Modeling turbulent transport with a velocity dependent diffusion operator in 2D gyrokinetics

Jaime Caballero[1], Kento Ketelaars[2,1], Manuare Francisquez[3], Sjoerd Reindhoudt[2,1],
Stefan Dasbach[1], Sven Wiesen[1], M.J. Pueschel[1]

[1]Dutch Institute for Fundamental Energy Research (DIFFER),
[2]Eindhoven University of Technology (TU/e),
[3]Princeton Plasma Physics Laboratory (PPPL)

TSVV-C annual meeting / IPP Garching, Germany / 1 - 3 June 2026

Motivation

Accurate and efficient modeling of turbulent transport at the edge/SOL of future reactors (e.g. ITER or SPARC) is crucial for the power exhaust design.

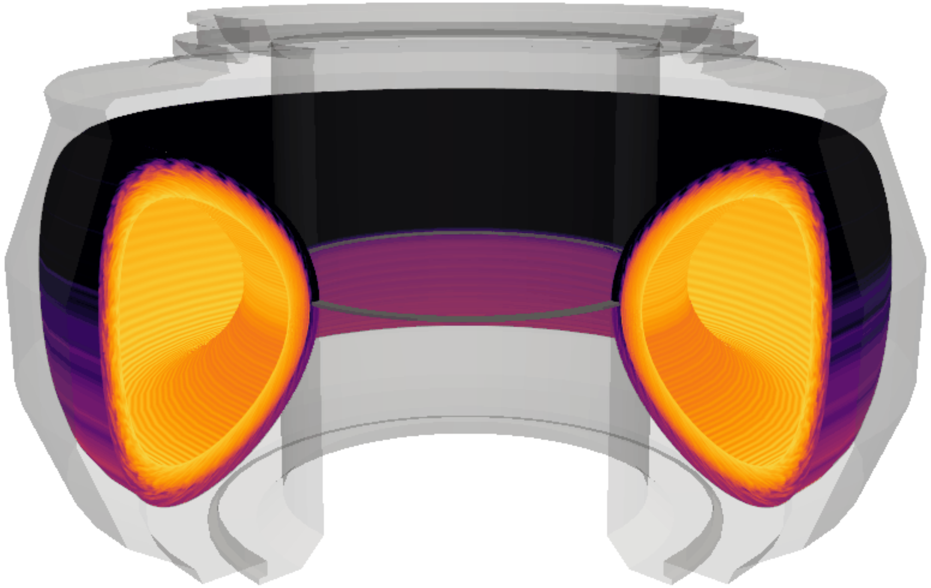
Outline

- Gkeyll
- 2D gyrokinetics vs. fluid codes
- Modeling turbulent transport
- Kinetic radiation operator
- Outlook

Gkeyll
full-f gyrokinetic code

Gkeyll

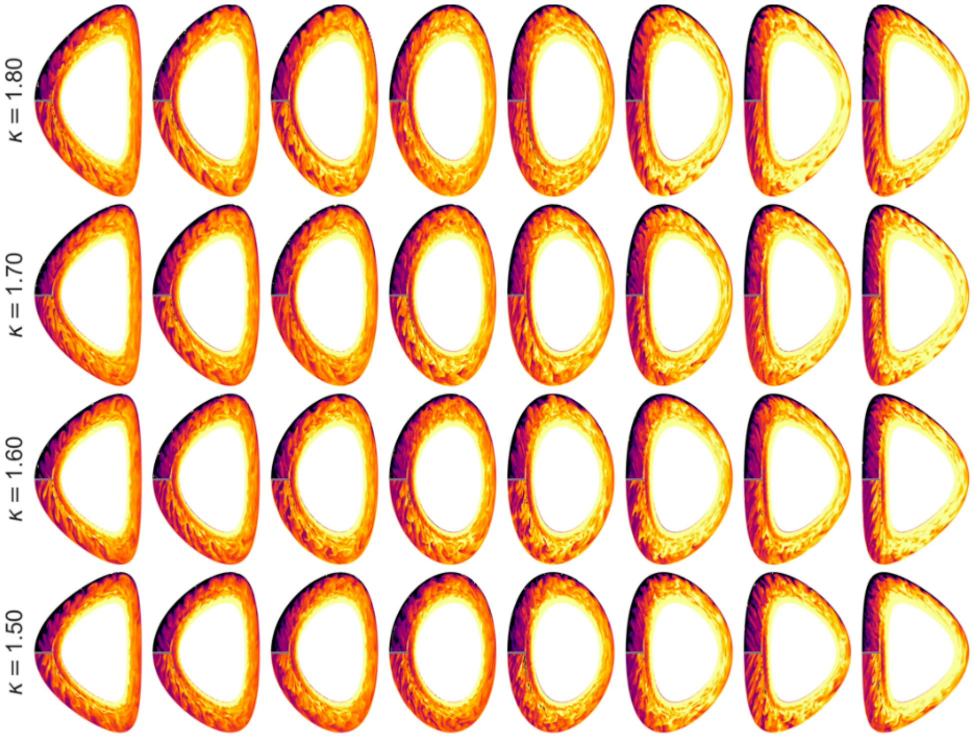
- Eulerian (grid-based) full-f electromagnetic gyrokinetic code
- Discretised with alias-free, quadrature-free discontinuous Galerkin (DG)
- Runs on CPUs or GPUs
- Open source: https://github.com/ammarhakim/gkeyll
- Robust even at low-resolution



Credit: A. Hoffmann (https://github.com/ammarhakim/gkylzero)

Gkeyll

High-throughput full-f gyrokinetics of the tokamak boundary, A. Hoffmann et al. 2026 https://arxiv.org/abs/2605.01117



