Modelling Phosphorus II Emission with Recombination via Upgrading and Translation of Code SPECTRA from Fortran to C++

Undergraduate Research Thesis

Presented in Partial Fulfillment of the Requirements for Graduation *with Research Distinction* in Astronomy and Astrophysics in the Undergraduate Colleges of The Ohio State University

By

Kevin M. Hoy

Astronomy and Astrophysics Undergraduate Program

The Ohio State University

April 2022

Project Advisors:

Anil K. Pradhan

Sultana N. Nahar

Abstract

Phosphorus is a key element in DNA. As such, building accurate models of its spectroscopic emission, which is necessary for confirming its detection, could also serve an important role in the search for life in the Universe. The goal of this work is to improve current emission models of the five lowest lying, fine-structure levels of singly-ionized Phosphorus, which could theoretically be detected in exoplanetary atmospheres and star-forming nebulae. To achieve this, we translate the line emissivity ratios code SPECTRA from Fortran to C++, while also adding the process of electron-ion recombination to its modelling capabilities. We run this new code with two sets of input parameters: those present in current literature and those calculated with higher accuracy for this work. We have thus far found that the new code aligns exactly with the original when not including recombination, which demonstrates its accuracy, given the previously verified efficacy of the original. We have also found that our calculated input values and those from the literature produce significantly different results, while incorporating recombination has virtually no effect on transitions among the levels considered.

Acknowledgements

First, I would like to extend my sincerest thanks to my advisors, Dr. Anil Pradhan and Dr. Sultana Nahar. I began this project with almost no knowledge in this field, and the amount I've learned over this past year under your guidance continues to surprise even myself. From the core principles of atomic physics, to multiple spectroscopic codes, to the philosophy with which I should conduct research, you both have done so much to enable and inspire me to become a better scientist. I have not been in this field for long, but all of my experiences working with both of you have absolutely convinced me that I have made the right decision to pursue a career in astronomy. The lessons I have learned will carry on with me throughout my entire career, and you have my utmost gratitude.

Next, thank you to everyone in the Department of Astronomy who make SURP possible. This amazing program offers its students so much. Through it, I have gained research experience, the opportunity to present said research, and have made many friends in this field, all while receiving funding for all of this. The experience has been, and will continue to be, invaluable to me as I begin my career, and I am very thankful for all that the program has given me.

Thank you to Dr. Wayne Schlingman. Not only did he do amazing work as the director of SURP, he has provided an immense amount of support as I pursue graduate studies, and become a good friend. His help has enabled myself and many of my classmates to succeed as students and researchers, and he has my thanks for everything he's done for all of us.

I would also like to thank my friends and colleagues Mr. Michael Rothman and Mr. Bilal Shafique. Despite not being my formal advisors, you have both helped me and taught me so much over the course of this project. I couldn’t have come this far without peers like you, and I’m very glad to have known you. The three of us will be scattered to all corners of the Earth in these coming years, and I hope you continue to do for others what you’ve done for me, and that I can follow in your great example.

Thank you to the Ohio Supercomputer Center and the Arts and Sciences Technology Services’ Unity Cluster for access to your services. They have enabled me to work with some huge and exciting programs which have been instrumental in my work.

Finally, I’d like to extend my thanks to my parents, Dr. Doug and Dr. Cindy Hoy, who have greatly loved, supported, and encouraged me for so many years. They have given me so much, and have allowed me to have an amazing experience here at Ohio State. I wouldn’t be the person I am today without them, and I cannot thank them enough.

Table of Contents

List of Tables

List of Figures

1. Introduction

Phosphorus is a key element in DNA. Phosphates are what ultimately bind nucleotides together, which allows for the long chains necessary for the existence of complex life. Thus, it occupies a unique place in the search of life in the universe. It is not itself a biosignature, but its absence from a planetary atmosphere or stellar nebulae will indicate a present or future lack of Earth-like life on that planet or in that system. To aid in future efforts to detect the presence of phosphorus, we have calculated spectral line-intensity ratios for singly-ionized phosphorus, PII. This low level of ionization was chosen because it is likely to be abundant in the characteristically low-temperature plasmas we were examining.

To this end, we performed a complete translation of the code SPECTRA, which calculates the aforementioned spectral line ratios, from its original Fortran to C++, while also including the process of electron-ion recombination into the new version. This new version is intended to not only run more efficiently, but be more accessible to future modification as C++ continues to grow in popularity.

We calculated these line ratios using a wide variety of atomic data, including newly calculated data and that reported in literature, and we probe how our higher-precision data affects the final calculated results. We ultimately hope to use these calculations to confirm the detection of PII in Cassiopeia A, as reported in Koo et al. (2013).

2. Methods

In this section, we describe two processes significant to the completion of this work. First, we detail how the code combines input data to calculate level populations and how these level populations can be used to calculate line emissivity ratios. Then, we describe the methods used in the translation of the code from Fortran to C++.

