Note

Modeling of particle transport, neutrals and radiation in magnetically-confined plasmas with Aurora

F Sciortino1,*, T Odstrcil1, A Cavallaro1, S P Smith2, O Meneghini2, R Reksoatmodjo1, O Linder1, J D Lore3, N T Howard1, E S Marmar1 and S Mordijck1

1 Massachusetts Institute of Technology, Cambridge, MA 02139, United States of America
2 General Atomics, PO Box 85608, San Diego, CA 92186-5608, United States of America
3 Department of Physics, College of William & Mary, Williamsburg, VA, United States of America
4 Max-Planck-Institut für Plasmaphysik, Boltzmannstraße 2, D-85748 Garching, Germany
5 Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States of America

E-mail: sciortino@psfc.mit.edu

Received 5 June 2021, revised 7 July 2021
Accepted for publication 20 September 2021
Published 13 October 2021

Abstract

We present Aurora, an open-source package for particle transport, neutrals and radiation modeling in magnetic confinement fusion plasmas. Aurora’s modern multi-language interface enables simulations of 1.5D impurity transport within high-performance computing frameworks, particularly for the inference of particle transport coefficients. A user-friendly Python library allows simple interaction with atomic rates from the Atomic Data and Atomic Structure database as well as other sources. This enables a range of radiation predictions, both for power balance and spectroscopic analysis. We discuss here the superstaging approximation for complex ions, as a way to group charge states and reduce computational cost, demonstrating its wide applicability within the Aurora forward model and beyond. Aurora also facilitates neutral particle analysis, both from experimental spectroscopic data and other simulation codes. Leveraging Aurora’s capabilities to interface SOLPS-ITER results, we demonstrate that charge exchange is unlikely to affect the total radiated power from the ITER core during high performance operation. Finally, we describe the ImpRad module in the one modeling framework for integrated task framework, developed to enable experimental analysis and transport inferences on multiple devices using Aurora.

Keywords: particle transport, neutrals, radiation, superstaging, integrated modeling, ITER, spectroscopy

(Some figures may appear in colour only in the online journal)

1. Introduction

As we approach the era of burning plasmas in fusion devices, advancing predictive capabilities and validating models of particle transport, impurity radiation and neutral particles has become increasingly important. The scaling of fusion power with fuel density squared makes an obvious case for the need to understand density peaking. Effective operation also requires the development of models that may help in controlling impurity accumulation near axis and actively mitigating divertor heat fluxes via edge puffing. Appreciating the
important interactions of neutrals particles, both in terms of fueling and as partners in charge exchange (CX) reactions, is also fundamental for core-edge integration. One common thread through these subjects is the need to simultaneously consider the roles of both transport and atomic physics. In this paper, we present the Aurora package, which aims to address some of these needs of the fusion community by providing a modern toolbox for impurity transport modeling, radiation predictions and spectroscopic analysis.

Aurora is well integrated with Python tools originally developed within the one modeling framework for integrated tasks (OMFIT) [1] and recently released for independent installation. This allows it to leverage a wide community effort to generalize common analysis routines and access data from multiple devices. For example, reading of files in standardized formats, e.g. from EFIT [2] or GACODE routines, is easily accomplished, permitting non-trivial manipulations of experimental data without need for detailed expertise. Integration of Aurora within integrated modeling frameworks is made particularly straightforward by its simple installation procedures, multi-language interfaces and device-agnostic implementation. Aurora is an entirely open source project, hosted on GitHub and welcoming contributions from users with a range of interests. Documentation is available at https://aurora-fusion.readthedocs.io.

In this paper, we aim to describe the current status of the project and provide a clear reference for some key features. Section 2 gives an overview of capabilities related to atomic data and radiation predictions, particularly making use of the atomic data and analysis structure (ADAS) database [3]. In section 3 we outline Aurora’s 1.5D simulation capabilities for ion transport in magnetically confined plasmas. Section 4 explains the concept of superstaging and its use in Aurora as a way to reduce the numerical complexity of heavy ions and to test approximations for edge codes. In section 5 we discuss how processing of SOLPS-ITER results with Aurora indicates that CX with edge neutrals should not significantly affect the total radiated power, $P_{\text{rad}}$, within the last closed flux surface (LCFS) in ITER. The subjects of superstaging and CX offer two examples of research activities where the modern interface and flexibility of Aurora has facilitated new insights, advancing both technical capabilities and physical interpretation for predictions relevant to future devices. In section 6 we conclude by describing opportunities for experimental analysis using the new ImpRad module in OMFIT, facilitating inferences of impurity transport on multiple devices based on Aurora.

2. Atomic data and spectral predictions

Aurora relies on ADAS data for much of its functionality. While it may be convenient to combine usage of ADAS Python routines with Aurora, it is not necessary to run (or even have access to) an ADAS framework. Aurora can work with atomic data from the OPEN-ADAS website (www.open-adas.ac.uk), fetching and interpreting files automatically through the internet. Most of the generalized collisional-radiative (GCR) formats distributed by ADAS can be processed. While users may indicate specific ADAS files that they wish to use, a set of defaults are conveniently listed within Aurora for most ions of interest in fusion. This offers the side benefit of lowering the ‘entry barrier’ for researchers who are not familiar with ADAS nomenclature and who may be unsure about which files offer the highest data quality. Cooling coefficients, both resolved for each ion stage and weighted by fractional abundances of each charge state at ionization equilibrium, can easily be loaded and processed for use in integrated modeling. For applications in spectroscopy, the aurora.read_aif15 function allows parsing and interpolation of photon emissivity coefficients in the ADAS ADF15 format, separating components driven by ionization, excitation or recombination processes, as well as metastable-resolved parts whenever these are available in the chosen files. Aurora also offers initial support to work with the ColRadPy [4] package, which makes use of ADAS resolved ion data collections in the ADF04 format for GCR model predictions. Specifically for the analysis of neutral H-isotope particles, some atomic data from the formulae by Janev and Smith [5] or Janev et al [6] are also available.