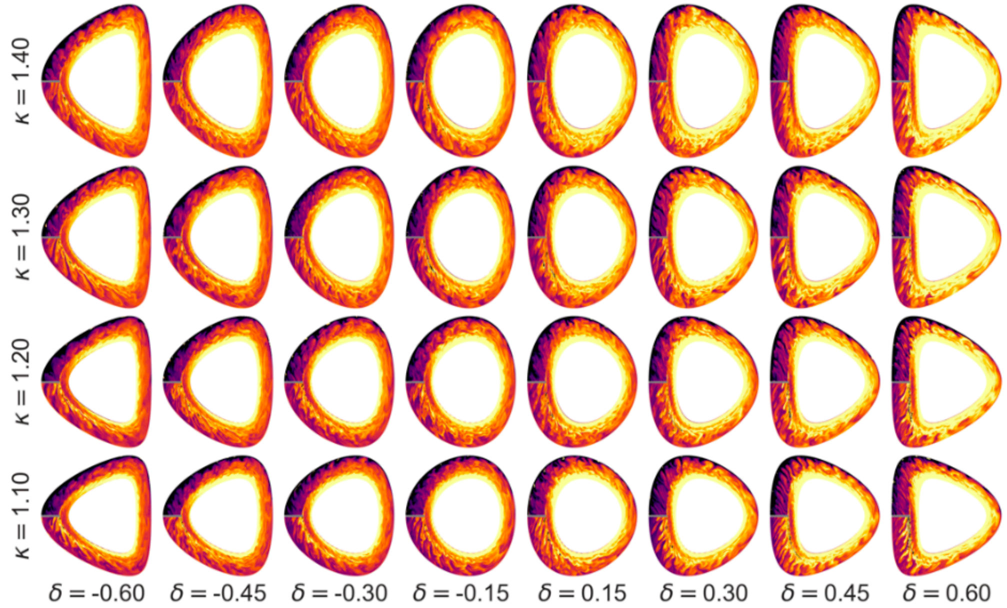
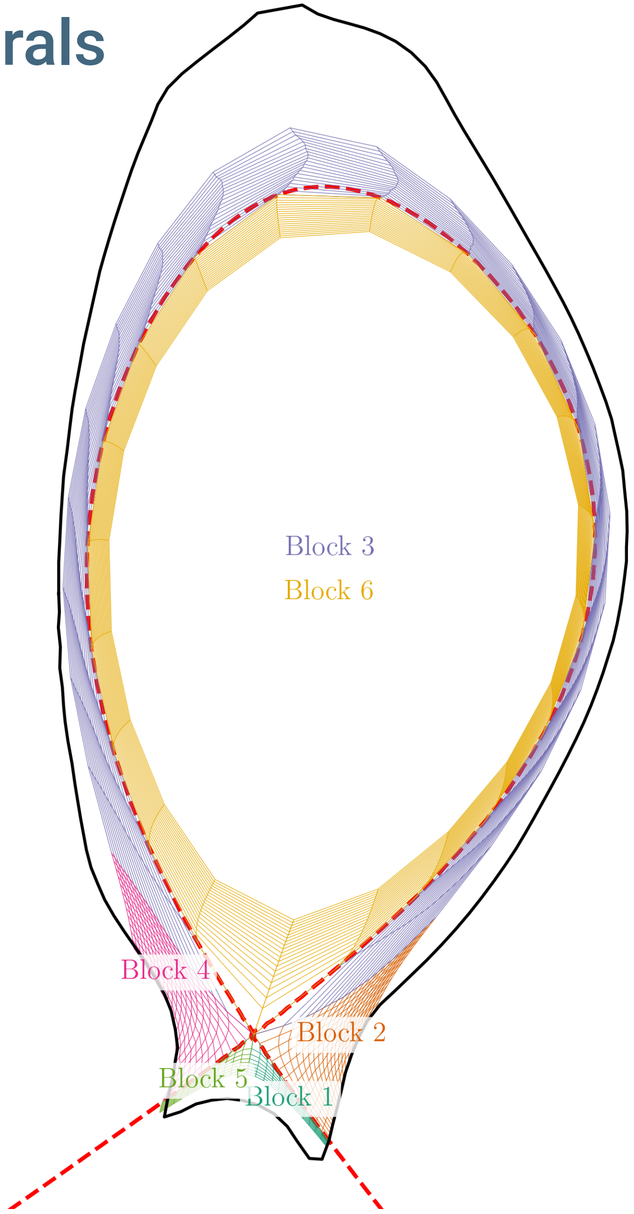


FIG. 3. Poloidal cut of the ion temperature at $t = 1$ ms for the 64 geometry considered with $P_{in} = 0.5$ MW. The same color mapping is used for each plot.

Gkeyll: hierarchical fidelity for dimensionality & neutrals

- Plasma neutral interactions
    - Ionisation, recombination, charge exchange, radiation, recycling
- Choice of model for evolving neutrals:
    - Gkeyll fluid model
    - Gkeyll Eulerian neutrals (T. Bernard, et al. PoP 29 2022)
    - EIRENE (adds more reactions and accurate wall shape)
- Can run in 1D2V, 2D2V or 3D2V, e.g.:
    - 2D2V limited or diverted
    - 3D2V limited (soon diverted, TCV-X21 benchmark happening now, D.S. Oliveira et al. NF 62, 2022)



2D Gyrokinetics vs fluid codes

2D2V supplements current 2D approaches with kinetics

- Like SOLPS-EIRENE but with kinetic effects (& modern algorithms/software)
- In the electrostatic long-wavelength limit (though a Pade FLR model is available in Gkeyll):

$$\frac{\partial f_s}{\partial t} + \dot{\mathbf{R}} \cdot \nabla f_s + \dot{v}_\parallel \frac{\partial f_s}{\partial v_\parallel} - \nabla \cdot (\mathbf{D} \cdot \nabla f_s) = C[f_s] + S_c + C_s^{iz} + C_s^{rec} + C_s^{rad}$$

$$-\nabla_\perp \cdot \left(\sum_s \frac{m_s n_{0s}}{B_0^2} \nabla_\perp \phi \right) = \sum_s q_s n_s(\mathbf{R})$$

- Handles drifts in 2D2V (at minimal extra cost)
- Overall much cheaper than 3D2V: Allows us to either add more physics or perform scans