2.1 Coefficient Calculation and Implementation

To calculate emission ratios, SPECTRA builds a coefficient matrix of all process that affect the equilibrium level populations of an ion, then solves that coefficient matrix for those level populations. Each element of the matrix, denoted as C_{ij}, when multiplied by the electron density N_e, contains all information on how the population of level j influences the population of level i. The code then combines those level populations with spontaneous emission to produce the desired line intensity ratios. In the following subsections, we detail the construction of this coefficient matrix.

2.1.1 Spontaneous Emission

The sole source of emission we are considering in this project is that from spontaneous de-excitation. This process is entirely quantified by the calculation of Einstein A coefficients, and are fed directly into SPECTRA as inputs, and are the first step in building the coefficient matrix. Thus, the coefficient matrix starts off as Equation 1, with C_{ij} being the coefficient matrix element and A_{ij} being the A coefficient for the transition between the two levels.

$$N_e C_{ij} = A_{ij} \tag{1}$$

Calculating these Einstein A coefficients is much less straightforward. To get them, we used the robust program SUPERSTRUCTURE (hereafter abbreviated to SS). SS is a very general atomic code that, for a given ion, calculates a wide variety of parameters in addition to A coefficients, including energy levels and wavefunctions, which are used elsewhere for other calculations. SS uses a Breit-Pauli approximation in its calculations, which means it includes the Breit interaction, relativistic mass corrections, the Darwin effect, and fine-structure splitting, but ignores other, less impactful relativistic effects (Nahar 2021). The atomic parameters used in this project are shown in Table 1, where they are compared to NIST values to illustrate their accuracy. While SS is capable of producing high-accuracy information, it requires highly tuned inputs to do so. At present, this fine tuning has not been performed. While this produces energy levels that are relatively close to those reported in NIST, the difference is still significant and the difference between the calculated and observed A values is much larger still. It is also the case that NIST does not presently report an A value for all transitions considered in the scope of this work. This should be considered when interpreting the accuracy of our final results, which depend on these calculations.

2.1.2 Collisional Excitation

The first of our means of population an excited state is collisional excitation between a free electron and one bound to a PII ion. Because electrons are much less massive than a PII ion, we ignore interactions between ions, as they interact much less frequently due to their significantly lower velocities.

To parameterize this interaction, we introduce a variable called collision strength. This collision strength is directly related to the probability of an electron-ion interaction producing an excited state in the ion, and are strongly a function of energy. This parameter has very strong resonant structures, so high precision calculations are necessary for values to be accurate.

For this project, these values were calculated by Dr. Sultana Nahar using R-matrix codes (Nahar, under preparation). The R-Matrix codes, much like SS, produce a wide variety of results, including the collision strengths and level-specific recombination rates used for this project. These values are shown in Figure 1.

To capture the full effect of collisional excitation on the population of an energy level, we must combine the number of interactions at a given energy with the relevant collision strength. This is much easier said than done for astronomical plasmas. It is virtually impossible to know the exact density of electrons at a given energy level at the high precision necessary to capture the resonant structures of collision strengths. Thus, we had to take a different approach.

While we may not know the exact energy distribution, we do know that, for a cloud of electrons at a given temperature, it can be closely approximated as a Maxwell-Boltzmann distribution. Using this fact, we can construct a new parameter, which we call the effective collision strength, that depends on the temperature of the plasma, rather than the energy of a given electron. Effective collision strength is also proportional to the *rate* at which excitations occur, rather than the probability one will occur, which makes it exactly suited to our purpose.

After transforming the collision strengths calculated by Dr. Nahar into the related effective collision strengths, we compare these results to those reported in

Tayal (2004). Our calculations were performed considering a higher number of states, which means they are overall more accurate than the literature values. They are plotted against each other in Figure 2. We performed the overall line ratio calculation using both collision strengths in order to compare those results.

These effective collision strengths are fed into the program as an input, for each possible transition and as a function of temperature. To calculate the effect collisional excitation has on population levels, we must simply multiply the effective collision strengths at a given temperature, Υ_{ij}, with the current electron density we're calculating the line ratios for. This product is added to the correlating A_{ij}, producing the current coefficient matrix element seen in Equation 2.

$$N_e C_{ij} = A_{ij} + N_e \Upsilon_{ij} \tag{2}$$

2.1.3 Electron-Ion Recombination

The largest introduction to the program was the addition of electron-ion recombination. Specifically for this project, we are considering an electron recombining with a PIII ion, producing a PII ion that we are considering. Now, this recombination can occur at any energy level, but the odds of recombining into crtain energy levels vary between levels and depend strongly on the temperature of the plasma. As such, level-specific recombination rates, noted as α_i, must be calculated for this process to be modelled. These values have been calculated by Dr. Sultana Nahar and are publicly available on her online NORAD database (Nahar, 2020).

