!
!! DESCRIPTION
!!
!!  Provide initial conditions for a WD merger viscous evolution
!!
!! ARGUMENTS
```

```
!!
!!    myPE - my processor number
!!
!! PARAMETERS
!!
!!
!!***
```



```
subroutine Simulation_init()

  use Simulation_data

  use RuntimeParameters_interface, ONLY : RuntimeParameters_get
  use Logfile_interface, ONLY : Logfile_stampMessage
```

```
  use Grid_interface, ONLY : Grid_getGeometry

  use Driver_interface, ONLY : Driver_abortFlash, Driver_getMype


  implicit none

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

  integer :: myPE  ! Ji
  real :: v0, p0, p1, p2
  real :: dens_u,pres_u, temp_u, eint_u, uint, pres, laminarWidth
```

```
real, save :: pi

real :: alpha

integer    :: jdens, jtemp

integer, parameter ::  NVARS_MODEL = 24

real,dimension( NVARS_MODEL)  :: tv

character (len=4)       :: unklabels(NVARS_MODEL)

character (len=40) :: infile, string

integer :: inlen

real :: fronts, deltae

integer :: op, r_kat, z_kat

real :: dyi, deltaeNse

real :: r_dist, z_dist


integer :: i,j
```

```
uint = 0.e0

pres = 0.e0


call Driver_getMyPE(GLOBAL_COMM,myPE)


call RuntimeParameters_get('smallu',     sim_smallU)


call RuntimeParameters_get('xc12', sim_xc12)

call RuntimeParameters_get('xo16', sim_xo16)

call RuntimeParameters_get('smallx', sim_smallx)

call RuntimeParameters_get('killdivb', sim_killdivb)


call RuntimeParameters_get('rstar', sim_rstar)
```

```
call RuntimeParameters_get('refine_ddens',          sim_refineDDens)
call RuntimeParameters_get('refine_dtvel',          sim_refineDtVel)
call RuntimeParameters_get('refine_xtvel',          sim_refineXtVel)
call RuntimeParameters_get('refine_dphi1',          sim_refineDphi1)
call RuntimeParameters_get('refine_xphi1',          sim_refineXphi1)
call RuntimeParameters_get('refine_sphi1',          sim_refineSphi1)
call RuntimeParameters_get('refine_xenuc',          sim_refineXenuc)

call RuntimeParameters_get('refine_allphi',          sim_refine_allphi)

call RuntimeParameters_get('refine_dens_min',       sim_refineDensMin)
call RuntimeParameters_get('refine_dens_max',       sim_refineDensMax)
call RuntimeParameters_get('refine_uni_dens',       sim_refineUniDens)
call RuntimeParameters_get('refine_uni_dx',         sim_refineUniDx)
```

```
call RuntimeParameters_get('refine_uni_radius',      sim_refineUniRadius)
call RuntimeParameters_get('refine_max_radius',      sim_refineMaxRadius)

call RuntimeParameters_get('refine_inner_dens_min',  sim_refine_inner_dens_min)
call RuntimeParameters_get('refine_inner_dens_dx',   sim_refine_inner_dens_dx)

call RuntimeParameters_get('gcd_refine_max_radius',  sim_GCDRefineMaxRadius)
call RuntimeParameters_get('gcd_refine_angle',       sim_GCDRefineAngle)
sim_GCDRefineAngle = PI/180.0*sim_GCDRefineAngle

call RuntimeParameters_get('gcd_focus_time',        sim_GCDFocusTime)
call RuntimeParameters_get('gcd_focus_max_radius',  sim_GCDFocusMaxRadius)
call RuntimeParameters_get('gcd_focus_min_radius',  sim_GCDFocusMinRadius)
call RuntimeParameters_get('gcd_focus_angle',       sim_GCDFocusAngle)
```

```
call RuntimeParameters_get('gcd_focus_dx',          sim_GCDFocusDx)
sim_GCDFocusAngle = PI/180.0*sim_GCDFocusAngle



call RuntimeParameters_get('mass_thres', mass_thres) ! Ji


call RuntimeParameters_get('restart',   sim_restart)


call Grid_getGeometry(sim_gridGeom)