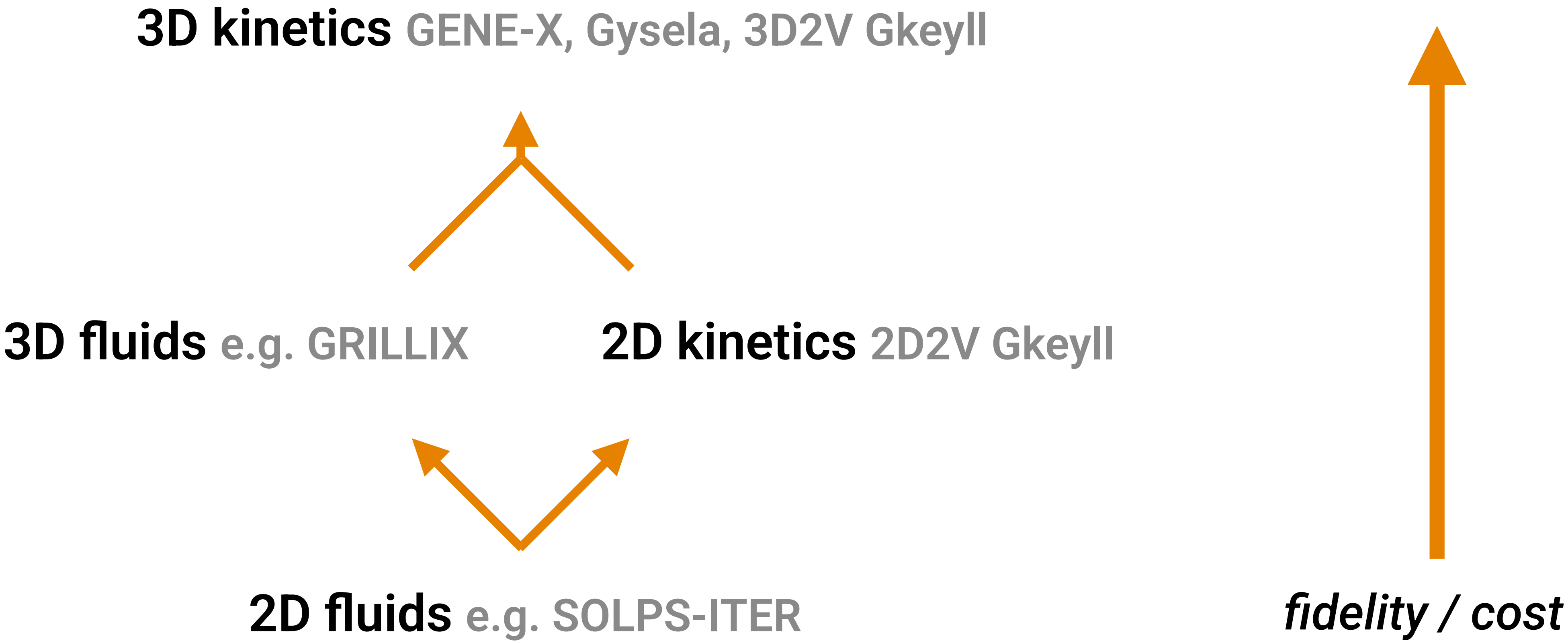
Why 2D gyrokinetics instead of fluids?

- Has non-Maxwellian effects

  e.g. in low recycling SOL collisionality << 1, even in high-recycling T_{ped} ~ keV so kinetics are important

- Does not assume $T_Z = T_D$ important for impurity transport in the SOL (A. Shukla, et al. AIP 15 2025)
- Includes the mirror force important near X-point and in long-leg divertor
- Easier to add drifts in modern GK codes
- Can leverage body of knowledge & tools from decades of 2D fluids (e.g. SOLPS-NN)
- GPUs and advanced algorithms make cost accessible

Why 2D gyrokinetics instead of fluids?



3D kinetics GENE-X, Gysela, 3D2V Gkeyll

3D fluids e.g. GRILLIX

2D kinetics 2D2V Gkeyll

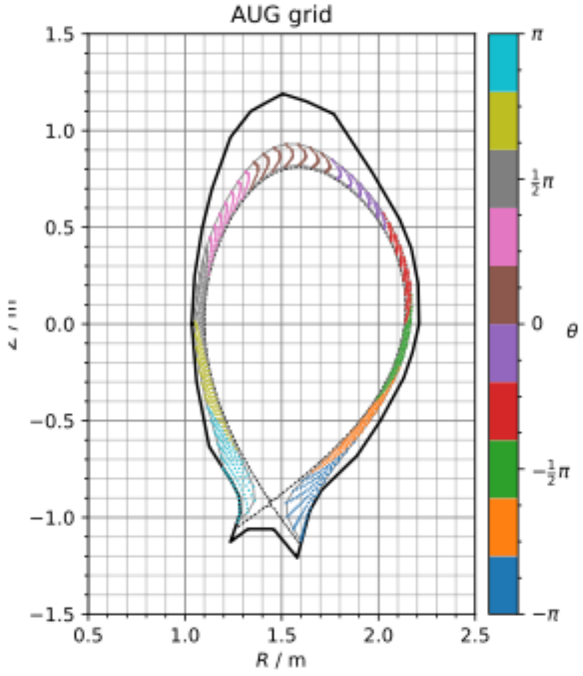
2D fluids e.g. SOLPS-ITER

fidelity / cost

2D2V: example

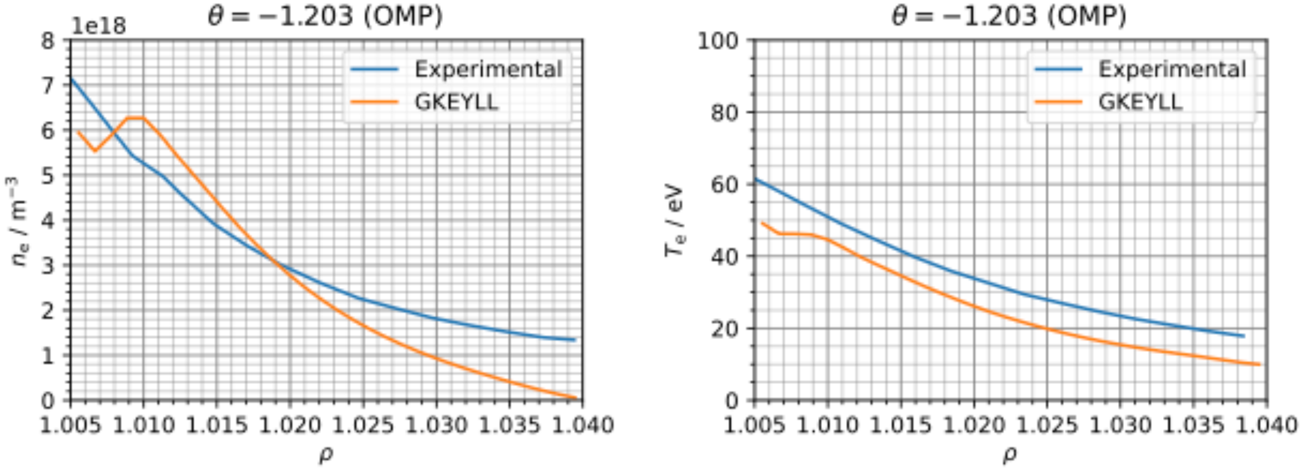
- “The impact of kinetic shielding on divertor impurity transport in ASDEX Upgrade and ITER”