All the tools discussed in this section are applicable to Aurora simulations of particle transport (section 3) as well as other codes and analysis frameworks. Indeed, Aurora has been used for experimental spectroscopic signal analysis [7, 8] and to post-process data from neutral codes such as KN1D [9], FIDASIM [10, 11] and EIRENE [12]. Aurora does not include capabilities to work with molecules at the time of writing, but the subject will be explored in future work.

3. 1.5D modeling of particle transport

Simulation of impurity transport is extremely useful for both the interpretation of experimental measurements as well as the assessment of radiative scenarios that reduce heat fluxes going into the boundary layer. A number of 1D/1.5D codes have been developed over the years for these purposes, including STRAHL [13, 14], SANCO [15], MIST [16] and ITC [17]. These codes were not designed to operate on high performance computing clusters, where one may want to keep data in memory rather than writing to disk. Since the task of writing data files can account for the majority of the runtime when dealing with large arrays, avoiding it can give a significant speed improvement, sometimes by a factor in excess of 10. Gaining such advantage was one of the original motivations for the development of Aurora as a tool to rapidly iterate over some input parameters in inferences of experimental impurity transport coefficients on Alcator C-Mod [7] and DIII-D [18]. In this context, Aurora’s simulation capabilities are referred to as a ‘forward model’, differentiating the task of obtaining numerical results from chosen inputs via a pre-defined model and the problem of iterating over forward model inputs until matching some termination conditions (‘inverse problem’).

Aurora’s high-level Python interface provides simple operation and interpretability of results, for example by permitting plotting of atomic rates or the quantification of particle conservation. The high level interface does not come at the cost of reduced speed, since in practice the coupled transport

---

Note

Plasma Phys. Control. Fusion 63 (2021) 112001

---
The radial (cross-field) particle flux $\Gamma_{iZ}$ is separated into diffusive and convective terms with the standard ansatz

$$
\Gamma_{iZ} = -D \frac{\partial n_{iZ}}{\partial r} + v n_{iZ},
$$

(2)

where $D$ is the diffusion coefficient and $v$ is the radial convective velocity. We remark that $D$ and $v$ are inputs to Aurora, not outputs. The coupling between charge states in equation (1) occurs through the atomic sources/sinks term, $Q_{iZ}$, which can be expanded into its components as

$$
Q_{iZ} = -\left(n_e S_{iZ} + n_e \alpha_{iZ} + n_H \alpha_{1Z}^{\mathrm{cs}}\right) n_{iZ}
+ n_e S_{iZ-1} n_{iZ-1}
+ \left(n_e \alpha_{iZ+1} + n_H \alpha_{1Z+1}^{\mathrm{cs}}\right) n_{iZ+1}.
$$

(3)

The right hand side of equation (8) is made of three terms multiplying adjacent charge states. Total ionization rates, obtained from the ADAS SCD data, are labeled as $S$. Radiative and dielectronic recombination rates are grouped into a rate labeled $\alpha$, obtained from ADAS ACD data. Finally, $\alpha^{\mathrm{cs}}$ stands for the contribution from CX of each impurity charge state with the atomic density of hydrogen-isotopes, $n_H$. Electron density is here labeled as $n_e$.

Theoretical transport models often make predictions at the LFS on a midplane radial coordinate, which we label as $r_m$. These results can be transformed to flux surface averaged (FSA) values of $D$ and $v$ for the Aurora forward model using [22]

$$
D = D_{\text{LFS}} \frac{n_{\text{LFS}}}{(n)} \left(\frac{\partial r_{\text{vol}}}{\partial r_m}\right)^2,
$$

(4)

$$
v = \nu_{\text{LFS}} \frac{n_{\text{LFS}}}{(n)} \frac{\partial r_{\text{vol}}}{\partial r_m} + D \frac{\partial}{\partial r_m} \left(\ln \frac{n_{\text{LFS}}}{(n)}\right)
$$

where factors of $n_{\text{LFS}}/(n)$ account for any poloidal density asymmetries. Centrifugal asymmetries can be particularly significant for the transport of heavy ions and may be estimated with Aurora given radial profiles of $n_e$, $T_e$ (electron temperature), $T_i$ (bulk ion temperature), $Z_{\text{eff}}$ (effective ion charge) and $\omega$ (toroidal rotation frequency) [23].

Currently, Aurora uses by default a first-order, vertex-centered, finite-volume scheme recently developed by Linder et al [24] which offers particle conservation and better numerical stability with respect to finite-difference schemes in the presence of large convection, as in pedestal regions of tokamaks. This scheme uses adaptive upwinding for the spatial discretization of the advective terms based on local evaluations at grid point ‘i’ of the parameter

$$
K_i = \max \left(0, 1 - 2/|\mu_i|\right) \cdot \text{sgn} (\mu_i),
$$

(5)

where $\mu_i = |v(r_i)| \Delta r_i / D(r_i)$ is the local Pécel number and $\Delta r_i$ is the local radial grid spacing, which can be flexibly set to be smaller in the pedestal. In the limit of $\mu_i \to 0$ ($K_i = 0$, diffusion-dominated case) a purely central scheme is used, while for $\mu_i \to \infty$ ($K_i = 1$, advection-dominated case) pure upwinding is adopted. It is recommended that users...
apply a highly non-uniform radial grid, maintaining \(|\mu_i| \leq 2\) at all locations and thus avoiding numerical damping. Suggestions on how to do so are provided in the Aurora documentation.