!-----------------------------------
!  READ INITIAL PROFILE FROM DATA FILE
!-----------------------------------
! Initial conditions given by WD-merger: Sato et. al.
```

```
!..open the data file


infile = 'wd_merger.dat'

!nvar_ = 24


inlen  = index(infile,' ') - 1


open(unit=2,file=infile,status='old')


sim_npoints = 8192 ! same number of points for r- and z-direction


if (MyPE == MASTER_PE) then

   write(*,'(1x,a,a,a,i4)')'  opened file = ',infile(1:inlen), &

        ' expected number of data points =',sim_npoints
```

```
end if


do i=1,sim_npoints !grid size of initial data, 512x512 cells
   do j=1,sim_npoints
      read(2,'(7e18.8)')sim_2dR(i),sim_2dZ(j),sim_2dDens(i,j),sim_2dTemp(i,j),
           sim_2dvr(i,j),sim_2dvphi(i,j),sim_2dvz(i,j)
        !! UNITS: cgs


   end do
end do


close(unit=2)


if (MyPE == MASTER_PE) then
```

```
     write(*,'(1x,a,a,a,i4)')' done reading file = ',infile(1:inlen), &
          ' number of data points =',sim_npoints
  end if

end subroutine Simulation_init
```

C.5 Simulation_initBlock.F90

```
!!****if* source/Simulation/SimulationMain/Merger
!!
!! NAME
!!
!!  Simulation_initBlock
!!
```

```
!!
!! SYNOPSIS
!!
!!  Simulation_initBlock(integer(IN) :: blockID,
!!                       integer(IN) :: myPE  )
!!
!! DESCRIPTION
!!
!!  Provide initial conditions for WD merger viscous evolution
!!
!! ARGUMENTS
!!
!!  blockID - my block number
!!  myPE    - local processor number
```

```
!!
!!***
subroutine Simulation_initBlock(blockID)

  use Simulation_data
  use Driver_interface, ONLY : Driver_abortFlash

  use Grid_interface, ONLY : Grid_getBlkIndexLimits, &
    Grid_getCellCoords, Grid_getDeltas, Grid_putRowData,&
    Grid_getBlkPtr,Grid_releaseBlkPtr  ! Ji
  use Eos_interface, ONLY : Eos
```

```
  implicit none
#include "constants.h"
#include "Flash.h"
#include "Eos.h"

  integer, intent(IN) :: blockID

  real, allocatable, dimension(:) :: xCenter, yCenter, zCenter
  real, allocatable, dimension(:) :: dx, dy, dz
  real, allocatable, dimension(:) :: rho, p, e, eint, t, game, gamc, &
       vx, vy, vz, f, trcr, ye_row, sumy, qbar_row
  logical                 :: contains_match
  real                    :: tsim_dens_b, tpres_b, ttemp_b, teint_b
```

```
real, allocatable, dimension(:,:) :: temp_var

real,allocatable, dimension(:) :: xvector, yvector, zvector

real,allocatable, dimension(:) :: velxvector, velyvector, velzvector

real,allocatable, dimension(:) :: densvector, presvector, &

     tempvector, enervector


real                    :: vr_temp, vphi_temp, vz_temp

real                    :: dens,   pres,   temp,   ener,   phi

real                    :: r, a, vr, ksi, r0 !,z <-- this is doubly defined

real                    :: fronts, deltae, alpha


integer                 :: i, j, k, n, err, ign

integer                 :: istat

character(len=16)       :: ionname
```

```
integer                 :: vecLen

real                    :: tdens_b

real :: specang


integer ::  op, r_kat, z_kat

real    ::  r_dist, z_dist, dyi, testr, testz, dist


real :: pel, eel, sel, uint

real :: dpt, dpd, ded, det, c_v, c_p, cs1, gamma, xalfa, xxni, xxne, xxnp

real :: entropy, dst, dsd, abar, zbar

real :: ye, sumyi, qbar


real, dimension(EOS_NUM) :: eosData

logical,dimension(EOS_VARS+1:EOS_NUM) :: eosMask
```

```
integer, save  :: i_seed

real           :: x_seed

integer, dimension(MDIM) :: pos

real,dimension(MDIM) :: del

integer,dimension(LOW:HIGH,MDIM)::blkLimits,blkLimitsGC

integer :: iSize, jSize, kSize

integer :: iSizeGC, jSizeGC, kSizeGC

integer :: ilo, ihi


real, allocatable, dimension(:) :: magxZone,magyZone,magzZone

real, pointer, dimension(:,:,:,:) :: solnData,facexData,faceyData,facezData,scrch_Ptr

