

# Jacobian-Free Newton-Krylov method for multilevel NLTE radiative transfer problems

D. Arramy, J. de la Cruz Rodríguez, and J. Leenaarts

Institute for Solar Physics, Dept. of Astronomy, Stockholm University, AlbaNova University Centre, SE-106 91 Stockholm, Sweden  
e-mail: dimitri.arramy@astro.su.se

Received ; accepted

## ABSTRACT

*Context.* The calculation of the emerging radiation from a model atmosphere requires knowledge of the emissivity and absorption coefficients, which are proportional to the atomic level population densities of the levels involved in each transition. Due to the intricate interdependency of the radiation field and the physical state of the atoms, iterative methods are required in order to calculate the atomic level population densities. A variety of different methods have been proposed to solve this problem, which is known as the Non-Local Thermodynamical Equilibrium (NLTE) problem.

*Aims.* Our goal is to develop an efficient and rapidly converging method to solve the NLTE problem under the assumption of statistical equilibrium. In particular, we explore the usability of a Jacobian-Free Newton-Krylov (JFNK) method, which does not require an explicit construction of the Jacobian matrix, by estimating the new correction with a Krylov-subspace method.

*Methods.* We have implemented a NLTE radiative transfer code with overlapping bound-bound and bound-free transitions, which solves the statistical equilibrium equations using a JFNK method, assuming a depth-stratified plane-parallel atmosphere. As a reference, we have also implemented the Rybicki & Hummer (1992) method based on linearization and operator splitting.

*Results.* Our tests with the FALC model atmosphere and two different 6-level Ca II and H I atoms show that the JFNK method can converge faster than our reference case, by up to a factor 2. This number is evaluated in terms of the total number of evaluations of the formal solution of the radiative transfer equation for all frequencies and directions. This method is also capable of reaching a lower residual error compared to the reference case.

*Conclusions.* The Newton-Krylov method developed in this study poses a new alternative to solve the NLTE problem. Because it is not based on operator splitting with a local approximate operator, it can improve the convergence of the NLTE problem in highly scattering cases. One major advantage of this method is that it should allow for a direct implementation of more complex problems, such as having overlapping transitions from different active atoms, charge conservation or a more efficient treatment of partial redistribution, without having to explicitly linearize the equations.

**Key words.** Radiative transfer – Sun : atmosphere – Methods: numerical – Line: profiles

## 1. Introduction

The statistical equilibrium equations describe the radiative and collisional transitions between the different levels of a model atom (see, e.g., Hubeny & Mihalas 2014). When the collisional terms dominate the rate equations, the assumption of Local Thermodynamical Equilibrium (LTE) is usually adequate and the atomic level population densities (hereafter population densities) can be obtained analytically using the Saha-Boltzmann equations. But when the radiative terms become relevant, the radiation field greatly influences the population densities (NLTE). Because of this cross-dependence of the population densities with the radiation field, the NLTE problem must be solved iteratively in order to make them consistent with each other. Moreover, the non-locality of the radiation field increases the complexity of the problem and its complexity because all grid cells must be solved simultaneously.

Early attempts solved the rate equations using Lambda iteration, which is based on the fixed point iteration method (see, e.g., Hubeny & Mihalas 2014). However, such scheme presents very poor convergence properties, and it is unusable in practice. Auer & Mihalas (1969) proposed a complete linearization method (of the second order transfer equation) to solve the structure and radiation emerging from static stellar atmospheres. Their imple-

mentation traded off physical accuracy in order to make the problem computationally tractable, but it inspired future developments in the field (see below). A different approach, introduced by Rybicki (1972), utilized the *core saturation* approximation to eliminate passive photon scatterings in the line core, while only keeping the much more efficient scatterings in the line wings. The latter improved the conditioning of the rate equations, allowing traditional Lambda iteration to converge in a reasonable (yet large) number of iterations.

The most successful methods to solve the statistical equilibrium equations are based on the operator splitting technique (Cannon 1973) combined with a linearization of the problem (e.g., Auer & Mihalas 1969; Scharmer & Carlsson 1985; Rybicki & Hummer 1992). Scharmer & Carlsson (1985) linearized the first order radiative transfer equation and the rate equations with respect to the population densities until they could derive a linear system to estimate a correction. (Rybicki & Hummer 1992, RH92 hereafter) followed a slightly different approach, replacing some of the quantities that depend on the population densities with the value from the previous iteration. The fundamental difference between these two methods is that the complete linearization method of Scharmer & Carlsson (1985) is a minimization method of the error in the rate equations, whereas

the RH92 method is closer to the fixed point iteration method but uses the operator splitting technique to drive the solution. Furthermore, the complete linearization method of [Scharmer & Carlsson \(1985\)](#) operates on the source function whereas the RH92 method operates on the emissivity, allowing for a simpler treatment of overlapping (active) transitions and partial redistribution effects ([Uitenbroek 2001](#); [Leenaarts et al. 2012](#); [Sukhorukov & Leenaarts 2017](#)).

The performance of these methods is largely determined by the choice of the approximate operator. The simplest *diagonal* (local) operator ([Olson et al. 1986](#)) decouples the explicit dependence of the rate equations with respect to space and it requires a minimal amount of storage and operations, making it a great choice for multi-dimensional problems (e.g., [Leenaarts & Carlsson 2009](#); [Amarsi et al. 2018](#)). The trade off is that it ignores information about the non-local contribution to the intensity and where it originates from. A better prediction of the mean intensity can be attained by using the single-point quadrature *global* operator ([Scharmer 1981](#); [Scharmer & Nordlund 1982](#)), greatly inspired by the Eddington-Barbier approximation ([Milne 1921](#); [Eddington 1926](#); [Barbier 1943](#)). Although it only requires one coefficient per ray and direction (two if linear interpolation is used), the equations become spatially-coupled again and they must be solved together. Schemes using the global operator generally converge in less iterations than those using the local operator. Several codes with implementations of the complete linearization ([Carlsson 1986](#); [Hubeny & Lites 1995](#)) and RH92 ([Uitenbroek 2001](#); [Leenaarts & Carlsson 2009](#); [Pereira & Uitenbroek 2015](#); [Socas-Navarro et al. 2015](#); [Amarsi et al. 2018](#); [Milić & van Noort 2018](#); [Osborne & Milić 2021](#)) methods have been extensively used by the solar and stellar communities.

A common way of solving non-linear systems of equations is the Newton-Raphson method ([Raphson 1690](#); [Newton 1736](#)). The main limitation to applying it to the statistical equilibrium equations is the expensive calculation of the Jacobian matrix that is required in each iteration. In this paper, we propose to use a modification of the Newton-Raphson method known as the Jacobian-Free Newton-Krylov method ([Knoll & Keyes 2004](#)), to solve the radiative transfer problem. In this method, the Jacobian matrices are neither inverted nor built or stored. Instead an iterative inversion solver based on Krylov subspaces ([Krylov 1931](#)) is used to estimate the Newton-Raphson correction to the unknowns. This method has already proved to be efficient in several fields such as hydrodynamics or neutron scattering problems. Compared to the method of [Scharmer & Carlsson \(1985\)](#), our method does not require any explicit linearization of the rate and radiative transfer equations and it does not utilize the operator splitting.

In section 2 we introduce the numerical problem under consideration and the proposed numerical method for its resolution, in sec. 3 we discuss our results and in sec. 4 we summarize our conclusions and discuss potentially interesting developments for future studies.

## 2. Problem & Methods

### 2.1. Mathematical description of the problem

#### 2.1.1. Theory

In this paper, we adopt the notation used in [Uitenbroek \(2001\)](#) to express the statistical equilibrium equations. We furthermore assume plane-parallel geometry hereafter. In all equations, unless mentioned otherwise, lower indices refer to atomic levels

whereas upper indices refer to depth points within the atmosphere. The RH92 notation elegantly unifies the expressions for bound-bound and bound-free transitions, allowing for a very clean implementation of the rate equations. For a bound-bound transition between a lower level  $i$  and upper level  $j$ , we can define:

$$V_{ij} = \frac{h\nu}{4\pi} B_{ij} \phi_{ij}(\nu, \mu), \quad (1)$$

$$V_{ji} = \frac{h\nu}{4\pi} B_{ji} \psi_{ij}(\nu, \mu), \quad (2)$$

$$U_{ji} = \frac{h\nu}{4\pi} A_{ji} \psi_{ij}(\nu, \mu), \quad (3)$$

where  $A_{ji}$ ,  $B_{ji}$  and  $B_{ij}$  are the Einstein coefficients,  $\phi_{ij}$  and  $\psi_{ij}$  are the line absorption and emission profiles.  $\nu$  is the frequency and  $\mu$  the line-of-sight angle cosine. Similarly, for bound-free transition we can define:

$$V_{ij} = \alpha_{ij}(\nu), \quad (4)$$

$$V_{ji} = n_e \Phi_{ij}(T) \exp\left\{-\frac{h\nu}{k_B T}\right\} \alpha_{ij}(\nu), \quad (5)$$

$$U_{ji} = V_{ji} \left(\frac{2h\nu^3}{c^2}\right), \quad (6)$$

where  $\alpha_{ij}$  is the photoionization cross-section,  $n_e$  is the electron density and  $\Phi_{ij}(T)$  is the Saha-Boltzmann function evaluated at temperature  $T$ :

$$\Phi_{ij}(T) = \frac{g_i}{2g_j} \left(\frac{h^2}{2\pi m_e k_B T}\right)^{3/2} \exp\left\{\frac{E_j - E_i}{k_B T}\right\}, \quad (7)$$

where  $g$  denotes the level statistical weight,  $E$  the level energy and  $m_e$  is the mass of the electron. In practice, these expressions can be further simplified using the Einstein relations between coefficients to obtain:

$$V_{ji} = g_{ij} V_{ij}, \quad (8)$$

$$U_{ji} = g_{ij} \left(\frac{2h\nu^3}{c^2}\right) V_{ij}. \quad (9)$$

For bound-bound transitions, assuming complete-redistribution of scattered photons,  $g_{ij} = g_i/g_j$ . For bound-free transitions:

$$g_{ij} = n_e \Phi_{ij}(T) \exp\left\{-\frac{h\nu}{k_B T}\right\} = \frac{n_i^*}{n_j^*} \exp\left\{-\frac{h\nu}{k_B T}\right\}, \quad (10)$$

where the  $*$ -superindex denotes the LTE atomic level population.

We can now write the rate equations as a function of  $V_{ij}$ , regardless of whether we are considering bound-bound or bound-free transitions. Let us recall the rate equation for the atomic level  $i$  at depth index  $k$ :

$$\sum_p \left\{ n_p^k (C_{pi}^k + R_{pi}^k) \right\} = \sum_p \left\{ n_i^k (C_{ip}^k + R_{ip}^k) \right\}, \quad (11)$$

where  $n_i^k$  is the population density of the atomic level  $i$  at depth index  $k$ .  $C_{ij}^k$  and  $R_{ij}^k$  are respectively the collisional and radiative rate coefficients of the transition  $i \rightarrow j$  at depth index  $k$  with  $C_{ii}^k = R_{ii}^k = 0$ . The radiative rate coefficients can be expressed as a double integral over angle and frequency of the intensity ([Uitenbroek 2001](#)):

$$R_{ij}^k = \frac{1}{2} \int_{-1}^1 d\mu \int_0^\infty \frac{d\nu}{h\nu} V_{ij}^k I_{\mu\nu}^k(\mathbf{n}) \quad i < j \quad (12)$$

$$R_{ji}^k = \frac{1}{2} \int_{-1}^1 d\mu \int_0^\infty \frac{d\nu}{h\nu} \left[ \left(\frac{2h\nu^3}{c^2}\right) + I_{\mu\nu}^k(\mathbf{n}) \right] g_{ij}^k V_{ij}^k \quad i < j \quad (13)$$

for a plane-parallel atmosphere. Expressions for  $V_{ij}^k$  and  $g_{ij}^k$  are given in Eqs. 1-10 for bound-bound and bound-free transitions. The last component  $I_{\mu\nu}^k$  is the intensity in direction  $\mu$  at frequency  $\nu$  and at depth index  $k$ . The vector  $\mathbf{n}$  contains the population densities with the chosen structure  $(n_1^1, \dots, n_{N_\ell}^1, \dots, n_1^{N_z}, \dots, n_{N_\ell}^{N_z})^T$  where  $N_z$  and  $N_\ell$  are respectively the number of depth points and active atomic levels. While all the other quantities do not depend of the population densities, the intensity involves them all in a non-linear and non-local fashion through the radiative transfer equation (RTE):

$$I_{\mu\nu}(\tau_{\mu\nu}) = \int_{\tau_{\mu\nu}}^{\infty} S_{\mu\nu}(t) e^{-(t-\tau_{\mu\nu})} dt \quad \mu > 0 \quad (14)$$

$$I_{\mu\nu}(\tau_{\mu\nu}) = \int_{\tau_{\mu\nu}}^0 S_{\mu\nu}(t) e^{-(\tau_{\mu\nu}-t)} dt \quad \mu < 0 \quad (15)$$

where  $S_{\mu\nu} = \eta_{\mu\nu}/\chi_{\mu\nu}$  is the source function,  $\chi_{\mu\nu}$  and  $\eta_{\mu\nu}$  are respectively the total opacity and emissivity, which can be calculated through:

$$\chi_{\mu\nu} = \chi_c + \chi_{\text{sca}} + \sum_i \sum_{j>i} V_{ij} (n_i - g_{ij} n_j) \quad (16)$$

$$\eta_{\mu\nu} = \eta_c + \chi_{\text{sca}} J_\nu + \sum_j \sum_{i<j} \left( \frac{2h\nu^3}{c^2} \right) g_{ij} V_{ij} n_j \quad (17)$$

where the subindex "c" refers to the background continuum contribution and "sca" indicates the background scattering contribution, which are assumed to be independent with respect to the active population densities. The presence of the mean intensity  $J_\nu$  in the scattering term of Eq. 17 could complexify the calculations as it depends of the intensity, which in turn depends on opacities and emissivities. But since those scattering terms do not originate from active transitions of the atom, we use a previous estimation of the mean intensity  $J_\nu^\dagger$  instead of  $J_\nu$  in Eq. 17. The optical thickness  $\tau_{\mu\nu}$  is obtained by integrating the opacity over depth:

$$\tau_{\mu\nu}(z) = \frac{1}{|\mu|} \int_0^z \chi_{\mu\nu}(z') dz' > 0. \quad (18)$$

Equation 11 describes a system of  $N_\ell \times N_z$  equations and variables  $n_i^k$  in which  $N_z$  equations are redundant. Therefore we replace one rate equation by a particle conservation equation per depth-point:

$$\sum_p n_p^k = n_{\text{tot}}^k, \quad (19)$$

where  $n_{\text{tot}}^k$  is the total atom density at depth index  $k$  and is kept constant. The replacement is done on the most populated atomic level, at each depth point for numerical stability purposes.