The temporal discretization is performed using the \(\theta\)-method, equally weighting contributions from densities at the previous time step and the new time step (i.e. \(\theta = 1/2\)). Ionization and recombinations rates are set to act on each charge state using the Lackner method [25], which considers ionization and recombinations in two half steps. In the first half step, the ionization term is computed using the density at the new time step (implicitly) and recombinations with the density at the previous time step (explicitly); in the second half step, ionization is computed at the previous time step and recombinations at the new time step. This method is unconditionally stable and allows one to use the fast Thomas algorithm to solve the resulting tri-diagonal system of equations for the charge states. The computational cost of this scheme scales linearly with the number of evolved charge states. Readers are referred to [24, 25] for details on the numerical scheme. Additionally, as proposed by Linder et al [24]. Aurora can evolve the neutral stage in the same way as ionized stages, even though the specific transport coefficients for neutrals is not trivial. Alternatively, users can choose not to evolve neutrals, in which case recombinations for the first ionized stage is artificially set to zero to prevent particle losses. The 1st order finite differences scheme described in the STRAHL manual [14] has also been implemented in Aurora to benchmark its routines against a more established code. Such benchmarks have proven to be perfectly satisfied when using identical modeling choices to those available in STRAHL.

In Aurora, \(D\) and \(v\) can be provided as a function of time, space and charge state (i.e. \(Z\) value). Typically, it is hard to determine what \(Z\) dependence should be assigned to transport coefficients, but in some cases this capability can enable validation efforts, e.g. to test the predictions of neoclassical transport in the pedestal region [7]. It is also possible to provide transport coefficients for each charge state as obtained via a neoclassical and/or turbulent transport code.

In concluding this section, we summarize and highlight key advantages of the Aurora 1.5D forward model with respect to pre-existing codes:

- In iterative frameworks, Aurora can be more than \(10 \times\) faster. While the specific run time depends on user-specified spatio-temporal grids and number of ion stages, generally a simulation over 100 ms can take as little as a few ms of CPU time.
- The current forward model algorithm gives better particle conservation at large \(v/D\) with respect to finite-difference schemes, even with half as many grid points.
- Interfacing with other modeling frameworks is made easier via dedicated modules and OMFIT tools.
- The superstaging method (described in the next section) offers additional speed improvements and a simple test bed for atomic physics in edge transport codes.

4. Superstaging

A significant challenge in investigations of impurity transport is the computational complexity of coupling atomic physics and transport for a large number of charge states. Particularly for edge codes, modeling of complex ions is often considered intractable. For simpler 1.5D modeling in the core, for example with Aurora’s forward model, heavy ions like W make it hard to iterate over some free parameters with sufficient speed. However, with a number of high-Z machines already in existence and more under construction, such modeling is clearly of great interest. The superstaging approximation offers the means to reduce computational complexity with a simple and interpretable method.

The technique described here is analogous to one explored in past JET [15, 26] and ITER [27] modeling. These works rely on the creation of ADAS atomic rates for superstages in a set of dedicated data files. This process has been significantly simplified in Aurora, where transformations of standard collisional-radiative coefficients are computed ‘on the fly’, without need for additional files. The simplicity of these operations calls for an exposition of these methods and a reconsideration of research opportunities associated with superstaging. In this section, we focus on providing an intuitive understanding to make the technique more accessible to modelers. Analogous ideas may be applied to bundle metastable states, but we shall not discuss the subject here for simplicity.

In the superstaging approximation, multiple charge states are ‘bundled’ together to reduce the number of species to evolve in a simulation. Suppose that we wish to group charge states \(i_0\) to \(i_1\) onto a superstage \(\zeta\). We then have

\[
n_{\zeta} = \sum_{i = i_0}^{i_1} n_i
\]

where we have used lower case \(n\) for the density of a general atomic species. Superscripts indicate the stage or superstage of interest, in slight departure from the more common notation used in section 3. In what follows, electron density is simply labeled as \(n_e\). The continuity equation for a charge state \(i\) can be written as

\[
\frac{\partial n_i}{\partial t} = -\nabla \cdot \Gamma_i + Q_i.
\]

and our objective is to obtain an equation of identical form, but with indices \(i\) replaced by \(\zeta\) and each term appropriately adapted to represent superstages. Eventually, after solving for the superstage densities, we wish to split these back into individual charge states densities.