S. Reindhout, M.Sc. Thesis TU/e 2025



#36190

(a) GKEYLL grid for AUG

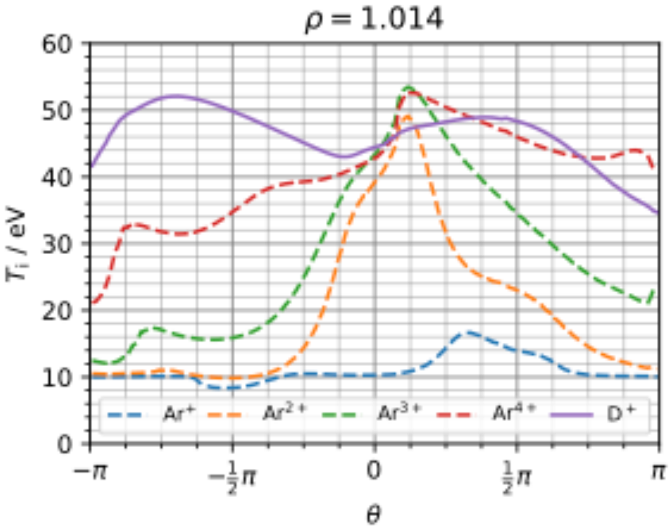


(a) Electron density

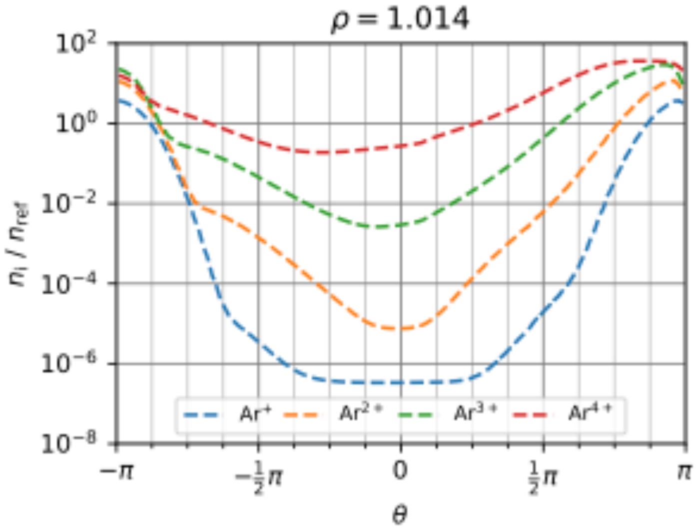
(b) Electron temperature

2D2V: example

- Static neutral background with just ionisation



(a) Ion temperatures

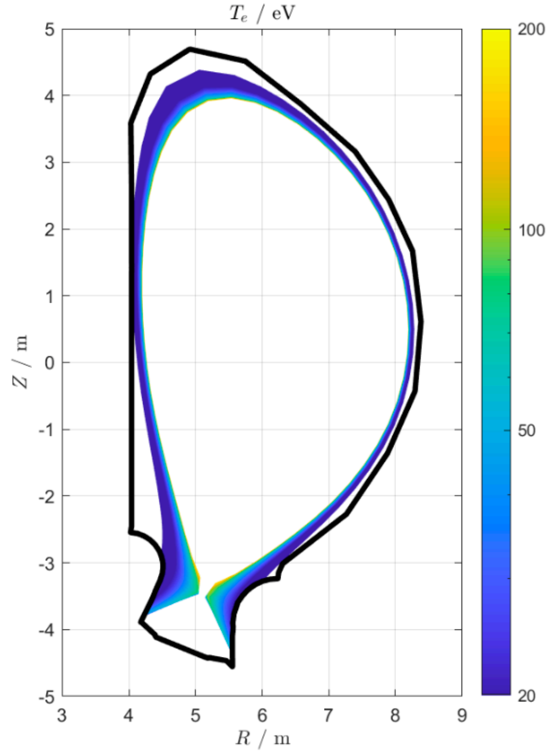


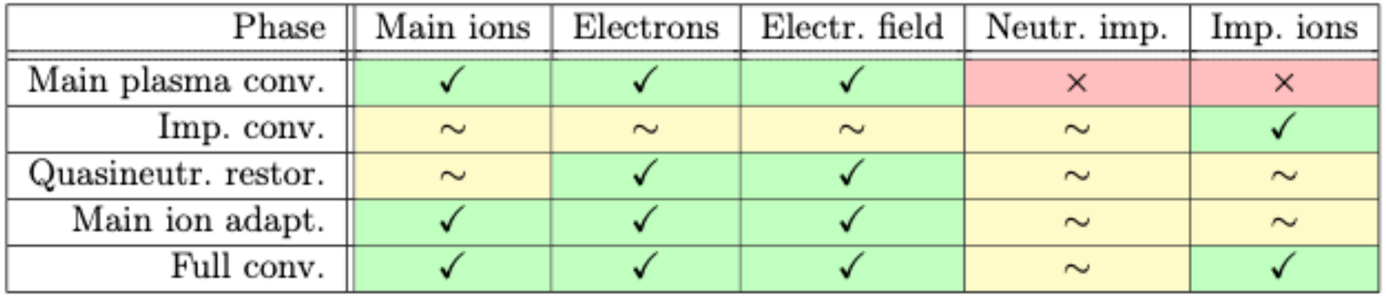
(b) Impurity density

S. Reindhout M.Sc. thesis TU/e 2025

2D2V: example

- ITER : from 250 days ~ 24,000 GPU-hours to 5 hours ~ 20 GPU-hrs using:





Phase	Main ions	Electrons	Electr. field	Neutr. imp.	Imp. ions
Main plasma conv.	✓	✓	✓	×	×
Imp. conv.	∼	∼	∼	∼	✓
Quasineutr. restor.	∼	✓	✓	∼	∼
Main ion adapt.	✓	✓	✓	∼	∼
Full conv.	✓	✓	✓	∼	✓

Table 4.1: The physics enabled in each phase of the simulation strategy. Legend: ✓ - Evolved dynamically, ∼ - Included but not evolved dynamically, × - Excluded from simulation.

S. Reindhout M.Sc. thesis TU/e 2025

Modeling turbulent transport in 2D gyrokinetics

Diffusion operator

- Major goal is to developed a turbulent transport model closer to first principles than the customary ad-hoc transport coefficients (still reduced model)
- Currently in Gkeyll 2D2V user sets D, in kinetics a Fick's law model forces $\chi = 1.5D$
- Current aim is to develop $D_{\text{kin}}(v)$, a velocity dependant operator $\frac{\partial f_s}{\partial t} = \nabla_\perp \cdot (D_{\text{kin}}(v) \nabla_\perp f_s)$
- The 0th and 2th moment of this operator are the particle Γ and heat flux q
- Idea is to match $\int D_{\text{kin}}(v, \alpha, \beta, ...) \nabla_\perp f_s d^3v = \Gamma(D, \nabla n, ...) = D\nabla n + Vn$

Gkeyll
Kinetic radiation operator

Kinetic radiation operator for gyrokinetics

- “A kinetic line-driven radiation operator and its application to Gyrokinetics”
- Used the OpenADAS PLT model $S_i = -n_e n_i L_i(n_e, T_e)$ to match the operator

$$\frac{\partial f}{\partial t} + \nabla \cdot (\mathbf{v}_e f_e) + \nabla_v \cdot (\mathbf{a}_e f_e) = C_{\text{rad}}[f_e] = \nabla_v \cdot (v\nu(v) f_e(v))$$