Finally, the system of equations is reformulated as:

$$F_i^k(\mathbf{n}) \stackrel{\text{def}}{=} \sum_p \left\{ n_i^k (C_{ip}^k + R_{ip}^k) - n_p^k (C_{pi}^k + R_{pi}^k) \right\}. \quad (20)$$

for radiative rate equations and:

$$F_i^k(\mathbf{n}) \stackrel{\text{def}}{=} n_{\text{tot}}^k - \sum_p n_p^k \quad (21)$$

for mass conservation equations. Altogether, these equations form a residual vector  $\mathbf{F}(\mathbf{n})$  with the same dimension as  $\mathbf{n}$ . Solving the system of equations for the vector of population densities  $\mathbf{n}$  is therefore equivalent to finding the root of the residual

vector  $\mathbf{F}$ . The calculation of  $\mathbf{F}$  for a given atmosphere and population densities is detailed in algorithm 1. The residual vector is the central part of the solving process and constitutes the major computational cost of it. Thus, we evaluate the performance of a solver by the number of computations calculations of  $\mathbf{F}$  (hereafter calls) needed to solve the problem to a given precision.

---

**Algorithm 1:** Calculation of the residual vector  $\mathbf{F}$ 


---

**Data:** a population densities vector  $\mathbf{n}$ , an estimation of  $\mathbf{J}$

**Result:**  $\mathbf{F}(\mathbf{n})$ , possibly a new estimation of  $\mathbf{J}$

**for**  $\nu = \nu_1, \dots, \nu_{N_\nu}$  **do**

$\mathbf{J}_\nu^\dagger \leftarrow \mathbf{J}_\nu$ ;

**if** updating  $\mathbf{J}$  **then**

$\mathbf{J}_\nu \leftarrow \mathbf{0}$ ;

**end**

**for**  $\mu = \mu_1, \dots, \mu_{N_\mu}$  **do**

$\chi_{\mu\nu} \leftarrow$  Eq. 16;

$\eta_{\mu\nu} \leftarrow$  Eq. 17;

$S_{\mu\nu} \leftarrow \eta_{\mu\nu}/\chi_{\mu\nu}$ ;

$I_{\mu\nu} \leftarrow$  Eq. 22 and 23;

**for** transitions  $i \rightarrow j$  with  $j < i$ , for all  $z$  **do**

$\mathbf{R}_{ij} \leftarrow \mathbf{R}_{ij} + (\mu, \nu)$ -contribution of Eq. 12;

$\mathbf{R}_{ji} \leftarrow \mathbf{R}_{ji} + (\mu, \nu)$ -contribution of Eq. 13;

**end**

**if** updating  $\mathbf{J}$  **then**

$\mathbf{J}_\nu \leftarrow \mathbf{J}_\nu + \omega_\mu \mathbf{I}_{\mu\nu}$ ;

**end**

**end**

**end**

$\mathbf{F} \leftarrow \mathbf{0}$ ;

**for** transitions  $i \rightarrow j$  with  $j < i$  **do**

$\kappa \leftarrow n_i (C_{ij} + R_{ij}) - n_j (C_{ji} + R_{ji})$ ;

$\mathbf{F}_i \leftarrow \mathbf{F}_i + \kappa$ ;

$\mathbf{F}_j \leftarrow \mathbf{F}_j - \kappa$ ;

**end**

**for**  $k = 1, \dots, N_z$  **do**

$i \leftarrow \{j; n_j^k = \max_p(n_p^k)\}$ ;

$\mathbf{F}_i^k \leftarrow n_{\text{tot}}^k - \sum_p n_p^k$ ;

**end**

---

### 2.1.2. Discretization of the RTE

In practice when computing the radiative rates, the angular and frequency integrals are discretized according to quadrature schemes and yield quadrature coefficients  $(\omega_\mu, \omega_\nu)$  for each set  $(\mu, \nu)$ . Equations 14 and 15 are discretized along the depth axis and the involved integrals can be calculated assuming a depth dependent profile for  $S_{\mu\nu}$ . Such profile is usually taken as simple piecewise polynomial functions. In this paper, we consider piece-wise linear functions (Olson & Kunasz 1987) which yields:

$$I_{\mu\nu}^k = I_{\mu\nu}^{k+1} e^{-\Delta\tau_{\mu\nu}^k} + a^k S_{\mu\nu}^k + b^k S_{\mu\nu}^{k+1} \quad \mu > 0 \quad (22)$$

$$I_{\mu\nu}^k = I_{\mu\nu}^{k-1} e^{-\Delta\tau_{\mu\nu}^{k-1}} + a^{k-1} S_{\mu\nu}^k + b^{k-1} S_{\mu\nu}^{k-1} \quad \mu < 0 \quad (23)$$

where the coefficients  $a^k$  and  $b^k$  are given in appendix A.1 and

$$\Delta\tau_{\mu\nu}^k \approx \frac{1}{2|\mu|} |z^k - z^{k+1}| (\chi_{\mu\nu}^k + \chi_{\mu\nu}^{k+1}) \quad (24)$$

is the optical tickness of the slab.

At the top of the atmosphere, we assume that there is no incoming radiation. The deepest point in the atmosphere is assumed to be thermalized so that the intensity at this location is the Planck function  $B_\nu$ :

$$I_{\mu\nu}^{N_z} = B_\nu(T^{N_z}) \quad \mu > 0 \quad (25)$$

$$I_{\mu\nu}^1 = 0 \quad \mu < 0 \quad (26)$$

The angular integrals are evaluated using a Gauss-Legendre quadrature defined in  $]0, 1[$ . For each angle the incoming and outgoing rays are evaluated in the calculation of  $\mathbf{J}$ . The number of quadrature points is a parameter set by the user. We normally run with five quadrature points with two rays per angle.

## 2.2. The Newton-Raphson method

### 2.2.1. Basics

Solving the system of non-linear equations  $\mathbf{F}(\mathbf{n}) = \mathbf{0}$  for the vector  $\mathbf{n}$  may be achieved through several numerical iterative methods. The Newton-Raphson method is one of the simplest and most powerful ones (e.g., Press et al. 2002). If  $\mathbf{n}^{(p)}$  is the estimation of the solution on the  $p^{\text{th}}$  iteration, the next iterate  $\mathbf{n}^{(p+1)}$  is sought such that  $\mathbf{F}(\mathbf{n}^{(p+1)}) = \mathbf{0}$ . If we further define  $\delta\mathbf{n}^{(p)} = \mathbf{n}^{(p+1)} - \mathbf{n}^{(p)}$  as the  $p^{\text{th}}$  incremental, the Newton-Raphson method relies on a linearization of  $\mathbf{F}(\mathbf{n}^{(p+1)})$ :

$$\mathbf{0} = \mathbf{F}(\mathbf{n}^{(p+1)}) = \mathbf{F}(\mathbf{n}^{(p)} + \delta\mathbf{n}^{(p)}) \approx \mathbf{F}(\mathbf{n}^{(p)}) + \mathbf{J}_F(\mathbf{n}^{(p)})\delta\mathbf{n}^{(p)} \quad (27)$$

where we have introduced the Jacobian matrix  $\mathbf{J}_F$  associated to the residual vector  $\mathbf{F}$  and evaluated at  $\mathbf{n}^{(p)}$ . A possible representation of  $\mathbf{J}_F$  is given in Fig. 1. Solving the latter linear system for  $\delta\mathbf{n}^{(p)}$  yields:

$$\delta\mathbf{n}^{(p)} = -\mathbf{J}_F^{-1}(\mathbf{n}^{(p)})\mathbf{F}(\mathbf{n}^{(p)}), \quad (28)$$

from which one may compute the next iterate  $\mathbf{n}^{(p+1)}$ . This new estimation is *a priori* not a solution to  $\mathbf{F}(\mathbf{n}) = \mathbf{0}$ , although the iterative process ensures in the best cases a quadratic convergence to a solution (e.g., Dennis Jr & Schnabel 1996). The initial guess  $\mathbf{n}^{(0)}$  might be given for instance by the LTE or the radiation-free predictions of the population densities. The linearization introduced by the Newton-Raphson method only consists in a mean to solve the raw statistical equations whereas RH92 solves a linearized, approximated version of the problem.

### 2.2.2. Limitations of the Newton-Raphson method

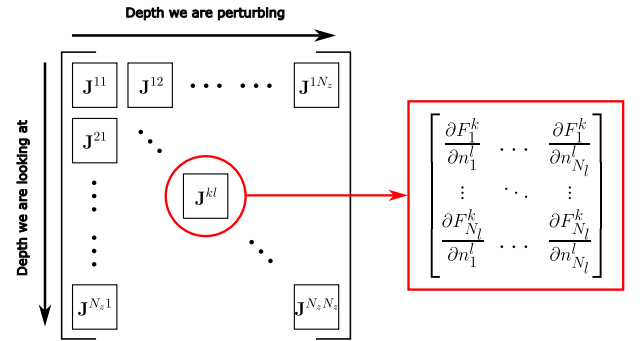
We note that  $\delta\mathbf{n}^{(p)}$  might also lead to a poorer estimation of the solution. This behaviour can occur when the correction  $\delta\mathbf{n}^{(p)}$  lies beyond the domain of linearity of the residual vector around the evaluation vector  $\mathbf{n}^{(p)}$ . A simple way to overcome this behaviour is to limit the incremental vector with a dampening factor  $\alpha$  and try  $\alpha = 1, 0.5, 0.25, \dots$  until  $\|\mathbf{F}(\mathbf{n}^{(p)} + \alpha\delta\mathbf{n}^{(p)})\| < \|\mathbf{F}(\mathbf{n}^{(p)})\|$ . This procedure is the simplest of the so-called line search methods, although more elaborated ones exist (see Dennis & Schnabel 1996, for instance).

A second problem deals with the possibility to produce solution estimations with negative entries. While mathematically correct, a solution estimation with negative population densities is physically incorrect and the solver may even overflow when solving the RTE. A possible solution to prevent negative entries consists in limiting the correction at each depth independently.

A third and inherent problem of the Newton-Raphson method deals with the quality of the initial guess. The initial guess is usually an important factor in the method's convergence rate or even the method's failure. The method will not converge or ventures outside the domain of definition of  $\mathbf{F}$  if a poor starting point is given. The method may also be trapped in a local minimum of the residual vector, which can be difficult to spot. Several tools such as continuation methods can be used to build a robust solver based on the Newton-Raphson method. More details are given in Knoll & Keyes (2004).

The Newton-Raphson requires the knowledge of the inverse of a Jacobian matrix  $\mathbf{J}_F^{-1}$  at each iterative step. Several issues can potentially arise when computing such quantities:

1. Implementation: most problems do not have an analytical expression for  $\mathbf{J}_F$  and an approximation needs to be given (for instance, by finite differences). Hence the convergence is likely to be less than quadratic. In the worst case the method may fail if the approximation is too coarse.
2. Storage: for large problems, storing  $\mathbf{J}_F$  can be problematic.
3. Time consumption: inversion of  $\mathbf{J}_F$  quickly becomes time consuming as the size of the problem increases, considering traditional inversion routines such as Gauss-Jordan elimination. The computation of the full Jacobian might also be expensive.



**Fig. 1.** The Jacobian matrix structure for a  $N_\ell$ -level atom problem.  $\mathbf{J}_F$  is a  $(N_\ell N_z) \times (N_\ell N_z)$  matrix that contains the derivative of the residual vector components with respect to the population densities. This matrix could be seen as a  $N_z \times N_z$  block matrix where each block  $\mathbf{J}^{k\ell}$  is a  $N_\ell \times N_\ell$  matrix.  $\mathbf{J}^{k\ell}$  stores the derivatives of  $\mathbf{F}$  at depth index  $k$  with respect to the population densities at depth index  $\ell$ .

The radiative transfer problem we are considering disqualifies a classical Newton-Raphson method mainly because of the computational cost derived from building of  $\mathbf{J}_F$ , even when using analytical expressions. Using the Newton method therefore requires the information of the Jacobian matrices without or partially building them in order to keep an efficient solver. The next sections will progressively detail a way to bypass this thorny problem.