Let us assume that the transport coefficients appearing in the \(\Gamma\) term in equation (7) are independent of \(Z\); we shall come back to this approximation later. Summing particle flux terms of multiple charge states in equation (7) is then trivial, since \(D\) and \(v\) are the same for all charge states and densities linearly sum up as in equation (6). The same is true for the time derivatives of particle density on the left hand side of equation (7).
Let us therefore focus our attention on the sources/sinks term, $Q^i$, which we write here for a charge state $i$ as

$$Q^i = -n_0 S^{i \rightarrow i+1} + n_0 \alpha^{i \rightarrow i-1} + n_0 S^{i \rightarrow i-1} n_i^{i-1} + n_0 \alpha^{i+1 \rightarrow i} n_{i+1}^{i+1}. \quad (8)$$

Here we have adopted a notation that makes it clear which charge states are involved in each reaction, e.g. $S^{i \rightarrow i+1}$ is the process of ionization from charge state $i$ to $i+1$ and $\alpha^{i \rightarrow i-1}$ is recombination from charge state $i$ to $i-1$. Upon summation of the $Q^i$ term for each of the charge states $i_0$ to $i_1$, many terms cancel out. The sum over these charge states can be written as

$$Q^\zeta = +n_0 S^\zeta \rightarrow \zeta+1 n_0^{\zeta-1} - n_0 \alpha^{\zeta \rightarrow \zeta-1} n_0^{\zeta} - n_0 S^\zeta \rightarrow \zeta-1 n_0^{\zeta} + n_0 \alpha^{\zeta+1 \rightarrow \zeta} n_0^{\zeta+1}. \quad (9)$$

The purpose of superstaging is to avoid explicit modeling of individual charge states, which must instead be only related to the correspondent (directly modeled) superstages. In the superstage approximation, we impose quasi-static ionization equilibrium for the charge states $i_0$ to $i_1$ within a superstage. This means that the fractional abundance of each charge state within the superstage is purely set by atomic physics, with no transport involved. Within a superstage, we therefore take

$$n_0^\zeta \big|_{eq} = (\alpha^{i_0+1 \rightarrow i_0} / S^{i_0 \rightarrow i_0+1}) n_0^{i_0+1} \big|_{eq},$$

$$n_0^\zeta \big|_{eq} = (\alpha^{i_0+2 \rightarrow i_0+1} / S^{i_0+1 \rightarrow i_0+2}) n_0^{i_0+2} \big|_{eq},$$

$$\ldots$$

$$n_0^\zeta \big|_{eq} = (\alpha^{i_1-1 \rightarrow i_1} / S^{i_1-1 \rightarrow i_1}) n_0^{i_1} \big|_{eq}. \quad (10)$$

In other words, we wish to evolve $n^\zeta$ in space and time, i.e. $dn^\zeta/dt \neq 0$ in a transport code, and we take $Q^\zeta = 0$ when decomposing each superstage density into its charge state components. This obviously incurs an error, which however is small in many circumstances, as we will show later.

In order to express equation (9) solely in terms of superstage properties, we recognize that both charge states $i_0$ and $i_1$ are absorbed into superstage $\zeta$, so by applying the changes $i_0 \rightarrow \zeta$ and $i_1 \rightarrow \zeta$ one can write

$$S_{i_0-1} \rightarrow i_0 n_{i_0-1} \rightarrow S_{i_1-1} \rightarrow \zeta n_{i_1} \zeta-1,$$

$$\alpha^{i_0 \rightarrow i_0-1} n_0^{i_0} \rightarrow \alpha^{\zeta \rightarrow \zeta-1} n_0^{\zeta},$$

$$S^{i_0+1 \rightarrow i_0+1 n_0^{i_0+1}} \rightarrow S^{i_1-1 \rightarrow \zeta+1} n_0^{\zeta+1},$$

$$\alpha^{i_1+1 \rightarrow i_1+1} n_1^{i_1+1} \rightarrow \alpha^{\zeta+1 \rightarrow \zeta} n_1^{\zeta+1}, \quad (11)$$

which allow us to define effective superstage rates as

$$\alpha^{\zeta \rightarrow \zeta-1} \equiv \alpha^{i_0 \rightarrow i_0-1} \left( \frac{n_0^{i_0}}{n_0^{\zeta}} \right)_{eq}, \quad (12)$$

$$S^{\zeta \rightarrow \zeta+1} \equiv S^{i_0 \rightarrow i_0+1} \left( \frac{n_0^{i_0}}{n_0^{\zeta}} \right)_{eq}. \quad (13)$$

The terms in brackets are the fractional abundances of the ‘edges’ of the superstage $\zeta$, which we explicitly labeled as being computed from ionization equilibrium. Using the superstage rates of equations (12) and (13), one can then express the sources/sinks term for superstages as

$$Q^\zeta = -n_0 S^\zeta \rightarrow \zeta+1 + n_0 \alpha^{\zeta \rightarrow \zeta-1} n_0^{\zeta} + n_0 S^{\zeta \rightarrow \zeta-1} n_0^{\zeta-1} + n_0 \alpha^{\zeta+1 \rightarrow \zeta} n_0^{\zeta+1}, \quad (14)$$

which has the same form as equation (8) and can be used for impurity transport simulations, including (but not exclusively) within the 1.5D Aurora forward model.

$D$ and $v$ can either be set to be the same for all superstages, as commonly done in simulations with individual charge states, or users may indicate an arbitrary Z dependence. This could, for example, be obtained from a neoclassical or turbulent transport code, in which case Aurora can compute an average of transport coefficients weighted by the fractional abundance of charge states within each superstage. This corresponds to taking

$$D^\zeta = \sum_{i_0} D^i \left( \frac{n_i}{n_0^{\zeta}} \right)_{eq}. \quad (15)$$

An analogous weighing by fractional abundances at ionization equilibrium allows one to decompose superstages into individual charges after a simulation:

$$n_i = n_0^{\zeta} \left( \frac{n_i}{n_0^{\zeta}} \right)_{eq}. \quad (16)$$

We remark that while $n_0^{i}$ may be significantly out of ionization equilibrium, resulting from a simulation with finite transport, the right hand side term within brackets in equation (16) is taken purely from ionization equilibrium (no transport).