Then, because this process can occur at any energy level, the newly added electron can cascade down to the levels we are considering in our calculation. Thus, we must also calculate how recombination into higher levels might impact the populations of lower lying levels. This is done with cascade coefficients, denoted here as D_{ji}, which are calculated using several Einstein A coefficients as seen in Equation 3 (Pradhan & Nahar, 2011).

$$D_{ji} = \frac{A_{ji}}{\sum_{i'<j} A_{ji'}} + \sum_{k>i} D_{jk} \frac{A_{ki}}{\sum_{i'<k} A_{ki'}} \qquad (3)$$

These cascade coefficients are related to the probability that an ion in an excited state j will go to the lower-lying state i via spontaneous emission, but allowing for any all allowed transitions for all states between i & j before reaching level i. These multiple emissions are hence the cascade that gives rise to this parameter's name.

The first term of the expression represents direct transition from the upper state to the lower state and is very simple to calculate. The second term represents the contributions for all possible cascades between the initial excited state and the final lower state. As you can see, this term includes other cascade coefficients, which means it is much more complicated than the first term, if not horribly complicated on its own. Because this term only includes cascade coefficients between *higher* levels than the lower level of the overall cascade coefficient, we simply need to calculate the cascade coefficients for the highest allowed i and j values and work our way down. This way, by the time we reach the lower coefficients which depend on the higher coefficients, they are available for the calculation. As we calculate the cascade coefficients in this manner, we ultimately produce a triangular matrix D_{ij} for all $j > i$.

This triangular matrix is still somewhat incomplete. Its intended use is to describe how recombination affects a given level, given a recombination rate at all levels, and thus far it only captures the effect that recombining to a higher level will have on a lower level. It does not include the effect of directly recombining into a given level, without the need for any further transitions. To incorporate this we simply have to populate the diagonal components of this matrix with ones, as recombining into a level directly must be 1:1 efficient in populating that level.

The overall effect of recombination between two levels is calculated by multiplying the level-specific recombination rate α_i of the higher energy state, the electron density N_e, and the matrix element D_{ij}. Thus, to complete the cascade coefficient, we combine the Einstein A coefficient, collisional excitation coefficient, and this new term to produce the final coefficient matrix, as seen in Equation 4.

$$N_e C_{ij} = A_{ij} + N_e \Upsilon_{ij} + N_e \alpha_i D_{ij} \tag{4}$$

Thus, with this contribution, we complete the interaction coefficient matrix, and it is ready to be solved for the level populations.

2.1.4 Stimulated Emission and Doppler Broadening

It is also worth noting that this program is configured to calculate the contribution of stimulated emission due to radiation from a background source and doppler line-width broadening. This component of the original code was translated and is included in the current version, and their individual components have been tested for accuracy, but they were not included in the overall testing and utilization of the program, as they are not pertinent to the present work. This functionality will be further tested and verified at a later date.

2.2 Solving for Level Populations and Line Ratios

With the coefficient matrix fully calculated, there are only a few steps remaining before arriving at the line ratios we are interested in. This coefficient matrix constitutes a series of linear equations for the equilibrium level populations. Luckily, we are not inherently interested in any innate level populations of the plasma being observed. We are instead interested in the level populations relative to each other. As such, we can set the relative population of the ground state to one, and solve for the other populations relative to it.

Setting the ground state to one also allows for this calculation. The matrix elements C_{1j} for any excited state j can be treated as the constants of the linear equation. Thus, we are solving for the relative level populations such that the population of the ground state is one and each level population satisfies their relation to the ground state. We are then left with the simple task of solving a series of linear equations to find the relative level populations.

Once the level populations have been acquired, the final line intensity ratios are calculated according to Equation 5. Here, ε_{ij} is the emissivity of the emission line, P_i is the population of the higher-energy state, A_{ij} is the Einstein A coefficient of the transition from level i to level j, and E_{ij} is the difference in energy between level i and j.

$$\frac{\varepsilon_{jk}}{\varepsilon_{mn}} = \frac{P_j A_{jk} E_{jk}}{P_m A_{mn} E_{mn}} \qquad (5)$$

The line intensity ratio is overall a ratio of the spontaneous emission rate, energy change, and upper level population. Formally, the numerator and denominator should both be understood as the total number of transitions that occur per unit time, multiplied by the energy of said transitions. However, given that we are solving for ratios, rather than raw emissivities, we need not actually know the total number of transitions occurring, just the relative number of transitions, and that information is contained entirely in the A values and relative level populations.

2.3 Translation of SPECTRA from Fortran to C++

To translate SPECTRA, we made use of the program's modular structure. There is very little work done in the main body of the program. Instead, it includes several independent functions, which are sequentially called by the main body. Thus, to translate the program, it is mostly a task of treating each of these functions as its own program that generates its specific output with a given input.