### 2.3. Iterative inversion: Krylov methods

Equation 28 is equivalent to a linear system of the form  $\mathbf{A}\mathbf{x} = \mathbf{b}$  where  $\mathbf{A} = \mathbf{J}_F(\mathbf{n}^{(p)})$ ,  $\mathbf{x} = \delta\mathbf{n}^{(p)}$  and  $\mathbf{b} = -\mathbf{F}(\mathbf{n}^{(p)})$ . This linear system can be solved for  $\mathbf{x}$  without inverting  $\mathbf{A}$  using iterative approaches such as Krylov methods. In short, Krylov methods are used to solve large linear systems through projections onto a Krylov subspace  $K_s$ :

$$K_s = \text{span}(\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \dots, \mathbf{A}^{s-1}\mathbf{r}_0),$$



where  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$  is the initial residual vector built from the initial guess  $\mathbf{x}_0$ . Since  $\mathbf{x}$  is meant to represent a Newton-Raphson correction, a typical initial guess would be zero and thus  $\mathbf{r}_0 = \mathbf{b}$  (we are considering that no correction is required initially). Another initial condition might be given by the solution to  $\mathbf{P}\mathbf{x}_0 = \mathbf{b}$  where  $\mathbf{P}$  is a preconditioner (§2.4.3). Then, the solution is estimated as:

$$\mathbf{x} - \mathbf{x}_0 = \sum_{i=1}^s \kappa_i \mathbf{q}_i, \quad (29)$$

where the set of vectors  $(\mathbf{q}_1, \dots, \mathbf{q}_s)$  is a basis of  $K_s$  and  $(\kappa_1, \dots, \kappa_s)$  are the corresponding coordinates. The purpose of a Krylov method is therefore to construct a basis of  $K_s$ , then determine the corresponding scalars  $\kappa_i$  through projection methods. This construction is done iteratively in  $s$  iterations (one iteration per basis vector). Each iteration adds a new component to the solution estimation. The solution is sought such that  $\|\mathbf{r}\|_2 = \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2 < \delta \|\mathbf{r}_0\|_2$  where  $\delta$  is a relative tolerance set by the user. We note that the given tolerance might be achieved in less than  $s$  iterations.

The size of the Krylov subspace is related to the precision of the solution one can achieve, the latter also depending of the method employed. If  $s$  is too small, the desired tolerance level might not be achieved. On the other hand, if  $s$  is chosen to be equal to the size of  $A$ , a Krylov method will eventually converge to the exact solution in theory. In practice, round-off and truncation errors will limit the maximum precision one can expect. From the definition of  $K_s$ , one may note that only matrix-vector products are required in such methods, which is the keystone of §2.4.

A plethora of Krylov methods has been developed over the past decades among which two popular techniques and their respective variants are broadly used in various physics problems:

1. The Generalized Minimal RESidual method (GMRES) is usually based on the Arnoldi process or Householder transforms to produce orthonormalized bases. It was first developed in Saad & Schultz (1986) as an improvement and an extension to nonsymmetric matrices of the MINRES method developed by Paige & Saunders (1975). It is a very versatile linear system solver.
2. The Bi-Conjugate Gradient STABilized (BiCGSTAB) is based on the Lanczos bi-orthogonalization procedure which generates non-orthogonal bases. It was first developed in van der Vorst (1992) as a variant of the biconjugate gradient method (BiCG).

Both methods can be used with any invertible matrix. GMRES requires one matrix-vector product per iteration whereas two are needed with BiCGSTAB but the latter requires less memory than GMRES. This is especially true whenever the number of iterations required for convergence is large, since BiCGSTAB uses a constant amount of memory per iteration and GMRES does not. The most crucial feature of GMRES is a strict decrease of the residual norm  $\|\mathbf{r}\|_2$  throughout the iterative process whereas the convergence behavior of BiCGSTAB is less regular (van der Vorst 1992).

## 2.4. Jacobian-Free Newton-Krylov methods

### 2.4.1. Setup

We still have not addressed the problem of the Jacobian matrix estimation and the potential high computational cost it represents. Fortunately, Krylov methods applied to the Newton-Raphson iteration (Eq. 28) only require the action of the Jacobian

matrix  $\mathbf{J}_F$  onto a generic vector  $\mathbf{v}$  (see §2.3). It turns out that the operation  $\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{v}$  can be approximated using finite differences (e.g., Knoll & Keyes 2004):

$$\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{v} = \frac{F(\mathbf{n}^{(p)} + \epsilon \frac{\mathbf{v}}{\|\mathbf{v}\|_2}) - F(\mathbf{n}^{(p)})}{\epsilon} + \mathcal{O}(\epsilon) \quad \text{forward} \quad (30)$$

$$\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{v} = \frac{F(\mathbf{n}^{(p)}) - F(\mathbf{n}^{(p)} - \epsilon \frac{\mathbf{v}}{\|\mathbf{v}\|_2})}{\epsilon} + \mathcal{O}(\epsilon) \quad \text{backward} \quad (31)$$

$$\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{v} = \frac{F(\mathbf{n}^{(p)} + \epsilon \frac{\mathbf{v}}{\|\mathbf{v}\|_2}) - F(\mathbf{n}^{(p)} - \epsilon \frac{\mathbf{v}}{\|\mathbf{v}\|_2})}{2\epsilon} + \mathcal{O}(\epsilon^2) \quad \text{central} \quad (32)$$

where  $\epsilon$  is the difference step. Such schemes do not use the Jacobian matrix but rather extra calls of  $F$ , which is a huge computational and storage gain especially for large problems but at the cost of precision. This is the keystone of Jacobian-Free Newton-Krylov solvers (hereafter JFNK). Such methods were first presented by Knoll & Keyes (2004). Since the residual vector  $F(\mathbf{n}^{(p)})$  is already computed and passed to the Krylov solver, first order schemes (forward and backward) only require one fresh call of  $F$ . In comparison, the second order scheme (central) requires two fresh calls of  $F$  which is a major drawback. In our problem, every evaluation of  $F$  translates into solving the radiative transfer equation for all rays at all frequencies for a given  $\mathbf{n}$ . We also note that the finite-differences calculations in Eq. 30-32, do not estimate the individual elements of the Jacobian matrix, but rather are used to directly estimate the matrix-vector product.

Since the Newton-Raphson method is iterative, the matrix-vector products estimations are only needed to be accurate enough to guarantee convergence. This is the main reason why the vast majority of JFNK solvers are using first order schemes rather than higher order ones (Knoll & Keyes 2004). In practice, round-off and truncation errors may occur and an optimal choice of  $\epsilon$  is hard to find. The first source of error is caused by the finite arithmetic precision of computers while the second source of error is due to the limited accuracy of the scheme. Several yet empirical choices for  $\epsilon$  are further detailed in Knoll & Keyes (2004) to minimize both sources of error.

### 2.4.2. Augmentation of numerical precision with complex numbers

Although, for the considered problem, the components of  $\mathbf{n}^{(p)}$  cover a wide range of values: for instance, considering a two level hydrogen atom in a FALC atmosphere, the LTE population densities cover the range  $10^4$  up to  $10^{17}$   $\text{cm}^{-3}$ . This leads to large values of  $\epsilon$  considering the empirical choices. A rescaling of the densities is possible to keep reasonable  $\epsilon$  values. However the large span of densities remains problematic within the residual vector  $F$  itself and leads to round off errors. Thus, the usual numerical derivative schemes are not suited for the given problem. Luckily, there is an alternative scheme that uses complex numbers to increase the precision and dynamic range of the calculations, which is far less affected by round off errors or subtractive cancellations (Dennis Jr & Schnabel 1996):

$$\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{v} = \frac{\Im[\mathbf{F}(\mathbf{n}^{(p)} + i\epsilon\mathbf{v})]}{\epsilon} + \mathcal{O}(\epsilon^2) \quad (33)$$

where  $i$  is the imaginary unit. This special scheme only requires one fresh call of  $F$  and is second-order accurate as the central difference scheme. It however requires to set up the main routines for complex arithmetic operations. We used the linear piecewise source function scheme described in §2.1 to integrate the radiative transfer equation along rays, for which the modifications are straightforward. The remaining truncation error can be

greatly attenuated by selecting a tiny  $\epsilon$  value. Dennis Jr & Schnabel (1996) recommend a typical value  $\epsilon \sim 10^{-200}$  for double-precision functions. Note that the imaginary part is only used for the Krylov solver, otherwise only the (unperturbed) real part is considered. Since the imaginary part is typically very small compared to the real part, one should linearize equations that involve the perturbations (e.g. computing the source function). This prevents introducing undesired arithmetic errors. Fig. 2 points out a typical accuracy issue in the computation of the Jacobian matrices when using traditional schemes. The complex scheme is the only one that can provide accurate estimations of Jacobian matrices for our given NLTE problem. For this sole reason, the complex scheme is the one that we use in our JFNK solvers.

### 2.4.3. Preconditioning

Although we have introduced an accurate method to evaluate the matrix-vector products  $\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{v}$ , the Jacobian matrix may potentially be ill-conditioned. Consequently, the Krylov solver may require many iterations to converge to the desired tolerance because of the high condition number of  $\mathbf{J}_F(\mathbf{n}^{(p)})$ . Therefore we propose to precondition the Krylov solver in order to increase its efficiency. The preconditioning process consists in using a preconditioning matrix  $\mathbf{P}$  (the preconditioner) such that  $\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{P}^{-1}$  (right preconditioning) or  $\mathbf{P}^{-1}\mathbf{J}_F(\mathbf{n}^{(p)})$  (left preconditioning) has a lower condition number than  $\mathbf{J}_F(\mathbf{n}^{(p)})$ . The system to be solved by the Krylov solver is not given by Eq. 28 anymore but rather by:

$$(\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{P}^{-1})(\mathbf{P}\delta\mathbf{n}^{(p)}) = -\mathbf{F}(\mathbf{n}^{(p)}) \quad (34)$$

for right preconditioning and:

$$(\mathbf{P}^{-1}\mathbf{J}_F(\mathbf{n}^{(p)}))\delta\mathbf{n}^{(p)} = -\mathbf{P}^{-1}\mathbf{F}(\mathbf{n}^{(p)}) \quad (35)$$

for left preconditioning. The preconditioned system is expected to be solved in less iterations than the original system. Equation 34 is solved for  $\mathbf{y} = \mathbf{P}\delta\mathbf{n}^{(k)}$  first then for  $\delta\mathbf{n}^{(k)}$ . In our solvers, we will use the left preconditioning, since the right preconditioning involves an extra inversion step. The matrix-vector product scheme (Eq. 33) can be adapted for such a system and reads:

$$\mathbf{J}_F(\mathbf{n}^{(p)})\mathbf{P}^{-1}\mathbf{v} = \frac{\Im[\mathbf{F}(\mathbf{n}^{(p)} + i\epsilon(\mathbf{P}^{-1}\mathbf{v}))]}{\epsilon} + \mathcal{O}(\epsilon^2). \quad (36)$$

Because the preconditioner is meant to improve the overall performance of the inversion process,  $\mathbf{P}$  should be easy to calculate and invert. A commonly used algebraic preconditioner is given by the diagonal or block diagonal of the matrix one attempts to invert (easy inversion and matrix-vector multiplication). This simple matrix is also known as a Jacobi (or block-Jacobi) preconditioner. This choice is particularly interesting in our case because the block diagonal of  $\mathbf{J}_F(\mathbf{n}^{(p)})$  is as costly to compute as a call of  $\mathbf{F}$  (see §2.1 and §2.5.1). We note however that other physics-based preconditioners could be very relevant for our case, as presented in Janett et al. (2024) for solving linear problems. The calculation of such a preconditioner can be performed when calculating  $\mathbf{F}(\mathbf{n}^{(p)})$ , thus reusing most of the variables that were already computed to obtain the residual vector. Algorithm 2 illustrates the final JFNK solving routine.

### 2.5. Analytical Jacobian matrix

In this part, we derive the expressions of the Jacobian matrix elements as a function of the population densities. In principle,

---

#### Algorithm 2: JFNK solver

---

**Data:** an initial population densities vector  $\mathbf{n}^{(0)}$ , an initial estimation of  $\mathbf{J}$ , a Newton-Raphson tolerance  $ftol$ , a Krylov tolerance  $r_{tol}$ , a maximum number of iterations  $maxiter$

**Result:** A solution vector  $\mathbf{n}$ , a mean intensity estimation  $\mathbf{J}$

```

niter ← 0;
err ← 1 + ftol;
f ← F(n(0))      ▷ update the preconditioner P;
while niter < maxiter and err > ftol do
  x0 ← 0 or x0 ← -P-1f      ▷ initial guess;
  δn ← solution of Eq. 35 with a Krylov solver using
  Eq. 33;
  nprev ← n;
  n ← n + δn;
  f ← F(n)      ▷ update J and P;
  err ← ||(n - nprev)/(n + nprev)||∞;
  niter ← niter + 1;
end

```

---

these equations could be used to compute the fully analytical Jacobian matrix. While the calculation of the full Jacobian is expensive, it is at least important to detail such derivations for preconditioning purposes (§2.4). A more general derivation is provided in Milić, I. & van Noort, M. (2017) for derivatives according to any atmospheric parameter. The Jacobian element  $J_{ij}^{k\ell}$  can be calculated as follows:

$$J_{ij}^{k\ell} = \left( \frac{\partial F_i^k}{\partial n_j^\ell} \right) = \frac{\partial}{\partial n_j^\ell} \left( \sum_p \left\{ n_i^k (C_{ip}^k + R_{ip}^k) - n_p^k (C_{pi}^k + R_{pi}^k) \right\} \right) \quad (37)$$

unless  $F_i^k$  is a particle conservation equation in which case the Jacobian element is simply given by:

$$J_{ij}^{k\ell} = \frac{\partial}{\partial n_j^\ell} \left( n_{tot}^k - \sum_p n_p^k \right) = -\delta_{k\ell} \quad (38)$$

where we used the fact that the population densities  $n_i^k$  are considered to be independent variables and that  $n_{tot}^k$  is kept constant, therefore:

$$\frac{\partial n_i^k}{\partial n_j^\ell} = \delta_{k\ell} \delta_{ij}. \quad (39)$$

In Eq. 37, the collisional coefficients do not depend on the population densities. Applying the chain rule to Eq. 37 yields:

