Bringing global gyrokinetic turbulence simulations to the transport timescale using a multiscale approach

Jeffrey B. Parker,[1, *] Lynda L. LoDestro,[1] Daniel Told,[2] Gabriele Merlo,[3] Lee F. Ricketson,[1] Alejandro Campos,[1] Frank Jenko,[2, 3] and Jeffrey A. F. Hittinger[1]

[1]*Lawrence Livermore National Laboratory, Livermore, CA 94550, USA*
[2]*Max-Planck-Institut für Plasmaphysik, Boltzmannstraße 2, 85748 Garching, Germany*
[3]*University of California, Los Angeles, California 90095, USA*

The vast separation dividing the characteristic times of energy confinement and turbulence in the core of toroidal plasmas makes first-principles prediction on long timescales extremely challenging. Here we report the demonstration of a multiple-timescale method that enables coupling global gyrokinetic simulations with a transport solver to calculate the evolution of the self-consistent temperature profile. This method, which exhibits resiliency to the intrinsic fluctuations arising in turbulence simulations, holds potential for integrating nonlocal gyrokinetic turbulence simulations into predictive, whole-device models.

I. INTRODUCTION

First-principles simulations of tokamaks will play an increasingly important role in the mission to understand and control magnetized fusion plasmas. Validated whole-device models aid in interpreting experimental results and help guide the planning of experiments to come. Predictive modeling is particularly important for reactor-scale burning-plasma machines, with their enormous cost, enormous stored energy, and constrained number of shots. While reduced modeling will be used for mapping out parameter space and initial shot planning in these devices, high-fidelity simulations will be required before finalizing shots, to ensure as well as possible that each shot will evolve as planned, and in particular that dangerous operating limits will not be approached.

One area of importance is high-fidelity prediction of heat and particle transport, which tend to be dominated by turbulent motions. In this area, the workhorse framework is the gyrokinetic theory [1, 2]. Gyrokinetic simulations of turbulence for tokamak core plasma have reached a state of maturity [3–12], enabling a recent increase in emphasis on validating simulations to experiment [13–20]. These validation efforts have had some success; for example, when experimentally measured quantities such as the temperature and density profiles are provided as simulation inputs, numerical predictions of heat transport are often in quantitative agreement with experimentally inferred levels. These results suggest that turbulence simulations are nearing a point where quantitative predictive capability is becoming plausible.

For such predictive value to fully materialize, gyrokinetic simulations must be incorporated into modeling on long experimental timescales. Recently, some full-f gyrokinetic simulations have been performed for an energy confinement time [21–25]. Because simulating to the energy confinement timescale is computationally expensive, however, gyrokinetic simulations often last for shorter durations and do not take into account the self-consistent evolution of plasma parameters occurring on the longer timescale. Due to the difference in timescales between turbulence and transport, a direct approach to self-consistent simulation from turbulence to energy-confinement timescales is exceedingly challenging. An estimate of the required computational resources, assuming gyro-Bohm scaling of the turbulent flux [26], the use of magnetic-field-aligned coordinates, and a fixed radial resolution relative to the gyroradius, yields a computational-cost scaling as $\sim \rho_*^{-3}$, where ρ_* is the ratio of the ion thermal gyroradius to the minor radius a of the tokamak. This cost renders such direct simulations of large reactors infeasible at the present time. To illustrate, proceeding from transport-timescale gyrokinetic simulations of relatively small tokamaks ($\rho_* \approx 1/100$) to ITER ($\rho_* \approx 1/1000$) [27] increases the computational cost of an already expensive scheme by three orders of magnitude.

A multiple-timescale approach could provide significantly more advantageous scaling to larger tokamaks such as ITER, by leapfrogging the timescale gap and reducing the computational-cost scaling from ρ_*^{-3} to ρ_*^{-1}. The ρ_*^{-2} difference in cost scaling results from simulating turbulence phenomena and transport phenomena only on their natural timescales and avoiding the simulation of all intermediate timescales. A multiple-timecale scale approach has previously been demonstrated with local, flux-tube gyrokinetic simulations [28, 29]. However, local simulations miss significant nonlocal effects such as turbulence spreading, transport avalanches, and transport in the core of high-performance discharges or internal transport barriers [10–12, 30]. A fully predictive capability requires global simulations. Some early work focused on global gyrokinetic simulations with profile adjustments for matching experimental power balance [31, 32]. In this letter, we focus on global gyrokinetic simulations, presenting a unique method to bridge the timescale separation.

* parker68@llnl.gov

II. THEORETICAL FRAMEWORK

Beginning from the plasma Fokker–Planck equation and assuming a separation of timescales, one can derive coupled equations describing the fast microturbulence and the slowly evolving background Maxwellian. Sugama and Horton first carried out this procedure, and more recently, Abel *et al.* have elucidated it in greater detail [33–35]. We provide a brief overview here.