real :: beta

real :: magAmplitude, B0, RB ! Initial Amplitude of the magnetic Field

real :: rhoMinMag
```

```
  real :: magPotXlow, magPotXhigh, magPotYlow, magPotYhigh


  real :: partialR_rey,partialZ_rey,partialR_mag,partialZ_mag


  real, allocatable,dimension(:) :: xCoord,xCoordL,xCoordR,&
                                    yCoord,yCoordL,yCoordR,&
                                    zCoord,zCoordL,zCoordR
  integer :: sizeX, sizeY, sizeZ
  real :: x1,x2,x3
  real :: xx,yy,zz
  real :: A_phi


#ifdef FIXEDBLOCKSIZE
  real, dimension(GRID_IHI_GC+1,GRID_JHI_GC+1,GRID_KHI_GC+1) :: Az
```

```
#else
  real, allocatable, dimension(:,:,:) :: Az
#endif

  real, dimension(SPECIES_BEGIN:SPECIES_END) ::  massFraction

  massFraction(:) = sim_smallx
  massFraction(C12_SPEC) = sim_xc12
  massFraction(O16_SPEC) = sim_xo16

  abar = sim_xc12 *12. + sim_xo16 * 16.
  zbar = sim_xc12 *6. + sim_xo16 * 8.

  call Grid_getBlkPtr(blockID,solnData,CENTER)
```

```
#if NFACE_VARS > 0
  if (sim_killdivb) then
    call Grid_getBlkPtr(blockID,facexData,FACEX)
    call Grid_getBlkPtr(blockID,faceyData,FACEY)
    if (NDIM == 3) call Grid_getBlkPtr(blockID,facezData,FACEZ)
  endif
#endif


  call Grid_getBlkPtr(blockID,scrch_Ptr,SCRATCH)


  call Grid_getBlkIndexLimits(blockID,blkLimits,blkLimitsGC)
  iSizeGC = blkLimitsGC(HIGH,IAXIS)-blkLimitsGC(LOW,IAXIS)+1
  jSizeGC = blkLimitsGC(HIGH,JAXIS)-blkLimitsGC(LOW,JAXIS)+1
  kSizeGC = blkLimitsGC(HIGH,KAXIS)-blkLimitsGC(LOW,KAXIS)+1
```

```
iSize = blkLimits(HIGH,IAXIS)-blkLimits(LOW,IAXIS)+1

jSize = blkLimits(HIGH,JAXIS)-blkLimits(LOW,JAXIS)+1

kSize = blkLimits(HIGH,KAXIS)-blkLimits(LOW,KAXIS)+1


ilo = blkLimits(LOW,IAXIS)

ihi = blkLimits(HIGH,IAXIS)


allocate(xCenter(iSizeGC),STAT=istat)

  if (istat /= 0) call Driver_abortFlash("Cannot allocate xCenter in Simulation_initBlock")

allocate(yCenter(jSizeGC),STAT=istat)

  if (istat /= 0) call Driver_abortFlash("Cannot allocate yCenter in Simulation_initBlock")

allocate(zCenter(kSizeGC),STAT=istat)

  if (istat /= 0) call Driver_abortFlash("Cannot allocate zCenter in Simulation_initBlock")