Once superstage densities have been unstaged, any calculation of radiation terms can proceed as usual using standard collisional-radiative coefficients, with no need for additional tabulation of atomic rates. Superstaging is therefore seen to only require charge state ionization and recombination rates. If CX is considered, the ADAS CCD data are also needed for inclusion within the $\alpha$ rates with appropriate weighting by $n_0/n_0$, where $n_0$ is the background H-isotope atomic neutral density.

The error incurred by the superstaging approximation is dependent on the intensity of plasma transport, since this technique effectively imposes ionization equilibrium within superstage partitions. For typical tokamak transport levels, there often exist multiple useful and safe partitioning strategies for medium- and high-Z impurities. It is however not possible to specify the ‘best’ partition in a universal manner, since the optimal choice of superstages depends on the application of interest. The top panel of figure 2 shows W fractional abundances of all charge states of W as a function of $T_e$, using dashed lines. Easily identifiable charge states have been indicated by numbers near the peak fractional abundance of each. A partition of charge states based on the valence electronic shells
Figure 2. Full ionization equilibria of (top) W and (bottom) Ca over electron temperature, $T_e$, shown by dashed lines. Numbers within the plot identify selected ionization stages. Continuous lines show one bundling scheme for each atomic species, in the case of W based on filled electronic shells, while for Ca only low-Z stages are grouped.

is also shown by continuous lines, corresponding to superstages $Z \in \{0\}, \{1, 2\}, \{3, 8\}, \{9, 18\}, \{19, 32\}, \{33, 50\}, \{51, 72\}, \{72, 74\}$. This partitioning is physically motivated by atomic physics and analogous to the ‘natural partition’ strategy defined in ADAS, whereby charge states are bundled based on variations of ionization potential ($I_z$) between charge states. Specifically, the natural partition is computed for each species by finding large deviations of the quantity $2(I_{z+1} - I_z)/(I_{z+1} + I_z)$ from a running mean over charge states [26]. To this end, Aurora can make use of ionization energies from NIST [28], collected via ColRadPy [4]. Alternatively, users may also use fractional abundances at ionization equilibrium to determine which charge states have a small range of existence in a given simulation and can therefore be safely bundled. Such ‘clustering’ of superstages can be easily dealt with using the K-means algorithm.

However, often the optimal partitioning of charge states may be specific to the application of interest. For example, for edge simulations, charge states with charge greater than W20+ are unlikely to be useful and can always be bundled. Similarly, core simulations may bundle low- and high-lying charge states that are never reached at the temperatures of interest, only simulating charge states that produce observable effects. We note that the 0th charge state is always needed in forward modeling where ion sources are introduced in the neutral state. The bottom panel of figure 2 shows a case where we bundle charge states $\{1, 9\}$ and keep individual charge states with $Z > 9$. This choice is favorable for core impurity transport studies where Ca is injected into tokamak plasmas via laser blow off since emission from charge states 1–9 is typically not measured and only higher charge states must be carefully resolved. In tests of the Aurora forward model with this partitioning strategy, we have found that a negligible error in impurity densities and radiation is incurred for any realistic transport levels. Recall that the computational cost of the Aurora forward model is linearly proportional to the number of modeled charge states, so superstaging as in figure 2 gives approximately a factor of 2 speed improvement.

Figure 3(a) shows two Aurora simulations for Fe injection in the ITER baseline scenario, where peak electron temperatures and densities reach $T_e \approx 25$ keV and $n_e \approx 1.1 \times 10^{20}$ $m^{-3}$. We compare the standard use of all charge states (thin lines) with a superstage partition that bundles Fe charge states with $Z < 16$ (thick lines), analogously to the Ca strategy of figure 2. Almost no difference is observed in the charge state densities of interest; the inset shows a zoomed in view of the peak of Fe$^{24+}$ near the pedestal, where a small inaccuracy is seen. For this comparison, a flat diffusion coefficient with magnitude $D = 10$ $m^2 s^{-1}$ was assumed, setting convection to zero for simplicity. We note that the value of $D = 10$ $m^2 s^{-1}$ is relatively high and, since superstaging becomes less accurate at high transport levels, this can be seen as a successful test in challenging, yet reasonable, conditions for the application of
Figure 3. Comparison of Aurora simulations for Fe transport in (a) the ITER baseline scenario ($T_e \approx 25$ keV and $n_e \approx 1.1 \times 10^{20}$ m$^{-3}$ on axis), and (b) an Alcator C-Mod discharge with $T_e \approx 3.5$ keV and $n_e \approx 1.5 \times 10^{20}$ m$^{-3}$ on axis. Transport coefficients $D = 10$ m$^2$ s$^{-1}$, flat across the radius, and $v = 0$ m s$^{-1}$ were assumed for illustrative purposes. Thin lines show results from a simulation resolving all charge states, thick lines from bundling $Z < 16$ states. We remark however that the same partitioning strategy can be much less successful in other plasma scenarios. Figure 3(b) shows the result of using the same Fe superstaging strategy for realistic Alcator C-Mod kinetic profiles with on-axis values of $T_e \approx 3.5$ keV and $n_e \approx 1.5 \times 10^{20}$ m$^{-3}$, much lower than in the ITER baseline. As in figure 3(a), thin lines correspond to the simulation resolving all charge states, while thicker lines show the result of bundling $Z<16$ states (except the neutral stage, as always). While core profiles are recovered very well, significant discrepancies are found in the outer part of the plasma. Of course, even larger errors would be incurred if a smaller number of superstages were directly modeled, e.g. taking only every other Fe state between $Z = 16$ and $Z = 26$.