The formal orderings assume $\tau_{\mathrm{turb}}/\tau \sim \epsilon^2$, $1/\Omega\tau_{\mathrm{turb}} \sim \epsilon$, $\rho/L \sim \rho_* \sim \epsilon$, and $\delta f/f_0 \sim \epsilon$, where τ_{turb} and τ are the characteristic timescales of turbulence and transport, respectively, Ω is the ion cyclotron frequency, ρ is the ion gyroradius, L is a characteristic scale length of the background density, temperature, or magnetic field, and f_0 and δf are the background and fluctuating distribution functions. The smallness of ϵ is well established in the core of a tokamak. In the pedestal and edge region, where background profiles can vary on scales comparable to the gyroradius and fluctuations may be large, these orderings may break down. An asymptotic expansion in ϵ yields the gyrokinetic equation and evolution equations for the density, angular momentum, and pressure of each species. The latter determine the density, rotation velocity, and temperature of the background Maxwellian f_0, which is in local thermal equilibrium on each flux surface.

On the microturbulence timescale, f_0 remains approximately constant. The microturbulence and turbulent fluxes are assumed to be determined by the macroscopic profiles. While the turbulence rapidly fluctuates, appropriate spatial and temporal averages over these quantities are assumed to evolve only on the slow timescale.

A. Multiple-timescale numerical method

Although gyrokinetic codes for simulating turbulence on the fast timescale are well established, solving the set of coupled gyrokinetic and transport equations is still an open problem. Even though the gyrokinetic equation is five-dimensional (5D) and the transport equations are 1D, solving the transport equations self-consistently is quite challenging. The difficulty is that the turbulent flux is strongly nonlinearly dependent on the profiles. One is forced to confront this nonlinearity when using implicit time-stepping, which is highly desirable to enable long time steps.

We use the LoDestro method [36, 37] for coupling global gyrokinetic simulations to a transport equation. We describe the method, which comprises several elements, as applied to the ion pressure equation:

$$\frac{3}{2}V'\frac{\partial p}{\partial t} + \frac{\partial}{\partial x}V'\hat{q} = V'S, \qquad (1)$$

where p is the ion pressure, x is a radial flux coordinate, $V' = dV/dx$ is the differential volume of a flux surface, $\hat{q} = \langle \mathbf{Q} \cdot \nabla x \rangle$ is the averaged radial projection of the heat flux, $\langle \cdot \rangle$ denotes a flux surface and intermediate time average, $\mathbf{Q} = \int d\mathbf{v}\,\frac{1}{2}mv^2\mathbf{v}f$ is the heat flux, and S represents other local terms and sources. For simplicity in this work, some of the terms in the full equation, such as turbulent heating and evolution of the magnetic equilibrium, have been neglected. The heat flux includes both turbulent and neoclassical physics.

In the LoDestro method, the first element is to represent the turbulent component of $\hat{q}$, denoted $\hat{q}^t[p]$ to emphasize it is a functional of the pressure profile p, as the sum of diffusive and convective contributions. Introducing a subscript m to represent the time index and a subscript l to represent the index of iteration within a timestep, we write

$$\hat{q}^t_{m,l} \rightarrow -D_{m,l-1}(\partial_x p_{m,l}) + c_{m,l-1}p_{m,l}. \qquad (2)$$

This key step of lagging the effective transport coefficients D, c (evaluating them at the previous iterate $l-1$) provides a practical scheme to iterate and converge to the nonlinear solution. There is some freedom in determining how to split the flux between D and c. For instance, we write

$$D_{m,l-1} = -\theta_{l-1}\hat{q}^t[p_{m,l-1}]\big/\partial_x p_{m,l-1}, \qquad (3a)$$

$$c_{m,l-1} = (1 - \theta_{l-1})\hat{q}^t[p_{m,l-1}]\big/p_{m,l-1}, \qquad (3b)$$

where one may choose the flux-split parameter θ in a variety of ways. Convective contributions can be important if the flux is nondiffusive. Here, we choose θ to be 1 or nearly 1, unless fluxes run up a gradient, in which case we choose θ locally to be zero. The iteration scheme becomes

$$\frac{3}{2}V'\frac{p_{m,l} - p_{m-1}}{\Delta t} + \partial_x\big[V'\big(-D_{m,l-1}\partial_x p_{m,l} \\ + c_{m,l-1}p_{m,l}\big)\big] = V'S_m, \qquad (4)$$

where Δt is the time step, and for notational convenience only the turbulent component of the flux has been written. The scheme is fully time-dependent, although the first demonstration presented here seeks a steady-state. Equation (4) is linear in the new iterate $p_{m,l}$ and hence straightforward to solve. Despite the simplicity in representing the turbulent flux, if the iteration in l converges, then a correct answer is obtained. This can be seen by substituting Eq. (3) into Eq. (2), in which case a converged answer means the representation for the flux on the right-hand side is equal to the actual turbulent flux on the left-hand side.