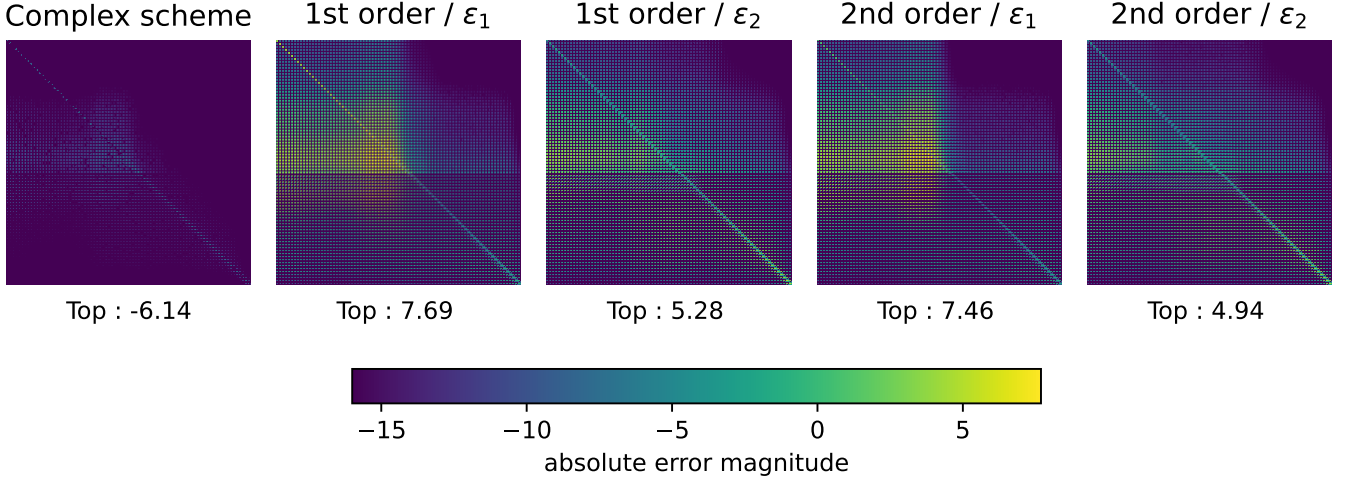
$$J_{ij}^{k\ell} = \left( \frac{\partial F_i^k}{\partial n_j^\ell} \right) = \delta_{k\ell} \Gamma_{ij}^k + A_{ij}^{k\ell} \quad (40)$$

where:

$$\Gamma_{ij}^k = \delta_{ij} \sum_p (C_{ip}^k + R_{ip}^k) - (C_{ji}^k + R_{ji}^k) \quad (41)$$

$$A_{ij}^{k\ell} = \frac{1}{2} \int_{-1}^1 d\mu \int_0^\infty \frac{dv}{hv} \alpha_i^k \left( \frac{\partial I_{\mu\nu}^k}{\partial n_j^\ell} \right) \quad (42)$$

$$\alpha_i^k = \sum_{p>i} \left\{ V_{ip}^k (n_i^k - g_{ip}^k n_p^k) \right\} - \sum_{p<i} \left\{ V_{pi}^k (n_p^k - g_{pi}^k n_i^k) \right\} \quad (43)$$



**Fig. 2.** The Jacobian matrix estimation error for the 3-level Ca II setup, evaluated at the LTE populations and for the different schemes. The difference steps  $\epsilon_1$  and  $\epsilon_2$  are common choices (Knoll & Keyes 2004) and read:  $\epsilon_1 = \sum_{i,k=1}^{N_\ell, N_\zeta} b |(n_i^k)^{(p)}| / (N_\ell N_\zeta \|v\|_2) + b$  where  $b = 10^{-6}$  and  $\epsilon_2 = \epsilon_{\text{mach}}^{1/r} \sqrt{1 + \|n^{(p)}\|_2 / \|v\|_2}$  where  $\epsilon_{\text{mach}}$  is the machine epsilon for double precision numbers,  $r = 2$  for forward and backward schemes and  $r = 3$  for the central scheme. The logarithm of the absolute error (colorbar) is clipped below  $-16$ . First and second order (i.e. forward / backward and central) schemes are terrible at evaluating the Jacobian matrix whereas the complex alternative highlights a remarkable precision.

by noticing that the derivative of the radiative rates only involves the derivative of the intensity  $I_{\mu\nu}^k$  with respect to  $n_j^\ell$ . The extinction profile within each coefficient  $V_{ip}^k$  is considered to be independent with respect to the population densities. Then, we can expand the derivative and further write:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial n_j^\ell}\right) = \sum_p \left\{ \left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^p}\right) \left(\frac{\partial \chi_{\mu\nu}^p}{\partial n_j^\ell}\right) + \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^p}\right) \left(\frac{\partial \eta_{\mu\nu}^p}{\partial n_j^\ell}\right) \right\} \quad (44)$$

because the intensity, through the RTE, is only a function of the optical depth and the source function. One may further develop equation 44 and write:

$$\left(\frac{\partial \chi_{\mu\nu}^p}{\partial n_j^\ell}\right) = \delta_{p\ell} \beta_j^\ell \quad (45)$$

$$\left(\frac{\partial \eta_{\mu\nu}^p}{\partial n_j^\ell}\right) = \delta_{p\ell} \gamma_j^\ell + \chi_{\text{sca}}^p \left(\frac{\partial J_{\nu}^p}{\partial n_j^\ell}\right) \quad (46)$$

where:

$$\beta_j^\ell = \sum_{s>j} V_{js}^\ell (1 - g_{js}^\ell) \quad (47)$$

$$\gamma_j^\ell = \sum_{s<j} \left(\frac{2h\nu^3}{c^2}\right) V_{sj}^\ell g_{sj}^\ell \quad (48)$$

so that Eq. 44 therefore reduces to:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial n_j^\ell}\right) = \beta_j^\ell \left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^\ell}\right) + \gamma_j^\ell \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^\ell}\right) + \sum_p \chi_{\text{sca}}^p \left(\frac{\partial J_{\nu}^p}{\partial n_j^\ell}\right) \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^p}\right) \quad (49)$$

Equation 49 consists in a linear contribution of the intensity and a summation of non-linear cross terms due to the background scattering. Both derivatives involving  $I_{\mu\nu}^k$  depend of the

scheme one will choose to solve the RTE. Since we use the linear piecewise source function scheme detailed in §2.1, the corresponding expressions for the derivatives are:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^\ell}\right) = \left(\frac{\partial I_{\mu\nu}^{k+1}}{\partial \chi_{\mu\nu}^\ell}\right) e^{-\Delta\tau_{\mu\nu}^k} + a_\chi^k \delta_{k\ell} + b_\chi^k \delta_{k+1\ell}, \quad (50)$$

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^\ell}\right) = \left(\frac{\partial I_{\mu\nu}^{k+1}}{\partial \eta_{\mu\nu}^\ell}\right) e^{-\Delta\tau_{\mu\nu}^k} + a_\eta^k \delta_{k\ell} + b_\eta^k \delta_{k+1\ell}, \quad (51)$$

for outgoing rays ( $\mu > 0$ ), and:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^\ell}\right) = \left(\frac{\partial I_{\mu\nu}^{k-1}}{\partial \chi_{\mu\nu}^\ell}\right) e^{-\Delta\tau_{\mu\nu}^{k-1}} + c_\chi^k \delta_{k\ell} + d_\chi^k \delta_{k-1\ell}, \quad (52)$$

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^\ell}\right) = \left(\frac{\partial I_{\mu\nu}^{k-1}}{\partial \eta_{\mu\nu}^\ell}\right) e^{-\Delta\tau_{\mu\nu}^{k-1}} + c_\eta^k \delta_{k\ell} + d_\eta^k \delta_{k-1\ell}, \quad (53)$$

for ingoing rays ( $\mu < 0$ ) where the expressions of the different involved coefficients are given in appendix A.2. These coefficients can be constructed from the variables used when solving the RTE to save computational time. The boundary conditions for this scheme (§2.1) imply that:

$$\left(\frac{\partial I_{\mu\nu}^{N_\zeta}}{\partial \chi_{\mu\nu}^\ell}\right) = 0, \quad \left(\frac{\partial I_{\mu\nu}^{N_\zeta}}{\partial \eta_{\mu\nu}^\ell}\right) = 0 \quad \mu > 0 \quad (54)$$

$$\left(\frac{\partial I_{\mu\nu}^1}{\partial \chi_{\mu\nu}^\ell}\right) = 0, \quad \left(\frac{\partial I_{\mu\nu}^1}{\partial \eta_{\mu\nu}^\ell}\right) = 0 \quad \mu < 0 \quad (55)$$

for each value of  $\ell$ . It can be shown that Eq. 50 to 53 define two upper ( $\mu > 0$ ) and two lower ( $\mu < 0$ ) triangular matrices:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^\ell}\right) = \begin{cases} 0, & \ell < k \\ a_\chi^k, & \ell = k \\ (b_\chi^{\ell-1} + a_\chi^\ell e^{-\Delta\tau_{\mu\nu}^{\ell-1}}) \prod_{i=k}^{\ell-2} e^{-\Delta\tau_{\mu\nu}^i}, & \ell > k \end{cases} \quad (56)$$

for  $\mu > 0$  and:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^\ell}\right) = \begin{cases} 0, & \ell > k \\ c_\chi^k, & \ell = k \\ (d_\chi^{\ell+1} + c_\chi^\ell e^{-\Delta\tau_{\mu\nu}^\ell}) \prod_{i=\ell+1}^{k-1} e^{-\Delta\tau_{\mu\nu}^i}, & \ell < k \end{cases} \quad (57)$$

for  $\mu < 0$ . These expressions are valid for internal depth points. The matrices related to the derivatives with respect to the emissivities are obtained by using the associated coefficients. Such matrices can be understood as Jacobian matrices of the intensity with respect to opacities and emissivities.

The last part to detail deals with the derivative of the mean intensity term. Through its definition, one may note that this quantity involves derivatives of the intensity with respect to the population densities. Shortly, the full expansion of Eq. 49 consists in an intricate, self-consistent yet linear system of the derivatives of the intensity with respect to the population densities. It is however possible to find a solution to this system which can be written as:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial n_j^\ell}\right) = \beta_j^\ell \left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^\ell}\right) + \gamma_j^\ell \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^\ell}\right) + \sum_p \chi_{\text{sca}}^p r_p \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^p}\right), \quad (58)$$

where the coefficients  $r_p$  and the solution derivation can be found in appendix B. In practice, the background scattering contribution to the derivatives is usually very weak and therefore can be neglected.

### 2.5.1. Preconditioning of JFNK with the analytical Jacobian matrix

Preconditioning the JFNK solver is troublesome if one uses the analytical derivation of the Jacobian matrix. Indeed, computing Eq. 49 is expensive if one is only interested in the Jacobi preconditioner, because all off-diagonal terms need to be calculated. Such an issue arises from the background scattering contribution and one therefore has to deal with the following in order to obtain the preconditioner:

- The summation in Eq. 49 involves all cross terms ( $k \neq p$  terms).
- The presence of the mean intensity in the scattering term also involves all cross terms (§2.5).

Therefore, only an approximate Jacobi preconditioner can be used at a cost of precision and potentially convergence rate of the Krylov solver. In this paper, we give three simple solutions to overcome both problems and still provide a preconditioner that dramatically improves the convergence properties of the Krylov solver:

- The very first solution consists in disregarding the background scattering contribution. In this case, Eq. 49 becomes for  $k = \ell$ :

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial n_j^k}\right) = \beta_j^k \left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^k}\right) + \gamma_j^k \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^k}\right) \quad (59)$$

- The second solution consists in discarding all of the cross terms to only keep the local one. The preconditioner further reduces to a local operator if one uses:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial n_j^k}\right) = \beta_j^k \left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^k}\right) + \left[\gamma_j^k + \chi_{\text{sca}}^k \left(\frac{\partial J_{\nu}^k}{\partial n_j^k}\right)^\dagger\right] \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^k}\right) \quad (60)$$

where a previous estimation of the derivative of the mean intensity is used instead of the current one. Such an estimation is easily computed for the next iteration by an integration of Eq. 60 over all angles. One can simply initialize this quantity to zero (zero radiation initial guess) or calculate the mean intensity given by the LTE solution.

- The last solution, which is the least constraining but requires additional operations, uses the solution given by Eq. 58 while considering only the local terms. The corresponding operator then has:

$$\left(\frac{\partial I_{\mu\nu}^k}{\partial n_j^k}\right) = \beta_j^k \left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^k}\right) + \left[\gamma_j^k + \chi_{\text{sca}}^k r_k\right] \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^k}\right) \quad (61)$$

where the coefficient  $r_k$  is:

$$r_k = \frac{\sum_\mu \omega_\mu \left[\beta_j^k \left(\frac{\partial I_{\mu\nu}^k}{\partial \chi_{\mu\nu}^k}\right) + \gamma_j^k \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^k}\right)\right]}{1 - \sum_\mu \omega_\mu \chi_{\text{sca}}^k \left(\frac{\partial I_{\mu\nu}^k}{\partial \eta_{\mu\nu}^k}\right)} \quad (62)$$

A preconditioner that follows one of the three solutions only requires the calculation of  $\ell = k$  terms in Eq. 49, which are only built from the coefficients  $a_\chi^k$ ,  $a_\eta^k$ ,  $c_\chi^k$  and  $c_\eta^k$  given in appendix A.2. Furthermore its calculation can be carried out in a comparable amount of operations than computing the residual vector  $F$ . We will use the first solution to calculate the preconditioner since the scattering terms are negligible in Eq. 58 compared to the other contributions, at least with the considered setups (§3.1).