- Assuming $\int_{-\infty}^{\infty} \frac{1}{2} m_e v^2 \nabla_v \cdot (v\nu(v) f_{Me}(v)) dv = -n_e n_i L_i(n_e, T_e)$ and $\nu(v) = A \frac{\alpha + \beta}{\beta x^{-\alpha} + \alpha x^{\beta}} v^{\gamma}$

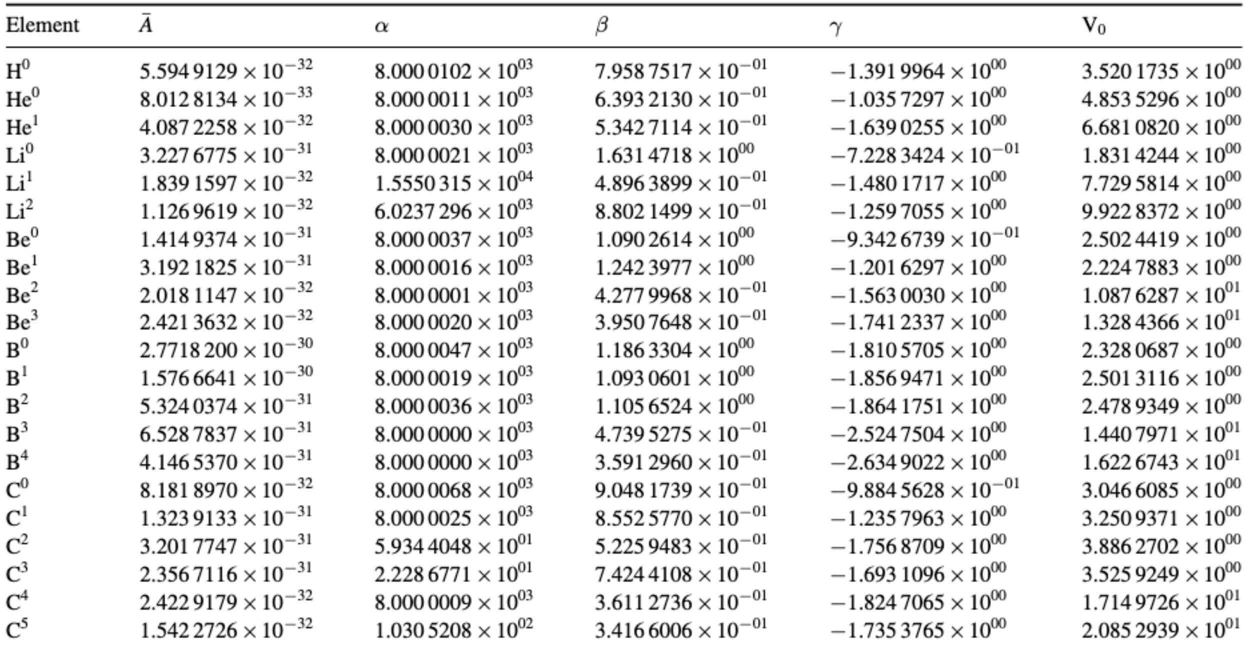
$$x = v/V_0$$

- $F(T_{e,j}, \bar{A}, \alpha, \beta, \gamma, V_0) = \sum_j \left(\left[\frac{1}{C} \int_0^{\infty} \frac{\bar{v}^4}{T_{e,j}^{3/2}} \bar{A} \frac{\alpha + \beta}{\beta x^{-\alpha} + \alpha x^{\beta}} \bar{v}^{\gamma} e^{-\bar{v}^2/T_{e,j}} d\bar{v} - L_i(T_{e,j}) \right] L_i(T_e, j)^{w-1} \right)^2$