The second element modifies the basic idea in Eq. (3) in order to stabilize the iteration. It was recognized in Ref. [37] that a diffusion coefficient's depending strongly on the gradient of the profile causes the iteration of Eqs. (3–4) to be numerically unstable. The iteration can be stabilized by a relaxation of the diffusion coefficient, or similarly, of the turbulent flux. Equation (3) is replaced by

$$D_{m,l-1} = -\theta_{l-1}\overline{\hat{q}^t}_{m,l-1}\big/\partial_x \overline{p}_{m,l-1}, \qquad (5a)$$

$$c_{m,l-1} = (1 - \theta_{l-1})\overline{\hat{q}^t}_{m,l-1}\big/\overline{p}_{m,l-1}, \qquad (5b)$$

where

$$\overline{\hat{q}^t}_{m,l-1} = \alpha \hat{q}^t[\overline{p}_{m,l-1}] + (1-\alpha)\overline{\hat{q}^t}_{m,l-2}, \qquad (6)$$

$$\overline{p}_{m,l-1} = \alpha p_{m,l-1} + (1-\alpha)\overline{p}_{m,l-2}. \qquad (7)$$

In addition to stabilizing the iteration, a secondary purpose of the relaxation parameter α is to provide an effective averaging over previous iterates, reducing the statistical variance from the turbulence simulation.

The third element of the method attempts to minimize the amount of computational time spent in the turbulence simulation, which dominates the computational cost. In our procedure, we run the turbulence simulation for a few eddy times and calculate an average flux $\hat{q}^t_{l-1}$ over that duration (suppressing the time-index subscript m here for simplicity). The relaxed flux $\overline{\hat{q}^t}_{l-1}$ is computed via Eq. (6), which is then used to find a new profile iterate p_l using Eqs. (4–5) and a new $\overline{p}_l$ using Eq. (7). Then, the process repeats for the next iteration. The turbulence simulation runs with the new background profile $\overline{p}_l$ (background Maxwellian f_0), while using the turbulent state δf from the end of the previous iteration as the initial condition.

This procedure efficiently takes advantage of the δf representation in gyrokinetic simulations and avoids the need to start an entirely new simulation for each iteration. The time-saving action of restarting a simulation with a new f_0 and the old δf can result in transient bursts associated with the fast relaxation to a self-consistent δf, but if the variation in f_0 per iteration is small enough, the bursts have minor impact. Computational efficiency is also promoted by averaging the flux over just a few turbulence eddy times per iteration. This strategy may result in the turbulent flux lagging a couple iterations behind the profile iterates, but should not cause significant delay in convergence.

We now compare the LoDestro method with a Newton-based method, which was used by earlier works with *local* gyrokinetic simulations [28, 29]. Those works developed an iteration scheme by expanding the flux as $\hat{q}_l = \hat{q}[p_l] \approx \hat{q}[p_{l-1}] + \delta \hat{q}/\delta p|_{p_{l-1}} \cdot (p_l - p_{l-1})$. Such a Newton-based approach, when applied to *global* simulations, becomes much more challenging because the flux depends in principle on the profile over the entire radial domain, rather than just its local value and gradient as in the local limit. The Jacobian $\delta \hat{q}/\delta p$ is therefore significantly higher dimensional; the number of nonzero elements is proportional to N_r^2 in the global problem rather than to N_r as in the local problem, where N_r is the number of radial grid points. Moreover, each Jacobian-vector product is an expensive prospect because it requires an additional full turbulence simulation to implement a finite-difference estimate of the derivative. In addition to being costly, computation of a derivative by finite difference is challenging as it amplifies error in $\hat{q}$ arising from turbulent fluctuations.

The LoDestro method, which employs a variant of fixed-point, or Picard, iteration, avoids these pitfalls.

Not needing to project differences offers a large gain in efficiency, as the lower sensitivity to error allows the use of as little as a couple eddy times of gyrokinetic simulation per iteration. This gain compensates for the primary disadvantage of fixed-point iteration, that an unaccelerated scheme theoretically achieves linear rather than superlinear asymptotic convergence.