Many of the envisioned applications of superstaging are with high-Z ions. Figure 4(a) shows two Aurora simulations of W in the ITER baseline scenario. The displayed charge states were the only ones that were explicitly modeled, i.e. all others were bundled. Except for small inaccuracies near axis, the agreement between thin and thick lines is excellent, particularly considering the large reduction of computational cost of this superstaging partition. Figure 4(b) shows the cumulative radiated power computed by Aurora as a function of plasma radius when setting the total W concentration to be $1.5 \times 10^{-5}$. 

Figure 4. Application of superstaging within an Aurora simulation of W in the ITER baseline scenario. In (a), thin lines show the result of simulating all charge states, while thicker lines show the result of superstaging. Here, only $Z = \{50, 53, 56, 59, 62, 65, 68\}$ were superstaged, using $D = 10$ m$^2$ s$^{-1}$, $v = 0$ m s$^{-1}$. In (b), we show the difference between the two cases for cumulative radiated power, integrated from the magnetic axis outwards, in the pedestal region.
on axis [29]. The radial range is focused on the pedestal region in order to better visualize the loss of accuracy due to supersta-
ging, shown by the difference between the full model, simulat-
ing all charge states (red) and the reduced superstaged model (blue). Clearly, such simulation strategy could be very advant-
geageous for a number of applications, including integrated mod-
eeling where impurity radiation can play a significant role. In
these cases, modeling impurities as having charge state den-
sities set by a constant fraction of the electron density can res-
ult in significant errors, whereas running Aurora with a small
number of superstages, appropriately chosen for the scenario
of interest, can be fast and effective.

5. CX in the ITER pedestal

Aurora’s routines for the post-processing of SOLPS-ITER [30, 31] results enable consideration of core-edge effects, for
example related to the influence of edge neutrals in the ped-
estal region [32]. Recent work by Dux et al [33] demonstrated the
discount of CX on ionization balance in the AUG ped-
estal, making use of detailed charged exchange recombination
(CER) spectroscopy measurements of Ne$^8^+^+$ and Ne$^{10^+}$ and a
1D Monte Carlo neutral code to model edge neutral penetra-
tion. An extrapolation to ITER based on these AUG obser-
vations suggested that CX could significantly affect the total
radiated power emitted within the LCFS, $P_{rad}$, increasing core
Ne radiation by up to a factor of 5. Here, we provide a new
assessment of this subject based on EIRENE modeling of neu-
tral penetration in ITER using the SOLPS-ITER code, which is
a higher-fidelity model with respect to previous descriptions.

The details of these SOLPS-ITER results, including realistic
levels of Ne puffing for divertor heat flux mitigation, have
been presented elsewhere as part of ITER divertor studies [34].
Here, we limit ourselves to a demonstration of post-processing
and interpretation of the SOLPS-ITER data using Aurora.
By providing the path to the SOLPS-ITER results directory,
including an EFIT EQDSK file, Aurora can load data on both
the EIRENE and B2 grids with just a couple of lines of Python
code. An additional call to the aurora.solps.get_radial_prof
function allows extraction of low- and high-field side (HFS) radial
profiles at the midplane, as well as flux surface averages.
Figure 5 shows these atomic D neutral density profiles, nor-
malized by the corresponding $n_s$ radial profiles (LFS, HFS and
FSA). We note that FSA neutral profiles are appropriate inputs
to Aurora’s 1.5D impurity transport model to examine the
effect of neutrals on impurity ionization balance via CX. This
is a physical modeling choice everywhere except very close to
the LCFS (typically, for $\rho_p < 0.99$) since the parallel motion
of impurities on flux surfaces tends to be much faster than their
ionization. As a result, the impact of neutrals on impurity ions
is effectively averaged out also in the presence of strong pol-
ooidal neutral asymmetries, e.g. near divertor x-points or neu-
tral beam injection [33]. Figure 5 shows that $n_n/n_e$ is expected
to be lower than $10^{-5}$, i.e. below the red dashed line. At the
realistic Ne densities of the ITER baseline scenario, Aurora
predicts the contribution to $P_{rad}$ directly attributable to CX
recombination to be only approximately 0.2 MW. The indirect
contribution from the modification of the Ne ionization equi-
lbrium due to CX cannot be self-consistently assessed without
turning CX on and off in the SOLPS-ITER simulation itself,
but at the low values of $n_n/n_e$ this effect is undoubtedly very
small. These conclusions can be interpreted as resulting from
the large width of the ITER SOL, which significantly reduces
neutral penetration into the confined region. We note that while
Alcator C-Mod’s highest density shots did reach similar opa-
city to high-performance ITER projections [35], the C-Mod
SOL was significantly thinner. As a result, C-Mod’s absolute
neutral densities in the pedestal were lower than in any other
current tokamak, but still higher than in ITER, whose core
fueling is bound to have different global character.