## 3. Results & Discussion

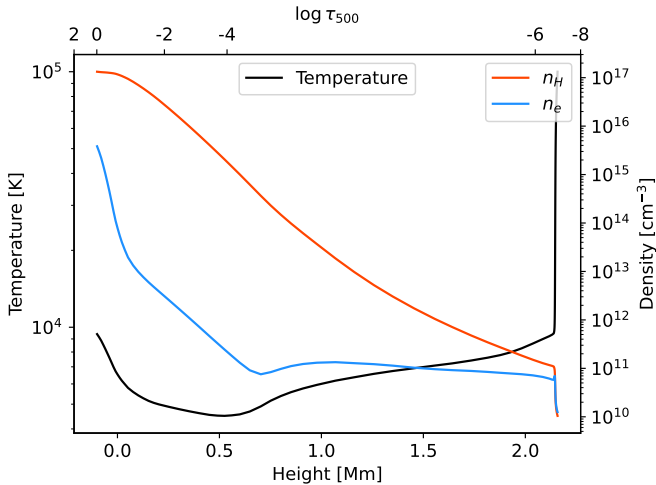
We have ported a simplified version of the excellent RH code (Uitenbroek 2001) to Python. The latter solves the statistical equilibrium equations using the RH92 method, but with the possibility of including partial redistribution effects of scattered photons (PRD). This python version does not include PRD effects and it is significantly slower than the C-version of RH. But it is implemented using modern programming constructions such as classes and inheritance, which has been extremely useful in the implementation of our proof-of-concept JFNK solver as we could reuse most of the atom structures, opacity calculation routines and formal solvers of the transport equation. We use the RH92 method as a reference in order to evaluate the performance of our JFNK method. Given that we are analyzing the convergence properties of different schemes, we have not included Ng-acceleration in our calculations (Ng 1974), which is implemented in the RH code.

### 3.1. Setup

All the results presented in this paper are computed using a FALC model atmosphere of the solar photosphere, chromosphere and transition region (Fontenla et al. 1993), that consists in 82 depths points covering the interval  $\tau_{500} = [10^{-10}, 23]$  where  $\tau_{500}$  is the optical depth at  $\lambda = 500$  nm. Figure 3 depicts the stratifications of the gas temperature, electron density and total hydrogen density. The atmosphere does not have a native line-of-sight velocity profile.

Three different atomic setups are used consisting in H I and Ca II. The different transitions are listed in table 1. Each setup also include collisional and bound-free transitions (Shull & van Steenberg 1982; Burgess & Chidichimo 1983; Arnaud & Rothenflug 1985). The absorption and emission profiles of each





**Fig. 3.** The FALC atmosphere used in this paper. The temperature, the total hydrogen and electron densities are shown. The height dimension origin corresponds to  $\tau_{500} = 1$ .

line at each location are modelled with the Voigt function which depends on the Doppler width and the damping parameter. The latter includes radiative, Stark, linear Stark (in calculations with H atoms) and van der Waals broadening. The angular integrals are discretized using a 5-points Gauss-Legendre quadrature.

We have performed our calculations using the Newton-Raphson, JFNK (using GMRES and BiCGSTAB internally) and the RH92 methods. Both JFNK solvers are systematically using the Jacobi left preconditioner (§2.4.3 and §2.5.1). All solvers require an exit condition that defines a good convergence. In our case, we keep track of the residual norm  $\|\mathbf{F}\|_\infty$  as well as the population relative change norm  $\|\delta\mathbf{n}/\bar{\mathbf{n}}\|_\infty = \frac{1}{2}\|(\mathbf{n}_{\text{new}} - \mathbf{n}_{\text{prev}})/(\mathbf{n}_{\text{new}} + \mathbf{n}_{\text{prev}})\|_\infty$ . Unless mentioned otherwise, we assume that a method has converged when  $\|\delta\mathbf{n}/\bar{\mathbf{n}}\|_\infty < 10^{-4}$ . This condition is most of the time sufficient although we also impose a minimal drop of  $\|\mathbf{F}\|_\infty$  by 2 magnitudes to avoid premature exits.

### 3.2. Krylov solver tolerance impact

The Krylov solver that is internally used in the JFNK method can generally have a number of parameters that must be chosen for a given run (such as the size of the subspace or the convergence criteria). In the case of simple solvers such as GMRES or BiCGSTAB, this set of parameters reduces to the accuracy to which the solution is desired. This single parameter steers the behavior of the JFNK method and its convergence properties. Thus, we have investigated the impact of the Krylov solver accuracy on the stability and the efficiency of the JFNK method.

Our first test is to assess the ability of a JFNK solver to match the Newton-Raphson solution. We would expect minimal differences between both solvers as the accuracy of the Krylov solver increases. Figure 4 shows the convergence properties of our solvers in the case of the 6-level H I atom setup. Several accuracy levels are displayed to highlight the evolution of the discrepancies between the different methods. The Newton-Raphson solver outperforms JFNK solvers in every situation. By truncating the precision of the correction provided by the Krylov solver, each JFNK iteration becomes less accurate, usually leading to extra iterations (compared to the standard Newton-Raphson case) in order to achieve a given convergence level in the population densities. The differences between the two methods decrease when

the precision of the Krylov solver is substantially increased. Both JFNK solvers display similar results and converge to the same solution. They do match the same behavior as the Newton-Raphson solver when  $r_{\text{tol}} \sim 10^{-4}$ , validating the implementation of our solvers.

A second chart is presented in Fig. 5 and directly compares our iterative solvers with the RH92 solver. In the JFNK method, we would expect a range of tolerances for which the method is optimal (with respect to the residual function calls):

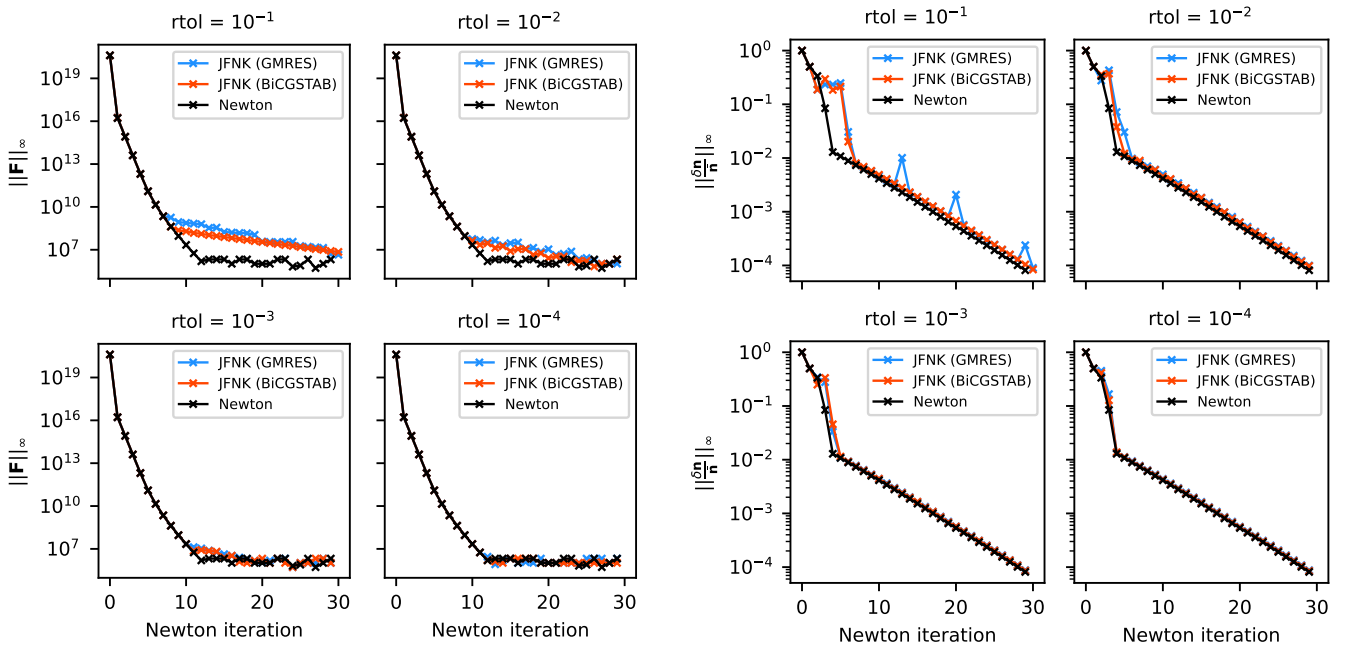
- Smaller tolerances result in more precise estimations of the inverse of the Jacobian. The JFNK method therefore requires less Newton-Raphson iterations but the overall number of calls to  $\mathbf{F}$  is nonetheless higher. This is due to the extra accuracy not being impactful enough on the convergence of the JFNK method.
- Higher tolerances result in coarse estimations of the inverse of the Jacobian. Even though the number of calls to  $\mathbf{F}$  is reduced, Newton-Raphson iterations usually yield poorer corrections. Therefore, the JFNK method requires extra Newton-Raphson iterations to converge. Overall, the solver will require more calls to  $\mathbf{F}$ .
- The optimal range thus consists in a trade-off between the accuracy of the inverse of the Jacobian and the calls to  $\mathbf{F}$  needed to obtain them.

We note that overly coarse estimations of the inverse of the Jacobian can yield an unstable behavior throughout Newton-Raphson iterations. We have found a few situations in which this feature can be helpful to escape potential local minima of the residual vector, and the method can even converge in less calls to  $\mathbf{F}$ . But more likely, such inaccurate corrections will not lead to the convergence of the JFNK method. For the sake of stability, it is recommended to use a Krylov relative tolerance smaller than  $\sim 10^{-2}$ .

Figure 5 (top) highlights for the 3-level Ca II setup an optimal range spanning from  $r_{\text{tol}} \sim 3 \times 10^{-4}$  to  $\sim 3 \times 10^{-2}$ . In this range, the JFNK (GMRES) solver outperforms the RH92 solver whereas the JFNK (BiCGSTAB) solver outperforms the latter in a much narrower range. It is also possible to witness the irregular convergence behaviour of the JFNK (BiCGSTAB) solver as the corresponding curve is less smooth than the JFNK (GMRES) solver one. In the case of a 6-level setup (Fig. 5 bottom), one may notice there is no such optimal range. Instead, the number of residual vector calls decreases almost monotonically with the Krylov solver tolerance. Both JFNK solvers outperform the RH92 solver for Krylov tolerances higher than  $\sim 3 \times 10^{-3}$  (GMRES) and  $\sim 1 \times 10^{-2}$  (BiCGSTAB). The JFNK (GMRES) solver has always proven to outperform the JFNK (BiCGSTAB) solver for most Krylov tolerances and various setups, as well as outperforming the RH92 solver in a wider range of Krylov tolerances than the BiCGSTAB counterpart. Note however that the residual vector calls from the JFNK / Newton-Raphson solvers and the iterations of the RH92 solver are different. Therefore Fig. 5 only makes sense if both are comparable in operations or execution time, which is the case for our setups (see §3.3). Thus, one can deduce that the Jacobi preconditioner allows the JFNK method to be more efficient than the RH92 one when used optimally. Although, the preconditioner is less efficient for a setup consisting of many frequencies such as the 6-level Ca II or H I ones. Moreover, the lack of an optimal range of Krylov tolerances points out that the Jacobi preconditioner is not enough for such setups. This statement also include every setup with strong scattering.

**Table 1.** A summary of the different setups used in this paper. The lines of each setup are indicated as well as the number of frequency points used per line. The Ca II setups consists of the K line (3 levels) and the H, K as well as the IR triplet lines (6 levels). The H I setup consists of a set of Lyman, Balmer, Paschen and Brackett lines.

Configuration	Lines wavelength [ $\text{\AA}$ ]	Frequency points
Ca II (2 levels + continuum)	3934	100
Ca II (5 levels + continuum)	3934 - 3968 - 8498 - 8542 - 8662	100 - 100 - 40 - 40 - 80
H I (5 levels + continuum)	Ly( $\alpha,\beta,\gamma,\delta$ ) Ba( $\alpha,\beta,\gamma$ ) Pa( $\alpha,\beta$ ) Br $\alpha$	(150,50,20,20) (70,40,40), (20,20), 20



**Fig. 4.** Comparison of the Newton-Raphson method with JFNK routines for the 6-level H I setup (zero radiation initial guess). As the Krylov solver relative tolerance  $r_{\text{tol}}$  becomes small, the discrepancy between the different methods reduces to truncation and round-off errors.

### 3.3. Performance of the solver

Table 2 shows the average execution time per call to  $\mathbf{F}$  (or equivalent for the RH92 solver) for different setups. The pure call to  $\mathbf{F}$  is always slightly faster to execute by the JFNK solvers than the RH92 solver equivalent. This is due to the RH92 solver's method itself requiring the computation of cross-coupling terms and the rest of the rate matrix elements in order to update the population densities. On the other hand, computing the residual vector and updating the preconditioner (JFNK) requires approximately twice the time of a pure  $\mathbf{F}$  call (by a JFNK solver), which was expected. The preconditioner update call, while more time consuming than the RH92 solver equivalent, is only performed once per Newton-Raphson iteration. The main contribution to the execution time is due to the Krylov solver calls, therefore pure residual vector estimations. As a result it can be shown that the JFNK solver calls, as implemented in our proof-of-concept code, do require slightly less time on average than RH92 calls even for extreme suboptimal Krylov tolerances.

In the following part, we compare the quality of the solutions provided by JFNK solvers with the reference RH92 case. The convergence condition is usually given by a sufficiently small change in the population densities. For that purpose, we use the JFNK residual norm  $\|\mathbf{F}\|_{\infty}$  as the metric because it is derived from the raw equations we are attempting to solve, although the RH92 solver is not designed to minimize the raw residual norm. Moreover, the residual norm might stall when using RH92 because of the deepest layers of the atmosphere. Indeed,

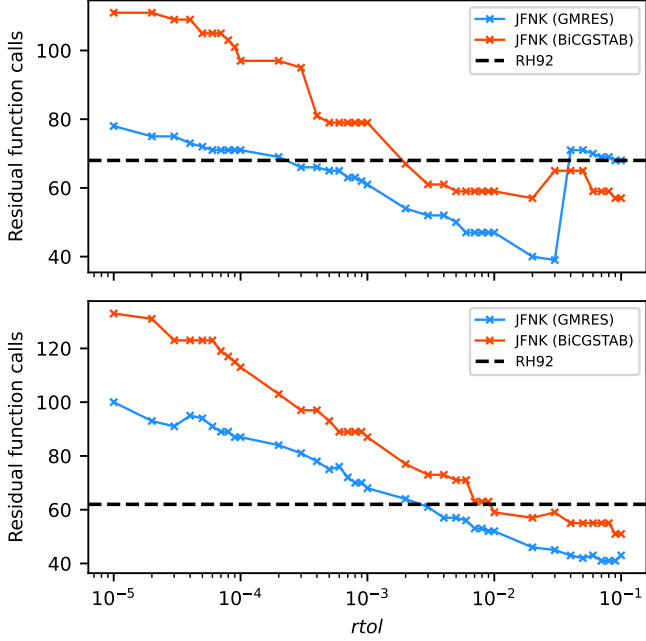
the medium becomes strongly collisional, therefore the radiative contribution and the changes that may occur during the solving process do not have a significant impact. Nevertheless, we can disregard such layers in the estimation of the residual norm because they are close to LTE. In the following, the residual norm is evaluated considering only the first 50 points ( $z > 700$  km) of our atmosphere where the chromosphere is located. Not doing this leads to a very large error in the RH92 curve, mostly driven by deeper layers where LTE should hold. Such behavior was not present in the JFNK calculations.