A complication to this is the frequent use of global variables, which means variables that are not explicitly passed to the function may be used or modified. This does not present a significant challenge, however. By building a base template that arbitrarily, but appropriately, populates the global variables present in the program in both Fortran and C++, we could call the function we intend to translate and compare the outputs. Once these outputs match, the first phase of translation is complete. It is also worth noting that each of the original functions produces an output file. While most of these do not contain any scientifically interesting information, they are invaluable for diagnostics and allowed for the creation of comparing Fortran and C++ results as a matter of course in the translation.

While this was a very efficient means of translating the code, it is far from perfect in terms of producing new functions that exactly map onto the original, even if the output files match. The inconsistencies that survived this process were nearly always a result of choices made during global variable population. For example, if there is a highly specific condition that must be handled in some way, but that condition does not arise in the arbitrary variable assignment, errors in the handling will go undetected until they arise in the final product.

Once this modular translation has been completed, we combined each of the functions back into a single program. We then translated the main body of the original program, which should have produced the full-fledged, functional C++ version of the program. However, as previously implied, we encountered several more bugs and errors after this step, as bugs which went unnoticed in the first pass of translation began to surface.

Again, the diagnostic outputs of every function were hugely important in this second phase of corrections. Because these outputs are produced in a specific order, they can be used to find what function first produced an inconsistent result. This significantly reduced the amount of time necessary to find and such bugs in the program. A recurring error in this translation effort, that will most likely be present in all future translations of this variety, is the handling of index variables. Fortran is an index-1 language while C++ is an index-0 language. This means that the first element in a two-dimensional array will be at (1,1) in Fortran, but at (0,0) in C++. Now, this problem could be fairly easily alleviated by exactly duplicating the process of the original code while changing the coordinate of every array call by -1, but this makes the program significantly more difficult to read and defeats some of the purpose of translation in the first place. Thus, we sacrificed a means of easing what was most likely the largest hurdle of the translation and favored the overall quality of the final product.

3. Results

There are three results we desired from this project. First, we wished to calculate PII line ratios that can be used to confirm detection of PII in observed spectra. Second, we sought to confirm the accuracy of the new C++ version relative to the original Fortran version, which has proven reliable for some time. Third, we wanted to see how incorporating recombination affects these line ratios. All of these ends can be met together by performing the calculations for PII and comparing the results between the original version, the translated version sans recombination, and the new final version incorporating recombination. The results of this can be seen in Figures 3 and 4. The former illustrates the accuracy of the new code relative to the original, while also probing the effects recombination has on this specific transition. The results from both codes are identical to a high degree of precision, and, as such, their lines exactly overlap. The latter plot displays line ratios for every transition among the 5 levels considered in this work. For those plots, we include calculations using both our effective collision strengths and those reported in Tayal (2004) while holding the temperature at a constant 10,000 K and varying electron density.

As you can see in Figure 3, despite 5 total calculations being performed, there are only two distinguishable lines in the plot; one using our calculated effective collision strengths and one for the literature values. On one hand, this does well to confirm the accuracy of the base code relative to the original. Inspection of the data itself reveals that the results are not only similar, they are exactly identical up to at least three significant figures, and are observed to match at much higher precision in some values. On the other hand, this demonstrates an overall lack of impact recombination has on these line ratios. The results do marginally differ, if recombination is considered, but only by fractions of a percent. The most probable explanations for this are either that recombination into these fine-structure levels is simply a very low-occurrence process, or that there is some error in implementing the effects of the process in the current iteration. Confirming which is the cause in this case will be a part of future work on the project.

4. Conclusions and Future Steps

In this project we performed something that has never been done before in our group: the translation and improvement of a Fortran code into C++. This has not only yielded a new iteration of SPECTRA that is as accurate as the original, it is more precise in its calculations, includes an entirely new physical process, and is more accessible to future improvement. This code will undergo extensive further testing, in order to verify its general utility and uncover any inaccuracies that did not arise in this specific calculation. We will especially be examining the implementation of recombination into the calculation, as it represents an important step forward in the physical accuracy of the code.

Once the results of this calculation have been verified, we will also be able to compare them to real, observed spectra to confirm the presence of PII in the object being observed. This key element to DNA could serve as a necessary prerequisite to life in a stellar system, and would thus serve as an important check in the confirmation of a detection of extraterrestrial life in the Universe.

References:

Koo BC, Lee YH, Moon DS, Yoon SC, Raymond JC. Phosphorus in the young supernova remnant Cassiopeia A. Science. 2013 Dec 13;342(6164):1346-8. doi: 10.1126/science.1243823. PMID: 24337291.

Nahar, S. N. (2021, October). *ATOMIC ASTROPHYSICS AND SPECTROSCOPY WITH COMPUTATIONAL WORKSHOPS ON SUPERSTRUCTURE AND R-MATRIX CODES* [Lecture series]. Computational Workshop,Columbus, OH, United States (Virtual).