B. Simulation Setup

A newly developed 1D transport solver, `Tango`, implements the LoDestro method. `Tango`, a Python code, is open source [38]. Here, `Tango` is coupled to the global gyrokinetic code GENE [4, 39, 40] to simulate ion-temperature-gradient-driven turbulence.

The GENE simulations are collisionless, electrostatic, and use deuterium ions and adiabatic electrons. Circular magnetic geometry is used [41], with safety factor $q = 0.868 + 2.2\rho^2$, where $\rho = r/a$ and the radius r plays the role of the flux coordinate. The toroidal rotation velocity is set to zero. The density profile is prescribed and fixed in time. The electron temperature is held equal to the ion temperature. GENE is run in the "gradient-driven" mode, in which a Krook-type source prevents the average profile from varying too much within an iteration as the gyrokinetic equation is integrated in time [39].

`Tango` solves the ion pressure equation in the domain $0 \le \rho \le 0.9$, and GENE solves the gyrokinetic equation in the domain $0.1 \le \rho \le 0.9$. `Tango` spatially discretizes with second-order finite differences and uses Neumann boundary conditions at the magnetic axis and Dirichlet conditions (fixed temperature) at the outer radius $\rho = 0.9$.

Two simulations are described here, with key parameters given in Table I. In each simulation, `Tango` seeks a steady-state ($\Delta t = \infty$) solution to Eq. (1). The tokamak in Simulation 1 is roughly half the size ($\rho_* \approx 1/149$) as that in Simulation 2 ($\rho_* \approx 1/292$). In Simulation 1, $120 \times 16 \times 16 \times 78 \times 48$ grid points are used for the radial, binormal, coordinate along the magnetic field, parallel velocity, and magnetic moment dimensions, respectively, in GENE, and 27 radial grid points are used in `Tango`. In Simulation 2, $240 \times 16 \times 16 \times 78 \times 48$ grid points are used in GENE and 53 radial grid points are used in `Tango`. In lieu of a full neoclassical model, a small uniform diffusivity is included in the heat flux. Further details on the simulations can be found in the Supplemental Material.

The setup for `Tango`'s simulation is as follows. In each iteration, GENE runs for 50 R_0/v_{T_i}, where R_0 is the major radius and v_{T_i} is the ion thermal velocity. The simulation is run for 50 iterations. The relaxation parameter is $\alpha = 0.3$. Prior to commencing iterations coupled with `Tango`, GENE is run standalone for 600 R_0/v_{T_i} to seed the initial condition.

flux.

Figure 3 shows results of simulations, using the parameters of Simulation 1, but starting from three different initial temperature profiles. The steady-state temperature profile, normalized temperature gradient, and heat flow obtained in the three cases are nearly identical.

As a test of sensitivity to numerical parameters, we varied the flux-split parameter θ and redid Simulation 1 using constant $\theta = 0.5$ or $\theta = 0.8$. After iterating to convergence, we obtained the same steady-state solution as before.

TABLE I. Key parameters for simulations.

Parameter	Simulation 1	Simulation 2
Minor radius a	0.594 m	1 m
Major radius R_0	1.65 m	3 m
Input power $P_{\rm in}$	3 MW	20 MW
Added diffusivity	0.05 m^2/s	0.25 m^2/s
Outer T boundary condition	0.44 keV	1 keV
ρ_*	1/149	1/292

III. RESULTS

The results from Simulation 1 are shown in Figure 1, which depicts the applied heat source, the temperature profile, the normalized temperature gradient, and the total heat flow (turbulent + added diffusivity) $H = V'\hat{q}$. The initial condition (dashed curve) and a few early iterates (grey curves) are plotted for both temperature and heat flow. After roughly 10–20 iterations, a noise floor is reached, where fluctuations in $\hat{q}^t$ prevent any further convergence. More iterations do not substantially help because the relaxation in Eqs. (6–7) has a finite memory.

Focusing on iterates 31–50, we take the mean as the solution (red curve) and the standard deviation (light blue shaded region) as a rough measure of uncertainty. The maximum coefficient of variation across radius is 7.1% for the temperature. For the heat flow, the maximum is 11.3% for $\rho \geq 0.28$; greater relative fluctuations occur at smaller ρ because the heat flow is near zero. There is a small bump in temperature at $\rho \approx 0.22$, at which the heat flux flows up the temperature gradient. The bump is likely due to the strong localization of the heat source in this region. The heat flux is still outward, an effect probably due to turbulence spreading from the adjacent region.

In steady state, the heat flow H can be determined exactly by integrating Eq. (1) over radius; $H(r)$ must equal $\int_0^r d\bar{r}\, V'(\bar{r})S(\bar{r})$. The average of the numerically simulated heat flow over the final 20 iterations agrees well with this exact result (black dots). The heat flows in both Simulation 1 and 2 are insensitive to the value of the ad-hoc diffusivity, as the heat flux resulting from this diffusivity is miniscule compared to the turbulent flux.