Jonathan Roeltgen et al 2025 NF 65 106020

Kinetic radiation operator for gyrokinetics

- For every element, charge state and some densities

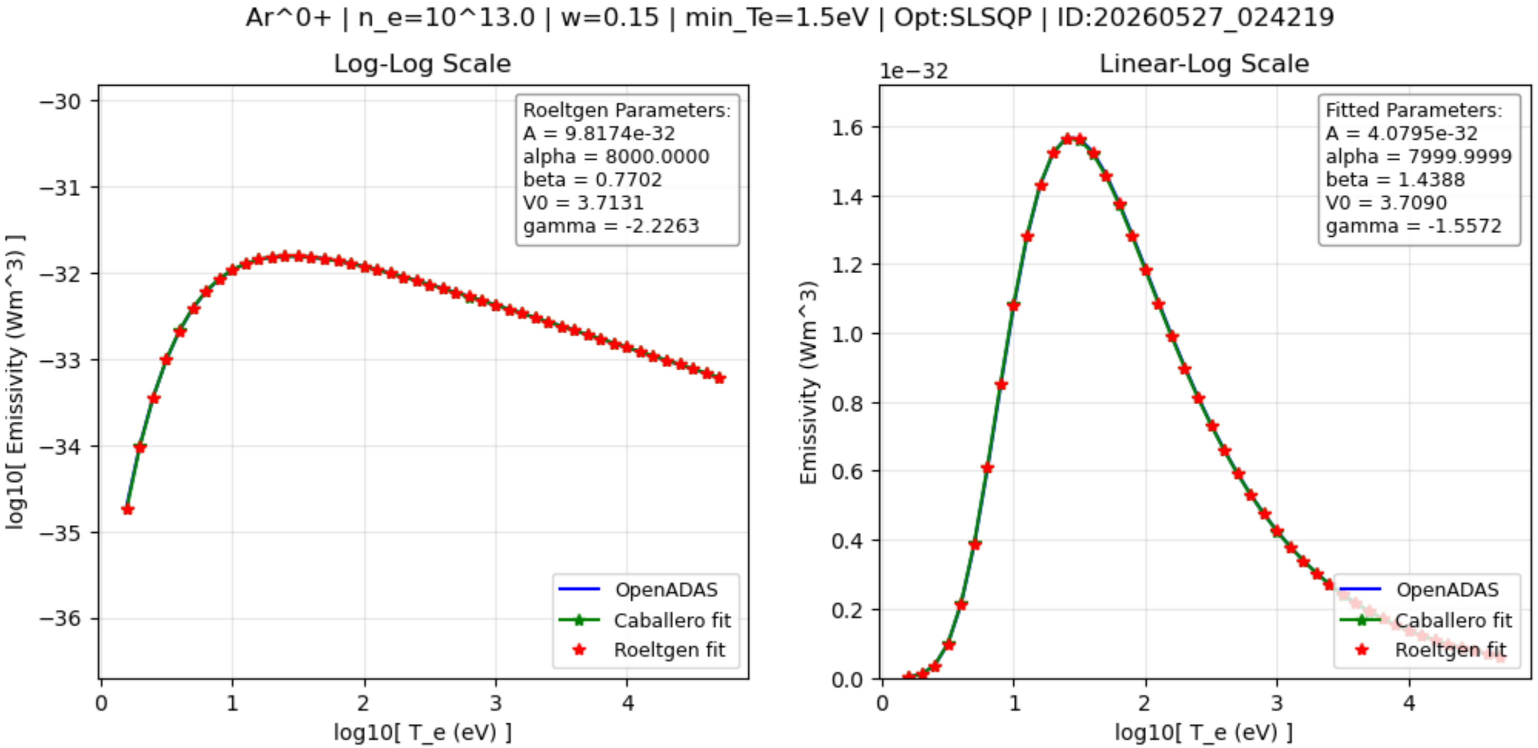


Element	$\bar{A}$	α	β	γ	V_0
H^0	$5.594\,9129 \times 10^{-32}$	$8.000\,0102 \times 10^{03}$	$7.958\,7517 \times 10^{-01}$	$-1.391\,9964 \times 10^{00}$	$3.520\,1735 \times 10^{00}$
He^0	$8.012\,8134 \times 10^{-33}$	$8.000\,0011 \times 10^{03}$	$6.393\,2130 \times 10^{-01}$	$-1.035\,7297 \times 10^{00}$	$4.853\,5296 \times 10^{00}$
He^1	$4.087\,2258 \times 10^{-32}$	$8.000\,0030 \times 10^{03}$	$5.342\,7114 \times 10^{-01}$	$-1.639\,0255 \times 10^{00}$	$6.681\,0820 \times 10^{00}$
Li^0	$3.227\,6775 \times 10^{-31}$	$8.000\,0021 \times 10^{03}$	$1.631\,4718 \times 10^{00}$	$-7.228\,3424 \times 10^{-01}$	$1.831\,4244 \times 10^{00}$
Li^1	$1.839\,1597 \times 10^{-32}$	$1.5550\,315 \times 10^{04}$	$4.896\,3899 \times 10^{-01}$	$-1.480\,1717 \times 10^{00}$	$7.729\,5814 \times 10^{00}$
Li^2	$1.126\,9619 \times 10^{-32}$	$6.0237\,296 \times 10^{03}$	$8.802\,1499 \times 10^{-01}$	$-1.259\,7055 \times 10^{00}$	$9.922\,8372 \times 10^{00}$
Be^0	$1.414\,9374 \times 10^{-31}$	$8.000\,0037 \times 10^{03}$	$1.090\,2614 \times 10^{00}$	$-9.342\,6739 \times 10^{-01}$	$2.502\,4419 \times 10^{00}$
Be^1	$3.192\,1825 \times 10^{-31}$	$8.000\,0016 \times 10^{03}$	$1.242\,3977 \times 10^{00}$	$-1.201\,6297 \times 10^{00}$	$2.224\,7883 \times 10^{00}$
Be^2	$2.018\,1147 \times 10^{-32}$	$8.000\,0001 \times 10^{03}$	$4.277\,9968 \times 10^{-01}$	$-1.563\,0030 \times 10^{00}$	$1.087\,6287 \times 10^{01}$
Be^3	$2.421\,3632 \times 10^{-32}$	$8.000\,0020 \times 10^{03}$	$3.950\,7648 \times 10^{-01}$	$-1.741\,2337 \times 10^{00}$	$1.328\,4366 \times 10^{01}$
B^0	$2.7718\,200 \times 10^{-30}$	$8.000\,0047 \times 10^{03}$	$1.186\,3304 \times 10^{00}$	$-1.810\,5705 \times 10^{00}$	$2.328\,0687 \times 10^{00}$
B^1	$1.576\,6641 \times 10^{-30}$	$8.000\,0019 \times 10^{03}$	$1.093\,0601 \times 10^{00}$	$-1.856\,9471 \times 10^{00}$	$2.501\,3116 \times 10^{00}$
B^2	$5.324\,0374 \times 10^{-31}$	$8.000\,0036 \times 10^{03}$	$1.105\,6524 \times 10^{00}$	$-1.864\,1751 \times 10^{00}$	$2.478\,9349 \times 10^{00}$
B^3	$6.528\,7837 \times 10^{-31}$	$8.000\,0000 \times 10^{03}$	$4.739\,5275 \times 10^{-01}$	$-2.524\,7504 \times 10^{00}$	$1.440\,7971 \times 10^{01}$
B^4	$4.146\,5370 \times 10^{-31}$	$8.000\,0000 \times 10^{03}$	$3.591\,2960 \times 10^{-01}$	$-2.634\,9022 \times 10^{00}$	$1.622\,6743 \times 10^{01}$
C^0	$8.181\,8970 \times 10^{-32}$	$8.000\,0068 \times 10^{03}$	$9.048\,1739 \times 10^{-01}$	$-9.884\,5628 \times 10^{-01}$	$3.046\,6085 \times 10^{00}$
C^1	$1.323\,9133 \times 10^{-31}$	$8.000\,0025 \times 10^{03}$	$8.552\,5770 \times 10^{-01}$	$-1.235\,7963 \times 10^{00}$	$3.250\,9371 \times 10^{00}$
C^2	$3.201\,7747 \times 10^{-31}$	$5.934\,4048 \times 10^{01}$	$5.225\,9483 \times 10^{-01}$	$-1.756\,8709 \times 10^{00}$	$3.886\,2702 \times 10^{00}$
C^3	$2.356\,7116 \times 10^{-31}$	$2.228\,6771 \times 10^{01}$	$7.424\,4108 \times 10^{-01}$	$-1.693\,1096 \times 10^{00}$	$3.525\,9249 \times 10^{00}$
C^4	$2.422\,9179 \times 10^{-32}$	$8.000\,0009 \times 10^{03}$	$3.611\,2736 \times 10^{-01}$	$-1.824\,7065 \times 10^{00}$	$1.714\,9726 \times 10^{01}$
C^5	$1.542\,2726 \times 10^{-32}$	$1.030\,5208 \times 10^{02}$	$3.416\,6006 \times 10^{-01}$	$-1.735\,3765 \times 10^{00}$	$2.085\,2939 \times 10^{01}$