Nahar, Sultana N., "Database NORAD-Atomic-Data for atomic processes in plasma", Atoms 8, issue 4, 68 (2020), (web: NORAD-Atomic-Data at https://norad.astronomy.osu.edu/)

Nahar, Sultana N., "Electron Impact Excitation Collision strengths of P II from R-matrix Calculations", (under preparation, 2022)

Pradhan, A. K., & Nahar, S. N. (2011). *Atomic astrophysics and spectroscopy*. Cambridge University Press.

Tables:

Spectroscopic Notation	SS Energy (Ryd)	NIST Energy (Ryd)	% Difference
$3s^23p^2\ ^3P_0$	0	0	-
$3s^23p^2\ ^3P_1$	0.00122650	0.0015027	20.24
$3s^23p^2\ ^3P_2$	0.00354104	0.0042749	18.78
$3s^23p^2\ ^1D_2$	0.10060374	0.0809416	21.66
$3s^23p^2\ ^1S_0$	0.20591896	0.1966116	4.624

Transition	SS A value (s^{-1})	NIST A value (s^{-1})	% Difference
$^3P_1 \rightarrow {}^3P_0$	4.384E-05	8.05E-05	59.00
$^3P_2 \rightarrow {}^3P_0$	2.382E-09	6.00E-09	86.33
$^3P_2 \rightarrow {}^3P_1$	2.210E-04	3.80E-04	52.91

Table 1. Comparison of SUPERSTRUCTURE calculations to trusted NIST values for Phosphorus II. Relatively small, but significant, errors in the energy calculation produce much larger errors in the A value calculation.

Figures:

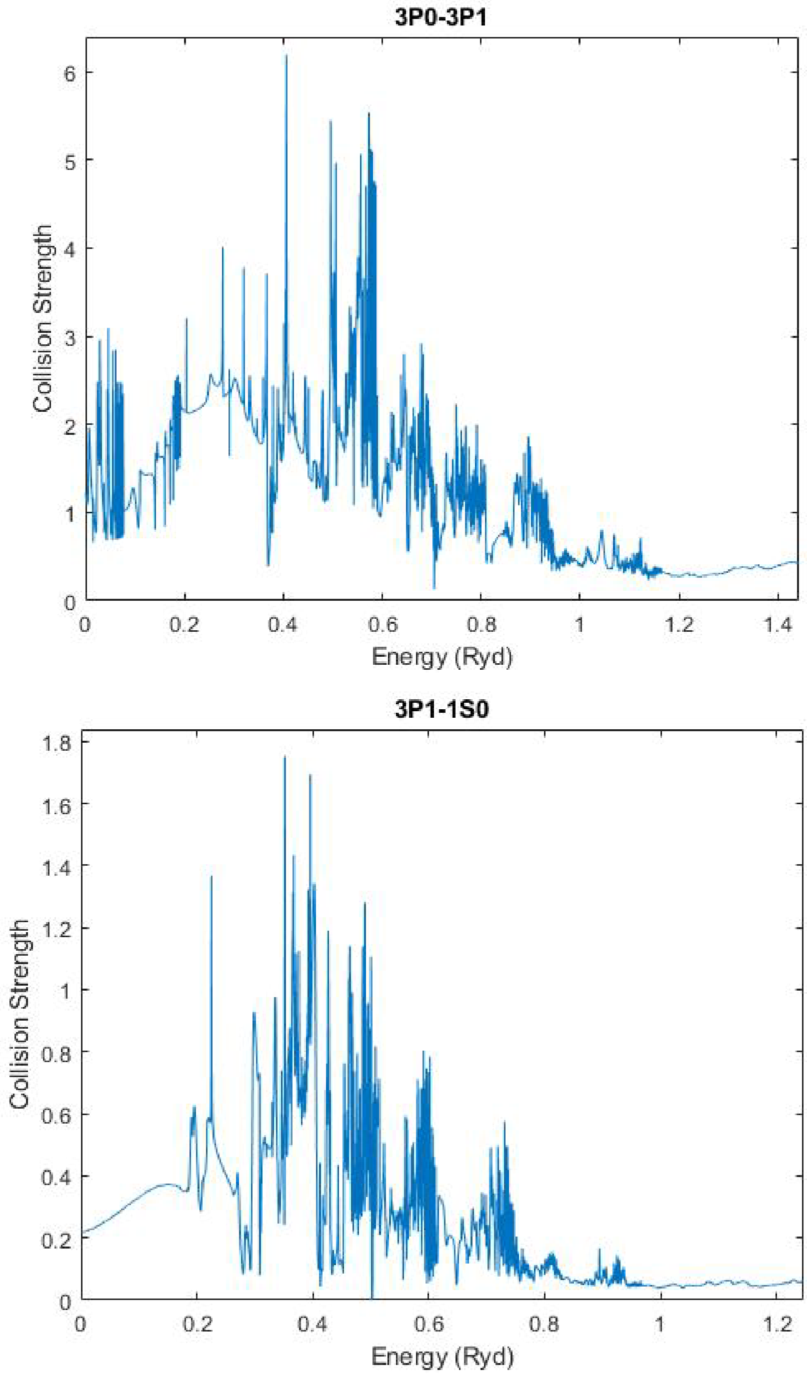

Figure 1. Select Collision Strengths calculated for this project for two transitions as a function of incident electron energy.