Figure 2 presents similar results for Simulation 2. The multiscale coupling has also found the solution here, confirmed by the consistency of the numerical heat flow with the integral of the source, as shown in Figure 2(d). The maximum coefficient of variation across radius is 5.7% for the temperature and 8% for the heat flow ($\rho \geq 0.35$). It is interesting to note that the self-consistent heat flow is quite different from the initial heat flow, even though the final temperature profile does not appear that different from the initial temperature profile. This behavior reflects the well-known nature of stiff transport, where small profile variations can lead to large changes in the
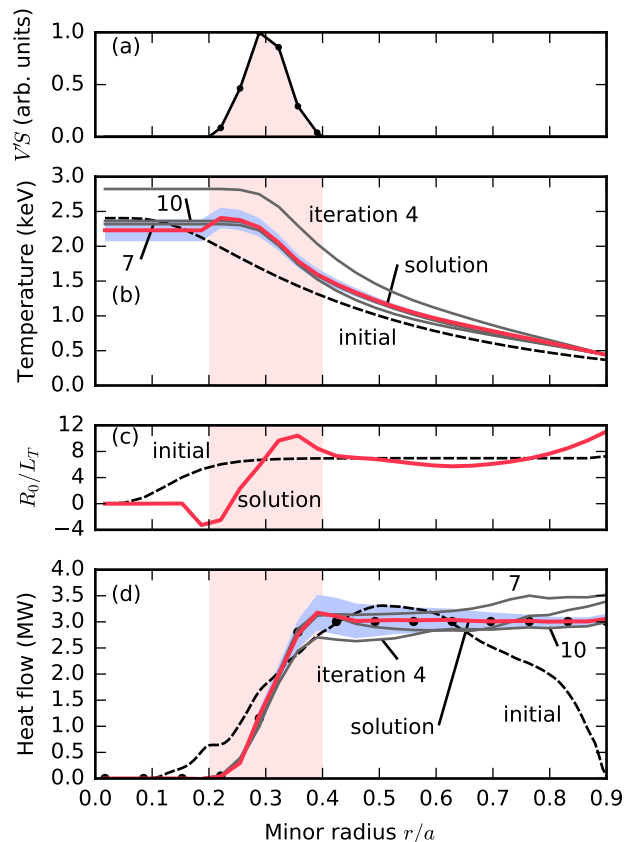

FIG. 1. Results from Simulation 1. (a) The 3 MW input heat source $V'S$. The red shaded region indicates the area where the heat source is applied. (b) Temperature profile. (c) Normalized temperature gradient $R_0/L_T = -(R_0/T)dT/dr$. (d) Heat flow $H = V'\hat{q}$. Shown in (b), (c), and (d) are the initial iterate (dashed curve), the mean over the final 20 out of 50 iterations (red curve labeled "solution"). Also shown in (b) and (d) are a few early iterations (numbered curves), and the standard deviation over the final 20 iterations (light blue shaded region). In (d), the black dots depict the integral of the source, with which the numerical solution agrees well. The initial iterate has a higher resolution than the other curves because it is taken directly from a GENE simulation rather than the Tango grid. This heat flow taken from GENE falls to zero at the outer edge because of the buffer zone in GENE. The larger R_0/L_T that develops toward the edge $r/a \approx 0.9$ in Simulations 1 and 2 is an artifact of the buffer zone and does not have a direct effect (see the Supplemental Material for more details about the buffer zone).
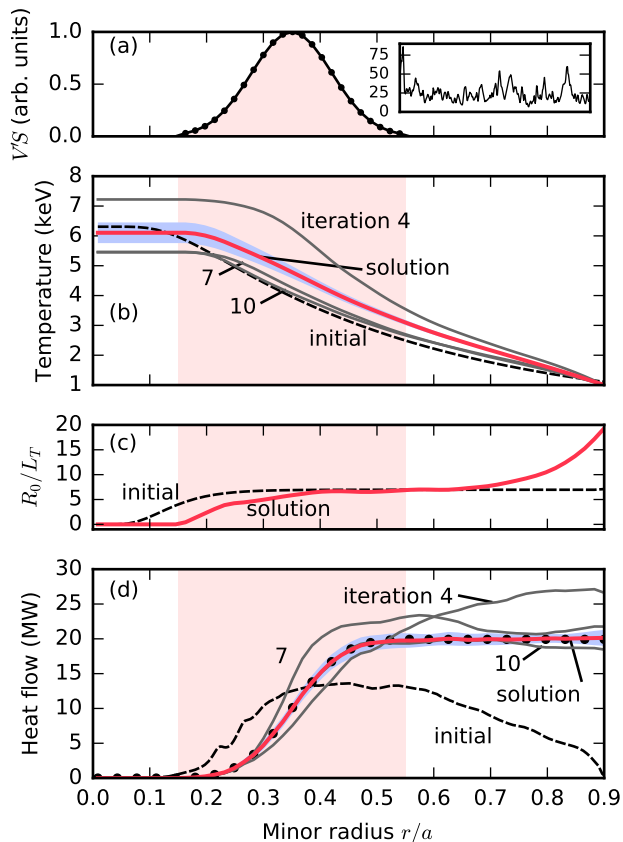
FIG. 2. Results from Simulation 2, in the same format as Figure 1. Here, 20 MW of input power is applied. Inset: turbulent heat flux $\hat{q}^t$ (arb. units), centered at $r/a = 0.4$ and radially averaged over 10 gyroradii, shown as a function of time over 400 R_0/v_{T_i} (simulated with the steady-state solution).