allocate(dx(iSizeGC),STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot allocate dx in Simulation_initBlock")
allocate(dy(jSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate dy in Simulation_initBlock")
allocate(dz(kSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate dz in Simulation_initBlock")
allocate(rho(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate rho in Simulation_initBlock")
allocate(p(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate p in Simulation_initBlock")
allocate(e(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate e in Simulation_initBlock")
allocate(eint(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate eint in Simulation_initBlock")
allocate(t(iSizeGC),STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot allocate t in Simulation_initBlock")
allocate(game(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate game in Simulation_initBlock")
allocate(gamc(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate gamc in Simulation_initBlock")
allocate(vx(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate vx in Simulation_initBlock")
allocate(vy(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate vy in Simulation_initBlock")
allocate(vz(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate vz in Simulation_initBlock")
allocate(f(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate f in Simulation_initBlock")
allocate(ye_row(iSizeGC),STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot allocate ye_row in Simulation_initBlock")
allocate(sumy(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate sumy in Simulation_initBlock")
allocate(qbar_row(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate qbar_row in Simulation_initBlock")
allocate(trcr(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate trcr in Simulation_initBlock")
allocate(xvector(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate xvector in Simulation_initBlock")
allocate(yvector(jSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate yvector in Simulation_initBlock")
allocate(zvector(kSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate zvector in Simulation_initBlock")
allocate(velxvector(iSizeGC),STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot allocate velxvector in Simulation_initBlock")
allocate(velyvector(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate velyvector in Simulation_initBlock")
allocate(velzvector(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate velzvector in Simulation_initBlock")
allocate(densvector(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate densvector in Simulation_initBlock")
allocate(presvector(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate presvector in Simulation_initBlock")
allocate(tempvector(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate tempvector in Simulation_initBlock")
allocate(enervector(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate enervector in Simulation_initBlock")
allocate(magxZone(iSizeGC),STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot allocate magxvector in Simulation_initBlock")
allocate(magyZone(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate magyvector in Simulation_initBlock")
allocate(magzZone(iSizeGC),STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot allocate magzvector in Simulation_initBlock")


!Set magnetic field paramters
B0 = 1e6
RB = 1e9
magAmplitude = B0*RB**3
rhoMinMag = 2.0*1e-1


xCenter(:) = 0.e0
yCenter(:) = 0.e0
```

```
zCenter(:) = 0.e0


dx(:) = 0.e0

dy(:) = 0.e0

dz(:) = 0.e0

call Grid_getDeltas(blockId, del)


if(NDIM > 2) then

   call Grid_getCellCoords(KAXIS,blockID, CENTER, .true.,zCenter,kSizeGC)

   dz=del(KAXIS)

end if


if(NDIM > 1) then

   call Grid_getCellCoords(JAXIS,blockID, CENTER, .true.,yCenter,jSizeGC)
```

```
   dy=del(JAXIS)
end if

call Grid_getCellCoords(IAXIS,blockID, CENTER, .true.,xCenter,iSizeGC)
dx=del(IAXIS)

sizeX = blkLimitsGC(HIGH,IAXIS)-blkLimitsGC(LOW,IAXIS)+1
sizeY = blkLimitsGC(HIGH,JAXIS)-blkLimitsGC(LOW,JAXIS)+1
sizeZ = blkLimitsGC(HIGH,KAXIS)-blkLimitsGC(LOW,KAXIS)+1

allocate(xCoord(sizeX), stat=istat)
allocate(xCoordL(sizeX),stat=istat)
allocate(xCoordR(sizeX),stat=istat)
```

```
allocate(yCoord(sizeY), stat=istat)

allocate(yCoordL(sizeY),stat=istat)

allocate(yCoordR(sizeY),stat=istat)


allocate(zCoord(sizeZ), stat=istat)

allocate(zCoordL(sizeZ),stat=istat)

allocate(zCoordR(sizeZ),stat=istat)
```



```
xCoord  = 0.0

xCoordL = 0.0

xCoordR = 0.0


yCoord  = 0.0

yCoordL = 0.0
```

```
  yCoordR = 0.0


  zCoord  = 0.0
  zCoordL = 0.0
  zCoordR = 0.0


#ifndef FIXEDBLOCKSIZE
     allocate(Az(sizeX+1,sizeY+1,1),stat=istat)
#endif


  if (NDIM >= 2) then
     call Grid_getCellCoords(JAXIS,blockId,CENTER,    .true.,yCoord, sizeY)
     call Grid_getCellCoords(JAXIS,blockId,LEFT_EDGE, .true.,yCoordL,sizeY)
     call Grid_getCellCoords(JAXIS,blockId,RIGHT_EDGE,.true.,yCoordR,sizeY)
```