6. Discussion and conclusions

In this paper we have presented Aurora, a package for particle
transport, neutrals and radiation modeling. Beyond its original
purpose as a modern impurity transport forward model, Aurora
has found applications as a simple interface to atomic data
from ADAS and other databases. It has also been used to post-
process results from other codes and integrate them within
a framework suited for core-edge research in magnetically-
confined plasmas. Aurora’s forward model inherits from pre-
existing numerical methods, particularly developed at JET
and AUG, with important new capabilities enabled by a
high-level Python interface with Fortran, as well as initial
support for Julia. In section 4, we presented the supersta-
ging approach used in Aurora. This could be adopted well
beyond the Aurora forward model, particularly benefiting
computationally-intensive edge codes, offering a fast interface to test the superstaging assumption in the presence of finite cross-field transport. Section 5 discussed the impact of CX on the total radiated power within the LCFS in ITER, making use of post-processing tools of SOLPS-ITER results within the package. The extremely low values of $n_e/n_x$ found inside the LCFS make CX negligible to the ITER $P_{rad}$.

Many of Aurora’s features are leveraged in the OMFIT ImpRad module, where they enable fast simulations of impurity transport, radiation predictions, development of synthetic diagnostics, and inferences of impurity transport coefficients. A graphical user interface (GUI) allows the creation of detailed Aurora namelists, simple loading of magnetic geometries and kinetic profiles from other OMFIT modules, visualization and post-processing of results, as well as benchmarking with STRAHL for some test cases. Complex models with time-dependent transport coefficients, possibly arising from sawteeth or edge localized modes, can be easily constructed via the GUI. A set of workflows for the inference of impurity transport coefficients from experimental data has also been implemented, permitting nonlinear optimization with LMFIT [36], Markov Chain Monte Carlo with the emcee package [37] and nested sampling with MultiNest [38]. These iterative frameworks make use of a set of synthetic diagnostics developed for a range of measurements, including from extreme ultraviolet (EUV) spectroscopy, soft x-ray arrays, CER, visible Bremstrah lung, and bolometry. Inferences of particle transport can be submitted to remote clusters to access high-performance computing resources, where Aurora simulations can be run in parallel. The ImpRad module has been generalized to minimize device-specific tasks and allow utilization of the framework on multiple devices.

We conclude by highlighting the open-source and collaborative nature of the Aurora project. In this paper, we have described and demonstrated some of its capabilities, showing how it builds on the success of previous modeling tools. This has led to new opportunities in fusion research, many of them naturally related to core-edge integration given the range of applicability of the current code base.

Data availability statement

Code and documentation for the Aurora package are available at https://aurora-fusion.readthedocs.io.

The data that support the findings of this study are openly available at the following URL/DOI: https://aurora-fusion.readthedocs.io.

Acknowledgments

The authors wish to thank all OMFIT contributors for their team work toward developing shared software tools. F S would also like to thank L Lauro-Taroni and A Foster for helpful conversations on superstaging, R Dux for making the STRAHL code and its manual available to us, and D Stańczak for facilitating online distribution of Aurora.

This material is based upon work supported by the Department of Energy under Award Numbers DE-SC0014264, DE-FC02-04ER54698, DE-AC05-00OR22725 and DE-SC0007880. This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training program 2014–2018 and 2019–2020 under Grant Agreement No. 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

ORCID iDs

F Sciortino https://orcid.org/0000-0002-5159-1889
T Odstrcil https://orcid.org/0000-0003-4173-4230
O Meneghini https://orcid.org/0000-0001-5100-5483
O Linder https://orcid.org/0000-0002-5637-2861
J D Lore https://orcid.org/0000-0002-9192-465X
N T Howard https://orcid.org/0000-0002-8787-6309

References

[1] Meneghini O et al 2015 Integrated modeling applications for tokamak experiments with OMFIT Nucl. Fusion 55 083008
[2] Lao L L, John H S, Stambaugh R D, Kellman A G and Pfeiffer W 1985 Reconstruction of current profile parameters and plasma shapes in tokamaks Nucl. Fusion 25 1611–22
[3] Summers H P, Dickson W J, O’Mullane M G, Badnell N R, Whiteford A D, Brooks D H, Lang J, Loch S D and Griffin D C 2006 Ionization state, excited populations and emission of impurities in dynamic finite density plasmas: I. The generalized collisional-radiative model for light elements Plasma Phys. Control. Fusion 48 263–93
[4] Johnson C A, Loch S D and Ennis D A 2019 CoRlRadPy: a python collisional radiative solver Nucl. Mater. Energy 20 100579
[5] Janev R K and Smith J J 1993 Cross sections for collisions of energetic ions with hydrogen and deuterium Nucl. Fusion 4 195 (available at: https://inis.iaea.org/search/search.aspx?orig_q=RN:25024275)
[6] Janev R K, Reiter D and Samm U 2003 Collisional processes of low-temperature hydrogen plasmas Technical Report
[7] Sciortino F et al 2020 Inference of experimental radial impurity transport on Alcator C-Mod: Bayesian parameter estimation and model selection Nucl. Fusion 60 126014
[8] Sciortino F, Cao N M, Howard N T, Marmar E S and Rice J E 2021 Particle transport constraints via Bayesian spectral fitting of multiple atomic lines Rev. Sci. Instrum. 92 53508
[9] LaBombard B 2001 KN1D: A 1-D Space, 2-D Velocity, Kinetic Transport Algorithm for Atomic and Molecular Hydrogen in an Ionizing Plasma (Available at: https://dspace.mit.edu/handle/1721.1/93211)
[10] Geiger B et al 2020 Progress in modelling fast-ion D-alpha spectra and neutral particle analyzer fluxes using FIDASIM Plasma Phys. Control. Fusion 62 105008
[11] Stagner L, Geiger B and Heidbrink W FIDASIM: A Neutral Beam and Fast-ion Diagnostic Modeling Suite (Available at: https://doi.org/10.5281/zenodo.1341369)
[12] Reiter D, Bacliams M and Birnir P 2005 The EIRENE and B2-EIRENE codes Fusion Sci. Technol. 47 172–86
[13] Behringer K H 1987 Description of the impurity transport code STRAHL Technical Report