IV. DISCUSSION

The energy confinement time can be estimated from $\tau_E \approx W_i/P_{\mathrm{in}}$, where W_i is the stored energy in the ions in the steady-state solution. We find $\tau_E \approx 26$ ms for Simulation 1 and $\tau_E \approx 55$ ms for Simulation 2. The ratio of confinement times does not follow gyro-Bohm scaling $\tau/\tau_{\mathrm{turb}} \sim \rho_*^{-2}$ exactly, as is known for small tokamaks [11, 42].

The total time used for gyrokinetic simulation is a useful measure of computational cost and efficiency of the multiscale method. In both cases, this expenditure is 3100 R_0/v_{T_i}, which corresponds to a physical time of 23 ms for both Simulations 1 and 2. Hence, our calculation has successfully bridged the timescale separation. Additionally, opportunity exists to optimize performance through parameter tuning.

Fluctuations in the turbulent flux create a tradeoff between computational expediency and precision. An example of the fluctuations can be seen in Fig. 2(a) (inset);
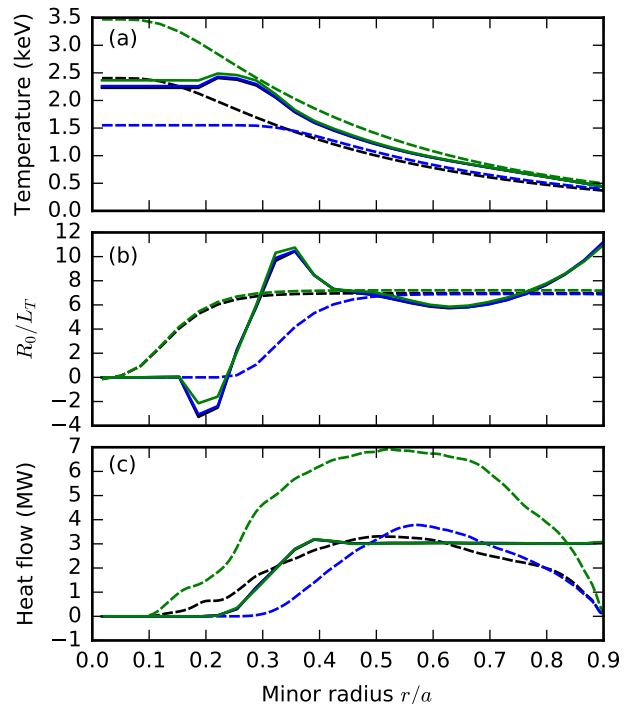

FIG. 3. Comparing results when starting from different initial conditions, using the parameters of Simulation 1. The initial temperature profile from Simulation 1 is used, along with two other distinct temperature profiles, distinguished by color. Shown are the initial (dashed) and final, averaged over 20 iterations (solid) (a) temperature profile, (b) normalized temperature gradient, and (c) heat flow. For the solutions, the solid curves are overlapping and are difficult to distinguish.

the largest bursts can be three times as large as the mean. These fluctuations, inherent to the equations regardless of the method of simulation, set a noise floor and limit the achievable precision. The variance of the fluctuations can be reduced through greater averaging of the flux, which in practice is accomplished by increasing the simulation time per iteration or decreasing the relaxation parameter α, either of which increases the computational cost. It is worth noting that uncertainties in experimental measurements and limitations in physics models may set bounds on the precision desirable in practice, as additional precision requires greater computational resources while providing little added value beyond a certain point.