Jonathan Roeltgen et al 2025 NF 65 106020

Kinetic radiation operator for gyrokinetics

- Python repository: https://github.com/jaimix4/kin-rad-gkeyll



Jonathan Roeltgen et al 2025 NF 65 106020

Back to modeling turbulent transport in 2D gyrokinetics

Diffusion operator methodology

- Currently developing methodology with Kento Ketelaars (BEP project)
- We assume pure plasma and ambipolarity $\Gamma_e = \Gamma_i$
- Matching to $$\Gamma_{e,i} = \int D_{kin,e}(v)\nabla_\perp f_e d^3v = \int D_{kin,i}(v)\nabla_\perp f_i d^3v \equiv -D\nabla_\perp n$$

$$q_e = \int \left(\frac{1}{2}m_e v^2\right) D_{kin,e}(v)\nabla_\perp f_e d^3v - \frac{3}{2}T_e\Gamma_e \equiv -n_e\chi_e\nabla_\perp T_e$$

$$q_i = \int \left(\frac{1}{2}m_i v^2\right) D_{kin,i}(v)\nabla_\perp f_i d^3v - \frac{3}{2}T_i\Gamma_i \equiv -n_i\chi_i\nabla_\perp T_i$$

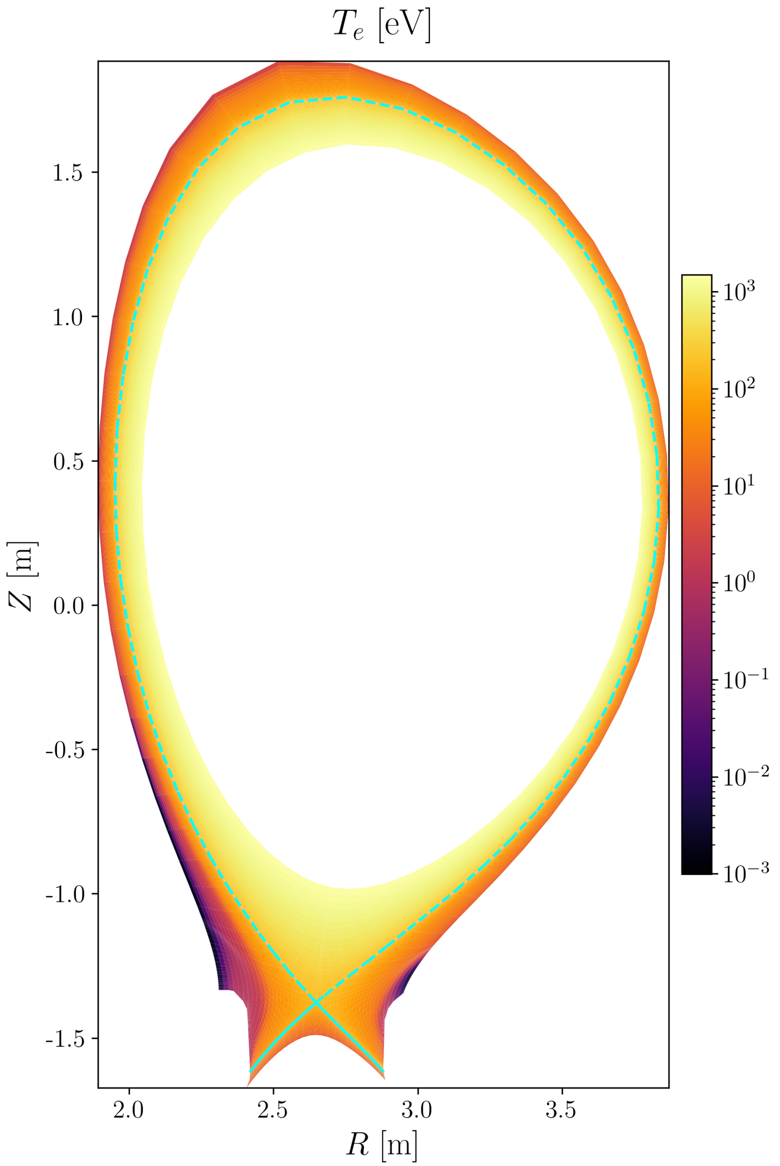
Matching to SOLPS-NN

- Neural network of SOLPS-ITER
  https://github.com/sdasbach/solps-nn

Inputs

Parameter	Range
R	$1 - 10\,\mathrm{m}$
B	$1 - 10\,\mathrm{T}$
P	$10 - 200\,\mathrm{MW}$
D_{puff}	$10^{18} - 10^{24}$ atoms / s
N_{puff}	$10^{18} - 10^{23}$ atoms / s
D_{core}	$10^{19} - 10^{24}$ atoms / s
$D_{\perp}$	$0.1 - 2\mathrm{m}^2$ / s
$\chi_{\perp}$	$0.1 - 2\mathrm{m}^2$ / s

outputs



$$\Gamma = -D_{\perp}\nabla n$$

$$q = -n\chi_{\perp}\nabla T$$

Dasbach, S., & Wiesen, S. (2023). Nuclear Materials and Energy, 34, 101396. https://doi.org/10.1016/

Wiesen, S., et al. (2024). Nuclear Fusion, 64(8), 086046. https://doi.org/10.1088/1741-4326/ad5a1d

Diffusion operator example

- $\Gamma_e = \Gamma_i = -D\nabla_\perp n_{\text{SOLPS-NN}}$ assuming $q_e \sim q_i$ and $D_{kin}(v) = D_0 \exp\left(-\alpha \frac{mv^2}{2T}\right)$
- Bell shape for $D_{\text{kin}}(v)$ to mimic bulk thermal particles contribute the most to turbulent transport

$$\Gamma = -\int D_0 \exp\left(-\alpha \frac{mv^2}{2T}\right) f_M \left[\frac{\nabla_\perp n}{n} + \left(\frac{mv^2}{2T} - \frac{3}{2}\right)\frac{\nabla_\perp T}{T}\right] d^3v$$

- $\Gamma = -D_{11}\nabla_\perp n - D_{12}\nabla_\perp T$ $\quad q = -D_{21}\nabla n - D_{22}\nabla T$
- $D_{11} = D_0 \frac{1}{(1+\alpha)^{3/2}}$ $\quad D_{12} = -\frac{3nD_0}{2T}\frac{\alpha}{(1+\alpha)^{5/2}}$ $\quad D_{22} = \frac{3(2+3\alpha^2)D_0}{4(1+\alpha)^{7/2}}$ $\quad D_{21} = -\frac{3\alpha D_0 T}{2(1+\alpha)^{5/2}}$