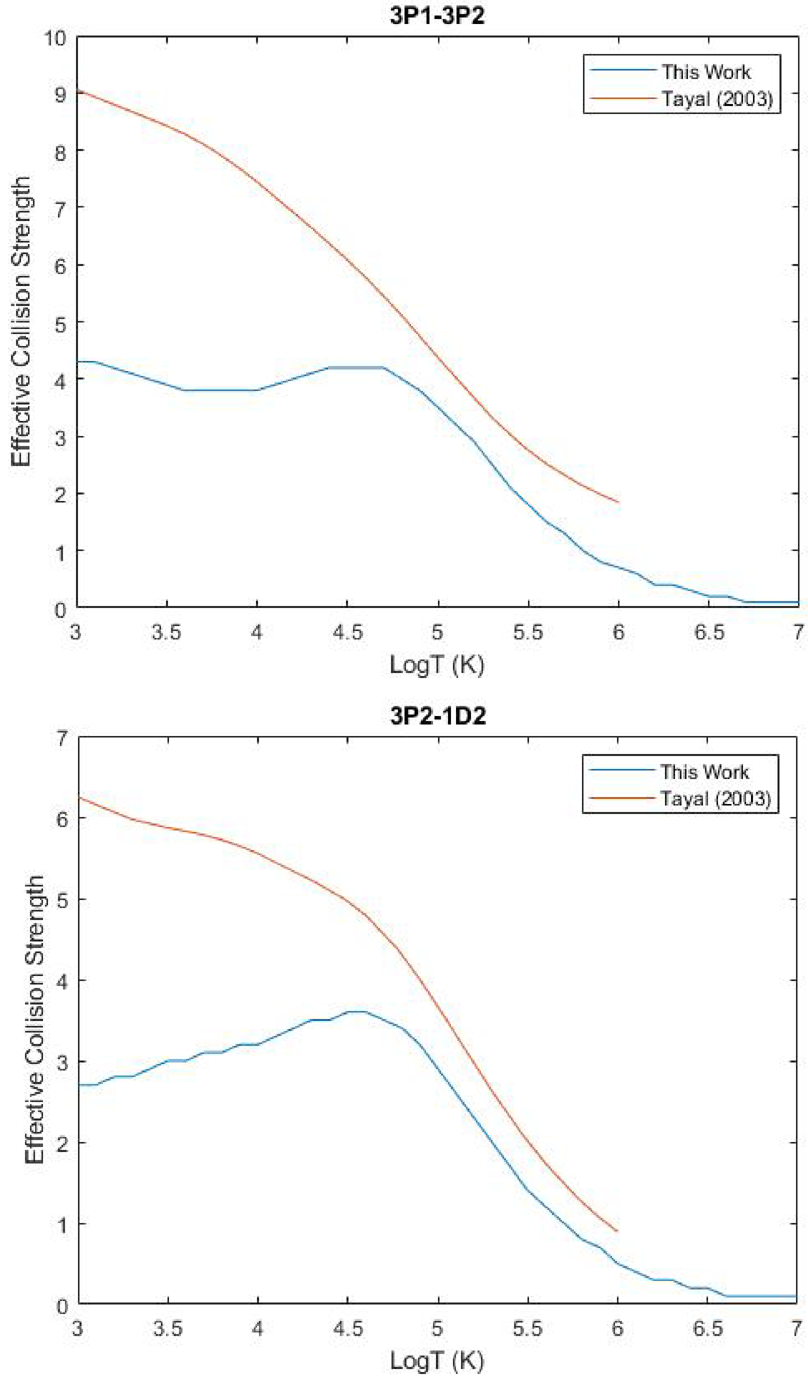

Figure 2. Select effective collision strength as a function of temperature comparison between this work's calculation and the literature values reported in Tayal (2004).