In summary, we have demonstrated multiple-timescale coupling of global gyrokinetic simulations to a transport solver. Using the LoDestro method, which prescribes an algorithm for solving the nonlinear implicit timestep of the transport equation, we have solved for the steady-state temperature profile. This method holds potential to play an important role in the integration of nonlocal gyrokinetic turbulence simulations into predictive, whole-device models. Extensions of this work that allow for evolution of ion pressure, electron pressure, and density simultaneously are underway.

[1] J. A. Krommes, Annu. Rev. Fluid Mech. **44**, 175 (2012).

[2] A. J. Brizard and T. S. Hahm, Rev. Mod. Phys. **79**, 421 (2007).

[3] X. Garbet, Y. Idomura, L. Villard, and T. Watanabe, Nucl. Fusion **50**, 043002 (2010).

[4] F. Jenko, W. Dorland, M. Kotschenreuther, and B. N. Rogers, Phys. Plasmas **7**, 1904 (2000).

[5] W. Dorland, F. Jenko, M. Kotschenreuther, and B. N. Rogers, Phys. Rev. Lett. **85**, 5579 (2000).

[6] J. Candy and R. Waltz, J. Comput. Phys. **186**, 545 (2003).

[7] Y. Chen and S. E. Parker, J. Comput. Phys. **220**, 839 (2007).

[8] V. Grandgirard, Y. Sarazin, P. Angelino, A. Bottino, N. Crouseilles, G. Darmet, G. Dif-Pradalier, X. Garbet, P. Ghendrih, S. Jolliet, G. Latu, E. Sonnendrcker, and L. Villard, Plasma Phys. Controlled Fusion **49**, B173 (2007).

[9] C. S. Chang, S. Ku, P. H. Diamond, Z. Lin, S. Parker, T. S. Hahm, and N. Samatova, Phys. Plasmas **16**, 056108 (2009).

[10] Z. Lin and T. S. Hahm, Phys. Plasmas **11**, 1099 (2004).

[11] B. F. McMillan, X. Lapillonne, S. Brunner, L. Villard, S. Jolliet, A. Bottino, T. Görler, and F. Jenko, Phys. Rev. Lett. **105**, 155001 (2010).

[12] T. Görler, X. Lapillonne, S. Brunner, T. Dannert, F. Jenko, S. K. Aghdam, P. Marcus, B. F. McMillan, F. Merz, O. Sauter, D. Told, and L. Villard, Phys. Plasmas **18**, 056103 (2011).

[13] T. Rhodes, C. Holland, S. Smith, A. White, K. Burrell, J. Candy, J. DeBoo, E. Doyle, J. Hillesheim, J. Kinsey, G. McKee, D. Mikkelsen, W. Peebles, C. Petty, R. Prater, S. Parker, Y. Chen, L. Schmitz, G. Staebler, R. Waltz, G. Wang, Z. Yan, and L. Zeng, Nucl. Fusion **51**, 063022 (2011).

[14] F. Jenko, D. Told, T. Görler, J. Citrin, A. B. Navarro, C. Bourdelle, S. Brunner, G. Conway, T. Dannert, H. Doerk, D. Hatch, J. Haverkort, J. Hobirk, G. Hogeweij, P. Mantica, M. Pueschel, O. Sauter, L. Villard, E. Wolfrum, and the ASDEX Upgrade Team, Nucl. Fusion **53**, 073003 (2013).

[15] C. Holland, J. Kinsey, J. DeBoo, K. Burrell, T. Luce, S. Smith, C. Petty, A. White, T. Rhodes, L. Schmitz, E. Doyle, J. Hillesheim, G. McKee, Z. Yan, G. Wang, L. Zeng, B. Grierson, A. Marinoni, P. Mantica, P. Snyder, R. Waltz, G. Staebler, and J. Candy, Nucl. Fusion **53**, 083027 (2013).

[16] T. Görler, A. E. White, D. Told, F. Jenko, C. Holland, and T. L. Rhodes, Phys. Plasmas **21**, 122307 (2014).

[17] J. Citrin, F. Jenko, P. Mantica, D. Told, C. Bourdelle, R. Dumont, J. Garcia, J. Haverkort, G. Hogeweij, T. Johnson, M. Pueschel, and J.-E. contributors, Nucl. Fusion **54**, 023008 (2014).

[18] A. Bañón Navarro, T. Happel, T. Görler, F. Jenko, J. Abiteboul, A. Bustos, H. Doerk, and D. Told, Phys. Plasmas **22**, 042513 (2015).

[19] D. R. Ernst, K. H. Burrell, W. Guttenfelder, T. L. Rhodes, A. M. Dimits, R. Bravenec, B. A. Grierson, C. Holland, J. Lohr, A. Marinoni, G. R. McKee, C. C. Petty, J. C. Rost, L. Schmitz, G. Wang, S. Zemedkun, and L. Zeng, Phys. Plasmas **23**, 056112 (2016).