[14] Dux R 2006 STRAHL user manual Technical Report
(Available at: https://pure.mpg.de/rest/items/item_2143869/component/file_2143868/content)

[15] Lauro-Taroni L et al 1994 Impurity transport of high performance discharges in JET Proc. 21st EPS Conf. Controlled Fusion Plasma Physics (Montpellier, France, 27 June–1 July 1994) vol 18B p 102

[16] Hulse R A 1983 Numerical studies of impurities in fusion plasmas Nucl. Technol./Fusion 3 259–72

[17] Parisot T, Guirlet B, Bourdelle C, Garbet X, Dubuit N, Imbeaux F and Thomas P R 2008 Experimental impurity transport and theoretical interpretation in a Tore Supra lower-hybrid heated plasma Plasma Phys. Control. Fusion 50 055010

[18] Odstrcil T, Howard N T, Sciortino F, Chrystal C, Holland C, Hollmann E, McKee G, Thome K E and Wilks T M 2020 Dependence of the impurity transport on the dominant turbulent regime in ELM-y H-mode discharges on the DIII-D tokamak Phys. Plasmas 27 082503

[19] Peterson P 2009 F2PY: a tool for connecting Fortran and Python programs Int. J. Comput. Sci. Eng. 4 296–305

[20] Bezanson J, Edelman A, Karpinski S and Shah V B 2017 Julia: a fresh approach to numerical computing SIAM Rev. 59 65–98

[21] Geiger B et al 2019 Observation of anomalous impurity transport during low-density experiments in W7-X with laser blow-off injections of iron Nucl. Fusion 59 046009

[22] Angioni C et al 2014 Tungsten transport in JET H-mode plasmas in hybrid scenario, experimental observations and modelling Nucl. Fusion 54 083028

[23] Odstrcil T, Pütterich T, Angioni C, Bilato R, Gude A and Odstrcil M 2018 The physics of W transport illuminated by recent progress in W density diagnostics at ASDEX upgrade Plasma Phys. Control. Fusion 60 014003

[24] Linder O, Fable E, Jenko F, Papp G and Pautasso G 2020 Self-consistent modeling of runaway electron generation in massive gas injection scenarios in ASDEX upgrade Nucl. Fusion 60 096031

[25] Lackner K, Behringer K, Engelhardt W and Wunderlich R 1982 An algorithm for the solution of impurity diffusion under finite reaction rates Z. Naturforsch. A 37 931–8

[26] Foster A 2008 On the behaviour and radiating properties of heavy elements in fusion plasmas PhD Thesis University of Strathclyde

[27] Bonnin X and Coster D 2011 Full-tungsten plasma edge simulations with SOLPS J. Nucl. Mater. 415 S488–91

[28] Kramida A, Ralchenko Y and Reader J (NIST ASD Team) 2020 NIST Atomic Spectra Database (version 5.8) (Gaithersburg, MD: National Institute of Standards and Technology) (Available at: https://physics.nist.gov/asd) (Accessed 26 May 2021)

[29] Pütterich T, Fable E, Dux R, O’Mullane M, Neu R and Sicciono M 2019 Determination of the tolerable impurity concentrations in a fusion reactor using a consistent set of cooling factors Nucl. Fusion 59 056013

[30] Bonnin X, Dekeyser W, Pitts R, Coster D, Voskoboynikov S and Wiesen S 2016 Presentation of the new SOLPS-ITER code package J. Nucl. Mater. 463 480–4

[31] Bonnin X, Dekeyser W, Pitts R, Coster D, Voskoboynikov S and Wiesen S 2016 Presentation of the new SOLPS-ITER code package for tokamak plasma edge modelling Plasma Fusion Res. 11 1403102

[32] Sciortino F, Howard N, Marmar E, Reksoatmodjo R, Odstrcil T, Rice J, Foster A and Lore J Experimental inference and modeling of neutral and impurity transport in L-, H- and I-mode Alcator C-Mod plasmas (in preparation)

[33] Dux R, Cavedon M, Kallenbach A, McDermott R M and Vogel G (the Asdex Upgrade Team) 2020 Influence of CX-reactions on the radiation in the pedestal region at ASDEX upgrade Nucl. Fusion 60 126039

[34] Pitts R A et al 2019 Physics basis for the first ITER tungsten divertor Nucl. Mater. Energy 20 100696

[35] Mordijck S et al 2020 Collisionality driven turbulent particle transport changes in DIII-D H-mode plasmas Nucl. Fusion 60 066019

[36] Newville M, Stensitzki T, Allen D B and Ingargiola A 2014 LMFIT: Non-Linear Least-Square Minimization and Curve-Fitting for Python (0.8.0) Zenodo (Available at: https://doi.org/10.5281/zenodo.11813

[37] Foreman-Mackey D, Hogg D W, Lang D and Goodman J 2013 emcee: the MCMC hammer Publ. Astron. Soc. Pac. 125 306–12

[38] Feroz F and Hobson M P 2008 Multimodal nested sampling: an efficient and robust alternative to Markov chain Monte Carlo methods for astronomical data analyses Mon. Not. R. Astron. Soc. 384 449–63