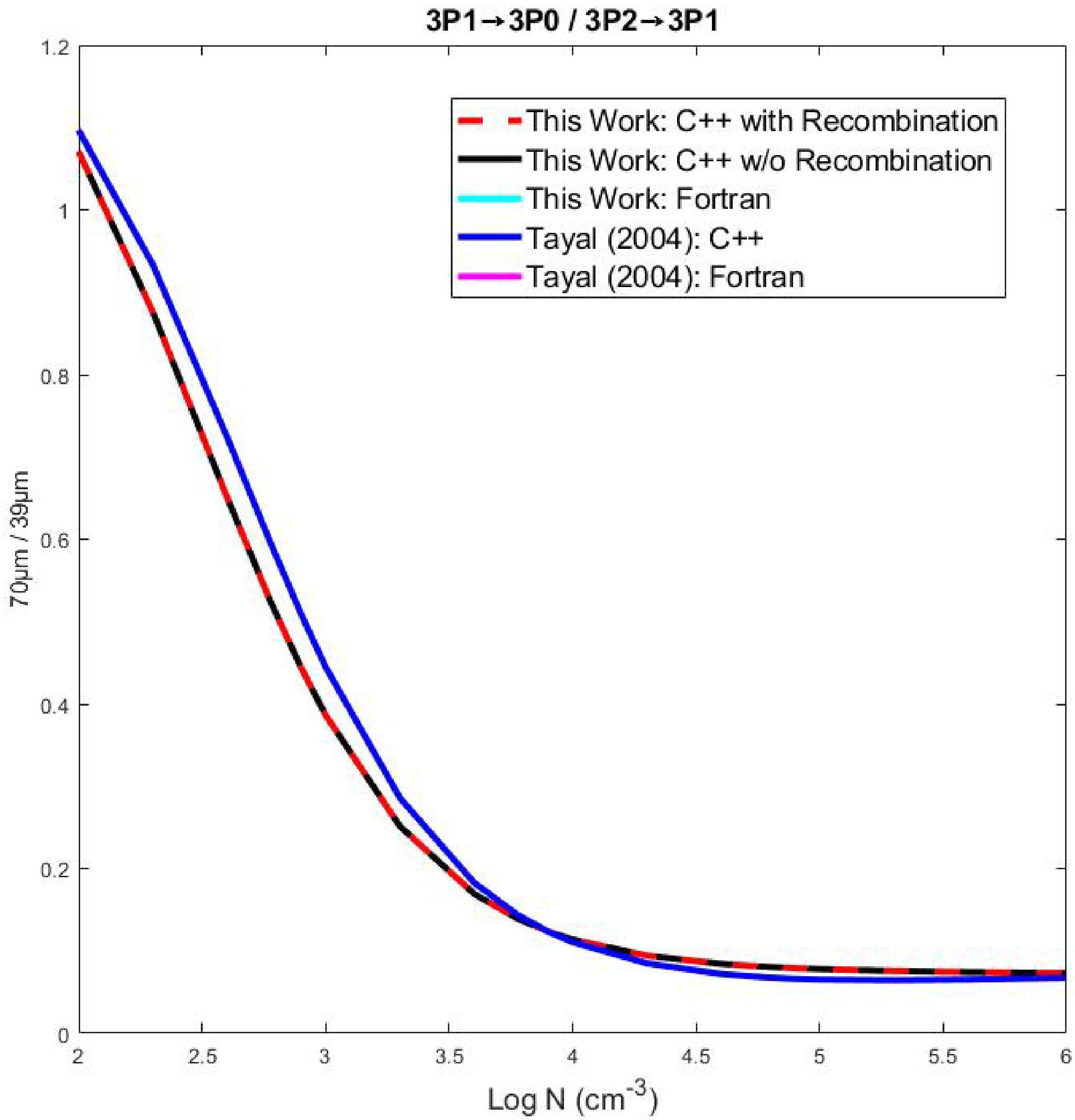

Figure 3. Single line ratio as a function of density at 10,000 Kelvin, including calculations from the original version of SPECTRA and our new one, both with and without recombination. This also compares the result of using our effective collision strengths and those from Tayal (2004).