Figures 6 and 7 points out this clamped residual norm as well as the population change norm for the Ca II and the H I setups respectively. It is clear that the RH92 solver displays a smaller convergence rate for the biggest part of the solving process. JFNK solvers on the other hand are outperformed at the beginning before the convergence rate surpasses the one of RH92 (Ca II) or equals the latter (H I). As an outcome, JFNK solvers can be better performing than the RH92 one, especially if the initial guess is close enough to the solution. Moreover, both figures show that the solution provided by the JFNK solvers is a hundred times more precise than the RH92 one. One should keep in mind that the success condition is dictated by the population change norm and not by the residual norm.

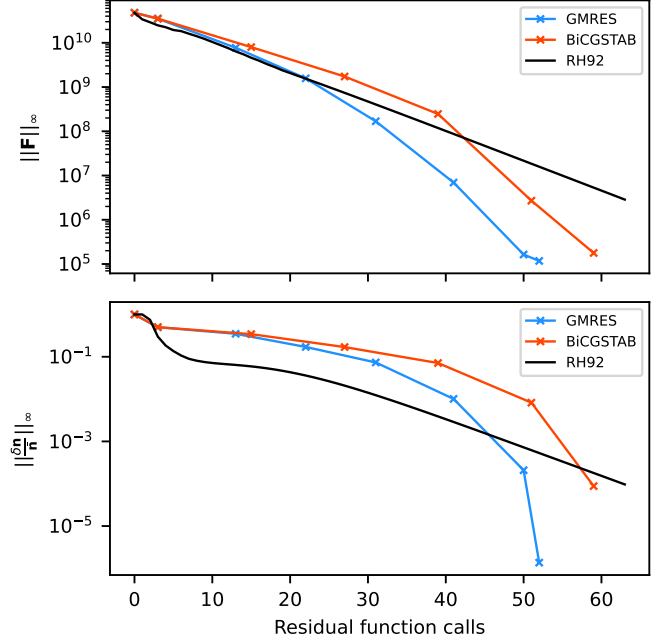
The size of the population change norm is not a good criterion for convergence. Figure 8 shows that the maximum error in the rate equations for the JFNK solver is lower than in the RH92 case for a same correction size. The JFNK solver achieves

**Table 2.** Average execution time per call for the JFNK and the RH92 solvers. Each call consists of the operations required to update the population densities once (and optionally the preconditioner for the JFNK solvers). Note also that all codes were not designed to be optimal. The JFNK solvers are slightly more performant than the RH92 solver (without the preconditioner update). Runs were executed on a MacBook Pro provided with a Apple M1 Pro chip.

Setup	JFNK [ms / call]	JFNK (with preconditioner) [ms / call]	RH92 [ms / call]
Ca II (2 levels + continuum)	55	121	65
Ca II (5 levels + continuum)	254	672	310
H I (5 levels + continuum)	510	1160	631



**Fig. 5.** Residual vector calls required for convergence as a function of the Krylov solver relative tolerance. Top: 3-level Ca II setup. Bottom: 6-level Ca II setup. Both setups use the LTE initial condition.



**Fig. 6.** Residual and population change norms during the solving process of the 6-level Ca II setup. The initial population densities are the LTE ones. Both JFNK solvers are using a Krylov relative tolerance of 10<sup>-2</sup>. Each cross marker corresponds to a Newton-Raphson iteration.

a lower absolute error in the residual norm than RH92 for any given convergence condition set on the size of the population change norm. This is not entirely unexpected, as the size of the correction per iteration is affected by the efficiency of the solver: for example, in the extreme case of traditional lambda iteration leads to very small corrections with a very large error in the rate equations (i.e. residual norm), whereas in accelerated lambda iteration the convergence is comparatively more efficient.

Finally, we provide in Fig. 9-10 the spectra of the 6-level Ca II and 6-level H I setups respectively. In both cases, the emerging spectra predicted by the RH92 and the JFNK solvers are essentially identical. We note that the extra accuracy of the solution provided by a JFNK solver does not yield noticeable changes in the emerging intensity, and we can therefore decide to terminate the solving process earlier and still output a similar result.

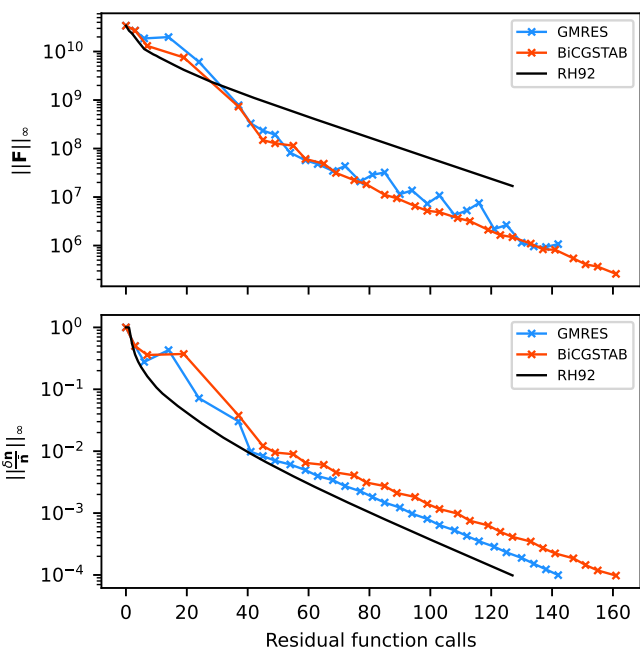
### 3.4. Calculations with velocity gradients

In this section we show that the JFNK solver can properly handle non-static atmospheres with velocity gradients as a function of depth. We have modified the FAL-C model atmosphere introducing a sharp velocity gradient around  $z = 1000$  km, corresponding to the lower chromosphere. Velocity gradients can be problematic if the velocity jump between consecutive grid cells is larger than approximately one third of the Doppler width (Ibgui et al.

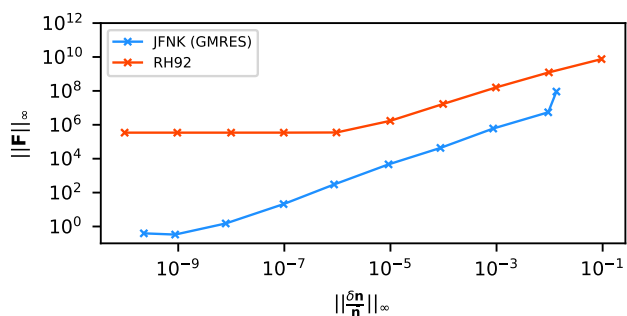
2013). Under those circumstances, the discretization of the radiative transfer equation can lead to artifacts in the intensity.

In order to avoid numerical artifacts in the calculation of the intensity, we have performed a depth optimization by placing more points where gradients in temperature, density, optical depth or line-of-sight velocity are large. All quantities were interpolated to the new depth grid by linear interpolation. The total number of depth points was kept equal to that in the original model. This method is essentially an extension of the depth-optimization included in the Multi code (Carlsson 1986), which now also accounts for the presence of velocity gradients. The upper left panel in Fig. 11 illustrates the artificial velocity gradient represented in the optimized grid.

If the velocity gradient is properly sampled with sufficient depth points, there is no fundamental reason why any of the algorithms would perform very differently than in the static case. Our convergence plots in Fig. 11 show a very similar behaviour than those in Fig. 6 for the Ca II atom. After a few iterations, the residual norm  $\|F\|_\infty$  is lower than in the RH92 curve, whereas the population change norm  $\|\delta n/n\|_\infty$  is larger. After approximately 80 iterations, the RH92 method has achieved a convergence (in the residual norm) that is similar to that of the GMRES after 50 iterations.



**Fig. 7.** Residual and population change norms during the solving process of the 6-level H I setup. The initial population densities are the zero radiation initial guess ones. Both JFNK solvers are using a Krylov relative tolerance of  $10^{-2}$ . Each cross marker corresponds to a Newton-Raphson iteration.



**Fig. 8.** Residual norm of the rate equations (6-level H I setup) as a function of the population change norm for the JFNK (blue) and RH92 (red) schemes. The Krylov relative tolerance was set to  $10^{-2}$ . The population densities were initialized using the zero radiation approximation. Each solver was run in order to achieve several Newton relative tolerances in the population change norm (e.g.,  $10^{-1}$ ,  $10^{-2}$ ). We then recorded the final residual and population change norms.

The emerging intensity spectrum shows now strong asymmetries around the core of all chromospheric lines, which become progressively more blue-shifted by the presence of the positive velocity gradient at the base of the chromosphere. The Ca II H&K lines (3968 Å and 3934 Å) show the well known enhancement of one of the k2 peaks (in the case on the red wing), as the blue-shifted line profile in the core frequencies leaves an opacity gap in the red wing of the line where photons can escape more efficiently compared to the static case (Scharmer 1984).

### 3.5. Prospects

The proposed JFNK method can be upgraded for a better efficiency and there are several ways of doing so. The external

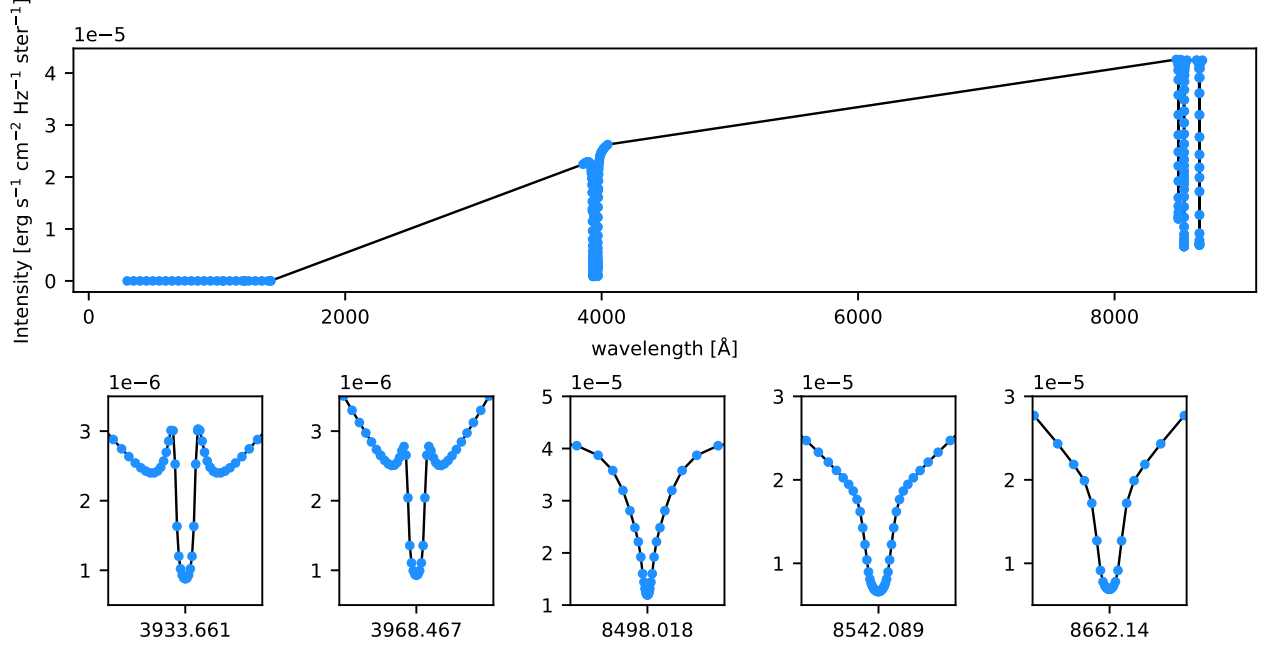
Newton-Raphson update does not leave much space for improvement, however it might be interesting to implement a continuation or a line search method to potentially reduce the number of Newton-Raphson iterations. A nice survey of continuation methods is given in Allgower & Georg (1993). A potentially simple but robust modification would be to implement a hybrid solver mixing the RH92 and the JFNK solvers. Starting with a few RH92 iterations before switching to JFNK ones would allow this hybrid solver to avoid the usual Newton methods deficiencies, as well as providing a better initial guess for the Newton-based solver. Such behaviour was encountered when attempting to solve the problem for instance with the 6-level H I setup starting with the LTE population densities. The performance of our JFNK solvers otherwise reduces to how efficient the Krylov solver can be, therefore dictated by the quality of the preconditioner and the solver itself.

The Jacobi preconditioner used in the present study has proven to be relatively inefficient in several of our setups and therefore it could be improved. In our implementation, the local preconditioner is a block-diagonal matrix. When it multiplies the Jacobian, it destroys the non-local derivatives in the left-hand side of Eq. 35, and therefore it has a similar effect as the adoption of a local approximate operator in the RH92 method. However, one should bear in mind that the preconditioner should be kept to be easily invertible and calculable thus leaving a narrow space for improvement. For that matter we provide two possible routes to calculate a more suitable preconditioner. The first option deals with the single point quadrature approximation of the RTE from Scharmer & Carlsson (1985). An approximation of this kind could greatly simplify the calculations of the non local part of the Jacobian matrices therefore providing a more accurate preconditioner than the Jacobi one. The second, more related to the JFNK formalism is presented in Chen & Shen (2006) and deals with an adaptive preconditioning technique. Shortly, one can take advantage of the matrix vector products calculated by a Krylov solver to iteratively update the preconditioner. It also allows the computation of a non local contribution to the preconditioner.