```
endif


call Grid_getCellCoords(IAXIS,blockId,CENTER,    .true.,xCoord, sizeX)

call Grid_getCellCoords(IAXIS,blockId,LEFT_EDGE, .true.,xCoordL,sizeX)

call Grid_getCellCoords(IAXIS,blockId,RIGHT_EDGE,.true.,xCoordR,sizeX)


Az = 0.


t  (:) = 0.e0

rho(:) = 0.e0

p  (:) = 0.e0

e  (:) = 0.e0

vx (:) = 0.e0

vy (:) = 0.e0
```

```
vz (:) = 0.e0

ye_row (:) = 0.e0

sumy (:) = 0.e0

qbar_row (:) = 0.e0


eosMask=.false.


pos(IAXIS)=blkLimits(LOW,IAXIS)

do k = blkLimitsGC(LOW,KAXIS),blkLimitsGC(HIGH,KAXIS)

   pos(KAXIS)=k

   do j = blkLimitsGC(LOW,JAXIS),blkLimitsGC(HIGH,JAXIS)

      pos(JAXIS)=j

      z_dist=yCenter(j)
```

```
do i = blkLimitsGC(LOW,IAXIS),blkLimitsGC(HIGH,IAXIS)


   r_dist = xCenter(i)


   if ( (r_dist <=( 2* sim_2dR(sim_npoints))).AND.(z_dist <= (2*sim_2dZ(sim_npoints)))) then
      testr = 1e10
      testz = 1e10
      do n = 1, sim_npoints
          if (abs(sim_2dR(n)-r_dist) < testr) then
              r_kat = n
              testr = abs(sim_2dR(n)-r_dist)
          end if
          if (abs(sim_2dZ(n)-z_dist) < testz) then
              z_kat = n
```

```
            testz = abs(sim_2dZ(n)-z_dist)
        end if
    end do
end if

dens      = sim_2dDens(r_kat,z_kat)
temp      = sim_2dTemp(r_kat,z_kat)
vr_temp   = sim_2dvr(r_kat,z_kat)
vphi_temp = sim_2dvphi(r_kat,z_kat)
vz_temp   = sim_2dvz(r_kat,z_kat)

trcr(i) = 0.e0

phi = 0.e0
```

```
eosData(EOS_TEMP) = temp

eosData(EOS_DENS) = dens


eosData(EOS_ABAR) = abar !  14.1176470588

eosData(EOS_ZBAR) = zbar !  7.0588235294

qbar=sim_qbarU


vecLen=1


call Eos(MODE_DENS_TEMP, vecLen,eosData,massFraction,eosMask)

pres = eosData(EOS_PRES)

uint = eosData(EOS_EINT)

gamma = eosData(EOS_GAMC)
```

```
vx(i) = 0.e0

vy(i) = 0.e0

vz(i) = 0.e0


vx(i)=vr_temp

vy(i)=vz_temp

vz(i)=-vphi_temp !NOTE: cylindrical r,phi,z is a right-handed trihedron, so is r,z,-phi


t   (i) = temp

rho (i) = dens

p   (i) = pres

e   (i) = uint + 0.5e0*(vx(i)**2+vy(i)**2+vz(i)**2)

eint(i) = uint

gamc(i) = gamma
```

```
            game(i) = pres/(dens*uint) + 1.e0


            f(i) = phi


solnData(DENS_VAR,i,j,k)=rho(i)
        solnData(PRES_VAR,i,j,k)=p(i)
solnData(TEMP_VAR,i,j,k)=t(i)
solnData(VELX_VAR,i,j,k)=vx(i)
solnData(VELY_VAR,i,j,k)=vy(i)
solnData(VELZ_VAR,i,j,k)=vz(i)
solnData(GAMC_VAR,i,j,k)=gamc(i)
solnData(GAME_VAR,i,j,k)=game(i)
solnData(ENER_VAR,i,j,k)=e(i)
        solnData(C12_SPEC,i,j,k)=sim_xc12
```