Short terms goals/ideas

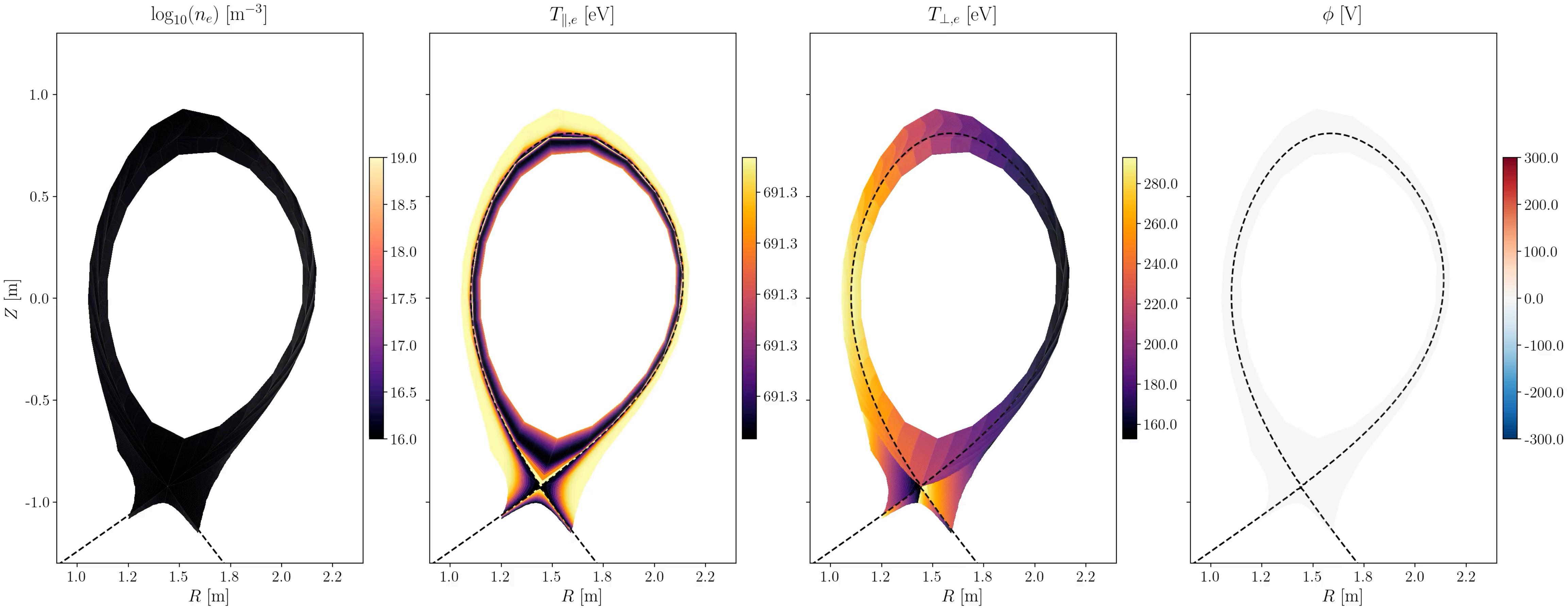
- Derive operator in field-align coordinates, discretised in DG and implement in AUG simulation
- Use $\{D, \chi\}$ profiles fitted experimentally with SOLPS-ITER and 3D simulations to fit the first iteration of $D_{\mathrm{kin}}(v)$ models
- Analyse how ratios of $\{D, \chi\}$ affects decay lengths in 2D2V-EIRENE simulations of AUG
- Possibly craft $D_{\mathrm{kin}}(v)$ to incorporate relevant regimes operational space of (e.g. SepOS) and/or theory-based transport models of micro/macro - instabilities
- Prepare for modeling of H-mode AUG shot with RT-01

Outlook

Outlook - potential long term plans

- Simulate TCV-X2X with 3D2V Gkeyll and test $D_{\mathrm{kin}}(v)$ 2D2V with Gkeyll-EIRENE
- Use 3D2V and experiments to calibrate 2D2V models, e.g. see exactly how the distribution function evolves from first principle simulations
- MSc project to look into detached simulations using separation of different scale dynamics approach from S. Reindhout M.Sc. Thesis 2025
- Work on surrogates of 2D2V Gkeyll-EIRENE (on going project from DIFFER at ENR)
- Long term goal is to apply models to future devices such as SPARC and ITER

Gkeyll - code for edge/SOL plasmas



Kinetic radiation operator for gyrokinetics

- Show it works on kinetic simulation and show good properties that it works
- Show figures that it can also model non-Maxwelliam distributions when compare to CR model
- https://github.com/jaimix4/kin-rad-gkeyll

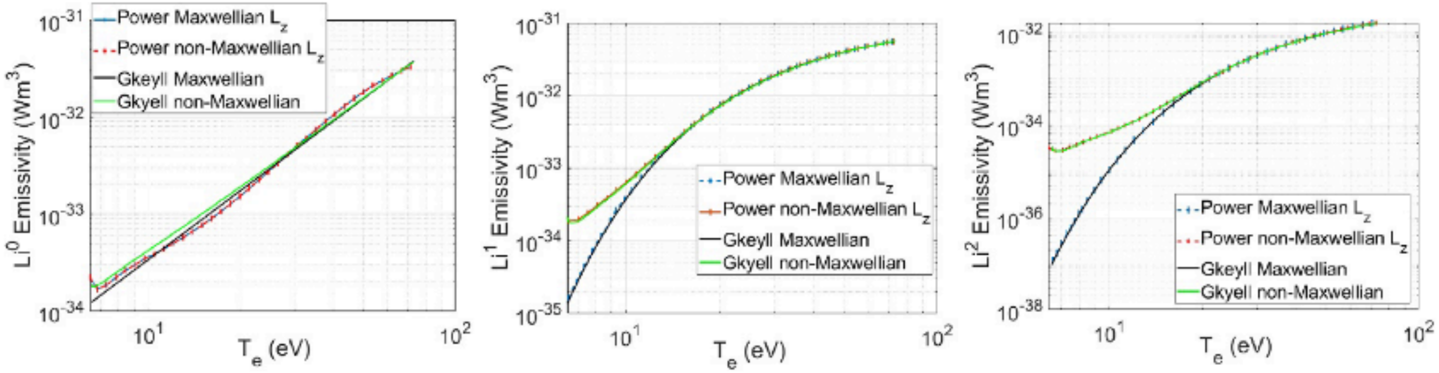


Figure 9. The emissivities of all Li charge states (Li^0 on the left, Li^{+1} in the middle, and Li^{+2} on the right) calculated using Gkeyll and the SIKE collisional radiative code. The Gkeyll emissivities here use the SIKE Maxwellian emissivity to calculate the fit parameters. The non-Maxwellian emissivities use those fitting parameters and the same distribution functions as were used in SIKE.

Jonathan Roeltgen etal 2025 Nucl.Fusion 65 106020