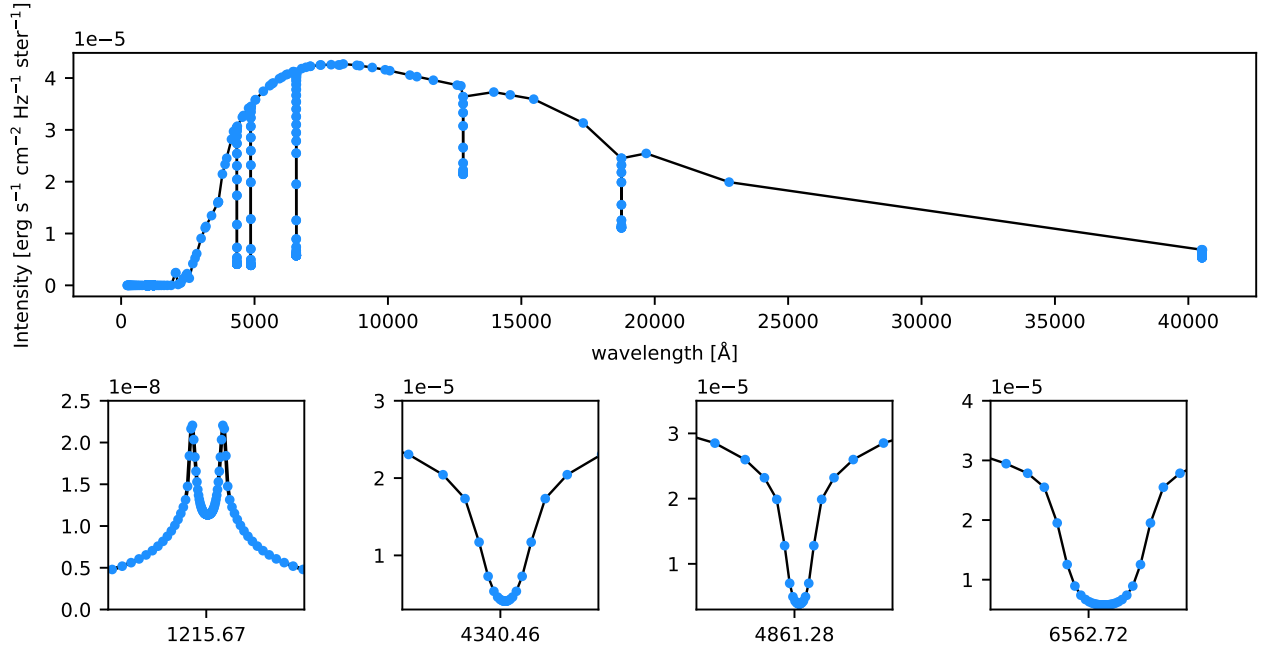
Another upgrade one may implement deals with the initial guess provided to the JFNK solver. In this paper, we used two possible initial guesses which are the LTE and the zero radiation ones. However, there might be other possibilities more suited for a Newton-based method applied to the radiative transfer problem such as the JFNK one. For instance, the population densities can be initialized with the ones derived from the escape probability theory (e.g., Hubeny & Mihalas 2014; Judge 2017).

In this paper we have only considered 1D plane-parallel NLTE problems. The extension to 3D geometry could be possible with some considerations. First, 3D radiative transfer codes are usually domain-decomposed for parallelization purposes (Leenaarts & Carlsson 2009), where each processor or machine has only access to the properties of the atmosphere, opacities, emissivities and population densities within one subdomain. In order to implement the inner Krylov solver we would need to collect all population densities from all subdomains and keep the vector basis of the Krylov subspace in one *manager* task. The key part is the evaluation of Eq. 33, which applies a perturbation to the population densities over the entire domain. The manager would need to propagate the relevant perturbed population densities to each subdomain, but the calculation of  $\mathbf{J}$  can be done the same domain-decomposed way. The cost is one extra communication from the manager to the worker tasks per Krylov iteration. By today memory standards in high-performance computing centers, this approach should be reasonably doable.





**Fig. 9.** Ca II (6 levels) spectrum. In black: output from the RH92 solver. In blue: output from the JFNK solver (GMRES) with a Krylov tolerance of  $10^{-2}$ .

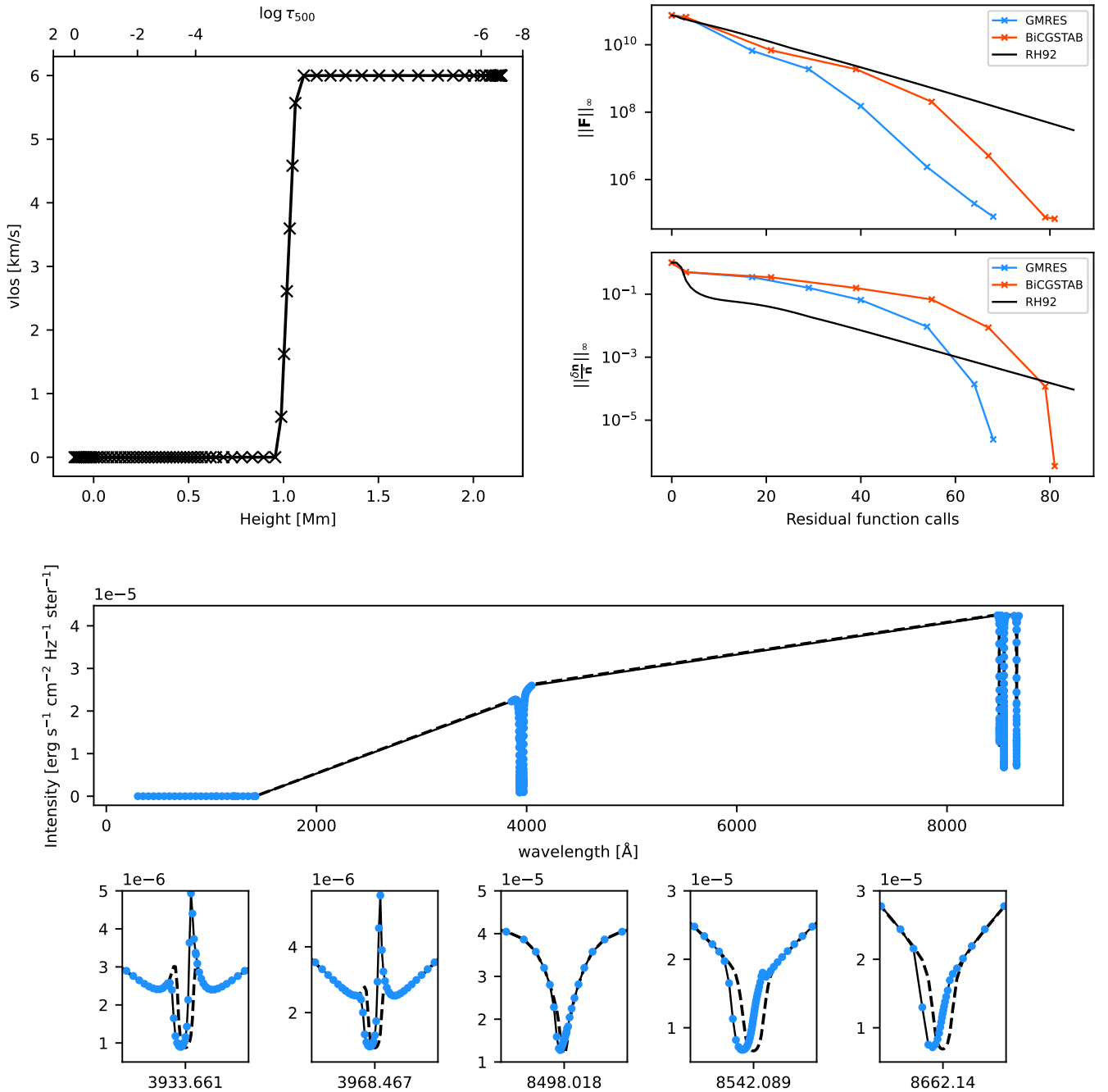


**Fig. 10.** 6-level H I spectrum. In black: output from the RH92 solver. In blue: output from the JFNK solver (GMRES) with a Krylov tolerance of  $10^{-2}$ .

#### 4. Conclusion

We present a Jacobian-Free Newton-Krylov method to solve the multi-level NLTE radiative transfer problem assuming statistical equilibrium. Our implementation shows a similar convergence as the Newton-Raphson method, without ever building the full Jacobian matrix explicitly. We have solved the NLTE problem assuming plane-parallel geometry and the FALC model for a 3-level Ca II as well as 6-level Ca II and H I atoms, which have been

commonly used in solar physics applications. In this study, we have shown that our solver can converge faster than other methods based on linearization, such as RH92. The improvement in the convergence rate is atom dependent, but it is usually a factor 1.5 – 2 in the best cases. The latter is evaluated in terms of the number of formal solutions needed to converge the problem. However, we note that the JFNK formal solutions are faster as no cross-terms summations are required compared to RH92. The downside to our method is that it relies on an appropriate elec-



**Fig. 11.** Velocity gradient convergence test for the different solvers. Top left panel: the line-of-sight velocity profile used for the test. Top right panel: the associated convergence plot for the 6-level Ca II setup with a Krylov tolerance of  $10^{-2}$  and initial LTE population densities. Note that all the solvers require more iterations to converge than in the case shown in Fig. 6. Bottom panel: converged spectrum including the velocity gradient for RH92 (black) and JFNK (GMRES) (blue). A JFNK (GMRES) velocity-free reference spectrum (dashed black) is also shown. There is clearly a blue shift occurring near the lines center due to the positive velocity gradient at the base of the chromosphere, resulting in very asymmetric output lines.

tion of the convergence tolerance for the Krylov inner solver. Our sensitivity study seem to indicate that an optimal performance can be attained when the tolerance is set in the range of  $10^{-3} - 10^{-2}$ .

In order to increment the accuracy of the Newton-Raphson correction per iteration, we have augmented the precision of the formal solver using complex numbers. This change was required given the enormous dynamic range of the atomic level population densities from the photosphere to the transition region.

Compared to other studies that have used Krylov-subspace methods to iteratively solve the linear 2-level atom problem (e.g., Hubeny & Burrows 2007; Anusha et al. 2009; Janett et al. 2024), our method handles multi-level non-linear problems. Given that the Jacobian matrix does not need to be explicitly computed in each iteration, this method becomes particularly interesting for more complex problems, which we briefly discuss hereafter as future prospects.

The more obvious application relates to problems where partial redistribution effects of scattered photons are important. While several efficient solutions are available for the 2-level atom problem (e.g., Scharmer 1983; Paletou & Auer 1995), similar methods for multi-level problems suffer from important limitations. For example, Hubeny & Lites (1995) presented a PRD method based on the complete linearization approach of Scharmer & Carlsson (1985), which does not consider overlapping active transitions. Uitenbroek (2001) overcomes that limitation by using the RH92 formalism and performing two iterative cycles, separating the correction to the atomic level population densities and the correction to the emissivity profile. The method converges, but it requires several evaluations of the redistribution integral per iteration. The method presented in this manuscript shows great potential to accelerate the convergence of PRD problems, since it does not require any explicit linearization of the problem.

Another extension could be the inclusion of charge conservation when H atoms are solved. The idea would be to add another conservation equation and update the electron density in each iteration as the ionization of H is usually dominant in the chromosphere. Previous studies have included such corrections, but needed to perform Newton-Raphson iterations due to the non-linear dependencies of the Saha equation and the rate equations with the electron density (e.g., Leenaarts et al. 2007; Bjørgen 2019). Since we do not perform any explicit linearization of the rate equations or the transfer equation, and we are already using Newton-Raphson iterations, the inclusion of charge conservation could be very efficient and relatively straight-forward.

*Acknowledgements.* We are very thankful to the referee for his/her constructive suggestions and careful evaluation of our manuscript. The Institute for Solar Physics is supported by a grant for research infrastructures of national importance from the Swedish Research Council (registration number 2021-00169). JL acknowledges financial support from the Swedish Research Council (VR, project number 2022-03535). No animals were harmed in the making of this manuscript. This project has been funded by the European Union through the European Research Council (ERC) under the Horizon 2020 research and innovation program (SUNMAG, grant agreement 759548) and the Horizon Europe program (MAGHEAT, grant agreement 101088184). Part of our computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725, at the PDC Center for High Performance Computing, KTH Royal Institute of Technology (project numbers NAISS 2023/1-15 and NAISS 2024/1-14).