```
        solnData(O16_SPEC,i,j,k)=sim_xo16


#ifdef SPECANG_MSCALAR
solnData(SPECANG_MSCALAR,i,j,k)=vz(i)*xCenter(i)
#endif


#ifdef EINT_VAR
solnData(EINT_VAR,i,j,k)=eint(i)
#endif




    end do
   end do
  end do
```

```
  write(0,*)"End of first initialization loop"


  if(.true.) then !Adding magnetic field


#if defined(MAGX_VAR) && defined(MAGY_VAR) && defined(MAGZ_VAR)
  do k = blkLimits(LOW,KAXIS),blkLimits(HIGH,KAXIS)
     do j = blkLimits(LOW,JAXIS),blkLimits(HIGH,JAXIS)+1
        do i = blkLimits(LOW,IAXIS),blkLimits(HIGH,IAXIS)+1
  dist = sqrt(xCenter(i)**2 + yCenter(j)**2)


          if(dist<=RB) then
            magxZone(i)= B0
            magyZone(i)= B0
          else
```

```
              magxZone(i) = 3*magAmplitude*(xCenter(i))*yCenter(j)/(dist**5)
              magyZone(i) =magAmplitude*(2*yCenter(j)**2 -xCenter(i)**2)/(dist**5)
            end if
            magzZone(i) = 0.0

#if NFACE_VARS > 0
            if (sim_killdivb) then
               facexData(MAG_FACE_VAR,i,j,k) = magxZone(i)
               faceyData(MAG_FACE_VAR,i,j,k) = magyZone(i)
               if (NDIM==3) facezData(MAG_FACE_VAR,i,j,k) = magzZone(i)
            else
               solnData(MAGX_VAR,i,j,k)=magxZone(i)
     solnData(MAGY_VAR,i,j,k)=magyZone(i)
               solnData(MAGZ_VAR,i,j,k)=magzZone(i)
```

```
  endif
#else
          solnData(MAGX_VAR,i,j,k)=magxZone(i)
          solnData(MAGY_VAR,i,j,k)=magyZone(i)
          solnData(MAGZ_VAR,i,j,k)=magzZone(i)
#endif



        end do !
     end do !
  end do !
#endif
    do k=blkLimits(LOW,KAXIS),blkLimits(HIGH,KAXIS)
       do j = blkLimits(LOW,JAXIS),blkLimits(HIGH,JAXIS)
```

```
          do i = blkLimits(LOW,IAXIS),blkLimits(HIGH,IAXIS)
#if NFACE_VARS > 0
             if (sim_killdivb) then
               solnData(MAGX_VAR,i,j,k) = 0.5*(facexData(MAG_FACE_VAR,i,j,k)
                        +facexData(MAG_FACE_VAR,i+1,j,k))
               solnData(MAGY_VAR,i,j,k) = 0.5*(faceyData(MAG_FACE_VAR,i,j,k)
                        +faceyData(MAG_FACE_VAR,i,j+1,k))
               solnData(MAGZ_VAR,i,j,k) = 0.0
               solnData(DIVB_VAR,i,j,k) = &
                     ((xCenter(i)+0.5*dx(i))*facexData(MAG_FACE_VAR,i+1,j,k) &
                 -  (xCenter(i)-0.5*dx(i))*facexData(MAG_FACE_VAR,i,j,k))/dx(i)/xCenter(i) &
                 + (faceyData(MAG_FACE_VAR,i,j+1,k) - faceyData(MAG_FACE_VAR,i,j,k))/dy(i)

             endif
```

```
#endif
                ! Update the magnetic pressure
                solnData(MAGP_VAR,i,j,k) = .5*dot_product(solnData(MAGX_VAR:MAGZ_VAR,i,j,k),&
                                                          solnData(MAGX_VAR:MAGZ_VAR,i,j,k))