[20] N. T. Howard, C. Holland, A. E. White, M. Greenwald, J. Candy, and A. J. Creely, Phys. Plasmas **23**, 056109 (2016).

[21] Y. Idomura, Phys. Plasmas **21**, 022517 (2014).

[22] Y. Idomura and M. Nakata, Phys. Plasmas **21**, 020706 (2014).

[23] G. Dif-Pradalier, G. Hornung, P. Ghendrih, Y. Sarazin, F. Clairet, L. Vermare, P. H. Diamond, J. Abiteboul, T. Cartier-Michaud, C. Ehrlacher, D. Estève, X. Garbet, V. Grandgirard, O. D. Gürcan, P. Hennequin, Y. Kosuga, G. Latu, P. Maget, P. Morel, C. Norscini, R. Sabot, and A. Storelli, Phys. Rev. Lett. **114**, 085004 (2015).

[24] K. Kim, C. S. Chang, J. Seo, S. Ku, and W. Choe, Phys. Plasmas **24**, 012306 (2017).

[25] K. Kim, J.-M. Kwon, C. S. Chang, J. Seo, S. Ku, and W. Choe, Phys. Plasmas **24**, 062302 (2017).

[26] R. E. Waltz, Fusion Sci. Technol. **48**, 1051 (2005).

[27] M. Shimada, D. Campbell, V. Mukhovatov, M. Fujiwara, N. Kirneva, K. Lackner, M. Nagami, V. Pustovitov, N. Uckan, J. Wesley, N. Asakura, A. Costley, A. Donn, E. Doyle, A. Fasoli, C. Gormezano, Y. Gribov, O. Gruber, T. Hender, W. Houlberg, S. Ide, Y. Kamada, A. Leonard, B. Lipschultz, A. Loarte, K. Miyamoto, V. Mukhovatov, T. Osborne, A. Polevoi, and A. Sips, Nucl. Fusion **47**, S1 (2007).

[28] J. Candy, C. Holland, R. E. Waltz, M. R. Fahey, and E. Belli, Phys. Plasmas **16**, 060704 (2009).

[29] M. Barnes, I. G. Abel, W. Dorland, T. Görler, G. W. Hammett, and F. Jenko, Phys. Plasmas **17**, 056109 (2010).

[30] K. Ida, Z. Shi, H. Sun, S. Inagaki, K. Kamiya, J. Rice, N. Tamura, P. Diamond, G. Dif-Pradalier, X. Zou, K. Itoh, S. Sugita, O. Grcan, T. Estrada, C. Hidalgo, T. Hahm, A. Field, X. Ding, Y. Sakamoto, S. Oldenbrger, M. Yoshinuma, T. Kobayashi, M. Jiang, S. Hahn, Y. Jeon, S. Hong, Y. Kosuga, J. Dong, and S.-I. Itoh, Nucl. Fusion **55**, 013022 (2015).

[31] R. Waltz, J. Candy, F. Hinton, C. Estrada-Mila, and J. Kinsey, Nucl. Fusion **45**, 741 (2005).

[32] R. E. Waltz, G. M. Staebler, and W. M. Solomon, Phys. Plasmas **18**, 042504 (2011).

[33] H. Sugama and W. Horton, Phys. Plasmas **4**, 405 (1997).

[34] H. Sugama and W. Horton, Phys. Plasmas **5**, 2560 (1998).

[35] I. G. Abel, G. G. Plunk, E. Wang, M. Barnes, S. C. Cowley, W. Dorland, and A. A. Schekochihin, Rep. Prog. Phys. **76**, 116201 (2013).

[36] J. Crotinger, L. LoDestro, L. Pearlstein, A. Tarditi, T. Casper, and E. Hooper, *CORSICA: A comprehensive simulation of toroidal magnetic-fusion devices. Final report to the LDRD Program* (1997).

[37] A. Shestakov, R. Cohen, J. Crotinger, L. LoDestro, A. Tarditi, and X. Xu, J. Comput. Phys. **185**, 399 (2003).

[38] https://github.com/LLNL/tango.

[39] T. Görler, X. Lapillonne, S. Brunner, T. Dannert, F. Jenko, F. Merz, and D. Told, J. Comput. Phys. **230**, 7053 (2011).

[40] www.genecode.org.

[41] X. Lapillonne, S. Brunner, T. Dannert, S. Jolliet, A. Marinoni, L. Villard, T. Görler, F. Jenko, and F. Merz, Phys. Plasmas **16**, 032308 (2009).

[42] J. Candy, R. E. Waltz, and W. Dorland, Phys. Plasmas **11**, L25 (2004).