## References

- Allgower, E. L. & Georg, K. 1993, *Acta Numerica*, 2, 1–64  
 Amarsi, A. M., Nordlander, T., Barklem, P. S., et al. 2018, *A&A*, 615, A139  
 Anusha, L. S., Nagendra, K. N., Paletou, F., & Léger, L. 2009, *ApJ*, 704, 661  
 Arnaud, M. & Rothenflug, R. 1985, *A&AS*, 60, 425  
 Auer, L. H. & Mihalas, D. 1969, *ApJ*, 158, 641  
 Barbier, D. 1943, *Annales d’Astrophysique*, 6, 113  
 Bjørgen, J. P. 2019, PhD thesis, Stockholm University  
 Burgess, A. & Chidichimo, M. C. 1983, *MNRAS*, 203, 1269  
 Cannon, C. J. 1973, *ApJ*, 185, 621  
 Carlsson, M. 1986, *Uppsala Astronomical Observatory Reports*, 33  
 Chen, Y. & Shen, C. 2006, *Power Systems*, *IEEE Transactions on*, 21, 1096  
 Dennis, J. E. & Schnabel, R. B. 1996, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (Society for Industrial and Applied Mathematics)  
 Dennis Jr, J. E. & Schnabel, R. B. 1996, *Numerical methods for unconstrained optimization and nonlinear equations* (SIAM)  
 Eddington, A. S. 1926, *The Internal Constitution of the Stars*  
 Fontenla, J. M., Avrett, E. H., & Loeser, R. 1993, *ApJ*, 406, 319  
 Hubeny, I. & Burrows, A. 2007, *ApJ*, 659, 1458  
 Hubeny, I. & Lites, B. W. 1995, *ApJ*, 455, 376  
 Hubeny, I. & Mihalas, D. 2014, *Theory of Stellar Atmospheres: An Introduction to Astrophysical Non-equilibrium Quantitative Spectroscopic Analysis*, Princeton Series in Astrophysics (Princeton University Press)

- Ibgui, L., Hubeny, I., Lanz, T., & Stehlé, C. 2013, *A&A*, 549, A126  
 Janett, G., Benedusi, P., & Riva, F. 2024, *A&A*, 682, A68  
 Judge, P. G. 2017, *ApJ*, 851, 5  
 Knoll, D. & Keyes, D. 2004, *Journal of Computational Physics*, 193, 357  
 Krylov, A. N. 1931, *Izv. Akad. Nauk SSSR Ser. Fiz.-Mat.*, 4, 491  
 Leenaarts, J. & Carlsson, M. 2009, in *Astronomical Society of the Pacific Conference Series*, Vol. 415, *The Second Hinode Science Meeting: Beyond Discovery-Toward Understanding*, ed. B. Lites, M. Cheung, T. Magara, J. Mariska, & K. Reeves, 87  
 Leenaarts, J., Carlsson, M., Hansteen, V., & Rutten, R. J. 2007, *A&A*, 473, 625  
 Leenaarts, J., Pereira, T., & Uitenbroek, H. 2012, *A&A*, 543, A109  
 Milić, I. & van Noort, M. 2018, *A&A*, 617, A24  
 Milić, I. & van Noort, M. 2017, *A&A*, 601, A100  
 Milne, E. A. 1921, *MNRAS*, 81, 361  
 Newton, I. 1736, *The Method of Fluxions and Infinite Series: With Its Application to the Geometry of Curve-lines*. By ... Sir Isaac Newton, ... Translated from the Author’s Latin Original Not Yet Made Publick. To which is Subjoin’d, a Perpetual Comment Upon the Whole Work, ... By John Colson, ... (Henry Woodfall; and sold by John Nourse)  
 Ng, K. C. 1974, *J. Chem. Phys.*, 61, 2680  
 Olson, G. L., Auer, L. H., & Buchler, J. R. 1986, *J. Quant. Spectr. Rad. Transf.*, 35, 431  
 Olson, G. L. & Kunasz, P. B. 1987, *J. Quant. Spectr. Rad. Transf.*, 38, 325  
 Osborne, C. M. J. & Milić, I. 2021, *ApJ*, 917, 14  
 Paige, C. C. & Saunders, M. A. 1975, *SIAM Journal on Numerical Analysis*, 12, 617  
 Paletou, F. & Auer, L. H. 1995, *A&A*, 297, 771  
 Pereira, T. M. D. & Uitenbroek, H. 2015, *A&A*, 574, A3  
 Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. 2002, *Numerical recipes in C++ : the art of scientific computing*  
 Raphson, J. 1690, *Analysis Aequationum Universalis Seu Ad Aequationes Algebraicas Resolendas Methodus Generalis, & Expedita, Ex Nova Infinitarum Serierum Methodo, Deducta Ac Demonstrata*  
 Rybicki, G. B. 1972, in *Line Formation in the Presence of Magnetic Fields*, 145  
 Rybicki, G. B. & Hummer, D. G. 1992, *A&A*, 262, 209  
 Saad, Y. & Schultz, M. H. 1986, *SIAM Journal on Scientific and Statistical Computing*, 7, 856  
 Scharmer, G. & Carlsson, M. 1985, *Journal of Computational Physics*, 59, 56  
 Scharmer, G. B. 1981, *ApJ*, 249, 720  
 Scharmer, G. B. 1983, *A&A*, 117, 83  
 Scharmer, G. B. 1984, in *Methods in Radiative Transfer*, 173–210  
 Scharmer, G. B. & Nordlund, Å. 1982, *Stockholms Observatoriums Reports*, 19  
 Shull, J. M. & van Steenberg, M. 1982, *ApJS*, 48, 95  
 Socas-Navarro, H., de la Cruz Rodríguez, J., Asensio Ramos, A., Trujillo Bueno, J., & Ruiz Cobo, B. 2015, *A&A*, 577, A7  
 Sukhorukov, A. V. & Leenaarts, J. 2017, *A&A*, 597, A46  
 Uitenbroek, H. 2001, *The Astrophysical Journal*, 557, 389  
 van der Vorst, H. A. 1992, *SIAM Journal on Scientific and Statistical Computing*, 13, 631

## Appendix A: Discretization coefficients

### Appendix A.1: Piecewise linear RTE

$$a^k = 1 - \frac{1 - e^{-\Delta\tau_{\mu\nu}^k}}{\Delta\tau_{\mu\nu}^k} \quad b^k = \frac{1 - e^{-\Delta\tau_{\mu\nu}^k}}{\Delta\tau_{\mu\nu}^k} - e^{-\Delta\tau_{\mu\nu}^k} \quad (\text{A.1})$$

### Appendix A.2: Derivatives of the intensity

For outgoing rays ( $\mu > 0$ ):

$$a_{\chi}^k = -\frac{S_{\mu\nu}^k}{\chi_{\mu\nu}^k} a^k + \frac{|z_{k+1} - z_k|}{2\mu} \left[ e^{-\Delta\tau_{\mu\nu}^k} (S_{\mu\nu}^{k+1} - I_{\mu\nu}^{k+1}) - \frac{S_{\mu\nu}^{k+1} - S_{\mu\nu}^k}{\Delta\tau_{\mu\nu}^k} b^k \right] \quad (\text{A.2})$$

$$b_{\chi}^k = -\frac{S_{\mu\nu}^{k+1}}{\chi_{\mu\nu}^{k+1}} b^k + \frac{|z_{k+1} - z_k|}{2\mu} \left[ e^{-\Delta\tau_{\mu\nu}^k} (S_{\mu\nu}^{k+1} - I_{\mu\nu}^{k+1}) - \frac{S_{\mu\nu}^{k+1} - S_{\mu\nu}^k}{\Delta\tau_{\mu\nu}^k} b^k \right] \quad (\text{A.3})$$

$$a_{\eta}^k = \frac{S_{\mu\nu}^k}{\eta_{\mu\nu}^k} a^k \quad (\text{A.4})$$

$$b_{\eta}^k = \frac{S_{\mu\nu}^{k+1}}{\eta_{\mu\nu}^{k+1}} b^k \quad (\text{A.5})$$

when  $k < N_z - 1$  otherwise one can set the coefficients to zero. For ingoing rays ( $\mu < 0$ ):

$$c_{\chi}^k = -\frac{S_{\mu\nu}^k}{\chi_{\mu\nu}^k} a^{k-1} + \frac{|z_{k-1} - z_k|}{2\mu} \left[ e^{-\Delta\tau_{\mu\nu}^{k-1}} (S_{\mu\nu}^{k-1} - I_{\mu\nu}^{k-1}) - \frac{S_{\mu\nu}^{k-1} - S_{\mu\nu}^k}{\Delta\tau_{\mu\nu}^{k-1}} b^{k-1} \right] \quad (\text{A.6})$$

$$d_{\chi}^k = -\frac{S_{\mu\nu}^{k-1}}{\chi_{\mu\nu}^{k-1}} b^{k-1} + \frac{|z_{k-1} - z_k|}{2\mu} \left[ e^{-\Delta\tau_{\mu\nu}^{k-1}} (S_{\mu\nu}^{k-1} - I_{\mu\nu}^{k-1}) - \frac{S_{\mu\nu}^{k-1} - S_{\mu\nu}^k}{\Delta\tau_{\mu\nu}^{k-1}} b^{k-1} \right] \quad (\text{A.7})$$

$$c_{\eta}^k = \frac{S_{\mu\nu}^k}{\eta_{\mu\nu}^k} a^{k-1} \quad (\text{A.8})$$

$$d_{\eta}^k = \frac{S_{\mu\nu}^{k-1}}{\eta_{\mu\nu}^{k-1}} b^{k-1} \quad (\text{A.9})$$

when  $k > 1$  otherwise one can set the coefficients to zero.



## Appendix B: Full Jacobian with background scattering terms

Let us recall Eq. 49 with different indices:

$$\left(\frac{\partial I_{j\nu}^i}{\partial n_r^\ell}\right) = \beta_r^\ell \left(\frac{\partial I_{j\nu}^i}{\partial \chi_{j\nu}^\ell}\right) + \gamma_r^\ell \left(\frac{\partial I_{j\nu}^i}{\partial \eta_{j\nu}^\ell}\right) + \sum_p \chi_{\text{sca}}^p \left(\frac{\partial J_\nu^p}{\partial n_r^\ell}\right) \left(\frac{\partial I_{j\nu}^i}{\partial \eta_{j\nu}^p}\right) \quad (\text{B.1})$$

If we develop the mean intensity term using the angular quadrature scheme with weights  $\omega_\mu$ , one can write:

$$\left(\frac{\partial I_{j\nu}^i}{\partial n_r^\ell}\right) = \beta_r^\ell \left(\frac{\partial I_{j\nu}^i}{\partial \chi_{j\nu}^\ell}\right) + \gamma_r^\ell \left(\frac{\partial I_{j\nu}^i}{\partial \eta_{j\nu}^\ell}\right) + \sum_p \chi_{\text{sca}}^p \left(\frac{\partial I_{j\nu}^i}{\partial \eta_{j\nu}^p}\right) \sum_q \omega_q \left(\frac{\partial J_{q\nu}^p}{\partial n_r^\ell}\right) \quad (\text{B.2})$$

Going any further requires to simplify the notations to keep as much clarity as possible. Let us define the following quantities:

$$M_{ij} = \left(\frac{\partial I_{j\nu}^i}{\partial n_r^\ell}\right) \quad (\text{B.3})$$

$$B_{ij} = \beta_r^\ell \left(\frac{\partial I_{j\nu}^i}{\partial \chi_{j\nu}^\ell}\right) + \gamma_r^\ell \left(\frac{\partial I_{j\nu}^i}{\partial \eta_{j\nu}^\ell}\right) \quad (\text{B.4})$$

$$Q_{ij}^k = \chi_{\text{sca}}^i \left(\frac{\partial I_{j\nu}^k}{\partial \eta_{j\nu}^i}\right) \quad (\text{B.5})$$

where we have omitted the indices  $r, \ell, \nu$ . From this point, we will no more mention nor write these indices, however it should be noted that the final solution should be computed for them as well. Equation B.2 can now be expressed as:

$$M_{ij} - \sum_p Q_{pj}^i \sum_q \omega_q M_{pq} = B_{ij} \quad (\text{B.6})$$

In the latter equation the unknowns one is seeking for are the coefficients  $M_{ij}$ . Equation B.6 can also be expressed as:

$$\mathbf{x}_i - \sum_p \langle \mathbf{x}_p, \mathbf{\Omega} \rangle (\mathbf{Q}^i)^\top \mathbf{e}_p = \mathbf{b}_i \quad (\text{B.7})$$

where  $\mathbf{x}_i = (M_{i1}, \dots, M_{iN_\mu})^\top$ ,  $\mathbf{b}_i = (B_{i1}, \dots, B_{iN_\mu})^\top$ ,  $\mathbf{\Omega} = (\omega_1, \dots, \omega_{N_\mu})^\top$  and  $\mathbf{Q}^i$  the matrix associated to the coefficients  $Q_{kl}^i$ . The vector  $\mathbf{e}_p$  is the  $p^{\text{th}}$  canonical basis vector. At this point, the unknowns are gathered into the vectors  $\mathbf{x}_i$ . It should be mentioned that if the background scattering is ignored,  $\mathbf{x}_i = \mathbf{b}_i$  and the solution is therefore simple. Otherwise, let us dot product Eq. B.7 with  $\mathbf{\Omega}$  to obtain:

$$r_i - \sum_p A_{ip} r_p = b_{\Omega,i} \quad (\text{B.8})$$

where  $r_i = \langle \mathbf{x}_i, \mathbf{\Omega} \rangle$ ,  $b_{\Omega,i} = \langle \mathbf{b}_i, \mathbf{\Omega} \rangle$  and  $A_{ij} = [\mathbf{Q}^i \mathbf{\Omega}]_j$ . The equation can also be written using matrix notations to obtain:

$$(\mathbf{I} - \mathbf{A})\mathbf{r} = \mathbf{b}_\Omega \quad (\text{B.9})$$

where  $\mathbf{I}$  is the identity matrix. This is a simple linear system of unknown  $\mathbf{r}$  for which the solution is:

$$\mathbf{r} = (\mathbf{I} - \mathbf{A})^{-1} \mathbf{b}_\Omega \quad (\text{B.10})$$

but can be simplified into:

$$\mathbf{r} \approx (\mathbf{I} + \mathbf{A})\mathbf{b}_\Omega \quad (\text{B.11})$$

if the background scattering is weak. Introducing this result in equation B.7 gives the final solution:

$$\mathbf{x}_i = \mathbf{b}_i + \sum_p r_p (\mathbf{Q}^i)^\top \mathbf{e}_p \quad (\text{B.12})$$

and if the background scattering is weak:

$$\mathbf{x}_i \approx \mathbf{b}_i + \sum_p b_{\Omega,p} (\mathbf{Q}^i)^\top \mathbf{e}_p \quad (\text{B.13})$$