          enddo
        enddo
      enddo

  endif

  write(0,*)"End of magnetic initialization loop"

  deallocate(xCoord, stat=istat)
```

```
  deallocate(xCoordL,stat=istat)
  deallocate(xCoordR,stat=istat)

  deallocate(yCoord, stat=istat)
  deallocate(yCoordL,stat=istat)
  deallocate(yCoordR,stat=istat)

  deallocate(zCoord, stat=istat)
  deallocate(zCoordL,stat=istat)
  deallocate(zCoordR,stat=istat)

  call Grid_releaseBlkPtr(blockID,solnData,CENTER)
#if NFACE_VARS > 0
  if (sim_killdivb) then
```

```
      call Grid_releaseBlkPtr(blockID,facexData,FACEX)
      call Grid_releaseBlkPtr(blockID,faceyData,FACEY)
      if (NDIM == 3) call Grid_releaseBlkPtr(blockID,facezData,FACEZ)
    endif
#endif
    call Grid_releaseBlkPtr(blockID,scrch_Ptr,SCRATCH)


    deallocate(xCenter,STAT=istat)
      if (istat /= 0) call Driver_abortFlash("Cannot deallocate xCenter in Simulation_initBlock")
    deallocate(yCenter,STAT=istat)
      if (istat /= 0) call Driver_abortFlash("Cannot deallocate yCenter in Simulation_initBlock")
    deallocate(zCenter,STAT=istat)
      if (istat /= 0) call Driver_abortFlash("Cannot deallocate zCenter in Simulation_initBlock")
    deallocate(dx,STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate dx in Simulation_initBlock")
deallocate(dy,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate dy in Simulation_initBlock")
deallocate(dz,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate dz in Simulation_initBlock")
deallocate(rho,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate rho in Simulation_initBlock")
deallocate(p,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate p in Simulation_initBlock")
deallocate(e,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate e in Simulation_initBlock")
deallocate(eint,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate eint in Simulation_initBlock")
deallocate(t,STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate t in Simulation_initBlock")
deallocate(game,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate game in Simulation_initBlock")
deallocate(gamc,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate gamc in Simulation_initBlock")
deallocate(vx,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate vx in Simulation_initBlock")
deallocate(vy,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate vy in Simulation_initBlock")
deallocate(vz,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate vz in Simulation_initBlock")
deallocate(f,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate f in Simulation_initBlock")
deallocate(ye_row,STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate ye_row in Simulation_initBlock")
deallocate(sumy,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate sumy in Simulation_initBlock")
deallocate(qbar_row,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate qbar_row in Simulation_initBlock")
deallocate(trcr,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate trcr in Simulation_initBlock")
deallocate(xvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate xvector in Simulation_initBlock")
deallocate(yvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate yvector in Simulation_initBlock")
deallocate(zvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate zvector in Simulation_initBlock")
deallocate(velxvector,STAT=istat)
```

```
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate velxvector in Simulation_initBlock")
deallocate(velyvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate velyvector in Simulation_initBlock")
deallocate(velzvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate velzvector in Simulation_initBlock")
deallocate(densvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate densvector in Simulation_initBlock")
deallocate(presvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate presvector in Simulation_initBlock")
deallocate(tempvector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate tempvector in Simulation_initBlock")
deallocate(enervector,STAT=istat)
  if (istat /= 0) call Driver_abortFlash("Cannot deallocate enervector in Simulation_initBlock")
deallocate(magxZone,STAT=istat)
```

```
    if (istat /= 0) call Driver_abortFlash("Cannot deallocate magxvector in Simulation_initBlock")
  deallocate(magyZone,STAT=istat)
    if (istat /= 0) call Driver_abortFlash("Cannot deallocate magyvector in Simulation_initBlock")
  deallocate(magzZone,STAT=istat)
    if (istat /= 0) call Driver_abortFlash("Cannot deallocate magzvector in Simulation_initBlock")


  write(0,*)"End Simulation_initBlock"


  return
end subroutine Simulation_initBlock
```