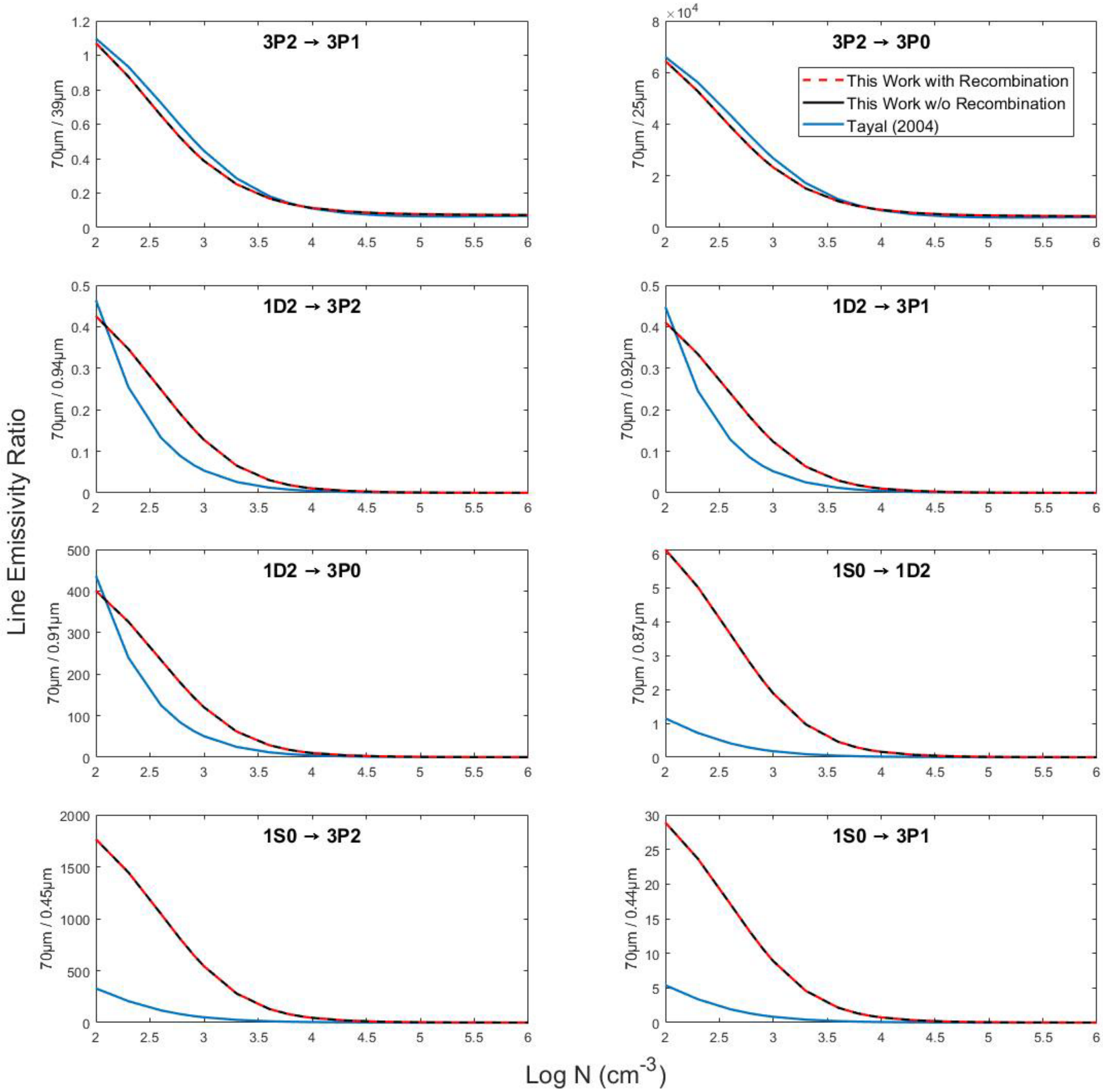

Figure 4. Line Ratios at 10,000 K for all transitions among the fine-structure levels considered in this work. Emission from transitions from the first excited state to the ground state are held constant in the numerator, with other lines being iterated in the denominator. We compare calculations performed with our effective collision strengths with those from Tayal (2004).

Presentations Based on this Work

***Modelling PII Emission to Aid in the Search for Life*, Hoy, K. M., Pradhan, A. K., Nahar, S. N., Fall 2021 Meeting of the Eastern Great Lakes Section, November 2021:** "Being the "backbone" element of DNA, Phosphorus is a key element in the search for life in the Universe. To aid in future search for Phosphorus in star-forming regions and thereby in exoplanets, we have constructed a line ratios emission model for PII. This low ionization state is likely to exist in gaseous nebulae with stellar formation and in exoplanetary atmospheres if those atmospheres contain any significant amount of P. There are also 3 primary lines in PII emission spectrum at 20, 30, and 60 μm that may be detected in the Far Infra-Red. These two facts make this specific ion especially useful in the search for DNA-based life in the universe."

***Collision Strengths and Line Ratios for PII as a Biosignature in Exoplanets,* Hoy, K. M., Pradhan, A. K., Nahar, S. N., 53rd Annual Meeting of the APS Division of Atomic, Molecular and Optical Physics, May-June 2022:** "Being the "backbone" element of DNA, Phosphorus is a key element in the search for life in the Universe. To aid in the future search for Phosphorus in star-forming regions and thereby in exoplanets, we have constructed a line ratios emission model for PII. This low ionization state is likely to exist in gaseous nebulae with stellar formation and in exoplanetary atmospheres if those atmospheres contain any significant amount of P. There are also 3 primary lines among the fine-structure $3s^2 3p^2$ transitions in the PII emission spectrum that we are particularly interested in: $^3P_2 \rightarrow {}^3P_0$ at 21 μm, $^3P_2 \rightarrow {}^3P_1$ at 33 μm, and $^3P_1 \rightarrow {}^3P_0$ at 60 μm. Collision strengths for these transitions have been newly calculated and compared the results of Tayal (2004). This level of ionization and especially useful lines make this specific ion especially useful in the search for DNA-based life in the universe. Our results have been found and verified against each other using two versions of the code SPECTRA: The original Fortran version, which has proven accurate and reliable, and a new, modernized C++ version that incorporates the process of electron-ion recombination in its calculations, further improving the code's accuracy."